EPFL

Thèse n° 7220

Fourier Sampling in Signal Processing and Numerical Linear Algebra

Présentée le 27 août 2020

à la Faculté informatique et communications
Laboratoire de théorie du calcul 4
Programme doctoral en informatique et communications

pour l'obtention du grade de Docteur ès Sciences

par

Amir ZANDIEH

Acceptée sur proposition du jury

Prof. R. Urbanke, président du jury
Prof. M. Kapralov, directeur de thèse
Prof. E. Price, rapporteur
Prof. R. Pagh, rapporteur
Prof. O. Svensson, rapporteur

École polytechnique fédérale de Lausanne

2020

An expert is a person who has made all the mistakes
that can be made in a very narrow field.
— Niels Bohr

To my parents, Sakineh and Sadegh

Acknowledgements

First and foremost, I would like to express my deepest appreciation to my advisor, Michael Kapralov, for his tremendous guidance, continuous support, and incredible patience throughout my PhD. I could not have imagined having a better advisor and mentor than Michael. Regardless of his busy schedule, he always made time to supervise me on a daily basis. He was the first and best teacher I had during my academic studies. It would not have been possible for me to do the PhD without the support and guidance from him. I strive to use the valuable lessons I learned from Michael during my PhD, throughout my life.

Besides my advisor, I would like to thank the rest of the jury members of my thesis; Prof. Eric Price (University of Texas at Austin), Prof. Rasmus Pagh (IT University of Copenhagen), Prof. Ola Svensson (EPFL Lausanne), and the jury president Prof. Rüdiger Urbanke (EPFL Lausanne) for agreeing to serve on the committee and the insightful discussion we had during the oral exam.

I am immensely thankful to Prof. David Woodruff for hosting me as a visiting researcher at Carnegie Mellon University. During my time at CMU, he always provided insightful discussions about my research. I earnestly hope to have the chance to contribute more to the field of "randomized numerical linear algebra" in the future through active collaborations with David.

My sincere appreciation goes to Pauline. She was always there with a fabulous attitude whenever I needed help. She is definitely one of the most supportive and positive people I have ever met. I am also grateful to all my fellow-labmates at EPFL for the stimulating discussions and for all the fun we have had in the past five years.

Above all, none of this would have been possible without the enormous encouragement and support from my family, particularly, my parents Sakineh and Sadegh. Despite living a thousand miles apart, they have never stopped giving me their unconditional love and support. Dear Mom and Dad, I owe you everything.

Lausanne, August 12, 2020 **A. Z.**

Abstract

This thesis focuses on developing efficient algorithmic tools for processing large datasets. In many modern data analysis tasks, the sheer volume of available datasets far outstrips our abilities to process them. This scenario commonly arises in tasks including parameter tuning of machine learning models – e.g., *Google Vizier* (Golovin et al., 2017) and training neural networks (Goel et al., 2017). These tasks often require solving *numerical linear algebraic problems* on large matrices, making the classical primitives prohibitively expensive. Hence, there is a crucial need to develop efficient algorithms that can *compress* the available datasets, while preserving their essential structure. In other important settings, even collecting the input dataset is extremely expensive, making it vital to design *optimal data sampling* strategies. This is common in applications such as MRI acquisition (Lustig et al., 2007), and spectrum sensing in cognitive radio networks (Ghasemi and Sousa, 2008). The fundamental questions above are often dual to each other, and hence can be addressed using the same set of core techniques. Indeed, exploiting *structured Fourier sparsity* is a recurring source of efficiency in this thesis, leading to both fast methods for numerical linear algebra and sample efficient data acquisition schemes.

One of the main results that we present in this thesis is the first *Sublinear-time Model-based Sparse Fourier Transform* algorithm. Our algorithm achieves a nearly optimal sample complexity for recovery of every signal whose Fourier transform is well approximated by a small number of blocks (e.g., such structure is common in spectrum sensing). Our method matches in sublinear time the result of Baraniuk et al. (2010a), which started the field of model-based compressed sensing. Another highlight of this thesis includes the first *Dimension-independent Sparse FFT* algorithm that, computes the Fourier transform of a sparse signal in sublinear runtime in any dimension. This is the first algorithm that just like the FFT of Cooley and Tukey is dimension independent and avoids the curse of dimensionality inherent to all previously known techniques. Finally, we give a *Universal Sampling Scheme* for the reconstruction of structured Fourier signals from continuous measurements. Our approach matches the classical results of (Slepian and Pollak, 1961; Landau and Pollak, 1961, 1962) on the reconstruction of bandlimited signals via *Prolate Spheroidal Wave Functions* and extends these results to a wide class of Fourier structure types.

Besides having classical applications in signal processing and data analysis, Fourier techniques have been at the core of many machine learning primitives such as *Kernel Matrix Approximation*. The second half of this thesis is dedicated to finding compressed and low-rank

representations to kernel matrices, which are the primary means of computation with large kernel matrices for kernel methods in machine learning. We build on techniques in *Fourier analysis* and achieve spectral approximation guarantees to the Gaussian kernel using an optimal number of samples, significantly improving upon the classical *Random Fourier Features* of Rahimi and Recht (2008). Finally, we present a nearly-optimal *Oblivious Subspace Embedding* for high-degree Polynomial kernels which leads to nearly-optimal oblivious embeddings of the high-dimensional Gaussian kernel. This is the first result that does not suffer from an exponential loss in the degree of the polynomial kernel or the dimension of the input point set, providing exponential improvements over the prior work, including the *TensorSketch* (Pham and Pagh, 2013) and application of the celebrated *Fast Multipole Method* of Greengard and Rokhlin (1986) to kernel approximation problem.

Keywords: Sparse Fourier Transform, Numerical Linear Algebra, Block Sparsity, Kernel Low-Rank Approximation, Random Fourier Features, Sketching, Oblivious Subspace Embedding

Zusammenfassung

Diese These konzentriert sich auf die Entwicklung effizienter Algorithmen zur Verarbeitung großer Datenmengen. Bei vielen modernen Datenanalyse Aufgaben übertrifft das breite Volumen der verfügbaren Datensätze unsere Fähigkeit sie zu verarbeiten, bei weitem. Dieses Szenario tritt häufig bei Aufgaben auf, sowie beispielsweise bei der Parametereinstellung von Modellen für maschinelles Lernen – e.g., *Google Vizier* (Golovin et al., 2017), Training neuronaler Netze (Goel et al., 2017). Diese Aufgaben erfordern häufig die Lösung *numerischer linearer algebraischer Probleme*, wodurch die klassischen Algorithmen unerschwinglich langsam werden. Daher besteht ein entscheidender Bedarf an der Entwicklung effizienter Algorithmen, mit denen die verfügbaren breiten Datensätze *komprimiert* werden können, während der Lösungsraum unserer Probleme annähernd unverändert bleibt. In anderen wichtigen Situationen ist sogar das Sammeln des Eingabedatensatzes extrem teuer, weshalb es wichtig ist, *optimale Datenstichprobenstrategien* zu entwickeln. Dies ist häufig der Fall bei Anwendungen wie der MRT-Erfassung (Lustig et al., 2007), der "Spectrum Sensing" in kognitiven Funknetzen (Ghasemi and Sousa, 2008). Tatsächlich sind die oben genannten Herausforderungen bei der *optimalen Probensammlung* und der *optimalen Datenkomprimierung* häufig doppelt miteinander verbunden und können daher mit denselben Techniken angegangen werden. In der Tat ist die Nutzung der *Fourier-Struktur* eine wiederkehrende Quelle der Effizienz in unseren Arbeiten, die sowohl zu schnellen Methoden für die numerische lineare Algebra als auch zu effizienten Datenerfassungsschemata führt.

Eines der Hauptergebnisse, das wir in dieser Arbeit präsentieren, ist der erste *sublineare Zeit modellbasierte Sparse-Fourier-Transformation*. Unser Algorithmus erreicht eine nahezu optimale Abtastkomplexität für die Wiederherstellung eines Signals, dessen Fourier-Transformation durch eine kleine Anzahl von Blöcken gut angenähert wird. Eine solche Struktur ist beispielsweise bei der Spectrum Sensing in kognitiven Funknetzen üblich. Unsere Methode entspricht in sublinearer Zeit dem Ergebnis von Baraniuk et al. (2010a), mit dem das Gebiet der modellbasierten komprimierten Abtastung begonnen wurde. Ein weiteres Highlight dieser These ist der erste *dimensionsunabhängige Sparse-FFT-Algorithmus*, der die Fourier-Transformation eines Sparse-Signals in sublinearer Laufzeit in einer beliebigen Dimension berechnet. Dies ist der erste spärliche FFT-Algorithmus, der genau wie der FFT-Algorithmus von Cooley und Tukey dimensionsunabhängig ist und den Fluch der Dimensionalität vermeidet, der in allen bisher bekannten Techniken innewohnt. Schließlich geben wir ein *universelles Abtastschema* für die Rekonstruktion von Signalen mit strukturierten Fourier-Transformationen aus kontinuierlichen Messungen. Unser Ansatz entspricht den klassischen Ergebnissen von

(Slepian and Pollak, 1961; Landau and Pollak, 1961, 1962) zur Rekonstruktion bandbegrenzter Signale über *Prolate Spheroidal Wave Functions* und erweitert diese Ergebnisse auf eine breite Klasse von Fourier-Strukturtypen (wobei die klassische Einstellung bandbegrenzter Signale ein Sonderfall ist).

Neben den klassischen Anwendungen der Fourier-Transformation in der Signalverarbeitung und Datenanalyse standen Fourier-Techniken im Mittelpunkt vieler Methoden des maschinellen Lernens wie der *Kernel-Matrix-Approximation*. Die zweite Hälfte dieser Arbeit befasst sich mit der Suche nach komprimierter und niedrigrangiger Näherung für Kernelmatrizen. Solche komprimierten Darstellungen sind das primäre Berechnungsmittel mit breiten Kernelmatrizen für Kernelmethoden beim maschinellen Lernen. Wir bauen auf Techniken der Fourier-Analyse auf und erzielen spektrale Approximationsgarantien für den Gaußschen Kernel unter Verwendung einer optimalen Anzahl von Stichproben, wodurch die klassischen *Random-Fourier-Features* (eine der beliebtesten Methoden für Kernel-Approximationen mit niedrigem Rang) erheblich verbessert werden. Schließlich präsentieren wir eine nahezu optimale *Oblivious-Subspace-Embedding* für den hochgradigen Polynomkern, die zu nahezu optimalen Einbettungen des hochdimensionalen Gaußschen Kerns führt. Dies ist das erste Ergebnis, das nicht unter einem exponentiellen Verlust des Grads des Polynomkerns oder der Dimension des Eingabepunktsatzes leidet und exponentielle Verbesserungen gegenüber früheren Arbeiten bietet, einschließlich *TensorSketch* (Pham and Pagh, 2013) und der Anwendung der berühmten *Fast-Multipole-Methode* von Greengard and Rokhlin (1986).

Schlüsselwörter: Sparse-Fourier-Transformation, Numerische lineare Algebra, Block Schwachbesetztheit, Kernel niedrigrangiger Näherung, Random-Fourier-Features, Sketching, Oblivious-Subspace-Embedding

Contents

Introduction

The design of efficient algorithms has always been at the core of computer science. The unprecedented growth of scientific and Internet datasets over the past few decades necessitates computational efficiency more than ever. Various notions of efficiency are of interest to the computer science community, one of the most important of which is the computational efficiency as time is a precious resource. Nonetheless, in general, efficiency can be measured with respect to other metrics. In this thesis, we focus on developing "efficient" algorithmic tools and techniques (in a broad sense) for solving large-scale problems in Signal Processing and Machine Learning with provable, worst-case guarantees on their performance. In light of this objective, we address the following two fundamental challenges in the field of big data analysis, each targeting a different notion of efficiency:

Sample-efficient Estimation. In many modern data analysis tasks, collecting the input dataset is extremely expensive, making it vital to design *optimal data sampling* strategies. This is common in applications such as MRI acquisition (Lustig et al., 2007; Pruessmann et al., 1999), and spectrum sensing in cognitive radio networks (Ghasemi and Sousa, 2008; Baraniuk et al., 2010a; Hassanieh et al., 2014; Lin et al., 2011) as well as hyperparameter tuning of machine learning models, where every data sample requires running a computationally expensive training process – e.g., (Golovin et al., 2017; Hazan et al., 2018). These problems typically reduce to learning an unknown function given some *prior belief* about the structure of the function using the optimal number of samples. By far one of the most common ways to impose structure on a function is through restricting its *Fourier transform*. Therefore, in many scenarios, there is a crucial need to devise optimal sampling schemes for learning Fourier structured functions – e.g., *Fourier sparsity* is a commonly studied structure.

Time/Memory-efficient Estimation. In other important data analysis settings, the sheer volume of the available datasets far outstrips our abilities to process them. This scenario commonly arises in tasks including parameter tuning of machine learning models (e.g., *Google Vizier* (Golovin et al., 2017), a popular Google internal service for performing black-box optimization), training neural networks (Goel et al., 2017), and Bayesian optimization (where one seeks to maximize cumulative reward by balancing exploration and exploitation (Sutton and Barto, 2018; Robbins, 1952; Srinivas et al., 2010)). These tasks often require solving

numerical linear algebraic problems on typically large matrices, making the classical primitives prohibitively expensive. Hence, there is a crucial need for developing efficient algorithms that can *compress* the unwieldy datasets available, while approximately preserving their essential structure.

The *optimal sample collection* and *optimal data compression* challenges above are often dual to each other, and hence can be addressed using the same set of core techniques. Indeed, exploiting *structured Fourier sparsity* is a recurring source of efficiency in this thesis, leading to both fast methods for numerical linear algebra and sample efficient data acquisition schemes. Fourier analysis is a major technical tool in many areas of computer science: applications in signal processing (e.g. compression schemes such as JPEG and MPEG) are motivated by the fact that Fourier transform concentrates the energy of natural signals, making them compressible, in numerical linear algebra (e.g., sketching) it is applied as a fast method of achieving anti-concentration of energy, and in many other areas it is a tool of choice due to the availability of fast algorithms (*Fast Fourier Transform, or FFT*). In machine learning, the *Random Fourier Features* method (Rahimi and Recht, 2008, test of time award winner at NeurIPS'17) has been the de facto standard method for designing low-rank approximations to *kernel matrices*. In all the above-mentioned applications, one usually applies the Fourier transform to *Fourier-sparse* signals that furthermore often satisfy additional structural assumptions. This thesis focuses on principled ways of exploiting structured sparsity in the above settings.

Following this, we summarize the contributions of this thesis and give a general outline of the remaining chapters. We present our results in separate self-contained chapters that can be read independently.

Overview of our Contributions

In this section, we give an overview of our results and techniques. The central focus of this thesis is *Fourier Sampling*. First, we define the Fourier transform in the discrete regime, and then we formulate our main contributions using this definition.

Definition 1 (Discrete Fourier transform)**.** For every positive integers n and d and any function $x : \mathbb{Z}_n^d \to \mathbb{C}$ the Fourier transform of x is defined as the function $\widehat{x} : \mathbb{Z}_n^d \to \mathbb{C}$ given by,

$$\widehat{x}_f \stackrel{\text{def}}{=} \sum_{t \in \mathbb{Z}_n^d} x_t e^{-2\pi i \frac{f^\top t}{n}} \quad \text{for every } f \in \mathbb{Z}_n^d.$$

In any applications of the Discrete Fourier Transform, the input signal x often satisfies sparsity or approximate sparsity constraints: the Fourier transform $\widehat{x}$ of x has a small number of coefficients k or is close to a signal with a small number of coefficients. To be precise, a signal x is called k-sparse if its Fourier transform is supported on at most k frequencies, i.e., $\|\widehat{x}\|_0 \leq k$. In many applications, we often wish to reconstruct a sparse signal from a small number of (inverse) Fourier samples. This is known as the *sparse recovery* problem, which

aims at recovering a k-sparse signal x using a number of time domain samples that is nearly linear in k. It was first shown in the celebrated work of Candes, Romberg, and Tao (Candès et al., 2006a) that one can recover Fourier sparse signals seamlessly in any dimension with high probability using optimal number of $O(k \log N)$ samples, where $N = n^d$ is the size of signal x. Their method works by observing the values of signal x on $O(k \log N)$ i.i.d. uniform samples from $\mathbb{Z}_n^d$ and solving an ℓ_1 minimization program.

Underlying the analysis of Candès et al. (2006a) for showing that their sample optimal exact sparse recovery method works, is the so-called *Uncertainty Principle* which says that it is impossible to localize a signal both in time and frequency domains at the same time. Many different versions of this principle have been studied since it was introduced by Heisenberg (1930). The classical discrete uncertainty principle (Donoho and Stark, 1989), which has found deep applications in signal processing, says that for any signal x and its Fourier transform $\widehat{x}$ the following holds,

$$\|x\|_0 \cdot \|\widehat{x}\|_0 \geq N.$$

Additionally, it was shown by the seminal series of works of (Candès and Tao, 2006, 2005; Donoho, 2006; Candès et al., 2006b) that *robust recovery* of approximately sparse signals is possible by solving the same ℓ_1 optimization provided that the corresponding Fourier measurement matrix satisfies a *uniform* notion of uncertainty principle known as the so-called *Restricted Isometry Property (RIP)*. In other words, if the Fourier matrix restricted to the rows corresponding to the time domain samples satisfies the RIP of order $4k$, then the ℓ_1 minimization method stably recovers any approximately k-sparse signal with probability one. A matrix $A \in \mathbb{C}^{q \times n}$ is said to satisfy the restricted isometry property (RIP) of order k with constant δ for some $\delta \in (0,1)$, if for every vector $y \in \mathbb{C}^n$ with $\|y\|_0 \leq k$, it holds that

$$(1-\delta)\|y\|_2^2 \leq \|Ay\|_2^2 \leq (1+\delta)\|y\|_2^2.$$

By utilizing the results on RIP of Fourier matrices (Rudelson and Vershynin, 2008; Cheraghchi et al., 2013; Bourgain, 2014; Haviv and Regev, 2017) it follows that picking $O\left(k \log^2 k \log N\right)$ rows of the Fourier matrix independently and uniformly at random results in a matrix that satisfies the RIP of order k. Hence, robust recovery of an (approximately) Fourier sparse signal is possible by observing its values on a subset of $O\left(k \log^2 k \log N\right)$ random points.

The ℓ_1 optimization and, in general, every currently known method that works based on unstructured random samples has a slow superlinear, $\Omega(N)$, runtime. On the other hand, an exciting line of work on the Sparse Fourier Transform problem (*Sparse FFT*), developed in the Theoretical Computer Science community, has been focused on achieving sublinear, i.e., $o(N)$, runtime by using structured samples with limited independence. Very efficient algorithms for the Sparse FFT problem have been developed in the literature. Initially, these results were established for the purpose of demonstrating learnability of Boolean functions. The first algorithms of this type were designed for the special case of Hadamard transform, i.e., the Fourier transform over the binary hypercube (Goldreich and Levin, 1989; Kushilevitz and Mansour, 1993).

Shortly thereafter, algorithms for the Sparse FFT in dimension one were proposed as well (Mansour, 1995; Gilbert et al., 2002; Akavia et al., 2003; Gilbert et al., 2005). All of those algorithms are randomized and have a constant probability of success and the most efficient one (Gilbert et al., 2005) computes the Sparse FFT in time $k \cdot (\log N)^{O(1)}$. Over the past few years, the topic has been the subject of extensive research, resulting in many new developments including the first deterministic algorithms (Iwen, 2010; Akavia, 2010), as well as the first *practical* algorithm that outperforms the optimized software packages such as FFTW (Hassanieh et al., 2012c). The fastest known algorithm for robust recovery of (approximately) k-sparse signals in dimension one is due to Hassanieh et al. (2012b), and has a runtime of $O\big(k \log N \log(N/k)\big)$. The recent works of Indyk et al. (2014); Kapralov (2016) also show how to achieve the optimal sample complexity of $O(k \log N)$, in linear time, or in time $k \cdot (\log N)^{O(1)}$ at the expense of poly(log log N) factors. More recently, Kapralov (2017) showed that it is possible to achieve the optimal sample complexity of $O(k \log N)$ in time $k \cdot (\log N)^{O(1)}$ for robust recovery of one dimensional signals. Moreover, recent works of Price and Song (2015); Chen et al. (2016) have considered this problem in the continuous setting as well.

The main idea behind the aforementioned algorithms is designing Fourier measurements of the signal x that can "hash" the dominant positions of $\widehat{x}$ into a number of "buckets". The number of buckets is chosen to be a constant factor larger than the sparsity k to ensure that a constant fraction of the large elements of $\widehat{x}$ are isolated. The idea of hashing is implemented via filtering: one designs a filter that approximates a bucket in the Fourier domain and additionally, has a small support in time domain (the support size of the filter directly influences the sample complexity, thus, it is critical to optimize this factor). The *Aliasing Filter*, whose time domain representation is the Dirac comb, has optimal performance from the point of view of uncertainty principle, i.e., the support size of an aliasing filter in time domain is exactly equal to the number of buckets it implements in the Fourier domain. However, designing a Sparse FFT algorithm based on aliasing filters is difficult because frequencies belonging to the same multiplicative subgroup get hashed together if such filters are used (we will elaborate more on this later), making it impossible to reason about isolation of individual frequencies. Therefore, all prior works mentioned in the above paragraph use filters that ensure isolation of frequencies, but at the expense of having suboptimal performance in terms of uncertainty principle. For example, the filters constructed in Hassanieh et al. (2012b), have a time domain support that is larger than k (the ideal support size) by a factor of $\Theta(\log N)$ in dimension one. This effect is even more pronounced in higher dimensions, resulting in a $\log^d N$ loss in sample and time complexities, causing all prior techniques to suffer from the curse of dimensionality. One of our contributions is resolving this curse of dimensionality.

On a technical level, the curse of dimensionality that appears in all prior Sparse FFT techniques is connected to a similar phenomenon observed in the state-of-the-art methods for the *kernel approximation* problem. In particular, the application of the celebrated *Fast Multipole Method* of Greengard and Rokhlin (1986) to kernel approximation problem, while very efficient in low dimensions, suffers from an exponential loss in the dimensionality of the input datasets. A notable contribution of this thesis is the first kernel approximation method that lifts the curse

of dimensionality for the Polynomial and Gaussian kernels.

In what follows we outline our main contributions.

Sample-optimal Model-based Fourier Transform in Sublinear-time. In Chapter 1 we present the first algorithm that achieves a nearly optimal sample complexity for recovery of signals whose Fourier transforms are well approximated by a small number of blocks (such structure is common in, for example, DNA microarrays (Stojnic et al., 2009), as well as spectrum sensing in cognitive radio networks (Ghasemi and Sousa, 2008; Lin et al., 2011)). The *Block-sparse* model was introduced by the seminal work of Baraniuk et al. (2010a), which started the field of model-based compressed sensing. Although the *model-based* framework of Baraniuk et al. (2010a) achieves an optimal sample complexity with non-adaptive algorithms, their result uses Gaussian measurements which are very slow to work with. Surprisingly, in stark contrast to the extensive work on exploiting model-based sparsity with general linear measurements, for over a decade, there has been no existing algorithm exploiting such structure using Fourier measurements. In this thesis we present the first such algorithm that runs in sublinear time. Moreover, we show, in Chapter 1, that *adaptivity is essential* for obtaining the sample complexity gains due to exploiting structure beyond sparsity, answering the important open question on *Model-based Restricted Isometry Property* of Fourier measurement matrices negatively.

To be precise, a signal $y : \mathbb{Z}_n \to \mathbb{C}$ is called (k_0, k_1)*-block sparse* if its support is the union of k_0 intervals of length k_1. We give an algorithm that for any input signal $x : \mathbb{Z}_n \to \mathbb{C}$ outputs a signal $\widehat{\chi} : \mathbb{Z}_n \to \mathbb{C}$ such that,

$$\left\|\widehat{x} - \widehat{\chi}\right\|_2 \leq (1+\epsilon) \min_{\widehat{y} \text{ is } (k_0,k_1)\text{-block sparse}} \|\widehat{x} - \widehat{y}\|_2$$

using $O^*\left(k_0 k_1 + k_0 \log k_0 \log n\right)$ accesses to x and similar runtime up to $\log n$ factors, which matches the result of Baraniuk et al. (2010a), in sublinear time.

The number of permitted sparsity patterns by the block-sparse model is far lower than the number of arbitrary sparsity patterns, $\binom{n}{k_0 k_1}$. One might hope that this restriction would translate into a considerably stronger uncertainty principle for such structured signals, hence, leading to a reduction in the sample complexity of recovery algorithms. However, this view is not correct 'as is' because, the energy of a block-sparse signal that contains one block of length k_1 in the Fourier domain is very far from being uniform in the time domain and, in fact, can be fully concentrated on a $\frac{1}{k_1}$ fraction of the time domain. Note that the uncertainty principle intuitively works because a k-sparse signal is a superposition of k (pairwise orthogonal) single tone waves in the time domain where the energy distribution of each single tone is completely flat over the time, hence, their superposition cannot be too concentrated. On the other hand, a (k_0, k_1)-block sparse signal is a superposition of k_0 blocks, where the energy distribution of each block in the time domain is very far from being flat, thus, one cannot hope that a significantly stronger uncertainty principle would naively hold for block-sparse signals.

The blocks of a block-sparse signal can behave very differently in time domain: the energy of some blocks could be spread out over the time domain while the others are completely concentrated in a specific region. So, by sampling the signal at any specific region of the time domain, we can learn a nontrivial amount of information about a block only if a nontrivial fraction of the energy of that block lies in that region. Consequently, in order to learn the location of any specific block using optimal number of samples, our sampling distribution must favor the time domain regions that contain a large fraction of the energy of that block. In order to recover the signal, we need to distribute our samples such that for almost every block we have enough number of samples from the regions that contain a nontrivial fraction of its energy, and we have to do so *without even knowing the energy distribution of the blocks in time domain*. We tackle this challenge by designing a novel energy-based importance sampling scheme.

Dimension-independent Sparse Fourier Transform. Another highlight of this thesis, presented in Chapter 2, includes the first *Dimension-independent Sparse FFT* algorithm that computes the Fourier transform of a sparse signal in sublinear runtime in any dimension. The state-of-the-art in high dimensional sparse Fourier transform poses an interesting conundrum: algorithms with optimal runtime are known for one dimensional discrete Fourier transform, see (Hassanieh et al., 2012b), but in the multi-dimensional setting, the runtime scales exponentially in the dimension. Given that FFT itself is dimension-insensitive, this strongly suggests that exciting new algorithmic techniques can be developed for the high-dimensional version of the problem. In this thesis, we design the first approach to high dimensional Sparse FFT that does not suffer from the curse of dimensionality.

More precisely, for any signal $x : \mathbb{Z}_n^d \to \mathbb{C}$ whose Fourier transform is k-sparse, i.e., $\|\widehat{x}\|_0 \leq k$, our algorithm computes $\widehat{x}$ using $k^3 \cdot \mathrm{poly}\left(\log N\right)$ runtime and samples in any dimension d. This is the first Sparse FFT algorithm that just like the FFT of Cooley and Tukey is dimension independent, and avoids the curse of dimensionality inherent to all previously known techniques. Recall that prior works on Sparse FFT have primarily focused on efficiently implementing hashing-based ideas using Fourier measurements. The filters used for emulating the hashing of the Fourier spectrum are, however, significantly suboptimal in high dimension causing the prior techniques to suffer from the curse of dimensionality.

The main technical innovation that allows us to avoid exponential dependence on the dimension is a new family of filters for isolating a subset of frequencies in the Fourier domain for a sparse signal $\widehat{x}$ using few samples in time domain. We refer to the family of filters as "adaptive aliasing filters". The aliasing filters have optimal performance in any dimension from the point of view of the uncertainty principle. However, the presence of multiplicative subgroups in $\mathbb{Z}_n^d$ has been a hurdle in designing algorithms using these filters in the past because frequencies that belong to the same subgroup get hashed together if such filters are used. This is precisely the reason we needed to design an adaptive approach to find frequencies that can be isolated cheaply using aliasing filters, leading to the first dimension-independent Sparse FFT

algorithm.

Near-optimal Recovery of Signals with Simple Fourier Transforms. We set forth in Chapter 3, a *Universal Sampling Scheme* for the reconstruction of structured signals from continuous Fourier measurements. Classically, the most standard example of such Fourier structured signals is the class of *Bandlimited* signals, meaning that the Fourier transform of the signal of interest is only non-zero on frequencies contained in a bounded interval around the origin. The seminal line of work by Slepian and Pollak (1961); Landau and Pollak (1961, 1962), who presented a set of explicit basis functions known as the *prolate spheroidal wave functions*, can be used to optimally interpolate bandlimited functions over a finite interval in time domain. It follows from the properties of these basis functions that the energy of a bandlimited function is more concentrated towards the endpoints of an interval in the time domain than the middle of the interval. Therefore, to efficiently interpolate a bandlimited function on an interval, one needs to employ *non-uniform* sampling schemes that sample near the edges of the interval more densely than the middle. Recall that this is in sharp contrast with the discrete setting, where a uniform sampling distribution works optimally for recovering Fourier sparse signals. Rokhlin et al. (2001) combines the prolate spheroidal wave functions with numerical quadrature methods to obtain a very efficient method for bandlimited reconstruction.

While the aforementioned line of work is beautiful and powerful, in today's world, we are interested in far more general Fourier structures than bandlimited functions. For example, there is a widespread interest in Fourier-sparse or Multiband (Block-sparse) signals. More generally, in statistical signal processing, a prior distribution which is specified by some probability measure is assumed on the frequency content of the signals of interest. However, despite its clear importance, the problem of fitting continuous signals under the most common Fourier transform priors is not theoretically well understood, even 50 years after the groundbreaking work of Slepian, Landau, and Pollak on the bandlimited problem.

In this thesis, we address this problem far more generally. Formally, our algorithm solves the following function fitting problem.

Problem 1 (Recovery of signals with constrained Fourier transforms)**.** Given a known probability measure μ on $\mathbb{R}$, define the inverse Fourier transform of a function $g(\xi)$ with respect to μ as $\left[\mathscr{F}_\mu^* g\right](t) \stackrel{\text{def}}{=} \int_{\mathbb{R}} g(\xi) e^{2\pi i \xi t}\, d\mu(\xi)$ for any $t \in [0,T]$. Suppose we can observe $y(t)+n(t)$, where $n(t)$ is some fixed noise function and $y = \mathscr{F}_\mu^* x$ for some frequency domain function $x(\xi)$. Then, for error parameter ϵ, our goal is to recover an approximation $\tilde{y}$ by observing $y(t)+n(t)$ on a small number of points $t_1, t_2, \cdots t_q \in [0,T]$ satisfying,

$$\|y-\tilde{y}\|_T^2 \le \epsilon \|x\|_\mu^2 + C\|n\|_T^2,$$

where $\|x\|_\mu^2 \stackrel{\text{def}}{=} \int_{\mathbb{R}} |x(\xi)|^2\, d\mu(\xi)$ and $\|z\|_T^2 \stackrel{\text{def}}{=} \frac{1}{T}\int_0^T |z(t)|^2 dt$, so that $\|y-\tilde{y}\|_T^2$ is our mean squared error and $\|n\|_T^2$ is the mean squared noise level. $C \ge 1$ is a fixed positive constant.

Surprisingly, we show that a universal non-uniform sampling strategy can be used to solve the above problem using a number of samples proportional to the *statistical dimension* of the allowed power spectrum μ. We prove that, in nearly all settings, this natural measure tightly characterizes the sample complexity of signal reconstruction. Our approach matches the classical results of (Slepian and Pollak, 1961; Landau and Pollak, 1961, 1962) on the reconstruction of bandlimited signals, and extends these results to a wide class of Fourier structure types (with the classical setting of bandlimited signals being a special case).

Our main technical tool is an extension of a powerful result from the randomized numerical linear algebra literature to continuous Fourier operators: every matrix contains a small subset of columns that span a near-optimal low-rank approximation to that matrix. By extending this result to continuous linear operators, we prove that the smoothness of a signal whose Fourier transform has $\|x\|_\mu^2$ bounded is tightly captured by the smoothness of an $O(s_{\mu,\varepsilon})$-Fourier sparse function, where $s_{\mu,\varepsilon}$ is the statistical dimension of class of functions that are constrained by prior μ. This lets us reduce every Fourier prior to Fourier sparsity, thus, Problem 1 reduces to recovery of Fourier sparse signals using a small number of continuous measurements in the time domain. Now, a continuous version of the uncertainty principle plays a central role. By this principle, a function that is $O(s_{\mu,\varepsilon})$-sparse in the Fourier domain cannot be too spiky in the time domain, therefore, $\widetilde{O}\left(s_{\mu,\varepsilon}\right)$ samples is sufficient for interpolating such functions to high precision, hence, this recovery problem can be solved using $\widetilde{O}\left(s_{\mu,\varepsilon}\right)$ samples.

Near-optimal Random Fourier Features. Besides classical applications of the Fourier transform in signal processing and data analysis, Fourier techniques have been at the core of many machine learning primitives such as *Kernel Matrix Approximation*. The *Random Fourier Features* method (Rahimi and Recht, 2008, test of time award winner at NeurIPS'17) has been the de facto standard method for solving the kernel matrix approximation problem via Fourier sampling. The second half of this thesis (Chapters 4 and 5) is mainly focused on finding compressed and low-rank representations to kernel matrices, which are the most fundamental and widely used structured objects in kernel-based learning.

More precisely, given a kernel matrix $K \in \mathbb{R}^{n\times n}$, defined as $K_{i,j} \stackrel{\text{def}}{=} k(\mathbf{x}_i, \mathbf{x}_j)$ for an arbitrary dataset $\mathbf{x}_1, \mathbf{x}_2, \cdots \mathbf{x}_n \in \mathbb{R}^d$ and kernel function $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, we aim at finding a low-rank matrix $Z \in \mathbb{R}^{m\times n}$ with small m that *spectrally approximates* the regularized kernel $K + \lambda I$ – that is[1],

$$(1-\epsilon)(K+\lambda I) \preceq Z^\top Z + \lambda I \preceq (1+\epsilon)(K+\lambda I), \quad \text{for } \epsilon, \lambda \geq 0. \tag{1}$$

Such compressed representations are the primary means of computation with large kernel matrices for kernel methods. In Chapter 4, we build on techniques in Fourier analysis and achieve spectral approximation guarantees to the Gaussian kernel using a (near) optimal number of samples, m, which is nearly linear in the *statistical dimension* of the kernel matrix – i.e., $m = O(s_\lambda \log s_\lambda)$, where $s_\lambda \stackrel{\text{def}}{=} \operatorname{tr}\left(K(K+\lambda I)^{-1}\right)$, in any constant dimension, significantly

[1] For any two Hermitian matrices of order n, A and B we say that $A \preceq B$ if $B - A$ is positive semi-definite

improving upon the classical random Fourier features of Rahimi and Recht.

The Fourier features method is essentially a consequence of the Bochner's theorem: for any shift invariant kernel function $k(\cdot)$, the kernel matrix can be decomposed as $K = \mathscr{F}^* \Sigma \mathscr{F}$ where $\mathscr{F} : \mathbb{R}^n \to L_2\left(\mathbb{R}^d\right)$ is the d-dimensional Fourier transform operator restricted to columns corresponding to the dataset $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$, that is, $[\mathscr{F}\boldsymbol{\alpha}](\boldsymbol{\xi}) \stackrel{\text{def}}{=} \sum_{j=1}^n \alpha_j \cdot e^{-2\pi i \boldsymbol{\xi}^\top \mathbf{x}_j}$ for every $\boldsymbol{\alpha} \in \mathbb{R}^n$ and $\boldsymbol{\xi} \in \mathbb{R}^d$, and $\Sigma : L_2\left(\mathbb{R}^d\right) \to L_2\left(\mathbb{R}^d\right)$ is a diagonal operator whose diagonal is a probability distribution corresponding to the Fourier transform of the kernel function $k(\cdot)$, that is, $\left[\Sigma g\right](\boldsymbol{\xi}) \stackrel{\text{def}}{=} \widehat{k}(\boldsymbol{\xi}) \cdot g(\boldsymbol{\xi})$ for every $g \in L_2\left(\mathbb{R}^d\right)$ and $\boldsymbol{\xi} \in \mathbb{R}^d$. The classical Fourier features method (Rahimi and Recht, 2008) approximates the kernel matrix $K = \mathscr{F}^* \Sigma \mathscr{F}$ by sampling frequencies (diagonal entries of the operator Σ) according to the probability distribution $\widehat{k}(\boldsymbol{\xi})$. However, we prove that this sampling distribution is significantly suboptimal for achieving the spectral approximation guarantee of (1).

It is a well-known fact in randomized numerical linear algebra that sampling the columns of a discrete matrix according to the *ridge leverage scores* distribution works near-optimally for achieving bounds of type (1) (with a number of samples that is nearly proportional to the statistical dimension of the matrix). First, we extend the ridge leverage scores sampling results to continuous operators. Then, we propose a simple distribution that tightly upper bounds the ridge leverage scores of the operator $\Sigma^{1/2}\mathscr{F}$, and sample the Fourier features from it. Intuitively, the leverage score of a frequency $\boldsymbol{\xi}$, measures how much the energy of a function that is in the column span of the restricted Fourier operator $\Sigma^{1/2}\mathscr{F}$ can be concentrated on frequency $\boldsymbol{\xi}$. Our main technical contribution is to reformulate the leverage score as the solution of two dual optimization problems. Then, by carefully constructing suitable test functions that have (nearly) optimal performance from the uncertainty principle viewpoint, we are able to give tight upper and lower bounds on the ridge leverage scores, and correspondingly on the sampling performance of classical and our modified Fourier features sampling.

Oblivious Sketching of High-degree Polynomial Kernels. Finally, another highlight of this thesis is a nearly-optimal *Oblivious Subspace Embedding* for the high-degree Polynomial kernel. The state-of-the-art method on sketching the Polynomial kernel is the *TensorSketch* (Pham and Pagh, 2013; Avron et al., 2014), whose runtime and sketching dimension scale exponentially in the degree of the kernel. In Chapter 5, we circumvent this exponential loss by proposing the first oblivious sketching solution for the Polynomial kernel that satisfies the spectral approximation guarantee of (1), with a target dimension that is linear in the *statistical dimension* of the kernel matrix and polynomial in the degree of the kernel. We prove that our sketch leads to the first nearly-optimal oblivious embedding of the high-dimensional Gaussian kernel with a target dimension that is linear in the statistical dimension of the kernel matrix. This is the first result that does not suffer from an exponential loss in the degree of the polynomial kernel or the dimension of the input point set, providing exponential improvements over the prior work, including the application of the celebrated *Fast Multipole Method* of Greengard and Rokhlin (1986) to kernel approximation problem.

Organization. The remaining chapters of this thesis contain our full results on the above-mentioned problems. Each chapter has its own introduction that is the basis for the corresponding problem of interest and covers the prior works and approaches that were previously used in the literature to tackle it. For completeness, we also include an appendix for each chapter, if need be, to present the basic tools that we borrowed from the literature to design and analyze our algorithms.

1 Sample-optimal Model-based Fourier Transform in Sublinear-time

This chapter is based on a joint work with Volkan Cevher, Michael Kapralov, and Jonathan Scarlett . It has been accepted to the 49th Annual ACM SIGACT Symposium on Theory of Computing (Cevher et al., 2017, STOC).

1.1 Introduction

The discrete Fourier transform (DFT) is one of the most important tools in modern signal processing, finding applications in audio and video compression, radar, geophysics, medical imaging, communications, and many more. The best known algorithm for computing the DFT of a general signal of length n is the Fast Fourier Transform (FFT), taking $O(n \log n)$ time, which matches the trivial $\Omega(n)$ lower bound up to a logarithmic factor.

In recent years, significant attention has been paid to exploiting *sparsity* in the signal's Fourier spectrum, which is naturally the case for numerous of the above applications. By sparse, we mean that the signal can be well-approximated by a small number of Fourier coefficients. Given this assumption, the computational lower bound of $\Omega(n)$ no longer applies. Indeed, the DFT can be computed in *sublinear time*, while using a sublinear number of samples in the time domain (Gilbert et al., 2014, 2008).

The problem of computing the DFT of signals that are approximately sparse in Fourier domain has received significant attention in several communities. The seminal works of (Candès and Tao, 2006; Rudelson and Vershynin, 2008) in *compressive sensing* first showed that only $k \log^{O(1)} n$ samples in time domain suffice to recover a length n signal with at most k nonzero Fourier coefficients. A different line of research on the *Sparse Fourier Transform* (sparse FFT), with origins in computational complexity and learning theory, has resulted in algorithms that use $k \log^{O(1)} n$ samples and $k \log^{O(1)} n$ runtime (i.e., the runtime is *sublinear* in the length of the input signal). Many such algorithms have been proposed in the literature (Goldreich and Levin, 1989; Kushilevitz and Mansour, 1993; Mansour, 1995; Gilbert et al., 2002; Akavia et al., 2003; Gilbert et al., 2005; Iwen, 2010; Akavia, 2010; Hassanieh et al., 2012c,b,a; Lawlor et al.,

2013; Pawar and Ramchandran, 2013; Heider et al., 2013; Indyk et al., 2014; Indyk and Kapralov, 2014; Boufounos et al., 2015; Kapralov, 2016; Price and Song, 2015; Chen et al., 2016; Kapralov, 2017; Nakos et al., 2019); we refer the reader to the recent surveys of Gilbert et al. (2014, 2008) for a more complete overview.

The best known runtime for computing the k-sparse FFT is due to Hassanieh et al. (2012b), and is given by $O(k\log n\log(n/k))$, asymptotically improving upon the FFT for all $k = o(n)$. The recent works of Indyk et al. (2014); Kapralov (2016) also show how to achieve the sample complexity of $O(k\log n)$, which is essentially optimal, in linear time, or in time $k\log^{O(1)} n$ at the expense of poly($\log\log n$) factors. More recently, Kapralov (2017) showed that it is possible to achieve the optimal sample complexity of $O(k\log n)$ in time $k\log^{O(1)} n$. Intriguingly, the aforementioned algorithms are all *non-adaptive*. That is, these algorithms do not exploit existing samples in guiding the selection of the new samples to improve approximation quality. In the same setting, it is also known that adaptivity cannot improve the sample complexity by more than an $O(\log\log n)$ factor (Hassanieh et al., 2012b).

Despite the significant gains permitted by sparsity, designing an algorithm for handling *arbitrary* sparsity patterns may be overly generic; in practice, signals often exhibit more specific sparsity structures. A common example is *block sparsity*, where significant coefficients tend to cluster on known partitions, as opposed to being unrestricted in the signal spectrum. Other common examples include tree-based sparsity, group sparsity, and dispersive sparsity (Baraniuk et al., 2010a; Baldassarre et al., 2016; Bach, 2010).

Such structured sparsity models can be captured via the *model-based* framework of Baraniuk et al. (2010a), where the number of sparsity patterns may be far lower than $\binom{n}{k}$. For the compressive sensing problem, this restriction has been shown to translate into a reduction in the sample complexity, even with non-adaptive algorithms. Specifically, one can achieve a sample complexity of $O(k+\log|\mathcal{M}|)$ with dense measurement matrices based on the Gaussian distribution, where $\mathcal{M}$ is the set of permitted sparsity patterns. Reductions in the sample complexity with other types of measurement matrices, e.g., sparse measurement matrices based on expanders, are typically less significant (Indyk and Razenshteyn, 2013; Bah et al., 2014). Other benefits of exploiting model-based sparsity include faster recovery and improved noise robustness (Baraniuk et al., 2010a; Bah et al., 2014).

Surprisingly, in stark contrast to the extensive work on exploiting model-based sparsity with general linear measurements, there are no existing sparse FFT algorithms exploiting such structure. In this chapter we present the first such algorithm (Cevher et al., 2017), focusing on the special case of block sparsity. Even for this relatively simple sparsity model, achieving the desiderata turns out to be quite challenging, needing a whole host of new techniques, and intriguingly, requiring *adaptivity* in the sampling scheme.

To clarify our contributions, we describe our model and the problem statement in more detail.

Model and Basic Definitions: The Fourier transform of a signal $X \in \mathbb{C}^n$ is denoted by $\widehat{X}$, and defined as

$$\widehat{X}_f = \frac{1}{n} \sum_{i \in [n]} X_i \omega_n^{-ft}, \quad f \in [n],$$

where ω_n is the n-th root of unity. With this definition, Parseval's theorem takes the form $\|X\|^2 = n\|\widehat{X}\|_2^2$.

We are interested in computing the Fourier transform of signals that, in frequency domain, are well-approximated by a *block sparse* signal with k_0 blocks of width k_1, formalized as follows.

Definition 1.1.1 (Block sparsity)**.** Given a sequence $X \in \mathbb{C}^n$ and an even block width k_1, the *j-th interval* is defined as $I_j = \big((j-1/2)k_1, (j+1/2)k_1\big] \cap \mathbb{Z}$ for $j \in \big[\frac{n}{k_1}\big]$, and we refer to $\widehat{X}_{I_j}$ as the *j-th block*. We say that a signal is *(k_0, k_1)-block sparse* if it is non-zero within at most k_0 of these intervals.

Block sparsity is of direct interest in several applications (Baraniuk et al., 2010a,b); we highlight two examples here: (i) In spectrum sensing, cognitive radios seek to improve the utilization efficiency in a sparsely used wideband spectrum. In this setting, the frequency bands being detected are non-overlapping and predefined. (ii) Audio signals often contain blocks corresponding to different noises at different frequencies. Such blocks may be *non-uniform*, and can be modeled by the (k, c) *model* in which k coefficients are arbitrarily spread across c different clusters. It was argued in (Cevher et al., 2009) that any signal from the (k, c) model is also $(3c, k/c)$-block sparse in the uniform model.

Our goal is to output a list of frequencies and values estimating $\widehat{X}$, yielding an ℓ_2-distance to $\widehat{X}$ not much larger than that of the best (k_0, k_1)-block sparse approximation. Formally, we say that an output signal $\widehat{X}'$ satisfies the *ℓ_2/ℓ_2 block-sparse recovery guarantee* if

$$\|\widehat{X} - \widehat{X}'\|_2 \leq (1+\epsilon) \min_{\widehat{Y} \text{ is } (k_0,k_1)\text{-block sparse}} \|\widehat{X} - \widehat{Y}\|_2$$

for an input parameter $\epsilon > 0$.

The sample complexity and runtime of our algorithm are parameterized by the *signal-to-noise ratio* (SNR) of the input signal, defined as follows.

Definition 1.1.2 (Tail noise and signal-to-noise ratio (SNR))**.** We define the *tail noise level* as

$$\mathrm{Err}(\widehat{X}, k_0, k_1) := \min_{\substack{S \subset [\frac{n}{k_1}] \\ |S| = k_0}} \sum_{j \in [\frac{n}{k_1}] \setminus S} \|\widehat{X}_{I_j}\|_2^2, \tag{1.1}$$

and its normalized version as $\mu^2 := \frac{1}{k_0}\mathrm{Err}^2(\widehat{X}, k_0, k_1)$, representing the average noise level per block. The *signal-to-noise ratio* is defined as $\mathrm{SNR} := \frac{\|\widehat{X}\|^2}{\mathrm{Err}^2(\widehat{X}, k_0, k_1)}$.

Throughout the paper, we assume that both n and k_1 are powers of two. For n, this is a standard assumption in the sparse FFT literature. As for k_1, the assumption comes without

too much loss of generality, since one can always round the block size up to the nearest power of two and then cover the original k_0 blocks with at most $2k_0$ larger blocks, thus yielding a near-identical recovery problem other than a possible increase in the SNR. We also assume that $\frac{n}{k_1}$ exceeds a large absolute constant; if this fails, our stated scaling laws can be obtained using the standard FFT.

We use $O^*(\cdot)$ notation to hide $\log\log\mathrm{SNR}$, $\log\log n$, and $\log\frac{1}{\epsilon}$ factors. Moreover, to simplify the notation in certain lemmas having free parameters that will be set in terms of ϵ, we assume throughout the chapter that $\epsilon = \Omega\big(\frac{1}{\mathrm{polylog}\, n}\big)$, and hence $\log\frac{1}{\epsilon} = O(\log\log n)$. This is done purely for convenience, and since the dependence on ϵ is not our main focus; the precise expressions with $\log\frac{1}{\epsilon}$ factors are easily inferred from the proofs. Similarly, since the low-SNR regime is not our key focus, we assume that $\mathrm{SNR} \geq 2$, and thus $\log\mathrm{SNR}$ is positive.

Contributions: We proceed by informally stating our main result; a formal statement is given in Section 1.5.2.

Theorem 1.1.1. (Upper bound – informal version) *There exists an adaptive algorithm for approximating the Fourier transform with (k_0, k_1)-block sparsity that achieves the ℓ_2/ℓ_2 guarantee for any constant ϵ, with a sample complexity of $O^*\left(\left(k_0k_1 + k_0\log(1+k_0)\log n\right)\log\mathrm{SNR}\right)$, and a runtime of $O^*\left(k_0k_1\log^3 n\log\mathrm{SNR}\right)$.*

Note that while we state the result for $\epsilon = \Theta(1)$ here, the dependence on this parameter is explicitly shown in the formal version.

The sample complexity of our algorithm *strictly improves* upon the sample complexity of $O(k_0k_1\log n)$ (essentially optimal under the standard sparsity assumption) when $\log(1+k_0)\log\mathrm{SNR} \ll k_1$ and $\log\mathrm{SNR} \ll \log n$ (e.g., $\mathrm{SNR} = O(1)$).

Our algorithm that achieves the above upper bound crucially uses adaptivity. This is in stark contrast with the standard sparse FFT, where we know how to achieve the optimal $O(k\log n)$ bound using non-adaptive sampling (Indyk et al., 2014). While relying on adaptivity can be viewed as a weakness, we provide a lower bound revealing that *adaptivity is essential* for obtaining the above sample complexity gains. We again state an informal version of our lower bound, which is formalized in Section 1.6.

Theorem 1.1.2. (Lower bound – informal version) *Any non-adaptive sparse FFT algorithm that achieves the ℓ_2/ℓ_2 sparse recovery guarantee with (k_0, k_1)-block sparsity must use $\Omega\left(k_0k_1\log\frac{n}{k_0k_1}\right)$ samples.*

To the best of our knowledge, these two theorems provide the first results along several important directions, giving **(a)** the first sublinear-time algorithm for model-based compressed sensing; **(b)** the first model-based result with provable sample complexity guarantees in the Fourier setting; **(c)** the first proven gap between the power of adaptive and non-adaptive

sparse FFT algorithms; and **(d)** the first proven gap between the power of structured (Fourier basis) and unstructured (random Gaussian entries) measurement matrices for model-based compressed sensing.

To see that **(d)** is true, note that the sample complexity $O(k_0 \log n + k_0 k_1)$ for block-sparse recovery can be achieved *non-adaptively* using Gaussian measurements (Baraniuk et al., 2010a), but we show that adaptivity is required in the Fourier setting.

Dependence of our results on SNR. The sample complexity and runtime of our upper bound depend logarithmically on the SNR of the input signal. This dependence is common for sparse FFT algorithms, and even for the case of standard sparsity, algorithms avoiding this dependence in the runtime typically achieve a suboptimal sample complexity (Hassanieh et al., 2012c,b). Moreover, all existing sparse FFT lower bounds consider the constant SNR regime, e.g., (Ba et al., 2010; Price and Woodruff, 2011; Hassanieh et al., 2012b).

We also note that our main result, as stated above, assumes that upper bounds on the SNR and the tail noise are known to within a constant factor (in fact, such tightness is not required, but the resulting bound replaces the true values by the assumed values). These assumptions can be avoided at the expense of a somewhat worse dependence on $\log \mathrm{SNR}$, but we present the algorithm in the above form for clarity. The theoretical guarantees for noise-robust compressive sensing algorithms often require similar assumptions (Foucart and Rauhut, 2013).

Our Techniques: At a high level, our techniques can be summarized as follows:

Upper bound. The high-level idea of our algorithm is to *reduce* the (k_0, k_1)-block sparse signal of length n to a number of downsampled $O(k_0)$-sparse signals of length $\frac{n}{k_1}$, and use standard sparse FFT techniques to locate their dominant values, thereby identifying the dominant blocks of the original signal. Once the blocks are located, their values can be estimated using hashing techniques. Despite the high-level simplicity, this is a difficult task requiring a variety of novel techniques, the most notable of which is an adaptive *importance sampling* scheme for allocating sparsity budgets to the downsampled signals. Further details are given in Section 1.2.

Lower bound. Our lower bound for non-adaptive algorithms follows the information-theoretic framework of Price and Woodruff (2011), but uses a significantly different ensemble of *structured* approximately block-sparse signals occupying only a fraction $O\left(\frac{1}{k_0 k_1}\right)$ of the time domain. Hence, whereas the analysis of Price and Woodruff (2011) is based on the difficulty of identifying one of (roughly) $\binom{n}{k}$ sparsity patterns, the difficulty in our setting is in *non-adaptively* finding where the signal is non-zero – one must take enough samples to cover the various possible time domain locations. The details are given in Section 1.6.

Interestingly, our upper bound uses adaptivity to circumvent the difficulty exploited in this lower bounding technique, by *first* determining where the energy lies, and *then* concentrating

the rest of its samples on the "right" parts of the signal.

Notation: For an even number n, we define $[n] := \left(-\frac{n}{2}, \frac{n}{2}\right] \cap \mathbb{Z}$, where $\mathbb{Z}$ denotes the integers. When we index signals having a given length m, all arithmetic should be interpreted as returning values in $[m]$ according to modulo-m arithmetic. For $x, y \in \mathbb{C}$ and $\Delta \in \mathbb{R}$, we write $y = x \pm \Delta$ to denote $|y - x| \leq \Delta$. The support of a vector X is denoted by supp (X). For a number $a \in \mathbb{R}$, we write $|a|_+ := \max\{0, a\}$ to denote the positive part of a.

Organization: The paper is organized as follows. In Section 1.2, we provide an outline of our algorithm and the main challenges involved. We formalize our energy-based importance sampling scheme in Section 1.3, and provide the corresponding techniques for energy estimation in Section 1.4. The block-sparse FFT algorithm and its theoretical guarantees are given in Section 1.5, and the lower bound is presented and proved in Section 1.6. Several technical proofs are relegated to the appendices.

1.2 Overview of the Algorithm

One of our key technical contributions consists of a reduction from the (k_0, k_1)-block sparse recovery problem for signals of length n to $O(k_0)$-sparse recovery on a set of carefully-defined signals of *reduced length* n/k_1, in sublinear time. We outline this reduction below.

A basic candidate reduction to $O(k_0)$-sparse recovery consists of first convolving $\widehat{X}$ with a filter $\widehat{G}$ whose support approximates the indicator function of the interval $(-k_1/2, k_1/2]$, and then considering a new signal whose Fourier transform consists of samples of $\widehat{X} \star \widehat{G}$ at multiples of k_1. The resulting signal $\widehat{Z}$ of length n/k_1 **(a)** naturally represents $\widehat{X}$, as every frequency of this sequence is a (weighted) sum of the frequencies in the corresponding block, and **(b)** can be accessed in time domain using a small number of accesses to X (if G is compactly supported; see below).

This is a natural approach, but its vanilla version does not work: Some blocks in $\widehat{X}$ may entirely cancel out, not contributing to $\widehat{Z}$ at all, and other blocks may add up constructively and contribute an overly large amount of energy to $\widehat{Z}$. To overcome this challenge, we consider not one, but rather $2k_1$ reductions: For each $r \in [2k_1]$, we apply the above reduction to the *shift of X by $r \cdot \frac{n}{2k_1}$ in time domain*, and call the corresponding vector Z^r. We show that all shifts cumulatively capture the energy of X well, and the major contribution of this work is an algorithm for locating the dominant blocks in $\widehat{X}$ from a small number of accesses to the Z^r's (via a novel importance sampling scheme).

Formal definitions: We formalize the above discussion in the following, starting with the notion of a *flat filter* that approximates a rectangle.

Definition 1.2.1 (Flat filter)**.** A sequence $G \in \mathbb{R}^n$ with Fourier transform $\widehat{G} \in \mathbb{R}^n$ symmetric

about zero is called an (n,B,F)-*flat filter* if (i) $\widehat{G}_f \in [0,1]$ for all $f \in [n]$; (ii) $\widehat{G}_f \geq 1 - \left(\frac{1}{4}\right)^{F-1}$ for all $f \in [n]$ such that $|f| \leq \frac{n}{2B}$; and (iii) $\widehat{G}_f \leq \left(\frac{1}{4}\right)^{F-1}\left(\frac{n}{B|f|}\right)^{F-1}$ for all $f \in [n]$ such that $|f| \geq \frac{n}{B}$.

The following lemma, proved in Appendix A.1.1, shows that it is possible to construct such a filter having $O(FB)$ support in time domain.

Lemma 1.2.1. (Compactly supported flat filter) *Fix the power of two integer n, integer $B < n$, and even integer $F \geq 2$. There exists an (n,B,F)-flat filter $\widehat{G} \in \mathbb{R}^n$, which (i) is supported on a length-$O(FB)$ window centered at zero in time domain, and (ii) has a total energy satisfying $\sum_{f\in[n]} |\widehat{G}_f|^2 \leq \frac{3n}{B}$.*

Throughout the paper, we make use of the filter construction from Lemma 1.2.1, except where stated otherwise. To ease the analysis, we assume that G and $\widehat{G}$ are pre-computed and can be accessed in $O(1)$ time. Without this pre-computation, evaluating $\widehat{G}$ is non-trivial, but possible using semi-equispaced Fourier transform techniques (*cf.*, Section 1.4.2).

With the preceding definition, the set of $2k_1$ downsampled signals is given as follows.

Definition 1.2.2 (Downsampling)**.** Given integers (n,k_1), a parameter $\delta \in \left(0, \frac{1}{20}\right)$, and a signal $X \in \mathbb{C}^n$, we say that the set of signals $\left\{Z^r\right\}_{r\in[2k_1]}$ with $Z^r \in \mathbb{C}^{\frac{n}{k_1}}$ is a (k_1,δ)-*downsampling* of X if,

$$Z^r_j = \frac{1}{k_1}\sum_{i\in[k_1]} \left(G\cdot X^r\right)_{j+\frac{n}{k_1}\cdot i},\ j \in \left[\frac{n}{k_1}\right]$$

for an $\left(n, \frac{n}{k_1}, F\right)$-flat filter with $F = 10\log\frac{1}{\delta}$ and support $O\left(F\frac{n}{k_1}\right)$, where we define $X^r_i = X_{i+a_r}$ with $a_r = \frac{nr}{2k_1}$. Equivalently, in frequency domain, this can be written as,

$$\widehat{Z}^r_j = \left(\widehat{X}^r \star \widehat{G}\right)_{jk_1} = \sum_{f\in[n]} \widehat{G}_{f-k_1\cdot j}\widehat{X}_f\omega_n^{a_r\cdot f},\ j \in \left[\frac{n}{k_1}\right] \tag{1.2}$$

by the convolution theorem and the duality of subsampling and aliasing (see Appendix A.3).

By the choice of F, we immediately obtain the following lemma, showing that we do not significantly increase the sample complexity by working with $\{Z^r\}_{r\in[2k_1]}$ as opposed to X itself.

Lemma 1.2.2. (Sampling the downsampling signals) *Let $\{Z^r\}_{r\in[2k_1]}$ be a (k_1,δ)-downsampling of $X \in \mathbb{C}^n$ for some n,k_1,δ. Then for any $i \in [n/k_1]$ and any $r \in [2k_1]$, the entry Z^r_i can be computed in $O\left(\log\frac{1}{\delta}\right)$ time using $O\left(\log\frac{1}{\delta}\right)$ samples of X.*

This idea of using $2k_1$ reductions fixes the above-mentioned problem of constructive and destructive cancellations: The $2k_1$ reduced signals Z^r ($r \in [2k_1]$) cumulatively capture all the energy of X well. That is, while the energy $|\widehat{Z}^r_j|_2^2$ can vary significantly as a function of r, we can tightly control the behavior of the sum $\sum_{r\in[2k_1]} |\widehat{Z}^r_j|_2^2$. This is formalized in the following.

Lemma 1.2.3. (Downsampling properties) *Fix n,k_1, a parameter $\delta \in \left(0,\frac{1}{20}\right)$, a signal $X \in \mathbb{C}^n$, and a (k_1,δ)-downsampling $\{Z^r\}_{r\in[2k_1]}$ of X. The following conditions hold:*

1. *For all* $j \in [\frac{n}{k_1}]$,

$$\frac{\sum_{r\in[2k_1]} |\widehat{Z}_j^r|^2}{2k_1} \geq (1-\delta)\|\widehat{X}_{I_j}\|_2^2 - 3\delta \cdot \left(\|\widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}}\|_2^2 + \delta \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \right).$$

2. *The total energy satisfies* $(1-12\delta)\|\widehat{X}\|_2^2 \leq \frac{\sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2}{2k_1} \leq 6\|\widehat{X}\|_2^2$.

The proof is given in Appendix A.1.2.

Location via sparse FFT: We expect each Z^r ($r \in [2k_1]$) to be approximately $O(k_0)$-sparse, as every block contributes *primarily* to one downsampled coefficient. At this point, a natural step is to run $O(k_0)$-sparse recovery on the signals Z^r to recover the dominant blocks. However, there are too many signals Z^r to consider! Indeed, if we were to run $O(k_0)$-sparse recovery on every Z^r, we would recover the locations of the blocks, but at the cost of $O(k_0 k_1 \log n)$ samples. This precludes any improvement on the vanilla sparse FFT.

It turns out, however, that it is possible to avoid running a k_0-sparse FFT on all $2k_1$ reduced signals, and to instead *allocate budgets* to them, some of which are far smaller than k_0, and some of which may be zero. This will be key in reducing the sample complexity.

Before formally defining budget allocation, we present the following definition and lemma, showing that we can use fewer samples to identify fewer of the dominant coefficients of a signal, or more samples to identify more dominant coefficients.

Definition 1.2.3. *(Covered frequency)* Given an integer m, a frequency component j of a signal $\widehat{Z} \in \mathbb{C}^m$ is called *covered* by budget s in the signal $\widehat{Z}$ if $|\widehat{Z}_j|^2 \geq \frac{\|\widehat{Z}\|_2^2}{s}$.

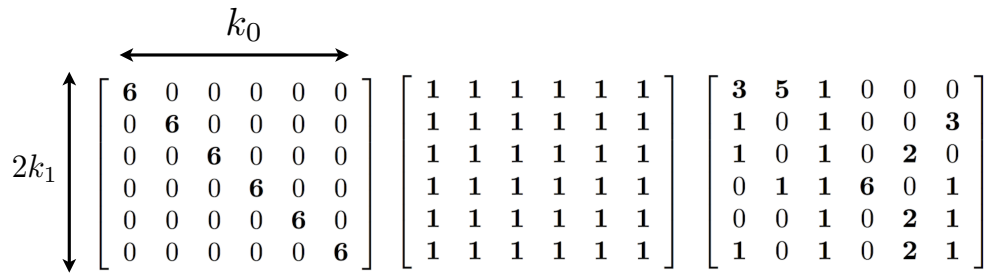
Lemma 1.2.4. (LOCATEREDUCEDSIGNAL guarantees – informal version) *There exists an algorithm such that if a signal $X \in \mathbb{C}^n$, a set of budgets $\{s^r\}_{r\in[2k_1]}$, and a confidence parameter p are given to it as input, then it outputs a list that, with probability at least $1-p$, contains any $j \in [\frac{n}{k_1}]$ that is covered by budget s^r in signal $\widehat{Z}^r$ for some $r \in [2k_1]$, where $\{\widehat{Z}^r\}_{r\in[2k_1]}$ denotes the (k_1,δ)-downsampling of X. Moreover, the list size is $O\big(\sum_{r\in[2k_1]} s^r\big)$, the number of samples that the algorithm takes is $O\big(\sum_{r\in[2k_1]} s^r \log n\big)$, and the runtime is $O\big(\sum_{r\in[2k_1]} s^r \log^2 n\big)$.*[1]

The formal statement and proof are given in Appendix A.8, and reveal that s^r essentially dictates how many buckets we hash $\widehat{Z}^r$ into in order to locate the dominant frequencies (e.g., see (Hassanieh et al., 2012b; Indyk et al., 2014)).

Hence, the goal of budget allocation is to approximately solve the following covering problem:

$$\text{Minimize}_{\{s^r\}_{r\in[2k_1]}} \sum_{r\in[2k_1]} s^r \quad \text{subject to} \quad \sum_{\substack{j \text{ is covered by } s^r \\ \text{in } \widehat{Z}^r \text{ for some } r\in[2k_1]}} \|\widehat{X}_{I_j}\|_2^2 \geq (1-\alpha)\cdot\|\widehat{X}^*\|_2^2, \qquad (1.3)$$

[1] As stated in the formal version, additional terms in the runtime are needed when it comes to subtracting off a current estimate to form a residual signal.



(a) Spiky energies (b) Flat energies (c) Mixed energies

Figure 1.1 – Three hypothetical examples of matrices with (r, j)-th entry given by $|\widehat{Z}_j^r|^2$, i.e., each row corresponds to a single sequence Z^r, but only at the entries corresponding to the k_0 blocks in X.

for a suitable constant $\alpha \in (0,1)$, where s^r is the budget allocated to $\widehat{Z}^r$, and $\widehat{X}^*$ is the best $(k_0.k_1)$-block sparse approximation of $\widehat{X}$. That is, we want to minimize the total budget while accounting for a constant proportion of the signal energy.

Challenges in budget allocation: Allocating the budgets is a challenging task, as each block in the spectrum of the signal may have very different energy concentration properties in time domain, or equivalently, different variations in $|\widehat{Z}_j^r|^2$ as a function of r. To see this more concretely, in Figure 1.1, we show three *hypothetical* examples of such variations, in the case that $k_0 = 2k_1 = 6$ and all of the blocks have equal energy, leading to equal column sums in the matrices.

In the first example, each block contributes to a different Z^r, and thus the blocks could be located by running 1-sparse recovery separately on the $2k_1$ downsampling signals. In stark contrast, in the second example, each block contributes equally to each Z^r, so we would be much better off running k_0-sparse recovery on a single (arbitrary) Z^r. Finally, in the third example, the best budget allocation scheme is completely unclear by inspection alone! We need to design an allocation scheme to handle all of these cases, and to do so *without even knowing the structure of the matrix.*

While the examples in Figure 1.1 may seem artificial, and are not necessarily feasible with the *exact* values given, we argue in Appendix A.7 that situations exhibiting the same general behavior are entirely feasible.

Importance sampling: Our solution is to *sample r values with probability proportional to an estimate of* $\|\widehat{Z}^r\|_2^2$, and sample sparsity budgets from a carefully defined distribution (see Section 1.3, Algorithm 1). We show that sufficiently accurate estimates of $\|\widehat{Z}^r\|_2^2$ for all $r \in [2k_1]$ can be obtained using $O(k_0 k_1)$ samples of X via hashing techniques (*cf.*, Section 1.4); hence, what we are essentially doing is using these samples to determine where most of the energy of the signal is located, and then favoring the parts of the signal that appear to have more energy. This is exactly the step that makes our algorithm adaptive, and we prove that it produces a total budget in (1.3) of the form $O(k_0 \log(1 + k_0))$, on average.

Ideally, one would hope to solve (1.3) using a total budget of $O(k_0)$, since there are only k_0 blocks. However, the $\log(1+k_0)$ factor is not an artifact of our analysis: We argue in Appendix A.7 that very different sampling techniques would be needed to remove it in general. Specifically, we design a signal X for which the *optimal* solution to (1.3) indeed satisfies $\sum_{r\in[2k_1]} s^r = \Omega(k_0 \log(1+k_0))$.

Iterative procedure and updating the residual: The techniques described above allow us to recover a list of blocks that contribute a constant fraction (e.g., 0.9) of the signal energy. We use $O(\log \mathrm{SNR})$ iterations of our main procedure to reduce the SNR to a constant, and then achieve $(1+\varepsilon)$-recovery with an extra "clean-up" step. Most of the techniques involved in this part are more standard, with a notable exception: Running a standard sparse FFT with budgets s^r on the reduced space (i.e., on the vectors Z^r) is not easy to implement in $k_0 k_1 \operatorname{poly}(\log n)$ time when Z^r are the *residual signals*. The natural approach is to subtract the current estimate $\widehat{\chi}$ of $\widehat{X}$ from our samples and essentially run on the residual, but subtraction in $k_0 k_1 \operatorname{poly}(\log n)$ time is not easy to achieve. Our solution crucially relies on a novel block semi-equispaced FFT (see Section 1.4.2), and the idea of letting the location primitives in the reduced space operate using common randomness (see Appendix A.8).

1.3 Location via Importance Sampling

As outlined above, our approach locates blocks by applying standard sparse FFT techniques to the downsampled signals arising from Definition 1.2.2. In this section, we present the techniques for assigning the corresponding sparsity budgets (*cf.*, (1.3)).

We use a novel procedure called *energy-based importance sampling*, which approximately samples r values with probability proportional to $\|\widehat{Z}^r\|^2$. Since these energies are not known exactly, we instead sample proportional to a general vector $\gamma = (\gamma^1, \ldots, \gamma^{2k_1})$, where we think of γ^r as approximating $\|\widehat{Z}^r\|^2$. The techniques for obtaining these estimates are deferred to Section 1.4.

The details are shown in Algorithm 1, where we repeatedly sample from the distribution w_q^r, corresponding to independently sampling r proportional to γ^r, and q from a truncated geometric distribution. The resulting sparsity level to apply to Z^r is selected to be $s^r = 10 \cdot 2^q$.

According to Definition 1.2.3, $s^r = 10 \cdot 2^q$ covers any given frequency j for which $|\widehat{Z}^r_j|^2 \geq \frac{\|\widehat{Z}^r\|_2^2}{10 \cdot 2^q}$. The intuition behind sampling q proportional to 2^{-q} is that this gives a high probability of producing small q values to cover the heaviest signal components, while having a small probability of producing large q values to cover the smaller signal components. We only want to do the latter rarely, since it costs significantly more samples.

We first bound the expected total sum of budgets returned by BUDGETALLOCATION.

Lemma 1.3.1. (BUDGETALLOCATION budget guarantees) *For any integers k_0 and k_1, any non-negative vector $\gamma \in \mathbb{R}_+^{2k_1}$, and any parameters $p \in \left(0, \frac{1}{2}\right)$ and $\delta \in (0,1)$, if the procedure* BUD-

Algorithm 1 Procedure for allocating sparsity budgets to the downsampled signals

procedure BUDGETALLOCATION($\gamma, k_0, k_1, \delta, p$)
  $S \leftarrow \emptyset$
  for $i \in \{1, \ldots, \frac{10}{\delta} k_0 \cdot \ln \frac{1}{p}\}$ **do**
    Sample $(r_i, q_i) \in [2k_1] \times \{1, \ldots, \log_2 \frac{10k_0}{\delta}\}$ with probability $w_q^r = \frac{2^{-q}}{1-\delta/(10k_0)} \frac{\gamma^r}{\|\gamma\|_1}$
    $S \leftarrow S \cup \{(r_i, q_i)\}$
  for $r \in [2k_1]$ **do**
    $q^* \leftarrow \max_{(r,q') \in S} \{q'\}$ ▷ By convention, $\max \emptyset = -\infty$
    $s^r \leftarrow 10 \cdot 2^{q^*}$
  return $s = \{s^r\}_{r \in [2k_1]}$

GETALLOCATION *in Algorithm 1 is run with inputs* $(\gamma, k_0, k_1, \delta, p)$*, then the total sum of returned budgets,* $\{s^r\}_{r \in [2k_1]}$*, satisfies* $\sum_{r \in [2k_1]} s^r \leq 400 \frac{k_0}{\delta} \log_2 \frac{10k_0}{\delta} \ln \frac{1}{p}$ *with probability at least* $1 - p$*. Moreover,* $\max_{r \in [2k_1]} s^r \leq \frac{100k_0}{\delta}$*. The runtime of the procedure is* $O\big(\frac{k_0}{\delta} \log \frac{1}{p} + k_1\big)$*.*

Proof. Each time a new (r, q) pair is sampled, the sum of the s^r values increases by at most $10 \cdot 2^q$, and hence the overall sum of budgets is upper bounded as follows,

$$\sum_{r \in [2k_1]} s^r \leq 10 \cdot \sum_{i=1}^{\frac{10}{\delta} k_0 \cdot \ln \frac{1}{p}} 2^{q_i},$$

where q_i for every $i \in \left\{1, 2, \cdots \frac{10}{\delta} k_0 \cdot \ln \frac{1}{p}\right\}$ are iid copies of a random variable q that takes values in the set $\left\{1, 2, \cdots \log_2 \frac{10k_0}{\delta}\right\}$ according to the probability distribution $\Pr[q = t] = \frac{2^{-t}}{1-\delta/(10k_0)}$. Since 2^{q_i} are positive and bounded random variables with $2^{q_i} \leq \frac{10k_0}{\delta}$ we can apply Chernoff inequality to bound the sum $S := \sum_{i=1}^{\frac{10}{\delta} k_0 \cdot \ln \frac{1}{p}} 2^{q_i}$. First note that,

$$\begin{aligned}
\mathbb{E}[S] &= \mathbb{E}\left[\sum_{i=1}^{\frac{10}{\delta} k_0 \cdot \ln \frac{1}{p}} 2^{q_i}\right] \\
&= \frac{10}{\delta} k_0 \cdot \ln \frac{1}{p} \cdot \mathbb{E}[2^q] \\
&= \frac{\frac{10}{\delta} k_0 \cdot \ln \frac{1}{p} \cdot \log_2 \frac{10k_0}{\delta}}{1 - \delta/(10k_0)}.
\end{aligned}$$

Because the number of iid copies in the summation S is larger than the maximum value of the random variable 2^q by an $\ln \frac{1}{p}$ factor, by Chernoff bound, the sum is no larger than 3 times its expected value with probability at least $1 - p$,

$$\Pr\left[S \geq 3 \cdot \mathbb{E}\left[\sum_{i=1}^{\frac{10}{\delta} k_0 \cdot \ln \frac{1}{p}} 2^{q_i}\right]\right] \leq p.$$

Therefore, $\sum_{r\in[2k_1]} s^r \le 30\cdot\mathbb{E}[S] \le 400\frac{k_0}{\delta}\log_2\frac{10k_0}{\delta}\ln\frac{1}{p}$ with probability at least $1-p$.

Moreover, $\max_{r\in[2k_1]}\mathbf{s}_r = \max 2^{q_i} \le \frac{100k_0}{\delta}$ due to the range of q from which we sample.

Runtime: Note that sampling from w_q^r amounts to sampling q and r values independently, and the corresponding alphabet sizes are $O\big(\log\frac{k_0}{\delta}\big)$ and $O(k_1)$ respectively. The stated runtime follows since we take $O\big(\frac{k_0}{\delta}\log\frac{1}{p}\big)$ samples, and sampling from discrete distributions can be done in time linear in the alphabet size and number of samples (Hagerup et al., 1993). The second loop in Algorithm 1 need not be done explicitly, since the maximum q value can be updated after taking each sample. □

As we discussed in Section 1.2, the $\log k_0$ term in the number of samples would ideally be avoided; however, in Appendix A.7.2, we argue that even the optimal solution to (1.3) can contain such a factor.

We now turn to formalizing the fact that the budgets returned by BUDGETALLOCATION are such that most of the dominant blocks are found. To do this, we introduce the following notion.

Definition 1.3.1 (Active frequencies)**.** Given integers n, k_0, k_1, a signal $X\in\mathbb{C}^n$, a non-negative vector $\gamma\in\mathbb{R}_+^{2k_1}$, a parameter $\delta\in(0,1)$, and a (k_1,δ)-downsampling $\{Z^r\}_{r\in[2k_1]}$ of X, the set of *active frequencies* $\tilde{S}$ is defined as,

$$\tilde{S} = \left\{ j\in\left[\frac{n}{k_1}\right] : \sum_{r\in[2k_1]}\left(|\widehat{Z}_j^r|^2\cdot\frac{\gamma^r}{\|\widehat{Z}^r\|_2^2}\right) \ge \delta\cdot\frac{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2}{k_0}\right\}. \tag{1.4}$$

Observe that if $\gamma^r = \|\widehat{Z}^r\|_2^2$, this reduces to $\sum_{r\in[2k_1]}|\widehat{Z}_j^r|^2 \ge \delta\cdot\frac{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2}{k_0}$, thus essentially stating that the sum of the energies over $r\in[2k_1]$ for the given block index j is an $\Omega\big(\frac{\delta}{k_0}\big)$ fraction of the total energy. Combined with Lemma 1.2.3, this roughly amounts to $\|\widehat{X}_{I_j}\|_2^2$ exceeding an $\Omega\big(\frac{\delta}{k_0}\big)$ fraction of $\|\widehat{X}\|_2^2$.

To formalize and generalize this intuition, the following lemma states that the frequencies within $\tilde{S}$ account for most of the energy in X, as long as each γ^r approximates $\|\widehat{Z}^r\|_2^2$ sufficiently well.

Lemma 1.3.2. (Properties of active frequencies) *Fix integers n, k_0, k_1, a parameter $\delta\in\big(0,\frac{1}{20}\big)$, a signal $X\in\mathbb{C}^n$, and a (k_1,δ)-downsampling $\{Z^r\}_{r\in[2k_1]}$ of X. Moreover, fix an arbitrary set $S^*\subseteq\big[\frac{n}{k_1}\big]$ of cardinality at most $10k_0$, and a vector $\gamma\in\mathbb{R}^{2k_1}$ satisfying,*

$$\sum_{r\in[2k_1]}\left|\|\widehat{Z}_{S^*}^r\|_2^2 - \gamma^r\right|_+ \le 40\delta\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2. \tag{*}$$

Fix the set of active frequencies $\tilde{S}$ according to Definition 1.3.1, and define the signal $\widehat{X}_{\tilde{S}}$ to equal $\widehat{X}$ on all intervals $\{I_j\,;\,j\in\tilde{S}\}$ (see Definition 1.1.1), and zero elsewhere. Then $\|\widehat{X}_{S^*\setminus\tilde{S}}\|_2^2 \le 100\sqrt{\delta}\|\widehat{X}\|_2^2$.

The proof of Lemma 1.3.2 is given in Appendix A.2.

What remains is to show that if j is active, then j is covered by some s^r in $\widehat{Z}^r$ with high constant probability upon running Algorithm 1. This is formulated in the following.

Lemma 1.3.3. (BUDGETALLOCATION covering guarantees) *Fix integers n, k_0, k_1, parameters $\delta \in (0,1)$ and $p \in \left(0, \frac{1}{2}\right)$, a signal $X \in \mathbb{C}^n$, a (k_1, δ)-downsampling $\{Z^r\}_{r\in[2k_1]}$ of X, and a non-negative vector $\gamma \in \mathbb{R}_+^{2k_1}$ satisfying,*

$$\|\gamma\|_1 \le 10 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2. \tag{1.5}$$

Suppose that BUDGETALLOCATION *in Algorithm 1 is run with inputs $(\gamma, k_0, k_1, \delta, p)$, and outputs the budgets $\{s^r\}_{r\in[2k_1]}$. Then for any active j (i.e., $j \in \tilde{S}$ as per Definition 1.3.1), the probability that there exists some $r \in [2k_1]$ such that j is covered by s^r in $\widehat{Z}^r$ is at least $1-p$.*

Proof. Recall from Definition 1.2.3 that if a pair (r,q) is sampled in the first loop of BUDGETALLOCATION, then j is covered provided that $|\widehat{Z}_j^r|^2 \ge \frac{\|\widehat{Z}^r\|_2^2}{10\cdot 2^q}$. We therefore define

$$q_j^r = \min\left\{ q \in \mathbb{Z}_+ : |\widehat{Z}_j^r|^2 \ge \frac{\|\widehat{Z}^r\|_2^2}{10\cdot 2^q} \right\}, \tag{1.6}$$

and note that the event described in the lemma statement is equivalent to some pair (r,q) being sampled with $q_j^r \le q$. Note that due to the range of q from which we sample (*cf.*, Algorithm 1), this can only occur if $q_j^r \le \log_2 \frac{10k_0}{\delta}$.

Taking a single sample: We first compute the probability of being covered for a *single* random sample of (q,r), denoting the corresponding probability by $\Pr_1[\cdot]$. Recalling from line 4 of Algorithm 1 that we sample each (q,r) with probability $w_q^r = \frac{2^{-q}}{1-\delta/(10k_0)} \frac{\gamma^r}{\|\gamma\|_1}$, we obtain

$$\begin{aligned}
\Pr_1[j \text{ covered}] &= \sum_{r\in[2k_1]} \sum_{q_j^r \le q \le \log_2 \frac{10k_0}{\delta}} w_q^r \\
&= \frac{1}{1-\delta/(10k_0)} \sum_{r\in[2k_1]} \sum_{q_j^r \le q \le \log_2 \frac{10k_0}{\delta}} 2^{-q} \frac{\gamma^r}{\|\gamma\|_1} \\
&\ge \sum_{r\in[2k_1]\,:\, q_j^r \le \log_2 \frac{10k_0}{\delta}} 2^{-q_j^r} \frac{\gamma^r}{\|\gamma\|_1} \\
&= \sum_{r\in[2k_1]} 2^{-q_j^r} \frac{\gamma^r}{\|\gamma\|_1} - \sum_{r\in[2k_1]\,:\, q_j^r > \log_2 \frac{10k_0}{\delta}} 2^{-q_j^r} \frac{\gamma^r}{\|\gamma\|_1},
\end{aligned} \tag{1.7}$$

where the third line follows since $\delta/(10k_0) \le 1/10$.

Bounding the first term in (1.7)**:** Observe from (1.6) that $2^{-q_j^r} \ge \frac{1}{2} \frac{10|\widehat{Z}_j^r|^2}{\|\widehat{Z}^r\|_2^2}$, and recall the definition of being active in (1.4). Combining these, we obtain the following when j is active:

$$\sum_{r\in[2k_1]} 2^{-q_j^r}\frac{\gamma^r}{\|\gamma\|_1} \geq \frac{10}{2\|\gamma\|_1}\sum_{r\in[2k_1]}\frac{|\widehat{Z}_j^r|^2\gamma^r}{\|\widehat{Z}^r\|_2^2}$$
$$\geq \frac{10\delta}{2\|\gamma\|_1}\cdot\frac{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2}{k_0} \geq \frac{\delta}{2k_0},$$

where the last inequality follows from the assumption on $\|\gamma\|_1$ in the lemma statement.

Bounding the second term in (1.7)**:** We have

$$\sum_{r\in[2k_1]:\,q_j^r>\log_2\frac{10k_0}{\delta}} 2^{-q_j^r}\frac{\gamma^r}{\|\gamma\|_1} \leq \sum_{r\in[2k_1]}\frac{\delta}{10k_0}\frac{\gamma^r}{\|\gamma\|_1} = \frac{\delta}{10k_0}.$$

Hence, we deduce from (1.7) that $\Pr_1[j \text{ covered}] \geq \frac{2\delta}{5k_0}$.

Taking multiple independent samples: Since the sampling is done $\frac{10}{\delta}k_0\cdot\log\frac{1}{p}$ times independently, the overall probability of an active block j being covered satisfies

$$\Pr[j \text{ covered}] \geq 1-\left(1-\frac{2\delta}{5k_0}\right)^{\frac{10}{\delta}k_0\cdot\log\frac{1}{p}} \geq 1-\exp\left(-4\log\frac{1}{p}\right) \geq 1-p,$$

where we have applied the inequality $1-\zeta \leq e^{-\zeta}$ for $\zeta \geq 0$. □

1.3.1 The Complete Location Algorithm

In Algorithm 2, we give the details of MULTIBLOCKLOCATE, which performs the above-described energy-based importance sampling procedure, runs the sparse FFT location algorithm (see Appendix A.8) with the resulting budgets, and returns a list L containing the block indices that were identified.

MULTIBLOCKLOCATE calls two primitives that are defined later in the paper, but their precise details are not needed in order to understand the location step:

- ESTIMATEENERGIES (see Section 1.4.4) computes a vector γ providing a good approximation of each $\|\widehat{Z}^r\|_2^2$, in the sense of satisfying the preconditions of Lemmas 1.3.2 and 1.3.3;
- LOCATEREDUCEDSIGNALS (see Appendix A.8) accepts the sparsity budgets $\{s^r\}_{r\in[2k_1]}$ and runs a standard s^r-sparse FFT algorithm on each downsampled signal Z^r in order to locate the dominant frequencies.

Note that in addition to X, these procedures accept a second signal $\widehat{\chi}$; this becomes relevant when we iteratively run the block sparse FFT (*cf.*, Section 1.5), representing previously-estimated components that are subtracted off to produce a residual.

Algorithm 2 Multi-block sparse location

procedure MULTIBLOCKLOCATE($X, \widehat{\chi}, n, k_0, k_1, \delta, p$)
  $a_r \leftarrow \frac{nr}{2k_1}$ for each $r \in [2k_1]$
  for $t \in \{1, \ldots, 10\log_2 \frac{1}{p}\}$ **do**
    $\gamma \leftarrow$ ESTIMATEENERGIES($X, \widehat{\chi}, n, k_0, k_1, \delta$) ▷ See Section 1.4.4
    $\boldsymbol{s}^{(t)} \leftarrow$ BUDGETALLOCATION$\left(\gamma, k_0, k_1, \delta, \frac{\delta p}{2\log_2(1/p)}\right)$ ▷ $\gamma = (\gamma^1, \ldots, \gamma^{2k_1})$
  $\boldsymbol{s} \leftarrow \max_t \boldsymbol{s}^{(t)}$ (element-wise with respect to $r \in [2k_1]$)
  $L \leftarrow$ LOCATEREDUCEDSIGNALS($X, \widehat{\chi}, n, k_0, k_1, \mathbf{s}, \delta, \frac{1}{2}\delta p$) ▷ See Appendix A.8
  return L

The required guarantees on LOCATEREDUCEDSIGNALS are given in Lemma 1.2.4 (and more formally in Appendix A.7.2), and in order to prove our main result on MULTIBLOCKLOCATE, we also need the following lemma ensuring that we can compute energy estimates satisfying the preconditions of Lemmas 1.3.2 and 1.3.3; the procedure and proof are presented in Section 1.4.4.

Lemma 1.3.4. (ESTIMATEENERGIES guarantees) *Given integers n, k_0, k_1, signals $X, \widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{\|\widehat{\chi}\|_2}{\mathrm{poly}(n)}$, and parameter $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$, the procedure* ESTIMATEENERGIES($X, \widehat{\chi}, n, k_0, k_1, \delta$) *returns a non-negative vector $\gamma \in \mathbb{R}_+^{2k_1}$ such that, for any given set S^* of cardinality at most $10k_0$, we have the following with probability at least $\frac{1}{2}$:*

1. $\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \right|_+ \leq 40\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;
2. $\|\gamma\|_1 \leq 10 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;

where $\{Z^r\}_{r\in[2k_1]}$ is the (k_1, δ)-downsampling of $X - \chi$ (see Definition 1.2.2).

Moreover, if $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, then the sample complexity is $O\left(\frac{k_0 k_1}{\delta^2} \log \frac{1}{\delta} \log \frac{1}{\delta p}\right)$, and the runtime is $O\left(\frac{k_0 k_1}{\delta^2} \log^2 \frac{1}{\delta} \log^2 n\right)$.

Remark 1.3.1. *The preceding lemma ensures that γ^r's provide good approximations of $\|\widehat{Z}^r\|_2^2$ in a "restricted" and "one-sided" sense, while not over-estimating the total energy by more than a constant factor. Specifically, the first part concerns the energy of $\widehat{Z}^r$ restricted to a fixed set of size $O(k_0)$, and characterizes the extent to which the energies are under-estimated. It appears to be infeasible to characterize over-estimation in the same way (e.g., replacing $|\cdot|_+$ by $|\cdot|$), since several of the samples could be overly large due to spiky noise.*

Remark 1.3.2. *Here and subsequently, the* $\mathrm{poly}(n)$ *lower bounds regarding $(\widehat{X}, \widehat{\chi})$ are purely technical, resulting from extremely small errors when subtracting off $\widehat{\chi}$. See Section 1.4.2 for further details.*

We are now in a position to provide our guarantees on MULTIBLOCKLOCATE, namely, on the behavior of the list size, and on the energy that the components in the list capture. Note that

the output of MULTIBLOCKLOCATE is random, since the same is true of ESTIMATEENERGIES, BUDGETALLOCATION, and LOCATEREDUCEDSIGNALS.

Lemma 1.3.5. (MULTIBLOCKLOCATE guarantees) *Given integers n, k_0, k_1, parameters $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$ and $p \in \left(0, \frac{1}{2}\right)$, and signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\frac{\|\widehat{\chi}\|_2}{\mathrm{poly}(n)}$ interval, the output L of the function* MULTIBLOCKLOCATE$(X, \widehat{\chi}, k_1, k_0, n, \delta, p)$ *satisfies the following properties for any set S^* of cardinality at most $10k_0$, with probability at least $1-p$:*

1. $|L| = O\left(\frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{p} \log^2 \frac{1}{\delta p}\right)$;
2. $\sum_{j \in S^* \setminus L} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \leq 200\sqrt{\delta}\|\widehat{X} - \widehat{\chi}\|_2^2$.

Moreover, if $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, and we have $\delta = \Omega\left(\frac{1}{\mathrm{poly}(\log n)}\right)$ and $p = \Omega\left(\frac{1}{\mathrm{poly}(\log n)}\right)$, then with probabilty at least $1 - \delta p$, the sample complexity is $O^*\left(\frac{k_0}{\delta} \log(1+k_0) \log n + \frac{k_0 k_1}{\delta^2}\right)$, and the runtime is $O^*\left(\frac{k_0 k_1}{\delta^2} \log^2 n + \frac{k_0 k_1}{\delta} \log^3 n\right)$.

Remark 1.3.3. MULTIBLOCKLOCATE *is oblivious to the choice of S^* in this lemma statement.*

Proof. **First claim:** Note that in each iteration of the outer loop when we run BUDGETALLOCATION$(\gamma, k_0, k_1, \delta, \frac{1}{2}\delta p)$, Lemma 1.3.1 implies that for any t, the following holds true,

$$\Pr\left[\sum_{r \in [2k_1]} \mathbf{s}_r^{(t)} = O\left(\frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{\delta p}\right)\right] \geq 1 - \frac{p\delta}{2\log_2(1/p)},$$

where $\mathbf{s}_r^{(t)}$ is the r-th entry of the budget allocation vector $\mathbf{s}^t$ at iteration t. Therefore, by union bound, we have,

$$\Pr\left[\sum_{r \in [2k_1]} \mathbf{s}_r = O\left(\frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{\delta p} \log \frac{1}{p}\right)\right] \geq 1 - 10\log_2 \frac{1}{p} \cdot \frac{p\delta}{2\log_2(1/p)} \geq 1 - 5\delta p. \tag{1.8}$$

We now apply Lemma 1.2.4, which is formalized in Appendix A.8. We set the target success probability to $1 - \frac{1}{2}\delta p$, which guarantees that the size of the list returned by the function LOCATEREDUCEDSIGNALS is $O\left(\sum_{r \in [2k_1]} \mathbf{s}_r \log \frac{1}{\delta p}\right)$. Therefore, by (1.8), we have,

$$\Pr\left[|L| = O\left(\frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{p} \log^2 \frac{1}{\delta p}\right)\right] \geq 1 - p,$$

yielding the first statement of the lemma.

Second claim: Let $X' = X - \chi$, and consider the set S^* given in the lemma statement, and an arbitrary iteration t. By Lemma 1.3.4 in Section 1.4.4 (also stated above), the approximate

energy vector γ in any given iteration of the outer loop satisfies,

$$\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \right|_+ \le 40\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$$
$$\|\gamma\|_1 \le 10 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 \tag{1.9}$$

with probability at least $\frac{1}{2}$. When this is the case, the vector γ meets the requirements of Lemmas 1.3.2 and 1.3.3. That means that the probability of having an energy estimate γ that meets these requirements in at least one iteration is lower bounded by $1-(\frac{1}{2})^{10\log\frac{1}{p}} \ge 1-p/10$. Consider an arbitrary iteration in which the above conditions (1.9) on γ are satisfied. We write,

$$\sum_{j\in S^*\setminus L} \|\widehat{X}'_{I_j}\|_2^2 = \sum_{j\in(S^*\cap\tilde{S})\setminus L} \|\widehat{X}'_{I_j}\|_2^2 + \sum_{j\in S^*\setminus(\tilde{S}\cup L)} \|\widehat{X}'_{I_j}\|_2^2. \tag{1.10}$$

The second term is bounded by,

$$\sum_{j\in S^*\setminus(\tilde{S}\cup L)} \|\widehat{X}'_{I_j}\|_2^2 \le \sum_{j\in S^*\setminus\tilde{S}} \|\widehat{X}'_{I_j}\|_2^2 \le 100\sqrt{\delta}\|\widehat{X}'\|_2^2 \tag{1.11}$$

by Lemma 1.3.2, which uses the first condition on γ in (1.9).

We continue by calculating the expected value of the first term in (1.10) with respect to the randomness of BUDGETALLOCATION and LOCATEREDUCEDSIGNALS:

$$\mathbb{E}\left[\sum_{j\in(S^*\cap\tilde{S})\setminus L} \|\widehat{X}'_{I_j}\|_2^2\right] = \mathbb{E}\left[\sum_{j\in(S^*\cap\tilde{S})} \|\widehat{X}'_{I_j}\|_2^2 \cdot \mathbb{1}[j\notin L]\right]$$
$$\le \sum_{j\in\tilde{S}} \|\widehat{X}'_{I_j}\|_2^2 \cdot \Pr\big[j\notin L\big]. \tag{1.12}$$

We thus consider the probability $\Pr[j\notin L]$ for an arbitrary $j\in\tilde{S}$. If $j\in\tilde{S}$, then by Lemma 1.3.3 and the choice of the failure probability parameter $\frac{1}{2}\delta p$ passed to BUDGETALLOCATION, there is at least one $r\in[2k_1]$ such that j is covered, with probability at least $1-\frac{1}{2}\delta p$. We also know from Lemma 1.2.4 that the failure probability of LOCATEREDUCEDSIGNALS for some covered j is at most $\frac{1}{2}\delta p$. A union bound on these two events gives

$$\Pr\big[j\notin L\big] \le \delta p, \quad \forall j\in\tilde{S}.$$

Hence, we deduce from (1.12) that,

$$\mathbb{E}\left[\sum_{j\in(S^*\cap\tilde{S})\setminus L} \|\widehat{X}'_{I_j}\|_2^2\right] \le \delta p\cdot\|\widehat{X}'\|_2^2,$$

and Markov's inequality gives,

$$\sum_{j\in(S^*\cap\tilde{S})\setminus L} \|\widehat{X}'_{I_j}\|_2^2 \le 10\delta\cdot\|\widehat{X}'\|_2^2$$

with probability at least $1-p/10$. Combining this with (1.10)–(1.11), and using the assumption $\delta\le\frac{1}{20}$ to write $\delta\le\sqrt{\delta}$, we complete the proof.

Sample complexity and runtime: There are two operations that cost us samples. The first is the call to EstimateEnergies, which costs $O\left(\frac{k_0k_1}{\delta^2}\log^2\frac{1}{\delta}\right)$ by Lemma 1.3.4. The second is the call to LocateReducedSignals; by Lemma 1.2.4 in Appendix A.8, with δp in place of p, this costs $O\left(\sum_{r\in[2k_1]}\mathbf{s}_r\cdot\log\frac{1}{\delta p}\log\frac{1}{\delta}\log n\right)$ samples (recall that $\widehat{\chi}$ is $(O(k_0),k_1)$-block sparse by assumption). Adding these contributions gives the desired result; the $\log\frac{1}{p}$ and $\log\frac{1}{\delta}$ factors are hidden in the $O^*(\cdot)$ notation, since we have assumed that δ and p behave as $\Omega(\frac{1}{\mathrm{poly}\log n})$.

The time complexity follows by a similar argument; The primitive EstimateEnergies has time complexity $O\left(\frac{k_0k_1}{\delta^2}\log^2\frac{1}{\delta}\log^2 n\right)$ by Lemma 1.3.4, and the primitive LocateReducedSignals requires $O\left(\sum_{r\in[2k_1]}\mathbf{s}_r\cdot\log\frac{1}{\delta p}\log\frac{1}{\delta}\log^2 n+\frac{k_0k_1}{\delta}\log\frac{1}{\delta p}\log^3 n\right)$ operations by Lemma 1.2.4 in Appendix A.8. The complexity of EstimateEnergies dominates that and of calling BudgetAllocation, which is $O\left(k_1+\frac{k_0}{\delta}\log\frac{1}{p}\right)$ by Lemma 1.3.1.

The sample complexity and runtime directly follows by plugging in the bound on $\sum_{r\in[2k_1]}\mathbf{s}_r$ given in (1.8). □

1.4 Energy Estimation

In this section, we provide the energy estimation procedure used in MultiBlockLocate primitive in Algorithm 2, and prove its guarantees that were used in the proof of Lemma 1.3.5. To do this, we introduce a variety of tools needed, including hashing and the semi-equispaced FFT. While such techniques are well-established for the standard sparsity setting (Indyk et al., 2014), applying the existing semi-equispaced FFT algorithms *separately* for each Z^r in our setting would lead to a runtime of $k_0k_1^2\mathrm{poly}(\log n)$. Our techniques allow us to compute the required FFT values for *all* r in $k_0k_1\mathrm{poly}(\log n)$ time, as we detail in Section 1.4.2.

1.4.1 Hashing Techniques

The notion of hashing plays a central role in our estimation primitives, and in turn makes use of random permutations.

Definition 1.4.1 (Approximately pairwise-independent permutation)**.** Fix integer n, and let $\pi:[n]\to[n]$ be a random permutation. We say that π is *approximately pairwise-independent* if, for any $i,i'\in[n]$ and any integer t, we have $\Pr[|\pi(i)-\pi(i')|\le t]\le\frac{4t}{n}$.

It is well known that such permutations exist in the form of a simple modulo-n multiplication;

we will specifically use the following lemma from (Indyk and Kapralov, 2014).

Lemma 1.4.1. (Choice of permutation (Indyk and Kapralov, 2014, Lemma 3.2)) *Let n be a power of two, and define $\pi(i) = \sigma \cdot i$, where σ is chosen uniformly at random from the odd numbers in $[n]$. Then π is an approximately pairwise-independent random permutation.*

We now turn to the notion of *hashing* a signal into buckets. We do this by applying the random permutation from Lemma 1.4.1 along with a random shift in time domain, and then applying a suitable filter according to Definition 1.2.1.

Definition 1.4.2 (Hashing)**.** Given integers n, B, parameters $\sigma, \Delta \in [n]$, and the signals $X \in \mathbb{C}^n$ and $G \in \mathbb{C}^n$, we say that $U \in \mathbb{C}^B$ is an $(n, B, G, \sigma, \Delta)$*-hashing* of X if

$$U_j = \frac{B}{n} \sum_{i \in [\frac{n}{B}]} X_{\sigma(\Delta + j + B \cdot i)} G_{j + B \cdot i}, \quad j \in [B]. \tag{1.13}$$

Moreover, we define the following quantities:

- $\pi(j) = \sigma \cdot j$, representing the approximately pairwise random permutation;
- $h(j) = \text{round}\big(j \frac{B}{n}\big)$, representing the bucket in $[B]$ into which a frequency j hashes;
- $o_j(j') = \pi(j') - h(\pi(j)) \cdot \frac{n}{B}$, representing the offset associated with two frequencies (j, j').

With these definitions, we have the following lemma, proved in Appendix A.3. Note that here we write the exact Fourier transform of U as $\widehat{U}^*$, since later we will use $\widehat{U}$ for its *near-exact* counterpart to simplify notation.

Lemma 1.4.2. (Fourier transform of hashed signal) *Fix integers n, B and the signals $X \in \mathbb{C}^n$ and $G \in \mathbb{C}^n$ with the latter symmetric about zero. If U is an $(n, B, G, \sigma, \Delta)$-hashing of X, then its exact Fourier transform $\widehat{U}^*$ is given by*

$$\widehat{U}^*_b = \sum_{f \in [n]} \widehat{X}_f \widehat{G}_{\sigma f - b \frac{n}{B}} \omega_n^{\sigma \Delta f}, \quad b \in [B].$$

We conclude this subsection by stating the following technical lemma regarding approximately pairwise independent permutations and flat filters.

Lemma 1.4.3. (Additional filter property) *Fix n, and let G be an (n, B, F)-flat filter. Let $\pi(\cdot)$ be an approximately pairwise-independent random permutation (*cf.*, Definition 1.4.1), and for $f, f' \in [n]$, define $o_f(f') = \pi(f') - \frac{n}{B}\text{round}\big(\pi(f)\frac{B}{n}\big)$. Then for any $x \in \mathbb{C}^n$ and $f \in [n]$, we have*

$$\sum_{f' \neq f} |\widehat{X}_{f'}|^2 \mathbb{E}_\pi\big[|\widehat{G}_{o_f(f')}|^2\big] \leq \frac{10}{B} \|\widehat{X}\|^2. \tag{1.14}$$

The proof is given in Appendix A.1.1.

Algorithm 3 Semi-equispaced inverse FFT for approximating the inverse Fourier transform, with standard sparsity (top) and block sparsity (bottom)

procedure SEMIEQUIINVERSEFFT($\widehat{X}, n, k, \zeta$)
  $\widehat{G} \leftarrow$ FILTER$(n, n/k, \zeta)$ ▷ See (Indyk et al., 2014, Sec. 12); same as proof of Lemma 1.4.5
  $\widehat{Y}_i \leftarrow (\widehat{X} \star \widehat{G})_{\frac{in}{2k}}$ for each $i \in [2k]$
  $Y \leftarrow$ INVERSEFFT$(\widehat{Y})$
  return $\{Y_j\}_{|j| \le \frac{k}{2}}$
procedure SEMIEQUIINVERSEBLOCKFFT($\widehat{X}, n, k_0, k_1, c$)
  $\widehat{G} \leftarrow$ FILTER(n, k_1, n^{-c}) ▷ See proof of Lemma 1.4.5
  for $j \in \left[\frac{2n}{k_1}\right]$ such that $(\widehat{X} \star \widehat{G})_{\frac{k_1}{2} j}$ may be non-zero ($O(k_0 \log n)$ in total) **do**
    $\widetilde{Y}_j^b \leftarrow \frac{k_1}{2} \sum_{l=1}^{\frac{n}{2k_1}} \widehat{X}_{b+2k_1 l} \widehat{G}_{\frac{k_1}{2} j - (b+2k_1 l)}$ for each $b \in [2k_1]$
    $(\widehat{Y}_j^1, \ldots, \widehat{Y}_j^{2k_1}) \leftarrow \mathrm{IFFT}(\widetilde{Y}_j^1, \ldots, \widetilde{Y}_j^{2k_1})$
  for $r \in [2k_1]$ **do**
    $\widehat{Y}^r \leftarrow (\widehat{Y}_1^r, \ldots, \widehat{Y}_{2n/k_1}^r)$
    $Y^r \leftarrow$ SEMIEQUIINVERSEFFT$(\widehat{Y}^r, \frac{n}{k_1}, k_0, n^{-(c+1)})$
  return $\{Y_j^r\}_{r \in [2k_1], |j| \le \frac{k_0}{2}}$

1.4.2 Semi-Equispaced FFT

One of the steps of our algorithm will be to take the inverse Fourier transform of our current estimate of the spectrum, so that it can be subtracted off and we can work with the residual. The *semi-equispaced inverse FFT* provides an efficient method for doing this, and is based on the application of the standard inverse FFT to a filtered and downsampled signal.

We start by describing an existing technique of this type for *standard* sparsity; the details are shown in the procedure SEMIEQUIINVERSEFFT in Algorithm 3, and the resulting guarantee from (Indyk et al., 2014, Sec. 12) is stated as follows.[2]

Lemma 1.4.4. (SEMIEQUIINVERSEFFT guarantees (Indyk et al., 2014, Lemma 12.1, Cor. 12.2))
(i) Fix n and a parameter $\zeta > 0$. If $\widehat{X} \in \mathbb{C}^n$ is k-sparse for some k, then SEMIEQUIINVERSEFFT$(\widehat{X}, n, k, \zeta)$ *returns a set of values $\{Y_j\}_{|j| \le k/2}$ in time $O(k \log \frac{n}{\zeta})$, satisfying the following for every j,*

$$|Y_j - X_j| \le \zeta \|X\|_2.$$

(ii) Given two additional parameters $\sigma, \Delta \in [n]$ with σ odd, it is possible to compute a set of values $\{Y_j\}$ for all j equaling $\sigma j' + \Delta$ for some j' with $|j'| \le k/2$, with the same running time and approximation guarantee.

For the block-sparse setting, we need to adapt the techniques of Indyk et al. (2014), making use of a *two-level* scheme that calls SEMIEQUIINVERSEFFT. The resulting procedure, SEMIEQUIIN-

[2] Note that the roles of time and frequency are reversed here compared to (Indyk et al., 2014).

VERSEBLOCKFFT, is described in Algorithm 3. The main result of the procedure is the following analog of Lemma 1.4.4.

Lemma 1.4.5. (SEMIEQUIINVERSEBLOCKFFT guarantees) *(i) Fix integers n, k_0, k_1, a (k_0, k_1)-block sparse signal $\widehat{X} \in \mathbb{C}^n$, and a constant $c \geq 1$. Define the shifted signals $\{X^r\}_{r\in[2k_1]}$ with $X^r_i = X_{i+\frac{nr}{2k_1}}$. The procedure* SEMIEQUIINVERSEBLOCKFFT*$(\widehat{X}, n, k_0, k_1, c)$ returns a set of values Y^r_j for all $r \in [2k_1]$ and $|j| \leq \frac{k_0}{2}$ in time $O(c^2 k_0 k_1 \log^2 n)$, satisfying,*

$$|Y^r_j - X^r_j| \leq 2n^{-c}\|X\|_2. \tag{1.15}$$

(ii) Given two additional parameters $\sigma, \Delta \in \left[\frac{n}{k_1}\right]$ with σ odd, it is possible to compute a set of values Y^r_j for all $r \in [2k_1]$ and j equaling $\sigma j' + \Delta$ (modulo $\frac{n}{k_1}$) for some $|j'| \leq \frac{k_0}{2}$, with the same running time and approximation guarantee.

The proof is given in Appendix A.3.1.

Remark 1.4.1. *In the preceding lemmas, the signal sparsity and the number of values we wish to estimate will not always be identical. However, this can immediately be resolved by letting the parameter k_0 therein equal the maximum of the two.*

1.4.3 Combining the Tools

In Algorithm 4, we describe two procedures combining the above tools. The first, HASHTOBINS, accepts the signal X and its current estimate in the Fourier domain $\widehat{\chi}$, uses SEMIEQUIINVERSEFFT to approximate the relevant entries of χ, and computes a hashing of $X - \chi$ as per Definition 1.4.2. The second, HASHTOBINSREDUCED, is analogous, but instead accepts a (k_1, δ)-downsampling of X, and uses SEMIEQUIINVERSEBLOCKFFT. It will prove useful to allow the function to hash into a different number of buckets for differing r values, and hence accept $\{G^r\}_{r\in[2k_1]}$ and $\{B^r\}_{r\in[2k_1]}$ as inputs. For simplicity, Algorithm 4 states the procedures without precisely giving the parameters passed to the semi-equispaced FFT, but the details are given in the proof of the following lemma.

Lemma 1.4.6. (HASHTOBINS and HASHTOBINSREDUCED guarantees) *(i) Fix integers n, k, B, F, an (n, B, F)-flat filter G supported on an interval of length $O(FB)$, a signal $X \in \mathbb{C}^n$, a k-sparse signal $\widehat{\chi}$. For any σ, Δ, the procedure* HASHTOBINS*$(X, \widehat{\chi}, G, n, B, \sigma, \Delta)$ returns a sequence $\widehat{U}$ such that,*

$$\|\widehat{U} - \widehat{U}^*\|_\infty \leq n^{-c}\|\widehat{\chi}\|_2,$$

where $\widehat{U}^*$ is the exact Fourier transform of the $(n, B, G, \sigma, \Delta)$-hashing of $X - \chi$ (see Definition 1.4.2), and $c = c' + O(1)$ for c' in Algorithm 4. Moreover, the sample complexity is $O(FB)$, and the runtime is $O(cF(B + k)\log n)$.

(ii) Fix integers n, k_0, k_1 and parameters $\{B^r\}_{r\in[2k_1]}, F, \delta$. For each $r \in [2k_1]$, fix an $\left(\frac{n}{k_1}, B^r, F\right)$-flat filter G^r supported on an interval of length $O(FB^r)$. Moreover, fix a signal $X \in \mathbb{C}^n$ and its (k_1, δ)-

Algorithm 4 Hash to bins functions for original signal (top) and reduced signals (bottom)

procedure HASHTOBINS($X, \widehat{\chi}, G, n, B, \sigma, \Delta$)
  Compute $\{\chi_i\}$ using SEMIEQUIINVERSEFFT with input $(\widehat{\chi}, n, O(FB), n^{-c'})$
    ▷ See Lemma 1.4.4; F equals the parameter of filter G, and c' is a large constant
  $U_X \leftarrow (n, B, G, \sigma, \Delta)$-hashing of X ▷ See Definition 1.4.2
  $U_\chi \leftarrow (n, B, G, \sigma, \Delta)$-hashing of χ
  $\widehat{U} \leftarrow$ FFT of $U_X - U_\chi$
  return $\widehat{U}$
procedure HASHTOBINSREDUCED($\{Z_X^r\}_{r\in[2k_1]}, \widehat{\chi}, \{G^r\}_{r\in[2k_1]}, n, k_1, \{B^r\}_{r\in[2k_1]}, \sigma, \Delta$)
  $B_{\max} \leftarrow \max_{r\in[2k_1]} B^r$
  $k_0 \leftarrow$ minimal value such that $\widehat{\chi}$ is (k_0, k_1)-block sparse
  Compute $\{\chi_i\}$ using SEMIEQUIINVERSEBLOCKFFT; input $(\widehat{\chi}, n, O(F_{\max}B_{\max} + k_0), k_1, c')$
▷ See Lemma 1.4.5; $F_{\max}$ equals the max parameter of filters $\{G^r\}$, and c' is a large constant
  $\{Z_\chi^r\}_{r\in[2k_1]} \leftarrow (k_1, \delta)$-downsampling of χ ▷ See Definition 1.2.2
  for $r \in [2k_1]$ **do**
    $U_X^r \leftarrow \left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing of Z_X^r ▷ See Definition 1.4.2
    $U_\chi^r \leftarrow \left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing of Z_χ^r
    $\widehat{U}^r \leftarrow$ FFT of $U_X^r - U_\chi^r$
  return $\{\widehat{U}^r\}_{r\in[2k_1]}$

downsampling $\{Z^r\}_{r\in[2k_1]}$ with $\delta \in \left(0, \frac{1}{20}\right)$, and a (k_0, k_1)-block sparse signal $\widehat{\chi}$. For any σ, Δ, the procedure HASHTOBINSREDUCED $\left(\{Z^r\}_{r\in[2k_1]}, \widehat{\chi}, \{G^r\}_{r\in[2k_1]}, n, k_1, \{B^r\}_{r\in[2k_1]}, \sigma, \Delta\right)$ *returns a set of sequences $\{\widehat{U}^r\}_{r\in[2k_1]}$ such that,*

$$\|\widehat{U}^r - \widehat{U}^{*r}\|_\infty \le n^{-c}\|\widehat{\chi}\|_2, \quad r \in [2k_1],$$

where $\widehat{U}^{*r}$ is the exact Fourier transform of the $\left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing for the (k_1, δ)-downsampling of $X - \chi$, and $c = c' + O(1)$ for c' in Algorithm 4. Moreover, the sample complexity is $O\left(F\sum_{r\in[2k_1]} B^r \log\frac{1}{\delta}\right)$, and the runtime is $O\left(c^2(B_{\max}F + k_0)k_1 \log^2 n\right)$ with $B_{\max} = \max_{r\in[2k_1]} B^r$.

The proof is given in Appendix A.3.2.

Remark 1.4.2. *Throughout the chapter, we consider c in Lemma 1.4.6 to be a large absolute constant. Specifically, various results make assumptions such as $\|\widehat{X} - \widehat{\chi}\|_2 \ge \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$, and the results hold true when c is sufficiently large compared to implied exponent in the* poly(n) *notation. Essentially, the n^{-c} error term is so small that it can be thought of as zero, but we nevertheless handle it explicitly for completeness.*

1.4.4 Estimating the Downsampled Signal Energies

We now come to the main task of this section, namely, approximating the energy of each $\widehat{Z}^r$. To do this, we hash into $B = \frac{4}{\delta^2} \cdot k_0$ buckets (*cf.*, Definition 1.4.2), and form the estimate as the energy of the hashed signal. The procedure is shown in Algorithm 5.

Algorithm 5 Procedure for estimating energies of downsampled signals

procedure EstimateEnergies($X, \widehat{\chi}, n, k_0, k_1, \delta$)
  $B \leftarrow \frac{40}{\delta^2} \cdot k_0$
  $F \leftarrow 10 \log \frac{1}{\delta}$
  $H \leftarrow (\frac{n}{k_1}, B, F)$-flat filter ▷ See Definition 1.2.1
  $\Delta \leftarrow$ uniform random sample from $[\frac{n}{k_1}]$
  $\sigma \leftarrow$ uniform random sample from odd numbers in $[\frac{n}{k_1}]$
  $\{Z^r\}_{r \in [2k_1]} \leftarrow (k_1, \delta)$-downsampling of X ▷ See Definition 1.2.2
  $\mathbf{H} \leftarrow (H, \dots, H)$
  $\mathbf{B} \leftarrow (B, \dots, B)$
  $\{\widehat{U}^r\}_{r \in [2k_1]} \leftarrow$ HashToBinsReduced($\{Z^r\}_{r \in [2k_1]}, \widehat{\chi}, \mathbf{H}, n, k_1, \mathbf{B}, \sigma, \Delta$) ▷ See Section 1.4.1
  for $r \in [2k_1]$ **do**
    $\gamma^r \leftarrow \|\widehat{U}^r\|_2^2$
  return γ ▷ Length-$2k_1$ vector of γ^r values

Before stating the guarantees of Algorithm 5, we provide the following lemma characterizing the approximation quality for an *exact* hashing of a signal, as opposed to the approximation returned by HashToBinsReduced. Intuitively, the first part states that we can accurately estimate the top coefficients well without necessarily capturing the noise, and the second part states that, in expectation, we do not over-estimate the total signal energy by more than a small constant factor.

Lemma 1.4.7. (Properties of exact hashing) *Fix the integers (m, B), the parameters $\delta \in \left(0, \frac{1}{20}\right)$ and $F' \geq 10 \log \frac{1}{\delta}$, and the signal $Y \in \mathbb{C}^m$ and (m, B, F')-flat filter H (*cf.*, Definition 1.2.1). Let U be an $(m, B, H, \sigma, \Delta)$-hashing of Y for uniformly random $\sigma, \Delta \in [m]$ with σ odd, and let $\pi(\cdot)$ be defined as in Definition 1.4.2. Then, letting $\widehat{U}^*$ denote the exact Fourier transform of U, we have the following:*

1. *For any set $S \subset [m]$,*

$$\mathbb{E}_{\Delta,\pi}\left[\left|\|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*\|_2^2\right|_+\right] \leq \left(10\sqrt{\frac{|S|}{B}} + 15\frac{|S|}{B} + 2\delta^2\right)\|\widehat{Y}\|_2^2,$$

   where $\|\widehat{Y}_S\|_2^2$ denotes $\sum_{j \in S} |\widehat{Y}_j|^2$.

2. *We have $\mathbb{E}_{\Delta,\pi}\left[\|\widehat{U}^*\|_2^2\right] \leq 3\|\widehat{Y}\|_2^2$.*

The proof is given in Appendix A.3.3.

We now present the following lemma, showing that the procedure EstimateEnergies provides us with an estimator satisfying the preconditions of Lemmas 1.3.2 and 1.3.3.

Lemma 1.3.4 (EstimateEnergies guarantees – restated from Section 1.3.1)**.** *Given integers n, k_0, k_1, signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2^2 \geq \frac{1}{\text{poly}(n)}\|\widehat{\chi}\|_2$, and parameter $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$, the*

procedure ESTIMATEENERGIES$(X, \widehat{\chi}, n, k_0, k_1, \delta)$ *returns a non-negative vector* $\gamma \in \mathbb{R}_+^{2k_1}$ *such that, for any given set* S^* *of cardinality at most* $10k_0$*, we have the following with probability at least* $\frac{1}{2}$*:*

1. $\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \right|_+ \le 40\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;
2. $\|\gamma\|_1 \le 10 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2$;

where $\{Z^r\}_{r\in[2k_1]}$ *is the* (k_1,δ)*-downsampling of* $X-\chi$ *(see Definition 1.2.2).*

Moreover, if $\widehat{\chi}$ *is* $(O(k_0), k_1)$*-block sparse, then the sample complexity is* $O\left(\frac{k_0 k_1}{\delta^2}\log^2\frac{1}{\delta}\right)$*, and the runtime is* $O\left(\frac{k_0 k_1}{\delta^2}\log^2\frac{1}{\delta}\log^2 n\right)$.

Proof. **Analysis for the exact hashing sequence:** We start by considering the case that the call to HASHTOBINSREDUCED is replaced by an evaluation of the exact hashing sequence $\widehat{U}^{*r}$, i.e., Definition 1.4.2 applied to Z^r resulting from the (k_1,δ)-downsampling of $X-\chi$. In this case, by applying Lemma 1.4.7 with $\widehat{Y} = \widehat{Z}^r$, $B = \frac{40}{\delta^2}k_0$ and $S = S^*$ (and hence $|S| \le 10k_0$), the right-hand side of the first claim therein becomes $(5\delta + (\frac{15}{4}+2)\delta^2)\|\widehat{Z}^r\|_2^2 \le 6\delta\|\widehat{Z}^r\|_2^2$, since $\delta \le \frac{1}{20}$. By applying the lemma separately for each $r \in [2k_1]$ with $\widehat{Y} = \widehat{Z}^r$, and summing the corresponding expectations in the two claims therein over r, we obtain $\sum_{r\in[2k_1]} \mathbb{E}\left[\left|\|\widehat{Z}^r_{S^*}\|_2^2 - \|\widehat{U}^*\|_2^2\right|_+\right] \le 6\delta\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2$ and $\sum_{r\in[2k_1]}\mathbb{E}\left[\|\widehat{U}^*\|_2^2\right] \le 3\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2$. We apply Markov's inequality with a factor of 6 in the former and 3 in the latter, to conclude that the quantities $\gamma^{*r} = \|\widehat{U}^{*r}\|_2^2$ satisfy,

$$\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^{*r} \right|_+ \le 36\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 \tag{1.16}$$

$$\|\gamma^*\|_1 \le 9 \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2, \tag{1.17}$$

with probability at least 1/2.

Incorporating $\frac{1}{\mathbf{n^c}}$ error from use of semi-equispaced FFT in HASHTOBINSREDUCED: Since $\widehat{U}^r$ is computed using HASHTOBINSREDUCED, the energy vector γ is different from the exact one γ^*, and we write

$$\sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r \right|_+ \le \sum_{r\in[2k_1]} \left| \|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^{*r} \right|_+ + \left|\gamma^r - \gamma^{*r}\right|. \tag{1.18}$$

By substituting $\gamma^r = \|\widehat{U}^r\|_2^2$ and $\gamma^{*r} = \|\widehat{U}^{*r}\|_2^2$, and using the identity $\left|\|a\|_2^2 - \|b\|_2^2\right| \le 2\|a-b\|_2 \cdot \|b\|_2 + \|a-b\|_2^2$, we can write

$$\sum_{r\in[2k_1]} \left|\gamma^r - \gamma^{*r}\right| \le \sum_{r\in[2k_1]} \left(2\|\widehat{U}^r - \widehat{U}^{*r}\|_2\|\widehat{U}^{*r}\|_2 + \|\widehat{U}^r - \widehat{U}^{*r}\|_2^2\right). \tag{1.19}$$

Upper bounding the ℓ_2 norm by the ℓ_∞ norm times the vector length, we have $\|\widehat{U}^r - \widehat{U}^{*r}\|_2 \le$

$\sqrt{n}\|\widehat{U}^r - \widehat{U}^{*r}\|_\infty \le n^{-c+1/2}\|\widehat{\chi}\|_2$, where the second inequality follows from Lemma 1.4.6. Moreover, from the definition of $\widehat{U}^{*r}$ resulting from Definition 1.4.2 applied to Z^r, along with the filter property $\|\widehat{G}\|_\infty$ in Definition 1.2.1, it follows that $\|\widehat{U}^{*r}\|_2 \le \|\widehat{G}\|_\infty \|\widehat{Z}^r\|_1 \le \sqrt{n}\|\widehat{Z}^r\|_2$. Combining these into (1.19) gives

$$\begin{aligned}\sum_{r\in[2k_1]} \left|\gamma^r - \gamma^{*r}\right| &\le \sum_{r\in[2k_1]} \left(2n^{-c+1}\|\widehat{\chi}\|_2\|\widehat{Z}^r\|_2 + n^{-2c+1}\|\widehat{\chi}\|_2^2\right) \\ &\le 2n^{-c+2}\sqrt{\sum_{r\in[2k_1]}\|\widehat{\chi}\|_2^2 \cdot \sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2} + n^{-2c+1}k_1\|\widehat{\chi}\|_2^2 \\ &\le 2n^{-c+3}\|\widehat{\chi}\|_2\sqrt{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2} + n^{-2c+2}\|\widehat{\chi}\|_2^2. \end{aligned} \tag{1.20}$$

where the second line is by Cauchy-Schwarz, and the third by $k_1 \le n$.

By the second part of Lemma 1.2.3 and the assumption $\delta \le \frac{1}{20}$, we have $\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2 \ge \frac{1}{4}\|\widehat{X} - \widehat{\chi}\|_2^2 \ge \frac{1}{4n^{c'}}\|\widehat{\chi}\|_2^2$, where the second equality holds for some $c' > 0$ by the assumption $\|\widehat{X} - \widehat{\chi}\|_2^2 \ge \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$. Hence, (1.20) gives

$$\sum_{r\in[2k_1]} \left|\gamma^r - \gamma^{*r}\right| \le 4\left(n^{-c+3}n^{c'/2} + n^{-2c+2}n^{c'}\right)\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2. \tag{1.21}$$

Since we have chosen $\delta > 1/n$, the coefficient to the summation is upper bounded by 4δ when c is sufficiently large, thus yielding the first part of the lemma upon combining with (1.16).

To prove the second part, note that by the triangle inequality,

$$\begin{aligned}\|\gamma\|_1 &\le \|\gamma^*\|_1 + \left|\|\gamma\|_1 - \|\gamma^*\|_1\right| \\ &\le 9\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2 + \sum_{r\in[2k_1]}\left|\gamma^r - \gamma^{*r}\right|, \end{aligned} \tag{1.22}$$

where we have applied (1.17). Again applying (1.21) and noting that the coefficient to the summation is less than one for sufficiently large c, the second claim of the lemma follows.

Sample complexity and runtime: The only step that uses samples is the call to HASHTOBINSREDUCED. By Lemma 1.4.6 and the choices $B = \frac{40}{\delta^2}k_0$ and $F = 10\log\frac{1}{\delta}$, this uses $O\left(k_1 FB\log\frac{1}{\delta}\right) = O\left(\frac{k_0k_1}{\delta^2}\log^2\frac{1}{\delta}\right)$ samples per call. The time complexity follows by the same argument along the assumption that $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, with an additional $\log^2 n$ factor following from Lemma 1.4.6. Note that the call to HASHTOBINSREDUCED dominates the computation of γ^r, which is $O(k_1B)$, □

1.5 The Block-Sparse Fourier Transform

In this section, we combine the tools from the previous sections to obtain the full sublinear-time block sparse FFT algorithm, and provide its guarantees.

1.5.1 Additional Estimation Procedures

Before stating the final algorithm, we note the main procedures that it relies on: MULTIBLOCKLOCATE, PRUNELOCATION, and ESTIMATEVALUES. We presented the first of these in Section 1.3. The latter two are somewhat more standard, and hence we relegate them to the appendices. However, for the sake of readability, we provide some intuition behind them here, and state their guarantees.

We begin with PRUNELOCATION. The procedure MULTIBLOCKLOCATE gives us a list of block indices containing the dominant signal blocks with high probability, with a list size $L = O^*\left(k_0 \log k_0\right)$. Estimating the values of all of these blocks in every iteration would not only cost $O^*(k_0 k_1 \log k_0)$ samples, but would also destroy the sparsity of the input signal: Most of the blocks correspond to noise, and thus the estimation error may dominate the values being estimated. The PRUNELOCATION primitive is designed to alleviate these issues, pruning L to a list that contains mostly "signal" blocks, i.e., blocks that contain a large amount of energy. Some false positives and false negatives occur, but are controlled by Lemma 1.5.1 below. The procedure is given in Algorithm 16 in Appendix A.4.

The following lemma shows that with high probability, the pruning algorithm retains most of the energy in the head elements, while removing most tail elements.

Lemma 1.5.1. (PRUNELOCATION guarantees) *Given integers n, k_0, k_1, a list of block indices $L \subseteq \left[\frac{n}{k_1}\right]$, parameters $\theta > 0$, $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$ and $p \in (0,1)$, and signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\text{poly}(n)}\|\widehat{\chi}\|_2$, the output L' of* PRUNELOCATION$(X, \widehat{\chi}, L, n, k_0, k_1, \delta, p, \theta)$ *has the following properties:*

a. *Let S_{tail} denote the tail elements in signal $\widehat{X} - \widehat{\chi}$, defined as,*

$$S_{\text{tail}} = \left\{ j \in \left[\frac{n}{k_1}\right] : \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \leq \sqrt{\theta} - \sqrt{\frac{\delta}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 \right\},$$

where I_j is defined in Definition 1.1.1. Then, we have,

$$\mathbb{E}\left[\left|L' \cap S_{\text{tail}}\right|\right] \leq \delta p \cdot |L \cap S_{\text{tail}}|.$$

b. *Let S_{head} denote the head elements in signal $\widehat{X} - \widehat{\chi}$, defined as,*

$$S_{\text{head}} = \left\{ j \in \left[\frac{n}{k_1}\right] : \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \geq \sqrt{\theta} + \sqrt{\frac{\delta}{k_0}}\|\widehat{X} - \widehat{\chi}\|_2 \right\}.$$

Then, we have,

$$\mathbb{E}\left[\sum_{j \in (L \cap S_{\text{head}}) \setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2\right] \leq \delta p \sum_{j \in L \cap S_{\text{head}}} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2.$$

Moreover, provided that $\|\widehat{\chi}\|_0 = O(k_0 k_1)$*, the sample complexity is* $O\left(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta p} \log \frac{1}{\delta}\right)$*, and the runtime is* $O\left(\frac{k_0 k_1}{\delta} \log \frac{1}{\delta p} \log \frac{1}{\delta} \log n + k_1 \cdot |L| \log \frac{1}{\delta p}\right)$.

The proof is given in Appendix A.4.

We are left with the procedure EstimateValues, which is a standard procedure for estimating the signal values at the frequencies within the blocks after they have been located. The details are given in Algorithm 17 in Appendix A.5.

Lemma 1.5.2. (EstimateValues guarantees) *For any integers* n, k_0, k_1*, any list of block indices* $L \subseteq \left[\frac{n}{k_1}\right]$*, parameters* $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$ *and* $p \in (0, 1/2)$*, and signals* $X \in \mathbb{C}^n$ *and* $\widehat{\chi} \in \mathbb{C}^n$ *with* $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\text{poly}(n)} \|\widehat{\chi}\|_2$*, with probability at least* $1 - p$*, the output* W *of the function* EstimateValues $(X, \widehat{\chi}, L, n, k_0, k_1, \delta, p)$ *(Algorithm 17) has the following property:*

$$\sum_{f \in \bigcup_{j \in L} I_j} |W_f - (\widehat{X} - \widehat{\chi})_f|^2 \leq \delta \frac{|L|}{3k_0} \|\widehat{X} - \widehat{\chi}\|_2^2,$$

where I_j *is the* j*-th block. Moreover, provided that* $\|\widehat{\chi}\|_0 = O(k_0 k_1)$*, the sample complexity is* $O\left(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta}\right)$*, and the runtime is* $O\left(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta} \log n + k_1 \cdot |L| \log \frac{1}{p}\right)$.

The proof is given in Appendix A.5.

1.5.2 Statement of the Algorithm and Main Result

Our overall block-sparse Fourier transform algorithm is given in Algorithm 6. It first calls ReduceSNR, which performs an iterative procedure that picks up high energy components of the signal, subtracts them from the original signal, and then recurses on the residual signal $X^{(i)} = X - \chi^{(i)}$. Once this is done, the procedure RecoverAtConstSNR performs a final "clean-up" step to obtain the $(1+\epsilon)$-approximation guarantee.

With these definitions in place, we can now state our final result, which formalizes Theorem 1.1.1.

Theorem 1.1.1 (Upper bound – formal version)**.** *Given integers* n, k_0, k_1*, parameter* $\epsilon \in \left(\frac{1}{n}, \frac{1}{20}\right)$*, and the signal* $X \in \mathbb{C}^n$*, if* $\widehat{X}$*,* SNR′*,* μ^2*, and* ν^2 *satisfy the following for* (μ^2, SNR) *given in Definition 1.1.2:*

1. $\mu^2 \leq \nu^2$;
2. $\|\widehat{X}\|_2^2 \leq (k_0 \nu^2) \cdot \text{SNR}'$;
3. $\text{SNR}' = O(\text{poly}(n))$;
4. $\mu^2 \geq \frac{\|\widehat{X}\|_2^2}{\text{poly}(n)}$;

Algorithm 6 Block-sparse Fourier transform.

procedure BLOCKSPARSEFT($X, n, k_0, k_1, \mathrm{SNR}', \nu^2, \epsilon$)
  ▷ $X \in \mathbb{C}^n$ is approximately (k_0, k_1)-block sparse
  ▷ (SNR', ν^2) are upper bounds on (SNR, μ^2) from Definition 1.1.2
  ▷ ϵ is the parameter for $(1+O(\epsilon))$-approximate recovery
  $\widehat{\chi} \leftarrow$ REDUCESNR($X, n, k_0, k_1, \mathrm{SNR}', \nu^2$).
  $\widehat{\chi} \leftarrow$ RECOVERATCONSTSNR($X, \widehat{\chi}, n, k_0, k_1, \nu^2, \epsilon$).
  return $\widehat{\chi}$
procedure REDUCESNR($X, n, k_0, k_1, \mathrm{SNR}', \nu^2$) ▷ Iteratively locate/estimate to reduce SNR
  $T \leftarrow \log \mathrm{SNR}'$
  $\delta \leftarrow$ small absolute constant
  $p \leftarrow \frac{\delta}{\log^2 \frac{k_0}{\delta} \log^4 \mathrm{SNR}'}$ ▷ Failure probability for subroutines
  $\widehat{\chi}^{(0)} \leftarrow 0$ ▷ $\widehat{\chi}^{(t)}$ is our current estimate of $\widehat{X}$
  for $t \in \{1, \ldots, T\}$ **do**
    $L \leftarrow$ MULTIBLOCKLOCATE($X, \widehat{\chi}^{(t-1)}, n, k_1, k_0, \delta, p$)
    $\theta \leftarrow 10 \cdot 2^{-t} \cdot \nu^2 \mathrm{SNR}'$ ▷ Threshold for pruning
    $L' \leftarrow$ PRUNELOCATION($X, \widehat{\chi}^{(t-1)}, L, n, k_0, k_1, \delta, p, \theta$)
    $\widehat{\chi}^{(t)} \leftarrow \widehat{\chi}^{(t-1)} +$ ESTIMATEVALUES($X, \widehat{\chi}^{(t-1)}, L', n, k_0, k_1, \delta, p$)
  return $\widehat{\chi}^{(T)}$
procedure RECOVERATCONSTSNR($X, \widehat{\chi}, n, k_0, k_1, \epsilon$) ▷ A final "clean-up" step
  $\eta \leftarrow$ small absolute constant
  $p \leftarrow \frac{\eta\epsilon}{\log^2 \frac{k_0}{\epsilon}}$ ▷ Upper bound on failure probability for subroutines
  $L \leftarrow$ MULTIBLOCKLOCATE($X, \widehat{\chi}, n, k_1, k_0, \epsilon^2, p$)
  $\theta \leftarrow 200\epsilon\nu^2$
  $L' \leftarrow$ PRUNELOCATION($X, \widehat{\chi}, L, n, k_0, k_1, \epsilon, p, \theta$)
  $W \leftarrow$ ESTIMATEVALUES($X, \widehat{\chi}, L', n, 3k_0/\epsilon, k_1, \epsilon, p$)
  $\widehat{\chi}' \leftarrow W + \widehat{\chi}$
  return $\widehat{\chi}'$

then with probability at least 0.8*, the procedure* BLOCKSPARSEFT$(X, n, k_0, k_1, \mathrm{SNR}', \nu^2, \epsilon)$ *satisfies the following: (i) The output* $\widehat{\chi}$ *satisfies,*

$$\|\widehat{X} - \widehat{\chi}\|_2^2 \le k_0 \left(\mu^2 + O(\epsilon\nu^2)\right).$$

(ii) The sample complexity is $O^*\left(\left(k_0 \log(1+k_0)\log n + k_0 k_1\right)\log \mathrm{SNR}' + \frac{k_0}{\epsilon^2}\log(1+k_0)\log n + \frac{k_0 k_1}{\epsilon^4}\right)$*, and the runtime is* $O^*\left(\left(k_0\log(1+k_0) + k_0 k_1 \log n\right)\log \mathrm{SNR}' \log^2 n + \left(\frac{k_0 k_1}{\epsilon^2}\log n + \frac{k_0 k_1}{\epsilon^4}\right)\log^2 n\right)$.

The assumptions of the theorem are essentially that we know upper bounds on the tail noise μ^2 and SNR. Moreover, in order to get the $(1+\epsilon)$-approximation guarantee, the former upper bound should be tight to within a constant factor.

In the remainder of the section, we provide the proof of Theorem 1.1.1, deferring the technical details to the appendices.

Guarantees for REDUCESNR and RECOVERATCONSTSNR.

The following theorem proves the success of the function REDUCESNR. We again recall the definitions of Err^2, μ^2, and SNR in Definition 1.1.2.

Lemma 1.5.3. (REDUCESNR guarantees) *Given integers n, k_0, k_1, parameters ν, SNR', and a signal $X \in \mathbb{C}^n$, if $\widehat{X}$, SNR', and ν^2 satisfy the following for (μ^2, SNR) given in Definition 1.1.2:*

1. $\mu^2 \le \nu^2$;
2. $\|\widehat{X}\|_2^2 \le (k_0\nu^2) \cdot \mathrm{SNR}'$;
3. $\mathrm{SNR}' = O(\mathrm{poly}(n))$;
4. $\nu^2 \ge \frac{\|\widehat{X}\|_2^2}{\mathrm{poly}(n)}$;

then the procedure $\mathrm{REDUCESNR}(X, n, k_0, k_1, \mathrm{SNR}', \nu^2)$ *satisfies the following guarantees with probability at least* 0.9 *when the constant δ therein is sufficiently small: (i) The output $\widehat{\chi}^{(T)}$ satisfies,*

$$\widehat{\chi}^{(T)} \text{ is } (3k_0, k_1)\text{-block sparse}$$
$$\left\|\widehat{X} - \widehat{\chi}^{(T)}\right\|_2^2 \le 100 k_0 \nu^2.$$

(ii) The number of samples is $O^*\left(k_0 \log(1+k_0) \log \mathrm{SNR}' \log n + k_0 k_1 \log \mathrm{SNR}'\right)$, and the runtime is $O^*\left(k_0 \log(1+k_0) \log \mathrm{SNR}' \log^2 n + k_0 k_1 \log \mathrm{SNR}' \log^3 n\right)$.

The proof is given in Appendix A.6.1.

The following theorem proves the success of the function RECOVERATCONSTSNR.

Lemma 1.5.4. (RECOVERATCONSTSNR guarantees) *Given integers n, k_0, k_1, parameters $\nu^2 \ge \frac{\|\widehat{X}\|_2^2}{\mathrm{poly}(n)}$ and $\epsilon \in \left(\frac{1}{n}, \frac{1}{20}\right)$, and signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ satisfying,*

1. $\mathrm{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1) \le k_0\nu^2$;
2. $\|\widehat{X} - \widehat{\chi}\|_2^2 \le 100 k_0 \nu^2$;

the procedure $\mathrm{RECOVERATCONSTSNR}(X, \widehat{\chi}, n, k_0, k_1, \epsilon)$ *satisfies the following guarantees with probability at least* 0.9 *when the constant η therein is sufficiently small: (i) The output $\widehat{\chi}'$ satisfies,*

$$\|\widehat{X} - \widehat{\chi}'\|_2^2 \le \mathrm{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1) + O(\epsilon) \cdot k_0\nu^2. \tag{1.23}$$

(ii) If $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, then its sample complexity is $O^*\left(\frac{k_0}{\epsilon^2}\log(1+k_0)\log n + \frac{k_0 k_1}{\epsilon^4}\right)$ and its runtime is $O^*\left(\frac{k_0 k_1}{\epsilon^4}\log^2 n + \frac{k_0 k_1}{\epsilon^2}\log^3 n\right)$.

The proof is given in Appendix A.6.2.

Proof of Theorem 1.1.1. We are now in a position to prove Theorem 1.1.1 via a simple combination of Lemmas 1.5.3 and 1.5.4.

Success event associated with REDUCESNR: Define a successful run of REDUCESNR$(X, n, k_0, k_1, \text{SNR}, \nu^2)$ to mean the following conditions on its output $\widehat{\chi}^{(T)}$:

$$\|\widehat{X} - \widehat{\chi}^{(T)}\|_2^2 \leq 100 k_0 \nu^2$$
$$\text{Err}^2(\widehat{X} - \widehat{\chi}^{(T)}, 10k_0, k_1) \leq k_0 \mu^2.$$

By Lemma 1.5.3, it follows that the probability of having a successful run of REDUCESNR is at least 0.9. Note that the second condition is not explicitly stated in Lemma 1.5.3, but it follows by using $3k_0$ blocks to cover the parts where $\widehat{\chi}^T$ is non-zero, and k_0 blocks to cover the dominant blocks of $\widehat{X}$, in accordance with Definition 1.1.2.

Success event associated with RECOVERATCONSTSNR: Define a successful run of RECOVERATCONSTSNR$(X, \widehat{\chi}, k_0, k_1, n, \epsilon)$ to mean the following conditions on its output $\widehat{\chi}'$:

$$\|\widehat{X} - \widehat{\chi}'\|_2^2 \leq \text{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1) + (4 \cdot 10^5)\epsilon k_0 \nu^2.$$

Conditioning on the event of having a successful run of REDUCESNR, by Lemma 1.5.4, it follows that the probability of having a successful run of RECOVERATCONSTSNR is at least 0.9.

By a union bound, the aforementioned events occur simultaneously with probability at least 0.8, as desired. Moreover, the sample complexity and runtime are a direct consequence of summing the contributions from Lemmas 1.5.3 and 1.5.4.

1.6 Lower Bound

Our upper bound in Theorem 1.1.1, in several scaling regimes, provides a strict improvement over standard sparse FFT algorithms in terms of sample complexity. The corresponding algorithm is inherently *adaptive*, which raises the important question of whether adaptivity is necessary in order to achieve these improvements. In this section, we show that the answer is affirmative, by proving the following formalization of Theorem 1.1.2.

Theorem 1.1.2 (Lower bound – formal version)**.** *Fix integers n, k_0, k_1 and constant $C > 0$, and suppose that there exists a non-adaptive algorithm that, when given a signal Y with Fourier transform $\widehat{Y}$, outputs a signal $\widehat{Y}'$ satisfying the following ℓ_2/ℓ_2-guarantee with probability at least $\frac{1}{2}$:*

$$\|\widehat{Y} - \widehat{Y}'\|_2^2 \leq C \min_{\widehat{Y}^* \text{ is } (k_0,k_1)-\text{block sparse}} \|\widehat{Y} - \widehat{Y}^*\|_2^2. \tag{1.24}$$

Then the number of samples taken by the algorithm must behave as $\Omega\big(k_0 k_1 \log \frac{n}{k_0 k_1}\big)$.

Hence, for instance, if $k_0 = O(1)$ and SNR $= O(1)$ then our adaptive algorithm uses $O(k_1 + \log n)$ samples, whereas any non-adaptive algorithm must use $\Omega\big(k_1 \log \frac{n}{k_1}\big)$ samples.
The remainder of this section is devoted to the proof of Theorem 1.1.2. Throughout the section, we let $k = k_0 k_1$ denote the total sparsity.

High-level overview: Our analysis follows the information-theoretic framework of Price and Woodruff (2011). However, whereas Price and Woodruff (2011) considers a signal with k arbitrary dominant frequency locations and uniform noise, we consider signals where the $k = k_0 k_1$ dominant frequencies are (nearly) contiguous, and both the noise and signal are concentrated on an $O\big(\frac{1}{k}\big)$ fraction of the time domain.

As a result, while the difficulty in Price and Woodruff (2011) arises from the fact that the algorithm needs to recover roughly $\log \frac{n}{k}$ bits per frequency location for k such locations, our source of difficulty is different. In our signal, there are only roughly $\log \frac{n}{k}$ bits to be learned about the location of *all* the blocks in frequency domain, but the signal is tightly concentrated on an $O\big(\frac{1}{k}\big)$ fraction of the input space. As a consequence, any non-adaptive algorithm is bound to waste most of its samples on regions of the input space where the signal is zero, and only an $O\big(\frac{1}{k}\big)$ fraction of its samples can be used to determine the single frequency that conveys the location of the blocks. In the presence of noise, this results in a lower bound on sample complexity of $\Omega\big(k \log \frac{n}{k}\big)$.

Information-theoretic preliminaries: We will make use of standard results from information theory, stated below. Here and subsequently, we use the notations $H(X)$, $H(Y|X)$, $I(X;Y)$ and $I(X;Y|U)$ for the (conditional) Shannon entropy and (conditional) mutual information (e.g., see Cover and Thomas (2012)).

We first state Fano's inequality, a commonly-used tool for proving lower bounds by relating a conditional entropy to an error probability.

Lemma 1.6.1. (Fano's Inequality (Cover and Thomas, 2012, Lemma 7.9.1)) *Fix the random variables (X,Y) with X being discrete, let X' be an estimator of X such that $X \to Y \to X'$ forms a Markov chain (i.e., X and X' are conditionally independent given Y), and define $P_e := \Pr[X' \neq X]$. Then*

$$H(X|Y) \leq 1 + P_e \log_2 |\mathcal{X}|,$$

where $\mathcal{X} = \text{supp}\,(X)$. Consequently, if X is uniformly distributed, then

$$I(X;Y) \geq -1 + (1 - P_e) \log_2 |\mathcal{X}|.$$

The next result gives the formula for the capacity of a complex-valued additive white Gaussian noise channel, often referred to as the Shannon-Hartley theorem. Here and subsequently, $\text{CN}(\mu, \sigma^2)$ denotes the complex normal distribution.

Lemma 1.6.2. (Complex Gaussian Channel Capacity (Cover and Thomas, 2012, Thm. 2.8.1))

For $Z \sim \mathrm{CN}(0,\sigma_z^2)$ and any complex random variable X with $\mathbb{E}[|X|^2] = \sigma_x^2$, we have

$$I(X;X+Z) \le \log_2\left(1+\frac{\sigma_x^2}{\sigma_z^2}\right),$$

with equality if $X \sim \mathrm{CN}(0,\sigma_x^2)$.

The following lemma states the data processing inequality, which formalizes the statement that processing a channel output cannot increase the amount of information revealed about the input.

Lemma 1.6.3. (Data Processing Inequality (Cover and Thomas, 2012, Thm. 2.8.1)) *For any random variables (X,Y,Z) such that $X \to Y \to Z$ forms a Markov chain, we have $I(X;Z) \le I(X;Y)$.*

Finally, the following lemma bounds the mutual information between two vectors in terms of the individual mutual information terms between components of those vectors.

Lemma 1.6.4. (Mutual Information for Vectors (Cover and Thomas, 2012, Lemma 7.9.2)) *For any random vectors $\mathbf{X} = (X_1,\ldots,X_n)$ and $\mathbf{Y} = (Y_1,\ldots,Y_n)$, if $\mathbf{Y}_j$ is conditionally independent of everything given $\mathbf{X}_j$ for every $j = 1,2,\cdots n$, then $I(\mathbf{X};\mathbf{Y}) \le \sum_{i=1}^n I(X_i;Y_i)$.*

A communication game: We consider a communication game consisting of channel coding with a state known at both the encoder (Alice) and decoder (Bob), where block-sparse recovery is performed at the decoder. Recalling that we are in the non-adaptive setting, by Yao's minimax principle, we can assume that the samples are deterministic and require probability-$\frac{1}{2}$ recovery over a random ensemble of signals, as opposed to randomizing the samples and requiring constant-probability recovery for any given signal in the ensemble. Hence, we denote the *fixed* sampling locations by $\mathscr{A}$.

We now describe our hard input distribution. Each signal in the ensemble is indexed by two parameters (u,f^*), and is given by

$$X_t = \begin{cases} \omega^{f^* t} & t \in \{u,\ldots,u+\frac{C'n}{k}-1\} \\ 0 & \text{otherwise,} \end{cases} \tag{1.25}$$

where $C' > 0$ is a constant that will be chosen later, and where all indices are modulo-n. Hence, each signal is non-zero only in a window of length $\frac{C'n}{k}$, and within that window, the signal oscillates at a rate dictated by f^*. Specifically, u specifies where the signal is non-zero in time domain, and f^* specifies where the energy is concentrated in frequency domain. We restrict the values of u and f^* to the following sets:

$$\mathscr{U} = \left\{\frac{C'n}{k}, \frac{2C'n}{k}\ldots,\left(\frac{k}{C'}-1\right)\frac{C'n}{k}, n\right\}$$
$$\mathscr{F} = \left\{k, 2k, \ldots, \left(\frac{n}{k}-1\right)k, n\right\}.$$

The communication game is as follows:

1. Nature selects a *state* U and a *message* F uniformly from $\mathscr{U}$ and $\mathscr{F}$, respectively.

2. An *encoder* maps (U, F) to the signal X according to (1.25).

3. A *state-dependent channel* adds independent $\mathrm{CN}(0, \alpha)$ noise to X_t for each $t \in \{u, \dots, u + \frac{C'n}{k} - 1\}$, while keeping the other entries noiseless. This is written as $Y_t = X_t + W_t$, where

$$W_t \sim \begin{cases} \mathrm{CN}(0, \alpha) & \{u, \dots, u + \frac{C'n}{k} - 1\} \\ 0 & \text{otherwise.} \end{cases} \tag{1.26}$$

   The channel output is given by $Y = \{Y_t\}_{t \in \mathscr{A}}$ for the sampling locations $\mathscr{A}$.

4. A *decoder* receives U and Y, applies (k_0, k_1)-block sparse recovery to Y to obtain a signal $\widehat{Y}'$, and then selects F' to be the frequency $f' \in \mathscr{F}$ such that the energy in $\widehat{Y}'$ within the length-k window centered at f' is maximized:

$$F' = \mathrm{argmax}_{f' \in \mathscr{F}} \| \widehat{Y}'_{I_k(f')} \|_2^2, \tag{1.27}$$

   where $I_k(f') = \{f' + \Delta : \Delta \in [k]\}$.

We observe that if adaptivity were allowed, then the knowledge of U at the decoder would make the block-sparse recovery easy – one could let all of the samples lie within the window given in (1.25). The problem is that we are in the *non-adaptive* setting, and hence we must take enough samples to account for all of the possible choices of U.

We denote the subset of $\mathscr{A}$ falling into $\left\{u, \dots, u + \frac{C'n}{k} - 1\right\}$ by $\mathscr{A}_u$, its cardinality by m_u and the total number of measurements by $m = |\mathscr{A}| = \sum_u m_u$. Moreover, we let $X_{\mathscr{A}_u}$ and $Y_{\mathscr{A}_u}$ denote the sub-vectors of X and Y indexed by $\mathscr{A}_u$.

Information-theoretic analysis: We first state the following lemma.

Lemma 1.6.5. (Mutual information bound) *In the setting described above, the conditional mutual information $I(F; Y|U)$ satisfies $I(F; Y|U) \le \frac{C'm}{k} \log\left(1 + \frac{1}{\alpha}\right)$, where α is variance of the additive Gaussian noise within $\mathscr{A}_u$.*

Proof. We have

$$\begin{aligned} I(F;Y|U) &= \frac{C'}{k}\sum_u I(F;Y|U=u) \\ &= \frac{C'}{k}\sum_u I(F;Y_{\mathscr{A}_u}|U=u) \\ &\le \frac{C'}{k}\sum_u I(X_{\mathscr{A}_u};Y_{\mathscr{A}_u}|U=u) \\ &\le \frac{C'}{k}\sum_u \sum_{t\in\mathscr{A}_u} I(X_t;Y_t|U=u) \\ &\le \frac{C'}{k}\sum_u m_u \log_2\left(1+\frac{1}{\alpha}\right) \\ &= \frac{C'm}{k}\log_2\left(1+\frac{1}{\alpha}\right), \end{aligned} \tag{1.28}$$

where:

- Line 1 follows since U is uniform on a set of cardinality $\frac{k}{C'}$;
- Line 2 follows since given $U=u$, only the entries of Y indexed by $\mathscr{A}_u$ are dependent on F (*cf.*, (1.25));
- Line 3 follows by noting that given $U=u$ we have the Markov chain $F \to X_{\mathscr{A}_u} \to Y_{\mathscr{A}_u}$, and applying the data processing inequality (Lemma 1.6.3);
- Line 4 follows from Lemma 1.6.4 and (1.26), where the conditional independence assumption holds because we have assumed the random variables W_t are independent;
- Line 5 follows from the Shannon-Hartley Theorem (Lemma 1.6.2); in our case, the signal power is exactly one by (1.25), and the average noise energy is α by construction.

□

Next, defining $\delta_u := \Pr[F' \neq F \,|\, U=u]$, Fano's inequality (Lemma 1.6.1) gives

$$I(F;Y|U=u) \ge -1 + (1-\delta_u)\log_2\frac{n}{k},$$

and averaging both sides over U gives

$$I(F;Y|U) \ge -1 + (1-\delta)\log_2\frac{n}{k},$$

where $\delta := \mathbb{E}[\delta_U] = \Pr[F' \neq F]$.

Hence, and by Lemma 1.6.5, if we can show that our ℓ_2/ℓ_2-error guarantee (1.24) gives $F' = F$

with constant probability, then we can conclude that,

$$m \geq \frac{k\big((1-\delta)\log_2 \frac{n}{k} - 1\big)}{C' \log_2\big(1+\frac{1}{\alpha}\big)} = \Omega\bigg(k \log \frac{n}{k}\bigg).$$

We therefore conclude the proof of Theorem 1.1.2 by proving the following lemma.

Lemma 1.6.6. (Probability of error characterization) *Fix integers n, k_0, k_1 and $C > 0$. If the (k_0, k_1)-block sparse recovery algorithm used in the above communication game satisfies* (1.24) *with probability at least $\frac{1}{2}$, then there exist choices of C' and α such that the decoder's estimate of F' according to* (1.27) *satisfies $F' = F$ with probability at least $\frac{1}{4}$.*

Proof. By the choice of estimator in (1.27), it suffices to show that $\widehat{Y}'$, the output of the block-sparse Fourier transform algorithm, has more than half of its energy within the length-k window $I_k(f^*)$ centered of f^*. We show this in three steps.

Characterizing the energy of $\widehat{X}$ within $I_k(f^*)$: The Fourier transform of X in (1.25) is a shifted sinc function of "width" $\frac{k}{C'}$ centered at f^* when the time window is centered at zero, and more generally, has the same magnitude as this sinc function. Hence, by letting C' be suitably large, we can ensure that an arbitrarily high fraction of the energy of $\widehat{X}$ falls within the length-k window centered at $f^* \in \mathscr{F}$. Formally, we have

$$\|\widehat{X}_{I_k(f^*)}\|_2^2 \geq (1-\eta)\|\widehat{X}\|_2^2 \tag{1.29}$$

for $\eta \in (0,1)$ that we can make arbitrarily small by choosing C' large.

Characterizing the energy of $\widehat{Y}$ within $I_k(f^*)$: We now show that, when the noise level α in (1.26) it sufficiently small, the energy in $\widehat{Y}$ within $I_k(f^*)$ is also large with high probability:

$$\sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 \geq (1-2\eta)\|\widehat{X}\|_2^2. \tag{1.30}$$

To prove this, we first note that $|\widehat{Y}_f|^2 = |\widehat{X}_f + \widehat{W}_f|^2$ for all $f \in [n]$, from which it follows that

$$\bigg| \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 - \sum_{f \in I_k(f^*)} |\widehat{X}_f|^2 \bigg| \leq \sum_{f \in I_k(f^*)} |\widehat{W}_f|^2 + 2 \sum_{f \in I_k(f^*)} |\widehat{X}_f| \cdot |\widehat{W}_f|.$$

Upper bounding the summation over $|\widehat{W}_f|^2$ by the total noise energy, and upper bounding the summation over $|\widehat{X}_f| \cdot |\widehat{W}_f|$ using the Cauchy-Schwarz inequality, we obtain

$$\bigg| \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 - \sum_{f \in I_k(f^*)} |\widehat{X}_f|^2 \bigg| \leq \|\widehat{W}\|_2^2 + 2\|\widehat{X}\|_2 \cdot \|\widehat{W}\|_2. \tag{1.31}$$

We therefore continue by bounding the *total* noise energy $\|\widehat{W}\|_2^2$; the precise distribution of the noise across different frequencies is not important for our purposes.

Recall that every non-zero entry of X has magnitude one, and every non-zero time-domain entry of W is independently distributed as $\mathrm{CN}(0,\alpha)$. Combining these observations gives $\mathbb{E}\left[\|W\|_2^2\right] = \alpha\|X\|_2^2$, or equivalently $\mathbb{E}\left[\|\widehat{W}\|_2^2\right] = \alpha\|\widehat{X}\|_2^2$ by Parseval. Therefore, by Markov's inequality, we have $\|\widehat{W}\|_2^2 \le 4\alpha\|\widehat{X}\|_2^2$ with probability at least $\frac{3}{4}$. When this occurs, (1.31) gives,

$$\left|\sum_{f\in I_k(f^*)}|\widehat{Y}_f|^2 - \sum_{f\in I_k(f^*)}|\widehat{X}_f|^2\right| \le 4(\alpha+\sqrt{\alpha})\|\widehat{X}\|_2^2. \tag{1.32}$$

If we choose $\alpha = \frac{\eta^2}{100}$, then we have $4(\alpha+\sqrt{\alpha}) = \frac{\eta^2}{25} + \frac{4\eta}{10} \le \eta$. In this case, by (1.29) and (1.32), the length-k window $I_k(f^*)$ centered at f^* satisfies (1.30).

Characterizing the energy of $\widehat{Y}'$ within $I_k(f^*)$: The final step is to prove that (1.30) and (1.24) imply the following with constant probability for a suitable choice of η:

$$\sum_{f\in I_k(f^*)}|\widehat{Y}'_f|^2 > \frac{1}{2}\|\widehat{Y}'\|_2^2, \tag{1.33}$$

where $\widehat{Y}'$ is the output of the block-sparse recovery algorithm. This clearly implies that $F = F'$, due to our choice of estimator in (1.27).

As a first step towards establishing (1.33), we rewrite (1.30) as

$$\sum_{f\in[n]\setminus I_k(f^*)}|\widehat{Y}_f|^2 \le \|\widehat{Y}\|_2^2 - (1-2\eta)\|\widehat{X}\|_2^2. \tag{1.34}$$

We can interpret (1.34) as an error term $\|\widehat{Y}-\widehat{Y}^*\|_2^2$ for a signal $\widehat{Y}^*$ coinciding with $\widehat{Y}$ within $I_k(f^*)$ and being zero elsewhere. Since $I_k(f^*)$ contains k contiguous elements, this signal is (k_0,k_1)-block sparse, and hence if the guarantee in (1.24) holds, then combining with (1.34) gives

$$\|\widehat{Y}-\widehat{Y}'\|_2^2 \le C\Big(\|\widehat{Y}\|_2^2 - (1-2\eta)\|\widehat{X}\|_2^2\Big). \tag{1.35}$$

We henceforth condition on both (1.24) and the above-mentioned event $\|\widehat{W}\|_2^2 \le 4\alpha\|\widehat{X}\|_2^2$. Since the former occurs with probability at least $\frac{1}{2}$ by assumption, and the latter occurs with probability at least $\frac{3}{4}$, their intersection occurs with probability at least $\frac{1}{4}$.

Next, we write the conditions in (1.30) and (1.35) in terms of $\|\widehat{Y}\|_2^2$, rather than $\|\widehat{X}\|_2^2$. Since $\widehat{X} = \widehat{Y} - \widehat{W}$, we can use the triangle inequality to write $\|\widehat{X}\|_2 \ge \|\widehat{Y}\|_2 - \|\widehat{W}\|_2$, and combining this with $\|\widehat{W}\|_2^2 \le 4\alpha\|\widehat{X}\|_2^2$, we obtain $\|\widehat{X}\|_2 \ge \frac{\|\widehat{Y}\|_2}{1+2\sqrt{\alpha}}$. Hence, we can weaken (1.30) and (1.35) to

$$\sum_{f\in I_k(f^*)}|\widehat{Y}_f|^2 \ge \frac{1-2\eta}{(1+2\sqrt{\alpha})^2}\|\widehat{Y}\|_2^2 \ge 0.99\|\widehat{Y}\|_2^2 \tag{1.36}$$

$$\|\widehat{Y}-\widehat{Y}'\|_2^2 \le C\Big(1-\frac{1-2\eta}{(1+2\sqrt{\alpha})^2}\Big)\|\widehat{Y}\|_2^2 \le 0.01\|\widehat{Y}\|_2^2, \tag{1.37}$$

where the second step in each equation holds for sufficiently small η due to the choice $\alpha = \frac{\eta^2}{100}$.

It only remains to use (1.36)–(1.37) to bound the left-hand side of (1.33). To do this, we first note that by interpreting both (1.36) and (1.37) as bounds on $\|\widehat{Y}\|_2^2$, and using $\|\widehat{Y} - \widehat{Y}'\|_2^2 \geq \|(\widehat{Y} - \widehat{Y}')_{I_k(f^*)}\|_2^2$ in the latter, we have

$$\sum_{f \in I_k(f^*)} |\widehat{Y}_f - \widehat{Y}'_f|^2 \leq 0.02 \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2,$$

since $\frac{0.01}{0.99} \leq 0.02$. Taking the square root and applying the triangle inequality to the ℓ_2-norm on the left-hand side, we obtain

$$\sum_{f \in I_k(f^*)} |\widehat{Y}'_f|^2 \geq (1 - \sqrt{0.02})^2 \sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2. \tag{1.38}$$

Next, writing $\|\widehat{Y}'\|_2 = \|\widehat{Y} + (\widehat{Y}' - \widehat{Y})\|_2$, and applying the triangle inequality followed by (1.37), we have $\|\widehat{Y}'\|_2 \leq 1.1\|\widehat{Y}\|_2$, and hence $\|\widehat{Y}\|_2 \geq 0.9\|\widehat{Y}'\|_2$. Squaring and substituting into (1.36), we obtain,

$$\sum_{f \in I_k(f^*)} |\widehat{Y}_f|^2 \geq 0.8\|\widehat{Y}'\|_2^2. \tag{1.39}$$

Finally, combining (1.38) and (1.39) yields (1.33), and we have thus shown that (1.33) holds (and hence $F = F'$) with probability at least $\frac{1}{4}$. □

2 Dimension-independent Sparse Fourier Transform

This chapter is based on a joint work with Michael Kapralov and Ameya Velingker. It has been accepted to the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (Kapralov et al., 2019, SODA).

2.1 Introduction

The Discrete Fourier Transform (DFT) is one of the most widely used computational primitives in modern computing, with numerous applications in data analysis, signal processing, and machine learning. The fastest algorithm for computing the DFT is the Fast Fourier Transform (FFT) algorithm of Cooley and Tukey, which has been recognized as one of the 10 most important algorithms of the 20th century (Cipra, 2000). The FFT algorithm is very efficient: it computes the Discrete Fourier Transform of a length N complex-valued signal in time $O(N \log N)$. This applies to vectors in any dimension: FFT works in $O(N \log N)$ time irrespective of whether the DFT is on the line, on a $\sqrt{N} \times \sqrt{N}$ grid, or is in fact the Hadamard transform on $\{0,1\}^d$, with $d = \log_2 N$.

In any applications of the Discrete Fourier Transform, the input signal $x \in \mathbb{C}^N$ often satisfies *sparsity* or *approximate sparsity* constraints: the Fourier transform $\widehat{x}$ of x has a small number of coefficients k or is close to a signal with a small number of coefficients (e.g., this phenomenon is the motivation for compression schemes such as JPEG and MPEG). This has motivated a rich line of work on the *Sparse FFT* problem: given access to a signal $x \in \mathbb{C}^N$ in time domain that is sparse in Fourier domain, compute the k nonzero coefficients in *sublinear* (i.e., $o(N)$) time.

Very efficient algorithms for the Sparse FFT problem have been developed in the literature (Goldreich and Levin, 1989; Kushilevitz and Mansour, 1993; Mansour, 1995; Gilbert et al., 2002; Akavia et al., 2003; Gilbert et al., 2005; Iwen, 2010; Akavia, 2010; Hassanieh et al., 2012c,b; Lawlor et al., 2013; Boufounos et al., 2015; Hassanieh et al., 2012a; Pawar and Ramchandran, 2013; Heider et al., 2013; Indyk et al., 2014; Indyk and Kapralov, 2014; Kapralov, 2016; Price

and Song, 2015; Chen et al., 2016; Cevher et al., 2017; Kapralov, 2017; Nakos et al., 2019; Amrollahi et al., 2019). The state-of-the-art approach, due to Hassanieh et al. (2012b), yields an $O(k \log N)$ runtime algorithm for the following exact k-sparse Fourier transform problem: given access to an input signal of length N whose Fourier transform has at most k non-zeros, output the non-zero coefficients and their values. This highly efficient algorithm comes with a caveat, however: the runtime of $O(k \log N)$ only holds for the Fourier transform on the line, namely, $\mathbb{Z}_N$. The algorithm naturally extends to higher dimensions, namely, $\mathbb{Z}_n^d$, where $N = n^d$, but with an exponential loss in runtime; the runtime becomes $O(k \log^d N)$ as opposed to $O(k \log N)$. Interestingly, the other extreme of $d = \log_2 N$, i.e., the Hadamard transform, has been known to admit an $O(k \log N)$ algorithm since the seminal work of Goldreich and Levin (1989). The recent work of Amrollahi et al. (2019) yields a sample optimal sparse Hadamard transform in sublinear time. However, all intermediate values of d exhibit a *curse of dimensionality*. This is in sharp contrast to FFT itself, which runs in time $O(N \log N)$, where $N = n^d$ is the length of the input signal, in *any dimension* d. The focus of our work is to design sublinear time algorithms for Sparse FFT that avoid this curse of dimensionality. Our main point of attention is the Sparse FFT problem:

$$\begin{aligned} \textbf{Input:} \quad & \text{access to } x : [n]^d \to \mathbb{C}, \\ & \text{integer } k \geq 1 \text{ such that } |\operatorname{supp} \widehat{x}| \leq k \\ \textbf{Output:} \quad & \text{nonzero elements of } \widehat{x} \text{ and their coefficients} \end{aligned} \tag{2.1}$$

Our main result is the first sublinear algorithm for exact Sparse FFT (2.1), as stated in the following theorem.

Theorem 2.1.1 (Main result, informal version)**.** *For any integer n that is a power of two and any positive integer d, there exists a deterministic algorithm that, given access to a signal $x : [n]^d \to \mathbb{C}$ with $\|\widehat{x}\|_0 \leq k$, recovers $\widehat{x}$ in time* $\operatorname{poly}(k, \log N)$.

We remark that this is the first sublinear time Sparse FFT algorithm that avoids an exponential dependence on the dimension d. One should note that the runtime still depends on d, since $\log_2 N = d \log_2 n$ is lower bounded by d, but this dependence is polynomial as opposed to exponential.

2.1.1 Significance of our results and related work

Significance of our results. The state of the art in high dimensional Sparse Fourier Transforms presents an interesting conundrum: algorithms with runtime $O(k \log N)$ are known for $d = 1$ (Discrete Fourier Transform on the line, see (Hassanieh et al., 2012b)) and $d = \log_2 N$ (the Hadamard transform, see (Goldreich and Levin, 1989; Amrollahi et al., 2019)), but for all intermediate values of d the runtime scales exponentially in d. Given that FFT itself is dimension-insensitive, this strongly suggests that exciting new algorithmic techniques can be developed for the high-dimensional version of the problem. We design the first approach to high dimensional Sparse FFT that does not suffer from the curse of dimensionality, and

naturally leads to several exciting open problems that we hope will spur further progress in this area.

In addition, we note that rather high-dimensional versions of the Fourier transform arise in applications (e.g., 2D, 3D and 4D-NMR in medical imaging), and designing practical Sparse FFT algorithms for this regime is an important problem. We hope that new techniques for dimension-independent Sparse FFT will lead to progress in this direction as well.

Sample complexity of high-dimensional Sparse FFT. We note that, besides runtime, another very important parameter of a Sparse FFT algorithm is *sample complexity*, i.e., the number of samples that an algorithm needs to access in time domain in order to compute the dominant coefficients of the Fourier transform. The sample complexity of Sparse FFT, unlike runtime, does not suffer from a curse of dimensionality. Indeed, there exist several algorithms with $\widetilde{O}(N)$ runtime that can recover the top k coefficients of $\widehat{x}$ using only $k\,\mathrm{poly}(\log N)$ accesses in time domain, irrespective of the dimensionality of the problem. This can be achieved, for example, using either results on the restricted isometry property (RIP) (Candès and Tao, 2006; Rudelson and Vershynin, 2008; Bourgain, 2014; Cheraghchi et al., 2013; Haviv and Regev, 2017), or using the filtering approach developed in the Sparse FFT literature, with $\widetilde{O}(N)$ decoding time. Thus, the challenge is to achieve sublinear *runtime* without an exponential dependence on the dimension.

We now outline existing approaches to Sparse FFT and explain why they fail to scale well in high dimensions:

State-of-the-art approaches to Sparse FFT and their lack of scalability in high dimensions. The main idea behind many recently developed algorithms for the Sparse FFT problem is the "hashing" approach inherited from sparse recovery with arbitrary linear measurements. Given access to a signal $x : [n]^d \to \mathbb{C}$, one designs linear measurements of x that allow one to "hash" the nonzero positions of $\widehat{x}$ into a number of "buckets." The number of buckets $B = b^d$ is chosen to be a constant factor larger than the sparsity k to ensure that a large constant fraction of the nonzero positions of $\widehat{x}$ are isolated in their buckets. Every isolated element can be recovered and subtracted from x for future iterations of the same hashing scheme, thereby ensuring convergence. The idea of hashing is implemented via filtering: one designs a filter $G : [n]^d \to \mathbb{C}$ such that $\widehat{G}$ approximates a "bucket," i.e., $\widehat{G}$ is close to 1 on an ℓ_∞ ball of side length $\approx (N/B)^{1/d} = n/b$ in dimension d. The content of the $\boldsymbol{j}$-th 'bucket', for $\boldsymbol{j} \in [b]^d$, is then

$$\widehat{(x_{\cdot - a} \cdot G)}_{\boldsymbol{j} \cdot n/b} = \sum_{\boldsymbol{f} \in [n]^d} \widehat{x}_{\boldsymbol{f}} e^{2\pi \boldsymbol{f}^T a/n} \cdot \widehat{G}_{\boldsymbol{j} \cdot n/b - \boldsymbol{f}}. \tag{2.2}$$

Since $\widehat{G}$ is essentially 1 on the ℓ_∞ ball around the center $\boldsymbol{j} \cdot n/b$ of the 'bucket' and essentially zero outside, (2.2) gives the algorithm time domain access to the restriction of $\widehat{x}$ to the "bucket," i.e., the essential support of $\widehat{G}$, where $a \in [n]^d$ is the location in time domain at which the

signal is being accessed. A pseudorandom permutation of the frequency space ensures that such a bucket is likely to contain just a single element of the support, which enables the algorithm to recover at least a constant fraction of elements in a single round and perform iterative recovery. Furthermore, if the (essential) support of G in time domain is small, one obtains an efficient algorithm.

The difficulty that arises in using (2.2) in high dimensions is the fact that it is not known how to ensure that $\widehat{G}$ is close to 1 in an appropriately defined "bucket" while simultaneously ensuring that $|\text{supp } G|$ is small. For example, the filters constructed in Hassanieh et al. (2012b) ensure that $\widehat{G}$ is polynomially close to 1 in Fourier domain, but this comes at the expense of $|\text{supp } G|$ being larger than k (the ideal support size) by a factor of $\Theta(\log n)$, and this effect is even more pronounced in higher dimensions, resulting in a $\log^d n$ loss in runtime. The other extreme would be to choose G to be equal to 1 on an ℓ_∞ ball with k points around the origin, but in that case, its Fourier transform $\widehat{G}$ is the sinc function, which is only a constant factor approximation to the indicator of the corresponding ℓ_∞ box in Fourier domain (i.e., the ideal "bucket"). In dimension d, the approximation degrades to c^d for some constant $c \in (0,1)$, leading to exponential loss in runtime. Indeed, suppose that all elements of $\widehat{x}$ have roughly the same value. Then for a given element $\boldsymbol{f} \in \text{supp } \widehat{x}$, the expected contribution of other elements to the noise in the "bucket" that $\boldsymbol{f}$ is hashed to is $||\widehat{x}||_2^2/B$, but the contribution of $\widehat{x}_{\boldsymbol{f}}$ to its own bucket is (most of the time) only c^d of its value, and, hence, only an exponentially small fraction of coefficients can be recovered in a given round of hashing. [1]

Related work. In (Cheraghchi and Indyk, 2017), the authors presented a deterministic Sparse Fourier transform algorithm for the Hadamard transform, i.e., $d = \log_2 N$, that runs in nearly linear time in the sparsity parameter k, but it is not known how this extends to lower dimensions. In (Iwen, 2010, 2013) the author gives a $\widetilde{O}(k^2)$ time deterministic algorithm for the Sparse Fourier Transform, but the algorithm only applies to a related but distinctly easier problem. Specifically, the problem considers a continuous function on $[0, 2\pi)$ whose Fourier transform is bandlimited and sparse. The presented algorithm requires sampling the signal at arbitrary locations in $[0, 2\pi)$. A natural approach is to emulate sampling off-grid (i.e., at arbitrary points in $[0, 2\pi)$) given discrete samples that we have access to, which is achieved in (Merhi et al., 2019) giving an $\widetilde{O}(k^2)$ time deterministic algorithm for one dimensional sparse FFT. But this is a challenging task in multi-dimensional setting for several reasons. First, we are operating under the sparsity assumption alone, and no powerful general interpolation techniques that work under the sparsity assumption alone are available, to the best of our knowledge. Furthermore, even if the function were bandlimited, a natural approach to interpolation would involve some form of Taylor expansion or semi-equispaced Fourier Transform, however, both approaches incur a $\log^d N$ loss in dimension d. Indeed, similar exponential

[1] In addition, the discussion above assumes the presence of an approximate pairwise hashing lemma for high dimensions that does not lose an exponential factor in the dimension (it is known that such a lemma holds with at most about a factor of 2^d loss (Indyk and Kapralov, 2014), but no dimension-independent version is available in the literature).

dependence on the dimensionality of the problem manifests itself in Fast Multipole Methods of Greengard and Rokhlin (1986) and the Sparse FFT algorithms mentioned above. Finally, one should also note that whereas the problem of computing the Fourier transform on a $p \times q$ grid with p mutually prime with q is equivalent to a one-dimensional Fourier transform on $\mathbb{Z}_{pq}$, the standard case of side lengths that are powers of two (for which we have the most efficient FFT algorithms) does not admit such a reduction. Furthermore, such a reduction appears to be quite challenging in high dimensions for reasons outlined above, and even more so for highly oscillatory functions that Sparse FFT algorithms need to handle.

2.2 Overview of Our Results and Techniques

Prior works on Sparse FFT have primarily focused on efficiently implementing hashing-based ideas developed in the extensive literature on sparse recovery using general linear measurements, e.g., (Ghazi et al., 2013), which meets with several difficulties. In particular, the presence of multiplicative subgroups in $\mathbb{Z}_n^d$ has been a hurdle in analyzing Sparse FFT algorithms: while aliasing filters have optimal performance from the point of view of the uncertainty principle, their applications have been limited due to the fact that frequencies that belong to the same subgroup get hashed together if such filters are used, making it impossible to reason about isolation of individual frequencies. At the same time, FFT itself owes much of its efficiency to the very same multiplicative subgroups of $\mathbb{Z}_n^d$, and a natural question is whether one can design a Sparse FFT algorithm that operates on similar principles. This is precisely the approach that we take.

Adaptive Aliasing Filters. The main technical innovation that allows us to avoid exponential dependence on the dimension and obtain Theorem 2.1.1 is a new family of filters for isolating a subset of frequencies in the Fourier domain for a sparse signal $\widehat{x}$ using few samples in time domain. We refer to the family of filters as *adaptive aliasing filters*.

Definition 2.2.1 (($\boldsymbol{f}, S$)-isolating filter, informal version of Definition 2.4.4, see Section 2.4)**.** Suppose n is a power of two integer and $S \subseteq [n]^d$ for a positive integer d. Then, for any frequency $\boldsymbol{f} \in S$, a filter $G : [n]^d \to \mathbb{C}$ is called $(\boldsymbol{f}, S)$*-isolating* if $\widehat{G}_{\boldsymbol{f}} = 1$ and $\widehat{G}_{\boldsymbol{f}'} = 0$ for every $\boldsymbol{f}' \in S \setminus \{\boldsymbol{f}\}$.

We explain the intuition behind the construction of the filter in Section 2.2.1 below and provide the details later in Section 2.4.

The reason why an $(\boldsymbol{f}, S)$-isolating filter G is useful lies in the fact that for every signal $x : [n]^d \to \mathbb{C}$ with supp $\widehat{x} \subseteq S$ we have, for all $\boldsymbol{t} \in [n]^d$,

$$\sum_{\boldsymbol{j} \in [n]^d} x_{\boldsymbol{j}} G_{\boldsymbol{t}-\boldsymbol{j}} = (x * G)_{\boldsymbol{t}} = \frac{1}{N} \sum_{\boldsymbol{j} \in [n]^d} \widehat{x}_{\boldsymbol{j}} \cdot \widehat{G}_{\boldsymbol{j}} \cdot e^{2\pi i \frac{\boldsymbol{j}^T \boldsymbol{t}}{n}} = \frac{1}{N} \widehat{x}_{\boldsymbol{f}} e^{2\pi i \frac{\boldsymbol{f}^T \boldsymbol{t}}{n}}.$$

Thus, the filter G enables access to the time domain representation of the restriction of $\widehat{x}$ to $\boldsymbol{f}$

using a running time proportional to |supp G|. Of course, this is only useful if the support of G is small. The main technical lemma of our paper shows that for every support set $S = \text{supp}\,\widehat{x}$, there exists an $\boldsymbol{f} \in S$ that can be isolated efficiently:

Lemma 2.2.1 (Informal version of Corollary 2.4.2 in Section 2.4)**.** *For every power of two $n \geq 1$, positive integer d, and set $S \subseteq [n]^d$, there exists an $\boldsymbol{f} \in S$ and an $(\boldsymbol{f}, S)$-isolating filter G such that* |supp G| $\leq |S|$.

The proof of the lemma uses Kraft–McMillan inequality and is given in Section 2.4.

Accessing the residual signal. Lemma 2.2.1 suggests a natural approach to the estimation problem with Fourier measurements in high dimensions: iteratively construct an $(\boldsymbol{f}, S)$-isolating filter G, estimate $\boldsymbol{f}$, remove $\boldsymbol{f}$ from S, and proceed. The hope is that we can essentially assume that we are given access to $\mathscr{F}^{-1}\{\widehat{x}_{S\setminus\{\boldsymbol{f}\}}\}$ once we have estimated $\boldsymbol{f}$. In general, if we have been able to estimate the values of $\widehat{x}_{\boldsymbol{f}}$ for all $\boldsymbol{f} \in C$ with some $C \subseteq S$, then we would like to obtain access to,

$$x_{\boldsymbol{t}} - \sum_{\boldsymbol{f}\in C} \widehat{x}_{\boldsymbol{f}} \cdot e^{2\pi i \boldsymbol{f}^T \boldsymbol{t}}.$$

Note that we would need $x_{\boldsymbol{t}}$ for $\boldsymbol{t}$ in the support of G at the next iteration, and this support is generally a rather complicated set of size $\Omega(k)$, from which we need to subtract the inverse Fourier transform of the signal estimated so far. This problem is the non-uniform Fourier transform problem, and no subquadratic methods for subtraction are known even in dimension $d = 1$ when the set in time domain that we want to compute the inverse Fourier transform on is arbitrary. Even if the target set is an ℓ_∞-box, the best known algorithms for this problem run in time $\Omega(k \log^d(1/\epsilon))$, where $\epsilon > 0$ is the precision parameter of the computation—this reduces to quadratic time even when $d = \Omega(\log k / \log\log k)$ and inverse polynomial in k precision is desired. Thus, subtracting from time domain would result in at least cubic runtime in k. Instead, we subtract the influence of the residual in frequency domain, which requires $O(k)$ evaluations of $\widehat{G}$ (as we show, $\widehat{G}$ can be evaluated at a cost of just $O(\log N)$). Note that it is crucial here that we peel off one coefficient at at time. Any improvements to this process, if they were to achieve $k^{2-\Omega(1)}$ runtime overall, would likely also imply improvements in the computation of approximate *non-uniform* Fourier transform: given a k-sparse signal $\widehat{x}$ and a set $T \subseteq [n]^d$ with $|T| \leq k$, output $y : [n]^d \to \mathbb{C}$ such that $||(x-y)_T||_2^2 \leq \epsilon ||x||_2^2$. However, it seems plausible that quadratic runtime in k is essentially optimal for the non-uniform Fourier transform problem: specifically, that under natural complexity theoretic assumptions there exists no algorithm for the ϵ-approximate non-uniform Fourier transform problem with runtime $k^{2-\Omega(1)}$ when $d = \Omega(\log k)$ and $\epsilon < 1/k^C$ for sufficiently large constant C. We note that current techniques do not provide a subquadratic algorithm even for simple sets T such as the ℓ_∞ box with k points in dimension $d = \Omega(\log k / \log\log k)$ (due to the $k \log^d(1/\epsilon)$ dependence mentioned above; a similar exponential dependence on the dimension is present in Fast Multipole Methods of Greengard and Rokhlin (1986)). For an arbitrary set T no subquadratic algorithm is known even when $d = 1$.

Putting it together: estimation with Fourier measurements. Combining the aforementioned ideas, we are able to develop a deterministic algorithm for the *estimation problem with Fourier measurements* in high dimensions:

$$\begin{aligned} \textbf{Input:} \quad & \text{access to } x : [n]^d \to \mathbb{C}, \\ & \text{subset } S \subseteq [n]^d \text{ such that supp } \widehat{x} \subseteq S \\ \textbf{Output:} \quad & \widehat{x}_S \end{aligned} \tag{2.3}$$

For the estimation problem (2.3) we obtain the following result.

Theorem 2.2.1 (Estimation guarantee)**.** *Suppose n is a power of two integer, d is a positive integer, and $S \subseteq [n]^d$. Then, for any signal $x : [n]^d \to \mathbb{C}$ with* supp $\widehat{x} \subseteq S$*, the procedure* ESTIMATE(x, S, n, d) *(see Algorithm 9) recovers $\widehat{x}$. Moreover, the sample complexity of this procedure is $O(|S|^2)$ and its runtime is $O(|S|^2 \cdot \log N)$. Furthermore,* ESTIMATE *is deterministic.*

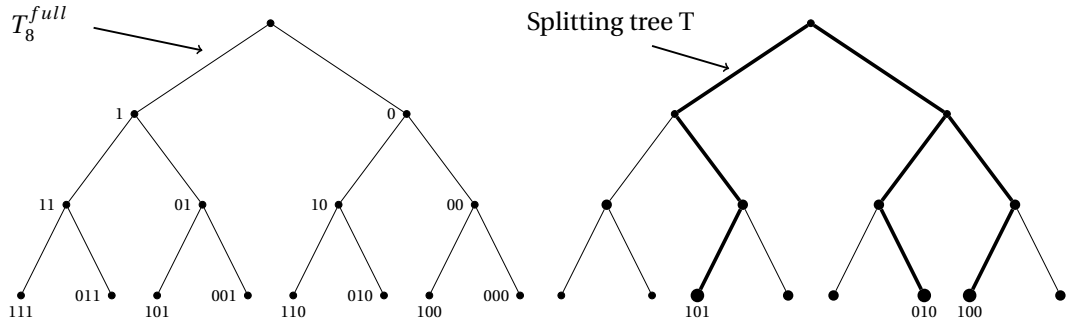
In the rest of this section, we give an overview of our techniques. Throughout the section, we present our results for the one-dimensional setting, as this makes notation simpler. All our results translate to the high-dimensional setting without any loss—see Section 2.4.2 for details.

2.2.1 Recovery via adaptive aliasing filters

Our main theorem is the following, which proves the existence of an efficient algorithm for problem (2.1) for worst-case signals.

Theorem 2.1.1 (Sparse FFT for worst-case signals, formal version)**.** *For any power of two integer n and any positive integer d and any signal $x : [n]^d \to \mathbb{C}$ with $\|\widehat{x}\|_0 = k$, the procedure* SPARSEFFT(x, n, d, k) *in Algorithm 11 recovers $\widehat{x}$. Moreover, the sample complexity of this procedure is $O(k^3 \log^2 k \log^2 N)$ and its runtime is $O(k^3 \log^2 k \log^2 N)$, where $N = n^d$.*

The major difference between estimation and recovery (i.e., problem (2.3) vs. (2.1)) is the fact that in the latter problem, the set S of frequencies is unknown to us: the algorithm is only given access to x and an upper bound on the sparsity of $\widehat{x}$. Since our (f, S)-isolating filter is adaptive, i.e., depends on S, this appears to present a challenge. However, we circumvent this challenge by constructing a sequence of successive approximations to the set S. In dimension 1, these approximations amount to reducing S modulo 2^j for all $j = 1, \ldots, \log_2 n$, and adaptively probing to learn which of the residue classes are nonzero. As before, our approach extends seamlessly to high dimensions by simply concatenating the d coordinates into a single vector. Note that this is in sharp contrast to all previously known approaches, which are more efficient in low dimensions, but incur an exponential loss overall. We would like to note that at a high level one can view our filtering approach as a way to prune the FFT computation graph in a way that suffices for recovery of a k-Fourier sparse vector.



(a) T_8^{full} with binary labelling.

(b) A splitting tree of depth 3 with three leaves. Set of leaves $S = \{2, 4, 5\}$.

Figure 2.1 – An example of T_n^{full} and a splitting tree with $n = 8$ and binary labelling.

We outline the main ideas in one-dimensional setting here to simplify the presentation (see Section 2.4.2 for the high-dimensional version of the argument). Let $N = n$ be the length of the signal and $d = 1$ be the dimension for n a power of two. We define T_n^{full} to be a full binary tree of height $\log_2 n$ and define a labelling scheme on the vertices as follows.

Definition 2.2.2. Suppose n is a power of two integer. Let T_n^{full} be a full binary tree of height $\log_2 n$, where for every $j \in \{0, 1, \ldots, \log_2 n\}$, the nodes at level j (i.e., at distance j from the root) are labeled with integers in $\mathbb{Z}_{2^j}$. For a node $v \in T_n^{\text{full}}$, we let f_v be its label. The label of the root is $f_{root} = 0$. The labelling of T_n^{full} satisfies the condition that for every $j \in [\log_2 n]$ and every v at level j, the right and left children of v have labels f_v and $f_v + 2^j$, respectively. Note that the root of T_n^{full} is at level 0, while the leaves are at level $\log_2 n$.

The tree captures the computation graph of FFT algorithm, where leaves correspond to frequencies in $\mathbb{Z}_n$ (given by the label), and for any $j \in \{0, 1, \ldots, \log_2 n\}$, the nodes at level j (i.e., at distance j from the root) correspond to congruence classes of frequencies modulo 2^j, as specified by the labelling (see Figure 2.1a).

Note that the full FFT algorithm starts from the root of T_n^{full} and computes the congruence classes of the Fourier transform of signal x at each level of this tree iteratively. Because it can reuse the computations from each level for computing the next levels, the total runtime of FFT is $O(n \log_2 n)$.

In order to speed up the computation for sparse signals, we introduce the notion of a *splitting tree*, which is nothing but the subtree of T_n^{full} that contains the nonzero locations of $\widehat{x}$ together with paths connecting them to the root. Given a set $S \subseteq [n]$ (the support of $\widehat{x}$ in the Fourier domain), we define the *splitting tree* of set S as follows:

Definition 2.2.3 (Splitting tree)**.** Let n be an integer power of two. For every $S \subseteq [n]$, the *splitting tree* $T = \text{Tree}(S, n)$ of a set S is a binary tree that is the subtree of T_n^{full} that contains, for every $j \in [\log_2 n]$, all nodes $v \in T_n^{\text{full}}$ at level j such that $\{f \in S : f \equiv f_v \pmod{2^j}\} \neq \emptyset$.

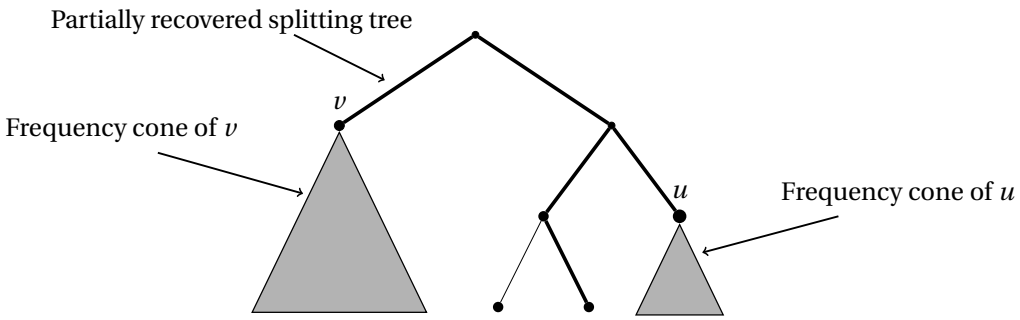


Figure 2.2 – A partially recovered splitting tree (shown in bold). Frequency cones of u and v correspond to the subtrees rooted at nodes u and v, respectively, which have not been discovered yet.

An illustration of such a tree is given in Figure 2.1b. In order to recover the identities of the elements in S, our algorithm performs an exploration of this tree. At every point, the algorithm constructs a filter G that isolates *frequencies in a given subtree* and tests whether that subtree contains a nonzero signal. In order to make this work, we need a construction of filters that isolates the entire subtree as opposed to only one element, as Definition 2.2.1 does. Fortunately, the actual (f, S)-isolating filters G constructed in Lemma 2.2.1 satisfy precisely this property. The stronger isolation properties are captured by the following definition:

Definition 2.2.4 (Frequency cone of a leaf of T)**.** For every power of two n, subtree T of T_n^{full}, and vertex $v \in T$ which is at level $l_T(v)$ from the root, the *frequency cone of v with respect to T* is defined as,

$$\text{FrequencyCone}_T(v) := \left\{ f \in [n] : f \equiv f_v \pmod{2^{l_T(v)}} \right\}.$$

Intuitively, the frequency cone of a node v in T captures all potential nonzeros of $\widehat{x}$ that belong to the subtree of v in T (see Figure 2.2). Our adaptive filter construction lets us obtain time domain access to the corresponding part of the frequency space:

Definition 2.2.5 ((v, T)-isolating filter)**.** For every integer n, subtree T of T_n^{full}, and leaf v of T, a filter $G \in \mathbb{C}^n$ is called (v, T)*-isolating* if the following conditions hold:

- For all $f \in \text{FrequencyCone}_T(v)$, we have $\widehat{G}_f = 1$.
- For every $f' \in \bigcup_{u:\ \text{leaf of } T,\ u \neq v} \text{FrequencyCone}_T(u)$, we have $\widehat{G}_{f'} = 0$.

Note that for every signal $x \in \mathbb{C}^n$ with $\operatorname{supp} \widehat{x} \subseteq \bigcup_{u:\ \text{leaf of } T} \text{FrequencyCone}_T(u)$ and all $t \in [n]$,

$$\sum_{j \in [n]} x_j G_{t-j} = \frac{1}{n} \sum_{f \in \text{FrequencyCone}_T(v)} \widehat{x}_f e^{2\pi i \frac{ft}{n}}.$$

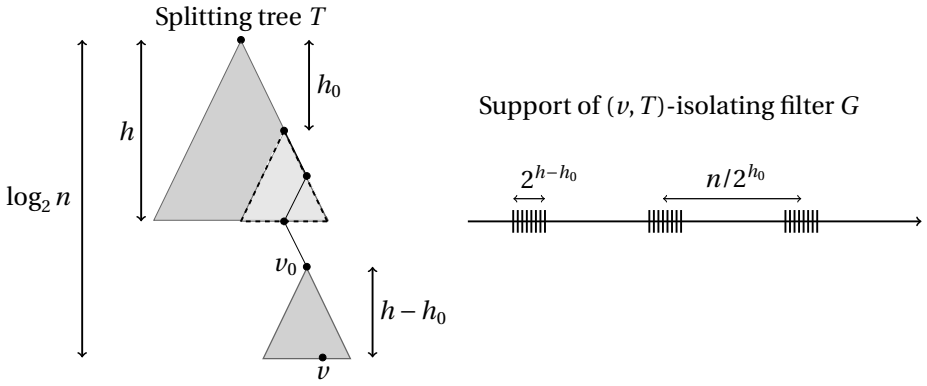


Figure 2.3 – An instance of a splitting tree (on the left) and a (v, T)-isolating filter G (right), where the weight of leaf v is h and hence the filter G satisfies $|\text{supp } G| = 2^{w_T(v)} = 2^h$.

Iterative tree exploration process leading to an algorithm with $\widetilde{O}(k^3)$ runtime. Now that we have defined the framework for our algorithm, we need to specify the order in which the algorithm will be accessing the leaves of the tree in order to minimize runtime. This is governed by the cost of constructing and using a (v, T)-isolating filter for various nodes v in T. To quantify cost, we introduce the notion of a *weight* of a leaf in the tree.

Definition 2.2.6 (Weight of a leaf)**.** Suppose n is a power of two. Let T be a subtree of T_n^{full}. Then for any leaf $v \in T$, we define its *weight* $w_T(v)$ *with respect to* T to be the number of ancestors of v in tree T with two children.

It turns out that the techniques from Lemma 2.2.1 also yield the following.

Lemma 2.2.2 (Informal version of Lemma 2.4.1 in Section 2.4)**.** *Suppose n is a power of two. Let T be a subtree of T_n^{full}. Then for any leaf $v \in T$, there exists a (v, T)-isolating filter G with $|\text{supp } G| \le 2^{w_T(v)}$ such that G and $\widehat{G}$ can be evaluated at $O(\log N)$ cost per point.*

Before describing the algorithm we give an example illustrating filter support in time domain. Consider a complete binary tree T of height $h \ll \log_2 n$. Suppose that v_0 is some vertex at level $h_0 < h$ of this tree. Now we take the subtree rooted at v_0 and move it away from the root by adding an appendage of length $\log_2 n - h$ to v_0. The appendage is a path of $\log_2 n - h$ nodes each of which has a single child. This does not change the weight of any of the leafs of the original tree because every node in the appendage has exactly one child. One can see an example of such tree in Figure 2.3. Suppose that the leaf v is a leaf of the subtree rooted at v_0, which was moved down by the appendage. In order to isolate v from the elements that are not in the subtree of v_0 we need a filter which is $\left(n/2^{h_0}\right)$-periodic in time domain and in order to isolate from the rest of the elements in subtree of v_0 the filter needs to sample the signal at a fine grid of length 2^{h-h_0}. Hence, the support of a (v, T)-isolating filter G is $\text{supp } G = \left\{i + \left(n/2^{h_0}\right) \cdot j;\ j \in \left[2^{h_0}\right], i \in \left[2^{h-h_0}\right]\right\}$. In Fig. 2.3 we exhibit a (v, T)-isolating filter G which is constructed based on Lemma 2.4.1, where v and T correspond to this instance of splitting tree that is described above.

Given Lemma 2.2.2, our algorithm is natural. We find the vertex $v^* = \text{argmin}_{v \in T} w_T(v)$, which, by Kraft's inequality, satisfies $w_T(v^*) \leq \log_2 k$. We then define an auxiliary tree T' by appending a left a and a right child b to v. Then for each of the children a, b, we, in turn, construct a filter G that isolates them from the rest of T (i.e., from the frequency cones of other nodes in T) and check whether the corresponding restricted signals are nonzero. The latter is unfortunately a nontrivial task, since the sparsity of these signals can be as high as k, and detecting whether a k-sparse signal is nonzero requires $\Omega(k)$ samples. However, a fixed set of $k \log^3 N$ locations that satisfies the restricted isometry property (RIP) can be selected, and accessing the signal on those values suffices to test whether it is nonzero. The overall runtime becomes $\widetilde{O}(k^3)$: the isolating filter has support at most $2k$, while the number of samples needed to test whether the two subtrees of v are nonempty is $\widetilde{O}(k)$, so peeling off $\leq k$ elements takes $\widetilde{O}(k^3)$ time overall. This results in Theorem 2.1.1 (the procedure is presented in Algorithm 11).

$\widetilde{O}(k^2)$ runtime under random phase assumption. We note that the runtime can be easily reduced to $\widetilde{O}(k^2)$ if assumptions are made on the signal that ensure that its energy is evenly spread across the time domain, making $\widetilde{O}(1)$ samples sufficient to detect whether the signal is zero or not. This occurs, for instance, if a signal's Fourier spectrum satisfy distributional assumptions (e.g., the values have random phases). We present such a result in Section 2.7. It seems that even under this assumption on the values of the signal, since the support of the signal in the Fourier domain is worst-case, reducing the runtime below k^2 likely requires a major advance in techniques for non-uniform Fourier transform computation.

More formally, we introduce the notion of a *worst-case signal with random phase* as follows:

Definition 2.2.7 (Worst-case signal with random phase)**.** For any positive integer d and power of two n, we define x to be a *worst-case signal with random phase* having values $\{\beta_{\boldsymbol{f}}\}_{\boldsymbol{f} \in [n]^d}$ if,

$$\widehat{x}_{\boldsymbol{f}} = \beta_{\boldsymbol{f}} e^{2\pi i \theta} \quad \text{for uniformly random } \theta \in [0, 2\pi),$$

independently for every $\boldsymbol{f} \in [n]^d$. Furthermore, if k of the values $\{\beta_{\boldsymbol{f}}\}_{\boldsymbol{f} \in [n]^d}$ are nonzero, then x is said to be a *worst-case k-sparse signal with random phase* and is guaranteed to have sparsity $\|\widehat{x}\|_0 = k$.

Note that "worst-case" in the above definition signifies the fact that the *support* of the signal is arbitrary (having no distributional assumptions), subject to a potential sparsity constraint. We then present the following theorem:

Theorem 2.2.2 (Sparse FFT for worst-case signals with random phase)**.** *For any power of two integer n, positive integer d, and worst-case k-sparse signal with random phase $x : [n]^d \to \mathbb{C}$, the procedure* SPARSEFFT-RANDOMPHASE(x, n, d, k) *in Algorithm 12 recovers $\widehat{x}$ with probability $1 - \frac{1}{N^2}$. Moreover, the sample complexity and runtime of this procedure are both $O\left(k^2 \log^4 N\right)$.*

Impossibility of reducing the number of iterations (rounds of adaptivity): signals with low Hamming weight support. We note that our algorithm differs from all prior works in that it uses many rounds of adaptivity. Indeed, the samples that our algorithm takes are guided by values of the signal that have been read in previously queried locations, which is in contrast to most prior Sparse Fourier Transform algorithms. The notable exception in recent literature is our adaptive block Sparse FFT algorithms (Cevher et al., 2017).

Our algorithm uses k rounds of adaptivity, peeling off one element at a time. It would be desirable to reduce the number of rounds of adaptivity by perhaps peeling off many elements in one batch as opposed to one at a time. For example, if the locations of the nonzeros of $\widehat{x}$ are uniformly random in $[n]^d$, then the splitting tree of x is likely to be rather balanced, so perhaps one can find a filter G that has small support and can be efficiently used to isolate many coefficients at once? Indeed, this intuition turns out to be correct for signals with uniformly random supports—we show in (Kapralov et al., 2019) that this idea yields a $\widetilde{O}(k)$ time algorithm. However, rather surprisingly, adversarial instances exist that force the peeling process to use $k^{1-o(1)}$ rounds of adaptivity in the worst case, making our analysis essentially tight. We now present this adversarial instance.

Definition 2.2.8 (Hamming ball)**.** For any power of two integer n any integer $0 \le c \le \log_2 n$, we define H_c^n to be the *closed Hamming ball* of radius c centered at 0:

$$H_c^n = \left\{ f \in [n] : w(f) \le c \right\},$$

where $w(f)$ is the Hamming weight of the binary representation of f, i.e., $w(f)$ is the number of ones in the binary representation of f.

By basic countings, $|H_c^n| = \sum_{j=0}^{c} \binom{\log_2 n}{j}$.

Definition 2.2.9 (Class of signals with low Hamming support)**.** For any power of two integer n and any integer c, Let $\mathcal{X}_c^n$ denote the class of signals in $\mathbb{C}^n$ with support H_c^n as in Definition 2.2.8,

$$\mathcal{X}_c^n = \left\{ x \in \mathbb{C}^n : \operatorname{supp} x \subseteq H_c^n \right\}.$$

Note that for any $x \in \mathcal{X}_c^n$ we have that $\|x\|_0 = \sum_{i=0}^{c} \binom{\log_2 n}{i}$, so for any $c \le \left(\frac{1}{2} - \epsilon\right) \log_2 n$ for any constant $\varepsilon > 0$, the signals in the class $\mathcal{X}_c^n$ are $\Theta\left(\binom{\log_2 n}{c}\right)$-sparse.

Definition 2.2.10 (Low Hamming weight splitting trees)**.** For any n a power of two integer, we define a *low Hamming weight splitting tree* T_c^n inductively for $c = 0, 1, \ldots, \log_2 n$:

1. T_0^n is the unique tree of depth $\log_2 n$ that has a single leaf and satisfies the property that each non-leaf node has a single right child only. Thus, T_0^n has $\log_2(n) + 1$ nodes.
2. For any $c > 0$, T_c^n is constructed as follows: Take T_0^n and label the nodes in order from the root to the leaf as $0, 1, \ldots, \log_2 n$. Then, for each node $0 \le j < \log_2 n$, take a copy of $T_{c-1}^{n/2^{j+1}}$ and append its root as the left child of node j. The resulting tree defines T_c^n.

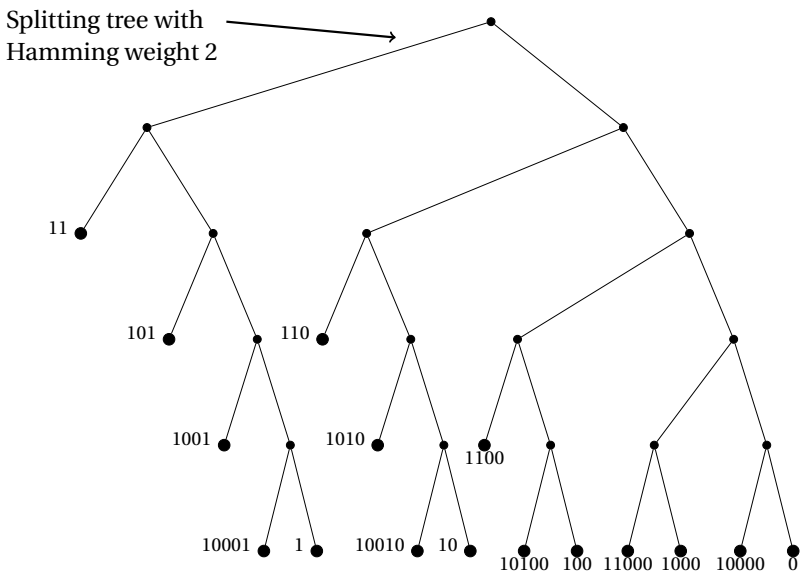


Figure 2.4 – The splitting tree corresponding to a family of signals with Hamming weight 2, T_2^n. For simplicity, we truncated terminal rightward paths from leaves to the bottom level of the tree. The corresponding support set of this tree is $S = \{0,1,2,3,4,5,6,8,9,10,12,16,17,18,20,24\}$.

Note that all leaves of T_c^n are at level $\log_2 n$.

It is not hard to see that T_c^n is in fact the splitting tree of the set H_c^n and, hence, the number of its leaves is $\sum_{i=0}^{c} \binom{\log_2 n}{i}$. An illustration of the tree T_c^n for $c = 2$ and $n = 32$ is shown in Figure 2.4.

We prove the following theorem in Section 2.6 (see Theorem 2.6.1):

Theorem 2.2.3 (Informal version of Theorem 2.6.1)**.** *A peeling process with threshold $\tau \leq \log_2 k + O(1)$ (i.e. any threshold that allows isolation of an element at cost bounded by $O(k)$) must take $k^{1-o(1)}$ iterations to terminate.*

To add to the result above, we note that the lower bound on the number of rounds of adaptivity is not the only cause for quadratic runtime in our algorithm. The other cause is the necessity to update the residual signal as more and more elements are recovered, i.e. perform non-uniform Fourier transform computations. Since no subquadratic approach to this problem are known in high dimensions, it seems plausible that a $k^{2-\Omega(1)}$ runtime algorithm for high-dimensional FFT would also shed light on the complexity of this intriguing problem.

Organization. In Section 2.3, we introduce basic definitions and notation that will be used throughout the chapter. Section 2.4 introduces our main technical tool of adaptive aliasing filter, which are used in the various algorithms found in this chapter. Section 2.5 shows how to use the adaptive aliasing filters to solve the problem of estimation for Fourier measurements for worst-case signals, i.e., problem (2.3), thereby proving Theorem 2.2.1. Section 2.6 then

shows that the inherent tree pruning process used to subtract off recovered frequencies and access residual signals in the estimation algorithm is essentially optimal.
Section 2.7 proves our main theorem, Theorem 2.1.1, for problem (2.1) on worst-case signals. Additionally, it shows how to improve on the runtime under the assumption that the signal is a worst-case signal with random phase, thereby proving Theorem 2.2.2.

2.3 Preliminaries and Notations

In this section, we introduce notation and basic definitions that we will use in this chapter. For any positive integer n, we use the notation $[n]$ to denote the set of integer numbers $\{0, 1, \ldots, n-1\}$. We are interested in computing the Fourier transform of discrete signals of size N in dimension d, where $N = n^d$ for some $n \geq 2$. Such a signal will be a function $[n]^d \to \mathbb{C}$. However, we will often identify $[n]^d \to \mathbb{C}$ with $\mathbb{C}^{n^d}$ for convenience (and often use the two interchangably depending on the context). This correspondence is formally defined later in Definition 2.4.2. We first need the notion of an inner product.

Definition 2.3.1 (Inner product)**.** Let $\boldsymbol{t}$ and $\boldsymbol{f}$ be two vectors in dimension d. We denote the inner product of $\boldsymbol{t}$ and $\boldsymbol{f}$ by $\boldsymbol{f}^T \boldsymbol{t} = \sum_{q=1}^{d} f_q t_q$.

Let us define the *Fourier transform* of a multidimensional signal.

Definition 2.3.2 (Multidimensional Fourier transform)**.** For any positive integers d and n, the *Fourier transform* of a signal $x \in \mathbb{C}^{n^d}$ is denoted by $\widehat{x}$, where for any $\boldsymbol{f} \in [n]^d$, we define $\widehat{x}_{\boldsymbol{f}} = \sum_{\boldsymbol{t} \in [n]^d} x_{\boldsymbol{t}} e^{-2\pi i \frac{\boldsymbol{f}^T \boldsymbol{t}}{n}}$.

Note that in the case of $n = 2$, the Fourier transform reduces to the Hadamard transform of size $N = 2^d$.

Claim 2.3.1 (Parseval's theorem)**.** *For any positive integers n and d, any signal $x \in \mathbb{C}^{n^d}$ satisfies $\|\widehat{x}\|_2^2 = n^d \cdot \|x\|_2^2$.*

Definition 2.3.3 (Unit impulse)**.** For any positive integers n and d, the *unit impulse function* $\delta \in \mathbb{C}^{n^d}$ is defined as the function given by $\delta(\boldsymbol{t}) = 1$ for $\boldsymbol{t} = 0$ and $\delta(\boldsymbol{t}) = 0$ for $\boldsymbol{t} \neq 0$.

Claim 2.3.2. *For any positive integers d, n, and any $\boldsymbol{a} \in [n]^d$, the inverse Fourier transform of $\widehat{x} : [n]^d \to \mathbb{C}$ given by $\widehat{x}_{\boldsymbol{f}} = e^{2\pi i \frac{\boldsymbol{a}^T \boldsymbol{f}}{n}}$ is $x_{\boldsymbol{t}} = \delta(\boldsymbol{t} + \boldsymbol{a})$.*

Claim 2.3.3 (Convolution theorem)**.** *Suppose d and n are positive integers. Then, for any signals $x, y \in \mathbb{C}^{n^d}$, $\widehat{(x * y)} = \widehat{x} \cdot \widehat{y}$, where $x * y$ is the convolution of x and y which itself is a signal in $\mathbb{C}^{n^d}$ defined as, $(x * y)_{\boldsymbol{t}} = \sum_{\boldsymbol{\tau} \in [n]^d} x_{\boldsymbol{\tau}} y_{\boldsymbol{t}-\boldsymbol{\tau}}$ for all $\boldsymbol{t} \in [n]^d$.*

We will require the notion of a *tensor product* of signals. Given d signals $G_1, G_2, \ldots, G_d : [n] \to \mathbb{C}$, the tensor product constructs a signal in $\mathbb{C}^{n^d}$ that is defined as follows.

Definition 2.3.4 (Tensor product)**.** Suppose d and n are positive integers. For any set of functions $G_1, G_2, \ldots, G_d : [n] \to \mathbb{C}$, we define the *tensor product* $(G_1 \times G_2 \times \cdots \times G_d) : [n]^d \to \mathbb{C}$ as $(G_1 \times G_2 \times \cdots \times G_d)(\boldsymbol{j}) = G_1(j_1) \cdot G_2(j_2) \cdots G_d(j_d)$ for all $\boldsymbol{j} = (j_1, j_2, \ldots, j_d) \in [n]^d$.

Note that the tensor product is essentially a generalization of the usual outer product on two vectors to d vectors.

Claim 2.3.4 (Fourier transform of a tensor product)**.** *For any integers n, d and $G_1, G_2, \ldots, G_d \in \mathbb{C}^n$, let $G : [n]^d \to \mathbb{C}$ denote the tensor product $G = G_1 \times G_2 \times \cdots \times G_d$. Then, the d-dimensional Fourier transform $\widehat{G}$ of G is the tensor product of $\widehat{G}_1, \widehat{G}_2, \cdots, \widehat{G}_d$, i.e., $\widehat{G} = \widehat{G}_1 \times \widehat{G}_2 \times \cdots \times \widehat{G}_d$.*

Definition 2.3.5. For any positive d, n, and k, a signal $x : [n]^d \to \mathbb{C}$ is called *Fourier k-sparse* if $\|\widehat{x}\|_0 = k$.

Definition 2.3.6 (Restricted Isometry Property)**.** We say that a matrix $A \in \mathbb{C}^{q \times n}$ satisfies the *restricted isometry property (RIP)* of *order k* if for every k-sparse vector $x \in \mathbb{C}^n$, i.e., $\|x\|_0 \leq k$, it holds that $\frac{1}{2}\|x\|_2^2 \leq \|Ax\|_2^2 \leq \frac{3}{2}\|x\|_2^2$.

We will use the following theorem from Haviv and Regev (2017).

Theorem 2.3.1. (Restricted Isometry Property (Haviv and Regev, 2017, Theorem 3.7)) *For sufficiently large N and k, a unitary matrix $M \in \mathbb{C}^{N \times N}$ satisfying $\|M\|_\infty = O\left(\frac{1}{\sqrt{N}}\right)$, and some $q = O\left(k \log^2 k \log N\right)$, if $A \in \mathbb{C}^{q \times N}$ is a matrix whose q rows are chosen uniformly and independently from the rows of M, multiplied by $\sqrt{\frac{N}{q}}$, then, with probability $1 - \frac{1}{N^{10}}$, the matrix A satisfies the restricted isometry property of order k, as per Definition 2.3.6.*

2.4 Adaptive Aliasing Filters

In this section, we introduce a new class of filters that forms the basis of our algorithm for estimation of worst-case Fourier sparse signals. For simplicity, we begin by introducing the filters in one-dimensional setting and then show how they naturally extend to the multidimensional setting (via tensoring). Throughout the section, we assume that the input is a signal $x \in \mathbb{C}^n$ with supp $\widehat{x} = S$ for some $S \subseteq [n]$.

2.4.1 One-dimensional Fourier transform

We restate the following definition for T_n^{full} and the corresponding labels of its vertices:

Definition 2.2.2. Suppose n is a power of two integer. Let T_n^{full} be a full binary tree of height $\log_2 n$, where for every $j \in \{0, 1, \ldots, \log_2 n\}$, the nodes at level j (i.e., at distance j from the root) are labeled with integers in $\mathbb{Z}_{2^j}$. For a node $v \in T_n^{\text{full}}$, we let f_v be its label. The label of the root is $f_{root} = 0$. The labelling of T_n^{full} satisfies the condition that for every $j \in [\log_2 n]$ and every v at level j, the right and left children of v have labels f_v and $f_v + 2^j$, respectively. Note that the root of T_n^{full} is at level 0, while the leaves are at level $\log_2 n$.

Algorithm 7 Splitting tree construction in time $O(|S|\log n)$

procedure TREE(S, n)
  $\mathscr{C}_0 \leftarrow \{(r, S)\}$
  Let T be a tree with one node, labeled $f_r = 0$
  for $j = 1$ to $\log_2 n$ **do**
    $\mathscr{C}_j \leftarrow \emptyset$
    for all $(v, S_v) \in \mathscr{C}_{j-1}$ **do** ▷ $\mathscr{C}_{j-1}$: set of every node at level $j-1$ and set of all frequencies in the subtree of each node
      $R \leftarrow \{g \in S_v : g = f_v \pmod{2^j}\}$
      $L \leftarrow \{g \in S_v : g = f_v + 2^{j-1} \pmod{2^j}\}$
      if $R \neq \emptyset$ **then**
        Add a right child, u, to node v of T with label $f_u \leftarrow f_v$
        $\mathscr{C}_j \leftarrow \mathscr{C}_j \cup \{(u, R)\}$
      if $L \neq \emptyset$ **then**
        Add a left child, w, to node u of T with label $f_w \leftarrow f_v + 2^{j-1}$
        $\mathscr{C}_j \leftarrow \mathscr{C}_j \cup \{(w, L)\}$
  return T
procedure TREE.REMOVE(T, v)
  $r \leftarrow$ root of T, $l \leftarrow l_T(v)$
  $v_0, v_1, \ldots, v_l \leftarrow$ path from r to v in T, where $v_0 = r$ and $v_l = v$
  $q \leftarrow$ largest $j \in \{0, 1, \cdots l\}$ such that v_j has two children
  Remove $v_{q+1}, ..., v_l$ and their connecting edges from T
  return T

Next, we recall the definition of the *splitting tree* of a set.

Definition 2.2.3 (Splitting tree)**.** Let n be an integer power of two. For every $S \subseteq [n]$, the *splitting tree* $T = \text{Tree}(S, n)$ of a set S is a binary tree that is the subtree of T_n^{full} that contains, for every $j \in [\log_2 n]$, all nodes $v \in T_n^{\text{full}}$ at level j such that $\{f \in S : f \equiv f_v \pmod{2^j}\} \neq \emptyset$.

For every node $v \in T$, the *level* of v, denoted by $l_T(v)$, is the distance from v to the root. The splitting tree $T = \text{Tree}(S, n)$ can be constructed easily in $O(|S|\log n)$ time, given S. We provide a simple pseudocode for this in Algorithm 7. The following basic claim will be useful and follows immediately from the definition of $T = \text{Tree}(S, n)$.

Claim 2.4.1. *For every integer power of two n, if T is a subtree of T_n^{full}, then for every node $v \in T$, the labels of nodes that belong to the subtree T_v of T rooted at v are congruent to f_v modulo $2^{l_T(v)}$. Furthermore, every node $u \in T$ at level $l_T(v)$ or higher which satisfies $f_u \equiv f_v \pmod{2^{l_T(v)}}$ belongs to T_v.*

Now let us recall the definition of the weight of a leaf. We need this definition to define our notion of isolating filters.

Definition 2.2.6 (Weight of a leaf)**.** Suppose n is a power of two. Let T be a subtree of T_n^{full}.

Then for any leaf $v \in T$, we define its *weight* $w_T(v)$ *with respect to* T to be the number of ancestors of v in tree T with two children.

Definition 2.4.1 ((f, S)-isolating filter)**.** For every power of two n, set $S \subseteq [n]$, and $f \in S$, a filter $G \in \mathbb{C}^n$ is called (f, S)*-isolating* if $\widehat{G}_f = 1$, and $\widehat{G}_{f'} = 0$ for all $f' \in S \setminus \{f\}$.

In particular, if G is (f, S)-isolating, then for every signal $x \in \mathbb{C}^n$ with supp $\widehat{x} \subseteq S$, we have

$$\begin{aligned}\sum_{j\in[n]} x_j G_{t-j} &= (x * G)_t \\ &= \frac{1}{n} \sum_{f\in[n]} \widehat{x}_f \cdot \widehat{G}_f \cdot e^{2\pi i \frac{ft}{n}} \\ &= \frac{1}{n} \widehat{x}_f e^{2\pi i \frac{ft}{n}}\end{aligned}$$

for all $t \in [n]$, by convolution theorem, see Claim 2.3.3.

While the definitions above suffice to state our estimation primitive, our Sparse FFT algorithm requires a filter G that satisfies a more refined property due to the fact that throughout the execution of the algorithm, the identity of supp $\widehat{x}$ is only partially known. We encode this knowledge as a subtree T of T_n^{full} whose leaves are not necessarily at level $\log_2 n$. Hence, every leaf $v \in T$ corresponds to a set of frequencies in the support of $\widehat{x}$ whose full identities have not been discovered yet. This is captured by the following definition:

Definition 2.2.4 (Frequency cone of a leaf of T)**.** For every power of two n, subtree T of T_n^{full}, and vertex $v \in T$ which is at level $l_T(v)$ from the root, the *frequency cone of* v *with respect to* T is defined as,

$$\text{FrequencyCone}_T(v) := \left\{ f \in [n] : f \equiv f_v \pmod{2^{l_T(v)}} \right\}.$$

Note that under this definition, the frequency cone of a vertex v of T corresponds to the subtree rooted at v when T is embedded inside T_n^{full} (see Figure 2.2).

Definition 2.2.5 ((v, T)-isolating filter)**.** For every integer n, subtree T of T_n^{full}, and leaf v of T, a filter $G \in \mathbb{C}^n$ is called (v, T)*-isolating* if the following conditions hold:

- For all $f \in \text{FrequencyCone}_T(v)$, we have $\widehat{G}_f = 1$.
- For every $f' \in \bigcup_{\substack{u:\text{ leaf of } T \\ u \neq v}} \text{FrequencyCone}_T(u)$, we have $\widehat{G}_{f'} = 0$.

Note that in particular, for all signals $x \in \mathbb{C}^n$ with supp $\widehat{x} \subseteq \bigcup_{u:\text{ leaf of } T} \text{FrequencyCone}_T(u)$ and $t \in [n]$,

$$\sum_{j\in[n]} x_j G_{t-j} = \frac{1}{n} \sum_{f\in\text{FrequencyCone}_T(v)} \widehat{x}_f e^{2\pi i \frac{ft}{n}}.$$

Now we are in a position to present an efficient algorithm to compute a (v, T)-isolating filter with compact support in the time domain and formally prove its theoretical guarantees.

Algorithm 8 Filter construction

procedure FILTERPREPROCESS(T, v, n)
  $r \leftarrow$ root of T, $l \leftarrow l_T(v)$, $f \leftarrow f_v$
  $v_0, v_1, \ldots, v_l \leftarrow$ path from r to v in T, where $v_0 = r$ and $v_l = v$
  $\mathbf{g} \leftarrow \{0\}^{\log_2 n}$
  for $j = 1$ to l **do**
    if v_{j-1} has two children in T **then**
      $g_j \leftarrow e^{-2\pi i \frac{f}{2^j}}$
  return g
procedure FILTERTIME($\mathbf{g}, n$)
  $G(t) \leftarrow \delta(t)$ for all $t \in [n]$
  for $l = 1$ to $\log_2 n$ **do**
    if $g_l \neq 0$ **then**
      $G(t) \leftarrow \frac{G(t)}{2} + g_l \cdot \frac{G\left(t+n/2^l\right)}{2}$ for all $t \in [n]$
  return G
procedure FILTERFREQUENCY($\mathbf{g}, n, \xi$)
  $\widehat{G}_\xi \leftarrow 1$
  for $l = 1$ to $\log_2 n$ **do**
    if $g_l \neq 0$ **then**
      $\widehat{G}_\xi \leftarrow \widehat{G}_\xi \cdot \left(1 + g_l \cdot e^{2\pi i \frac{\xi}{2^l}}\right) \Big/ 2$
  return $\widehat{G}_\xi$

Lemma 2.4.1 (Filter properties)**.** *For every integer power of two n, subtree T of T_n^{full}, and leaf $v \in T$, the procedure* FILTERPREPROCESS(T, v, n) *outputs a static data structure $\mathbf{g} \in \mathbb{C}^{\log_2 n}$ in time $O(\log n)$ such that, given $\mathbf{g}$, the following conditions hold:*

1. *The primitive* FILTERTIME$(\mathbf{g}, n)$ *outputs a filter G such that $|\text{supp } G| = 2^{w_T(v)}$ and G is a (v, T)-isolating filter. Moreover, the procedure runs in time $O\left(2^{w_T(v)} + \log n\right)$.*
2. *For every $\xi \in [n]$, the primitive* FILTERFREQUENCY$(\mathbf{g}, n, \xi)$ *computes the Fourier transform of G at frequency ξ, namely, $\widehat{G}(\xi)$, in time $O(\log n)$.*

Before proving Lemma 2.4.1, we establish a corollary, assuming that Lemma 2.4.1 holds.

Corollary 2.4.1. *Suppose n is a power of two, $S \subseteq [n]$, and $f \in S$. Then, let $T = \text{Tree}(S, n)$ be the splitting tree of S. If v is the leaf of T with label $f_v = f$, while $\mathbf{g}$ is the output of* FILTERPREPROCESS(T, v, n)*, and G is the filter computed by* FILTERTIME$(\mathbf{g}, n)$*, then the following conditions hold:*

(1) *G is an (f, S)-isolating filter.*

(2) $|\text{supp } G| = 2^{w_T(v)}$.

Proof. Indeed, given a subset S, if $T = \text{Tree}(S, n)$, then all the leaves of T are at level $\log_2 n$ and the set of labels of the leaves is exactly equal to S. Hence, for every leaf v of T, one has $\text{FrequencyCone}_T(v) = \{f_v\}$. By Lemma 2.4.1, G is a (v,T)-isolating filter. Therefore, by Definition 2.2.5,

$$\emptyset = \text{supp}\,\widehat{G} \cap \left(\bigcup_{\substack{u:\text{ leaf of } T \\ u \neq v}} \text{FrequencyCone}_T(u) \right) = \text{supp}\,\widehat{G} \cap \left(\bigcup_{\substack{u:\text{ leaf of } T \\ u \neq v}} \{f_u\} \right) = \text{supp}\,\widehat{G} \cap \left(S \setminus f_v\right),$$

and $\widehat{G}(f) = 1$ for all $f \in \text{FrequencyCone}_T(v) = \{f_v\}$. This implies **(1)**, see definition of (f_v, S)-isolating filters in 2.4.1. Property **(2)** follows directly from Lemma 2.4.1. □

Now, we prove Lemma 2.4.1.

Proof of Lemma 2.4.1: Let v be a leaf of T, $l = l_T(v)$ denote the level of v (i.e., distance from the root), r denote the root of T, and $v_0, v_1, \ldots, v_l$ denote the path from root to v in T, where $v_0 = r$ and $v_l = v$.

We first show how to efficiently construct a (v,T)-isolating filter in the *Fourier* domain, i.e., how to efficiently construct $\widehat{G}$. Then we derive the time domain representation of G. We iteratively define a sequence of functions $G_0, G_1, \ldots, G_l$ (with Fourier transforms $\widehat{G}_0, \widehat{G}_1, \ldots, \widehat{G}_l$, respectively) by traversing the path from the root to v in T, after which we let G be the final filter constructed on this path, i.e., $G := G_l$ (and $\widehat{G} := \widehat{G}_l$). We start with $\widehat{G}_0(\xi) = 1$ for all $\xi \in [n]$. Then, we iteratively define $\widehat{G}_q$ in terms of $\widehat{G}_{q-1}$ according to the following update rule for all $q = 1, 2, \ldots, l$:

$$\widehat{G}_q(\xi) = \begin{cases} \widehat{G}_{q-1}(\xi) \cdot \frac{1+e^{2\pi i \frac{\xi - f_v}{2^q}}}{2} & \text{if } v_{q-1} \text{ has two children in } T \\ \widehat{G}_{q-1}(\xi) & \text{otherwise} \end{cases}. \tag{2.4}$$

for every $\xi \in [n]$.
In order to prove that $G = G_l$ is a (v,T)-isolating filter, it is enough to show that G satisfies,

$$\text{supp}\,\widehat{G} \cap \left(\bigcup_{\substack{u:\text{ leaf of } T \\ u \neq v}} \text{FrequencyCone}_T(u) \right) = \emptyset, \tag{2.5}$$

and,

$$\widehat{G}(f) = 1 \text{ for all } f \in \text{FrequencyCone}_T(v). \tag{2.6}$$

We now prove (2.5). Consider a leaf u of T distinct from v. Recall that $v_0, v_1, \ldots, v_l$ denotes the root to v path in T. Let j be the largest integer such that v_j is a common ancestor of v and u. By definition of tree T (Definition 2.2.2) and noting that v_j is at level j, one has that the label of the right child a of v_j is f_{v_j}, and the label of the left child b is $f_{v_j} + 2^j$. Furthermore, using

this together with Claim 2.4.1, we get that the labels of all nodes in subtree T_a of T subtended at the right child a of v are congruent to $f_a = f_{v_j}$ modulo 2^{j+1}, and labels in the subtree T_b rooted at the left child b of v_j are all congruent to $f_b = f_{v_j} + 2^j$ modulo 2^{j+1}.

Suppose that v belongs to the right subtree of v_j, and u belongs to the left subtree (the other case is symmetric). We thus get that $f_v \equiv f_{v_j} \pmod{2^{j+1}}$, and $f_u \equiv f_{v_j} + 2^j \pmod{2^{j+1}}$. It now suffices to note that by construction of $\widehat{G}$ (see (2.4)), we have that for all $\xi \in [n]$,

$$\widehat{G}_{j+1}(\xi) = \widehat{G}_j(\xi) \cdot \frac{1 + e^{2\pi i \frac{\xi - f_v}{2^{j+1}}}}{2}.$$

By Claim 2.4.1, for all $f \in \text{FrequencyCone}_T(u)$ one has that $f \equiv f_u \pmod{2^{l_T(u)}}$ and hence, $f \equiv f_u \pmod{2^{j+1}}$ because $j+1 \le l_T(u)$. Therefore, by substituting $\xi = f$ in the above, we get

$$\widehat{G}_{j+1}(f) = \widehat{G}_j(f) \cdot \frac{1 + e^{2\pi i \frac{f - f_v}{2^{j+1}}}}{2} = \widehat{G}_j(f) \cdot \frac{1 + e^{2\pi i \frac{f_u - f_v}{2^{j+1}}}}{2} = 0,$$

implying that $\widehat{G}_{j+1}(f) = 0$ and, hence, $\widehat{G}_l(f) = 0$, as required.

It remains to prove (2.6). Consider any $f' \in \text{FrequencyCone}_T(v)$, and note that by Claim 2.4.1, $f' \equiv f_v \pmod{2^l}$. Using this in (2.4), we get

$$\widehat{G}(f') = \prod_{\substack{q \in \{1,2,\ldots,l\} \\ v_{q-1} \text{ has two children in } T}} \frac{1 + e^{2\pi i \frac{f' - f_v}{2^q}}}{2} = 1,$$

since $f' - f_v \equiv 0 \pmod{2^q}$ for every $q = 0, \ldots, l$.

Next, note that the primitive FILTERPREPROCESS(T, v, n) preprocesses the tree T by traversing the path from root to leaf v in time $O(\log_2 n)$. Given $\mathbf{g}$, the primitive FILTERFREQUENCY $(\mathbf{g}, n, \xi)$ implements (2.4) for successive values of q, and the runtime of this algorithm is $O(\log_2 n)$ because of the *for* loop passing through vector $\mathbf{g}$.

Finally, it remains to show that the filter G in *time domain* can be computed efficiently and has a small support. First note that by Claim 2.3.2, the inverse Fourier transform of $\frac{1+e^{2\pi i \frac{\xi - f_v}{2^q}}}{2}$ is $\frac{\delta(t) + e^{-2\pi i \frac{f_v}{2^q}} \delta\left(t + \frac{n}{2^q}\right)}{2}$.

By duality of convolution in the time domain and multiplication in the Fourier domain (see Claim 2.3.3), we can equivalently define G (see (2.4)) by letting $G_0(t) = \delta(t)$ and setting,

$$G_q(\xi) = \begin{cases} G_{q-1}(t) * \frac{\delta(t) + e^{-2\pi i \frac{f_v}{2^q}} \delta\left(t + \frac{n}{2^q}\right)}{2} & \text{if } v_{q-1} \text{ has two children in } T \\ G_{q-1}(t) & \text{otherwise} \end{cases} \tag{2.7}$$

for every $q = 1, \ldots, l$. Thus, $G = G_l$ is the time domain representation of the filter $\widehat{G}$ defined

in (2.4). We now note that convolving any function with a function supported on two points, e.g., $\frac{1}{2}\left(\delta(t) + e^{-2\pi i f_v/2^q}\delta(t+\frac{n}{2^q})\right)$, at most doubles the support. Since the number of times the convolution is performed in obtaining G_l from G_0 (as per (2.7)) is $w_T(v)$, the support size of G is at most $2^{w_T(v)}$. Given $\mathbf{g}$, the primitive FILTERTIME $(\mathbf{g}, n)$ implements the above algorithm for construction of G and, therefore, runs in time $O(2^{w_T(v)} + \log_2 n)$. □

2.4.2 d-dimensional Fourier transform

In this section, we show that our construction of adaptive aliasing filters from the previous section naturally extends to higher dimensions without any loss (by tensoring).

Definition 2.4.2 (Flattening of $[n]^d$ to $[n^d]$. Unflattening of $[n^d]$ to $[n]^d$)**.** For every integer power of two n, positive integer d, and $\boldsymbol{f} = (f_1, \ldots, f_d) \in [n]^d$ we define the *flattening* of $\boldsymbol{f}$ as,

$$\widetilde{\boldsymbol{f}} = \sum_{r=1}^{d} f_r \cdot n^{r-1}.$$

Similarly, for a subset $S \subseteq [n]^d$ we let $\widetilde{S} := \{\widetilde{\boldsymbol{f}} : \boldsymbol{f} \in S\}$ denote the flattening of S.

For $\widetilde{\boldsymbol{\xi}} \in [n^d]$, the *unflattening* of $\widetilde{\boldsymbol{\xi}}$ is uniquely defined as $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_d) \in [n]^d$, where

$$\xi_q = \frac{\widetilde{\boldsymbol{\xi}} - \left(\widetilde{\boldsymbol{\xi}} \mod n^{q-1}\right)}{n^{q-1}} \pmod{n}.$$

for every $q = 1, \ldots, d$. Similarly, for a subset $\widetilde{R} \subseteq [n^d]$, we let $R := \{\boldsymbol{\xi} \in [n]^d : \widetilde{\boldsymbol{\xi}} \in \widetilde{R}\}$ denote the unflattening of $\widetilde{R}$.

Definition 2.4.3 (Multidimensional splitting tree)**.** Suppose d is a positive integer and n is a power of two. For every $S \subseteq [n]^d$, the *flattened splitting tree* of S is defined as $\widetilde{T} = \text{Tree}(\widetilde{S}, n^d)$ where $\widetilde{S}$ is flattening of S.

The unflattened splitting tree of S is denoted by T and is obtained from the flattened splitting tree $\widetilde{T}$ by unflattening the labels $\widetilde{\boldsymbol{f}}_v$ of all nodes $v \in \widetilde{T}$.

Definition 2.4.4 (Multidimensional $(\boldsymbol{f}, S)$-isolating filter)**.** Suppose n is a power of two integer and $S \subseteq [n]^d$ for a positive integer d. Then, for any frequency $\boldsymbol{f} \in S$, a filter $G : [n]^d \to \mathbb{C}$ is called $(\boldsymbol{f}, S)$-*isolating* if $\widehat{G}_{\boldsymbol{f}} = 1$ and $\widehat{G}_{\boldsymbol{f}'} = 0$ for every $\boldsymbol{f}' \in S \setminus \{\boldsymbol{f}\}$.

Definition 2.4.5 (Frequency cone of a leaf of T in high dimensions)**.** Suppose d is a positive integer, n is a power of two, and $N = n^d$. For every unflattened subtree T of T_N^{full} and $v \in T$, we define the *frequency cone of* v as,

$$\text{FrequencyCone}_T(v) := \left\{\boldsymbol{f} \in [n]^d : \widetilde{\boldsymbol{f}} \equiv \widetilde{\boldsymbol{f}}_v \pmod{2^{l_T(v)}}\right\},$$

where $l_T(v)$ denotes the level of v in T (i.e., the distance from the root).

We use the following straightforward claim about the properties of frequency cones.

Claim 2.4.2. *For every positive integer d, power of two integer n, and every subtree T of $T^{\text{full}}_{n^d}$ and every leaf $v \in T$ of height $l_T(v) < d\log_2 n$, if $T' = T \cup \{\text{left child } u \text{ of } v\} \cup \{\text{right child } w \text{ of } v\}$, then the following holds,*

$$\text{FrequencyCone}_T(v) = \text{FrequencyCone}_{T'}(u) \cup \text{FrequencyCone}_{T'}(w)$$

Now we are ready to extend the definition of our isolating filters to high dimensions.

Definition 2.4.6 (Multidimensional (v,T)-isolating filter)**.** Suppose d is a positive integer, n is a power of two, and $N = n^d$. For every subtree T of T^{full}_N and vertex $v \in T$, a filter $G \in \mathbb{C}^{n^d}$ is called (v,T)-*isolating* if $\widehat{G}_{\boldsymbol{f}} = 1$ for every $\boldsymbol{f} \in \text{FrequencyCone}_T(v)$ and $\widehat{G}_{\boldsymbol{f}'} = 0$ for every $\boldsymbol{f}' \in \bigcup_{\substack{u:\text{ leaf of } T \\ u \neq v}} \text{FrequencyCone}_T(u)$.

In particular, for every signal $x \in \mathbb{C}^{n^d}$ with $\operatorname{supp} \widehat{x} \subseteq \bigcup_{u:\text{ leaf of } T} \text{FrequencyCone}_T(u)$ and for all $\boldsymbol{t} \in [n]^d$,

$$\sum_{\boldsymbol{j} \in [n]^d} x_{\boldsymbol{j}} G_{\boldsymbol{t}-\boldsymbol{j}} = \frac{1}{N} \sum_{\boldsymbol{f} \in \text{FrequencyCone}_T(v)} \widehat{x}_{\boldsymbol{f}} e^{2\pi i \frac{\boldsymbol{f}^T \boldsymbol{t}}{n}}.$$

Lemma 2.4.2 (Construction of a multidimensional isolating filter)**.** *Suppose n is a power of two integer and d is a positive integer. Let $N = n^d$. For every subtree T of T^{full}_N and every leaf $v \in T$, there exists a (v,T)-isolating filter G such that $|\operatorname{supp} G| = 2^{w_T(v)}$. Such a filter G can be constructed in time $O\left(2^{w_T(v)} + \log N\right)$. Moreover, for any frequency $\boldsymbol{\xi} \in [n]^d$, the Fourier transform of G at frequency $\boldsymbol{\xi}$, i.e., $\widehat{G}(\boldsymbol{\xi})$, can be computed in time $O(\log N)$.*

Proof. The key idea is to choose q^* to be the smallest positive integer such that $l_T(v) \leq q^* \cdot \log_2 n$. One then defines successive filters $G^{(0)}, G^{(1)}, \ldots, G^{(q^*)}$ by letting $\widehat{G}^{(0)} = 1$ and

$$\widehat{G}^{(q)}(\boldsymbol{f}) = \widehat{G}^{(q-1)}(\boldsymbol{f}) \cdot \widehat{G}_q(f_q)$$

for $q = 1, 2, \ldots, q^*$, where $\widehat{G}_q$ is an isolating filter corresponding to the projection of the leaves of tree T into coordinate q. The final filter $G = G^{(q^*)}$ turns out to be (v,T)-isolating.

Let v be a leaf of T, let $l = l_T(v)$ denote the level of v, let r denote the root of T, and let $v_0, v_1, \ldots, v_l$ denote the path from root to v in T, where $v_0 = r$ and $v_l = v$. Let q^* denote the smallest positive integer such that $l \leq q^* \cdot \log_2 n$. Note that $q^* \leq d$.

For every $q \in \{0, 1, \ldots, d\}$ let $T^{(q)}$ be a subtree of T which denotes the result of truncating the path $v_0, v_1, \ldots, v_l$ of T to contain only the nodes that are at distance at most $q\log_2 n$ from the root, i.e., removing the subtree rooted at $v_{q\log_2 n+1}$. We construct the (v,T)-isolating filter $\widehat{G}$ iteratively by starting with $\widehat{G}^{(0)} = 1$ and refining $\widehat{G}^{(q-1)}$ to $\widehat{G}^{(q)}$ over q^* steps. The filters $\widehat{G}^{(q)}$ will be $\left(v_{q\cdot\log_2 n}, T^{(q)}\right)$-isolating for $q = 0, 1, \ldots, q^* - 1$ and $\widehat{G}^{(q^*)}$ will be $\left(v_l, T^{(q^*)}\right)$-isolating. Since $T^{(q^*)} = T$ and $v_l = v$, the filter $\widehat{G}^{(q^*)}$ will be (v,T)-isolating, as required.

For every $q \in \{1, ..., q^*\}$ let T^v_q be the subtree of T which is rooted at $v_{(q-1)\cdot\log_2 n}$ and is restricted to contain only the nodes that are at distance at most $\log_2 n$ from $v_{(q-1)\cdot\log_2 n}$. For every node

$u \in T_q^v$ the label of u is defined to be $f_u = (\boldsymbol{f}_u)_q$, i.e., the qth coordinate of $\boldsymbol{f}_u$, where $\boldsymbol{f}_u$ is the label of node u in tree T.

Iteratively define $\widehat{G}^{(q)}$ for $q = 1, \dots q^*$, assuming $\widehat{G}^{(0)} := 1$, as follows for $\boldsymbol{f} = (f_1, \dots f_d) \in [n]^d$,

$$\widehat{G}^{(q)}(\boldsymbol{f}) = \widehat{G}^{(q-1)}(\boldsymbol{f}) \cdot \widehat{G}_q(f_q). \tag{2.8}$$

where $\widehat{G}_q$ is a $\left(v_{q\cdot\log_2 n}, T_q^v\right)$-isolating filter for all $q = 1, ..., q^* - 1$ and $\widehat{G}_{q^*}$ is a $\left(v_l, T_{q^*}^v\right)$-isolating filter. By lemma 2.4.1, for every $q = 1, \dots, q^*$ there exists such G_q with $|\text{supp } G_q| = 2^{w_{T_q^v}(v_{q\cdot\log_2 n})}$ and can be constructed in time $O\left(2^{w_{T_q^v}(v_{q\cdot\log_2 n})} + \log n\right)$. Such a filter can be computed in the Fourier domain at any desired frequency in time $O(\log n)$. Note that $\widehat{G}^{(q)}$ is a tensor product of q one-dimensional filters. We now show by induction on q that $\widehat{G}^{(q)}$ is a $\left(v_{q\cdot\log_2 n}, T^{(q)}\right)$-isolating filter.

The **base** of the induction is provided by $q = 0$: since v_0 is the root of $T^{(0)}$, we have that $\text{FrequencyCone}_{T^{(0)}}(v_0) = [n]^d$ and $\widehat{G}^{(0)} \equiv 1$ as required.

Inductive step ($q-1 \to q$): We first show $\widehat{G}_{\boldsymbol{f}'}^{(q)} = 0$ for every $\boldsymbol{f}' \in \bigcup_{\substack{u:\text{ leaf of } T^{(q)} \\ u \neq v_{q\cdot\log_2 n}}} \text{FrequencyCone}_{T^{(q)}}(u)$. Let u be a leaf of $T^{(q)}$ distinct from $v_{q\cdot\log_2 n}$. Let u' denote the leaf of $T^{(q-1)}$ that is is the ancestor of u (note that u' could be the same node as u too). Consider the two possible cases.

Case 1: $\boldsymbol{f}' \notin \text{FrequencyCone}_{T^{(q-1)}}\left(v_{(q-1)\log_2 n}\right)$**.** Suppose that $u' \neq v_{(q-1)\cdot\log_2 n}$. Note that $l_T(u') \leq (q-1)\log_2 n$, and additionally, $\text{FrequencyCone}_{T^{(q)}}(u) \subseteq \text{FrequencyCone}_{T^{(q-1)}}(u')$. Thus for every $\boldsymbol{f}' \in \text{FrequencyCone}_{T^{(q)}}(u)$ it is true that $\boldsymbol{f}' \in \text{FrequencyCone}_{T^{q-1}}(u')$. By the inductive hypothesis we have that $\widehat{G}^{(q-1)}$ is $(v_{(q-1)\log_2 n}, T^{(q-1)})$-isolating, and hence by the assumption of $u' \neq v_{(q-1)\cdot\log_2 n}$, one has $\widehat{G}^{(q-1)}(\boldsymbol{f}') = 0$ for every such $\boldsymbol{f}'$, and thus $\widehat{G}^{(q)}(\boldsymbol{f}') = \widehat{G}^{(q-1)}(\boldsymbol{f}') \cdot \widehat{G}_q(f'_q) = 0$ as required.

Case 2: $\boldsymbol{f}' \in \text{FrequencyCone}_{T^{(q-1)}}\left(v_{(q-1)\log_2 n}\right)$**.** Suppose that $v_{(q-1)\cdot\log_2 n}$ is ancestor of u. Therefore, by definition of T_q^v, one can see that u is a leaf in T_q^v. Hence, by definition of T_q^v, for every $\boldsymbol{f}' \in \text{FrequencyCone}_{T^{(q)}}(u)$, it is true that $f'_q \in \text{FrequencyCone}_{T_q^v}(u)$. Recall that $\widehat{G}_q$ is a $(v_{q\cdot\log_2 n}, T_q^v)$-isolating filter and therefore, $\widehat{G}_q(f'_q) = 0$, and thus $\widehat{G}^{(q)}(\boldsymbol{f}') = \widehat{G}^{(q-1)}(\boldsymbol{f}') \cdot \widehat{G}_q(f'_q) = 0$ as required.

Now we show that $\widehat{G}_{\boldsymbol{f}}^{(q)} = 1$ for all $\boldsymbol{f} \in \text{FrequencyCone}_{T^{(q)}}\left(v_{q\cdot\log_2 n}\right)$. Note that $v_{q\cdot\log_2 n}$ is a leaf in T_q^v. Hence, for every $\boldsymbol{f} \in \text{FrequencyCone}_{T^{(q)}}\left(v_{q\cdot\log_2 n}\right)$, $f_q \in \text{FrequencyCone}_{T_q^v}\left(v_{q\cdot\log_2 n}\right)$. Since $\widehat{G}_q$ is a $\left(v_{q\cdot\log_2 n}, T_q^v\right)$-isolating filter, $\widehat{G}_q(f_q) = 1$. Because $\text{FrequencyCone}_{T^{(q)}}\left(v_{q\cdot\log_2 n}\right) \subseteq \text{FrequencyCone}_{T^{(q-1)}}\left(v_{(q-1)\cdot\log_2 n}\right)$, for every $\boldsymbol{f} \in \text{FrequencyCone}_{T^{(q)}}\left(v_{q\cdot\log_2 n}\right)$ it is true that $\boldsymbol{f} \in \text{FrequencyCone}_{T^{(q-1)}}\left(v_{(q-1)\cdot\log_2 n}\right)$. By the inductive hypothesis we have that $\widehat{G}^{(q-1)}$ is $\left(v_{(q-1)\log_2 n}, T^{(q-1)}\right)$-isolating, and hence $\widehat{G}^{(q-1)}(\boldsymbol{f}) = 1$, and thus $\widehat{G}^{(q)}(\boldsymbol{f}) = \widehat{G}^{(q-1)}(\boldsymbol{f}) \cdot \widehat{G}_q(f_q) = 1$ as required.

It remains to note that $w_T(v) = \sum_{q=1}^{q^*} w_{T_q^v}(v_{q\cdot\log_2 n})$. By Lemma 2.4.1, for every $q \in \{1,...,q^*\}$ one has $|\text{supp } G_q| = 2^{w_{T_q^v}(v_{q\cdot\log_2 n})}$, so $|\text{supp } G| = 2^{w_T(v)}$, as required (note that the support size of the tensor product of two filters is equal to the product of support sizes of each filter).

The total runtime for constructing this filter has two parts; First part is the computation time of G_q's for all $q \in \{1,...,q^*\}$ which takes $\sum_{q=1}^{q^*} O\left(2^{w_{T_q^v}(v_{q\cdot\log_2 n})} + \log n\right) = O\left(2^{w_T(v)} + d\log_2 n\right)$ by Lemma 2.4.1. Second part is the time needed for computing the tensor product of all G_q's which is $O\left(\|G_1\|_0 \cdot ... \cdot \|G_{q^*}\|_0\right) = O(2^{w_T(v)})$. Therefore the total runtime is $O\left(2^{w_T(v)} + d\log n\right)$. Moreover, the total time for computing $\widehat{G}(\boldsymbol{\xi})$ is the sum of the times needed for computing all $\widehat{G}_q(\xi_q)$'s for $q = 1,\cdots,q^*$, which is $O(d\log n) = O(\log N)$ by Lemma 2.4.1. □

2.4.3 Putting it together

Now we are ready to prove that for any splitting tree there exists a leaf that can be isolated via our filters using small runtime.

Claim 2.4.3. *For any binary tree T let L be the set of leaves of T. There exists a leaf $v \in L$ such that $w_T(v) \le \log_2 |L|$.*

Proof. Let T' be the tree obtained by "collapsing" T, i.e., removing all nodes (and incident edges) of T that have exactly one child. Then, observe that the leaves of T are still preserved in T', except that they are at possibly varying levels. In particular, a leaf v in T' will be at level $w_T(v)$. Thus, by applying Kraft's inequality to T' (which is an equality because every node in T' is either a leaf or has two children), we see that,

$$\sum_{v\in L} 2^{-w_T(v)} = 1.$$

Therefore, there exists a $v \in L$ such that $2^{-w_T(v)} \ge \frac{1}{|L|}$, implying $w_T(v) \le \log_2 |L|$, as desired. □

This gives us the main result of this section, and the main technical lemma of the paper:

Corollary 2.4.2. *For every integer $n \ge 1$ a power of two and every positive integer d, every $S \subseteq [n]^d$, there exists an $\boldsymbol{f} \in S$ and an $(\boldsymbol{f}, S)$-isolating filter G (as defined in Definition 2.4.4) such that $|\text{supp } G| \le |S|$.*

Proof. Follows by combining Lemma 2.4.2 with Claim 2.4.3. □

2.5 Estimation of Sparse High-dimensional Signals in Quadratic Time

In this section, we use the filters that we have constructed in Section 2.4 in order to show the first result of the paper, a deterministic algorithm for estimation of Fourier-sparse signals in time which is quadratic in the sparsity.

Algorithm 9 d-dimensional Estimation for Sparse FFT with sample and time complexity k^2

procedure ESTIMATE(x, S, n, d)
  $\widetilde{T} \leftarrow$ TREE($\widetilde{S}, n^d$) ▷ $\widetilde{S}$: flattening of S ▷ $\widetilde{T}$: flattened splitting tree of S
  $T \leftarrow$ the unflattening of $\widetilde{T}$
  $\widehat{\chi} \leftarrow \{0\}^{n^d}$
  while $T \neq \emptyset$ **do**
    $v \leftarrow \operatorname{argmin}_{u:\text{ leaf of } T} w_T(u)$, $\boldsymbol{f} \leftarrow \boldsymbol{f}_v$ ▷ $\boldsymbol{f}$ is the label of node v
    $v_0, v_1, \ldots, v_{d\cdot\log_2 n} \leftarrow$ path from r to v in T, where $v_0 = r$ and $v_{d\cdot\log_2 n} = v$
    for $q = 1$ `to` d **do**
      $T_q^v \leftarrow$ subtree of T rooted at $v_{(q-1)\cdot\log_2 n}$
      Remove all nodes of T_q^v which are at distance more than $\log_2 n$ from $v_{(q-1)\cdot\log_2 n}$
      Label every node $u \in T_q^v$ as $f_u = (\boldsymbol{f}_u)_q$
      $\mathbf{g} \leftarrow$ FILTERPREPROCESS($T_q^v, v_{q\cdot\log_2 n}, n$)
      $G_q \leftarrow$ FILTERTIME($\mathbf{g}_q, n$)
      $\widehat{G}_q(\xi_q) =$ FILTERFREQUENCY($\mathbf{g}_q, n, \xi_q$)
    $G \leftarrow G_1 \times G_2 \times \ldots \times G_d$
    $h_{\boldsymbol{f}} \leftarrow \sum_{\boldsymbol{\xi}\in[n]^d} \left(\widehat{\chi}_{\boldsymbol{\xi}} \cdot \prod_{q=1}^d \widehat{G}_q(\xi_q)\right)$
    $\widehat{\chi}_{\boldsymbol{f}} \leftarrow \widehat{\chi}_{\boldsymbol{f}} + \left(n^d \cdot \sum_{\boldsymbol{j}\in[n]^d} x_{\boldsymbol{j}} \cdot G_{-\boldsymbol{j}}\right) - h_{\boldsymbol{f}}$
    $T \leftarrow$ Tree.REMOVE(T, v)
  return $\widehat{\chi}$

Theorem 2.2.1 (Estimation guarantee)**.** *Suppose n is a power of two integer and d is a positive integer and $S \subseteq [n]^d$. Then, for any signal $x \in \mathbb{C}^{n^d}$ with* supp $\widehat{x} \subseteq S$*, the procedure* ESTIMATE(x, S, n, d) *(see Algorithm 9) returns $\widehat{x}$. Moreover, the sample complexity of this procedure is $O(|S|^2)$ and its runtime is $O(|S|^2 \cdot d\log_2 n)$. Furthermore,* ESTIMATE *is deterministic.*

Proof. The proof is by induction on the iteration number $t = 0, 1, 2, \ldots$ of the *while* loop in Algorithm 9. One can see that since at each iteration the tree T looses one of its leaves, the algorithm terminates after $|S|$ iterations, since initially the number of leaves of T is $|S|$. Let $\widehat{\chi}^{(t)}$ and $T^{(t)}$ denote the signal $\widehat{\chi}$ and the tree T after finishing up iteration t, respectively, and let $S^{(t)}$ denote the set of frequencies corresponding to leaves of $T^{(t)}$, i.e., $S^{(t)} = \left\{\boldsymbol{f}_u : u \text{ is a leaf of } T^{(t)}\right\}$. In particular, $\widehat{\chi}^{(0)} = 0$ and $T^{(0)}$ is the unflattened splting tree of S and $S^{(0)} = S$.

We claim that for each $t = 0, 1, \ldots, |S|$, the following holds,

$$\text{supp}\left(\widehat{x} - \widehat{\chi}^{(t)}\right) \subseteq S^{(t)} \text{ and } \left|S^{(t)}\right| = |S| - t \tag{2.9}$$

Base case of induction: We have $S^{(0)} = S$ and $\widehat{\chi}^{(0)} \equiv 0$, which immediately implies (2.9) for $t = 0$.

Inductive step: For the inductive hypothesis, let $r \geq 1$ and assume that (2.9) holds for $t = r - 1$. The main loop of the algorithm finds $v = \operatorname{argmin}_{u:\text{ leaf of } T^{(r-1)}} w_{T^{(r-1)}}(u)$. By Claim 2.4.3 along

with inductive hypothesis, $w_{T^{(r-1)}}(v) \le \log_2 |S^{(r-1)}| \le \log_2 |S|$. Note that the main loop of the algorithm constructs an $\left(\boldsymbol{f}_v, S^{(r-1)}\right)$-isolating filter G, along with $\widehat{G}$. In order to do so, the algorithm constructs trees T_q^v for all $q \in \{1, ..., d\}$ which in total takes time $O(|S| d \log n)$. Given T_q^v's, the algorithm constructs filter G and $\widehat{G}$ in time $O\left(2^{w_{T^{(r-1)}}(v)} + d \log n\right) = O\left(|S| + d \log n\right)$, by Lemma 2.4.2. Moreover, the filter G has support size $2^{w_{T^{(r-1)}}(v)} \le |S|$ by Lemma 2.4.2.

By Lemma 2.4.2 computing the quantity $h_{\boldsymbol{f}} = \sum_{\boldsymbol{\xi} \in [n]^d} \widehat{\chi}_{\boldsymbol{\xi}}^{(r-1)} \cdot \widehat{G}(\boldsymbol{\xi})$ in line 16 of Algorithm 9 can be done in time $O\left(\|\widehat{\chi}^{(r-1)}\|_0 \cdot d \log n\right) = O\left(|S| \cdot d \log n\right)$. By convolution theorem 2.3.3, the quantity $h_{\boldsymbol{f}}$ is $h_{\boldsymbol{f}} = n^d \cdot \left(\chi^{(r-1)} * G\right)_0$, and thus

$$\begin{aligned}\left(n^d \cdot \sum_{\boldsymbol{j} \in [n]^d} x_{\boldsymbol{j}} \cdot G_{-\boldsymbol{j}}\right) - h_{\boldsymbol{f}} &= n^d \cdot \left(\left(x - \chi^{(r-1)}\right) * G\right)_0 \\ &= \widehat{x}_{\boldsymbol{f}_v} - \widehat{\chi}_{\boldsymbol{f}_v}^{(r-1)},\end{aligned}$$

where the last transition is due to the fact that G is $\left(\boldsymbol{f}_v, S^{(r-1)}\right)$-isolating along with the inductive hypothesis of $\operatorname{supp}\left(\widehat{x} - \widehat{\chi}^{(r-1)}\right) \subseteq S^{(r-1)}$.

We thus get that $\widehat{\chi}^{(r)}(\cdot) \leftarrow \widehat{\chi}^{(r-1)}(\cdot) + \left(\widehat{x} - \widehat{\chi}^{(r-1)}\right)_{\boldsymbol{f}_v} \cdot \delta_{\boldsymbol{f}_v}(\cdot)$. Moreover, it updates the tree $T^{(r)} \leftarrow$ Tree.REMOVE($T^{(r-1)}, v$). Also note that the set $S^{(r)}$ gets updated to $S^{(r-1)} \setminus \{\boldsymbol{f}_v\}$ accordingly. This establishes (2.9) for $t = r$, thereby completing the inductive step.

Runtime: The number of steps is exactly $|S|$, as follows from the inductive claim. Thus, the total runtime is $O\left(|S|^2 \cdot d \log n\right)$. □

2.6 A Lower Bound of $k^{1-o(1)}$ Rounds of Tree Pruning

One apparent disadvantage of our algorithm presented in the previous section is the fact that it only estimates elements of the Fourier spectrum one at a time, thereby taking k rounds to estimate all elements in the spectrum. Since the isolation of one element takes up to k time due to the support size of G, the resulting bound on the runtime is quadratic in k. A natural conjecture is that our analysis is not tight, and one can achieve better runtime by removing several nodes of weight at most $\log_2 k + O(1)$ at once. If one could argue that the filters G that isolate the nodes removed in one round have nontrivial overlap, runtime improvements could be achieved. In this section we present a class of signals on which $k^{1-o(1)}$ rounds of pruning the tree are required, showing that our analysis is essentially optimal.

Tree pruning process. Suppose n is a power of two integer and τ is a positive integer. Let T be a subtree of T_n^{full}. The *tree pruning process*, $\mathscr{P}(T, \tau, n)$, is an iterative algorithm that performs the following operations on T successively until T is empty:

1. Find $\tilde{S}_\tau = \{\text{leaves } v \text{ of } T : w_T(v) \le \tau\}$, i.e., set of vertices of weight no more than τ.

2. For each $v \in \tilde{S}_\tau$ (in an arbitrary order) remove v from T (i.e., $T.remove(v)$; Algorithm 7).

We show that for every k and sufficiently large integer n there exists a tree T with k leaves such that $\mathscr{P}(T,\tau,n)$ with $\tau = \log_2 k + O(1)$ requires $k^{1-o(1)}$ rounds to terminate. This in particular shows that our k^2 runtime analysis from section 2.5 cannot be improved by reusing work done in a single iteration, and hence our analysis is essentially optimal. Our construction is one-dimensional, although higher dimensional extensions can be readily obtained.

Theorem 2.6.1. *For any integer constant $c \geq 1$, sufficiently large power of two integer n there exists an integer $k = \Theta(\log^c n)$ and a subtree T of T_n^{full} with k leaves such that if $\tau = \log_2 k + O(1)$, then the tree pruning process $\mathscr{P}(T,\tau,n)$ requires $k^{1-o(1)}$ iterations to terminate.*

The following simple lemma is crucial to our analysis

Lemma 2.6.1 (Monotonicity of tree pruning process)**.** *Suppose n is a power of two integer, T' a subtree of T_n^{full}, and T a subtree of T'. Then for every integer τ the number of rounds that it takes $\mathscr{P}(T,\tau,n)$ to collapse T is at most the number of rounds that it takes $\mathscr{P}(T',\tau,n)$ to collapse T'.*

Proof. For $j = 0,1,2,\ldots$, let $T^{(j)}$ (respectively $T'^{(j)}$) denote the tree obtained by performing j rounds of the tree pruning process (with threshold τ) to T (respectively T').

We claim that $T^{(j)}$ is a subtree of $T'^{(j)}$ for all $j = 0,1,\ldots$, which will obviously imply the desired conclusion. We use induction on j. Note that the **base** of induction is trivial for $j = 0$ since $T^{(0)} = T$ and $T'^{(0)} = T'$. Now, we prove the **inductive step**. Suppose $j > 0$. By the inductive hypothesis, we have that $T^{(j-1)}$ is a subtree of $T'^{(j-1)}$. Thus, for any leaf v that appears in both $T^{(j-1)}$ and $T'^{(j-1)}$, we have $w_{T^{(j-1)}}(v) \leq w_{T'^{(j-1)}}(v)$ (this is because any node in $T'^{(j-1)}$ along the path from the root to v that has exactly one child will also have exactly one child in $T^{(j-1)}$). Hence, if v is removed from $T'^{(j-1)}$ in the j-th iteration of the process, then it is also removed from $T^{(j-1)}$ during the j-th iteration. Hence, $T^{(j)}$ is a subtree of $T'^{(j)}$, which completes the inductive step and, therefore, proves the claim. □

We recall a few definitions.

Definition 2.2.8 (Hamming ball)**.** For any power of two integer n any integer $0 \leq c \leq \log_2 n$, we define H_c^n to be the *closed Hamming ball* of radius c centered at 0:

$$H_c^n = \left\{ f \in [n] : w(f) \leq c \right\},$$

where $w(f)$ is the Hamming weight of the binary representation of f, i.e., $w(f)$ is the number of ones in the binary representation of f.

Note that $|H_c^n| = \sum_{j=0}^{c} \binom{\log_2 n}{j}$.

Definition 2.2.9 (Class of signals with low Hamming support)**.** For any power of two integer n and any integer c, Let $\mathscr{X}_c^n$ denote the class of signals in $\mathbb{C}^n$ with support H_c^n as in Definition 2.2.8,

$$\mathscr{X}_c^n = \left\{ x \in \mathbb{C}^n : \operatorname{supp} x \subseteq H_c^n \right\}.$$

Note that for any $x \in \mathscr{X}_c^n$ we have that $\|x\|_0 = \sum_{i=0}^{c} \binom{\log_2 n}{i}$, so for any $c \leq (\frac{1}{2} - \epsilon)\log_2 n$, the signals in the class $\mathscr{X}_c^n$ are $\Theta\left(\binom{\log_2 n}{c}\right)$-sparse.

Definition 2.2.10 (Low Hamming weight splitting trees)**.** For any n a power of two integer, we define a *low Hamming weight splitting tree* T_c^n inductively for $c = 0, 1, \ldots, \log_2 n$:

1. T_0^n is the unique tree of depth $\log_2 n$ that has a single leaf and satisfies the property that each non-leaf node has a single right child only. Thus, T_0^n has $\log_2(n) + 1$ nodes.
2. For any $c > 0$, T_c^n is constructed as follows: Take T_0^n and label the nodes in order from the root to the leaf as $0, 1, \ldots, \log_2 n$. Then, for each node $0 \leq j < \log_2 n$, take a copy of $T_{c-1}^{n/2^{j+1}}$ and append its root as the left child of node j. The resulting tree defines T_c^n.

Note that all leaves of T_c^n are at level $\log_2 n$.

It is not hard to see that T_c^n is in fact the splitting tree for the set H_c^n and, hence, the number of its leaves is $\sum_{i=0}^{c} \binom{\log_2 n}{i}$.

Now, we are ready to prove Theorem 2.6.1.

Proof of Theorem 2.6.1: Let us choose the tree T to be T_c^n for some positive integer c. We will set parameter c at the end of proof. Let $D(n, c, \tau)$ denote the number of iterations required to collapse T_c^n with threshold τ. We prove that,

$$D(n, c, \tau) \geq \frac{\log_2^c n}{c! \cdot \tau^c}, \tag{2.10}$$

for any power of two integer n, any integer $0 \leq c \leq \log_2 n$, and any positive integer τ. We use induction on c.

Base: Note that for $c = 0$, the tree T_c^n has one leaf, which gets removed in the first iteration of the tree pruning process. Thus, $D(n, 0, \tau) = 1$ for any n and $\tau \geq 1$, and so, (2.10) holds for $c = 0$.

Inductive step: Suppose $c > 0$. For any T_c^n, we label the nodes along the path from the root to the rightmost leaf (i.e., the path formed by starting at the root and repeatedly following the right child) in order as $0, 1, \ldots, \log_2 n$.

Note that if $n \leq 2^\tau$, then

$$\frac{\log_2^c n}{c! \cdot \tau^c} \leq \frac{\tau^c}{c! \cdot \tau^c} \leq 1.$$

Thus, (2.10) does indeed hold for $n \leq 2^\tau$.

Now, suppose $n > 2^\tau$. Recall that a copy of $T_{c-1}^{n/2^{j+1}}$ is embedded at the left child of node j of T_c^n for all $j = 0, 1, \ldots, \tau - 1$. We divide the pruning process on T_c^n into two phases. The first phase consists of the process up until the point at which the left subtree of node j in T_c^n completely collapses for some $j \in \{0, 1, \ldots, \tau - 1\}$, while the second phases consists of the process thereafter.

Thus, the number of rounds in the first phase is just the number of rounds till at least one of the top τ left subtrees collapses.

Note that during the first phase, the behavior of the collapsing process on the left subtree of node j corresponds to running a collapsing process with threshold $\tau - j - 1$ on $T_{c-1}^{n/2^{j+1}}$. Thus, the number of rounds in the first phase is,

$$R = \min_{0 \le j < \tau} \left\{ D\left(n/2^{j+1}, c-1, \tau - j - 1\right) \right\}.$$

By the inductive hypothesis (on c), we have that for $j = 0, 1, \ldots, \tau - 1$

$$D\left(n/2^{j+1}, c-1, \tau - j - 1\right) \ge \frac{1}{(c-1)!} \cdot \left(\frac{\log_2 n - j - 1}{\tau - j - 1}\right)^{c-1},$$

which implies that $R \ge \frac{1}{(c-1)!} \cdot \left(\frac{\log_2 n - 1}{\tau - 1}\right)^{c-1}$ since we assumed $\tau \le \log_2 n$.

Now, let T' be the tree obtained after performing R rounds of the collapsing process on T_c^n. Moreover, let T'' be the tree obtained by further removing any left subtrees of nodes $0, 1, \ldots, \tau - 1$. By Lemma 2.6.1, we have that the number of rounds needed to collapse T' is at least the number of rounds needed to collapse T''. Moreover, observe that the number of rounds needed to collapse T'' is precisely $D(n/2^\tau, c, \tau)$, thus, the number of rounds in the second phase is at least $D\left(n/2^\tau, c, \tau\right)$, and so,

$$\begin{aligned} D(n, c, \tau) &\ge R + D\left(n/2^\tau, c, \tau\right) \\ &\ge \frac{1}{(c-1)!} \cdot \left(\frac{\log_2 n - 1}{\tau - 1}\right)^{c-1} + D\left(n/2^\tau, c, \tau\right). \end{aligned}$$

Note that a similar argument gives,

$$D\left(n/2^{a\tau}, c, \tau\right) \ge \frac{1}{(c-1)!} \cdot \left(\frac{\log_2 n - a\tau - 1}{\tau - 1}\right)^{c-1} + D\left(n/2^{(a+1)\tau}, c, \tau\right)$$

for every $a = 0, 1, \ldots, \lfloor (\log_2 n - 1)/\tau \rfloor - 1$ (this condition ensures that $\tau \le \log_2(n/2^{a\tau})$, as required by our argument above). Hence, it follows that,

$$\begin{aligned} D(n, c, \tau) &\ge \sum_{a=0}^{\lfloor (\log_2 n - 1)/\tau \rfloor - 1} \frac{1}{(c-1)!} \cdot \left(\frac{\log_2 n - a\tau - 1}{\tau - 1}\right)^{c-1} + D\left(n/2^{\tau \cdot \lfloor (\log_2 n - 1)/\tau \rfloor}, c, \tau\right) \\ &\ge \frac{1}{(c-1)!} \sum_{a=0}^{\lfloor (\log_2 n - 1)/\tau \rfloor - 1} \left(\frac{\log_2 n}{\tau} - a\right)^{c-1} + 1 \\ &\ge \frac{1}{(c-1)!} \cdot \int_1^{\frac{\log_2 n}{\tau}} u^{c-1}\, du + 1 \\ &= \frac{1}{(c-1)!} \cdot \frac{1}{c} \left(\left(\frac{\log_2 n}{\tau}\right)^c - 1 \right) + 1 \ge \frac{\log_2^c n}{c! \cdot \tau^c}, \end{aligned}$$

which establishes (2.10) for $n > 2^\tau$. This completes the inductive step.

Recall that $k = \Theta\left(\binom{\log_2 n}{c}\right)$, so for any constant c one has $k = \Theta\left(\binom{\log_2 n}{c}\right) \leq \frac{\log_2^c n}{c!}$. Setting $\tau = \log_2 k + O(1)$, we get

$$D(n,c,\tau) \geq \frac{\log_2^c n}{c! \cdot \tau^c} = \Theta(k/(\log_2 k)^c) = k^{1-o(1)},$$

as required. □

2.7 Sparse FFT for Worst-case Sparse Signals and Worst-case Signals with Random Phase

In this section we prove the main result of the paper, namely,

Theorem 2.1.1 (Sparse FFT for worst-case signals, formal version)**.** *For any power of two integer n and any positive integer d and any signal $x \in \mathbb{C}^{n^d}$ with $\|\widehat{x}\|_0 = k$, the procedure* SPARSEFFT(x,n,d,k) *in Algorithm 11 recovers $\widehat{x}$. Moreover, the sample complexity of this procedure is $O(k^3 \log^2 k \log^2 N)$ and its runtime is $O(k^3 \log^2 k \log^2 N)$, where $N = n^d$.*

We also present improved recovery algorithms for Fourier sparse signals x whose nonzero frequencies are distributed arbitrarily (worst-case) and values at the nonzero frequencies are independently chosen to have a uniformly random phase. Recall Definition 2.2.7:

Definition 2.2.7 (Worst-case signal with random phase)**.** For any positive integer d and power of two n, we define x to be a *worst-case signal with random phase* having values $\{\beta_{\boldsymbol{f}}\}_{\boldsymbol{f} \in [n]^d}$ if,

$$\widehat{x}_{\boldsymbol{f}} = \beta_{\boldsymbol{f}} e^{2\pi i \theta} \quad \text{for uniformly random } \theta \in [0, 2\pi),$$

independently for every $\boldsymbol{f} \in [n]^d$. Furthermore, if k of the values $\{\beta_{\boldsymbol{f}}\}_{\boldsymbol{f} \in [n]^d}$ are nonzero, then x is said to be a *worst-case k-sparse signal with random phase* and is guaranteed to have sparsity $\|\widehat{x}\|_0 = k$.

For this model we prove the stronger result:

Theorem 2.2.2 (Sparse FFT for worst-case signals with random phase)**.** *For any power of two integer n, positive integer d, and worst-case k-sparse signal with random phase $x : [n]^d \to \mathbb{C}$, the procedure* SPARSEFFT-RANDOMPHASE(x,n,d,k) *in Algorithm 12 recovers $\widehat{x}$ with probability $1 - \frac{1}{N^2}$. Moreover, the sample complexity and runtime of this procedure are both $O\left(k^2 \log^4 N\right)$.*

The main property that allows us to obtain the stronger result is that a small number of time domain samples from such a signal suffice to approximate its energy with high confidence (whereas $\Omega(k)$ samples are required in general for a worst-case k-sparse signal). This is reflected by the following lemma.

Lemma 2.7.1. *For any power of two integr n, any positive integer d, and any worst-case signal with random phase x,*

$$\Pr\left[\frac{1}{2}\cdot\frac{\|\beta\|_2^2}{n^{2d}} \le \frac{1}{s}\sum_{j=1}^{s}|x_{\boldsymbol{t}_j}|^2 \le \frac{3}{2}\cdot\frac{\|\beta\|_2^2}{n^{2d}}\right] \ge 1-\frac{1}{n^{4d}},$$

where $s = Cd^3\log_2^3 n$ for some absolute constant $C > 0$ and $\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_s \sim \mathrm{Unif}([n]^d)$ are i.i.d. random variables. The probability is over the randomness in choosing the variables $\boldsymbol{t}_j$ as well the randomness in the choice of the phase for each frequency of $\widehat{x}$.

Proof. This sort of guarantee for signals with random phase is well-known and follows from standard application of Bernstein's inequality. See for example (Kapralov et al., 2019). □

2.7.1 Proofs of Theorems 2.1.1 and 2.2.2

Given the construction of our adaptive aliasing filter from the previous section, our sparse recovery algorithms follow by a reduction to the estimation problem. Our algorithm starts by first finding the vertex $v^* = \mathrm{argmin}_{v\in T} w_T(v)$, which, by Kraft's inequality, satisfies $w_T(v^*) \le \log_2 k$. We then define an auxiliary tree T' by appending a left a and a right child b to v. Then for each of the children a, b, we, in turn, construct a filter G that isolates them from the rest of T (i.e., from the frequency cones of other nodes in T) and check whether the corresponding restricted signals are nonzero. The latter is unfortunately a nontrivial task, since the sparsity of these signals can be as high as k, and detecting whether a k-sparse signal is nonzero requires $\Omega(k)$ samples. However, a fixed set of $O\left(k\log^3 N\right)$ locations that satisfies the restricted isometry property (RIP) can be selected, and accessing the signal on those values suffices to test whether it is nonzero. If the signal is further assumed to be a worst-case random phase signal, then a polylogarithmic number of samples suffices. The following lemma (Lemma 2.7.2) makes the latter claim formal.

Lemma 2.7.2 (ZEROTEST guarantee)**.** *Suppose d is a positive integer and n is a power of two. Assume that signals $x, \widehat{\chi} \in \mathbb{C}^{n^d}$ satisfy* supp $(\widehat{x}-\widehat{\chi}) \subseteq \bigcup_{u:\ \text{leaf of } T} \mathrm{FrequencyCone}_T(u)$ *for some T that is a subtree of $T^{\mathrm{full}}_{n^d}$. For every leaf v of T if $\boldsymbol{\Delta}$ is a multiset of elements from $[n]^d$ which satisfies the following:*

$$\frac{1}{2}\cdot\frac{\|\widehat{y}\|_2^2}{n^{2d}} \le \frac{1}{|\boldsymbol{\Delta}|}\cdot\sum_{\Delta\in\boldsymbol{\Delta}}|y_\Delta|^2 \le \frac{3}{2}\cdot\frac{\|\widehat{y}\|_2^2}{n^{2d}},$$

where $y = (\widehat{x}-\widehat{\chi})_{\mathrm{FrequencyCone}_T(v)}$ is the signal obtained by restricting $\widehat{x}-\widehat{\chi}$ to frequencies in $\mathrm{FrequencyCone}_T(v)$ *and zeroing it out everywhere else, then the following conditions hold:*

- ZEROTEST$(x, \widehat{\chi}, T, v, n, d, \boldsymbol{\Delta})$ *outputs* **true** *if* supp $(\widehat{x}-\widehat{\chi}) \cap \mathrm{FrequencyCone}_T(v) \neq \emptyset$*; otherwise, it outputs* **false**.
- *The sample complexity of this procedure is $O\left(2^{w_T(v)}\cdot|\boldsymbol{\Delta}|\right)$, where $w_T(v)$ is the weight of leaf v in T (see Definition 2.2.6).*

- *The runtime of the* ZEROTEST *procedure is* $O\left(\|\widehat{\chi}\|_0 \cdot |\mathbf{\Delta}| + |T| \cdot d \log n + 2^{w_T(v)} \cdot |\mathbf{\Delta}|\right)$*, where* $|T|$ *denotes the number of leaves of* T*.*

Proof. Consider lines 13-14 of Algorithm 10. By Claim 2.3.3, we have,

$$\begin{aligned} h_{\boldsymbol{f}}^{\Delta} &= \frac{1}{n^d} \sum_{\boldsymbol{\xi} \in [n]^d} e^{2\pi i \frac{\boldsymbol{\xi}^T \Delta}{n}} \cdot \widehat{\chi}_{\boldsymbol{\xi}} \widehat{G}_{\boldsymbol{\xi}} \\ &= \sum_{\boldsymbol{j} \in [n]^d} G_{\Delta - \boldsymbol{j}} \cdot \chi_{\boldsymbol{j}}, \end{aligned}$$

where $G \in \mathbb{C}^{n^d}$ is the filter constructed in lines 5-12 of Algorithm 10. Thus,

$$\begin{aligned} H_{\boldsymbol{f}}^{\Delta} &= \left(\sum_{\boldsymbol{j} \in [n]^d} G_{\Delta - \boldsymbol{j}} \cdot x_{\boldsymbol{j}} \right) - h_{\boldsymbol{f}}^{\Delta} \\ &= \sum_{\boldsymbol{j} \in [n]^d} G_{\Delta - \boldsymbol{j}} \cdot (x - \chi)_{\boldsymbol{j}}. \end{aligned}$$

Note that, by Lemma 2.4.2, the filter G in Algorithm 10 is a (v, T)-isolating filter. Therefore, by the assumption $\operatorname{supp}(\widehat{x} - \widehat{\chi}) \subseteq \bigcup_{u:\text{ leaf of } T} \text{FrequencyCone}_T(u)$ and the definition of a (v, T)-isolating filter (see Definition 2.4.6), we have,

$$\begin{aligned} H_{\boldsymbol{f}}^{\Delta} &= \sum_{\boldsymbol{j} \in [n]^d} G_{\Delta - \boldsymbol{j}} \cdot (x - \chi)_{\boldsymbol{j}} \\ &= \frac{1}{n^d} \sum_{\boldsymbol{\xi} \in \text{FrequencyCone}_T(v)} (\widehat{x} - \widehat{\chi})_{\boldsymbol{\xi}} \cdot e^{2\pi i \frac{\boldsymbol{\xi}^T \Delta}{n}}. \end{aligned}$$

Therefore, $H_{\boldsymbol{f}}^{\Delta}$ is the inverse Fourier transform of $(\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}$ at time Δ, where $(\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}$ denotes the signal obtained by restricting $\widehat{x} - \widehat{\chi}$ to frequencies $\boldsymbol{\xi} \in \text{FrequencyCone}_T(v)$ and zeroing out the signal on all other frequencies. By the assumption of lemma the following holds:

$$\frac{1}{2} \cdot \frac{\|(\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}\|_2^2}{n^{2d}} \leq \frac{1}{|\mathbf{\Delta}|} \cdot \sum_{\Delta \in \mathbf{\Delta}} |H_{\boldsymbol{f}}^{\Delta}|^2 \leq \frac{3}{2} \cdot \frac{\|(\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}\|_2^2}{n^{2d}}.$$

Therefore, the test performed in line 15 of Algorithm 10 correctly identifies if the restricted signal $(\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}$ is zero or not, hence, the first claim of the lemma holds.

Note that in order to construct a (v, T)-isolating filter G, along with $\widehat{G}$, the algorithm constructs trees T_q^v for all $q \in \{1, ..., d\}$, which has a total time complexity $O\left(|T| d \log n\right)$. Given T_q^v's, the algorithm constructs filter G and $\widehat{G}$ in time $O\left(2^{w_T(v)} + d \log n\right)$, by Lemma 2.4.2. Moreover, the filter G has support size $2^{w_T(v)}$, by Lemma 2.4.2.

By Lemma 2.4.2, computing the quantities $h_{\boldsymbol{f}}^{\Delta} = \frac{1}{n^d} \sum_{\boldsymbol{\xi} \in [n]^d} e^{2\pi i \frac{\boldsymbol{\xi}^T \Delta}{n}} \cdot \widehat{\chi}_{\boldsymbol{\xi}} \widehat{G}_{\boldsymbol{\xi}}$ for all $\Delta \in \mathbf{\Delta}$ in line 13 of Algorithm 10 can be done in time $O\left(\|\widehat{\chi}\|_0 \cdot \left(|\mathbf{\Delta}| + d \log n\right)\right) = O\left(\|\widehat{\chi}\|_0 \cdot |\mathbf{\Delta}|\right)$. Given the values of

Algorithm 10 Procedure for testing zero hypothesis

1: **procedure** ZEROTEST($x, \widehat{\chi}, T, v, n, d, \mathbf{\Delta}$) ▷ $\mathbf{\Delta}$: multiset of elements from $[n]^d$
2: $\boldsymbol{f} \leftarrow \boldsymbol{f}_v$, $l \leftarrow l_T(v)$, $q^* \leftarrow \left\lceil \frac{l}{\log_2 n} \right\rceil$
3: $v_0, v_1, \ldots, v_l \leftarrow$ path from r to v in T, where $v_0 = r$ and $v_l = v$
4: $(u_1, u_2, \cdots, u_{q^*-1}, u_{q^*}) \leftarrow (v_{\log_2 n}, v_{2\log_2 n}, \cdots, v_{(q^*-1)\cdot\log_2 n}, v_l)$
5: **for** $q = 1$ `to` q^* **do**
6: $T_q^v \leftarrow$ subtree of T rooted at u_{q-1}
7: Remove all nodes of T_q^v which are at distance more than $\log_2 n$ from u_{q-1}
8: Label every node $w \in T_q^v$ as $f_w = (\boldsymbol{f}_w)_q$
9: $\mathbf{g}_q \leftarrow$ FILTERPREPROCESS(T_q^v, u_q, n)
10: $G_q \leftarrow$ FILTERTIME($\mathbf{g}_q, n$)
11: $\widehat{G}_q(\xi_q) =$ FILTERFREQUENCY($\mathbf{g}_q, n, \xi_q$)
12: $G \leftarrow G_1 \times G_2 \times \ldots \times G_{q^*}$
13: $h_{\boldsymbol{f}}^{\Delta} \leftarrow \frac{1}{n^d} \sum_{\boldsymbol{\xi} \in [n]^d} \left(e^{2\pi i \frac{\xi^T \Delta}{n}} \cdot \widehat{\chi}_{\boldsymbol{\xi}} \cdot \prod_{q=1}^{q^*} \widehat{G}_q(\xi_q) \right)$ for all $\Delta \in \mathbf{\Delta}$
14: $H_{\boldsymbol{f}}^{\Delta} \leftarrow \left(\sum_{\boldsymbol{j} \in [n]^d} G(\Delta - \boldsymbol{j}) \cdot x_{\boldsymbol{j}} \right) - h_{\boldsymbol{f}}^{\Delta}$ for all $\Delta \in \mathbf{\Delta}$
15: **if** $\frac{1}{|\mathbf{\Delta}|} \sum_{\Delta \in \Delta} |H_{\boldsymbol{f}}^{\Delta}|^2 = 0$ **then**
16: **return false**
17: **else**
18: **return true**

$h_{\boldsymbol{f}}^{\Delta}$ for all $\Delta \in \mathbf{\Delta}$, computing $\left\{ |H_{f^*}^{\Delta}|^2 \right\}_{\Delta \in \mathbf{\Delta}}$ in line 14 takes time $O\left(2^{w_T(v)} \cdot |\mathbf{\Delta}|\right)$ because $|\text{supp } G| = 2^{w_T(v)}$. Therefore, the total runtime of this procedure is,

$$O\left(|T| \cdot d \log n + 2^{w_T(v)} \cdot |\mathbf{\Delta}| + \|\widehat{\chi}\|_0 \cdot |\mathbf{\Delta}|\right),$$

as desired.

Because support size of G is $2^{w_T(v)}$, computing all $\left\{ |H_{f^*}^{\Delta}|^2 \right\}_{\Delta \in \mathbf{\Delta}}$ in line 14 of the algorithm requires $O\left(2^{w_T(v)} \cdot |\mathbf{\Delta}|\right)$ samples from x which proves the second claim of the lemma.

□

Now we are in a position to prove our main result:

Proof of Theorems 2.1.1 and 2.2.2: Note that Algorithms 11 and 12 are identical except in line 2. We first analyze the common code of the algorithms (after line 2) under the assumption that the set $\mathbf{\Delta}$ are replaced with a more powerful set which satisfies the precondition of Lemma 2.7.2 in all calls to ZEROTEST, hence, ZEROTEST correctly tests the zero hypothesis on its input signal with probability 1. We then establish a coupling between this idealized execution and the actual execution for both Algorithms 11 and 12, leading to our result.

Let m denote the size of the set $m = |\mathbf{\Delta}|$. By induction, we prove that the following properties are maintained throughout the execution of SPARSEFFT (Algorithm 11) and SPARSEFFT-

Algorithm 11 Sparse FFT for worst-case sparse signals

procedure SPARSEFFT(x, n, d, k)
  $\mathbf{\Delta} \leftarrow$ Multiset of $[n]^d$ corresponding to Fourier measurements satisfying RIP of order k
  ▷ $|\mathbf{\Delta}| = O\left(k \log^2 k \cdot d \log n\right)$, see Theorem 2.3.1
  $T \leftarrow \{r\}, \boldsymbol{f}_r \leftarrow 0$
  while $T \neq \emptyset$ **do**
    $v \leftarrow \operatorname{argmin}_{u:\text{ leaf of } T} w_T(u)$, $\boldsymbol{f} \leftarrow \boldsymbol{f}_v$ and $l \leftarrow l_T(v)$
    if $l = d \log_2 n$ **then** ▷ All bits of v have been discovered
      $v_0, v_1, \ldots, v_{d \cdot \log_2 n} \leftarrow$ path from r to v in T, where $v_0 = r$ and $v_{d \cdot \log_2 n} = v$
      for $q = 1$ `to` d **do**
        $T_q^v \leftarrow$ subtree of T rooted at $v_{(q-1) \cdot \log_2 n}$
        Remove all nodes of T_q^v at distance more than $\log_2 n$ from $v_{(q-1) \cdot \log_2 n}$
        Label every node $u \in T_q^v$ as $f_u = (\boldsymbol{f}_u)_q$
        $\mathbf{g}_q \leftarrow$ FILTERPREPROCESS($T_q^v, v_{q \cdot \log_2 n}, n$)
        $G_q \leftarrow$ FILTERTIME($\mathbf{g}_q, n$)
        $\widehat{G}_q(\xi_q) =$ FILTERFREQUENCY($\mathbf{g}_q, n, \xi_q$)
      $G \leftarrow G_1 \times G_2 \times \ldots \times G_d$
      $h_{\boldsymbol{f}} \leftarrow \sum_{\boldsymbol{\xi} \in [n]^d} \left(\widehat{\chi}_{\boldsymbol{\xi}} \cdot \prod_{q=1}^d \widehat{G}_q(\xi_q)\right)$
      $\widehat{\chi}_{\boldsymbol{f}} \leftarrow \widehat{\chi}_{\boldsymbol{f}} + \left(n^d \cdot \sum_{\boldsymbol{j} \in [n]^d} x_{\boldsymbol{j}} \cdot G_{-\boldsymbol{j}}\right) - h_{\boldsymbol{f}}$
      $T \leftarrow$ Tree.REMOVE(T, v)
    else
      $T' \leftarrow T \cup \{\text{left child } u \text{ of } v\} \cup \{\text{right child } w \text{ of } v\}$
      if ZEROTEST$\left(x, \widehat{\chi}, T', w, n, d, \mathbf{\Delta}\right)$ **then**
        Add w as the right child of node v to tree T
        $\boldsymbol{f}_w \leftarrow \boldsymbol{f}$ ▷ Frequency corresponding to node w
      if ZEROTEST$\left(x, \widehat{\chi}, T', u, n, d, \mathbf{\Delta}\right)$ **then**
        Add u as the left child of node v to tree T
        $\boldsymbol{f}_u \leftarrow \boldsymbol{f} + 2^l$; ▷ Frequency corresponding to node u
  return $\widehat{\chi}$;

RANDOMPHASE (Algorithm 12):

(1) $\operatorname{supp}(\widehat{x} - \widehat{\chi}) \subseteq \bigcup_{u:\text{ leaf of } T} \text{FrequencyCone}_T(u)$;

(2) For every leaf u of tree T one has $\operatorname{supp}(\widehat{x} - \widehat{\chi}) \cap \text{FrequencyCone}_T(u) \neq \emptyset$;

(3) If $\widehat{x}$ is a worst-case signal with random phase, then $\widehat{x} - \widehat{\chi}$ is a worst-case signal with random phase;

(4) The quantity $\phi = (d \log_2 n + 1)\|\widehat{x} - \widehat{\chi}\|_0 - \sum_{u:\text{ leaf of } T} l_T(u)$ decreases by at least 1 in every iteration of Algorithms 11 and 12;

(5) Always $\|\widehat{x} - \widehat{\chi}\|_0 \leq k$;

The **base** of induction is provided by the first iteration, at which point T is a single vertex

Algorithm 12 Sparse FFT for worst-case sparse signals with random phase

1: **procedure** SPARSEFFT-RANDOMPHASE(x, n, d, k)
2: $\mathbf{\Delta} \leftarrow$ Multiset$\left\{\Delta_i : \Delta_i \sim \text{Unif}([n]^d), \forall i \in \left[Cd^3 \log_2^3 n\right]\right\}$ ▷ C: constant; see Lemma 2.7.1
3: $T \leftarrow \{r\}, \boldsymbol{f}_r \leftarrow 0$
4: **while** $T \neq \emptyset$ **do**
5: $v \leftarrow \text{argmin}_{u:\text{ leaf of } T} w_T(u)$, $\boldsymbol{f} \leftarrow \boldsymbol{f}_v$ and $l \leftarrow l_T(v)$
6: **if** $l = d \log_2 n$ **then** ▷ All bits of v have been discovered
7: $v_0, v_1, \ldots, v_{d\cdot\log_2 n} \leftarrow$ path from r to v in T, where $v_0 = r$ and $v_{d\cdot\log_2 n} = v$
8: **for** $q = 1$ `to` d **do**
9: $T_q^v \leftarrow$ subtree of T rooted at $v_{(q-1)\cdot\log_2 n}$
10: Remove all nodes of T_q^v at distance more than $log_2 n$ from $v_{(q-1)\cdot\log_2 n}$
11: Label every node $u \in T_q^v$ as $f_u = (\boldsymbol{f}_u)_q$
12: $\mathbf{g}_q \leftarrow$ FILTERPREPROCESS$(T_q^v, v_{q\cdot\log_2 n}, n)$
13: $G_q \leftarrow$ FILTERTIME$(\mathbf{g}_q, n)$
14: $\widehat{G}_q(\xi_q) =$ FILTERFREQUENCY$(\mathbf{g}_q, n, \xi_q)$
15: $G \leftarrow G_1 \times G_2 \times \ldots \times G_d$
16: $h_{\boldsymbol{f}} \leftarrow \sum_{\boldsymbol{\xi}\in[n]^d} \left(\widehat{\chi}_{\boldsymbol{\xi}} \cdot \prod_{q=1}^d \widehat{G}_q(\xi_q)\right)$
17: $\widehat{\chi}_{\boldsymbol{f}} \leftarrow \widehat{\chi}_{\boldsymbol{f}} + \left(n^d \cdot \sum_{\boldsymbol{j}\in[n]^d} x_{\boldsymbol{j}} \cdot G_{-\boldsymbol{j}}\right) - h_{\boldsymbol{f}}$
18: $T \leftarrow$ Tree.REMOVE(T, v)
19: **else**
20: $T' \leftarrow T \cup \{\text{left child } u \text{ of } v\} \cup \{\text{right child } w \text{ of } v\}$
21: **if** ZEROTEST$\left(x, \widehat{\chi}, T', w, n, d, \mathbf{\Delta}\right)$ **then**
22: Add w as the right child of node v to tree T
23: $\boldsymbol{f}_w \leftarrow \boldsymbol{f}$ ▷ Frequency corresponding to node w
24: **if** ZEROTEST$\left(x, \widehat{\chi}, T', u, n, d, \mathbf{\Delta}\right)$ **then**
25: Add u as the left child of node v to tree T
26: $\boldsymbol{f}_u \leftarrow \boldsymbol{f} + 2^l$; ▷ Frequency corresponding to node u
27: **return** $\widehat{\chi}$;

$T = \{r\}$ where r is the root with $\boldsymbol{f}_r = 0$ and $\widehat{\chi} = 0$. The conditions **(1)** and **(2)** and **(3)** and **(5)** are satisfied since FrequencyCone$_T(r) = [n]^d$ and supp $(\widehat{x} - \widehat{\chi}) =$ supp $\widehat{x} \neq \emptyset$ and $\widehat{x} - \widehat{\chi} = \widehat{x}$ is a worst-case signal with random phase if $\widehat{x}$ is a worst-case signal with random phase.

We now prove the **inductive step** by assuming that conditions **(1)** and **(2)** and **(3)** and **(5)** of the inductive hypothesis are satisfied at the beginning of a certain iteration and arguing that conditions **(1)** and **(2)** and **(3)** and **(5)** are maintained at the end of iteration. We also show that the value of the quantity ϕ defined in **(4)**, decreases by at least one at every iteration. Let $v \in T$ be the smallest weight leaf chosen by the algorithms in line 5. We now consider two cases.

Case 1: $l_T(v) = d \log_2 n$**.** Since the filter G constructed in Algorithms 11 and 12 is a (v, T)-isolating filter, we have by Definition 2.2.5 that for every signal $z \in \mathbb{C}^{n^d}$ with Fourier support

supp $\widehat{z} \subseteq \bigcup_{u\text{: leaf in } T} \text{FrequencyCone}_T(u)$ and for all $t \in [n]^d$,

$$\sum_{\boldsymbol{j}\in[n]^d} z_{\boldsymbol{j}} G_{\boldsymbol{t}-\boldsymbol{j}} = \frac{1}{n^d} \sum_{\boldsymbol{f}'\in \text{FrequencyCone}_T(v)} \widehat{z}_{\boldsymbol{f}'} e^{2\pi i \frac{\boldsymbol{f}'^T \boldsymbol{t}}{n}}. \tag{2.11}$$

By condition **(1)** of the inductive hypothesis one has supp $(\widehat{x}-\widehat{\chi}) \subseteq \bigcup_{u\text{: leaf of } T} \text{FrequencyCone}_T(u)$, and thus we can apply (2.11) with $z = x - \chi$ and $\boldsymbol{t} = 0$, obtaining,

$$\sum_{\boldsymbol{j}\in[n]^d} (x-\chi)_{\boldsymbol{j}} G_{-\boldsymbol{j}} = \frac{1}{n^d} \sum_{\boldsymbol{f}'\in \text{FrequencyCone}_T(v)} \widehat{(x-\chi)}_{\boldsymbol{f}'}. \tag{2.12}$$

Note that by Claim 2.3.3,

$$n^d \cdot \sum_{\boldsymbol{j}\in[n]^d} \chi_{\boldsymbol{j}} G_{-\boldsymbol{j}} = \sum_{f\in[n]} \widehat{\chi}_f \widehat{G}_f = h_{\boldsymbol{f}},$$

where $h_{\boldsymbol{f}}$ is the quantity computed in line 16. We thus get that,

$$\begin{aligned} n^d \sum_{\boldsymbol{j}\in[n]^d} x_{\boldsymbol{j}} \cdot G_{-\boldsymbol{j}} - h_{\boldsymbol{f}} &= \sum_{\boldsymbol{f}'\in \text{FrequencyCone}_T(v)} \widehat{(x-\chi)}_{\boldsymbol{f}'} \\ &= \widehat{(x-\chi)}_{\boldsymbol{f}_v}, \end{aligned}$$

because $\text{FrequencyCone}_T(v) = \{\boldsymbol{f}_v\}$ due to the assumption $l_T(v) = d\log_2 n$. Therefore, the operation in line 17 is in fact $\widehat{\chi}(\cdot) \leftarrow \widehat{\chi}(\cdot) + \widehat{(x-\chi)}_{\boldsymbol{f}_v} \delta_{\boldsymbol{f}_v}(\cdot)$ and hence, at the end of the loop we have $\widehat{(x-\chi)}_{\boldsymbol{f}_v} = 0$ which means that $\boldsymbol{f}_v$ will no longer be in supp $\widehat{(x-\chi)}$. And also v gets removed from tree T implying that $\{\boldsymbol{f}_v\} = \text{FrequencyCone}_T(v)$ will be excluded from $\bigcup_{u\text{: leaf of } T} \text{FrequencyCone}_T(u)$. Note that this also implies that $\widehat{(x-\chi)}$ will remain a worst-case signal with random phase. Therefore, condition **(1)** and **(2)** and **(3)** hold.

Additionally, note that $\|\widehat{(x-\chi)}\|_0$ will decrease by exactly 1 because $\boldsymbol{f}_v$ is no longer in supp $\widehat{(x-\chi)}$ and the rest of the support is unchanged. This shows that condition **(5)** holds. Moreover, $\sum_{u\text{: leaf of } T} l_T(u)$ decreases by exactly $d\log_2 n$ because the level of v was $l_T(v) = d\log_2 n$ and v gets removed from T. So ϕ will decrease by one as required in condition **(4)**.

Case 2: Suppose that $l_T(v) < d\log_2 n$. We first verify that the preconditions of Lemma 2.7.2 are satisfied. We need to ensure that for the residual signal $\widehat{x} - \widehat{\chi}$ it holds that,

$$\text{supp}\,(\widehat{x}-\widehat{\chi}) \subseteq \bigcup_{u\text{: leaf of } T'} \text{FrequencyCone}_{T'}(u),$$

where T' is the tree obtained from T by adding two children of v (line 20). This follows, since by the inductive hypothesis,

$$\text{supp}\,(\widehat{x}-\widehat{\chi}) \subseteq \bigcup_{u\text{: leaf of } T} \text{FrequencyCone}_T(u),$$

and by claim 2.4.2 we have,

$$\text{FrequencyCone}_T(v) = \text{FrequencyCone}_{T'}(u) \cup \text{FrequencyCone}_{T'}(w).$$

We thus get that the preconditions of Lemma 2.7.2 are satisfied, therefore, the output of ZEROTEST$(x, \widehat{\chi}, T', w, n, d, \Delta)$ is **true** if $(\widehat{x}-\widehat{\chi})_{\text{FrequencyCone}_{T'}(w)} \neq 0$ and **false** otherwise. A similar argument shows that the algorithm correctly tests the zero hypothesis on $(\widehat{x}-\widehat{\chi})_{\text{FrequencyCone}_{T'}(u)}$. Thus, by letting T_{new} denote the tree T at the end of the while loop, we have that,

$$\text{supp}\,(\widehat{x}-\widehat{\chi}) \subseteq \bigcup_{u:\text{ leaf of } T_{new}} \text{FrequencyCone}_{T_{new}}(u),$$

and for every $v \in T_{new}$ one has $\text{supp}\,(\widehat{x}-\widehat{\chi}) \cap \text{FrequencyCone}_{T_{new}}(v) \neq \emptyset$. Hence, because $\widehat{(x-\chi)}$ remains unchanged, conditions **(1)** and **(2)** and **(3)** hold at the end of the iteration.

Now, we show that ϕ has decreased by at least one. By inductive hypothesis we have that $\text{supp}\,(\widehat{x}-\widehat{\chi}) \cap \text{FrequencyCone}_T(v) \neq \emptyset$ and at least one of w or u will be added to T because $\text{FrequencyCone}_T(v) = \text{FrequencyCone}_{T'}(u) \cup \text{FrequencyCone}_{T'}(v)$. Since $l_{T_{new}}(w) = l_{T_{new}}(u) = l_T(v)+1$, $\sum_{u':\text{ leaf of } T} l_{T_{new}}(u') \geq \sum_{u':\text{ leaf of } T} l_T(u')+1$. Because $\|\widehat{x}-\widehat{\chi}\|_0$ remains unchanged, the value of ϕ decreases by at least one hence conditions **(4)** and **(5)** hold.

We have established by induction that conditions **(1)**, **(2)**, **(3)**, **(4)**, and **(5)** hold throughout the execution of SPARSEFFT (Algorithm 11) and SPARSEFFT-RANDOMPHASE (Algorithm 12). Now we show that these conditions are sufficient to prove the correctness and convergence of our algorithms.

Because $l_T(u) \leq d\log_2 n$ for every leaf $u \in T$, it follows from condition (2) that the quantity $\phi = (d\log_2 n+1)\|\widehat{x}-\widehat{\chi}\|_0 - \sum_{u:\text{ leaf of } T} l_T(u)$ is non-negative. At the first iteration, $\widehat{\chi}=0$ and $T=\{r\}$ where r is the root with $l_T(f)=0$. Hence, $\phi = \|\widehat{x}\|_0 \cdot (1+d\log_2 n)$ at the start of our algorithms. Because ϕ is decreasing by at least 1 at each iteration, the algorithm must terminate after $O\left(\|\widehat{x}\|_0 \cdot d\log n\right)$ iterations. By Lemma 2.7.2 along with Claim 2.4.3 and noting that $\|\widehat{\chi}\|_0 = O(k)$ by condition **(5)**, the runtime as well as the sample complexityof each iteration of our algorithms are $O(km)$, therefore, the total runtime and sample complexity will both be $O\left(k^2 m \cdot d\log n\right)$.

Finally, observe that throughout this analysis we have assumed that the set $\mathbf{\Delta}$ satisfies the precondition of Lemma 2.7.2 for all the invocations of ZEROTEST by our algorithm. In reality, there are two cases.

The first case is for worst-case signals (Algorithm 11, Theorem 2.1.1). In this case, the algorithm chooses $\mathbf{\Delta}$ to be a multiset of Fourier measurements that satisfies the RIP of order k. Let F_N^{-1} be the d dimensional inverse Fourier transform's matrix with $N = n^d$ points. By Theorem 2.3.1

there exists a multiset $\mathbf{\Delta}$ of size $m = O\left(k \log^2 k \cdot d \log n\right)$ such that, for every signal $y \in \mathbb{C}^{n^d}$:

$$\frac{1}{2} \cdot \frac{\|\widehat{y}\|_2^2}{n^{2d}} \leq \frac{1}{|\mathbf{\Delta}|} \cdot \sum_{\Delta \in \mathbf{\Delta}} |y_\Delta|^2 \leq \frac{3}{2} \cdot \frac{\|\widehat{y}\|_2^2}{n^{2d}}$$

As we have shown in condition **(5)** of the induction, $\|\widehat{x} - \widehat{\chi}\|_0 \leq k$. Therefore, for every leaf v of the tree T, $\|(\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}\|_0 \leq k$, and so, the precondition of lemma 2.7.2 is satisfied.

The second case is for worst-case signals with random phase (Algorithm 12, Theorem 2.2.2). We have shown in condition **(3)** of the induction that $x - \chi$ is a worst-case signal with random phase in every iteration of the algorithm. Therefore for every leaf v of the tree it is true that $(\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}$ is a worst-case signal with random phase. In this case, therefore, Lemma 2.7.1 implies that for every fixed leaf v of tree T, the multiset $\mathbf{\Delta}$ defined in line 2 of Algorithm 12 satisfies the following with probability at least $1 - 1/n^{4d}$,

$$\frac{1}{2} \cdot \frac{\|\widehat{y}\|_2^2}{n^{2d}} \leq \frac{1}{|\Delta|} \cdot \sum_{\Delta \in \Delta} |y_\Delta|^2 \leq \frac{3}{2} \cdot \frac{\|\widehat{y}\|_2^2}{n^{2d}}$$

where $y = (\widehat{x} - \widehat{\chi})_{\text{FrequencyCone}_T(v)}$.

This shows that in the second case which corresponds to theorem 2.2.2, the failure probability of procedure ZEROTEST is at most $\frac{1}{n^{4d}}$. Moreover, the above analysis shows that SPARSEFFT-RANDOMPHASE makes at most $O\left(kd \log n\right)$ calls to ZEROTEST. Therefore, by a union bound, the overall failure probability of the calls to ZEROTEST is $O\left((kd \log n) \frac{1}{n^{4d}}\right) \leq n^{-2d}$. Hence, we obtain the desired result.

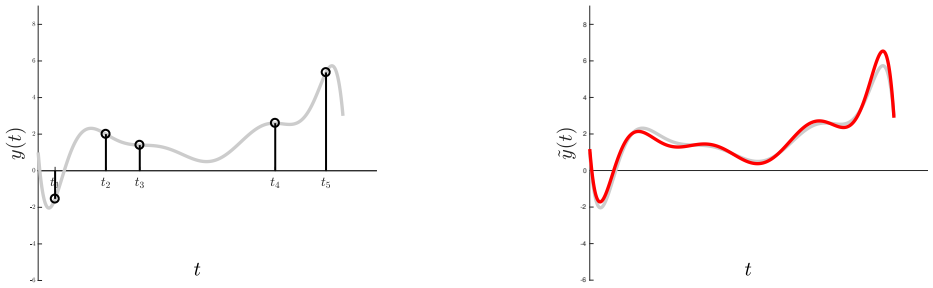
□

3 Near-optimal Recovery of Signals with Simple Fourier Transforms

This chapter is based on a joint work with Haim Avron, Michael Kapralov, Cameron Musco, Christopher Musco, and Ameya Velingker. It has been accepted to the 51st Annual ACM SIGACT Symposium on Theory of Computing (Avron et al., 2019, STOC).

3.1 Introduction

Consider the following fundamental function fitting problem, pictured in Figure 3.1. We can access a continuous signal $y(t)$ at any time $t \in [0, T]$. We wish to select a finite set of sample times $t_1, \ldots, t_q$ such that, by observing the signal values $y(t_1), \ldots, y(t_q)$ at those samples, we are able to find a good approximation $\tilde{y}$ to y over the entire range $[0, T]$. We also study the problem in a noisy setting, where for each sample t_i, we only observe $y(t_i) + n(t_i)$ for some fixed noise function n.



(a) Observed signal y sampled at times $t_1, \ldots, t_q$. (b) Reconstructed signal $\tilde{y}$ based on samples.

Figure 3.1 – Our basic function fitting problem requires reconstructing a continuous signal based on a small number of (possibly noisy) discrete samples.

We seek to understand:

1. How many samples q are required to approximately reconstruct y and how should we select these samples?

2. After sampling, how can we find and represent $\tilde{y}$ in a computationally efficient way?

Answering these questions requires assumptions about the underlying signal y. In particular, for the information at our samples $t_1, \ldots, t_q$ to be useful in reconstructing y on the entirety of $[0, T]$, the signal must be smooth, structured, or otherwise "simple" in some way.

Across science and engineering, by far one of the most common ways in which structure arises is through various assumptions about $\hat{y}$, the *Fourier transform* of y:

$$\hat{y}(\xi) = \int_{-\infty}^{\infty} y(t) e^{-2\pi i t \xi} \, dt.$$

Our goal is to understand signal reconstruction under natural constraints on the complexity of $\hat{y}$.

3.1.1 Classical sampling theory and bandlimited signals

Classically, the most standard example of such a constraint is requiring y to be *bandlimited*, meaning that $\hat{y}$ is only non-zero for frequencies ξ with $|\xi| \leq F$ for some bandlimit F. In this case, we recall the famous sampling theory of Nyquist, Shannon, and others (Whitaker, 1915; Kotelnikov, 1933; Nyquist, 1928; Shannon, 1949). This theory shows that y can be reconstructed exactly using sinc interpolation (i.e, Whittaker-Shannon interpolation) if $1/2F$ uniformly spaced samples of y are taken per unit of time (the *Nyquist rate*).

Unfortunately, this theory is asymptotic: it requires infinite samples over the entire real line to interpolate y, even at a single point. When a finite number of samples are taken over an interval $[0, T]$, sinc interpolation is not a good reconstruction method, either in theory or in practice (Xiao, 2002).[1]

This well-known issue was resolved through a seminal line of work by Slepian and Pollak (1961); Landau and Pollak (1961, 1962), who presented a set of explicit basis functions for interpolating bandlimited functions when a finite number of samples are taken from a finite interval. Their so-called "prolate spheroidal wave functions" can be combined with numerical quadrature methods of Rokhlin et al. (2001) to obtain sample efficient (and computationally efficient) methods for bandlimited reconstruction. Overall, this work shows that roughly $O\big(FT + \log(1/\epsilon)\big)$ samples from $[0, T]$ are required to interpolate a signal with bandlimit F to accuracy ϵ on that same interval.[2]

[1] Approximation bounds can be obtained by truncating the Whittaker-Shannon method; however, they are weak, depending *polynomially*, rather than *logarithmically*, on the desired error ϵ (see Appendix A of Avron et al. (2019) for an in depth overview of related work.

[2] We formalize our notion of accuracy in Section 3.2.

3.1.2 More general Fourier structure

While the aforementioned line of work is beautiful and powerful, in today's world, we are interested in far more general constraints than bandlimits. For example, there is wide-spread interest in *Fourier-sparse* signals (Donoho, 2006), where $\hat{y}$ is only non-zero for a small number of frequencies, and *multiband* (Block-sparse) signals, where the Fourier transform is confined to a small number of intervals. Methods for recovering signals in these classes have countless applications in communication, imaging, statistics, and a wide variety of other disciplines (Eldar, 2015).

More generally, in statistical signal processing, a *prior distribution*, specified by some probability measure μ, is often assumed on the frequency content of y (Eldar and Unser, 2006; Ramani et al., 2005). For signals with bandlimit F, μ would be the uniform probability measure on $[-F, F]$. Alternatively, instead of assuming a hard bandlimit, a zero-centered Gaussian prior on $\hat{y}$ can encode knowledge that higher frequencies are less likely to contribute significantly to y, although they may still be present. Such a prior naturally suits a Bayesian approach to signal reconstruction (Handcock and Stein, 1993) and, in fact, is essentially equivalent to assuming y is a stationary stochastic process with a certain covariance function (Avron et al., 2019). Under various names, including "Gaussian process regression" and "kriging," likelihood estimation under a covariance prior is the dominant statistical approach to fitting continuous signals in many scientific disciplines, from geostatistics to economics to medical imaging (Ripley, 2005; Rasmussen, 2003).

3.1.3 Our contributions

Despite their clear importance, accurate methods for fitting continuous signals under most common Fourier transform priors are not well understood, even 50 years after the groundbreaking work of Slepian, Landau, and Pollak on the bandlimited problem. The only exception is Fourier sparse signals: the *noiseless* interpolation problem can be solved using classical methods (de Prony, 1795; Pisarenko, 1973; Bresler and Macovski, 1986), and recent work has resolved the much more difficult noisy case (Chen et al., 2016; Chen and Price, 2019).

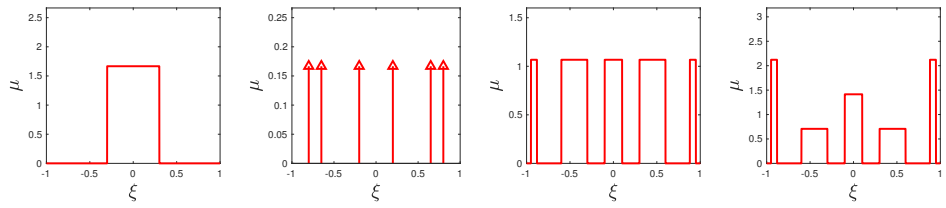
In this chapter, we address the problem far more generally. Our contributions are as follows:

1. We tightly characterize the information theoretic sample complexity of reconstructing y under any Fourier transform prior, specified by probability measure μ. In essentially all settings, we can prove that this complexity scales nearly linearly with a natural *statistical dimension* parameter associated with μ. See Theorem 3.2.1.

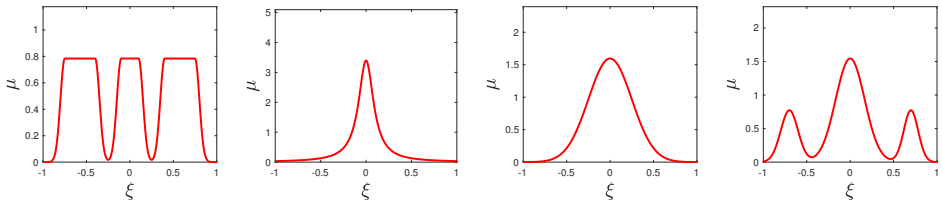
2. We present a method for sampling from y that achieves the aforementioned statistical dimension bound to within a polylogarithmic factor. Our approach is randomized and *universal*: we prove that it is possible to draw $t_1, \ldots, t_q$ from a fixed non-uniform distribution over $[0, T]$ that is *independent of* μ, i.e., "spectrum-blind." In other words,

the same sampling scheme works for bandlimited, sparse, or more general priors. See Theorem 3.2.2.

3. We show that y can be recovered from $t_1, \ldots, t_q$ using a simple, efficient, and completely general interpolation method. In particular, we just need to solve a kernel ridge regression problem using $y(t_1), \ldots, y(t_q)$, with an appropriately chosen kernel function for μ. This method runs in $O(q^3)$ time and is already widely used for signal reconstruction in practice, albeit with sub-optimal strategies for choosing $t_1, \ldots, t_q$. See Theorem 3.2.3.



(a) Bandlimited. (b) Sparse. (c) Multiband. (d) Non-uniform multiband.



(e) Smooth multiband. (f) Cauchy-Lorentz. (g) Gaussian. (h) Gaussian mixture.

Figure 3.2 – Examples of Fourier transform "priors" induced by various measures μ (we plot the corresponding density). Our algorithm can reconstruct signals under any of these priors.

Overall, this approach gives the first finite sample, provable approximation bounds for all common Fourier-constrained signal reconstruction problems beyond bandlimited and sparse functions.

Our results are obtained by drawing on a rich set of tools from randomized numerical linear algebra, including sampling methods for approximate regression and deterministic column-based low-rank approximation methods (Batson et al., 2009; Cohen et al., 2016a). Many of these methods view matrices as sums of rank-1 outer products and approximate them by sampling or deterministically selecting a subset of these outer products. We adapt these tools to the approximation of continuous operators, which can be written as the (weak) integral of rank-1 operators. For example, our universal time domain sampling distribution is obtained using the notion of *statistical leverage* (Spielman and Srivastava, 2008; Alaoui and Mahoney, 2015; Drineas and Mahoney, 2016), extended to a continuous Fourier transform operator that arises in the signal reconstruction problem. We hope that, by extending many of the fundamental contributions of randomized numerical linear algebra to build a toolkit for 'randomized operator theory', our work will offer a starting point for progress on many signal processing problems using randomized methods.

3.2 Formal Statement of Results

As suggested, we formally capture Fourier structure through any probability measure μ over the reals.[3] We often refer to μ as a "prior", although we will see that it can be understood beyond the context of Bayesian inference. The simplicity of a set of constraints will be quantified by a natural *statistical dimension* parameter for μ, defined in Section 3.2.1.

For signals with bandlimit F, μ is the uniform probability measure on $[-F, F]$. For multiband signals, it is uniform on the union of k intervals, while for Fourier-sparse functions, μ is uniform on a union of k Dirac measures. More general priors are visualized in Figure 3.2. Those based on Gaussian or Cauchy-Lorentz distributions are especially common in scientific applications, and we will discuss examples shortly. For now, we begin with our main problem formulation.

Problem 3.2.1. Given a known probability measure μ on $\mathbb{R}$, for any $t \in [0, T]$, define the inverse Fourier transform of a function $g(\xi)$ with respect to μ as,

$$\left[\mathscr{F}_\mu^* g\right](t) \stackrel{\text{def}}{=} \int_{\mathbb{R}} g(\xi) e^{2\pi i \xi t} \, d\mu(\xi). \tag{3.1}$$

Suppose our input y can be written as $y = \mathscr{F}_\mu^* x$ for some frequency domain function $x(\xi)$ and, for any chosen t, we can observe $y(t) + n(t)$ for some fixed noise function $n(t)$. Then, for error parameter ϵ, our goal is to recover an approximation $\tilde{y}$ satisfying,

$$\|y - \tilde{y}\|_T^2 \leq \epsilon \|x\|_\mu^2 + C\|n\|_T^2, \tag{3.2}$$

where $\|x\|_\mu^2 \stackrel{\text{def}}{=} \int_{\mathbb{R}} |x(\xi)|^2 \, d\mu(\xi)$ is the energy of the function x with respect to μ, while $\|z\|_T^2 \stackrel{\text{def}}{=} \frac{1}{T}\int_0^T |z(t)|^2 dt$, so that $\|y - \tilde{y}\|_T^2$ is our mean squared error and $\|n\|_T^2$ is the mean squared noise level. $C \geq 1$ is a fixed positive constant.

Unlike the $\|x\|_\mu^2$ term in (3.2), which we can control by adjusting ϵ, we can never hope to recover y to accuracy better than $\|n\|_T^2$. Accordingly, we consider $\|n\|_T^2$ to be small and are happy with any solution of Problem 3.2.1 that is within a constant factor of optimal – i.e., where $C = O(1)$.

Problem 3.2.1 captures signal reconstruction under all standard Fourier transform constraints, including bandlimited, multiband, and sparse signals.[4] The error in (3.2) naturally scales with the average energy of the signal over the allowed frequencies. For more general priors, $\|x\|_\mu^2$ will be larger when y contains a significant component of frequencies with low density in μ.[5]

[3] Formally, we consider the measure space $(\mathbb{R}, \mathscr{B}, \mu)$ where $\mathscr{B}$ is the Borel σ-algebra on $\mathbb{R}$.

[4] For sparse or multiband signals, Problem 3.2.1 assumes frequency or band locations are known *a priori*. There has been significant work on algorithms that can recover y when these locations are not known (Mishali and Eldar, 2009; Moitra, 2015; Price and Song, 2015; Chen et al., 2016). Understanding this more complicated problem in the multiband case is an important future direction.

[5] Informally, decreasing $d\mu(\xi)$ by a factor of $c > 1$ requires increasing $x(\xi)$ by a factor of c to give the same time domain signal. This increases $x(\xi)^2$ by a factor of c^2 and so increases its contribution to $\|x\|_\mu^2$ by a factor of $c^2/c = c$.

For a given number of samples, we would thus incur larger error in (3.2) in comparison to a signal that uses more "likely" frequencies.

As an alternative to Problem 3.2.1, we can formulate signal fitting from a Bayesian perspective. We assume that n is independent random noise, and y is a stationary stochastic process with expected power spectral density μ. This assumption on y's power spectral density is equivalent to assuming that y has covariance function (a.k.a. autocorrelation) $\widehat{\mu}(t)$, which is the type of prior used in kriging and Gaussian process regression.

Examples and applications

As discussed in Section 3.1.2, "hard constraint" versions of Problem 3.2.1, such as bandlimited, sparse, and multiband signal reconstruction, have many applications in communications, imaging, audio, and other areas of engineering. Generalizations of the multiband problem to non-uniform measures (see Figure 3.2d) are also useful in various communication problems (Mishali and Eldar, 2010).

On the other hand, "soft constraint" versions of the problem are widely applied in scientific applications. In medical imaging, images are often denoised by setting μ to a heavy-tailed Cauchy-Lorentz measure on frequencies (Fuderer, 1989; Lettington and Hong, 1995). This corresponds to assuming an exponential covariance function for spatial correlation. Exponential covariance and its generalization, Matérn covariance, are also common in the earth and geosciences (Ripley, 1991, 2005), as well as in general image processing (Pesquet-Popescu and Véhel, 2002).

A Gaussian prior μ, which corresponds to Gaussian covariance, is also used to model both spatial and temporal correlation in medical imaging (Friston et al., 1994; Worsley et al., 1996) and is very common in machine learning. Other choices for μ are practically unlimited. For example, the popular ArcGIS kriging library also supports the following covariance functions: circular, spherical, tetraspherical, pentaspherical, rational quadratic, hole effect, k-bessel, and j-bessel, and stable (ESRI, 2001).

3.2.1 Sample complexity

With Problem 3.2.1 defined, our first goal is to characterize the number of samples required to reconstruct y, as a function of the *accuracy parameter* ϵ, the *range* T, and the *measure* μ. We do so using what we refer to as the *Fourier statistical dimension* of μ, which corresponds to the standard notion of statistical or 'effective dimension' for regularized function fitting problems (Hastie et al., 2009; Zhang, 2005).

Definition 3.2.2 (Fourier statistical dimension)**.** For a probability measure μ on $\mathbb{R}$ and time

length T, define the kernel operator $\mathcal{K}_\mu : L_2(T) \to L_2(T)$[6] as:

$$[\mathcal{K}_\mu z](t) \stackrel{\text{def}}{=} \int_{\xi \in \mathbb{R}} e^{2\pi i \xi t} \left[\frac{1}{T} \int_{s \in [0,T]} z(s) e^{-2\pi i \xi s} \, ds \right] d\mu(\xi). \tag{3.3}$$

Note that $\mathcal{K}_\mu$ is self-adjoint, positive semidefinite and trace-class.[7] The Fourier statistical dimension for μ, T, and error ϵ is denoted by $s_{\mu,\epsilon}$ and defined as:

$$s_{\mu,\epsilon} \stackrel{\text{def}}{=} \operatorname{tr}\left(\mathcal{K}_\mu (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1}\right), \tag{3.4}$$

where $\mathcal{I}_T$ is the identity operator on $L_2(T)$. Letting $\lambda_i(\mathcal{K}_\mu)$ denote the i^{th} largest eigenvalue of $\mathcal{K}_\mu$, we may also write,

$$s_{\mu,\epsilon} = \sum_{i=1}^{\infty} \frac{\lambda_i\left(\mathcal{K}_\mu\right)}{\lambda_i\left(\mathcal{K}_\mu\right) + \epsilon}. \tag{3.5}$$

Note that $\mathcal{K}_\mu$ and $s_{\mu,\epsilon}$ as defined above, depend on T and thus could naturally be denoted $\mathcal{K}_{\mu,T}$ and $s_{\mu,\epsilon,T}$. However, since T is fixed throughout our results, for conciseness we do not use T in our notation for these and related notions.

It is not hard to see that $s_{\mu,\epsilon}$ increases as ϵ decreases, meaning that we will require more samples to obtain a more accurate solution to Problem 3.2.1. The operator $\mathcal{K}_\mu$ corresponds to taking the Fourier transform of a time domain input $z(t)$, scaling that transform by μ, and then taking the inverse Fourier transform. Readers familiar with the literature on bandlimited signal reconstruction will recognize $\mathcal{K}_\mu$ as the natural generalization of the frequency limiting operator studied in the celebrated work of Slepian and Pollak (1961); Landau and Pollak (1961, 1962) on prolate spheroidal wave functions. In that work, it is established that a quantity nearly identical to $s_{\mu,\epsilon}$ bounds the sample complexity of solving Problem 3.2.1 for bandlimited functions.

Our first technical result is that this is actually true *for any prior* μ.

Theorem 3.2.1 (Main result, sample complexity)**.** *For any probability measure μ, Problem 3.2.1 can be solved using $q = O\left(s_{\mu,\epsilon} \cdot \log s_{\mu,\epsilon}\right)$ noisy signal samples $y(t_1) + n(t_1), \ldots, y(t_q) + n(t_q)$.*

What does Theorem 3.2.1 imply for common classes of functions with constrained Fourier transforms? Table 3.1 includes a list of upper bounds on $s_{\mu,\epsilon}$ for many standard priors.

A complexity of $O(s_{\mu,\epsilon} \cdot \log s_{\mu,\epsilon})$ equates to $\tilde{O}(k)$ samples for Fourier k-sparse functions and $\tilde{O}(FT + \log 1/\epsilon)$ for bandlimited functions. Up to log factors, these bounds are tight for these well studied problems. In Section 3.6, we show that Theorem 3.2.1 is actually tight for all common Fourier transform priors: $\Omega(s_{\mu,\epsilon})$ time points are required for solving Problem 3.2.1

[6] $L_2(T)$ denotes the complex-valued square integrable functions with respect to the uniform measure on $[0, T]$.
[7] See Section 3.3 for a formal explanation of these facts.

Fourier prior, μ	Statistical dimension, $s_{\mu,\epsilon}$	Proof
k-sparse	k	Since $\mathcal{K}_\mu$ has rank k.
bandlimited to $[-F,F]$	$O\big(FT+\log(1/\epsilon)\big)$	Theorem B.3.1.
multiband, widths $F_1,\ldots,F_s$	$O\big(\sum_i F_i T + s\log(1/\epsilon)\big)$	Theorem B.4.1.[8]
Gaussian, variance F	$O\big(FT\sqrt{\log(1/\epsilon)}+\log(1/\epsilon)\big)$	Theorem B.4.2.
Cauchy-Lorentz, scale F	$O\big(FT\sqrt{1/\epsilon}+\sqrt{1/\epsilon}\big)$	Theorem B.4.3.

Table 3.1 – Statistical dimension upper bounds for common Fourier interpolation problems. Our result (Theorem 3.2.1) requires $O(s_{\mu,\epsilon}\cdot\log s_{\mu,\epsilon})$ samples.

as long as $s_{\mu,\epsilon}$ grows slower than $1/\epsilon^p$ for some $p<1$. This property holds for all Fourier priors μ listed in Table 3.1.

To compliment the sample complexity bound of Theorem 3.2.1, we introduce a *universal method* for selecting samples $t_1,\ldots,t_q$ that nearly matches this complexity. Our method selects samples at random, in a way that *does not depend* on the specific prior μ.

Theorem 3.2.2 (Main result, sampling distribution)**.** *For any sample size q, there is a fixed probability density p_q over $[0,T]$ such that, if q time points $t_1,\ldots,t_q$ are selected independently at random according to p_q, and $q\geq c\cdot s_{\mu,\epsilon}\cdot\log^2 s_{\mu,\epsilon}$ for some fixed constant c, then it is possible to solve Problem 3.2.1 with probability 99/100 using the noisy signal samples $y(t_1)+n(t_1),\ldots,y(t_q)+n(t_q)$.*[9]

Theorem 3.2.2 is our main technical contribution. By achieving near-optimal sample complexity with a universal distribution, it shows that wide range of Fourier constrained interpolation problems considered in the literature are more closely related than previously understood.

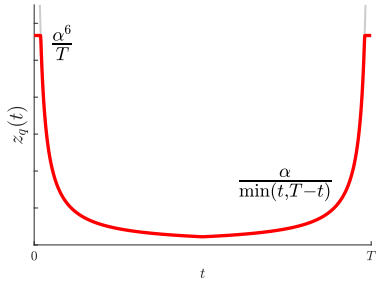
Moreover, p_q (which is formally defined in Theorem 3.5.6) is very simple to describe and sample from. As may be intuitive from results on polynomial interpolation, bandlimited approximation, and other function fitting problems, it is more concentrated towards the endpoints of $[0,T]$, so our sampling scheme selects more time points in these regions. The density is shown in Figure 3.3.

3.2.2 Algorithmic complexity

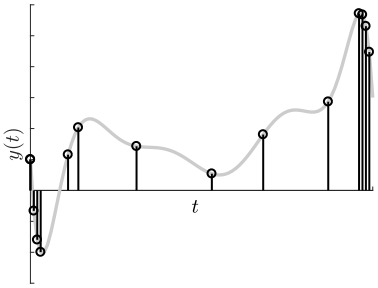
While Theorem 3.2.2 immediately yields an approach for selecting samples $t_1,\ldots,t_q$, it is only useful if we can *efficiently* solve Problem 3.2.1 given the noisy measurements $y(t_1)+n(t_1),\ldots,y(t_q)+n(t_q)$. We show that this is possible for a broad class of constraint measures.

[8] Just as Theorem B.3.1 intuitively matches the Nyquist sampling rate, Theorem B.4.1 intuitively matches the Landau rate for asymptotic recovery of multiband functions (Landau, 1967).

[9] In Section 3.5.4, we formally quantify the tradeoff between success probability and sample complexity.



(a) Density for selecting time points.



(b) Example set of nodes sampled according to p_q.

Figure 3.3 – A visualization of the universal sampling distribution, p_q, which can be used for reconstructing a signal under any Fourier prior μ. To obtain p_q for a given number of samples q, choose α so that $q = \Theta(\alpha \log^2 \alpha)$. Set $z_q(t)$ equal to $\alpha / \min(t, T-t)$, except near 0 and T, where the function is capped at $z_q(t) = \alpha^6$. Construct p_q by normalizing z_q to integrate to 1.

Specifically, we only need to efficiently compute the positive-definite kernel function[10]:

$$k_\mu(t_1, t_2) = \int_{\xi \in \mathbb{R}} e^{-2\pi i (t_1 - t_2)\xi} d\mu(\xi). \tag{3.6}$$

The above integral can be approximated via numerical quadrature, but for many of the aforementioned applications, it has a closed-form. For example, when μ is supported on just k frequencies, it is a sum of these frequencies. When μ is uniform on $[-F, F]$, $k_\mu(t_1, t_2) = \text{sinc}(2\pi F(t_1 - t_2))$. For multiband signals with s bands, $k_\mu(t_1, t_2)$ is a sum of s modulated sinc functions. In fact, $k_\mu(t_1, t_2)$ has a closed-form for all μ illustrated in Figure 3.2. Further details are discussed in Appendix B.5. Assuming a subroutine for computing $k_\mu(t_1, t_2)$, our main algorithmic result is as follows:

Theorem 3.2.3. *(Main result, algorithmic complexity) There is an algorithm that solves Problem 3.2.1 with probability* 99/100 *which uses* $O\left(s_{\mu,\epsilon} \cdot \log^2(s_{\mu,\epsilon})\right)$ *time domain samples (sampled according to the distribution given by Theorem 3.2.2) and runs in* $\tilde{O}\left(s^\omega_{\mu,\epsilon} + s^2_{\mu,\epsilon} \cdot Z\right)$ *time, assuming the ability to compute* $k_\mu(t_1, t_2)$ *for any* $t_1, t_2 \in [0, T]$ *in* Z *time.*[11] *The algorithm returns a representation of* $\tilde{y}(t)$ *that can be evaluated in* $\tilde{O}(s_{\mu,\epsilon} \cdot Z)$ *time for any* t.

For bandlimited, Gaussian, or Cauchy-Lorentz priors μ, $Z = O(1)$. For s sparse signals or multiband signals with s blocks, $Z = O(s)$.

We note that, while Theorem 3.2.3 holds when $\tilde{O}\left(s_{\mu,\epsilon}\right)$ samples are taken, $s_{\mu,\epsilon}$ may not be known and thus it may be unclear how to set the sample size. In our full statement of the Theorem in Section 3.5.4 we make it clear that any upper bound on $s_{\mu,\epsilon}$ suffices to set the

[10] When y is real valued, it makes sense to consider symmetric μ. In this case, k_μ is also real valued. However, in general it may be complex valued.

[11] For conciseness, we use $\tilde{O}(z)$ to denote $\tilde{O}(z \log^c z)$, where c is some fixed constant (usually ≤ 2). In formal theorem statements we give c explicitly. $\omega < 2.373$ is the current exponent of fast matrix multiplication (Williams, 2012).

sample size. The sample complexity will depend on how tight this upper bound is. In Appendix B.4 we give upper bounds on $s_{\mu,\epsilon}$ for a number of common μ, which can be plugged into Theorem 3.2.3.

3.2.3 Our approach

Theorems 3.2.1, 3.2.2, and 3.2.3 are achieved through a simple and practical algorithmic framework. In Section 3.4, we show that Problem 3.2.1 can be modeled as a least squares regression problem with ℓ_2 regularization. As long as we can compute $k_\mu(t_1, t_2)$, we can solve this problem using *kernel ridge regression*, a popular function fitting technique in nonparametric statistics (Shawe-Taylor et al., 2004).

Naively, the kernel regression problem is infinite dimensional: it needs to be solved over the *continuous* time domain $[0, T]$ to solve our signal reconstruction problem. This is where sampling comes in. We need to discretize the problem and establish that our solution over a fixed set of time samples nearly matches the solution over the continuous interval. To bound the error of discretization, we turn to a tool from randomized numerical linear algebra: *statistical leverage score sampling* (Spielman and Srivastava, 2008; Drineas and Mahoney, 2016). We show how to *randomly* discretize Problem 3.2.1 by sampling time points with probability proportional to an appropriately defined non-uniform leverage score distribution on $[0, T]$. The required number of samples is $O(s_{\mu,\epsilon} \log s_{\mu,\epsilon})$, which proves Theorem 3.2.1.

Unfortunately, the leverage score distribution does not have a closed-form, varies depending on ϵ, T, and μ, and likely cannot be sampled from exactly. To prove Theorem 3.2.2, we show that for any μ, for large enough q, the closed form distribution p_q *upper bounds* the leverage score distribution. This upper bound closely approximates the true leverage score distribution and, therefore, can be used in its place during sampling, losing only a $\log s_{\mu,\epsilon}$ factor in the sample complexity.

The leverage score distribution roughly measures, for each time point t, how large $|y(t)|^2$ can be compared to $\|y\|_T^2$ when y's Fourier transform is constrained by μ (i.e., when $\|x\|_\mu^2$ as defined in Problem 3.2.1 is bounded). To upper bound this measure we turn to another powerful result from the randomized numerical linear algebra literature: every matrix contains a small subset of columns that span a near-optimal low-rank approximation to that matrix (Sarlos, 2006; Boutsidis et al., 2009; Deshpande and Rademacher, 2010). In other words, every matrix admits a near-optimal low-rank approximation with *sparse column support*. By extending this result to continuous linear operators, we prove that the smoothness of a signal whose Fourier transform has $\|x\|_\mu^2$ bounded is tightly captured by the smoothness of an $O(s_{\mu,\epsilon})$-sparse Fourier function. This lets us reduce every Fourier prior to Fourier sparsity and apply recent results of (Chen et al., 2016; Chen and Price, 2019) that bound $|y(t)|^2$ in terms of $\|y\|_T^2$ for any Fourier sparse function y . Intuitively, our result shows that the simplicity of sparse Fourier functions governs the simplicity of *any class* of Fourier constrained functions.

The above argument yields Theorem 3.2.2. Since we can sample from p_q in $O(1)$ time, we can efficiently sample the time domain to $O\left(s_{\mu,\epsilon} \cdot \log^2 s_{\mu,\epsilon}\right)$ points and then solve Problem 3.2.1 by applying kernel ridge regression to these points, which takes $\tilde{O}\left(s_{\mu,\epsilon}^{\omega} + s_{\mu,\epsilon}^2 \cdot Z\right)$ time, assuming the ability to compute $k_\mu(\cdot,\cdot)$ in Z time. This yields the algorithmic result of Theorem 3.2.3.

3.2.4 Roadmap

The rest of this chapter is structured as follows:

Section 3.3 We lay out basic notation that is used throughout the paper.

Section 3.4 We reduce Problem 3.2.1 to a kernel ridge regression problem and explain how to randomly discretize and solve this problem via leverage score sampling, proving Theorem 3.2.1.

Section 3.5 We give an upper bound on the leverage score distribution for general priors, proving Theorems 3.2.2 and 3.2.3.

Section 3.6 We prove that, under a mild assumption, the statistical dimension tightly characterizes the sample complexity of solving Problem 3.2.1, and thus that our results are nearly optimal.

In Appendix B.1 we give operator theory preliminaries. In Appendix B.2 we prove our extensions of a number of randomized linear algebra primitives to continuous operators. In Appendix B.3, we bound the statistical dimension for the important case of bandlimited functions. We use this result in Appendix B.4 to prove statistical dimension bounds for multiband, Gaussian, and Cauchy-Lorentz priors (shown in Table 3.1). In Appendix B.5, we show how to compute the kernel function k_μ for these common priors.

3.3 Notation

Let μ be a probability measure on $(\mathbb{R}, \mathscr{B})$, where $\mathscr{B}$ is the Borel σ-algebra on $\mathbb{R}$. Let $L_2(\mu)$ denote the space of complex-valued square integrable functions with respect to μ. For $a, b \in L_2(\mu)$, let $\langle a, b\rangle_\mu$ denote $\int_{\xi\in\mathbb{R}} a(\xi)^* b(\xi)\, d\mu(\xi)$ where for any $x \in \mathbb{C}$, x^* is its complex conjugate. Let $\|a\|_\mu^2$ denote $\langle a, a\rangle_\mu$. Let $\mathscr{I}_\mu$ denote the identity operator on $L_2(\mu)$. Note that for any μ, $L_2(\mu)$ is a separable Hilbert space and thus has a countably infinite orthonormal basis (Hunter and Nachtergaele, 2001).

We overload notation and use $L_2(T)$ to denote the space of complex-valued square integrable functions with respect to the uniform probability measure on $[0, T]$. It will be clear from context that T is not a measure. For $a, b \in L_2(T)$, let $\langle a, b\rangle_T$ denote $\frac{1}{T}\int_0^T a(t)^* b(t)\, dt$ and let $\|a\|_T^2$ denote $\langle a, a\rangle_T$. Let $\mathscr{I}_T$ denote the identity operator on $L_2(T)$.

Define the Fourier transform operator $\mathscr{F}_\mu : L_2(T) \to L_2(\mu)$ as:

$$\left[\mathscr{F}_\mu f\right](\xi) = \frac{1}{T}\int_0^T f(t)e^{-2\pi i t\xi}\,dt. \tag{3.7}$$

The adjoint of $\mathscr{F}_\mu$ is the unique operator $\mathscr{F}_\mu^* : L_2(\mu) \to L_2(T)$ such that for all $f \in L_2(T), g \in L_2(\mu)$ we have $\langle g, \mathscr{F}_\mu f\rangle_\mu = \langle \mathscr{F}_\mu^* g, f\rangle_T$. It is not hard to see that $\mathscr{F}_\mu^*$ is the inverse Fourier transform operator with respect to μ as defined in Section 3.2, equation (3.1):

$$\left[\mathscr{F}_\mu^* g\right](t) \stackrel{\text{def}}{=} \int_{\mathbb{R}} g(\xi)e^{2\pi i\xi t}\,d\mu(\xi). \tag{3.8}$$

Note that the kernel operator $\mathscr{K}_\mu : L_2(T) \to L_2(T)$ originally defined in (3.3) is equal to

$$\mathscr{K}_\mu = \mathscr{F}_\mu^*\mathscr{F}_\mu.$$

$\mathscr{K}_\mu$ is self-adjoint, positive semidefinite and trace-class and an integral operator with kernel k_μ:

$$[\mathscr{K}_\mu z](t) = \frac{1}{T}\int_0^T k_\mu(s,t)z(s)\,ds,$$

where k_μ is as defined in (3.6). The trace of $\mathscr{K}_\mu$ is equal to 1.[12] We will also make use of the Gram operator: $\mathscr{G}_\mu \stackrel{\text{def}}{=} \mathscr{F}_\mu\mathscr{F}_\mu^*$. $\mathscr{G}_\mu$ is also self-adjoint, positive semidefinite, and trace-class.

Remark: It may be useful for the reader to informally regard $\mathscr{F}_\mu$ as an infinite matrix with rows indexed by $\xi \in \mathbb{R}$ and columns indexed by $t \in [0, T]$. Following the definition of $\mathscr{F}_\mu$ above, and assuming that μ has a density p, this infinite matrix has entries given by:

$$\mathscr{F}_\mu(\xi, t) = \sqrt{\frac{p(\xi)}{T}} \cdot e^{-2\pi i t\xi}. \tag{3.9}$$

The results we apply on leverage score sampling can all be seen as extending results for finite matrices from the randomized numerical linear algebra literature to this infinite matrix.

3.4 Function Fitting with Least Squares Regression

Least squares regression provides a natural approach to solving the interpolation task of Problem 3.2.1. In particular, consider the following regularized minimization problem over

[12] Since the kernel is a Fourier transform of a probability measure, it is Hermitian positive definite (Bochner's Theorem). Then we can conclude that $\mathscr{K}_\mu$ is trace-class from Mercer's theorem, and calculate $\mathrm{tr}(\mathscr{K}_\mu) = \frac{1}{T}\int_0^T k_\mu(t,t)\,dt = 1$.

functions $g \in L_2(\mu)$[13]:

$$\min_{g \in L_2(\mu)} \|\mathscr{F}_\mu^* g - (y+n)\|_T^2 + \epsilon \|g\|_\mu^2. \tag{3.10}$$

The first term encourages us to find a function g whose inverse Fourier transform is close to our measured signal $y+n$. The second term encourages us to find a low energy solution – ultimately, we solve (3.10) based on only a small number of samples $y(t_1), \ldots, y(t_k)$, and smoother, lower energy solutions will better generalize to the entire interval $[0, T]$. We remark that it is well known that least squares approximations benefit from regularization even in the noiseless case (Cohen et al., 2013).

We first state a straightforward fact: if we minimize (3.10), even to a coarse approximation, then we are able to solve Problem 3.2.1.

Claim 3.4.1. *Let $y = \mathscr{F}_\mu^* x$, $n \in L_2(T)$ be an arbitrary noise function, and for any $C \geq 1$, let $\tilde{g} \in L_2(\mu)$ be a function satisfying:*

$$\|\mathscr{F}_\mu^* \tilde{g} - (y+n)\|_T^2 + \epsilon \|\tilde{g}\|_\mu^2 \leq C \cdot \min_{g \in L_2(\mu)} \left[\|\mathscr{F}_\mu^* g - (y+n)\|_T^2 + \epsilon \|g\|_\mu^2 \right].$$

Then

$$\|\mathscr{F}_\mu^* \tilde{g} - y\|_T^2 \leq 2C\epsilon \|x\|_\mu^2 + 2(C+1)\|n\|_T^2.$$

Proof. Since $y = \mathscr{F}_\mu^* x$, $\min_{g \in L_2(\mu)} \left[\|\mathscr{F}_\mu^* g - (y+n)\|_T^2 + \epsilon \|g\|_\mu^2 \right] \leq \|n\|_T^2 + \epsilon \|x\|_\mu^2$. Thus, $\|\mathscr{F}_\mu^* \tilde{g} - (y+n)\|_T^2 \leq C\epsilon \|x\|_\mu^2 + C\|n\|_T^2$. The claim then follows via triangle inequality:

$$\begin{aligned} \|\mathscr{F}_\mu^* \tilde{g} - y\|_T - \|n\|_T &\leq \|\mathscr{F}_\mu^* \tilde{g} - (y+n)\|_T \\ \|\mathscr{F}_\mu^* \tilde{g} - y\|_T &\leq \sqrt{C\epsilon \|x\|_\mu^2 + C\|n\|_T^2} + \|n\|_T \\ \|\mathscr{F}_\mu^* \tilde{g} - y\|_T^2 &\leq 2C\epsilon \|x\|_\mu^2 + 2(C+1)\|n\|_T^2. \end{aligned}$$

□

Claim 3.4.1 shows that approximately solving the regression problem in (3.10), with regularization parameter ϵ gives a solution to Problem 3.2.1 with parameter $2C\epsilon$ (decreasing the regularization parameter to $\frac{\epsilon}{2C}$ will let us solve with parameter ϵ). But how can we solve the regression problem efficiently? Not only does the problem involve a possibly infinite dimensional parameter vector g, but the objective function also involves the continuous time interval $[0, T]$.

[13] The fact that the minimum is attainable is a simple consequence of the extreme value theorem, since the search space can be restricted to $\|g\|_\mu^2 \leq \|(y+n)\|_T^2/\epsilon$.

3.4.1 Random discretization via leverage function sampling

The first step is to deal with the latter challenge, i.e., that of a continuous time domain. We show that it is possible to *randomly discretize* the time domain of (3.10), thereby reducing our problem to a regression problem on a finite set of times $t_1, \ldots, t_q$. In particular, we can sample time points with probability proportional to the so-called *ridge leverage function*, a specific non-uniform distribution that has been applied widely in randomized algorithms for regression and other linear algebra problems on discrete matrices (Alaoui and Mahoney, 2015; Cohen et al., 2017; Musco and Musco, 2017; Musco and Woodruff, 2017).

While we cannot compute the leverage function explicitly for our problem, an issue highlighted by Bach (2017), our main result (Theorem 3.2.2) uses a simple, but very accurate, closed form approximation in its place. We start with the definition of the ridge leverage function:

Definition 3.4.1 (Ridge leverage function)**.** For a probability measure μ on $\mathbb{R}$, time length $T > 0$, and $\epsilon \geq 0$, we define the ϵ-ridge leverage function for $t \in [0, T]$ as[14]:

$$\tau_{\mu,\epsilon}(t) = \frac{1}{T} \cdot \max_{\{\alpha \in L_2(\mu):\, \|\alpha\|_\mu > 0\}} \frac{\left| [\mathscr{F}^*_\mu \alpha](t) \right|^2}{\left\| \mathscr{F}^*_\mu \alpha \right\|^2_T + \epsilon \|\alpha\|^2_\mu}. \tag{3.11}$$

Intuitively, the ridge leverage function at time t is an upper bound of how much a function can "blow up" at t when its Fourier transform is constrained by μ. The denominator term $\left\| \mathscr{F}^*_\mu \alpha \right\|^2_T$ is the average squared magnitude of the function $F^*_\mu \alpha$, while the numerator term, $|[\mathscr{F}^*_\mu \alpha](t)|^2$, is the squared magnitude at t. The regularization term $\epsilon \|\alpha\|^2_\mu$ reflects the fact that, to solve (3.10), we only need to bound the smoothness for functions with bounded Fourier energy under μ. As observed in (Pauwels et al., 2018), the ridge leverage function can be viewed as a type of *Christoffel function*, studied in the literature on orthogonal polynomials and approximation theory (Pauwels et al., 2018; Totik, 2000; Borwein and Erdélyi, 1995).

The larger the leverage "score" $\tau_{\mu,\epsilon}(t)$, the higher the probability we will sample time t, to ensure that our sample points well reflect any possibly significant components or 'spikes' of the function y. Ultimately, the integral of the ridge leverage function $\int_0^T \tau_{\mu,\epsilon}(t)\,dt$ determines how many samples we require to solve (3.10) to a given accuracy. Theorem 3.4.1 below states the already known fact that the ridge leverage function integrates to the statistical dimension (Avron et al., 2017c), which will ultimately allow us to achieve the $\tilde{O}(s_{\mu,\epsilon})$ sample complexity bound of Theorems 3.2.1 and 3.2.2. Theorem 3.4.1 also gives two alternative characterizations of the leverage function that will prove useful. The theorem is proven in Appendix B.2, using techniques for finite matrices, adapted to the operator setting.

[14] Formally $L_2(T)$ is a space of equivalence classes of functions that differ at a set of points with measure 0. For notational simplicity, here and throughout we use $\mathscr{F}^*_\mu \alpha$ to denote the specific representative of the equivalence class $\mathscr{F}^*_\mu \alpha \in L_2(T)$ given by (3.8). In this way, we can consider the pointwise value $[\mathscr{F}^*_\mu \alpha](t)$, which we could alternatively express as $\langle \varphi_t, \alpha \rangle_\mu$, for $\varphi_t(\xi) \stackrel{\text{def}}{=} e^{-2\pi i t \xi}$.

Theorem 3.4.1 (Leverage function properties)**.** *Let $\tau_{\mu,\epsilon}(t)$ be the ridge leverage function (Definition 3.4.1) and define $\varphi_t \in L_2(\mu)$ by $\varphi_t(\xi) \stackrel{\text{def}}{=} e^{-2\pi i t\xi}$. We have:*

- *The ridge leverage function integrates to the statistical dimension:*

$$\int_0^T \tau_{\mu,\epsilon}(t)\,dt = s_{\mu,\epsilon} \stackrel{\text{def}}{=} \operatorname{tr}\left(\mathscr{K}_\mu(\mathscr{K}_\mu + \epsilon\mathscr{I}_T)^{-1}\right). \tag{3.12}$$

- *Inner Product characterization:*

$$\tau_{\mu,\epsilon}(t) = \frac{1}{T}\cdot\langle\varphi_t, (\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_t\rangle_\mu. \tag{3.13}$$

- *Minimization Characterization:*

$$\tau_{\mu,\epsilon}(t) = \frac{1}{T}\cdot\min_{\beta\in L_2(T)}\frac{\left\|\mathscr{F}_\mu\beta - \varphi_t\right\|_\mu^2}{\epsilon} + \left\|\beta\right\|_T^2. \tag{3.14}$$

In Theorem 3.4.2, we give our formal statement that the ridge leverage function can be used to randomly sample time domain points to discretize the regression problem in (3.10) and solve it approximately. While complex in appearance, readers familiar with randomized linear algebra will recognize Theorem 3.4.2 as closely analogous to standard approximate regression results for leverage score sampling from finite matrices (Clarkson and Woodruff, 2017). As discussed, since we are typically unable to sample according to the true ridge leverage function, we give a general result, showing that sampling with any upper bound function with a finite integral suffices.

Theorem 3.4.2 (Approximate regression via leverage function sampling)**.** *Assume that $\epsilon \leq \left\|\mathscr{K}_\mu\right\|_{\text{op}}$.*[15] *Consider a measurable function $\tilde{\tau}_{\mu,\epsilon}(t)$ with $\tilde{\tau}_{\mu,\epsilon}(t) \geq \tau_{\mu,\epsilon}(t)$ for all t and let $\tilde{s}_{\mu,\epsilon} = \int_0^T \tilde{\tau}_{\mu,\epsilon}(t)\,dt$. Let $s = c\cdot\tilde{s}_{\mu,\epsilon}\cdot\left(\log\tilde{s}_{\mu,\epsilon} + 1/\delta\right)$ for sufficiently large fixed constant c and let $t_1,\ldots,t_s$ be i.i.d. time points selected by drawing each randomly from $[0,T]$ with probability proportional to $\tilde{\tau}_{\mu,\epsilon}(t)$. For $j\in 1,\ldots,s$, let $w_j = \sqrt{\frac{1}{sT}\cdot\frac{\tilde{s}_{\mu,\epsilon}}{\tilde{\tau}_{\mu,\epsilon}(t_j)}}$. Let $\mathbf{F}:\mathbb{C}^s\to L_2(\mu)$ be the operator defined by:*

$$\left[\mathbf{F}\,g\right](\xi) = \sum_{j=1}^{s} w_j\cdot g(j)\cdot e^{-2\pi i\xi t_j}$$

and $\mathbf{y},\mathbf{n}\in\mathbb{R}^s$ be the vectors with $\mathbf{y}(j) = w_j\cdot y(t_j)$ and $\mathbf{n}(j) = w_j\cdot n(t_j)$. Let:

$$\tilde{g} = argmin_{g\in L_2(\mu)}\left[\|\mathbf{F}^*g - (\mathbf{y}+\mathbf{n})\|_2^2 + \epsilon\|g\|_\mu^2\right] \tag{3.15}$$

With probability $\geq 1-\delta$:

$$\|\mathscr{F}_\mu^*\tilde{g} - (y+n)\|_T^2 + \epsilon\|\tilde{g}\|_\mu^2 \leq 3\min_{g\in L_2(\mu)}\left[\|\mathscr{F}_\mu^*g - (y+n)\|_T^2 + \epsilon\|g\|_\mu^2\right]. \tag{3.16}$$

[15] If $\epsilon > \left\|\mathscr{K}_\mu\right\|_{\text{op}}$ then (3.10) is solved to a constant approximation factor by the trivial solution $g = 0$.

A generalized version of this result is proven in Appendix B.2, which holds even when $\tilde{g}$ is only an approximate minimizer of (3.15).

Theorem 3.4.2 shows that $\tilde{g}$ obtained from solving the discretized regression problem provides an approximate solution to (3.10) and by Claim 3.4.1, $\tilde{y} = \mathscr{F}_\mu^* \tilde{g}$ solves Problem 3.2.1 with parameter $\Theta(\epsilon)$. If we have $\tilde{\tau}_{\mu,\epsilon}(t) = \tau_{\mu,\epsilon}(t)$, Theorem 3.4.2 combined with Claim 3.4.1 shows that Problem 3.2.1 with parameter $\Theta(\epsilon)$ can be solved with sample complexity $O\left(s_{\mu,\epsilon} \cdot \log s_{\mu,\epsilon}\right)$, since by (3.12), $\int_0^T \tau_{\mu,\epsilon}(t)\,dt = s_{\mu,\epsilon}$. Note that, by simply decreasing the regularization parameter in (3.10) by a constant factor, we can solve Problem 3.2.1 with parameter ϵ. The asymptotic complexity is identical since, by (3.14), for any $c \le 1$, any $t \in [0, T]$, $\tau_{\mu,c\epsilon}(t) \le \frac{1}{c}\tau_{\mu,\epsilon}(t)$ and so:

$$s_{\mu,c\epsilon} \le \frac{1}{c} s_{\mu,\epsilon}. \tag{3.17}$$

This proves the sample complexity result of Theorem 3.2.1. However, since it is not clear that sampling according to $\tau_{\mu,\epsilon}(t)$ can be done efficiently (or at all), it does not yet give an algorithm yielding this complexity.[16] This issue will be addressed in Section 3.5, where we prove Theorem 3.2.2.

We prove Theorem 3.4.2 in Appendix B.2. We show that leverage function sampling satisfies, with good probability, an affine embedding guarantee: that $\|\mathbf{F}^* g - (\mathbf{y} + \mathbf{n})\|_2^2 + \epsilon\|g\|_\mu^2$ closely approximates $\|\mathscr{F}_\mu^* g - (y + n)\|_T^2 + \epsilon\|g\|_\mu^2$ *for all* $g \in L_2(\mu)$. Thus, a (near) optimal solution to the discretized problem, $\min_{g \in L_2(\mu)} \left[\|\mathbf{F}^* g - (\mathbf{y} + \mathbf{n})\|_2^2 + \epsilon\|g\|_\mu^2\right]$, gives a near optimal solution to the original problem, $\min_{g \in L_2(\mu)} \left[\|\mathscr{F}_\mu^* g - (y + n)\|_T^2 + \epsilon\|g\|_\mu^2\right]$. Our proof of the affine embedding property is analogous to existing proofs for finite dimensional matrices (Clarkson and Woodruff, 2017; Avron et al., 2017b).

3.4.2 Efficient solution of the discretized problem

Given an upper bound on the ridge leverage function $\tilde{\tau}_{\mu,\epsilon}(t) \ge \tau_{\mu,\epsilon}(t)$, we can apply Theorem 3.4.2 to approximately solve the ridge regression problem of (3.10) and therefore Problem 3.2.1 by Claim 3.4.1. In Section 3.5 we show how to obtain such an upper bound for any μ using a universal distribution.

First, however, we demonstrate how to apply Theorem 3.4.2 algorithmically. Specifically, we show how to solve the randomly discretized problem of (3.15) efficiently. Combined with Theorem 3.4.2 and our bound on $\tau_{\mu,\epsilon}(t)$ given in Section 3.5, this yields a randomized algorithm (Algorithm 13) for Problem 3.2.1. The formal analysis of Algorithm 13 is given in Theorem 3.4.3.

Theorem 3.4.3 (Efficient signal reconstruction given leverage function upper bounds)**.** *Assume*

[16] We conjecture that the existential sample complexity can in fact be upper bounded by $O(s_{\mu,\epsilon})$ by adapting deterministic sampling methods for finite matrices to the operator setting (Cohen et al., 2016a), like we do in Lemma B.2.3.

Algorithm 13 Time Point Sampling and Signal Reconstruction

input: Probability measure $\mu(\xi)$, $\epsilon, \delta > 0$, time bound T, and function $y : [0,T] \to \mathbb{R}$. Ridge leverage function upper bound $\tilde{\tau}_{\mu,\epsilon}(t) \geq \tau_{\mu,\epsilon}(t)$ with $\tilde{s}_{\mu,\epsilon} = \int_0^T \tilde{\tau}_{\mu,\epsilon}(t)dt$.
output: $t_1, \ldots, t_s \in [0,T]$ and $\mathbf{z} \in \mathbb{C}^s$.

1: Let $s = c \cdot \tilde{s}_{\mu,\epsilon} \cdot \left(\log \tilde{s}_{\mu,\epsilon} + \frac{1}{\delta}\right)$ for a sufficiently large constant c.
2: Independently sample $t_1, \ldots, t_s \in [0,T]$ with probability proportional to $\tilde{\tau}_{\mu,\epsilon}(t)$ and set the weight $w_i := \sqrt{\frac{1}{sT} \cdot \frac{\tilde{s}_{\mu,\epsilon}}{\tilde{\tau}_{\mu,\epsilon}(t_i)}}$.
3: Let $\mathbf{K} \in \mathbb{C}^{s \times s}$ be the matrix with $\mathbf{K}(i,j) = w_i w_j \cdot k_\mu(t_i, t_j)$.
4: Let $\bar{\mathbf{y}} \in \mathbb{C}^s$ be the vector with $\bar{\mathbf{y}}(i) = w_i \cdot [y(t_i) + n(t_i)]$.
5: Compute $\bar{\mathbf{z}} := (\mathbf{K} + \epsilon \mathbf{I})^{-1} \bar{\mathbf{y}}$.
6: **return** $t_1, \ldots, t_s \in [0,T]$ and $\mathbf{z} \in \mathbb{C}^s$ with $\mathbf{z}(i) = \bar{\mathbf{z}}(i) \cdot w_i$.

Algorithm 14 Evaluation of Reconstructed Signal

input: Probability measure $\mu(\xi)$, $t_1, \ldots, t_s \in [0,T]$, $\mathbf{z} \in \mathbb{C}^s$, and evaluation point $t \in [0,T]$.
output: Reconstructed function value $\tilde{y}(t)$.

1: For $i \in \{1, \ldots, s\}$, compute $k_\mu(t_i, t) = \int_{\xi \in \mathbb{R}} e^{-2\pi i (t_i - t)} d\mu(\xi)$.
2: **return** $\tilde{y}(t) = \sum_{i=1}^s \mathbf{z}(i) \cdot k_\mu(t_i, t)$.

that $\epsilon \leq \left\|\mathcal{K}_\mu\right\|_{\text{op}}$.[17] *Algorithm 13 returns* $t_1, \ldots, t_s \in [0,T]$ *and* $\mathbf{z} \in \mathbb{C}^s$ *such that* $\tilde{y}(t) = \sum_{i=1}^s \mathbf{z}(i) \cdot k_\mu(t_i, t)$ *(as computed in Algorithm 14) satisfies with probability* $\geq 1 - \delta$*:*

$$\left\|\tilde{y} - y\right\|_T^2 \leq 6\epsilon \left\|x\right\|_\mu^2 + 8 \left\|n\right\|_T^2.$$

Suppose we can sample $t \in [0,T]$ *with probability proportional to* $\tilde{\tau}_{\mu,\epsilon}(t)$ *in time* W *and compute the kernel function* $k_\mu(t_1, t_2) = \int_{\xi \in \mathbb{R}} e^{-2\pi i (t_1 - t_2)} d\mu(\xi)$ *in time* Z*. Algorithm 13 queries* $y + n$ *at* s *points and runs in* $O\left(s \cdot W + s^2 \cdot Z + s^\omega\right)$ *time*[18] *where* $s = O\left(\tilde{s}_{\mu,\epsilon} \cdot \left(\log \tilde{s}_{\mu,\epsilon} + 1/\delta\right)\right)$*. Algorithm 14 evaluates* $\tilde{y}(t)$ *in* $O(s \cdot Z)$ *time for any* t*.*

Proof. In Step 2 of Algorithm 13, $t_1, \ldots, t_s$ are sampled according to $\tilde{\tau}_{\mu,\epsilon}(t)$, which upper bounds $\tau_{\mu,\epsilon}(t)$. We can thus apply Theorem 3.4.2. If the constant c in Step 1 is set large enough, with probability $\geq 1 - \delta$, letting $\mathbf{F}, \mathbf{y}$, and $\mathbf{n}$ be as defined in that theorem, (3.16) holds for

$$\tilde{g} = \text{argmin}_{g \in L_2(\mu)} \left[\left\|\mathbf{F}^* g - (\mathbf{y} + \mathbf{n})\right\|_2^2 + \epsilon \left\|g\right\|_\mu^2 \right].$$

Therefore, letting $\tilde{y} \overset{\text{def}}{=} \mathcal{F}_\mu^* \tilde{g}$ and applying Claim 3.4.1, with probability $\geq 1 - \delta$,

$$\left\|\tilde{y} - y\right\|_T^2 \leq 6\epsilon \left\|x\right\|_\mu^2 + 8 \left\|n\right\|_T^2. \tag{3.18}$$

Furthermore, the minimizer $\tilde{g}$ is indeed unique and can be written as (see Lemma B.2.1 in

[17] As discussed for Theorem 3.4.2, if $\epsilon > \left\|\mathcal{K}_\mu\right\|_{\text{op}}$, Problem 3.2.1 is trivially solved by $\tilde{y} = 0$.

[18] Here $\omega < 2.373$ is the exponent of fast matrix multiplication. s^ω is the theoretically fastest runtime required to invert a dense $s \times s$ matrix. We note that the s^ω term may be thought of as s^3 in practice, and potentially could be accelerated using a variety of techniques for fast (regularized) linear system solvers.

Appendix B.2):

$$\tilde{g} = \mathbf{F}(\mathbf{K} + \epsilon\mathbf{I})^{-1}(\mathbf{y} + \mathbf{n}) = \mathbf{F}(\mathbf{K} + \epsilon\mathbf{I})^{-1}\bar{\mathbf{y}}$$

where $\mathbf{K} = \mathbf{F}^*\mathbf{F}$ is as defined in Step 3 of Algorithm 13 and $\bar{\mathbf{y}} = \mathbf{y} + \mathbf{n}$ is formed in Step 4. If we let $\bar{\mathbf{z}} = (\mathbf{K} + \epsilon\mathbf{I})^{-1}\bar{\mathbf{y}}$ and let $\mathbf{z}$ have $\mathbf{z}(i) = \bar{\mathbf{z}}(i) \cdot w_i$ as in Steps 5 and 6, we can see that:

$$\begin{aligned}\tilde{y} = \mathscr{F}_\mu^* \tilde{g} &= \sum_{i=1}^{s} \bar{\mathbf{z}}(i) \cdot w_i \cdot k_\mu(t_i, t) \\ &= \sum_{i=1}^{s} \mathbf{z}(i) \cdot k_\mu(t_i, t),\end{aligned}$$

giving the expression returned in Algorithm 14. Combined with (3.18), this completes the accuracy bound of the theorem. The runtime and sample complexity bounds follow from observing that:

- $s \cdot W$ time is required to sample $t_1, \ldots, t_s$ in Step 2.
- $s^2 \cdot Z$ time is required to form $\mathbf{K}$ in Step 3.
- s queries to $y + n$ are required to form $\bar{\mathbf{y}}$ in Step 4.
- $O(s^\omega)$ time is required to compute $\bar{\mathbf{z}} := (\mathbf{K} + \epsilon\mathbf{I})^{-1}\bar{\mathbf{y}}$ in Step 5. This runtime could potentially be improved with a variety of fast system solvers. We take s^ω as a simple upper bound.
- $O(s \cdot Z)$ time is required to compute $k(t_1, t), \ldots, k(t_s, t)$ to evaluate $\tilde{y}(t)$ in Algorithm 14.

This completes the proof of Theorem 3.4.3. □

Remark: As discussed, in Section 3.5 we will give a ridge leverage function upper bound that can be sampled from in $W = O(1)$ time and closely bounds the true leverage function for any μ, giving $\tilde{s}_{\mu,\epsilon} = O\left(s_{\mu,\epsilon} \cdot \log s_{\mu,\epsilon}\right)$. Using this upper bound to sample time domain points, our sample complexity s is thus within a $O(\log s_{\mu,\epsilon})$ factor of the best possible using Theorem 3.4.2, which we would achieve if sampling using the true ridge leverage function.

In Appendix B.3 we prove a tighter leverage function bound than the one in Section 3.5 for bandlimited signals, removing the logarithmic factor in this case. It is not hard to see that for general μ we can also achieve optimal sample complexity by further subsampling $t_1, \ldots, t_s$ using the ridge leverage scores of $\mathbf{K}^{1/2}$. These scores can be computed in $\tilde{O}(s \cdot s_{\mu,\epsilon}^2)$ time using known techniques for finite kernel matrices (Musco and Musco, 2017). Subsampling $O\left(\frac{s_{\mu,\epsilon} \log s_{\mu,\epsilon}}{\Delta^2}\right)$ time domain points according to these scores lets us approximately solve the discretized problem of (3.15) to error $(1 + \Delta)$.

Applying the more general version of Theorem 3.4.2 stated in Appendix B.2, this yields an approximate solution to (3.10) and thus to Problem 3.2.1. For constant δ, we need just

$O\left(s_{\mu,\epsilon} \cdot \log s_{\mu,\epsilon}\right)$ time samples to to solve the subsampled regression problem, matching the best possible sample complexity of Theorem 3.4.2. By the lower bound given in Section 3.6, Theorem 3.6.2, this complexity is within a $O(\log s_{\mu,\epsilon})$ factor of optimal in nearly all settings. We conjecture that one can in fact achieve within an $O(1)$ factor of the optimal sample complexity by applying deterministic selection methods to $\mathbf{F}$ (Cohen et al., 2016a), similar to the techniques used to prove Lemma B.2.3.

3.5 A Near-optimal Spectrum Blind Sampling Distribution

In the previous section, we showed how to solve Problem 3.2.1 given the ability to sample time points according to the ridge leverage function $\tau_{\mu,\epsilon}$. In general, this function depends strongly on T, μ, and ϵ, and it is not clear if it can be computed or sampled from directly.

Nevertheless, in this section we show that it is possible to efficiently obtain samples from a function that *very closely* approximates the true leverage function for *any* constraint measure μ. In particular we describe a set of closed form functions $\tilde{\tau}_\alpha(t)$, each parameterized by $\alpha > 0$. $\tilde{\tau}_\alpha$ upper bounds the leverage function $\tau_{\mu,\epsilon}$ for any μ and ϵ, as long as the statistical dimension $s_{\mu,\epsilon} = O(\alpha)$. Our upper bound satisfies

$$\int_0^T \tilde{\tau}_\alpha(t)\,dt = O\left(s_{\mu,\epsilon} \cdot \log s_{\mu,\epsilon}\right),$$

which means it can be used in place of the true ridge leverage function to give near optimal sample complexity via Theorem 3.4.2 and 3.4.3. This result is proven formally in Theorem 3.5.6, which as a consequence immediately yields our main technical result, Theorem 3.2.2. The majority of this section is devoted to building tools necessary for proving Theorem 3.5.6.

3.5.1 Uniform leverage bound via Fourier sparsification

We seek a simple closed form function that upper bounds the leverage function $\tau_{\mu,\epsilon}$. Ultimately, we want this upper bound to be very tight, but a natural first question is whether it should exists at all. Is it possible to prove any finite upper bound on $\tau_{\mu,\epsilon}$ without using specific knowledge of μ?

We answer this first question by showing that $\tau_{\mu,\epsilon}$ can be upper bounded by a constant function. Specifically, we show that for $t \in [0,T]$, $\tau_{\mu,\epsilon}(t) \le C$ for $C = \text{poly}(s_{\mu,\epsilon})$. This upper bound depends on the statistical dimension, but importantly, it does not depend on μ. Formally we show:

Theorem 3.5.1 (Uniform leverage function bound)**.** *For all $t \in [0,T]$ and $\epsilon \le 1$*[19]

$$\tau_{\mu,\epsilon}(t) \le \frac{2^{41}(s_{\mu,\epsilon})^5 \log^3(40 s_{\mu,\epsilon})}{T}.$$

[19] If $\epsilon > 1 = \text{tr}(\mathcal{K}_\mu)$, Problem 3.2.1 is trivially solved by returning $\tilde{y} = 0$.

While Theorem 3.5.1 appears to give a relatively weak bound, proving this statement is a key technical challenge. Ultimately, it is used in Section 3.5.3 as one of two main ingredients in proving the much tighter leverage function bound that yields Theorem 3.5.6 and Theorem 3.2.2.

Towards a proof of Theorem 3.5.1, we consider the operator $\mathscr{F}_\mu$ defined in Section 3.3. Since $\mathscr{F}_\mu$ has statistical dimension $s_{\mu,\epsilon}$, $\mathscr{K}_\mu = \mathscr{F}_\mu^* \mathscr{F}_\mu$ can have at most $2s_{\mu,\epsilon}$ eigenvalues $\geq \epsilon$:

$$s_{\mu,\epsilon} = \sum_{i=1}^{\infty} \frac{\lambda_i(\mathscr{K}_\mu)}{\lambda_i(\mathscr{K}_\mu)+\epsilon} \geq \sum_{i:\lambda_i(\mathscr{K}_\mu)\geq\epsilon} \frac{\lambda_i(\mathscr{K}_\mu)}{\lambda_i(\mathscr{K}_\mu)+\epsilon} \geq \frac{\left|i : \lambda_i(\mathscr{K}_\mu) \geq \epsilon\right|}{2}. \tag{3.19}$$

Thus, if we project $\mathscr{F}_\mu$ onto the span of $\mathscr{K}_\mu$'s top $2s_{\mu,\epsilon}$ eigenfunctions (when μ is uniform on an interval these are the prolate spherical wave functions of Slepian and Pollak (1961)) we will approximate $\mathscr{K}_\mu$ up to its small eigenvalues. The total mass of these eigenvalues is bounded by:

$$\sum_{i:\lambda_i(\mathscr{K}_\mu)\leq\epsilon} \lambda_i(\mathscr{K}_\mu) \leq 2\epsilon \cdot \sum_{i:\lambda_i(\mathscr{K}_\mu)\leq\epsilon} \frac{\lambda_i(\mathscr{K}_\mu)}{\lambda_i(\mathscr{K}_\mu)+\epsilon} \leq 2\epsilon \cdot s_{\mu,\epsilon}.$$

Alternatively, instead of projecting onto the span of the eigenfunctions, we can approximate $\mathscr{K}_\mu$ nearly optimally by projecting $\mathscr{F}_\mu$ onto the span of a subset of $O(s_{\mu,\epsilon})$ of its "rows" – i.e. frequencies in the support of μ. For finite linear operators, is well known that such a subset exists: the problem of finding these subsets has been studied extensively in the literature on randomized low-rank matrix approximation under the name *column subset selection* (Sarlos, 2006; Boutsidis et al., 2009; Deshpande and Rademacher, 2010). In Appendix B.2 we show that an analogous result extends to the continuous operator $\mathscr{F}_\mu$:

Theorem 3.5.2 (Frequency subset selection)**.** *For some $s \leq \lceil 36 \cdot s_{\mu,\epsilon} \rceil$ there exists a set of distinct frequencies $\xi_1, \ldots, \xi_s \in \mathbb{R}$ such that, if $\mathbf{C}_s : L_2(T) \to \mathbb{C}^s$ and $\mathbf{Z} : L_2(\mu) \to \mathbb{C}^s$ are defined by:*

$$[\mathbf{C}_s g](j) = \frac{1}{T}\int_0^T g(t) e^{-2\pi i \xi_j t}\, dt \qquad\qquad \mathbf{Z} = (\mathbf{C}_s \mathbf{C}_s^*)^{-1} \mathbf{C}_s \mathscr{F}_\mu^*, \text{[20]} \tag{3.20}$$

then

$$\operatorname{tr}\left(\mathscr{K}_\mu - \mathbf{C}_s^* \mathbf{Z}\mathbf{Z}^* \mathbf{C}_s\right) \leq 4\epsilon \cdot s_{\mu,\epsilon}. \tag{3.21}$$

Note that, if $\varphi_t \in L_2(\mu)$ is defined as $\varphi_t(\xi) \stackrel{\text{def}}{=} e^{-2\pi i t \xi}$ and $\boldsymbol{\phi}_t \in \mathbb{C}^s$ is defined as $\boldsymbol{\phi}_t(j) \stackrel{\text{def}}{=} \varphi_t(\xi_j)$, then we have:

$$\operatorname{tr}\left(\mathscr{K}_\mu - \mathbf{C}_s^* \mathbf{Z}\mathbf{Z}^* \mathbf{C}_s\right) = \frac{1}{T}\int_{t\in[0,T]} \left\|\varphi_t - \mathbf{Z}^* \boldsymbol{\phi}_t\right\|_\mu^2 dt.$$

[20] The fact that $\xi_1, \ldots, \xi_s$ are distinct ensures that $(\mathbf{C}_s \mathbf{C}_s^*)^{-1}$ exists.

Leverage function bound proof sketch. With Theorem 3.5.2 in place, we explain how to use this result to prove Theorem 3.5.1, i.e., to establish a universal bound on the leverage function of $\mathscr{F}_\mu$. For the sake of exposition, we use the term "row" of an operator $\mathscr{A} : L_2(\mu) \to L_2(T)$ to refer to the corresponding operator restricted to some time t. We use the term "column" of an operator as the row of the adjoint operator $\mathscr{A}^* : L_2(T) \to L_2(\mu)$, i.e., the adjoint operator restricted to some frequency ξ.

By Theorem 3.5.2, $\mathbf{C}_s^*\mathbf{Z} : L_2(\mu) \to L_2(T)$ (the projection of $\mathscr{F}_\mu^*$ onto the range of $\mathbf{C}_s$) closely approximates the operator $\mathscr{F}_\mu^*$ yet has columns spanned by just $O(s_{\mu,\epsilon})$ frequencies: $\xi_1, \ldots, \xi_s$. Thus, for any $\alpha \in L_2(\mu)$, $\mathbf{C}_s^*\mathbf{Z}\alpha \in L_2(T)$ is just a Fourier $O(s_{\mu,\epsilon})$-sparse function. Using the maximization characterization of Definition 3.4.1, we can thus bound the time domain ridge leverage function of $\mathbf{C}_s^*\mathbf{Z}$ by appealing to known smoothness bounds for Fourier sparse functions (Chen et al., 2016; Chen and Price, 2019), even for $\epsilon = 0$. When $\epsilon = 0$, the ridge leverage function is known as the *standard leverage function* in the randomized numerical linear algebra literature, and we will refer to them as such.

We can use a similar argument to bound the row norms of the residual operator $\left[\mathscr{F}_\mu^* - \mathbf{C}_s^*\mathbf{Z}\right]$. The columns of this residual operator are each spanned by $O(s_{\mu,\epsilon})$ frequencies, and so are again Fourier sparse functions whose smoothness we can bound. This smoothness ensures that no row can have norm significantly higher than average.

Finally, we note that the time domain ridge leverage function of $\mathscr{F}_\mu^*$ is approximated to within a constant factor by the sum of the standard row leverage function of $\mathbf{C}_s^*\mathbf{Z}$ along with row norms of $\mathscr{F}_\mu^* - \mathbf{C}_s^*\mathbf{Z}$. This gives us a bound on $\mathscr{F}_\mu^*$'s ridge leverage function. We prove this formally below:

Theorem 3.5.3 (Ridge leverage function approximation)**.** *Let $\mathbf{C}_s$ and $\mathbf{Z}$ be the operators guaranteed to exist by Theorem 3.5.2. Let $\ell(t)$ be the standard leverage function of t in $\mathbf{C}_s^*\mathbf{Z}$:*[21]

$$\ell(t) \stackrel{\text{def}}{=} \max_{\{\alpha \in L_2(\mu):\, \|\alpha\|_\mu > 0\}} \frac{1}{T} \cdot \frac{|[\mathbf{C}_s^*\mathbf{Z}\alpha](t)|^2}{\left\|\mathbf{C}_s^*\mathbf{Z}\alpha\right\|_T^2}.$$

Let $r(t)$ be the residual:

$$r(t) \stackrel{\text{def}}{=} \frac{1}{T} \cdot \left\|\varphi_t - \mathbf{Z}^*\boldsymbol{\phi}_t\right\|_\mu^2,$$

where φ_t and $\boldsymbol{\phi}_t$ are as defined in Theorem 3.5.2. Then for all t:

$$\tau_{\mu,\epsilon}(t) \le 2 \cdot \left(\ell(t) + \frac{r(t)}{\epsilon}\right)$$

Proof. For any $\alpha \in L_2(\mu)$ we can write $[\mathscr{F}_\mu^*\alpha](t) = \langle \varphi_t, \alpha \rangle_\mu$ and $[\mathbf{C}_s^*\mathbf{Z}\alpha](t) = \langle \boldsymbol{\phi}_t, \mathbf{Z}\alpha \rangle_{\mathbb{C}^s} =$

[21] Analogously to how $[\mathscr{F}_\mu^*\alpha](t)$ is used in Definition 3.4.1, while $L_2(T)$ is formally a space of equivalence classes of functions, here we use $\mathbf{C}_s^*\mathbf{Z}\alpha$ to denote the specific representative of the equivalence class $\mathbf{C}_s^*\mathbf{Z}\alpha \in L_2(T)$ given by $[\mathbf{C}_s^*\mathbf{Z}\alpha](t) = \sum_{j=1}^s [\mathbf{Z}\alpha](j) \cdot e^{2\pi i \xi_j t} = \langle \boldsymbol{\phi}_t, \mathbf{Z}\alpha \rangle_{\mathbb{C}^s}$. In this way, we can consider the pointwise value $[\mathbf{C}_s^*\mathbf{Z}\alpha](t)$.

$\langle \mathbf{Z}^* \boldsymbol{\phi}_t, \alpha \rangle_\mu$. By the maximization characterization of the ridge leverage function in Definition 3.4.1,

$$\begin{aligned}
\tau_{\mu,\epsilon}(t) &= \frac{1}{T} \cdot \max_{\{\alpha \in L_2(\mu) : \|\alpha\|_\mu > 0\}} \frac{\langle \varphi_t, \alpha \rangle_\mu^2}{\left\| \mathscr{F}_\mu^* \alpha \right\|_T^2 + \epsilon \|\alpha\|_\mu^2} \\
&\leq \frac{2}{T} \cdot \max_{\{\alpha \in L_2(\mu) : \|\alpha\|_\mu > 0\}} \left(\frac{\langle \mathbf{Z}^* \boldsymbol{\phi}_t, \alpha \rangle_\mu^2}{\left\| \mathscr{F}_\mu^* \alpha \right\|_T^2} + \frac{\langle \varphi_t - \mathbf{Z}^* \boldsymbol{\phi}_t, \alpha \rangle_\mu^2}{\epsilon \|\alpha\|_\mu^2} \right) \\
&\leq \frac{2}{T} \cdot \max_{\{\alpha \in L_2(\mu) : \|\alpha\|_\mu > 0\}} \left(\frac{\langle \mathbf{Z}^* \boldsymbol{\phi}_t, \alpha \rangle_\mu^2}{\left\| \mathbf{C}_s^* \mathbf{Z} \alpha \right\|_T^2} + \frac{\left\| \varphi_t - \mathbf{Z}^* \boldsymbol{\phi}_t \right\|_\mu^2}{\epsilon} \right) \\
&= 2 \cdot \left(\ell(t) + \frac{r(t)}{\epsilon} \right)
\end{aligned}$$

where the second to last line follows from observing that due to Cauchy-Schwarz,

$$\langle \varphi_t - \mathbf{Z}^* \boldsymbol{\phi}_t, \alpha \rangle_\mu^2 \leq \|\alpha\|_\mu^2 \cdot \left\| \varphi_t - \mathbf{Z}^* \boldsymbol{\phi}_t \right\|_\mu^2,$$

and that, letting $\mathscr{P}_s = \mathbf{C}_s^* (\mathbf{C}_s \mathbf{C}_s^*)^{-1} \mathbf{C}_s$:

$$\begin{aligned}
\left\| \mathscr{F}_\mu^* \alpha \right\|_T^2 &= \langle \alpha, \mathscr{F}_\mu \mathscr{F}_\mu^* \alpha \rangle_\mu \\
&\geq \langle \alpha, \mathscr{F}_\mu \mathscr{P}_s \mathscr{F}_\mu^* \alpha \rangle_\mu \\
&= \langle \alpha, \mathbf{Z}^* \mathbf{C}_s \mathbf{C}_s^* \mathbf{Z} \alpha \rangle_\mu = \left\| \mathbf{C}_s^* \mathbf{Z} \alpha \right\|_T^2.
\end{aligned}$$

In the above, the inequality is due to the fact that $\mathscr{P}_s$ is an orthogonal projection, so $\mathscr{P}_s \preceq \mathscr{I}_\mu$. This completes the proof. □

With Theorem 3.5.3 in place, we now bound $\bar{\tau}_{\mu,\epsilon}(t) = 2\left(\ell(t) + \frac{r(t)}{\epsilon}\right)$, which yields a uniform bound on the true ridge leverage scores.

Lemma 3.5.1. *Let $\ell(t), r(t)$ be as defined in Theorem 3.5.3 and $\bar{\tau}_{\mu,\epsilon}(t) \stackrel{\text{def}}{=} 2 \cdot \left(\ell(t) + \frac{r(t)}{\epsilon}\right)$. For all $t \in [0, T]$:*

$$\bar{\tau}_{\mu,\epsilon}(t) \leq \frac{15400(36 s_{\mu,\epsilon} + 2)^5 \log^3(36 s_{\mu,\epsilon} + 2)}{T}.$$

Combining Lemma 3.5.1 with Theorem 3.5.3 yields Theorem 3.5.1. We just simplify the constants by noting that for $\epsilon \leq 1$, $s_{\mu,\epsilon} \geq \frac{\mathrm{tr}(\mathscr{K}_\mu)}{2} = \frac{1}{2}$ and so $36 s_{\mu,\epsilon} + 2 \leq 40 s_{\mu,\epsilon}$.

Proof of Lemma 3.5.1. We separately bound the leverage score $\ell(t)$ and residual $r(t)$ components of $\bar{\tau}_{\mu,\epsilon}(t)$ using a similar argument based on the smoothness of sparse Fourier functions for both. Specifically, for both bounds we employ the following smoothness bound of (Chen et al., 2016).

Lemma 3.5.2 (Follows from Lemma 5.1 of Chen et al. (2016))**.** *For any* $f(t) = \sum_{j=1}^{k} v_j e^{2\pi i \xi_j t}$,

$$\max_{x\in[0,T]} \frac{|f(x)|^2}{\|f\|_T^2} \leq 1540 \cdot k^4 \log^3 k.$$

Proof. Lemma 5.1 of Chen et al. (2016), gives the bound without an explicit constant. It is not hard to check that their proof gives the constant of 1540 stated above. □

Bounding the leverage scores $\ell(t)$ of $\mathbf{C}_s^*\mathbf{Z}$.

For every $\alpha \in L_2(\mu)$, $\mathbf{C}_s^*\mathbf{Z}\alpha$ is a Fourier $s = O(s_{\mu,\epsilon})$-sparse function. Specifically, we have $\left[\mathbf{C}_s^*\mathbf{Z}\alpha\right](t) = \sum_{j=1}^{s} [\mathbf{Z}\alpha](j) \cdot e^{2\pi i \xi_j t}$, for frequencies $\xi_1, \ldots, \xi_s \in \mathbb{R}$ given by Theorem 3.5.2. We can thus directly apply Lemma 3.5.2 giving for any $t \in [0, T]$:

$$\begin{aligned}
\ell(t) &\stackrel{\text{def}}{=} \max_{\{\alpha\in L_2(\mu):\|\alpha\|_\mu>0\}} \frac{1}{T} \cdot \frac{|[\mathbf{C}_s^*\mathbf{Z}\alpha](t)|^2}{\left\|\mathbf{C}_s^*\mathbf{Z}\alpha\right\|_T^2} \\
&\leq \max_{\{\alpha\in L_2(\mu):\|\alpha\|_\mu>0\}} \left[\frac{1}{T} \cdot \max_{t'\in[0,T]} \frac{|[\mathbf{C}_s^*\mathbf{Z}\alpha](t')|^2}{\left\|\mathbf{C}_s^*\mathbf{Z}\alpha\right\|_T^2}\right] \\
&\leq \frac{1540}{T} \cdot s^4 \log^3 s
\end{aligned} \tag{3.22}$$

Bounding the residuals $r(t)$.

We start by some intuition. To bound the squared row norms of the residual $\mathscr{F}_\mu^* - \mathbf{C}_s^*\mathbf{Z}$ we show that each "column" of this residual is an $s+1 = O(s_{\mu,\epsilon})$ sparse Fourier function. Thus, applying Lemma 3.5.2, no entry's squared value can significantly exceed the average squared value in the column. This lets us show that no squared row norm $r(t)$ can significantly exceed the average squared row norm, which is bounded by Theorem 3.5.2.

Concretely, define $\vartheta_\xi \in L_2(T)$ by $\vartheta_\xi(t) \stackrel{\text{def}}{=} e^{2\pi i t\xi}$, and notice that given $g \in L_2(T)$ the function $\xi \mapsto \langle \vartheta_\xi, g\rangle_T$ is equal to $\mathscr{F}_\mu g$ in the $L_2(T)$ sense (i.e., is a member of the equivalence class $\mathscr{F}_\mu g$). For $\xi \in \mathbb{R}$, let $\mathbf{z}_\xi \in \mathbb{C}^s$ be given by $\mathbf{z}_\xi(j) = \langle \vartheta_\xi, \mathbf{C}_s^*(\mathbf{C}_s\mathbf{C}_s^*)^{-1}\mathbf{e}_j\rangle_T^*$ where $\mathbf{e}_j$ is the j^{th} standard basis vector in $\mathbb{C}^s$. The function $\xi \mapsto \langle \mathbf{z}_\xi, \boldsymbol{\phi}_t\rangle = \sum_{j=1}^{s} \mathbf{z}_\xi^*(j) e^{-2\pi i \xi_j t}$ is equal in the $L_2(\mu)$ sense to $\mathbf{Z}^*\boldsymbol{\phi}_t$. Let us define:

$$r_\xi(t) = e^{-2\pi i \xi t} - \sum_{j=1}^{s} \mathbf{z}_\xi^*(j) e^{-2\pi i \xi_j t}.$$

For a fixed t, consider the function $\xi \mapsto r_\xi(t)$. We have $r_\xi(t) = \varphi_t - \mathbf{Z}^*\boldsymbol{\phi}_t$. Thus, we can write

$$r(t) = \frac{1}{T}\left\|\varphi_t - \mathbf{Z}^*\boldsymbol{\phi}_t\right\|_\mu^2 = \frac{1}{T}\int_{\xi\in\mathbb{R}} |r_\xi(t)|^2 \, d\mu(\xi). \tag{3.23}$$

Further, for a fixed ξ, if we consider the function $t \mapsto r_\xi(t)$, which we denote by $r_\xi(\cdot)$, we notice that it is a $s+1 = O(s_{\mu,\epsilon})$ sparse Fourier function, so applying Lemma 3.5.2 we have for any

$\xi \in \mathbb{R}$ and $t \in [0, T]$:

$$\frac{|r_\xi(t)|^2}{\|r_\xi(\cdot)\|_T^2} \leq 1540(s+1)^4 \log^3(s+1). \tag{3.24}$$

Combining (3.24) with (3.23) we can thus bound for any $t \in [0, T]$:

$$\begin{aligned} r(t) &\leq 1540(s+1)^4 \log^3(s+1) \cdot \frac{1}{T} \int_{\xi \in \mathbb{R}} \|r_\xi(\cdot)\|_T^2 \, d\mu(\xi) \\ &= 1540(s+1)^4 \log^3(s+1) \cdot \frac{1}{T^2} \int_{w \in [0,T]} \int_{\xi \in \mathbb{R}} |r_\xi(w)|^2 \, d\mu(\xi) \, dw \\ &= 1540(s+1)^4 \log^3(s+1) \cdot \frac{1}{T^2} \int_{w \in [0,T]} \|\varphi_w - \mathbf{Z}^* \boldsymbol{\phi}_w\|_\mu^2 \, dw. \end{aligned} \tag{3.25}$$

By Theorem 3.5.2 we have $\frac{1}{T} \int_{w \in [0,T]} \|\varphi_w - \mathbf{Z}^* \boldsymbol{\phi}_w\|_\mu^2 \, dw \leq 4\epsilon \cdot s_{\mu,\epsilon}$. Plugging into (3.25) and using that we can choose $s \leq 36 \cdot s_{\mu,\epsilon} + 1$, for all $t \in [0, T]$:

$$r(t) \leq \frac{\epsilon \cdot 6160(36 s_{\mu,\epsilon} + 2)^5 \log^3(36 s_{\mu,\epsilon} + 2)}{T}. \tag{3.26}$$

Combining (3.22) and (3.26) completes the proof of Lemma 3.5.1:

$$\bar{\tau}_{\mu,\epsilon}(t) \stackrel{\text{def}}{=} 2 \cdot \left(\ell(t) + \frac{r(t)}{\epsilon} \right) \leq \frac{15400(36 s_{\mu,\epsilon} + 2)^5 \log^3(36 s_{\mu,\epsilon} + 2)}{T}.$$

□

Theorem 3.5.1 gives a universal uniform bound on the ridge leverage scores corresponding to measure μ in terms of $s_{\mu,\epsilon}$. If we directly sample time points according to the uniform distribution over $[0, T]$, this theorem shows that $\text{poly}(s_{\mu,\epsilon})$ samples and $\text{poly}(s_{\mu,\epsilon})$ runtime suffice to apply Theorem 3.4.3 and solve Problem 3.2.1 with good probability. This is already a surprising result, showing that the simplest sampling scheme, uniform random sampling, can give bounds in terms of the optimal complexity $s_{\mu,\epsilon}$ for *any* μ. Existing methods with similar complexity, such as those that interpolate bandlimited signals using prolate spheroidal wave functions (Rokhlin et al., 2001) require nonuniform sampling. Methods that use uniform sampling, such as truncated Whittaker-Shannon, have sample complexity depending polynomially rather than logarithmically on the desired error ϵ.

3.5.2 Gap-based leverage score bound

Our final result gives a much tighter bound on the ridge leverage scores than the uniform bound of Theorem 3.5.1. The key idea is to show that the bound is loose for t bounded away from the edges of $[0, T]$. Specifically we have:

Theorem 3.5.4 (Gap-Based Leverage Score Bound)**.** *For all t,*

$$\tau_{\mu,\epsilon}(t) \leq \frac{s_{\mu,\epsilon}}{\min(t, T-t)}.$$

Proof. Consider $t \in [0, T/2]$. We will show that $\tau_{\mu,\epsilon}(t) \leq \frac{s_{\mu,\epsilon}}{t}$. A symmetric proof will hold for $t \in [T/2, T]$, giving the theorem. We define an auxiliary operator: $\mathscr{F}_{\mu,t} : L_2(T) \to L_2(\mu)$ which is given by restricting the integration in $\mathscr{F}_\mu$ to $[0, t]$. Specifically, for $f \in L_2(T)$ we have:

$$\left[\mathscr{F}_{\mu,t} f\right](\xi) = \frac{1}{T}\int_0^t f(s) e^{-2\pi i s\xi}\, ds. \tag{3.27}$$

We can see that $[\mathscr{F}^*_{\mu,t} g](s) = \int_{\mathbb{R}} g(\xi) e^{2\pi i s\xi}\, d\mu(\xi)$ for $s \in [0, t]$ and $[\mathscr{F}^*_{\mu,t} g](s) = 0$ for $s \in (t, T]$. We will use the leverage score of some $s \in [0, t]$ in the restricted operator $\mathscr{F}_{\mu,t}$ to upper bound those of t in $\mathscr{F}_\mu$. We start by defining these scores analogously to Definition 3.4.1 for $\mathscr{F}_\mu$.

Definition 3.5.1 (Restricted ridge leverage scores)**.** For probability measure μ on $\mathbb{R}$, time length T, $t \in [0, T]$ and $\epsilon \geq 0$, define the ϵ-ridge leverage score of $s \in [0, t]$ in $\mathscr{F}_{\mu,t}$ as:

$$\tau_{\mu,\epsilon,t}(s) = \frac{1}{T} \cdot \max_{\{\alpha \in L_2(\mu):\, \|\alpha\|_\mu > 0\}} \frac{|[\mathscr{F}_{\mu,t}\alpha](s)|^2}{\left\|\mathscr{F}^*_{\mu,t}\alpha\right\|_T^2 + \epsilon\|\alpha\|_\mu^2}.$$

We have the following leverage score properties, analogous to those given for $\mathscr{F}_\mu$ in Theorem 3.4.1:

Theorem 3.5.5 (Restricted leverage score properties)**.** *Let $\tau_{\mu,\epsilon,t}(s)$ be as in Definition 3.5.1.*

- *The leverage scores integrate to the statistical dimension:*

$$\int_0^t \tau_{\mu,\epsilon,t}(s)\, ds = s_{\mu,\epsilon,t} \overset{\text{def}}{=} \operatorname{tr}\left(\mathscr{F}^*_{\mu,t}\mathscr{F}_{\mu,t}(\mathscr{F}^*_{\mu,t}\mathscr{F}_{\mu,t} + \epsilon\mathscr{I}_T)^{-1}\right). \tag{3.28}$$

- *Inner Product Characterization: Letting $\varphi_s \in L_2(\mu)$ have $\varphi_s(\xi) = e^{-2\pi i s\xi}$ for $s \in [0, t]$,*

$$\tau_{\mu,\epsilon,t}(s) = \frac{1}{T} \cdot \left\langle \varphi_s, (\mathscr{F}_{\mu,t}\mathscr{F}^*_{\mu,t} + \epsilon\mathscr{I}_\mu)^{-1}\varphi_s \right\rangle_\mu. \tag{3.29}$$

- *Minimization Characterization:*

$$\tau_{\mu,\epsilon,t}(s) = \frac{1}{T} \cdot \min_{\beta \in L_2(T)} \frac{\left\|\mathscr{F}_{\mu,t}\beta - \varphi_s\right\|_\mu^2}{\epsilon} + \|\beta\|_T^2. \tag{3.30}$$

We first show that the restricted leverage scores of Definition 3.5.1 are not too large on average.

Claim 3.5.1 (Restricted statistical dimension bound)**.**

$$\int_0^T \tau_{\mu,\epsilon,t}(s)\,ds \le s_{\mu,\epsilon}. \tag{3.31}$$

Proof. Via (3.28) we have $\int_0^t \tau_{\mu,\epsilon,t}(s)\,ds = s_{\mu,\epsilon,t}$ which we can write as:

$$s_{\mu,\epsilon,t} = \operatorname{tr}\left(\mathscr{F}_{\mu,t}^*\mathscr{F}_{\mu,t}(\mathscr{F}_{\mu,t}^*\mathscr{F}_{\mu,t}+\epsilon\mathscr{I}_T)^{-1}\right) = \operatorname{tr}\left(\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^*(\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^*+\epsilon\mathscr{I}_\mu)^{-1}\right).$$

By Claim B.1.10, $\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^* \preceq \mathscr{F}_\mu\mathscr{F}_\mu^* = \mathscr{G}_\mu$. Since $\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^*(\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^*+\epsilon\mathscr{I}_\mu)^{-1} = \mathscr{I}_\mu - \epsilon(\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^*+\epsilon\mathscr{I}_\mu)^{-1}$ and $\mathscr{G}_\mu(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1} = \mathscr{I}_\mu - \epsilon(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}$. Therefore, by Claim B.1.2 and since the trace is monotone for trace-class operators,

$$s_{\mu,\epsilon,t} = \operatorname{tr}\left(\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^*(\mathscr{F}_{\mu,t}\mathscr{F}_{\mu,t}^*+\epsilon\mathscr{I}_\mu)^{-1}\right) \le \operatorname{tr}\left(\mathscr{G}_\mu(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}\right) = s_{\mu,\epsilon}$$

which gives the claim. □

From Claim 3.5.1 we immediately have:

Claim 3.5.2. *There exists $s^\star \in [0,t]$ with $\tau_{\mu,\epsilon,t}(s^\star) \le \frac{s_{\mu,\epsilon}}{t}$.*

Proof. Assume for the sake of contradiction that $\tau_{\mu,\epsilon,t}(s) > \frac{s_{\mu,\epsilon}}{t}$ for all $s \in [0,t]$. Then by (3.28),

$$\int_0^t \tau_{\mu,\epsilon,t}(s)\,ds > t\cdot\frac{s_{\mu,\epsilon}}{t} = s_{\mu,\epsilon}.$$

This contradicts Claim 3.5.1, giving the claim. □

We now show that the leverage score of $s^\star$ in $\mathscr{F}_{\mu,t}$ upper bounds the leverage score of t in $\mathscr{F}_\mu$, completing the proof of Theorem 3.5.4. We apply the minimization characterization of Theorem 3.5.5, equation (3.30), to prove that by simply shifting an optimal solution for $s^\star$ we can show the existence of a good solution for t, upper bound its leverage score by that of $s^\star$ and give $\tau_{\mu,\epsilon}(t) \le \tau_{\mu,\epsilon,t}(s^\star) \le \frac{s_{\mu,\epsilon}}{t}$ by Claim 3.5.2.

Formally, by Claim 3.5.2 and (3.30), there is some $\beta^\star \in L_2(T)$ achieving:

$$\frac{1}{T}\cdot\frac{\left\|\mathscr{F}_{\mu,t}\beta^\star - \varphi_{s^\star}\right\|_\mu^2}{\epsilon} + \left\|\beta^\star\right\|_T^2 = \tau_{\mu,\epsilon,t}(s^\star) \le \frac{s_{\mu,\epsilon}}{t}. \tag{3.32}$$

We can assume without loss of generality that $\beta^\star(s) = 0$ for $s \notin [0,t]$, since $\mathscr{F}_{\mu,t}\beta^\star$ is unchanged if we set $\beta^\star(s) = 0$ on this range and since doing this cannot increase $\left\|\beta\right\|_T^2$. Now, let $\bar\beta \in L_2(T)$ be given by $\bar\beta(s) = \beta^\star(s-(t-s^\star))$. That is, $\bar\beta$ is just $\beta^\star$ shifted from the range $[0,t]$ to the range

$[t-s^\star, 2t-s^\star]$. Note that since we are assuming $t \le T/2$, $[t-s^\star, 2t-s^\star] \subset [0,T]$. For any ξ:

$$\begin{aligned}
[\mathscr{F}_\mu \bar{\beta}](\xi) &= \frac{1}{T}\int_0^T \bar{\beta}(s) e^{-2\pi i s\xi} ds \\
&= \frac{1}{T}\int_{t-s^\star}^{2t-s^\star} \beta^\star(s-(t-s^\star)) e^{-2\pi i s\xi} ds \\
&= \frac{1}{T}\int_0^t \beta^\star(s) e^{-2\pi i (s+(t-s^\star))\xi} ds \\
&= [\mathscr{F}_{\mu,t}\beta^\star](\xi)\cdot e^{-2\pi i (t-s^\star)\xi}.
\end{aligned} \tag{3.33}$$

Now,

$$\varphi_t(\xi) = e^{-2\pi i t\xi} = e^{-2\pi i (t-s^\star)\xi}\cdot \varphi_{s^\star}(\xi).$$

Combined with (3.33) this gives:

$$\begin{aligned}
\left\|\mathscr{F}_\mu\bar{\beta} - \varphi_t\right\|_\mu^2 = \int_\xi \left|[\mathscr{F}_\mu\bar{\beta}](\xi) - \varphi_t(\xi)\right|^2 d\mu(\xi) &= \int_\xi \left|\left([\mathscr{F}_{\mu,t}\beta^\star](\xi) - \varphi_{s^\star}(\xi)\right)\cdot e^{-2\pi i (t-s^\star)\xi}\right|^2 d\mu(\xi) \\
&= \int_\xi \left|[\mathscr{F}_{\mu,t}\beta^\star](\xi) - \varphi_{s^\star}(\xi)\right|^2 d\mu(\xi) \\
&= \left\|\mathscr{F}_{\mu,t}\beta^\star - \varphi_{s^\star}\right\|_\mu^2.
\end{aligned} \tag{3.34}$$

Finally, noting that $\left\|\bar{\beta}\right\|_T = \left\|\beta^\star\right\|_T$ and applying the minimization characterization of Theorem 3.4.1, the bound in (3.34) along with (3.32) gives:

$$\tau_{\mu,\epsilon}(t) \le \frac{1}{T}\cdot\frac{\left\|\mathscr{F}_\mu\bar{\beta}-\varphi_t\right\|_\mu^2}{\epsilon} + \left\|\bar{\beta}\right\|_T^2 = \frac{\left\|\mathscr{F}_{\mu,t}\beta^\star - \varphi_{s^\star}\right\|_\mu^2}{\epsilon} + \left\|\beta^\star\right\|_T^2 \le \frac{s_{\mu,\epsilon}}{t},$$

which completes the theorem.

□

3.5.3 Nearly tight leverage score bound

Combining Theorems 3.5.1 and 3.5.4 gives our tight, spectrum blind leverage score bound.

Theorem 3.5.6 (Spectrum Blind Leverage Score Bound)**.** *For any $\alpha, T \ge 0$ let $\tilde{\tau}_\alpha(t)$ be given by:*

$$\tilde{\tau}_\alpha(t) = \begin{cases} \frac{\alpha}{256\cdot\min(t,T-t)} \text{ for } t \in [T/\alpha^5, T(1-1/\alpha^5)] \\ \frac{\alpha^6}{T} \text{ for } t \in [0, T/\alpha^5]\cup[T(1-1/\alpha^5), T]. \end{cases}$$

For any probability measure μ, $T \ge 0$, $0 \le \epsilon \le 1$ and $t \in [0,T]$, if $\alpha \ge 256\cdot s_{\mu,\epsilon}$:

$$\tau_{\mu,\epsilon}(t) \le \tilde{\tau}_\alpha(t) \text{ and } \tilde{s}_\alpha \overset{\text{def}}{=} \int_0^T \tilde{\tau}_\alpha(t)\, dt \le \frac{\alpha\cdot\log\alpha}{2}.$$

A visualization of $\tilde{\tau}_\alpha$ is given in Figure 3.3.

Proof. The fact that $\tau_{\mu,\epsilon}(t) \le \tilde{\tau}_\alpha(t)$ follows from Theorems 3.5.1 and 3.5.4:

- For $t \in [T/\alpha^5, T(1-1/\alpha^5)]$, by Theorem 3.5.4 if $\alpha \ge 256 \cdot s_{\mu,\epsilon}$ we have

$$\tilde{\tau}_\alpha(t) = \frac{\alpha}{256 \cdot \min(t, T-t)} \ge \tau_{\mu,\epsilon}(t).$$

- For $t \in [0, T/\alpha^5] \cup [T(1-1/\alpha^5), T]$, by Theorem 3.5.1 we can bound,

$$\tau_{\mu,\epsilon}(t) \le \frac{2^{41} s_{\mu,\epsilon}^5 \log^3(40 s_{\mu,\epsilon})}{T} \le \frac{2^{47} s_{\mu,\epsilon}^6}{T} \le \frac{\alpha^6}{T}$$

  for $\alpha \ge 256 \cdot s_{\mu,\epsilon}$. Note that the second inequality uses that $\log^3(40x) \le 64x$ for any x.

The integral of the approximate scores $\tilde{s}_\alpha$ is bounded as:

$$\begin{aligned}
\int_0^T \tilde{\tau}_\alpha(t)\,dt &= \int_{T/\alpha^5}^{T(1-1/\alpha^5)} \frac{\alpha}{256 \cdot \min(t, T-t)}\,dt + 2\int_0^{T/\alpha^5} \frac{\alpha^6}{T}\,dt \\
&= \frac{2}{256}\int_{T/\alpha^5}^{T/2} \frac{\alpha}{t}\,dt + 2\alpha \\
&= \frac{\alpha}{128} \cdot \left[\log(T/2) - \log(T/\alpha^5)\right] + 2\alpha \qquad (3.35) \\
&\le \frac{5\alpha \log \alpha}{128} + 2\alpha \le \frac{\alpha \log \alpha}{2}.
\end{aligned}$$

where the last inequality follows since for $\epsilon \le 1$, $s_{\mu,\epsilon} \ge 1/2$ and so $\log(\alpha) \ge 9/2$. □

3.5.4 Putting it all together: generic signal reconstruction

Finally, we combine the leverage score bound of Theorem 3.5.6 with Theorem 3.4.3 to give our main algorithmic result, Theorem 3.2.3 (and as a corollary, Theorem 3.2.2). We state the full theorem below:

Theorem 3.2.3 (Main result, algorithmic complexity)**.** *Consider any measure μ, for which we can compute the kernel $k_\mu(t_1, t_2) = \int_{\xi \in \mathbb{R}} e^{-2\pi i (t_1 - t_2)}\,d\mu(\xi)$ for any $t_1, t_2 \in [0, T]$ in time Z.*

Let $\tilde{\tau}_\alpha(t)$ be as defined in Theorem 3.5.6. For any $\epsilon \le \left\|\mathcal{K}_\mu\right\|_{\text{op}}$ and $T > 0$, let $\tilde{\tau}_{\mu,\epsilon}(t) = \tilde{\tau}_\alpha(t)$ for $\alpha = \beta \cdot s_{\mu,\epsilon}$ with $\beta \ge 256$. Algorithm 13 applied with $\tilde{\tau}_{\mu,\epsilon}(t)$ and failure probability δ returns $t_1, \ldots, t_s \in [0, T]$ and $\mathbf{z} \in \mathbb{C}^s$ such that $\tilde{y}(t) = \sum_{i=1}^s \mathbf{z}(i) \cdot k_\mu(t_i, t)$ solves Problem 3.2.1 with parameter 6ϵ and probability $\ge 1 - \delta$. That is, with probability of at least $1 - \delta$:

$$\left\|\tilde{y} - y\right\|_T^2 \le 6\epsilon \left\|x\right\|_\mu^2 + 8 \left\|n\right\|_T^2.$$

The algorithm queries $y + n$ at s points and runs in $O\left(s^2 \cdot Z + s^\omega\right)$ time, where

$$s = O\left(\beta \cdot s_{\mu,\epsilon} \log(\beta \cdot s_{\mu,\epsilon}) \cdot \left[\log(\beta \cdot s_{\mu,\epsilon}) + 1/\delta\right]\right) = \tilde{O}\left(\frac{\beta \cdot s_{\mu,\epsilon}}{\delta}\right).$$

The output $\tilde{y}(t)$ can be evaluated in $O(s \cdot Z)$ time at any t using Algorithm 14.

If we want to solve Problem 3.2.1 with parameter ϵ, it suffices to apply Theorem 3.2.3 with parameter $\epsilon' = \epsilon/6$. The asymptotic complexity will be identical since, by (3.17), $s_{\mu,\epsilon/6} \leq 6 s_{\mu,\epsilon}$.

Proof. The theorem follows directly from Theorem 3.4.3, along with Theorem 3.5.6 which shows that, for $\alpha = \beta \cdot s_{\mu,\epsilon}$ with $\beta \geq 256$ and $\tilde{\tau}_{\mu,\epsilon}(t) = \tilde{\tau}_{\alpha}(t)$ we have:

1. $\tilde{\tau}_{\mu,\epsilon}(t) \geq \tau_{\mu,\epsilon}(t)$ for all $t \in [0, T]$.
2. $\tilde{s}_{\mu,\epsilon} = \int_0^T \tilde{\tau}_{\mu,\epsilon}(t)\, dt = O\big(\beta \cdot s_{\mu,\epsilon} \log(\beta \cdot s_{\mu,\epsilon})\big)$.

The runtime bound follows after noting that we can sample according to τ_α in $W = O(1)$ time using inverse transform sampling since it is straightforward to derive an explicit expression for the CDF and compute the inverse (see (3.35)).

□

3.6 Lower Bound

We conclude by showing that the statistical dimension $s_{\mu,\epsilon}$ tightly characterizes the sample complexity of solving Problem 3.2.1, under a very mild assumption on μ that holds for all natural constraints we discuss in this chapter. Thus, Theorem 3.2.1 is nearly tight.

We first define a quantity, $n_{\mu,\epsilon}$ that gives a natural lower bound on $s_{\mu,\epsilon}$. For any μ, ϵ, let

$$n_{\mu,\epsilon} \stackrel{\text{def}}{=} \sum_{i=1}^{\infty} \mathbb{1}[\lambda_i(\mathscr{K}_\mu) \geq \epsilon]. \tag{3.36}$$

That is, $n_{\mu,\epsilon}$ is the number of eigenvalues of $\mathscr{K}_\mu$ that are larger than ϵ. As shown in (3.19), we always have $n_{\mu,\epsilon} \leq 2 s_{\mu,\epsilon}$. We first prove that solving Problem 3.2.1 requires $\Omega(n_{\mu,\epsilon})$ samples. We then show that, under a very mild constraint on μ (which holds for all μ we consider including sparse, bandlimited, multiband, Gaussian, and Cauchy-Lorentz), $n_{\mu,\epsilon} = \Omega(s_{\mu,\epsilon})$. Thus, $s_{\mu,\epsilon}$ gives a tight bound on the query complexity of solving Problem 3.2.1.

Theorem 3.6.1 (Lower bound in terms of eigenvalue count)**.** *Consider a measure μ, an error parameter $\epsilon > 0$, and any algorithm that solves Problem 3.2.1 with probability $\geq 2/3$ for any function y and makes at most r (possibly adaptive) queries on any input. Then $r \geq n_{\mu,72\epsilon}/20$.*

Proof. We describe a distribution on inputs y on which any deterministic algorithm that takes $r = o(n_{\mu,72\epsilon})$ samples fails with probability $> 1/3$. The theorem then follows by Yao's principle.

Notation: Let $v_1, \ldots, v_{n_{\mu,72\epsilon}} \in L_2(\mu)$ be the eigenfunctions of $\mathscr{G}_\mu$ corresponding to its top $n_{\mu,72\epsilon}$ eigenvalues. Let $\mathbf{Z} : L_2(\mu) \to \mathbb{C}^{n_{\mu,72\epsilon}}$ be the operator with v_i as its i^{th} row – i.e., $[\mathbf{Z}g](i) =$

$\langle v_i, g\rangle_\mu$. Note that $\mathbf{Z}$ has orthonormal rows. Let $\mathbf{D} \in \mathbb{R}^{n_{\mu,72\epsilon} \times n_{\mu,72\epsilon}}$ be a diagonal matrix with $\mathbf{D}_{ii} = \sqrt{\lambda_i(\mathcal{K}_\mu)}$. Let $\mathbf{U} = \mathcal{F}_\mu^* \mathbf{Z}^* \mathbf{D}^{-1}$. We can see that $\mathbf{Z}\mathcal{F}_\mu \mathcal{F}_\mu^* \mathbf{Z}^* = \mathbf{Z}\mathcal{G}_\mu \mathbf{Z}^* = \mathbf{D}^2$ and hence, $\mathbf{U}^*\mathbf{U} = \mathbf{D}^{-1}\mathbf{Z}\mathcal{F}_\mu \mathcal{F}_\mu^* \mathbf{Z}^* \mathbf{D}^{-1} = \mathbf{I}$. While not needed for our proof, $\mathbf{U} : \mathbb{C}^{n_{\mu,72\epsilon}} \to L_2(T)$ is an operator with columns corresponding to all eigenfunctions of $\mathcal{K}_\mu$ with eigenvalue $\geq 72\epsilon$.

Hard Input Distribution: Let $\mathbf{c} \in \mathbb{R}^{n_{\mu,72\epsilon}}$ be a random vector with each entry distributed independently as a Gaussian: $\mathbf{c}(i) \sim \mathcal{N}\left(0, \frac{1}{n_{\mu,72\epsilon}}\right)$. Let $\bar{\mathbf{c}} = \mathbf{D}^{-1}\mathbf{c}$, $x = \mathbf{Z}^*\bar{\mathbf{c}}$, and the random input be $y = \mathcal{F}_\mu^* x$. That is, $y = \mathcal{F}_\mu^* \mathbf{Z}^* \mathbf{D}^{-1}\mathbf{c} = \mathbf{U}\mathbf{c}$ is a random linear combination of the top eigenfunctions of $\mathcal{K}_\mu$. While, formally, $\mathcal{F}_\mu^* x \in L_2(T)$ is an equivalence class of functions, since our input model requires that y admits pointwise evaluation, we will abuse notation, letting y denote the member of this class with $y(t) = \langle \varphi_t, \mathbf{Z}^*\mathbf{D}^{-1}\mathbf{c}\rangle_\mu = \langle \mathbf{D}^{-1}\mathbf{Z}\varphi_t, \mathbf{c}\rangle$, where $\varphi_t(\xi) = e^{-2\pi i t\xi}$.

We prove that accurately reconstructing y drawn from the hard input distribution yields an accurate reconstruction of the random vector $\mathbf{c}$. Since $\mathbf{c}$ is $n_{\mu,72\epsilon}$ dimensional, this reconstruction requires $\Omega(n_{\mu,72\epsilon})$ samples, giving us a lower bound for accurately reconstructing y.

Claim 3.6.1. *For random x distributed as described above, with probability $\geq 5/6$, $\|x\|_\mu^2 \leq \frac{1}{12\epsilon}$.*

Proof.

$$\|x\|_\mu^2 = \langle \mathbf{Z}^*\bar{\mathbf{c}}, \mathbf{Z}^*\bar{\mathbf{c}}\rangle_\mu = \langle \bar{\mathbf{c}}, \mathbf{Z}\mathbf{Z}^*\bar{\mathbf{c}}\rangle = \|\bar{\mathbf{c}}\|_2^2.$$

We then bound $\|\bar{\mathbf{c}}\|_2^2 \leq \|\mathbf{c}\|_2^2 / \lambda_{n_{\mu,72\epsilon}}(\mathcal{K}_\mu) \leq \frac{\|\mathbf{c}\|_2^2}{72\epsilon}$ since $\lambda_{n_{\mu,72\epsilon}}(\mathcal{K}_\mu) \geq 72\epsilon$ by definition. Finally, note that $\|\mathbf{c}\|_2^2$ is a Chi-squared random variable, with $\mathbb{E}\left[\|\mathbf{c}\|_2^2\right] = 1$. So loosely, by Markov's inequality, with probability $\geq 5/6$, $\|\mathbf{c}\|_2^2 \leq 6$, which gives the claim. □

From Claim 3.6.1 we have:

Claim 3.6.2. *Given random input $y = \mathcal{F}_\mu^* x$ generated as described above, with probability $\geq 5/6$, to solve Problem 3.2.1, an algorithm must return a representation of $\tilde{y}$ with $\left\|y - \tilde{y}\right\|_T^2 \leq \frac{1}{12}$.*

Proof. Solving Problem 3.2.1 requires finding a representation of $\tilde{y}$ with $\left\|y - \tilde{y}\right\|_T^2 \leq \epsilon\|x\|_\mu^2 + C\|n\|_T^2$. By Claim 3.6.1 and the fact that for our input $\|n\|_T^2 = 0$, with probability $\geq 5/6$ one has that $\epsilon\|x\|_\mu^2 + C\|n\|_T^2 \leq \frac{1}{12}$, yielding the claim. □

We next show that finding a $\tilde{y}$ satisfying the condition of Claim 3.6.2 is at least as hard as finding an accurate approximation to $\mathbf{c}$.

Claim 3.6.3. *For $\tilde{y}$ with $\left\|y - \tilde{y}\right\|_T^2 \leq \frac{1}{12}$, $\tilde{\mathbf{c}} = \mathbf{U}^*\tilde{y}$ satisfies $\|\mathbf{c} - \tilde{\mathbf{c}}\|_2^2 \leq \frac{1}{12}$.*

Proof. Recalling that $y = \mathbf{U}\mathbf{c}$, for $\tilde{\mathbf{c}} = \mathbf{U}^*\tilde{y}$ we have:

$$\tilde{\mathbf{c}} = \mathbf{U}^* y + \mathbf{U}^*(\tilde{y} - y) = \mathbf{U}^*\mathbf{U}\mathbf{c} + \mathbf{U}^*(\tilde{y} - y).$$

Recalling that $\mathbf{U}^*\mathbf{U} = \mathbf{I}$ we thus have:

$$\begin{aligned}\|\mathbf{c} - \tilde{\mathbf{c}}\|_2^2 &= \left\|\mathbf{U}^*(\tilde{y} - y)\right\|_2^2 \\ &\leq \left\|\tilde{y} - y\right\|_T^2 \leq \frac{1}{12}.\end{aligned}$$

The second to last inequality follows since $\mathbf{U}^*\mathbf{U} = \mathbf{I}$ and $\mathbf{U}\mathbf{U}^*$ are finite rank, so are compact and share the same non-zero eigenvalues. Thus, $\mathbf{U}\mathbf{U}^* \preceq \mathscr{I}_T$ (Hunter and Nachtergaele, 2001, Lemma 8.26). This completes the claim. □

Combining Claims 3.6.2 and 3.6.3 we have:

Claim 3.6.4. *If a deterministic algorithm solves Problem 3.2.1 with probability* $\geq 2/3$ *over our random input* $y = \mathbf{U}\mathbf{c}$*, then with probability* $\geq 1/2$*, letting* $\tilde{y}$ *be the output of the algorithm,* $\tilde{\mathbf{c}} = \mathbf{U}^*\tilde{y}$ *satisfies* $\|\mathbf{c} - \tilde{\mathbf{c}}\|_2^2 \leq \frac{1}{12}$.

Proof. If an algorithm solves Problem 3.2.1 probability $\geq 2/3$ then by Claim 3.6.2, it returns $\tilde{y}$ with $\left\|y - \tilde{y}\right\|_T^2 \leq \frac{1}{12}$ with probability $\geq 2/3 - 1/6 = 1/2$. Thus, by Claim 3.6.3, $\tilde{\mathbf{c}}$ satisfies $\|\mathbf{c} - \tilde{\mathbf{c}}\|_2^2 \leq \frac{1}{12}$ with probability $\geq 1/2$. □

Finally, we complete the proof of Theorem 3.6.1 by arguing that if $\tilde{y}$ is formed using $o(n_{\mu,72\epsilon})$ queries, then for $\tilde{\mathbf{c}} = \mathbf{U}^*\tilde{y}$, $\|\mathbf{c} - \tilde{\mathbf{c}}\|_2^2 > \frac{1}{12}$ with good probability. Thus the bound of Claim 3.6.4 cannot hold and so $\tilde{y}$ cannot be a solution to Problem 3.2.1 with good probability.

Assume for the sake of contradiction that there is a deterministic algorithm solving Problem 3.2.1 with probability $\geq 2/3$ over the random input $\mathbf{U}\mathbf{c}$ that makes $r = \frac{n_{\mu,72\epsilon}}{20}$ queries on any input (note that if there exists an algorithm that makes fewer queries on some inputs, we can always modify it to make exactly $\frac{n_{\mu,72\epsilon}}{20}$ queries and return the same output.)

As discussed, each query to y is a query to $y(t) = \langle \mathbf{D}^{-1}\mathbf{Z}\varphi_t, \mathbf{c}\rangle$. Consider a deterministic function Q, that is given input $\mathbf{V} \in \mathbb{C}^{i \times n_{\mu,72\epsilon}}$ (for any positive integer i) and outputs $Q(\mathbf{V}) \in \mathbb{C}^{n_{\mu,72\epsilon} \times n_{\mu,72\epsilon}}$ such that $Q(\mathbf{V})$ has orthonormal rows with the first i spanning the i rows of $\mathbf{V}$. For example, Q may run Gram-Schmidt orthogonalization on $\mathbf{V}$ fixing its first $\text{rank}(\mathbf{V}) \leq i$ rows and then fill out the remaining $n_{\mu,72\epsilon} - \text{rank}(\mathbf{V})$ rows using some canonical approach. Letting $\mathbf{D}^{-1}\mathbf{Z}\varphi_{t_1}, \ldots, \mathbf{D}^{-1}\mathbf{Z}\varphi_{t_r}$ denote the queries made by our algorithm on random input $\mathbf{c}$, let $\mathbf{Q}^i = Q([\mathbf{D}^{-1}\mathbf{Z}\varphi_{t_1}, \ldots, \mathbf{D}^{-1}\mathbf{Z}\varphi_{t_i}]^*)$. That is $\mathbf{Q}^i$ is an orthonormal matrix whose first i rows span our first i queries. Note that since our algorithm is deterministic, $\mathbf{Q}^i$ is a deterministic function of the random input $\mathbf{c}$. We have the following claim:

Claim 3.6.5. *Conditioned on the queries* $y(t_1), \ldots y(t_r)$*, for* $j > r$*, each* $[\mathbf{Q}^r\mathbf{c}](j)$ *is distributed independently as* $\mathcal{N}\left(0, \frac{1}{n_{\mu,72\epsilon}}\right)$.

Proof. We prove the claim via induction on the number of queries considered. For the base case set $i = 1$. $\mathbf{Q}^1$ is a deterministic matrix (since the choice of our first query is made de-

terminstically before seeing any input) and so by the rotational invariance of the Gaussian distribution, the entries of $\mathbf{Q}^1\mathbf{c}$ are distributed independently as $\mathcal{N}\left(0, \frac{1}{n_{\mu,72\epsilon}}\right)$ (the same as the entries of $\mathbf{c}$). The first row of $\mathbf{Q}^1$ spans our first query, and thus this row is just equal to $\mathbf{D}^{-1}\mathbf{Z}\varphi_{t_1}$ scaled to have unit norm. Thus $y(t_1) = \mathbf{D}^{-1}\mathbf{Z}\varphi_{t_1}\mathbf{c}$ is just a fixed scaling of $[\mathbf{Q}^1\mathbf{c}](1)$. So conditioning on $y(t_1)$, we still have $[\mathbf{Q}^1\mathbf{c}](j)$ for $j > 1$ distributed independently as $\mathcal{N}\left(0, \frac{1}{n_{\mu,72\epsilon}}\right)$.

Now, consider $i > 1$. By the inductive assumption, conditioned on $y(t_1), \ldots y(t_{i-1})$, for $j \geq i$, $[\mathbf{Q}^{i-1}\mathbf{c}](j)$, are distributed independently as $\mathcal{N}\left(0, \frac{1}{n_{\mu,72\epsilon}}\right)$. We can see that both $\mathbf{Q}^{i-1}$ and $\mathbf{Q}^i$ are fixed conditioned on $y(t_1), \ldots y(t_{i-1})$ (since the i^{th} query is chosen deterministically, possibly adaptively as a function of the previously seen queries $y(t_1), \ldots y(t_{i-1})$). Additionally, since they share their first $i-1$ rows, the remaining $n_{\mu,72\epsilon} - i + 1$ rows of $\mathbf{Q}^{i-1}$ and $\mathbf{Q}^i$ have the same rowspans. Thus we can write $\mathbf{Q}^i = [\mathbf{I}; \mathbf{R}]\mathbf{Q}^{i-1}$ where $\mathbf{R} \in \mathbb{C}^{n_{\mu,72\epsilon}-i+1 \times n_{\mu,72\epsilon}-i+1}$ is some fixed rotation with $\mathbf{R}^*\mathbf{R} = \mathbf{I}$. Thus, by the rotational invariance of the Gaussian, for all $j \geq i$, $[\mathbf{Q}^i\mathbf{c}](j)$ are distributed independently as $\mathcal{N}\left(0, \frac{1}{n_{\mu,72\epsilon}}\right)$ (the same as $[\mathbf{Q}^{i-1}\mathbf{c}](j)$). Further conditioning on $y(t_i)$, which is a deterministic function of $[\mathbf{Q}^i\mathbf{c}](i)$ and $y(t_1)\ldots y(t_{i-1})$, we still have that for $j > i$, $[\mathbf{Q}^i\mathbf{c}](j)$ are distributed independently as $\mathcal{N}\left(0, \frac{1}{n_{\mu,72\epsilon}}\right)$. This completes the inductive step and so the claim. □

Armed with Claim 3.6.5 we can compute:

$$\begin{aligned}
\Pr\left[\|\mathbf{c} - \tilde{\mathbf{c}}\|_2^2 \leq \frac{1}{12}\right] &= \Pr\left[\left\|\mathbf{Q}^r\mathbf{c} - \mathbf{Q}^r\tilde{\mathbf{c}}\right\|_2^2 \leq \frac{1}{12}\right] && \text{(Since } \mathbf{Q}^r \text{ is orthonormal.)} \\
&\leq \Pr\left[\sum_{i=r+1}^{n_{\mu,72\epsilon}} \left|[\mathbf{Q}^r\mathbf{c}](i) - [\mathbf{Q}^r\tilde{\mathbf{c}}](i)\right|^2 \leq \frac{1}{12}\right] \\
&= \mathbb{E}_{y(t_1),\ldots,y(t_r)}\left[\Pr\left[\sum_{i=r+1}^{n_{\mu,72\epsilon}} \left|[\mathbf{Q}^r\mathbf{c}](i) - [\mathbf{Q}^r\tilde{\mathbf{c}}](i)\right|^2 \leq \frac{1}{12} \,\middle|\, y(t_1)\ldots, y(t_r)\right]\right] \\
&\leq \mathbb{E}_{y(t_1),\ldots,y(t_r)}\left[\Pr\left[\sum_{i=r+1}^{n_{\mu,72\epsilon}} \left|[\mathbf{Q}^r\mathbf{c}](i)\right|^2 \leq \frac{1}{12} \,\middle|\, y(t_1)\ldots, y(t_r)\right]\right] && (3.37)
\end{aligned}$$

where the last line follows since, conditioned on $y(t_1)\ldots, y(t_r)$, $\mathbf{Q}^r\tilde{\mathbf{c}}$ is fixed and for $i \geq r+1$, $\mathbf{Q}^r\mathbf{c}(i)$ are distributed independently as Gaussians centered around 0 (by Claim 3.6.5). So the probability of the sum of differences being small is only smaller than if we replaced each $\mathbf{Q}^r\tilde{\mathbf{c}}(i)$ by 0.

Now, conditioned on $y(t_1)\ldots, y(t_r)$, $\sum_{i=r+1}^{n_{\mu,72\epsilon}} |[\mathbf{Q}^r\mathbf{c}](i)|^2$ is a Chi-squared random variable with

$$\mathbb{E}\left[\sum_{i=r+1}^{n_{\mu,72\epsilon}} \left|[\mathbf{Q}^r\mathbf{c}](i)\right|^2 \,\middle|\, y(t_1)\ldots, y(t_r)\right] = \frac{n_{\mu,72\epsilon} - r}{n_{\mu,72\epsilon}}.$$

For $r = \frac{n_{\mu,72\epsilon}}{20}$, we thus have $\mathbb{E}\left[\sum_{i=r+1}^{n_{\mu,72\epsilon}} |[\mathbf{Q}^r\mathbf{c}](i)|^2 \mid y(t_1)\ldots, y(t_r)\right] \geq \frac{19}{20}$. We can loosely upper bound the probability in (3.37), using the fact that for a Chi-squared random variable X with

k degrees of freedom, $\Pr[X \le \delta \mathbb{E}[X]] \le \left(\delta e^{1-\delta}\right)^{k/2} \le \left(\delta e^{1-\delta}\right)^{1/2}$. So,

$$\Pr\left[\sum_{i=k(\mathbf{c})+1}^{n_{\mu,72\epsilon}} \left|[\mathbf{Q}^r\mathbf{c}](i)\right|^2 \le \frac{1}{12}\middle| y(t_1)..., y(t_r)\right] \le \left(\frac{20}{19\cdot 12}e^{1-\frac{20}{19\cdot 12}}\right)^{1/2} < \frac{47}{100}.$$

Plugging back into (3.37) gives:

$$\Pr\left[\|\mathbf{c}-\tilde{\mathbf{c}}\|_2^2 \le \frac{1}{12}\right] \le \mathbb{E}_{y(t_1),...,y(t_r)}\left[\Pr\left[\sum_{i=r+1}^{n_{\mu,72\epsilon}} \left|[\mathbf{Q}^r\mathbf{c}](i)\right|^2 \le \frac{1}{12}\middle| y(t_1)..., y(t_r)\right]\right] < \frac{47}{100}.$$

However, we have assumed that our algorithm solves Problem 3.2.1 with probability $\ge 2/3$, and hence, by Claim 3.6.4, $\Pr\left[\|\mathbf{c}-\tilde{\mathbf{c}}\|_2^2 \le \frac{1}{12}\right] \ge \frac{1}{2}$. This is a contradiction, yielding the theorem.

□

3.6.1 Statistical Dimension Lower Bound

We now use Theorem 3.6.1 to prove that the statistical dimension tightly characterizes the sample complexity of solving Problem 3.2.1 for any constraint measure μ satisfying a simple condition: we must have $s_{\mu,\epsilon} = O(1/\epsilon^p)$ for some $p < 1$. Note that this assumption holds for all μ considered in this work (including bandlimited, multiband, sparse, Gaussian, and Cauchy-Lorentz), where $s_{\mu,\epsilon}$ either grows as $\log(1/\epsilon)$ or $1/\sqrt{\epsilon}$. Also note that by (3.5) we can always bound $s_{\mu,\epsilon} \le \mathrm{tr}(\mathcal{K}_\mu)/\epsilon = 1/\epsilon$. So this assumption holds whenever we have a nontrivial upper bound on $s_{\mu,\epsilon}$.

Theorem 3.6.2 (Statistical Dimension Lower Bound)**.** *For any probability measure μ, suppose that $s_{\mu,\epsilon} = O(1/\epsilon^p)$ for some constant $p < 1$. Consider any (possibly randomized) algorithm that solves Problem 3.2.1 with probability $\ge 2/3$ for any function y and any $\epsilon > 0$ and makes at most $r_{\mu,\epsilon}$ (possibly adaptive) queries on any input. Then $r_{\mu,\epsilon} = \Omega(s_{\mu,\epsilon})$.*[22]

Proof. We simply prove that for this class of measures, $n_{\mu,72\epsilon} = \Omega(s_{\mu,\epsilon})$ and then apply Theorem 3.6.1. It suffices to show that $n_{\mu,\epsilon} = \Omega(s_{\mu,c\epsilon})$ for any fixed constant $c \ge 1$ since by (3.17), $s_{\mu,c\epsilon} \ge \frac{s_{\mu,\epsilon}}{c}$. Thus $n_{\mu,\epsilon} = \Omega(s_{\mu,c\epsilon})$ gives that $n_{\mu,72\epsilon} = \Omega(s_{\mu,72c\epsilon}) = \Omega(s_{\mu,\epsilon})$, giving the theorem.

Let $c_p = 2^{\frac{4}{1-p}} > 1$. Assume for the sake of contradiction that $n_{\mu,\epsilon} = o(s_{\mu,c_p\epsilon})$. By this assumption, there is some fixed ϵ_0 such that,

$$\text{For all } \epsilon \le \epsilon_0\,,\, n_{\mu,\epsilon} \le \frac{s_{\mu,c_p\epsilon}}{2}. \tag{3.38}$$

[22] Here we follow the Hardy-Littlewood definition (Hardy et al., 1914), using $f(\epsilon) = \Omega(g(\epsilon))$ to denote that $\limsup_{x\to\infty} \frac{f(\epsilon)}{g(\epsilon)} > 0$. Thus the lower bound shows that, for some fixed constant $c > 0$, there is at least some ϵ' such that for every $\epsilon \le \epsilon'$, the number of queries used by any algorithm solving Problem 3.2.1 with probability $\ge 2/3$ is at least $c \cdot s_{\mu,\epsilon}$. In other words, the lower bound rules out the possibility that the number of queries is $o(s_{\mu,\epsilon})$.

We can bound:

$$s_{\mu,c_p\epsilon} = \sum_{i=1}^{\infty} \frac{\lambda_i(\mathcal{K}_\mu)}{\lambda_i(\mathcal{K}_\mu) + c_p\epsilon} \leq n_{\mu,\epsilon} + \sum_{i=n_{\mu,\epsilon}+1}^{\infty} \frac{\lambda_i(\mathcal{K}_\mu)}{c_p\epsilon}$$

and thus by (3.38) have for any $\epsilon \leq \epsilon_0$:

$$\frac{1}{2} \cdot s_{\mu,c_p\epsilon} \leq \sum_{i=n_{\mu,\epsilon}+1}^{\infty} \frac{\lambda_i(\mathcal{K}_\mu)}{c_p\epsilon}. \tag{3.39}$$

Now we also have:

$$\begin{aligned} s_{\mu,\epsilon} = \sum_{i=1}^{\infty} \frac{\lambda_i(\mathcal{K}_\mu)}{\lambda_i(\mathcal{K}_\mu) + \epsilon} &\geq \sum_{i=n_{\mu,\epsilon}+1}^{\infty} \frac{\lambda_i(\mathcal{K}_\mu)}{\lambda_i(\mathcal{K}_\mu) + \epsilon} \\ &\geq \sum_{i=n_{\mu,\epsilon}+1}^{\infty} \frac{\lambda_i(\mathcal{K}_\mu)}{2\epsilon} \\ &= \frac{c_p}{2} \cdot \sum_{i=n_{\mu,\epsilon}+1}^{\infty} \frac{\lambda_i(\mathcal{K}_\mu)}{c_p\epsilon}. \end{aligned}$$

Combined with (3.39) this gives that for any $\epsilon \leq \epsilon_0$:

$$s_{\mu,\epsilon} \geq \frac{c_p}{4} \cdot s_{\mu,c_p\epsilon}. \tag{3.40}$$

By (3.40) we in turn have that, for every $\epsilon \leq \epsilon_0$,

$$s_{\mu,\epsilon} \geq s_{\mu,\epsilon_0} \cdot \left(\frac{c_p}{4}\right)^{\lfloor \log_{c_p} \epsilon_0/\epsilon \rfloor}.$$

Using that $\lfloor \log_{c_p} \epsilon_0/\epsilon \rfloor \geq \log_{c_p} \epsilon_0/\epsilon - 1$ and that $c_p = 2^{\frac{4}{1-p}} \geq 16$ we can then bound, for all $\epsilon \leq \epsilon_0$:

$$\begin{aligned} \frac{s_{\mu,\epsilon}}{s_{\mu,\epsilon_0}} \geq \left(\frac{c_p}{4}\right)^{\log_{c_p}\epsilon_0 - \log_{c_p}\epsilon - 1} &= \left(\frac{c_p}{4}\right)^{\log_{c_p}\epsilon_0 - 1} \cdot c_p^{\log_{c_p} 1/\epsilon} \cdot \left(\frac{1}{4}\right)^{\log_{c_p} 1/\epsilon} \\ &\geq \left(\frac{c_p}{4}\right)^{\log_{c_p}\epsilon_0 - 1} \cdot \frac{1}{\epsilon} \cdot \epsilon^{\frac{1-p}{2}} \\ &\geq \left(\frac{c_p}{4}\right)^{\log_{c_p}\epsilon_0 - 1} \cdot \frac{1}{\epsilon^{p + \frac{1-p}{2}}}. \end{aligned}$$

Note that $s_{\mu,\epsilon_0} \cdot \left(\frac{c_p}{4}\right)^{\log_{c_p}\epsilon_0 - 1}$ is a constant independent of ϵ. Thus, the above contradicts the assumption that $s_{\mu,\epsilon} = O(1/\epsilon^p)$, giving the theorem. □

Remark We remark that a similar technique to Theorem 3.6.2 can be used to show that $n_{\mu,\epsilon} = \Omega(s_{\mu,\epsilon}/\epsilon^p)$ for any $p > 0$, without any assumptions on $s_{\mu,\epsilon}$.

4 Modified Random Fourier Features for Kernel Ridge Regression

This chapter is based on a joint work with Haim Avron, Michael Kapralov, Cameron Musco, Christopher Musco, and Ameya Velingker. It has been accepted to the 34th International Conference on Machine Learning (Avron et al., 2017c, ICML).

4.1 Introduction

Kernel methods constitute a powerful paradigm for devising non-parametric modeling techniques for a wide range of problems in machine learning. One of the most elementary is *Kernel Ridge Regression (KRR)*. Given training data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathscr{X} \times \mathscr{Y}$, where $\mathscr{X} \subseteq \mathbb{R}^d$ is an input domain and $\mathscr{Y} \subseteq \mathbb{R}$ is an output domain, a positive definite kernel function $k : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$, and a regularization parameter $\lambda > 0$, the response for a given input $\mathbf{x}$ is estimated as:

$$\bar{f}(\mathbf{x}) \equiv \sum_{j=1}^{n} k(\mathbf{x}_j, \mathbf{x}) \alpha_j$$

where $\boldsymbol{\alpha} = (\alpha_1, \cdots \alpha_n)^\top$ is the solution of the equation

$$(\mathbf{K} + \lambda \mathbf{I}_n)\boldsymbol{\alpha} = \mathbf{y}. \tag{4.1}$$

In the above, $\mathbf{K} \in \mathbb{R}^{n \times n}$ is the *kernel matrix* or *Gram matrix* defined by $\mathbf{K}_{ij} \equiv k(\mathbf{x}_i, \mathbf{x}_j)$ and $\mathbf{y} \equiv (y_1, \cdots y_n)^\top$ is the vector of responses. The KRR estimator can be derived by minimizing a regularized square loss objective function over a hypothesis space defined by the reproducing kernel Hilbert space associated with $k(\cdot, \cdot)$; however, the details are not important for this chapter.

While simple, KRR is a powerful technique that is well understood statistically and capable of achieving impressive empirical results. Nevertheless, the method has a key weakness: computing the KRR estimator can be prohibitively expensive for large datasets. Solving (4.1) generally requires $\Theta(n^3)$ time[1] and $\Theta(n^2)$ memory. Thus, the design of scalable methods for

[1] The running time can be improved using fast matrix products.

KRR (and other kernel based methods) has been the focus of intensive research in recent years (Zhang et al., 2013; Alaoui and Mahoney, 2015; Musco and Musco, 2017; Avron et al., 2017a; Kapralov et al., 2020).

One of the most popular approaches to scaling up kernel based methods is random Fourier features sampling, originally proposed by Rahimi and Recht (2008). For shift-invariant kernels (e.g. the Gaussian kernel), Rahimi and Recht (2008) presented a distribution D on functions from $\mathscr{X}$ to $\mathbb{C}^s$ (s is a parameter) such that for every $\mathbf{x}, \mathbf{z} \in \mathbb{R}^d$,

$$k(\mathbf{x}, \mathbf{z}) = \mathbb{E}_{\varphi \sim D}\left[\varphi(\mathbf{x})^* \varphi(\mathbf{z})\right].$$

The random features approach is then to sample a φ from D and use $\tilde{k}(\mathbf{x}, \mathbf{z}) \equiv \varphi(\mathbf{x})^* \varphi(\mathbf{z})$ as a surrogate kernel. The resulting approximate KRR estimator can be computed in $O(ns^2)$ time and $O(ns)$ memory (see Section 4.2.2), giving substantial computational savings if $s \ll n$.

This approach naturally raises the question: how large should s be to ensure a high quality estimator? Or, using the exact KRR estimator as a natural baseline: how large should s be for the random Fourier features estimator to be almost as good as the exact KRR estimator? Answering this question can help us determine when random Fourier features can be useful, whether the method needs to be improved, and how to go about improving it.

The original analysis of Rahimi and Recht (2008) bounds the point-wise distance between $k(\cdot,\cdot)$ and $\tilde{k}(\cdot,\cdot)$ (for other approaches to analyzing random Fourier features, see Section 4.2.3). However, the bounds do not naturally lead to an answer to the aforementioned question. In contrast, spectral approximation bounds on the entire surrogate kernel matrix, i.e. of the form

$$(1-\Delta)(\mathbf{K} + \lambda \mathbf{I}_n) \preceq \tilde{\mathbf{K}} + \lambda \mathbf{I}_n \preceq (1+\Delta)(\mathbf{K} + \lambda \mathbf{I}_n), \tag{4.2}$$

naturally have statistical and algorithmic implications. Indeed, in Section 4.3 we show that when (4.2) holds we can bound the excess risk introduced by the random Fourier features estimator compared to the KRR estimator. We also show that $\tilde{\mathbf{K}} + \lambda \mathbf{I}_n$ can be used as an effective preconditioner for the solution of (4.1). This motivates the study of how large s should be as a function of Δ for (4.2) to hold.

In this chapter we rigorously analyze the relation between the number of random Fourier features and the spectral approximation bound (4.2). Our main results are the following:

- We give an upper bound on the number of random features needed to achieve (4.2) (Theorem 4.4.1). This bound, in conjunction with the results in Section 4.3, positively shows that random Fourier features can give guarantees for KRR under reasonable assumptions.

- We give a lower bound showing that our upper bound is tight for the Gaussian kernel (Theorem 4.5.1).

- We show that the upper bound can be improved dramatically by modifying the sampling distribution used in the classical random Fourier features (Section 4.4). Our sampling distribution is based on an appropriately defined *leverage function* of the kernel, closely related to so-called leverage scores frequently encountered in the analysis of sampling based methods for linear regression. Unfortunately, it is unclear how to efficiently sample using the leverage function.

- To address the lack of an efficient way to sample using the leverage function, we propose a novel, easy-to-sample distribution for the Gaussian kernel which approximates the true leverage function distribution and allows random Fourier features to achieve a significantly improved upper bound (Theorem 4.6.1). The upper bound has an exponential dependence on the data dimension, so it is only applicable to low dimensional datasets. Nevertheless, our results demonstrate that the classic random Fourier sampling distribution can be improved for spectral approximation and motivates further study. As an application, our improved understanding of the leverage function yields a novel asymptotic bound on the statistical dimension of Gaussian kernel matrices over bounded datasets, which may be of independent interest (Corollary 4.7.1).

4.2 Preliminaries

4.2.1 Setup and Notation

The complex conjugate of $x \in \mathbb{C}$ is denoted by x^*. For a vector $\mathbf{x}$ or a matrix $\mathbf{A}$, $\mathbf{x}^*$ or $\mathbf{A}^*$ denotes the Hermitian transpose. The $l \times l$ identity matrix is denoted $\mathbf{I}_l$. We use the convention that vectors are column-vectors.
A Hermitian matrix $\mathbf{A}$ is positive semidefinite (PSD) if $\mathbf{x}^*\mathbf{A}\mathbf{x} \geq 0$ for every vector $\mathbf{x}$. For any two Hermitian matrices $\mathbf{A}$ and $\mathbf{B}$ of the same size, $\mathbf{A} \preceq \mathbf{B}$ means that $\mathbf{B} - \mathbf{A}$ is PSD.
We use $L_2(\rho) = L_2(\mathbb{R}^d, d\rho)$ to denote the space of complex-valued square-integrable functions with respect to some measure $\rho(\cdot)$. $L_2(\rho)$ is a Hilbert space equipped with the following inner product:

$$\langle f, g \rangle_{L_2(\rho)} = \int_{\mathbb{R}^d} f(\boldsymbol{\eta}) g(\boldsymbol{\eta})^* d\rho(\boldsymbol{\eta}) = \int_{\mathbb{R}^d} f(\boldsymbol{\eta}) g(\boldsymbol{\eta})^* p_\rho(\boldsymbol{\eta}) d\boldsymbol{\eta}.$$

In the above, $p_\rho(\cdot)$ is the probability density induced by $\rho(\cdot)$ (assuming one exists).

We denote the training set by $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathcal{X} \times \mathcal{Y} \subseteq \mathbb{R}^d \times \mathbb{R}$. Note that n denotes the number of training examples, and d their dimension. We denote the kernel, which is a function from $\mathcal{X} \times \mathcal{X}$ to $\mathbb{R}$, by k. We denote the kernel matrix by $\mathbf{K}$, with $\mathbf{K}_{ij} \equiv k(\mathbf{x}_i, \mathbf{x}_j)$. The associated reproducing kernel Hilbert space (RKHS) is denoted by $\mathscr{H}_k$, and the associated inner product by $\langle \cdot, \cdot \rangle_{\mathscr{H}_k}$. Some results are stated for the Gaussian kernel $k(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\|_2^2 / 2\sigma^2)$ for some bandwidth parameter σ.

We use λ to denote the ridge regularization parameter. We remark that the choice of regular-

ization parameter generally depends on n. Typically, $\lambda = \omega(1)$ and $\lambda = o(n)$. See Caponnetto and De Vito (2007) and Bach (2013) for discussion on the asymptotic behavior of λ, noting that in our notation, λ is scaled by an n factor as compared to those works. As the ratio between n and λ will be an important quantity in our bounds, we denote it as $n_\lambda \stackrel{\text{def}}{=} n/\lambda$.
The *statistical dimension* or *effective degrees of freedom* given the regularization parameter λ is denoted by $s_\lambda(\mathbf{K}) \stackrel{\text{def}}{=} \operatorname{tr}\left((\mathbf{K}+\lambda\mathbf{I}_n)^{-1}\mathbf{K}\right)$.

4.2.2 Random Fourier features

Classical random Fourier features

Random Fourier features (Rahimi and Recht, 2008) is an approach to scaling up kernel methods for shift-invariant kernels. A shift-invariant kernel is a kernel of the form $k(\mathbf{x},\mathbf{z}) = k(\mathbf{x}-\mathbf{z})$ where $k(\cdot)$ is a positive definite function (we abuse notation by using k to denote both the kernel and the defining positive definite function).

The underlying observation behind random Fourier features is a simple consequence of Bochner's Theorem: for every shift-invariant kernel with $k(\mathbf{0}) = 1$, there is a probability measure $\mu_k(\cdot)$ which induces a probability density function $p_k(\cdot)$, both on $\mathbb{R}^d$, such that

$$k(\mathbf{x},\mathbf{z}) = \int_{\mathbb{R}^d} e^{-2\pi i \boldsymbol{\eta}^\top(\mathbf{x}-\mathbf{z})} d\mu_k(\boldsymbol{\eta}) = \int_{\mathbb{R}^d} e^{-2\pi i \boldsymbol{\eta}^\top(\mathbf{x}-\mathbf{z})} p_k(\boldsymbol{\eta}) d\boldsymbol{\eta}\,. \tag{4.3}$$

In other words, the Fourier transform of the kernel $k(\cdot)$ is a probability density function, $p_k(\cdot)$. For simplicity we typically drop the k subscript, writing $\mu(\cdot) = \mu_k(\cdot)$ and $p(\cdot) = p_k(\cdot)$, with the associated kernel function clear from context. We remark that while it is not always the case that the probability measure $\mu_k(\cdot)$ has an associated density function $p_k(\cdot)$, we assume the existence of a density function for the kernels we consider.

If $\boldsymbol{\eta}_1,\ldots,\boldsymbol{\eta}_s$ are drawn according to $p(\cdot)$, and we define $\varphi(\mathbf{x}) \stackrel{\text{def}}{=} \frac{1}{\sqrt{s}}\left(e^{-2\pi i \boldsymbol{\eta}_1^T\mathbf{x}},\cdots,e^{-2\pi i \boldsymbol{\eta}_s^T\mathbf{x}}\right)^*$, then:

$$k(\mathbf{x},\mathbf{z}) = \mathbb{E}_\varphi\left[\varphi(\mathbf{x})^*\varphi(\mathbf{z})\right].$$

The idea of random Fourier features method is then to define the substitute kernel:

$$\tilde{k}(\mathbf{x},\mathbf{z}) \equiv \varphi(\mathbf{x})^*\varphi(\mathbf{z}) = \frac{1}{s}\sum_{l=1}^{s} e^{-2\pi i \boldsymbol{\eta}_l^\top(\mathbf{x}-\mathbf{z})} \tag{4.4}$$

To summarize, the random Fourier features method approximates $k(\cdot)$ by sampling s frequencies $\boldsymbol{\eta}_1,\ldots,\boldsymbol{\eta}_s \in \mathbb{R}^d$ according to their weight in the density function $p(\cdot)$ which is just the d-dimensional Fourier transform of the kernel $k(\cdot)$. Note that in order for $p(\cdot)$ to be a proper probability density function (integrating to 1) we must have $k(\mathbf{0}) = 1$. We assume this without loss of generality, since any kernel can be scaled to satisfy this condition.

Now suppose that $\mathbf{Z} \in \mathbb{C}^{n\times s}$ is the matrix whose j^{th} row is $\varphi(\mathbf{x}_j)^*$, and let $\tilde{\mathbf{K}} = \mathbf{Z}\mathbf{Z}^*$. $\tilde{\mathbf{K}}$ is the

kernel matrix corresponding to $\tilde{k}(\cdot,\cdot)$. The resulting random Fourier features KRR estimator is $\tilde{f}(\mathbf{x}) \equiv \sum_{j=1}^{n} \tilde{k}(\mathbf{x}_j,\mathbf{x})\tilde{\alpha}_j$ where $\tilde{\boldsymbol{\alpha}}$ is the solution of $(\tilde{\mathbf{K}}+\lambda\mathbf{I}_n)\tilde{\boldsymbol{\alpha}} = \mathbf{y}$. Typically, $s < n$ and we can represent $\tilde{f}(\cdot)$ more efficiently as:

$$\tilde{f}(\mathbf{x}) = \varphi(\mathbf{x})^*\mathbf{w},$$

where $\mathbf{w} = (\mathbf{Z}^*\mathbf{Z}+\lambda\mathbf{I}_s)^{-1}\mathbf{Z}^*\mathbf{y}$ (a simple consequence of the Woodbury formula). We can compute $\mathbf{w}$ in $O(ns^2)$ time, making random Fourier features computationally attractive if $s = o(n)$.

Modified random Fourier features

While it seems to be a natural choice, there is no fundamental reason that we must sample the frequencies $\boldsymbol{\eta}_1,\ldots,\boldsymbol{\eta}_s$ using the Fourier transform density function $p(\cdot)$. In fact, we will see that it is advantageous to use a different sampling distribution based on the kernel leverage function (defined later).

Let $q(\cdot)$ be any probability density function whose support includes that of $p(\cdot)$. If we sample $\boldsymbol{\eta}_1,\ldots,\boldsymbol{\eta}_s$ using $q(\cdot)$, and define

$$\varphi(\mathbf{x}) \stackrel{\text{def}}{=} \frac{1}{\sqrt{s}}\left(\sqrt{\frac{p(\boldsymbol{\eta}_1)}{q(\boldsymbol{\eta}_1)}}e^{-2\pi i\boldsymbol{\eta}_1^T\mathbf{x}},\cdots,\sqrt{\frac{p(\boldsymbol{\eta}_s)}{q(\boldsymbol{\eta}_s)}}e^{-2\pi i\boldsymbol{\eta}_s^T\mathbf{x}}\right)^*$$

we still have $k(\mathbf{x},\mathbf{z}) = \mathbb{E}_\varphi\left[\varphi(\mathbf{x})^*\varphi(\mathbf{z})\right]$. We refer to this method as *modified random Fourier features* and remark that it can be viewed as a form of importance sampling.

Additional Notations and Identities

With the definition of (modified) random Fourier features in hand, we can introduce additional notation and identities. The (j,l) entry of $\mathbf{Z}$ from previous section is given by:

$$\mathbf{Z}_{jl} = \frac{1}{\sqrt{s}}e^{-2\pi i\mathbf{x}_j^\top\boldsymbol{\eta}_l}\sqrt{p(\boldsymbol{\eta}_l)/q(\boldsymbol{\eta}_l)}. \tag{4.5}$$

Let $\mathbf{z}:\mathbb{R}^d\to\mathbb{C}^n$ be defined by

$$\mathbf{z}(\boldsymbol{\eta})_j = e^{-2\pi i\mathbf{x}_j^\top\boldsymbol{\eta}}.$$

Note that column l of $\mathbf{Z}$ is exactly $\mathbf{z}(\boldsymbol{\eta}_l)\sqrt{p(\boldsymbol{\eta}_l)/[s\cdot q(\boldsymbol{\eta}_l)]}$. So we have:

$$\mathbf{Z}\mathbf{Z}^* = \frac{1}{s}\sum_{l=1}^{s}\frac{p(\boldsymbol{\eta}_l)}{q(\boldsymbol{\eta}_l)}\mathbf{z}(\boldsymbol{\eta}_l)\mathbf{z}(\boldsymbol{\eta}_l)^*.$$

Finally, by (4.3),

$$\mathbf{K} = \int_{\mathbb{R}^d}\mathbf{z}(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*\,d\mu(\boldsymbol{\eta}) = \int_{\mathbb{R}^d}\mathbf{z}(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*\,p(\boldsymbol{\eta})\,d\boldsymbol{\eta}.$$

Therefore, $\mathbb{E}[\mathbf{Z}\mathbf{Z}^*] = \mathbf{K}$.

4.2.3 Related Work

The original analysis of random Fourier features (Rahimi and Recht, 2008) bounded the pointwise distance between $k(\cdot,\cdot)$ and $\tilde{k}(\cdot,\cdot)$. In follow-up work, Rahimi and Recht (2009) give learning rate bounds for a broad class of estimators using random Fourier features. However, their results do not apply to classic KRR. Furthermore, their main bound becomes relevant only when the number of sampled features is on the order of the training set size.

Rudi and Rosasco (2017) prove generalization properties for KRR with random features, under somewhat difficult to verify technical assumptions, some of which can be seen as constraining the leverage function distribution that we study. They leave open improving their bounds via a more refined sampling approach. Bach (2017) analyzes random Fourier features from a function approximation point of view. He defines a similar leverage function distribution to the one that we consider, but leaves open establishing bounds on and effectively sampling from this distribution, both of which we address in this work. Finally, Tropp et al. (2015) analyzes the distance between the kernel matrix and its approximation in terms of the spectral norm, $\left\|\mathbf{K}-\tilde{\mathbf{K}}\right\|_{\text{op}}$, which can be a significantly weaker error metric than (4.2).

Outside of work on random Fourier features, risk inflation bounds for approximate KRR and leverage score sampling have been used to analyze and improve the Nyström method for kernel approximation (Bach, 2013; Alaoui and Mahoney, 2015; Rudi et al., 2015; Musco and Musco, 2017). We apply a number of techniques from this line of work.

Spectral approximation bounds, such as (4.2), are quite popular in the sketching literature; see Woodruff's survey (Woodruff, 2014). Most closely related to our work is analysis of spectral approximation bounds without regularization (i.e. $\lambda = 0$) for the polynomial kernel (Avron et al., 2014). Improved bounds with regularization (still for the polynomial kernel) were recently proved by Avron et al. (2017a) and Ahle et al. (2020).

4.3 Spectral Bounds and Statistical Guarantees

Given a feature transformation, like random Fourier features, how do we analyze it and relate its use to non-approximate methods? A common approach, taken for example in the original paper on random Fourier features (Rahimi and Recht, 2008), is to bound the difference between the true kernel $k(\cdot,\cdot)$ and the approximate kernel $\tilde{k}(\cdot,\cdot)$. However, it is unclear how such bounds translate to downstream guarantees on statistical learning methods, such as KRR. In this paper we advocate and focus on spectral approximation bounds on the regularized kernel matrix, specifically, bounds of the form,

$$(1-\Delta)(\mathbf{K}+\lambda\mathbf{I}_n) \preceq \mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n \preceq (1+\Delta)(\mathbf{K}+\lambda\mathbf{I}_n) \tag{4.6}$$

for some $\Delta < 1$.

Definition 4.3.1. Matrix $\mathbf{A}$ is a Δ-*spectral approximation* of matrix $\mathbf{B}$, if $(1-\Delta)\mathbf{B} \preceq \mathbf{A} \preceq (1+\Delta)\mathbf{B}$.

Remark 4.3.1. *When $\lambda = 0$, bound of* (4.6) *can be viewed as a low-distortion subspace embedding. Indeed, when $\lambda = 0$ it follows from* (4.6) *that* $\mathbf{Span}\left(k(\mathbf{x}_1,\cdot),\ldots,k(\mathbf{x}_n,\cdot)\right) \subseteq \mathscr{H}_k$ *can be embedded with Δ-distortion in* $\mathbf{Span}\left(\varphi(\mathbf{x}_1),\ldots,\varphi(\mathbf{x}_n)\right) \subseteq \mathbb{R}^s$.

The main mathematical question we seek to address is: when using random Fourier features, how large should s be in order to guarantee that $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is a Δ-spectral approximation of $\mathbf{K} + \lambda\mathbf{I}_n$? To motivate this question, in the following two subsections we show that such bounds can be used to derive risk inflation bounds for approximate kernel ridge regression. We also show that they can be used to analyze the use of $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ as a preconditioner for $\mathbf{K} + \lambda\mathbf{I}_n$.

While this chapter focuses on KRR for conciseness, we remark that in the sketching literature, spectral approximation bounds also form the basis for analyzing sketching based methods for tasks like low-rank approximation, k-means and more. In the kernel setting, such applications where analyzed, without regularization, for the polynomial kernel (Avron et al., 2014). Cohen et al. (2017) recently showed that (4.6) along with a trace condition on $\mathbf{Z}\mathbf{Z}^*$ (which holds for all sampling approaches we consider) yields a so called "projection-cost preservation" condition for the kernel approximation. With λ chosen appropriately, this condition ensures that $\mathbf{Z}\mathbf{Z}^*$ can be used in place of $\mathbf{K}$ for approximately solving kernel k-means clustering and for certain versions of kernel PCA and kernel CCA. See Musco and Musco (2017) for details, where this analysis is carried out for the Nyström method.

4.3.1 Risk Bounds

One way to analyze estimators is via risk bounds; several recent papers on approximate KRR employ such an analysis (Bach, 2013; Alaoui and Mahoney, 2015; Musco and Musco, 2017). In particular, these papers consider the fixed design setting and seek to bound the expected in-sample predication error of the KRR estimator $\bar{f}$, viewing it as an empirical estimate of the statistical risk. More specifically, the underlying assumption is that y_i satisfies

$$y_i = f^\star(\mathbf{x}_i) + \nu_i \tag{4.7}$$

for some $f^\star : \mathscr{X} \to \mathbb{R}$. The $\{\nu_i\}$'s are i.i.d noise terms, distributed as normal variables with variance σ_ν^2. The empirical risk of an estimator f, which can be viewed as a measure of the quality of the estimator, is

$$\mathscr{R}(f) \equiv \mathbb{E}_{\{\nu_i\}}\left[\frac{1}{n}\sum_{j=1}^{n}\left|f(\mathbf{x}_i) - f^\star(\mathbf{x}_i)\right|^2\right]$$

(note that f itself might be a function of $\{\nu_i\}$).

Let $\mathbf{f} \in \mathbb{R}^n$ be the vector whose j^{th} entry is $f^\star(\mathbf{x}_j)$. It is quite straightforward to show that for the KRR estimator $\bar{f}$ we have (Bach, 2013; Alaoui and Mahoney, 2015):

$$\mathscr{R}(\bar{f}) = n^{-1}\lambda^2\mathbf{f}^\top(\mathbf{K} + \lambda\mathbf{I}_n)^{-2}\mathbf{f} + n^{-1}\sigma_\nu^2\mathrm{tr}\left(\mathbf{K}^2(\mathbf{K} + \lambda\mathbf{I}_n)^{-2}\right).$$

Since $\lambda^2 \mathbf{f}^\top (\mathbf{K}+\lambda \mathbf{I}_n)^{-2}\mathbf{f} \leq \lambda \mathbf{f}^\top (\mathbf{K}+\lambda \mathbf{I}_n)^{-1}\mathbf{f}$ and $\operatorname{tr}\left(\mathbf{K}^2(\mathbf{K}+\lambda \mathbf{I}_n)^{-2}\right) \leq \operatorname{tr}\left(\mathbf{K}(\mathbf{K}+\lambda \mathbf{I}_n)^{-1}\right) = s_\lambda(\mathbf{K})$, we define

$$\widehat{\mathscr{R}}_{\mathbf{K}}(\mathbf{f}) \stackrel{\text{def}}{=} n^{-1}\lambda \mathbf{f}^\top (\mathbf{K}+\lambda \mathbf{I}_n)^{-1}\mathbf{f} + n^{-1}\sigma_v^2 s_\lambda(\mathbf{K})$$

and note that $\mathscr{R}(\bar{f}) \leq \widehat{\mathscr{R}}_{\mathbf{K}}(\mathbf{f})$. The first term in the above expressions for $\mathscr{R}(\bar{f})$ and $\widehat{\mathscr{R}}_{\mathbf{K}}(\mathbf{f})$ is frequently referred to as the bias term, while the second is the variance term.

Lemma 4.3.1. *Suppose that* (4.7) *holds, and let* $\mathbf{f} \in \mathbb{R}^n$ *be the vector whose* j^{th} *entry is* $f^\star(\mathbf{x}_j)$. *Let* $\bar{f}$ *be the KRR estimator, and let* $\tilde{f}$ *be KRR estimator obtained using some other kernel* $\widetilde{k}(\cdot,\cdot)$ *whose kernel matrix is* $\widetilde{\mathbf{K}}$. *If* $\widetilde{\mathbf{K}}+\lambda \mathbf{I}_n$ *is a* Δ*-spectral approximation to* $\mathbf{K}+\lambda \mathbf{I}_n$ *for some* $\Delta < 1$, *and* $\|\mathbf{K}\|_{op} \geq 1$, *then the following bound holds:*

$$\mathscr{R}(\tilde{f}) \leq \widehat{\mathscr{R}}_{\widetilde{\mathbf{K}}}(\mathbf{f}) \leq (1-\Delta)^{-1}\widehat{\mathscr{R}}_{\mathbf{K}}(\mathbf{f}) + \frac{\Delta}{(1+\Delta)} \cdot \frac{\operatorname{rank}\widetilde{\mathbf{K}}}{n} \cdot \sigma_v^2 \tag{4.8}$$

Proof. Note that $\mathbf{A} \preceq \mathbf{B}$ implies that $\mathbf{B}^{-1} \preceq \mathbf{A}^{-1}$ so for the bias term we have:

$$\mathbf{f}^\top (\widetilde{\mathbf{K}}+\lambda \mathbf{I}_n)^{-1}\mathbf{f} \leq (1-\Delta)^{-1}\mathbf{f}^\top (\mathbf{K}+\lambda \mathbf{I}_n)^{-1}\mathbf{f}. \tag{4.9}$$

We now consider the variance term. Denote $s = \operatorname{rank}\widetilde{\mathbf{K}}$. We have:

$$\begin{aligned}
s_\lambda(\widetilde{\mathbf{K}}) = \operatorname{tr}\left((\widetilde{\mathbf{K}}+\lambda \mathbf{I}_n)^{-1}\widetilde{\mathbf{K}}\right) &= \sum_{i=1}^{s} \frac{\lambda_i(\widetilde{\mathbf{K}})}{\lambda_i(\widetilde{\mathbf{K}})+\lambda} \\
&= s - \sum_{i=1}^{s} \frac{\lambda}{\lambda_i(\widetilde{\mathbf{K}})+\lambda} \\
&\leq s - (1+\Delta)^{-1}\sum_{i=1}^{s} \frac{\lambda}{\lambda_i(\mathbf{K})+\lambda} \\
&= s - \sum_{i=1}^{s} \frac{\lambda}{\lambda_i(\mathbf{K})+\lambda} + \frac{\Delta}{1+\Delta}\sum_{i=1}^{s} \frac{\lambda}{\lambda_i(\mathbf{K})+\lambda} \\
&\leq n - \sum_{i=1}^{n} \frac{\lambda}{\lambda_i(\mathbf{K})+\lambda} + \frac{\Delta \cdot s}{1+\Delta} \\
&= s_\lambda(\mathbf{K}) + \frac{\Delta \cdot s}{1+\Delta} \\
&\leq (1-\Delta)^{-1} s_\lambda(\mathbf{K}) + \frac{\Delta \cdot s}{1+\Delta},
\end{aligned}$$

where we used the fact that $\mathbf{A} \preceq \mathbf{B}$ implies $\lambda_i(\mathbf{A}) \leq \lambda_i(\mathbf{B})$ (a simple consequence of the Courant-Fischer minimax theorem).

Combining the above variance bound with the bias bound in (4.9) yields:

$$\widehat{\mathscr{R}}_{\widetilde{\mathbf{K}}}(\mathbf{f}) \leq (1-\Delta)^{-1}\widehat{\mathscr{R}}_{\mathbf{K}}(\mathbf{f}) + \frac{\Delta}{(1+\Delta)} \cdot \frac{\operatorname{rank}\widetilde{\mathbf{K}}}{n} \cdot \sigma_v^2.$$

□

In short, Lemma 4.3.1 bounds the risk of the approximate KRR estimator as a function of both the risk upper bound $\widehat{\mathscr{R}}_{\mathbf{K}}(\mathbf{f})$ and an additive term which is small if $\operatorname{rank}\widetilde{\mathbf{K}}$ and/or Δ is small. In particular, it is instructive to compare the additive term $(\Delta/(1+\Delta))n^{-1}\sigma_v^2 \cdot \operatorname{rank}\widetilde{\mathbf{K}}$ to the variance term $n^{-1}\sigma_v^2 \cdot s_\lambda(\mathbf{K})$.

Remark 4.3.2. *An approximation $\widetilde{\mathbf{K}}$ is only useful computationally if* $\operatorname{rank}\widetilde{\mathbf{K}} \ll n$ *so $\widetilde{\mathbf{K}}$ gives a significantly compressed approximation to the original kernel matrix. Ideally we should have* $\operatorname{rank}\widetilde{\mathbf{K}}/n \to 0$ *as $n \to \infty$ and so the additive term in* (4.8) *will also approach* 0 *and generally be small when n is large.*

4.3.2 Random Features Preconditioning

Suppose we choose to solve $(\mathbf{K}+\lambda\mathbf{I}_n)\boldsymbol{\alpha} = \mathbf{y}$ using an iterative method (e.g. CG). In this case, we can apply $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ as a preconditioner. Using standard analysis of Krylov-subspace iterative methods it is immediate that if $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is a Δ-spectral approximation of $\mathbf{K}+\lambda\mathbf{I}_n$ then the number of iterations until convergence is $O\left(\sqrt{(1+\Delta)/(1-\Delta)}\right)$. Thus, if $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is, say, a $1/2$-spectral approximation of $\mathbf{K}+\lambda\mathbf{I}_n$, then the number of iterations is bounded by a constant. The preconditioner can be efficiently applied (after preprocessing) via the Woodbury formula, giving cost per iteration (if $s \le n$) of $O(n^2)$. The overall cost of computing the KRR estimator is therefore $O(ns^2 + n^2)$. Thus, as long as $s = o(n)$ this approach gives an advantage over direct methods which cost $O(n^3)$. For small s it also beats non-preconditioned iterative methods cost $O\left(n^2\sqrt{\kappa(\mathbf{K})}\right)$. See Cutajar et al. (2016) and Avron et al. (2017a) for a detailed discussion. The upshot though is that we reach again the question that was poised earlier: how big should s be so that $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is a $1/2$-spectral approximation of $\mathbf{K}+\lambda\mathbf{I}_n$?

4.4 Ridge Leverage Function Sampling and Random Fourier Features

In this section we present upper bounds on the number of random Fourier features needed to guarantee that $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is a Δ-spectral approximation to $\mathbf{K}+\lambda\mathbf{I}_n$. Our bounds apply to *any* shift-invariant kernel and a wide range of feature sampling distributions (in particular, classical random Fourier features).

Our analysis is based on relating the sampling density to an appropriately defined *ridge leverage function*. This function is a continuous generalization of the popular leverage scores (Mahoney and Drineas, 2009) and ridge leverage scores (Alaoui and Mahoney, 2015; Cohen et al., 2017) used in the analysis of linear methods. Bach (2017) defined the leverage function of the integral operator given by the kernel function and the data distribution. For our purposes, a more appropriate definition is with respect to a fixed input dataset:

Definition 4.4.1. For $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ and shift-invariant kernel $k(\cdot,\cdot)$, define the *ridge leverage function* as

$$\tau_\lambda(\boldsymbol{\eta}) \stackrel{\text{def}}{=} p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K}+\lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}).$$

In the above, $\mathbf{K}$ is the kernel matrix and $p(\cdot)$ is the distribution given by the inverse Fourier transform of $k(\cdot,\cdot)$.

We begin with two simple propositions. Recall that we assume $k(\mathbf{x},\mathbf{x}) = k(\mathbf{0}) = 1$ for any $\mathbf{x}$, however our results apply to general shift invariant kernel after appropriate scaling.

Proposition 1. *For all $\boldsymbol{\eta}$,*

$$\frac{n}{n+\lambda} \cdot p(\boldsymbol{\eta}) \leq \tau_\lambda(\boldsymbol{\eta}) \leq \frac{n}{\lambda} \cdot p(\boldsymbol{\eta}).$$

Proof. Since k is positive definite and $k(\mathbf{0}) = 1$, $|k(\mathbf{x},\mathbf{z})| \leq 1$ for all $\mathbf{x}$ and $\mathbf{z}$. This implies that the maximum eigenvalue of $\mathbf{K}$ is bounded by n. The lower bound follows, after noting that $\left\|\mathbf{z}(\boldsymbol{\eta})\right\|_2^2 = n$. The upper bound follows similarly, since all eigenvalues of $\mathbf{K} + \lambda \mathbf{I}_n$ are lower bounded by λ. □

Proposition 2. $\int_{\mathbb{R}^d} \tau_\lambda(\boldsymbol{\eta}) d\boldsymbol{\eta} = s_\lambda(\mathbf{K})$.

Proof.

$$\begin{aligned}
\int_{\mathbb{R}^d} \tau_\lambda(\boldsymbol{\eta}) d\boldsymbol{\eta} &= \int_{\mathbb{R}^d} p(\boldsymbol{\eta}) \mathbf{z}(\boldsymbol{\eta})^* (\mathbf{K} + \lambda \mathbf{I}_n)^{-1} \mathbf{z}(\boldsymbol{\eta}) d\boldsymbol{\eta} \\
&= \int_{\mathbb{R}^d} \operatorname{tr}\left(p(\boldsymbol{\eta}) (\mathbf{K} + \lambda \mathbf{I}_n)^{-1} \mathbf{z}(\boldsymbol{\eta}) \mathbf{z}(\boldsymbol{\eta})^* \right) d\boldsymbol{\eta} \\
&= \operatorname{tr}\left((\mathbf{K} + \lambda \mathbf{I}_n)^{-1} \int_{\mathbb{R}^d} p(\boldsymbol{\eta}) \mathbf{z}(\boldsymbol{\eta}) \mathbf{z}(\boldsymbol{\eta})^* d\boldsymbol{\eta} \right) \\
&= \operatorname{tr}\left((\mathbf{K} + \lambda \mathbf{I}_n)^{-1} \mathbf{K} \right) = s_\lambda(\mathbf{K}).
\end{aligned}$$

The second and third lines follow from the cyclic property and linearity of trace respectively. □

Recall that we denote the ratio n/λ, which appears frequently in our analysis, by $n_\lambda = n/\lambda$. As discussed, theoretical bounds generally set $\lambda = \omega(1)$ (as a function of n) so $n_\lambda = o(n)$. However we remark that in practice, it may sometimes be the case that λ is very small and $n_\lambda \boldsymbol{g} n$.

An immediate result of Propositions 1 and 2 (which can also be obtained algebraically from $\mathbf{K}$) is a generic bound on the statistical dimension of a kernel matrix:

Corollary 4.4.1. *For any $\mathbf{K} \in \mathbb{R}^{n \times n}$, $\mathbf{s}_\lambda(\mathbf{K}) \leq n_\lambda$.*

For any shift-invariant kernel with $k(\mathbf{x},\mathbf{x}) = 1$ and $k(\mathbf{x},\mathbf{z}) \to 0$ as $\|\mathbf{x}-\mathbf{z}\|_2 \to \infty$ (e.g., the Gaussian kernel) if we allow points to be arbitrarily spread out, the kernel matrix converges to the identity matrix, and $s_\lambda(\mathbf{I}_n) = n/(1+\lambda) = \Omega(n_\lambda)$ so the above bound is tight. However, this requires datasets of increasingly large diameter (as n grows). In contrast, the usual assumption in statistical learning is that the data is sampled from a bounded domain $\mathscr{X}$. In Section 4.7.4 we

show via a leverage function upper bound that for the important Gaussian kernel on bounded datasets $s_\lambda(\mathbf{K}) = o(n_\lambda)$.

In Chapter 3 we proved that spectral approximation bounds similar to (4.6) can be constructed by sampling columns relative to upper bounds on the leverage scores. In the following, we formalize this for the case of sampling Fourier features from a continuous domain. First, we need an auxiliary lemma which is a special case of Lemma B.1.1 in Appendix B.1.

Lemma 4.4.1. *Let $\mathbf{B}$ be a fixed $d \times d$ matrix. Construct a $d \times d$ random matrix $\mathbf{R}$ that satisfies*

$$\mathbb{E}[\mathbf{R}] = \mathbf{B} \quad \textit{and} \quad \|\mathbf{R}\|_{\mathrm{op}} \leq L.$$

Let $\mathbf{M}$ be semidefinite upper bounds for the expected squares, $\mathbb{E}[\mathbf{R}\mathbf{R}^*] \preceq \mathbf{M}$.
Form the matrix sampling estimator $\bar{\mathbf{R}}_n = \frac{1}{n}\sum_{k=1}^n \mathbf{R}_k$, where each $\mathbf{R}_k$ is an independent copy of $\mathbf{R}$. Then, for all $t \geq \sqrt{\|\mathbf{M}\|_{\mathrm{op}}/n} + 2L/3n$,

$$\Pr\left[\left\|\bar{\mathbf{R}}_n - \mathbf{B}\right\|_{\mathrm{op}} \geq t\right] \leq \frac{8\mathrm{tr}(\mathbf{M})}{\|\mathbf{M}\|_{\mathrm{op}}} \exp\left(\frac{-nt^2/2}{\|\mathbf{M}\|_{\mathrm{op}} + 2Lt/3}\right).$$

We prove a more general version of Lemma 4.4.1 in Appendix B.1 (see Lemma B.1.1).

Lemma 4.4.2. *Let $\tilde{\tau} : \mathbb{R}^d \to \mathbb{R}$ be a measurable function such that $\tilde{\tau}(\boldsymbol{\eta}) \geq \tau_\lambda(\boldsymbol{\eta})$ for all $\boldsymbol{\eta} \in \mathbb{R}^d$, and furthermore assume that*

$$s_{\tilde{\tau}} \equiv \int_{\mathbb{R}^d} \tilde{\tau}(\boldsymbol{\eta})\, d\boldsymbol{\eta}$$

is finite. Denote $p_{\tilde{\tau}}(\boldsymbol{\eta}) = \tilde{\tau}(\boldsymbol{\eta})/s_{\tilde{\tau}}$. Let $\Delta \leq 1/2$ and $\rho \in (0,1)$. Assume that $\|\mathbf{K}\|_{op} \geq \lambda$. Suppose we take $s \geq \frac{8}{3}\Delta^{-2} s_{\tilde{\tau}} \ln(16 s_\lambda(\mathbf{K})/\rho)$ i.i.d. samples $\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_s$ from the distribution associated with the density $p_{\tilde{\tau}}(\cdot)$ and then construct the matrix $\mathbf{Z}$ according to (4.5) with $q = p_{\tilde{\tau}}$. Then $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is Δ-spectral approximation of $\mathbf{K} + \lambda\mathbf{I}_n$ with probability at least $1 - \rho$.

Proof. Let $\mathbf{K} + \lambda\mathbf{I}_n = \mathbf{V}^\top \mathbf{\Sigma}^2 \mathbf{V}$ be an eigendecomposition of $\mathbf{K} + \lambda\mathbf{I}_n$. Note that the Δ-spectral approximation guarantee (4.2) is equivalent to

$$\mathbf{K} - \Delta(\mathbf{K} + \lambda\mathbf{I}_n) \preceq \mathbf{Z}\mathbf{Z}^* \preceq \mathbf{K} + \Delta(\mathbf{K} + \lambda\mathbf{I}_n),$$

so by multiplying by $\mathbf{\Sigma}^{-1}\mathbf{V}$ on the left and $\mathbf{V}^\top\mathbf{\Sigma}^{-1}$ on the right we find that it suffices to show that

$$\left\|\mathbf{\Sigma}^{-1}\mathbf{V}\mathbf{Z}\mathbf{Z}^*\mathbf{V}^\top\mathbf{\Sigma}^{-1} - \mathbf{\Sigma}^{-1}\mathbf{V}\mathbf{K}\mathbf{V}^\top\mathbf{\Sigma}^{-1}\right\|_{\mathrm{op}} \leq \Delta \tag{4.10}$$

holds with probability of at least $1 - \rho$. Let

$$\mathbf{Y}_l = \frac{p(\boldsymbol{\eta}_l)}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)} \mathbf{\Sigma}^{-1}\mathbf{V}\mathbf{z}(\boldsymbol{\eta}_l)\mathbf{z}(\boldsymbol{\eta}_l)^*\mathbf{V}^\top\mathbf{\Sigma}^{-1}.$$

Note that $\mathbb{E}[\mathbf{Y}_l] = \mathbf{\Sigma}^{-1}\mathbf{V}\mathbf{K}\mathbf{V}^\top\mathbf{\Sigma}^{-1}$ and $\frac{1}{s}\sum_{l=1}^s \mathbf{Y}_l = \mathbf{\Sigma}^{-1}\mathbf{V}\mathbf{Z}\mathbf{Z}^*\mathbf{V}^\top\mathbf{\Sigma}^{-1}$. Thus, we can use matrix concentration result of Lemma 4.4.1 to prove (4.10).

To apply this bound we need to bound the norm of $\mathbf{Y}_l$ as well as the stable rank $\mathbb{E}\left[\mathbf{Y}_l^2\right]$. Since $\mathbf{Y}_l$ is always a rank one matrix,

$$\begin{aligned}
\|\mathbf{Y}_l\|_{\text{op}} &= \frac{p(\boldsymbol{\eta}_l)}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)}\text{tr}\left(\boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{z}(\boldsymbol{\eta}_l)\mathbf{z}(\boldsymbol{\eta}_l)^*\mathbf{V}^\top\boldsymbol{\Sigma}^{-1}\right) \\
&= \frac{p(\boldsymbol{\eta}_l)}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)}\mathbf{z}(\boldsymbol{\eta}_l)^*\mathbf{V}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{z}(\boldsymbol{\eta}_l) \\
&= \frac{p(\boldsymbol{\eta}_l)}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)}\mathbf{z}(\boldsymbol{\eta}_l)^*(\mathbf{K}+\lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}_l) \\
&= \frac{s_{\tilde{\tau}}\cdot\tau_\lambda(\boldsymbol{\eta}_l)}{\tilde{\tau}(\boldsymbol{\eta}_l)} \le s_{\tilde{\tau}}
\end{aligned}$$

since $\tilde{\tau}_\lambda(\boldsymbol{\eta}_l) \ge \tau(\boldsymbol{\eta}_l)$ by assumption of the lemma. We also have:

$$\begin{aligned}
\mathbf{Y}_l^2 &= \frac{p(\boldsymbol{\eta}_l)^2}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)^2}\boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{z}(\boldsymbol{\eta}_l)\mathbf{z}(\boldsymbol{\eta}_l)^*\mathbf{V}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{z}(\boldsymbol{\eta}_l)\mathbf{z}(\boldsymbol{\eta}_l)^*\mathbf{V}^\top\boldsymbol{\Sigma}^{-1} \\
&= \frac{p(\boldsymbol{\eta}_l)^2}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)^2}\boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{z}(\boldsymbol{\eta}_l)\mathbf{z}(\boldsymbol{\eta}_l)^*(\mathbf{K}+\lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta}_l)^*\mathbf{V}^\top\boldsymbol{\Sigma}^{-1} \\
&= \frac{p(\boldsymbol{\eta}_l)\tau(\boldsymbol{\eta}_l)}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)^2}\boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{z}(\boldsymbol{\eta}_l)\mathbf{z}(\boldsymbol{\eta}_l)^*\mathbf{V}^\top\boldsymbol{\Sigma}^{-1} \\
&= \frac{\tau(\boldsymbol{\eta}_l)}{p_{\tilde{\tau}}(\boldsymbol{\eta}_l)}\mathbf{Y}_l \\
&= \frac{s_{\tilde{\tau}}\tau(\boldsymbol{\eta}_l)}{\tilde{\tau}(\boldsymbol{\eta}_l)}\mathbf{Y}_l \preceq s_{\tilde{\tau}}\mathbf{Y}_l.
\end{aligned}$$

Let $\lambda_1 \ge \cdots \ge \lambda_n$ be the eigenvalues of $\mathbf{K}$. We have

$$\begin{aligned}
\mathbb{E}[s_{\tilde{\tau}}\mathbf{Y}_l] &= s_{\tilde{\tau}}\boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{K}\mathbf{V}^\top\boldsymbol{\Sigma}^{-1} \\
&= s_{\tilde{\tau}}\left(\mathbf{I}_n - \lambda\boldsymbol{\Sigma}^{-2}\right) \\
&= s_{\tilde{\tau}}\cdot\text{diag}\left(\lambda_1/(\lambda_1+\lambda),\ldots,\lambda_n/(\lambda_n+\lambda)\right) := \mathbf{D}.
\end{aligned}$$

So,

$$\begin{aligned}
\Pr\left(\left\|\frac{1}{s}\sum_{l=1}^{s}\mathbf{Y}_l - \boldsymbol{\Sigma}^{-1}\mathbf{V}\mathbf{K}\mathbf{V}^\top\boldsymbol{\Sigma}^{-1}\right\|_{\text{op}} \ge \Delta\right) &\le \frac{8\text{tr}(\mathbf{D})}{\|\mathbf{D}\|_{\text{op}}}\exp\left(\frac{-s\Delta^2/2}{\|\mathbf{D}\|_{\text{op}}+2s_{\tilde{\tau}}\Delta/3}\right) \\
&\le 16 s_\lambda(\mathbf{K})\exp\left(\frac{-s\Delta^2}{2s_{\tilde{\tau}}(1+2\Delta/3)}\right) \\
&\le 16 s_\lambda(\mathbf{K})\exp\left(\frac{-3s\Delta^2}{8s_{\tilde{\tau}}}\right) \le \rho
\end{aligned}$$

where the second inequality is due to the assumption that $\lambda_1 = \|\mathbf{K}\|_{\text{op}} \ge \lambda$ and hence $\|\mathbf{D}\|_{\text{op}} \ge s_{\tilde{\tau}}/2$. The last inequality is due to the bound on s. □

Lemma 4.4.2 shows that if we sample using the ridge leverage function, then $O\left(s_\lambda(\mathbf{K})\log(s_\lambda(\mathbf{K}))\right)$ samples suffice for spectral approximation of $\mathbf{K}$ (for a fixed Δ and failure probability). While

there is no straightforward way to perform this sampling, we can consider how well the classic random Fourier features sampling distribution approximates the leverage function, obtaining a bound on its performance:

Theorem 4.4.1. *Let $\Delta \le 1/2$ and $\rho \in (0,1)$. Assume that $\|\mathbf{K}\|_2 \ge \lambda$. If we use $s \ge \frac{8}{3}\Delta^{-2} n_\lambda \ln\left(16\frac{s_\lambda(\mathbf{K})}{\rho}\right)$ random Fourier features (i.e., sampled according to $p(\cdot)$), then $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is Δ-spectral approximation of $\mathbf{K} + \lambda\mathbf{I}_n$ with probability of at least $1-\rho$.*

Proof. Define $\tilde{\tau}(\boldsymbol{\eta}) = p(\boldsymbol{\eta}) \cdot n_\lambda$ and note that $\tilde{\tau}(\boldsymbol{\eta}) \ge \tau_\lambda(\boldsymbol{\eta})$ by Proposition 1 and that $s_{\tilde{\tau}} = n_\lambda$. Finally, note that $p_{\tilde{\tau}}(\boldsymbol{\eta}) = p(\boldsymbol{\eta})$, the classic Fourier features sampling probability. □

Theorem 4.4.1 establishes that if $\lambda = \omega(\log(n))$ and Δ is fixed, $o(n)$ random Fourier features suffice for spectral approximation, and so the method can provably speed up KRR. Nevertheless, the bound depends on n_λ instead of $s_\lambda(\mathbf{K})$, as is possible with true leverage function sampling (see Lemma 4.4.2). This gap arises from our use of the simple, often loose, leverage function upper bound given by Proposition 1.

Unfortunately, the bound in Theorem 4.4.1 cannot be improved. Even for the special case of a one-dimensional Gaussian kernel, the classic random Fourier features sampling distribution is far enough from the ridge leverage distribution that $\Omega(n_\lambda)$ features may be needed even when $s_\lambda(\mathbf{K}) = o(n_\lambda)$. On the otherhand, a simple modified sampling approach *does* closely approximate the true ridge leverage distribution and so yields significantly better bounds for the Gaussian kernel. We present these results in Section 4.5 and Section 4.6 respectively. We defer a discussion of their proofs to Section 4.7, where we develop our main technical contribution: a sharper understanding of the ridge leverage function based on a formulation as the solution to two dual optimization problems which give corresponding upper and lower bounds on the distribution and, correspondingly, on sampling performance.

4.5 Lower Bound for Classic Random Fourier Features

Our lower bound shows that the upper bound of Theorem 4.4.1 on the number of samples required by classic random Fourier features to obtain a spectral approximation to $\mathbf{K} + \lambda\mathbf{I}_n$ is essentially best possible. The full proof is given in Appendix C.5.

Theorem 4.5.1. *Consider the d-dimensional Gaussian kernel with $\sigma = (2\pi)^{-1}$ (so $p(\boldsymbol{\eta}) = (2\pi)^{-d/2} e^{-\|\boldsymbol{\eta}\|_2^2/2}$). For any odd integer $n \ge 8\ln n_\lambda$, any λ satisfying $0 < \lambda \le \frac{n}{256}$, and every radius R such that $600\ln^{3/2} n_\lambda \le R \le \frac{n}{80\sqrt{\ln(n_\lambda)}}$, there exists a dataset of n points $\{\mathbf{x}_j\}_{j=1}^n \subseteq [-R,R]^d$ such that if s random Fourier features (i.e., sampled according to $p(\cdot)$) are sampled for some s satisfying $s \le \frac{n_\lambda}{2^{15}}$, then with probability at least $1/2$, there exists a vector $\boldsymbol{\alpha} \in \mathbb{R}^n$ such that:*

$$\boldsymbol{\alpha}^\top(\mathbf{K} + \lambda\mathbf{I}_n)\boldsymbol{\alpha} < \frac{2}{3}\boldsymbol{\alpha}^\top(\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n)\boldsymbol{\alpha}. \tag{4.11}$$

Furthermore, for the said dataset, $s_\lambda(\mathbf{K}) = O\left(R \cdot \operatorname{polylog} n_\lambda\right)$.

Theorem 4.5.1 shows that the number of samples s required for $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ to be a $1/2$-spectral approximation to $\mathbf{K} + \lambda\mathbf{I}_n$ for a bounded dataset of points must depend at least linearly on n_λ. So, there is an asymptotic gap between what is achieved with classical random Fourier features and what is achieved by modified random Fourier features using leverage function sampling.

As we will see in Section 4.7, the key idea behind the proof of Theorem 4.5.1 is to show that for a dataset contained in $[-R, R]^d$, the ridge leverage function is large on a range of bandlimited frequencies. In contrast, the classic random Fourier features distribution is very small at the edges of this frequency band, and so significantly undersamples some frequencies and does not achieve spectral approximation.

We remark that it would have been preferable if Theorem 4.5.1 applied to bounded datasets (i.e. with R fixed), as the usual assumption in statistical learning theory is that data is sampled from a bounded domain. However, our current techniques are unable to address this scenario. Nevertheless, our analysis allows R to grow very slowly with n and we conjecture that the lower bound is tight even for bounded domains.

4.6 Improved Sampling for the Gaussian Kernel

Contrasting with the lower bound of Theorem 4.5.1, we now propose a modified Fourier feature sampling distribution that performs near-optimally for the Gaussian kernel on bounded input sets. Furthermore, unlike the true ridge leverage function, this distribution is simple and efficient to sample from. To reduce clutter, we state the result for a fixed bandwidth $\sigma = (2\pi)^{-1}$. This is without loss of generality since we can always rescale the points by $(2\pi\sigma)^{-1}$.

Our modified distribution essentially corrects the classic distribution by "capping" the probability of sampling low frequencies near the origin. This allows it to allocate more samples to higher frequencies, which are undersampled by classical random Fourier features. See Figure 4.1 for a visual comparison of the two distributions.

Definition 4.6.1 (Improved Fourier feature distribution for the Gaussian kernel)**.** Define the function

$$\bar{\tau}_R(\boldsymbol{\eta}) \equiv \begin{cases} \left(6.2R + 1240\ln^{1.5} n_\lambda\right)^d + 1 & \|\boldsymbol{\eta}\|_\infty \leq 10\sqrt{\ln(n_\lambda)} \\ n_\lambda p(\boldsymbol{\eta}) \prod_{j=1}^d \max\left(1, |\eta_j|\right) & \text{otherwise} \end{cases}$$

Let $s_{\bar{\tau}_R} = \int_{\mathbb{R}} \bar{\tau}_R(\boldsymbol{\eta}) d\boldsymbol{\eta}$ and define the probability density function $\bar{p}_R(\boldsymbol{\eta}) = \bar{\tau}_R(\boldsymbol{\eta}) / s_{\bar{\tau}_R}$.

Note that $\bar{p}_R(\boldsymbol{\eta})$ is just the uniform distribution for low frequencies with $\|\boldsymbol{\eta}\|_\infty \leq 10\sqrt{\log(n_\lambda)}$, and a slightly modified classic Fourier features distribution, appropriately scaled, outside this range. As we show in Section 4.7, $\bar{\tau}_R(\boldsymbol{\eta})$ upper bounds the true ridge leverage function $\tau_\lambda(\boldsymbol{\eta})$ for all $\boldsymbol{\eta}$. Hence, simply applying Lemma 4.4.2:

Theorem 4.6.1. *Consider the d-dimensional Gaussian kernel with $\sigma = (2\pi)^{-1}$ (so $p(\boldsymbol{\eta}) = (2\pi)^{-d/2} e^{-\|\boldsymbol{\eta}\|_2^2/2}$) and any dataset of n points $\{\mathbf{x}_j\}_{j=1}^n \subseteq \mathbb{R}^d$ contained in a ℓ_∞-ball of radius*

R (i.e $\left\|\mathbf{x}_i - \mathbf{x}_j\right\|_\infty \le 2R$ for all $i, j \in [n]$). For any λ such that $d \le 50\ln(n_\lambda) + O(1)$, if we sample $s \ge \frac{8}{3}\Delta^{-2} s_{\bar\tau_R} \ln(16 s_\lambda(\mathbf{K})/\rho)$ Fourier features according to $\bar p_R(\cdot)$ and construct $\mathbf{Z}$ according to (4.5)*, then with probability at least $1-\rho$, $\mathbf{Z}\mathbf{Z}^* + \lambda\mathbf{I}_n$ is Δ-spectral approximation of $\mathbf{K} + \lambda\mathbf{I}_n$. Furthermore, $s_{\bar\tau_R} = O\Big((248R)^d \ln^{d/2} n_\lambda + (223\ln n_\lambda)^{2d}\Big)$ and $\bar p_R(\cdot)$ can be sampled from in $O(d)$ time.*

Proof. The result follows from Lemma 4.4.2 and the fact that $\bar\tau_R(\cdot)$ upper bounds the true ridge leverage function, which follows from Theorem 4.7.1 of Section 4.7 along with Proposition 1. The bound on $s_{\bar\tau_R}$ can be computed as follows. Let us denote $g_1(\eta) = (2\pi)^{-1/2} e^{-\eta^2/2} \max(1, |\eta|)$ and $g(\boldsymbol{\eta}) = g_1(\eta_1) \cdot \ldots \cdot g_1(\eta_d)$. We calculate

$$A \equiv \int_{-\infty}^{\infty} g_1(\eta) d\eta = \operatorname{erf}(1/\sqrt{2}) + \sqrt{2/e\pi} \approx 1.1663$$

$$B \equiv 2\int_{10\sqrt{\ln n_\lambda}}^{\infty} g_1(\eta) d\eta = \sqrt{\frac{2}{\pi}} n_\lambda^{-50}.$$

We now have $\int_{\|\boldsymbol{\eta}\|_\infty > 10\sqrt{\ln(n_\lambda)}} g(\boldsymbol{\eta}) d\boldsymbol{\eta} = A^d - (A-B)^d$. The assumption $d \le 50\ln(n_\lambda) + O(1)$ ensures that,

$$\begin{aligned} s_{\bar\tau_R} &= \int_{\mathbb{R}^d} \bar\tau_R(\boldsymbol{\eta}) d\boldsymbol{\eta} \\ &= \left(\left(6.2R + 1240\ln^{1.5} n_\lambda\right)^d + 1\right)\left(20\sqrt{\ln n_\lambda}\right)^d + n_\lambda \cdot \int_{\|\boldsymbol{\eta}\|_\infty > 10\sqrt{\ln(n_\lambda)}} g(\boldsymbol{\eta}) d\boldsymbol{\eta} \\ &= O\Big((248R)^d \ln(n_\lambda)^{d/2} + (223\ln n_\lambda)^{2d}\Big). \end{aligned}$$

Sampling from $\bar\tau_R(\eta)$ amounts to sampling from the mixture of a uniform distribution on $\left[-10\sqrt{\ln n_\lambda}, 10\sqrt{\ln n_\lambda}\right]^d$ and the tail of the distribution defined by $\bar\tau_R$: with probability $\frac{1}{s_{\bar\tau_R}}\left(20\sqrt{\ln n_\lambda}\right)^d \cdot \left(\left(6.2R + 1240\ln^{1.5} n_\lambda\right)^d + 1\right)$ sample from the uniform distribution and with the remaining probability sample from the tail. Using the above expression for the total mass of the tail, we can decide whether to sample from the uniform part or from the tail part by generating a single sample with uniform distribution on $[0,1]$.

Sampling from the uniform part, clearly takes $O(d)$ time. Sampling from the tail can be easily done via rejection sampling at $O(d)$ expected cost, as we show now. The density p_t of the tail is:

$$p_t(\boldsymbol{\eta}) = \frac{g(\boldsymbol{\eta}) \cdot \mathbb{1}\left[\|\boldsymbol{\eta}\|_\infty \ge 10\sqrt{\ln n_\lambda}\right]}{\int_{\|\boldsymbol{\eta}'\|_\infty \ge 10\sqrt{\ln n_\lambda}} g(\boldsymbol{\eta}') d\boldsymbol{\eta}'}.$$

We can write $\mathbb{1}\left[\|\boldsymbol{\eta}\|_\infty \ge 10\sqrt{\ln n_\lambda}\right]$ as a union of disjoint partitions as follows:

$$\mathbb{1}\left[\|\boldsymbol{\eta}\|_\infty \ge 10\sqrt{\ln n_\lambda}\right] = \sum_{j=1}^{d} \mathbb{1}\left[|\eta_j| \ge 10\sqrt{\ln n_\lambda}\right] \mathbb{1}\left[|\eta_k| < 10\sqrt{\ln n_\lambda}\ \forall k \in \{1,..,j-1\}\right]$$

Let R_j denote the jth region in the above partition for every $j \in [d]$:

$$R_j = \left\{ \boldsymbol{\eta} \in \mathbb{R}^d \,:\, |\eta_j| \geq 10\sqrt{\ln n_\lambda}\,, |\eta_k| < 10\sqrt{\ln n_\lambda}\, \forall k \in \{1,..,j-1\} \right\}$$

Thus, the density p_t can written as follows:

$$p_t(\boldsymbol{\eta}) = \frac{g(\boldsymbol{\eta}) \cdot \sum_{j=1}^{d} \mathbb{1}\left[\boldsymbol{\eta} \in R_j\right]}{\int_{\|\boldsymbol{\eta}'\|_\infty \geq 10\sqrt{\ln n_\lambda}} g(\boldsymbol{\eta}')d\boldsymbol{\eta}'}$$

Now because R_j's are disjoint sets we can sample from p_t in the following fashion.

1. First generate a sample $j \in [d]$ with probability $\frac{\int_{\boldsymbol{\eta}\in R_j} g(\boldsymbol{\eta})d\boldsymbol{\eta}}{\int_{\|\boldsymbol{\eta}'\|_\infty \geq 10\sqrt{\ln n_\lambda}} g(\boldsymbol{\eta})d\boldsymbol{\eta}'}$. In order to execute this step, we first compute:

   $$\int_{\boldsymbol{\eta}\in R_j} g(\boldsymbol{\eta})d\boldsymbol{\eta} = A^{d-j}(A-B)^{j-1}B.$$

   Then given the probabilities we can sample j in $O(d)$ time.

2. Next, generate a sample from the distribution:

   $$\begin{aligned} p_{t,j}(\boldsymbol{\eta}) &= \frac{g(\boldsymbol{\eta}) \cdot \mathbb{1}\left[\boldsymbol{\eta} \in R_j\right]}{\int_{\boldsymbol{\eta}'\in R_j} g(\boldsymbol{\eta}')d\boldsymbol{\eta}'} \\ &= \frac{g_1(\eta_j) \cdot \mathbb{1}\left[|\eta_j| \geq 10\sqrt{\ln n_\lambda}\right]}{B} \cdot \prod_{k=1}^{j-1} \frac{g_1(\eta_k) \cdot \mathbb{1}\left[|\eta_k| < 10\sqrt{\ln n_\lambda}\right]}{A-B} \cdot \prod_{k=j+1}^{d} \frac{g_1(\eta_k)}{A} \end{aligned}$$

   We explain how to sample from this distribution in the subsequent paragraphs.

We now explain how to perform the second step of sampling. It can be seen in the above expression that sampling from the distribution whose density is $p_{t,j}(\boldsymbol{\eta})$ amounts to sampling each of d coordinates of $\boldsymbol{\eta}$ independently from their corresponding distributions. There are three types of distributions that we need to sample from. Either we need to sample proportional to $g_1(\cdot)$ (coordinates $j+1, j+2, \cdots d$) or we need to sample from the (rescaled) head of g_1 (coordinates $1, 2, \ldots j-1$), or we sample from the tail of $g_1(\cdot)$ (coordinate j).

We start with sampling proportional to g_1. This distribution is a mixture of Gaussian on $[-1,1]$ and enlarged Gaussian outside. The total mass is A, and the relative mass of the Gaussian part is $\mathrm{erf}(1/\sqrt{2})/A$. First, we sample a random variable $U \sim \mathrm{Unif}([0,1])$, which will decide which part of the mixture we sample. If U is bigger than $\mathrm{erf}(1/\sqrt{2})/A$, then the sample comes from the tail. In that case, we generate the sample by computing $G^{-1}(U)$ where $G(\xi) \equiv A^{-1}\int_{-\xi}^{\xi} g_1(\eta)d\eta$ (i.e., we use inverse transform sampling). Note that G has a simple invertible closed form for values larger than 1, we have $G(1) = \mathrm{erf}(1/\sqrt{2})/A$. If $U \leq \mathrm{erf}(1/\sqrt{2})/A$, then the sample comes from the Gaussian part. To generate the sample from the head, we sample a standard Gaussian

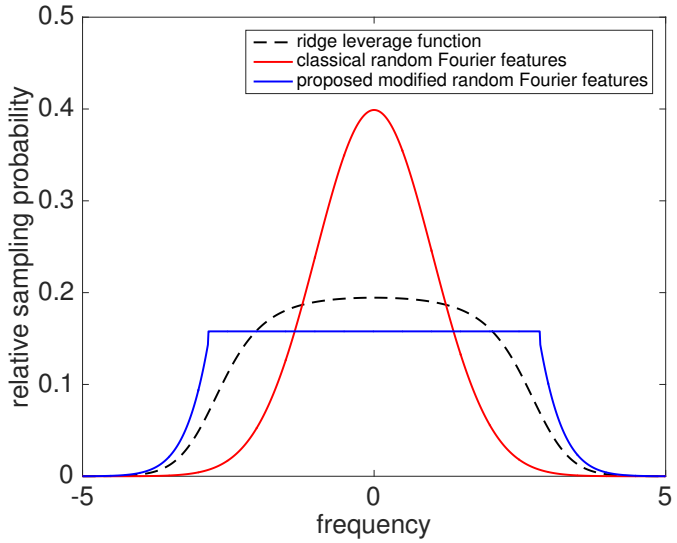


Figure 4.1 – Plot of the true ridge leverage function vs. the classic random Fourier features distribution and our modified distribution, for a dataset of $n = 401$ equispaced points on the range $[-5, 5]$. Our modified distribution closely matches the true leverage scores to within a small multiplicative factor. In contrast, the classical distribution oversamples low frequencies, at the expense of substantially undersampling higher frequencies.

X, and test whether $X \leq 1$. If it is, then we use the sample, otherwise we reject and repeat. Obviously, the expected number of samples we need is $O(1)$.

To sample proportional to the head of g_1, we repeat the above procedure and test whether the sample is smaller than $10\sqrt{\ln n_\lambda}$. If it is not, we reject the sample and repeat.

To sample proportional to the tail of g_1, we sample a uniform random variable T on $[0, B/A]$, and return $G^{-1}(1-T)$, using the closed from expression for G^{-1} for values close to 1.

Thus, we can generate a sample in step 2 in $O(d)$ expected time, and overall the sampling procedure takes $O(d)$. □

Theorem 4.6.1 represents a possibly exponential improvement over the bound obtainable by classic random Fourier features. Consider $d = 1$ and $R \geq \log^{1.5}(n_\lambda)$. The bound on $s_{\tilde{\tau}_R}$ shows that our modified distribution requires $O\left(R\sqrt{\log(n_\lambda)}\right)$ samples, as opposed to $\Omega(n_\lambda)$ lower bound given by Theorem 4.5.1.

4.7 Bounding the Ridge Leverage Function

We now bound the ridge leverage function of the Gaussian kernel, which leads to Theorems 4.5.1 and 4.6.1. The key idea is to reformulate the leverage function as the solution of two dual optimization problems. By exhibiting suitable test functions for these optimization

problems, we are able to give tight upper and lower bounds on the ridge leverage function, and correspondingly on the sampling performance of classic and modified Fourier features.

4.7.1 Primal-Dual Characterization

Before introducing our primal-dual characterization of the ridge leverage function, we present a few definitions. Define the operator $\mathbf{\Phi} : L_2(\mu) \to \mathbb{C}^n$ by

$$\mathbf{\Phi} y \stackrel{\text{def}}{=} \int_{\mathbb{R}^d} \mathbf{z}(\boldsymbol{\xi}) y(\boldsymbol{\xi}) d\mu(\boldsymbol{\xi}). \tag{4.12}$$

We first prove that the operator $\mathbf{\Phi}$ is defined on all $L_2(\mu)$ and is a bounded linear operator. Indeed, for $y \in L_2(\mu)$,

$$\begin{aligned}
\left\|\mathbf{\Phi} y\right\|_2^2 &= \left\| \int_{\mathbb{R}^d} \mathbf{z}(\boldsymbol{\xi}) y(\boldsymbol{\xi}) d\mu(\boldsymbol{\xi}) \right\|_2^2 \\
&\leq \int_{\mathbb{R}^d} \left\| \mathbf{z}(\boldsymbol{\xi}) y(\boldsymbol{\xi}) \right\|_2^2 d\mu(\boldsymbol{\xi}) \\
&= \int_{\mathbb{R}^d} |y(\boldsymbol{\xi})|^2 \cdot \|\mathbf{z}(\boldsymbol{\xi})\|_2^2 \, d\mu(\boldsymbol{\xi}) \\
&= n \cdot \left\| y \right\|_{L_2(d\mu)}^2 .
\end{aligned}$$

Therefore, there is a unique adjoint operator $\mathbf{\Phi}^* : \mathbb{C}^n \to L_2(\mu)$, such that $\langle \mathbf{\Phi} y, \mathbf{x} \rangle = \langle y, \mathbf{\Phi}^* \mathbf{x} \rangle_{L_2(\mu)}$ for every $y \in L_2(\mu)$ and $\mathbf{x} \in \mathbb{C}^n$. One can verify that $[\mathbf{\Phi}^* \mathbf{x}](\boldsymbol{\eta}) = \mathbf{z}(\boldsymbol{\eta})^* \mathbf{x}$. The following holds:

Proposition 3. *For every $\mathbf{x} \in \mathbb{C}^n$:*

$$\mathbf{\Phi}\mathbf{\Phi}^* \mathbf{x} = \mathbf{K}\mathbf{x}.$$

Proof. For every $\mathbf{x} \in \mathbb{C}^n$,

$$\begin{aligned}
\mathbf{\Phi}\mathbf{\Phi}^* \mathbf{x} &= \int_{\mathbb{R}^d} \mathbf{z}(\boldsymbol{\xi}) [\mathbf{\Phi}^* \mathbf{x}](\boldsymbol{\xi}) d\mu(\boldsymbol{\xi}) \\
&= \int_{\mathbb{R}^d} \mathbf{z}(\boldsymbol{\xi}) \mathbf{z}(\boldsymbol{\xi})^* \mathbf{x} \, d\mu(\boldsymbol{\xi}) \\
&= \left(\int_{\mathbb{R}^d} \mathbf{z}(\boldsymbol{\xi}) \mathbf{z}(\boldsymbol{\xi})^* d\mu(\boldsymbol{\xi}) \right) \mathbf{x} = \mathbf{K}\mathbf{x}.
\end{aligned}$$

□

We can now equivalently define the ridge leverage function $\tau_\lambda(\cdot)$ via the following optimization problems. Similar characterization are known for the finite dimensional case. Here we extend these results to an infinite dimensional case.

Lemma 4.7.1. *The ridge leverage function (Definition 4.4.1) can alternatively be defined as:*

$$\tau_\lambda(\boldsymbol{\eta}) = \min_{y \in L_2(\mu)} \lambda^{-1} \left\| \mathbf{\Phi} y - \sqrt{p(\boldsymbol{\eta})} \mathbf{z}(\boldsymbol{\eta}) \right\|_2^2 + \left\| y \right\|_{L_2(\mu)}^2 . \tag{4.13}$$

Proof. The minimizer of the right-hand side of (4.13) can be obtained from the usual normal equations, and simplified using the matrix inversion lemma for operators (Ogawa, 1988):

$$\begin{aligned} y^\star &= \sqrt{p(\boldsymbol{\eta})}\left(\boldsymbol{\Phi}^*\boldsymbol{\Phi} + \lambda \mathbf{I}_{L_2(\mu)}\right)^{-1}\boldsymbol{\Phi}^*\mathbf{z}(\boldsymbol{\eta}) \\ &= \sqrt{p(\boldsymbol{\eta})}\boldsymbol{\Phi}^*(\boldsymbol{\Phi}\boldsymbol{\Phi}^* + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) \\ &= \sqrt{p(\boldsymbol{\eta})}\boldsymbol{\Phi}^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) \end{aligned}$$

where we used Proposition 3. So, for $y^\star(\boldsymbol{\xi}) = \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\xi})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})$ we have:

$$\begin{aligned} \|y^\star\|^2_{L_2(\mu)} &= p(\boldsymbol{\eta})\int_{\mathbb{R}^d} \left|\mathbf{z}(\boldsymbol{\xi})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})\right|^2 d\mu(\boldsymbol{\xi}) \\ &= p(\boldsymbol{\eta})\int_{\mathbb{R}^d} \mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\xi})\mathbf{z}(\boldsymbol{\xi})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})\, d\mu(\boldsymbol{\xi}) \\ &= p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\left(\int_{\mathbb{R}^d} \mathbf{z}(\boldsymbol{\xi})\mathbf{z}(\boldsymbol{\xi})^* d\mu(\boldsymbol{\xi})\right)(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) \\ &= p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{K}(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) \\ &= p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) - \lambda p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-2}\mathbf{z}(\boldsymbol{\eta}). \end{aligned}$$

Additionally,

$$\begin{aligned} \left\|\boldsymbol{\Phi} y^\star - \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta})\right\|_2^2 &= p(\boldsymbol{\eta})\left\|\boldsymbol{\Phi}\boldsymbol{\Phi}^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) - \mathbf{z}(\boldsymbol{\eta})\right\|_2^2 \\ &= p(\boldsymbol{\eta})\left\|(\mathbf{K}(\mathbf{K} + \lambda \mathbf{I}_n)^{-1} - \mathbf{I}_n)\mathbf{z}(\boldsymbol{\eta})\right\|_2^2 \\ &= p(\boldsymbol{\eta})\left\|\lambda(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})\right\|_2^2 \\ &= \lambda^2 p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-2}\mathbf{z}(\boldsymbol{\eta}),. \end{aligned}$$

Plugging the above equations into (4.13) gives:

$$\begin{aligned} &\|y^\star\|^2_{L_2(\mu)} + \lambda^{-1}\left\|\boldsymbol{\Phi} y^\star - \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta})\right\|_2^2 \\ &\quad = p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) - \lambda p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-2}\mathbf{z}(\boldsymbol{\eta}) \\ &\qquad + \lambda p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-2}\mathbf{z}(\boldsymbol{\eta}) \\ &\quad = p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda \mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta}) \\ &\quad = \tau_\lambda(\boldsymbol{\eta}). \end{aligned}$$

□

Recall that we define $\mathbf{z}(\boldsymbol{\eta})_j = e^{-2\pi i \mathbf{x}_j^\top \boldsymbol{\eta}}$. So $\boldsymbol{\Phi}$ is just a d-dimensional Fourier transform of the function y weighted by the probability measure $\mu(\boldsymbol{\xi})$, and evaluated at the frequencies given by the data points $\mathbf{x}_1, ..., \mathbf{x}_n$. Thus, the optimization problem of Lemma 4.7.1 asks us to produce a function y whose Fourier transform is close to the pure cosine wave on our datapoints $\sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta})$. At the same time, to keep the second term of (4.13) small, y should have bounded norm under the $\mu(\boldsymbol{\xi})$ measure. So, the trivial solution of setting y to be a Dirac

delta function at $\boldsymbol{\eta}$ (whose Fourier transform is a pure cosine with frequency $\boldsymbol{\eta}$) fails. A more carefully chosen function must be constructed whose Fourier transform looks like the cosine at our datapoints but diverges elsewhere. Such a function certifies that, on our datapoints, the cosine of frequency $\boldsymbol{\eta}$ can be approximately reconstructed with low energy using other frequencies. Hence $\boldsymbol{\eta}$ is not a critical frequency for sampling, so $\tau_\lambda(\boldsymbol{\eta})$ is small.

Dual to the minimization objective of Lemma 4.7.1, which allows us to certify upper bounds on the ridge leverage function, we can define a maximization objective allowing us to certify lower bounds:

Lemma 4.7.2. *The ridge leverage function can alternatively be defined as:*

$$\tau_\lambda(\boldsymbol{\eta}) = \max_{\boldsymbol{\alpha}\in\mathbb{C}^n} \frac{p(\boldsymbol{\eta})\cdot|\mathbf{z}(\boldsymbol{\eta})^*\boldsymbol{\alpha}|^2}{\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\|_{L_2(\mu)}^2 + \lambda\|\boldsymbol{\alpha}\|_2^2}. \tag{4.14}$$

Proof. The optimization problem (4.13) can equivalently be reformulated as,

$$\begin{aligned} \tau_\lambda(\boldsymbol{\eta}) = \text{minimum} \quad & \|y\|_{L_2(\mu)}^2 + \|\mathbf{u}\|_2^2 \\ y\in L_2(\mu); \quad & \mathbf{u}\in\mathbb{C}^n \\ \text{subject to:} \quad & \boldsymbol{\Phi}y + \sqrt{\lambda}\mathbf{u} = \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta}). \end{aligned}$$

First, we show that for any $\boldsymbol{\alpha}\in\mathbb{C}^n$, the argument of the minimization problem in (4.14) is no larger than $\tau_\lambda(\boldsymbol{\eta})$. That is because for the optimal solution to above optimization, namely $\bar{\mathbf{u}}$ and $\bar{y}$, we have:

$$\boldsymbol{\Phi}\bar{y} + \sqrt{\lambda}\bar{\mathbf{u}} = \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta}).$$

Hence,

$$\begin{aligned} \sqrt{p(\boldsymbol{\eta})}\cdot|\mathbf{z}(\boldsymbol{\eta})^*\boldsymbol{\alpha}| &= |\boldsymbol{\alpha}^*(\boldsymbol{\Phi}\bar{y} + \sqrt{\lambda}\bar{\mathbf{u}})| \\ &\le |\boldsymbol{\alpha}^*\boldsymbol{\Phi}\bar{y}| + \sqrt{\lambda}\left|\boldsymbol{\alpha}^*\bar{\mathbf{u}}\right| \\ &= |\langle\boldsymbol{\alpha}, \boldsymbol{\Phi}\bar{y}\rangle| + \sqrt{\lambda}|\boldsymbol{\alpha}^*\bar{\mathbf{u}}| \\ &= |\langle\boldsymbol{\Phi}^*\boldsymbol{\alpha}, \bar{y}\rangle_{L_2(\mu)}| + \sqrt{\lambda}|\boldsymbol{\alpha}^*\bar{\mathbf{u}}| \\ &\le \left\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\right\|_{L_2(\mu)}\cdot\left\|\bar{y}\right\|_{L_2(\mu)} + \sqrt{\lambda}\|\boldsymbol{\alpha}\|_2\cdot\|\bar{\mathbf{u}}\|_2, \end{aligned}$$

where the last inequality follows from Cauchy-Schwarz. By another application of Cauchy-Schwarz:

$$p(\boldsymbol{\eta})|\mathbf{z}(\boldsymbol{\eta})^*\boldsymbol{\alpha}|^2 \le \left(\left\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\right\|_{L_2(\mu)}^2 + \lambda\|\boldsymbol{\alpha}\|_2^2\right)\cdot\left(\left\|\bar{y}\right\|_{L_2(\mu)}^2 + \|\bar{\mathbf{u}}\|_2^2\right).$$

Therefore, for every $\boldsymbol{\alpha}\in\mathbb{C}^n$,

$$\frac{p(\boldsymbol{\eta})|\boldsymbol{\alpha}^*\mathbf{z}(\boldsymbol{\eta})|^2}{\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\|_{L_2(\mu)}^2 + \lambda\|\boldsymbol{\alpha}\|_2^2} \le \left\|\bar{y}\right\|_{L_2(\mu)}^2 + \|\bar{\mathbf{u}}\|_2^2 = \tau_\lambda(\boldsymbol{\eta}). \tag{4.15}$$

Now it is enough to show that for the optimal $\boldsymbol{\alpha}$ the objective of the dual problem (4.14) achieves the leverage score value $\tau_\lambda(\boldsymbol{\eta})$. First, note that for any $\boldsymbol{\alpha} \in \mathbb{C}^n$:

$$\begin{aligned}\left\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\right\|_{L_2(\mu)}^2 + \lambda\|\boldsymbol{\alpha}\|_2^2 &= \langle \boldsymbol{\Phi}^*\boldsymbol{\alpha}, \boldsymbol{\Phi}^*\boldsymbol{\alpha}\rangle_{L_2(\mu)} + \lambda\boldsymbol{\alpha}^*\boldsymbol{\alpha} \\ &= \langle \boldsymbol{\Phi}\boldsymbol{\Phi}^*\boldsymbol{\alpha}, \boldsymbol{\alpha}\rangle + \lambda\boldsymbol{\alpha}^*\boldsymbol{\alpha} \\ &= \boldsymbol{\alpha}^*(\mathbf{K} + \lambda\mathbf{I}_n)\boldsymbol{\alpha}.\end{aligned}$$

Now we show that for $\bar{\boldsymbol{\alpha}} = \sqrt{p(\boldsymbol{\eta})}(\mathbf{K} + \lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})$ the objective of (4.14) matches leverage score $\tau_\lambda(\boldsymbol{\eta})$. By substituting $\bar{\boldsymbol{\alpha}} = \sqrt{p(\boldsymbol{\eta})}(\mathbf{K} + \lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})$ we have:

$$\begin{aligned}\frac{p(\boldsymbol{\eta})|\mathbf{z}(\boldsymbol{\eta})^*\bar{\boldsymbol{\alpha}}|^2}{\|\boldsymbol{\Phi}^*\bar{\boldsymbol{\alpha}}\|_{L_2(\mu)}^2 + \lambda\|\bar{\boldsymbol{\alpha}}\|_2^2} &= \frac{p(\boldsymbol{\eta})^2|\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})|^2}{p(\boldsymbol{\eta})\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda\mathbf{I}_n)^{-1}(\mathbf{K} + \lambda\mathbf{I}_n)(\mathbf{K} + \lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})} \\ &= p(\boldsymbol{\eta})|\mathbf{z}(\boldsymbol{\eta})^*(\mathbf{K} + \lambda\mathbf{I}_n)^{-1}\mathbf{z}(\boldsymbol{\eta})| \\ &= \tau_\lambda(\boldsymbol{\eta}).\end{aligned}$$

□

The optimization problem in (4.14) asks us to exhibit a set of coefficients $\boldsymbol{\alpha} \in \mathbb{C}^n$, such that the Fourier domain representation of our point set weighted by these coefficients (i.e. $\boldsymbol{\Phi}^*\boldsymbol{\alpha}$) is concentrated at frequency $\boldsymbol{\eta}$ and hence $\frac{p(\boldsymbol{\eta})\cdot|\mathbf{z}(\boldsymbol{\eta})^*\boldsymbol{\alpha}|^2}{\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\|_{L_2(\mu)}^2}$ is large. Such $\boldsymbol{\alpha}$ certifies that $\boldsymbol{\eta}$ is a critical frequency for representing our point set and so $\tau_\lambda(\boldsymbol{\eta})$ must be large. The regularization term $\lambda\|\boldsymbol{\alpha}\|^2$, decreases the ridge leverage function when $p(\boldsymbol{\eta})$ is very small, i.e. when $\boldsymbol{\eta}$ has small weight in the Fourier transform of our kernel.

4.7.2 The Gaussian Kernel Leverage Function: Upper Bound

We start by applying Lemma 4.7.1 to prove a ridge leverage function upper bound for the Gaussian kernel. Again, to reduce clutter, we state the result for a fixed bandwidth $\sigma = (2\pi)^{-1}$.

Theorem 4.7.1. *Consider the d-dimensional Gaussian kernel with $\sigma = (2\pi)^{-1}$. For any integer n, any $0 < \lambda \leq \frac{n}{2}$ such that $d \leq 10n_\lambda$, and any radius $R > 0$, if $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ is contained in a ℓ_∞-ball of radius R (i.e $\left\|\mathbf{x}_i - \mathbf{x}_j\right\|_\infty \leq 2R$ for all $i, j \in [n]$), then for every $\|\boldsymbol{\eta}\|_\infty \leq 10\sqrt{\ln n_\lambda}$,*

$$\tau_\lambda(\boldsymbol{\eta}) \leq \left(6.2R + 1240\ln^{1.5} n_\lambda\right)^d + 1.$$

Combining the bound of Theorem 4.7.1 for $\boldsymbol{\eta}$ with $\|\boldsymbol{\eta}\|_\infty < 10\sqrt{\ln n_\lambda}$ and Proposition 1 for $\boldsymbol{\eta}$ outside this range immediately implies our improved sampling bound in Theorem 4.6.1.

Theorem 4.7.1 Proof Outline (Details and a full proof in Appendix C.2).

For simplicity we focus on the case of $d = 1$. Our proof for higher dimensions uses similar

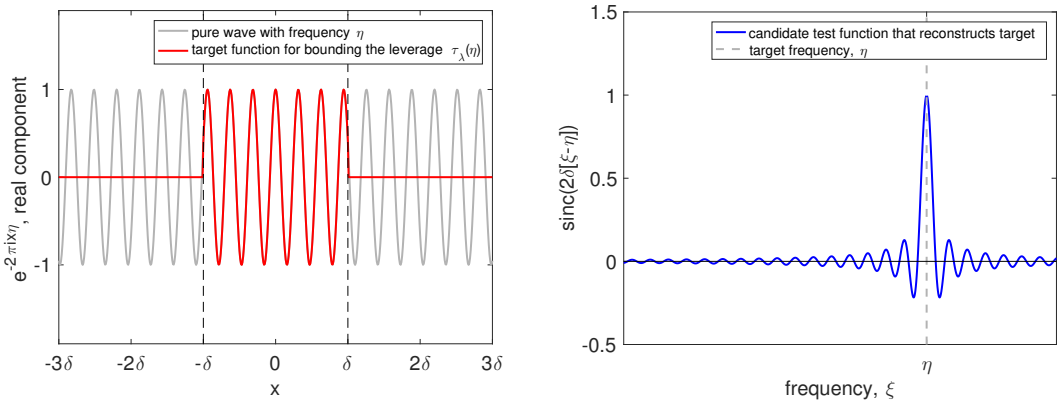


Figure 4.2 – To minimize $\|\mathbf{\Phi} y_\eta - \mathbf{z}(\eta)\|_2^2$, we can choose a test function $y_\eta(\xi)$ whose ($\mu(\cdot)$ weighted) Fourier transform $\mathbf{\Phi} y_\eta$ is the pure cosine $e^{-2\pi i x\eta}$ multiplied by the box function on $[-R,R]$. Specifically, $y_\eta(\xi)p(\xi)$ is a sinc function centered at η. Unfortunately, $\|y_\eta\|_{L_2(\mu)}^2$ is too large to get a good leverage function bound from Lemma 4.7.1. However, this construction is the starting point for our final test function, pictured in Figure 4.3.

ideas. To upper bound $\tau_\lambda(\eta)$ using Lemma 4.7.1 it suffices to exhibit any function $y_\eta \in L_2(\mu)$ (i.e. with bounded norm $\|y_\eta\|_{L_2(\mu)}^2$) such that, when reweighted by $\mu(\xi) = p(\xi)d\xi$, y_η's Fourier transform is close to the pure cosine target function $\mathbf{z}(\eta)$ on our datapoints. In general, the test function depends on η and hence our subscript notation $y_\eta(\cdot)$.

One simple attempt is $y_\eta(\xi) = \frac{1}{\sqrt{p(\eta)}}\delta(\eta - \xi)$ where $\delta(\cdot)$ is the Dirac delta function. This choice zeros out the first term of (4.13). However $\delta(\cdot)$ is not square integrable, $y_\eta \notin L_2(\mu)$, so the lemma cannot be used (the norm is unbounded). Another attempt is $y_\eta(\xi) = 0$, which zeros out the second term and recovers the trivial bound $\tau_\lambda(\eta) \leq \lambda^{-1}\|\sqrt{p(\eta)}\mathbf{z}(\eta)\|_2^2 = p(\eta)n_\lambda$ of Proposition 1.

We improve this bound by replacing the Dirac delta function at η with a 'soft spike' whose Fourier transform still looks approximately like a cosine wave on $[-R,R]$, and hence at our data points, which are bounded on this range. The smaller R is, the more spread out this function can be, and hence the smaller its norm $\|y_\eta\|_{L_2(\mu)}^2$, and the better the leverage function bound.

A natural idea is to consider the inverse Fourier transform of the cosine with frequency η restricted to the range $[-R,R]$ – i.e. multiplied by the box function on this range. It is well known that this is a sinc function with width $\frac{1}{2R}$, centered at η: $g_\eta(\xi) = 2R \cdot \operatorname{sinc}\big(2R(\xi-\eta)\big)$, where $\operatorname{sinc} x = \frac{\sin \pi x}{\pi x}$ (see Figure 4.2). If we set $y_\eta(\xi) = g_\eta(\xi) \cdot \frac{\sqrt{p(\eta)}}{p(\xi)}$, the μ weighted Fourier transform at $x_j \in [-R,R]$, $[\mathbf{\Phi} y_\eta]_j$, will be identical to the target $\mathbf{z}(\eta)_j$ and so again the first term of (4.13) will be 0. Unfortunately, $\|y_\eta\|_{L_2(\mu)}^2$ will still be too large. The reweighting function $\frac{1}{p(\xi)} = \sqrt{2\pi}e^{\xi^2/2}$ grows exponentially, while $\operatorname{sinc}\big(2R(\xi-\eta)\big)$ only falls off linearly, so y_η will have unbounded energy in the high frequencies.

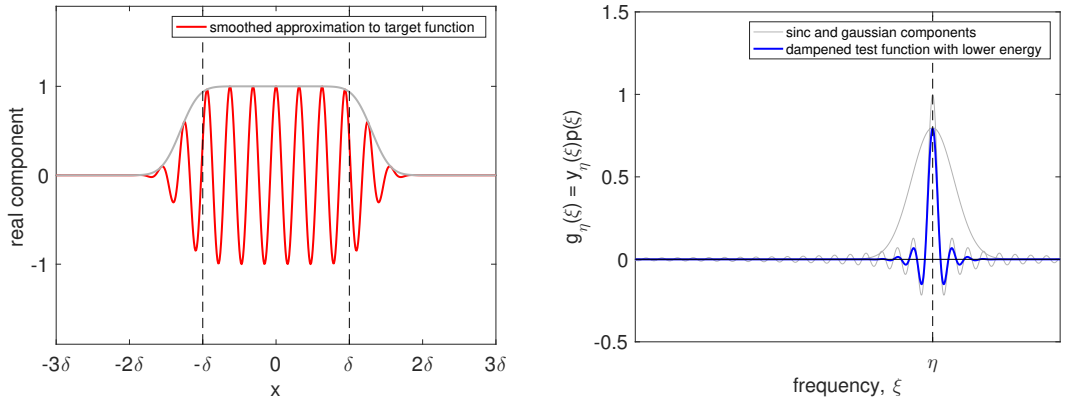


Figure 4.3 – In comparison to Figure 4.2, damping the sinc function with a Gaussian decreases the energy $\|y_\eta\|^2_{L_2(d\mu)}$ but does not significantly affect the Fourier transform on $[-R,R]$. $\mathbf{\Phi} y_\eta$ is a pure cosine with frequency η multiplied by a blurred box function and thus $(\mathbf{\Phi} y_\eta)_j \approx \mathbf{z}(\eta)_j$ for $x_j \in [-R,R]$. Accordingly, y_η is ideal for bounding the leverage function via Lemma 4.7.1.

To correct this issue, we dampen the sinc at higher frequencies by multiplying with a Gaussian, which decreases $\|y_\eta\|^2_{L_2(\mu)}$, but does not significantly affect the Fourier transform on $[-R,R]$.

Specifically, for some parameters u, v set $g_\eta(\xi)$ to be product of a Gaussian with standard deviation $1/u$ with a sinc function with width $1/v$, both centered at η. The corresponding Fourier transform $\hat{g}_\eta(x)$ is the convolution of a Gaussian with standard deviation u with a box of width v – i.e. a blurred box.

If we set $v = \Theta\left(R + u\sqrt{\log n_\lambda}\right)$ then the box, when centered at $x \in [-R,R]$ nearly covers the full mass of the Gaussian. Specifically, we have $1 - 1/n_\lambda^c \le |\hat{g}_\eta(x)| \le 1$ for $x \in [-R,R]$ and some large constant c. Since $g_\eta(\xi)$ is centered at η, $\hat{g}_\eta(x)$ is multiplied by the cosine wave $e^{-2\pi i x\eta}$, and so we have $(\mathbf{\Phi} y_\eta)_j = \sqrt{p(\eta)}\hat{g}_\eta(x_j) \approx \sqrt{p(\eta)}\mathbf{z}(\eta)_j$. Thus, when applying Lemma 4.7.1 to bound the leverage function, the first term of (4.13) will be negligible (see Figure 4.3).

Theorem 4.7.1 then follows from adjusting u to minimize $\|y_\eta\|^2_{L_2(\mu)}$ – balancing increased damping for large η with increased energy due to a more concentrated Gaussian. We eventually choose $u = \Theta(\log n_\lambda)$. Obtaining tight bounds and in particular achieving the right dependence on $\log n_\lambda$ requires several modifications, but the general intuition described above works!

4.7.3 The Gaussian Kernel Leverage Function: Lower Bound

Using the dual leverage function characterization in Lemma 4.7.2, we can give a near matching leverage function lower bound for the Gaussian kernel.

Theorem 4.7.2. *Consider the d-dimensional Gaussian kernel with $\sigma = (2\pi)^{-1}$. For every odd integer $m \ge 8\ln n_\lambda$, positive integer $d \le 8n_\lambda$, where $n = m^d$, every parameter $0 < \lambda \le \left(\frac{1}{2}\right)^{2d} \cdot \frac{n}{64}$, and every radius $60\ln^{3/2} n_\lambda \le R \le \frac{m}{80\sqrt{\ln n_\lambda}}$, there exist $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \in [-R,R]^d$ such that for every*

$\boldsymbol{\eta} \in \left[-10\sqrt{\ln n_\lambda}, 10\sqrt{\ln n_\lambda}\right]^d$,

$$\tau_\lambda(\boldsymbol{\eta}) \geq \frac{1}{128}\left(\frac{R}{3}\right)^d \cdot \frac{p(\boldsymbol{\eta})}{2p(\boldsymbol{\eta}) + (4R/3)^d n_\lambda^{-1}}.$$

Theorem 4.7.2 Proof Outline (Details and a full proof in Appendix C.3). The main idea of the proof is to use Lemma 4.7.2 to get a lower bound on $\tau_\lambda(\boldsymbol{\eta})$. Note that the expression given under the maximum in (4.14) provides a lower bound for any choice of $\boldsymbol{\alpha}$. However, we provide a judiciously chosen $\boldsymbol{\alpha}$ that is related to the test function $y_{\boldsymbol{\eta}} \in L_2(\mu)$ used in the proof of Theorem 4.7.1 which provides an upper bound on $\tau_\lambda(\boldsymbol{\eta})$. The choice of $y_{\boldsymbol{\eta}}$ in the proof of the upper bound is essentially a sinc function that is dampened by a Gaussian centered at $\boldsymbol{\eta}$. Due to the duality of the corresponding minimization and maximization problems in Lemma 4.7.1 and Lemma 4.7.2, respectively, the optimal $\boldsymbol{\alpha}$ must essentially be a scalar multiple of $\boldsymbol{\Phi} y_{\boldsymbol{\eta}}$, which is a (weighted) Fourier transform of $y_{\boldsymbol{\eta}}$ evaluated on the data points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$. Hence, we should intuitively choose $\boldsymbol{\alpha}$ to be the samples of $y_{\boldsymbol{\eta}}$ on the data points. Moreoever, to provide the tightest possible lower bound, we wish to choose our data points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ to be as spread apart as possible, as this corresponds to a higher statistical dimension (which corresponds to higher leverage scores on average). Thus, we choose our points to be evenly spaced points on a d-dimensional grid located inside an L_∞ ball of radius R around the origin.

4.7.4 Bounding the Statistical Dimension of Gaussian Kernel Matrices

Theorems 4.7.1 and 4.7.2 together imply a tight bound on the statistical dimension of Gaussian kernel matrices corresponding to bounded points sets (the proof appears in Appendix C.4):

Corollary 4.7.1. *Consider the d-dimensional Gaussian kernel with $\sigma = (2\pi)^{-1}$. For any positive integer n, parameter $0 < \lambda \leq \frac{n}{2}$, integer $1 \leq d \leq \frac{50 \ln n_\lambda}{\ln(10 \ln n_\lambda)}$, and any $R > 0$, if $\mathbf{x}_1, \ldots, \mathbf{x}_n \in [-R, R]^d$:*

$$\begin{aligned} s_\lambda(\mathbf{K}) &\leq \left(20\sqrt{\ln n_\lambda}\right)^d \left(\left(6.2R + 1240 \ln^{1.5} n_\lambda\right)^d + 1\right) \Big/ \Gamma(d/2+1) + 1 \\ &= O\left(\frac{(248R)^d \ln^{d/2} n_\lambda + (223 \ln n_\lambda)^{2d}}{\Gamma(d/2+1)}\right) \end{aligned}$$

Furthermore, if $n = m^d$ for some odd integer $m \geq 8 \ln n_\lambda$, and additionally $\lambda \leq \left(\frac{1}{2}\right)^{2d} \cdot \frac{n}{64}$, and radius $60 \ln^{3/2} n_\lambda \leq R \leq \frac{m}{80\sqrt{\ln n_\lambda}}$ there exists a set of points $\mathbf{x}_1, \ldots, \mathbf{x}_n \subseteq [-R, R]^d$ such that:

$$s_\lambda(\mathbf{K}) = \Omega\left(\frac{\left(\frac{\sqrt{\pi}R}{3}\sqrt{\ln \frac{n_\lambda}{(4R/3)^d}}\right)^d}{\Gamma(d/2+1)}\right).$$

5 Oblivious Sketching of High-degree Polynomial Kernels

This chapter is based on a joint work with Michael Kapralov, Rasmus Pagh, Ameya Velingker, and David Woodruff. It has been accepted to the 31st Annual ACM-SIAM Symposium on Discrete Algorithms (Ahle et al., 2020, SODA).

5.1 Introduction

Data dimensionality reduction, or *sketching*, is a common technique for quickly reducing the size of a large-scale optimization problem while approximately preserving the solution space, thus allowing one to instead solve a much smaller optimization problem, typically in a smaller amount of time. This technique has led to near-optimal algorithms for a number of fundamental problems in numerical linear algebra and machine learning, such as least squares regression, low rank approximation, canonical correlation analysis, k-means, and robust variants of these problems. In a typical instance of such a problem, one is given a large matrix $X \in \mathbb{R}^{d\times n}$ as input, and one wishes to choose a random map Π from a certain family of random maps and replace X with ΠX. As Π typically has many fewer rows than columns, ΠX compresses the original matrix X, which allows one to perform the original optimization problem on the much smaller matrix ΠX. For a survey of such techniques, we refer the reader to the survey by Woodruff (2014).

A key challenge in this area is to extend sketching techniques to kernel-variants of the above linear algebra problems. Suppose each column of X corresponds to an example while each row corresponds to a feature. Then the existing sketching algorithms require an explicit representation of X to be made available to the algorithm. This is unsatisfactory in many machine learning applications, since typically the actual learning is performed in a much higher (possibly infinite) dimensional feature space, by first mapping each column of X to a much higher dimensional space. Fortunately, due to the kernel trick, one need not ever perform this mapping explicitly; indeed, if the optimization problem at hand only depends on inner product information between the input points, then the kernel trick allows one to quickly compute the inner products of the high dimensional transformations of the input

points, without ever explicitly computing the transformation itself. However, evaluating kernel matrices easily becomes a bottleneck in algorithms that rely on the kernel trick because primitives such as kernel PCA or kernel ridge regression generally take prohibitively large quadratic space and (at least) quadratic time, as kernel matrices are usually dense. There are a number of recent works which try to improve the running times of kernel methods; we refer the reader to the recent work of Musco and Musco (2017) and the references therein. A natural question is whether it is possible to instead apply sketching techniques on the high-dimensional feature space without ever computing the high-dimensional mapping.

For the important case of *polynomial kernel*, such sketching techniques are known to be possible[1]. This was originally shown by Pham and Pagh (2013) in the context of kernel support vector machines, using the TensorSketch technique for compressed matrix multiplication due to Pagh (2013). This was later extended in (Avron et al., 2014) to a wide array of kernel problems in linear algebra, including kernel low-rank approximation, principal component analysis, principal component regression, and canonical correlation analysis.

The running times of the algorithms above, while nearly linear in the number of non-zero entries of the input matrix X, depend *exponentially* on the degree q of the polynomial kernel. For example, suppose one wishes to perform low-rank approximation on A, the matrix obtained by replacing each column of X with its kernel-transformed version. One would like to express $A \approx UV$, where $U \in \mathbb{R}^{d^p \times k}$ and $V \in \mathbb{R}^{k \times n}$. Writing down U explicitly is problematic, since the columns belong to the much higher d^p-dimensional space. Instead, one can express UV implicitly via column subset selection, by expressing it as a $AZZ^\top$ and then outputting Z. Here Z is an $n \times k$ matrix. In (Avron et al., 2014), an algorithm running in $\mathrm{nnz}(X) + (n+d)\,\mathrm{poly}\left(3^p, k, \epsilon^{-1}\right)$ time was given for outputting such Z with the guarantee that $\left\|A - AZZ^\top\right\|_F^2 \leq (1+\epsilon)\|A - A_k\|_F^2$ with constant probability, where A_k is the best rank-k approximation to A. Algorithms with similar runtimes were proposed for PCR and CCA. The main message here is that all analyses of all existing sketches require the sketch Π to have at least 3^p rows in order to guarantee their correctness. Moreover, the existing sketches work with constant probability and no high probability result is known for the polynomial kernel.

The main drawback of previous works on sketching the polynomial kernel is the exponential dependence on the kernel degree p in the sketching dimension and consequently in the running time. Ideally, one would like a polynomial dependence. This is especially useful for the application of approximating the Gaussian kernel by a sum of polynomial kernels of various degrees, for which large values of p, e.g., $p = \mathrm{polylog}\, n$ are used (Cotter et al., 2011). This raises the main question of our work:

Is it possible to desing a data oblivious sketch with a sketching dimension (and, hence, running time) that is not exponential in p for the above applications in the context of the polynomial kernel?

[1] The lifting function corresponding to the polynomial kernel of degree p maps $x \in \mathbb{R}^d$ to $\phi(x) \in \mathbb{R}^{d^p}$, where $\phi(x)_{i_1,i_2,\ldots,i_p} = x_{i_1} x_{i_2} \cdots x_{i_p}$, for every $i_1, i_2, \ldots, i_p \in \{1,2,\ldots,d\}$

While we answer the above question, we also investigate it in a more general context, namely, that of regularization. In many machine learning problems, it is crucial to regularize so as to prevent overfitting or ill-posed problems. Sketching and related sampling-based techniques have also been extensively applied in this setting. For a small sample of such work see (Rahimi and Recht, 2008; Alaoui and Mahoney, 2015; Pilanci and Wainwright, 2015; Musco and Musco, 2017; Avron et al., 2017b,a,c, 2019). As an example application, in ordinary least squares regression one is given a $d \times n$ matrix A, and a $d \times 1$ vector b, and one seeks to find a $y \in \mathbb{R}^n$ that minimizes $\|Ay - b\|_2^2$. In ridge regression, we instead seek a y so as to minimize $\|Ay - b\|_2^2 + \lambda \|y\|_2^2$, for a parameter $\lambda > 0$. Intuitively, if λ is much larger than the operator norm $\|A\|_{\text{op}}$ of A, then a good solution is obtained simply by setting $y = \{0\}^d$. On the other hand, if $\lambda = 0$, the problem just becomes an ordinary least squares regression. In general, the *statistical dimension* (or *effective degrees of freedom*), s_λ, captures this tradeoff, defined as $s_\lambda \stackrel{\text{def}}{=} \sum_{i=1}^{d} \frac{\lambda_i(A^\top A)}{\lambda_i(A^\top A)+\lambda}$, where $\lambda_i(A^\top A)$ is the i^{th} eigenvalue of $A^\top A$. Note that the statistical dimension is always at most $\min(n, d)$, but in fact can be much smaller. A key example of its power is that for ridge regression, it is known (Avron et al., 2017b) that if one chooses a random Gaussian matrix Π with $O(s_\lambda/\epsilon)$ rows, and if y^* is the minimizer to $\|\Pi Ay - \Pi b\|_2^2 + \lambda \|y\|_2^2$, then $\|Ay^* - b\|_2^2 + \lambda \|y^*\|_2^2 \leq (1+\epsilon) \min_{y'} (\|Ay' - b\|_2^2 + \lambda \|y'\|_2^2)$. Note that for ordinary regression ($\lambda = 0$) one would need that Π has $\Omega(\text{rank}(A)/\epsilon)$ rows (Clarkson and Woodruff, 2009). Another drawback of existing sketches for the polynomial kernel is that their running time and target dimension depend at least quadratically on s_λ and no result is known with optimal linear dependence on s_λ. We also ask if the exponential dependence on p is avoidable in the *regularized* setting:

Is it possible to obtain sketching dimension bounds and running times that are not exponential in p in the context of regularization? Moreover, is it possible to obtain a running time that depends only linearly on s_λ?

5.1.1 Our contributions

In this chapter, we answer the above questions in the affirmative. In other words, for each of the aforementioned applications, our algorithm depends only *polynomially* on p. We state these applications as corollaries of our main results, which concern approximate matrix product and subspace embeddings. In particular, we devise a new distribution on oblivious linear maps $\Pi \in \mathbb{R}^{m \times d^p}$ (i.e., a randomized family of maps that does not depend on the dataset X), so that for any fixed $X \in \mathbb{R}^{d \times n}$, it satisfies the approximate matrix product and subspace embedding properties. These are the key properties needed for kernel low-rank approximation. We remark that our *data oblivious sketching* is greatly advantageous to data dependent methods because it results in a one-round distributed protocol for kernel low-rank approximation (Kannan et al., 2014). We show that our oblivious linear map $\Pi \in \mathbb{R}^{m \times d^p}$ has the following key properties:

Oblivious Subspace Embeddings (OSEs). Given $\varepsilon > 0$ and an n-dimensional subspace $E \subseteq \mathbb{R}^d$, we say that $\Pi \in \mathbb{R}^{m\times d}$ is an ε-subspace embedding for E if $(1-\varepsilon)\|x\|_2 \le \|\Pi x\|_2 \le (1+\varepsilon)\|x\|_2$ for all $x \in E$. In this chapter we focus on *Oblivious Subspace Embeddings (OSEs)* in the regularized setting. In order to define (regularized) OSEs, we first recall the notion of *statistical dimension*:

Definition 5.1.1 (Statistical Dimension)**.** Given $\lambda \ge 0$, for every positive semidefinite matrix $K \in \mathbb{R}^{n\times n}$, we define the λ-statistical dimension of K to be

$$s_\lambda(K) \stackrel{\text{def}}{=} \operatorname{tr}\left(K(K+\lambda I_n)^{-1}\right).$$

Now, we can define the notion of an oblivious subspace embedding (OSE):

Definition 5.1.2 (Oblivious Subspace Embedding (OSE))**.** Given $\varepsilon, \delta, \mu > 0$ and integers $d, n \ge 1$, an $(\varepsilon, \delta, \mu, d, n)$-*Oblivious Subspace Embedding (OSE)* is a distribution $\mathscr{D}$ over $m \times d$ matrices (for arbitrary m) such that for every $\lambda \ge 0$, every $A \in \mathbb{R}^{d\times n}$ with λ-statistical dimension $s_\lambda(A^\top A) \le \mu$, the following holds,

$$\Pr_{\Pi\sim\mathscr{D}}\left[(1-\epsilon)(A^\top A + \lambda I_n) \preceq (\Pi A)^\top \Pi A + \lambda I_n \preceq (1+\epsilon)(A^\top A + \lambda I_n)\right] \ge 1-\delta. \tag{5.1}$$

The goal is to have the target dimension m small so that Π provides dimensionality reduction. If we consider the non-oblivious setting where we allow the sketch matrix Π to depend on A, then as we saw in Chapters 3 and 4, we can achieve a target dimension of $m \approx s_\lambda(A^\top A)$ by leverage scores sampling, which is essentially optimal (see Theorem 3.6.2). But as we discussed the importance of oblivious embeddings, the ultimate goal is to get an oblivious subspace embedding with target dimension of $m \approx s_\lambda(A^\top A)$.

Approximate Matrix Product. We formally define this property in the following definition.

Definition 5.1.3 (Approximate Matrix Product)**.** Given $\varepsilon, \delta > 0$, we say that a distribution $\mathscr{D}$ over $m \times d$ matrices has the (ε, δ)-*Approximate Matrix Product* property if for every $C, D \in \mathbb{R}^{d\times n}$,

$$\Pr_{\Pi\sim\mathscr{D}}\left[\|C^\top \Pi^\top \Pi D - C^\top D\|_F \le \varepsilon \|C\|_F \|D\|_F\right] \ge 1-\delta.$$

Now we present our main theorems, which provide the aforementioned guarantees. Our first theorem optimizes the runtime's dependence on the degree p of the kernel.[2]

Theorem 5.1.1. *For every positive integers n, p, d, every $\varepsilon, s_\lambda > 0$, there exists a distribution on linear sketches $\Pi^p \in \mathbb{R}^{m\times d^p}$ such that:* **(1)** *If $m = \Omega\left(p s_\lambda^2 \epsilon^{-2}\right)$, then Π^p is an $(\varepsilon, 1/10, s_\lambda, d^p, n)$-oblivious subspace embedding as in Definition 5.1.2.* **(2)** *If $m = \Omega\left(p\varepsilon^{-2}\right)$, then Π^p has the $(\varepsilon, 1/10)$-approximate matrix product property as in Definition 5.1.3.*
Moreover, for any $X \in \mathbb{R}^{d\times n}$, if $A \in \mathbb{R}^{d^p\times n}$ is the matrix whose columns are obtained by the p-fold self-tensoring of each column of X then $\Pi^p A$ can be computed in time $\widetilde{O}\left(pnm + p\,\mathrm{nnz}(X)\right)$.

[2] Throughout this chapter, the notations $\widetilde{O}, \widetilde{\Omega}, \widetilde{\Theta}$ suppress $\mathrm{polylog}(nd/\varepsilon)$ factors.

Our next theorem optimally achieves linear dependence on the statistical dimension s_λ.

Theorem 5.1.2. *For every positive integers p, d, n, every $\varepsilon, s_\lambda > 0$, there exists a distribution on linear sketches $\Pi^p \in \mathbb{R}^{m \times d^p}$ which is an $(\varepsilon, 1/\operatorname{poly} n, s_\lambda, d^p, n)$-oblivious subspace embedding as in Definition 5.1.2, provided that the integer m satisfies $m = \widetilde{\Omega}\left(p^4 s_\lambda / \epsilon^2\right)$.*
Moreover, for any $X \in \mathbb{R}^{d \times n}$, if $A \in \mathbb{R}^{d^p \times n}$ is the matrix whose columns are obtained by the p-fold self-tensoring of each column of X then $\Pi^p A$ can be computed in time $\widetilde{O}\left(pnm + p^5 \epsilon^{-2} \operatorname{nnz}(X)\right)$.

We can immediately apply these theorems to *kernel ridge regression* with respect to the polynomial kernel of degree p. In this problem, we are given a regularization parameter $\lambda \geq 0$, a $d \times n$ matrix X, and vector $b \in \mathbb{R}^n$ and would like to find a $y \in \mathbb{R}^n$ so as to minimize $\left\| A^\top A y - b \right\|_2^2 + \lambda \| A y \|_2^2$, where $A \in \mathbb{R}^{d^p \times n}$ is the matrix obtained by applying self tensoring of degree p to each column of X. To solve this problem via sketching, we choose a random matrix Π^p according to the theorems above and compute $\Pi^p A$. We then solve the sketched ridge regression problem which seeks to minimize $\left\| (\Pi^p A)^\top \Pi^p A y - b \right\|_2^2 + \lambda \| \Pi^p A y \|_2^2$ over y. By the OSE property, we have $\left\| (\Pi^p A)^\top \Pi^p A y - b \right\|_2^2 + \lambda \| \Pi^p A y \|_2^2 = (1 \pm \epsilon) \left(\left\| A^\top A y - b \right\|_2^2 + \lambda \| A y \|_2^2 \right)$ simultaneously for all $y \in \mathbb{R}^n$; thus, solving the sketched ridge regression problem gives a $(1 \pm \epsilon)$-approximation to the original problem. If we apply Theorem 5.1.1, then the number of rows of Π^p needed to ensure success with probability $9/10$ is $\Theta\left(p s_\lambda^2 \epsilon^{-2}\right)$. The running time to compute $\Pi^p A$ is $\widetilde{O}\left(p^2 s_\lambda^2 \varepsilon^{-2} n + p \operatorname{nnz}(X)\right)$, after which a ridge regression problem can be solved in $O\left(n s_\lambda^4 / \epsilon^4\right)$ time via an exact closed-form solution for linear ridge regression. An alternative approach to obtaining a very high-accuracy approximation is to use the sketched kernel as a preconditioner to solve the original ridge regression problem, which improves the dependence on ε to $\log(1/\varepsilon)$ (Avron et al., 2017a). To obtain a high probability of success along with a near-optimal target dimension that depends only linearly on the statistical dimension, we can instead apply Theorem 5.1.2, which would allow us to compute the sketched matrix $\Pi^p A$ in $\widetilde{O}\left(p^5 s_\lambda \varepsilon^{-2} n + p^5 \varepsilon^{-2} \operatorname{nnz}(X)\right)$ time. This is the first sketch to achieve the optimal dependence on s_λ for the polynomial kernel. Importantly, both running times are polynomial in p, whereas all previously known methods incurred running times that were exponential in p.

Although there has been much work on sketching methods for kernel approximation which nearly achieve the optimal target dimension $m \approx s_\lambda$, such as Nyström sampling (Musco and Musco, 2017), all known methods are data-dependent unless strong conditions are assumed about the kernel matrix (small condition number or incoherence). Data oblivious methods provide nice advantages, such as one-round distributed protocols and single-pass streaming algorithms. However, for kernel methods they are poorly understood and previously had worse theoretical guarantees than data-dependent methods. Furthermore, note that the Nyström method requires to sample at least $m = \Omega(s_\lambda)$ landmarks to satisfy the subspace embedding property even given an oracle access to the exact leverage scores distribution. This results in a runtime of $\Omega\left(s_\lambda^2 n + s_\lambda \operatorname{nnz}(X)\right)$. Whereas our method achieves a target dimension that nearly matches the best dimension possible with data-dependent Nyström method and with strictly better running time of $\widetilde{O}(n s_\lambda + \operatorname{nnz}(X))$ (assuming $p = \operatorname{polylog} n$). Therefore, for a large range of parameters, our sketch runs in input sparsity time whereas the Nyström methods are slower

by an s_λ factor at best.

Application: Polynomial Kernel Rank-k Approximation. Approximate matrix product and subspace emebedding are key properties that imply efficient algorithms for rank-k kernel approximation. The following corollary of Theorem 5.1.1 immediately follows from (Avron et al., 2014, Theorem 6).

Corollary 5.1.1 (Rank-k Approximation)**.** *For every positive integers k, n, p, d, every $\varepsilon > 0$, any $X \in \mathbb{R}^{d\times n}$, if $A \in \mathbb{R}^{d^p\times n}$ is the matrix whose columns are obtained by the p-fold self-tensoring of each column of X then there exists an algorithm that finds an $n \times k$ matrix V in time $O\left(p\,\mathrm{nnz}(X)\right) + \mathrm{poly}\left(k, p, \varepsilon^{-1}\right)$ such that with probability $9/10$,*

$$\left\|A - AVV^\top\right\|_F^2 \le (1+\varepsilon) \min_{\substack{U\in\mathbb{R}^{d^p\times n}\\ rank(U)=k}} \|A-U\|_F^2.$$

Note that this corollary improves the runtime of (Avron et al., 2014) by exponential factors in the polynomial kernel's degree p.

Additional Applications. Our results also imply improved bounds for each of the applications in Avron et al. (2014), including canonical correlation analysis (CCA), and principal component regression (PCR). Importantly, we obtain the first sketching-based solutions for these problems with running time polynomial rather than exponential in p.

Oblivious Subspace Embedding for the Gaussian Kernel. One very important implication of our result is an OSE of the Gaussian kernel. Most work in this area is related to the Random Fourier Features method (Rahimi and Recht, 2008). As shown in Chapter 4, one requires $\Omega(n)$ samples of the classic Fourier features to obtain a subspace embedding for the Gaussian kernel, while a modified distribution for sampling frequencies yields provably better performance. Our proposed sketch for the Gaussian kernel improves upon Theorem 4.6.1, which has an exponential dependence on the dimension d. We for the first time, embed the Gaussian kernel with a target dimension that depends only linearly on the statistical dimension of the kernel and is not exponential in the dimensionality of the data-point.

Theorem 5.1.3. *For every $r > 0$, every positive integers n, d, and every $X \in \mathbb{R}^{d\times n}$ such that $\|x_i\|_2 \le r$ for all $i \in [n]$, where x_i is the i^{th} column of X, suppose $G \in \mathbb{R}^{n\times n}$ is the Gaussian kernel matrix – i.e., $G_{j,k} = e^{-\|x_j - x_k\|_2^2/2}$ for all $j, k \in [n]$. There exists an algorithm that computes $S_g(X) \in \mathbb{R}^{m\times n}$ in time $\widetilde{O}\left(q^6\epsilon^{-2}ns_\lambda + q^6\epsilon^{-2}\,\mathrm{nnz}(X)\right)$ such that for every $\varepsilon, \lambda > 0$,*

$$\Pr_{S_g}\left[(1-\epsilon)(G+\lambda I_n) \preceq \left(S_g(X)\right)^\top S_g(X) + \lambda I_n \preceq (1+\epsilon)(G+\lambda I_n)\right] \ge 1 - \frac{1}{\mathrm{poly}(n)},$$

where $m = \widetilde{\Theta}\left(q^5 s_\lambda/\epsilon^2\right)$ and $q = \Theta(r^2 + \log\frac{n}{\epsilon\lambda})$ and s_λ is λ-statistical dimension of G.

We remark that for datasets with radius $r = \text{polylog}\, n$ even if one has oracle access to the exact leverage scores for Fourier features of the Gaussian kernel, in order to get subspace embedding guarantee one needs to use $m = \Omega(s_\lambda)$ features which requires $\Omega(s_\lambda \operatorname{nnz}(X))$ operations to compute. Whereas our embedding in Theorem 5.1.3 runs in time $\widetilde{O}(ns_\lambda + \operatorname{nnz}(X))$. Therefore, for a large range of parameters, our Gaussian sketch runs in near input sparsity time while the Fourier features method is, at best, slower by an s_λ factor.

5.1.2 Technical overview

Our goal is to design a sketching matrix Π^p that satisfies the oblivious subspace embedding property with an optimal embedding dimension and which can be efficiently applied to vectors of the form $x^{\otimes p} \in \mathbb{R}^{d^p}$[3]. We start by describing some natural approaches to this problem (some of which have been used before), and show why they incur an exponential loss in the degree of the polynomial kernel. We then present our sketch and outline our proof of its correctness.

We first discuss two natural approaches to tensoring classical sketches, namely the Johnson-Lindenstrauss transform and the CountSketch. We show that both lead to an exponential dependence of the target dimension on p and then present our new approach.

Tensoring the Johnson-Lindenstrauss Transform. Perhaps the most natural approach to designing a sketch Π^p is the idea of tensoring p independent Johnson-Lindenstrauss matrices. Specifically, let m be the target dimension. For every $r = 1, \dots, p$ let $M^{(r)}$ denote an $m \times d$ matrix with iid uniformly random ± 1 entries, and let the sketching matrix $M \in \mathbb{R}^{m \times d^p}$ be

$$M = \frac{1}{\sqrt{m}} M^{(1)} \bullet \ldots \bullet M^{(p)},$$

where $\bullet$ stands for the operation of tensoring the rows of matrices $M^{(r)}$ (see Definition 5.2.4). This would be a very efficient matrix to apply, since for every $j = 1, \dots, m$ the j-th entry of $Mx^{\otimes p}$ is exactly $\prod_{r=1}^{p} \left[M^{(r)} x\right]_j$, which can be computed in time $O(p \operatorname{nnz}(x))$, giving overall evaluation time $O(pm \operatorname{nnz}(x))$. One would hope that $m = O(\varepsilon^{-2} \log n)$ would suffice to ensure that $\|Mx^{\otimes p}\|_2^2 = (1 \pm \epsilon)\|x^{\otimes q}\|_2^2$. However, this is not true: it is shown in (Ahle et al., 2020) that one must have $m = \Omega\left(\varepsilon^{-2} 3^p \log n / p + \varepsilon^{-1} (\log n / p)^p\right)$ in order to preserve the norm with high probability. Thus, the dependence on degree p of the polynomial kernel must be exponential.

Tensoring of CountSketch (TensorSketch). Pham and Pagh (2013) introduced the following tensorized version of CountSketch. For every $i = 1, \dots, p$ let $h_i : [d] \to [m]$ denote a random hash function, and $\sigma_i : [d] \to [m]$ a random sign function. Then let $S : \mathbb{R}^{d^p} \to \mathbb{R}^m$ be defined by

$$S_{r,(j_1,\dots,j_p)} := \sigma(i_1) \cdots \sigma(i_p)\, \mathbb{1}[h_1(i_1) + \ldots h_p(i_p) = r]$$

[3] Tensor product of x with itself p times.

for $r = 1, \ldots, m$. For every $x \in \mathbb{R}^d$ one can compute $Sx^{\otimes p}$ in time $O(pm \log m + p\,\mathrm{nnz}(x))$. Since the time to apply the sketch only depends linearly on the dimension p (due to the Fast Fourier Transform) one might hope that the dependence of the sketching dimension on p is polynomial. However, this turns out to not be the case: the argument in Avron et al. (2014) implies that $m = O\left(3^p s_\lambda^2\right)$ suffices to construct a subspace embedding for a matrix with regularization λ and statistical dimension s_λ, and the lower bound in (Ahle et al., 2020) shows that exponential dependence on p is necessary.

Our Approach: Recursive Tensoring. The initial idea behind our sketch is as follows. To apply our sketch Π^p to $x^{\otimes p}$, for $x \in \mathbb{R}^d$, we first compute the sketches $T_1 x, T_2 x, \ldots, T_p x$ for independent sketching matrices $T_1, \ldots, T_p \sim T_{\text{base}}$ – see the leaves of the sketching tree in Figure 5.1. Note that we choose these sketches as CountSketch (Charikar et al., 2002; Charikar, 2002) or OSNAP (Nelson and Nguyên, 2013) to ensure that the leaf sketches can be applied in time proportional to the number of nonzeros of the input data (in the case of OSNAP this is true up to polylogarithimic factors).

Each of these is a standard sketching matrix mapping d-dimensional vectors to m-dimensional vectors for some common value of m. We refer the reader to the survey by Woodruff (2014). The next idea is to choose new sketching matrices $S_1, S_2, \ldots, S_{p/2} \sim S_{\text{base}}$, mapping m^2-dimensional vectors to m-dimensional vectors and apply S_1 to $(T_1 x) \otimes (T_2 x)$, as well as apply S_2 to $(T_3 x) \otimes (T_4 x)$, and so on, applying $S_{p/2}$ to $(T_{p-1} x) \otimes (T_p x)$. These sketches are denoted by S_{base} – see internal nodes of the sketching tree in Figure 5.1. We note that in order to ensure efficiency of our construction (in particular, running time that depends only linearly on the statistical dimension s_λ) we must choose S_{base} as a sketch that can be computed on tensored data without explicitly constructing the actual tensored input, i.e., S_{base} supports fast matrix vector product on tensor product of vectors. We use either TensorSketch (for results that work with constant probability) and a new variant of the Subsampled Randomized Hadamard Transform SRHT (Ailon and Chazelle, 2006) which supports fast multiplication for the tensoring of two vectors (for high probability bounds) – we call the last sketch TensorSRHT.

At this point we have reduced our number of input vectors from p to $p/2$, and the dimension is m, which will turn out to be roughly s_λ. We have made progress, as we now have fewer vectors each in roughly the same dimension we started with. After $\log_2 p$ levels in the tree we are left with a single output vector.

Intuitively, the reason that this construction avoids an exponential dependence on p is that at every level in the tree we use target dimension m larger than the statistical dimension of our matrix by a factor polynomial in p. This ensures that the accumulation of error is limited, as the total number of nodes in the tree is $O(p)$. This is in contrast to the direct approaches discussed above, which use a rather direct tensoring of classical sketches, thereby incurring an exponential dependence on p due to dependencies that arise.

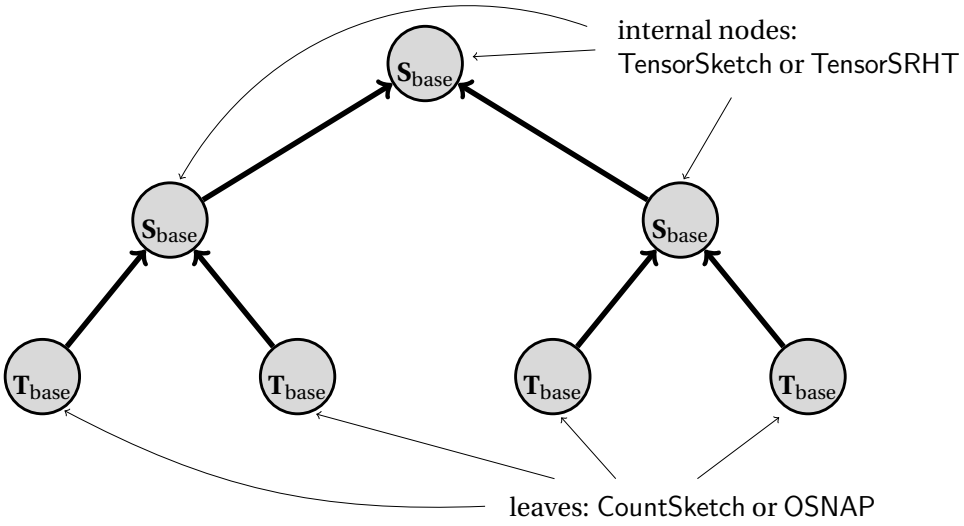


Figure 5.1 – S_{base} is chosen from the family of sketches which support fast matrix-vector product for tensor inputs, e.g., TensorSketch and TensorSRHT. The T_{base} is chosen from the family of sketches which operate in input sparsity time, e.g., CountSketch and OSNAP.

Showing Our Sketch is a Subspace Embedding. In order to show that our recursive sketch is a subspace embedding, we need to argue it preserves norms of arbitrary vectors in $\mathbb{R}^{d^p}$, not only vectors of the form $x^{\otimes p}$, i.e., p-fold self-tensoring of d-dimensional vectors[4]. Indeed, all known methods for showing the subspace embedding property at the very least argue that the norms of each of the columns of an orthonormal basis for the subspace in question are preserved (Woodruff, 2014). While our subspace may be formed by the span of vectors which are tensor products of p d-dimensional vectors, we are not guaranteed that there is an orthonormal basis of this form. Thus, we first observe that our mapping is indeed linear over $\mathbb{R}^{d^p}$, making it well-defined on the elements of any basis for our subspace, and hence our task essentially reduces to proving that our mapping preserves norms of arbitrary vectors in $\mathbb{R}^{d^p}$.

We present two approaches to analyzing our construction. One is based on the idea of propagating moment bounds through the sketching tree, and results in a nearly linear dependence of the sketching dimension m on the degree p of the polynomial kernel, at the expense of a quadratic dependence on the statistical dimension s_λ. This approach is presented in Section 5.4. The other approach achieves the (optimal) linear dependence on s_λ, albeit at the expense of a worse polynomial dependence on p. This approach uses sketches that succeed with high probability, and uses matrix concentration bounds.

Optimizing the dependence on the degree p. We analyze our recursively constructed sketch by showing how second moment bounds can be propagated through the tree structure of the

[4] $x^{\otimes p}$ denotes $\underbrace{x \otimes x \cdots \otimes x}_{p \text{ terms}}$, the p-fold self-tensoring of x.

sketch. This analysis is presented in Section 5.4, and results in the proof of Theorem 5.1.1. The analysis obtained this way gives particularly sharp dependence on p. The idea is to consider the sketch matrix $\Pi \in \mathbb{R}^{m \times d^p}$ that we have described it recursively above. This matrix could in principle be applied to any vector $x \in \mathbb{R}^{d^p}$ (though it would be slow to realise). We can nevertheless show that this matrix has the $(\varepsilon, \delta, 2)$-JL Moment Property, which is for parameters $\varepsilon, \delta \in [0,1]$, and every $x \in \mathbb{R}^d$ the statement $\mathbb{E}\left[\left|\|\Pi x\|_2^2 - 1\right|^2\right] \leq \left(\varepsilon \|x\|_2^2\right)^2 \delta$.

It can be shown that Π is built from our various S_{base} and T_{base} matrices using three different operations: multiplication, direct sum, and row-wise tensoring. In other words, it is sufficient to show that if Q and Q' both have the $(\varepsilon, \delta, 2)$-JL Moment Property, then so does QQ' and $Q \oplus Q'$. This turns out to hold for $Q \oplus Q'$, but QQ' is more tricky. (Here $\oplus$ is the direct sum. See Section 5.2 on notation.) For multiplication, a simple union bound allows us to show that $Q^{(1)}Q^{(2)} \cdots Q^{(p)}$ has the $(p\varepsilon, p\delta, 2)$-JL Moment Property. This would unfortunately mean a factor of p^2 in the final dimension. The union bound is clearly suboptimal, since implicitly it assumes that all the matrices conspire to either shrink or increase the norm of a vector, while in reality with independent matrices, we should get a random walk on the real line. Using an intricate decoupling argument, we show that this is indeed the case, and that $Q^{(1)}Q^{(2)} \cdots Q^{(p)}$ has the $(\sqrt{p}\varepsilon, \delta, 2)$-JL Moment Property, saving a factor of p in the output dimension.

Optimizing the dependence on s_λ. Our proof of Theorem 5.1.2 relies on instantiating our framework with OSNAP at the leaves of the tree (T_{base}) and a novel version of the SRHT that we refer to as TensorSRHT at the internal nodes of the tree. We outline the analysis of why our sketch preserves norm of an arbitrary vector $y \in \mathbb{R}^{d^p}$. In the bottom level of the tree, we can view our sketch as $T_1 \times T_2 \times \cdots \times T_p$, where $\times$ for denotes the tensor product of matrices (see Definition 5.2.2). Then, we can reshape y to a $d^{q-1} \times d$ matrix Y, such that the entries of $T_1 \times T_2 \times \cdots \times T_p y$ are in bijective correspondence with those of $T_1 \times T_2 \times \cdots \times T_{p-1} Y T_p^\top$. By definition of T_p, it preserves the Frobenius norm of Y, and consequently, we can replace Y with $Y T_p^\top$. We next look at $(T_1 \times T_2 \times \cdots \times T_{p-2}) Z (I_d \times T_{p-1}^\top)$, where Z is the $d^{p-2} \times d^2$ matrix with entries in bijective correspondence with those of $Y T_p^\top$. Then we know that T_{p-1} preserves the Frobenius norm of Z. Iterating in this fashion, we can show the first layer of our tree preserves the norm of y, by union bounding over p events – i.e., each sketch preserves the norm of an intermediate matrix. The core of the analysis consists of applying spectral concentration bounds based analysis to sketches that act on blocks of the input vector in a correlated fashion. We give the details in Section 5.5.

Sketching the Gaussian kernel. Our techniques yield the first oblivious sketching method for the Gaussian kernel with target dimension that does not depend exponentially on the dimensionality of the input data points. The main idea is to Taylor expand the Gaussian function and apply our sketch to the polynomial terms in the expansion. It is crucial here that the target dimension of our sketch depends only polynomially on the degree of the polynomial kernel, as otherwise we would not be able to truncate the Taylor expansion sufficiently far

in the tail (the number of terms in the Taylor expansion depends logarithmically on the dataset size). Overall, our subspace embedding for the Gaussian kernel has optimal target dimension up to logarithmic factors in the dataset size and is the first to run in near input sparsity time $\widetilde{O}(\text{nnz}(X))$ for datasets with polylogarithmic radius. The result is summarized in Theorem 5.1.3, and the analysis is presented in Section 5.6.

5.1.3 Related work

Work related to sketching of tensors and explicit kernel embeddings is found in fields ranging from pure mathematics to physics and machine learning. Hence we only compare ourselves with the most common types.

Johnson-Lindenstrauss Transform A cornerstone result in the field of subspace embeddings is the Johnson-Lindenstrauss lemma (Johnson and Lindenstrauss, 1984). It has been shown in (Clarkson and Woodruff, 2013; Cohen et al., 2016b) that the Johnson-Lindenstrauss Lemma implies that for any r-dimensional subspace $U \subseteq \mathbb{R}^d$ there exists a subspace embedding with $m = O(\varepsilon^{-2} r)$.

It is not enough to know that the subspace embedding exists, we also need to find the dimensionality reduction map, and we want the map to be applicable to the data quickly. Achlioptas (2003) showed that if $\Pi \in \mathbb{R}^{m \times d}$ is random matrix with i.i.d. entries where $\Pi_{i,j} = 0$ with probability 2/3, and otherwise $\Pi_{i,j}$ is uniform in $\{-1, 1\}$, and $m = O(\varepsilon^{-2} \log(1/\delta))$, then $\|\Pi x\|_2 = (1 \pm \varepsilon)\|x\|_2$ with probability $1 - \delta$ for any $x \in \mathbb{R}^d$. This gives a running time of $O(m\text{nnz}(x))$ to sketch a vector $x \in \mathbb{R}^d$. Later, the Fast JL Transform (Ailon and Chazelle, 2006), which exploits the FFT algorithm, improved the running time for dense vectors to $O(d \log d + m^3)$. The related Subsampled Randomized Hadamard Transform has been extensively studied (Sarlos, 2006; Drineas et al., 2006b, 2011; Tropp, 2011; Drineas et al., 2012; Lu et al., 2013), which uses $O(d \log d)$ time but obtains suboptimal dimension $O\left(\varepsilon^{-2} \log(1/\delta)^2\right)$, hence it can not use the above argument to get subspace embedding, but it has been proven in Tropp (2011) that if $m = O\left(\varepsilon^{-2}(r + \log(1/\delta)^2)\right)$, then one get a subspace embedding. This improvement has a running time of $O(d \log d)$, which can be worse than $O(m\text{nnz}(x))$ if $x \in \mathbb{R}^d$ is very sparse. This inspired a line of work trying to obtain sparse Johnson Lindenstrauss transforms (Dasgupta et al., 2010; Kane and Nelson, 2014; Nelson and Nguyên, 2013; Cohen, 2016). They obtain a running time of $O\left(\varepsilon^{-1} \log(1/\delta)\text{nnz}(x)\right)$. In Nelson and Nguyên (2013) they define the ONSAP transform and investigate the trade-off between sparsity and subspace embedding dimension. This was further improved in (Cohen, 2016).

In the context of this paper all the above mentioned methods have the same shortcoming, they do not exploit the extra structure of the tensors. The Subsampled Randomized Hadamard Transform has a running time of $\Omega(pd^p \log(p))$ in the model considered in this paper, and the sparse embeddings have a running time of $\Omega(\text{nnz}(x)^p)$. This is clearly unsatisfactory and inspired the TensorSketch (Pham and Pagh, 2013; Avron et al., 2014), which has a running time

of $\Omega(p\,\mathrm{nnz}(x))$. Unfortunately, they need $m = \Omega\left(3^p \varepsilon^{-2}\delta^{-1}\right)$ and one of the main contributions of this paper is get rid of the exponential dependence on p.

Approximate Kernel Expansions A classic result by Rahimi and Recht (2008) shows how to compute an embedding for any shift-invariant kernel function in time $O(dm)$. In (Le et al., 2013) this is improved to time $O((m+d)\log d)$, however, the method does not handle kernel functions that can't be specified as a function of the inner product, and it doesn't provide subspace embeddings. See also (Avron et al., 2017c) for more approaches along the same line. Unfortunately, these methods are unable to operate in input sparsity time and their runtime, at best, is off by an s_λ factor.

Tensor Sparsification There is a literature of tensor sparsification based on sampling (Nguyen et al., 2015), however, unless the vectors tensored are very smooth (such as ± 1 vectors), the sampling has to be weighted by the data. This means that these methods aren't applicable in general to the types of problems we consider, where the tensor usually isn't known when the sketching function is sampled.

Hyper-plane rounding An alternative approach is to use hyper-plane rounding to get vectors of the form ± 1 and after this we can simply sample from the tensor product. The sign-sketch was first brought into the field of data-analysis by Charikar (2002) and Valiant (2012) was the first to use it with tensoring. The main issue with this approach is that it isn't a linear sketch, which hinders the applications we consider in this paper, such as kernel low rank approximation, CCA, PCR, and ridge regression. It takes $O(dm)$ time to sketch a single vector which is unsatisfactory.

5.1.4 Organization

In section 5.2 we introduce basic definitions and notations that will be used throughout the paper. Section 5.3 introduces our recursive construction of the sketch which is our main technical tool for sketching high degree tensor products. Section 5.4 analyzes how the moment bounds propagate through our recursive construction thereby proving Theorem 5.1.1 which has linear dependence on the degree p. Section 5.5 introduces a high probability Oblivious Subspace Embedding with linear dependence on the statistical dimension thereby proving Theorem 5.1.2. Finally, section 5.6 uses the tools that we build for sketching polynomial kernel and proves that, for the first time, the curse of dimensionality csn be avoided when sketching the Gaussian kernel.

5.2 Preliminaries

In this section we introduce notation and present useful properties of tensor product of vectors and matrices.

Definition 5.2.1 (Tensor product of vectors)**.** Given $a \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$ we define the *twofold tensor product* $a \otimes b$ to be

$$a \otimes b = \begin{bmatrix} a_1 b_1 & a_1 b_2 & \cdots & a_1 b_n \\ a_2 b_1 & a_2 b_2 & \cdots & a_2 b_n \\ \vdots & \vdots & & \vdots \\ a_m b_1 & a_m b_2 & \cdots & a_m b_n \end{bmatrix} \in \mathbb{R}^{m \times n}.$$

Although tensor products are multidimensional objects, it is often convenient to associate them with single-dimensional vectors. In particular, we will often associate $a \otimes b$ with the single-dimensional column vector $(a_1 b_1, a_2 b_1, \ldots, a_m b_1, a_1 b_2, a_2 b_2, \ldots, a_m b_2, \ldots, a_m b_n)^\top$. Given $v_1 \in \mathbb{R}^{d_1}, v_2 \in \mathbb{R}^{d_2} \cdots v_k \in \mathbb{R}^{d_k}$, we define the *k-fold tensor product* $v_1 \otimes v_2 \cdots \otimes v_k \in \mathbb{R}^{d_1 d_2 \cdots d_k}$. For shorthand, we use the notation $v^{\otimes k}$ to denote $\underbrace{v \otimes v \cdots \otimes v}_{k \text{ terms}}$, the k-fold self-tensoring of v.

Tensor product can be naturally extended to matrices which is formally defined as follows,

Definition 5.2.2. Given $A_1 \in \mathbb{R}^{m_1 \times n_1}, A_2 \in \mathbb{R}^{m_2 \times n_2}, \cdots, A_k \in \mathbb{R}^{m_k \times n_k}$, we define $A_1 \times A_2 \times \cdots \times A_k$ to be the matrix in $\mathbb{R}^{m_1 m_2 \cdots m_k \times n_1 n_2 \cdots n_k}$ whose element at row $(i_1, \cdots, i_k)$ and column $(j_1, \cdots, j_k)$ is $A_1(i_1, j_1) \cdots A_k(i_k, j_k)$. As a consequence the following holds for any $v_1 \in \mathbb{R}^{n_1}, v_2 \in \mathbb{R}^{n_2}, \cdots, v_k \in \mathbb{R}^{n_k}$: $(A_1 \times A_2 \times \cdots \times A_k)(v_1 \otimes v_2 \otimes \cdots \otimes v_k) = (A_1 v_1) \otimes (A_2 v_2) \otimes \cdots \otimes (A_k v_k)$.

The tensor product has the useful *mixed product property*, given in the following Claim,

Claim 5.2.1. *For every matrices A, B, C, D with appropriate sizes, the following holds,*

$$(A \cdot B) \times (C \cdot D) = (A \times C) \cdot (B \times D).$$

We also define the column wise tensoring of matrices as follows,

Definition 5.2.3. Given $A_1 \in \mathbb{R}^{m_1 \times n}, A_2 \in \mathbb{R}^{m_2 \times n}, \cdots, A_k \in \mathbb{R}^{m_k \times n}$, we define $A_1 \otimes A_2 \otimes \cdots \otimes A_k$ to be the matrix in $\mathbb{R}^{m_1 m_2 \cdots m_k \times n}$ whose j^{th} column is $A_1^j \otimes A_2^j \otimes \cdots \otimes A_k^j$ for every $j \in [n]$, where A_l^j is the j^{th} column of A_l for every $l \in [k]$.

Similarly the row wise tensoring of matrices are introduced in the following Definition,

Definition 5.2.4. Given $A^1 \in \mathbb{R}^{m \times n_1}, A^2 \in \mathbb{R}^{m \times n_2}, \cdots, A^k \in \mathbb{R}^{m \times n_k}$, we define $A^1 \bullet A^2 \bullet \cdots A^k$ to be the matrix in $\mathbb{R}^{m \times n_1 n_2 \cdots n_k}$ whose j^{th} row is $(A_j^1 \otimes A_j^2 \otimes \cdots \otimes A_j^k)^\top$ for every $j \in [m]$, where A_j^l is the j^{th} row of A^l as a column vector for every $l \in [k]$.

Definition 5.2.5. Another related operation is the *direct sum* for vectors: $x \oplus y = \left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$ and for matrices: $A \oplus B = \left[\begin{smallmatrix} A & 0 \\ 0 & B \end{smallmatrix}\right]$. When the sizes match up, we have $(A \oplus B)(x \oplus y) = Ax + By$. Also notice that if I_k is the $k \times k$ identity matrix, then $I_k \otimes A = \underbrace{A \oplus \cdots \oplus A}_{k \text{ times}}$.

5.3 Construction of the Sketch

In this section, we present the basic construction for our new sketch. Suppose we are given $v_1, v_2, \ldots v_q \in \mathbb{R}^m$. Our main task is to map the tensor product $v_1 \otimes v_2 \otimes \cdots \otimes v_q$ to a vector of size m using a linear sketch. Our sketch construction is recursive in nature. To illustrate the general idea, let us first consider the case in which $q \geq 2$ is a power of two. Our sketch involves first sketching each pair $(v_1 \otimes v_2), (v_3 \otimes v_4), \cdots, (v_{q-1} \otimes v_q) \in \mathbb{R}^{m^2}$ independently using independent instances of some linear base sketch (e.g., TensorSketch, SRHT, CountSketch, OSNAP). The number of vectors after this step is half of the number of vectors that we began with. The natural idea is to recursively apply the same procedure on the sketched tensors and half the number of instances of the base sketch in each successive step.

More precisely, we first choose a (randomized) base sketch $S_{\text{base}} : \mathbb{R}^{m^2} \to \mathbb{R}^m$ that sketches twofold tensor products of vectors in $\mathbb{R}^m$ (we will describe how to choose the base sketch later). Then, for any power of two $q \geq 2$, we define $Q^q : \mathbb{R}^{m^q} \to \mathbb{R}^m$ on $v_1 \otimes v_2 \otimes \cdots \otimes v_q$ recursively by,

$$Q^q(v_1 \otimes v_2 \otimes \cdots \otimes v_q) = Q^{q/2}\left(S_1^q(v_1 \otimes v_2) \otimes S_2^q(v_3 \otimes v_4) \otimes \cdots \otimes S_{q/2}^q(v_{q-1} \otimes v_q)\right),$$

where $S_1^q, S_2^q, \cdots, S_{q/2}^q : \mathbb{R}^{m^2} \to \mathbb{R}^m$ are independent instances of S_{base} and $Q^1 : \mathbb{R}^m \to \mathbb{R}^m$ is simply the identity map on $\mathbb{R}^m$.

The above construction of Q^q has been defined in terms of its action on q-fold tensor products of vectors in $\mathbb{R}^m$, but it extends naturally to a linear mapping from $\mathbb{R}^{m^q}$ to $\mathbb{R}^m$. The formal definition of Q^q is presented below.

Definition 5.3.1 (Sketch Q^q)**.** Let $m \geq 2$ be a positive integer and let $S_{\text{base}} : \mathbb{R}^{m^2} \to \mathbb{R}^m$ be a linear map that specifies some base sketch. Then, for any integer power of two $q \geq 2$, we define $Q^q : \mathbb{R}^{m^q} \to \mathbb{R}^m$ to be the linear map specified as follows:

$$Q^q \stackrel{\text{def}}{=} S^2 \cdot S^4 \cdots S^{q/2} \cdot S^q,$$

where for each $l \in \{2^1, 2^2, \cdots, q/2, q\}$, S^l is a matrix in $\mathbb{R}^{m^{l/2} \times m^l}$ defined as

$$S^l \stackrel{\text{def}}{=} S_1^l \times S_2^l \times \cdots \times S_{l/2}^l, \tag{5.2}$$

where the matrices $S_1^l, \cdots, S_{l/2}^l \in \mathbb{R}^{m \times m^2}$ are drawn independently from the base distribution S_{base}.

This sketch construction can be best visualized using a balanced binary tree with q leaves.

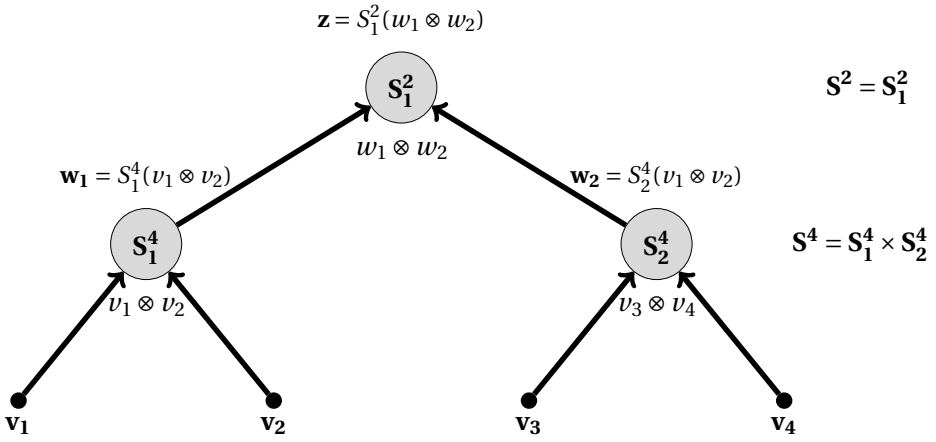


Figure 5.2 – Visual illustration of the recursive construction of Q^q for degree $q = 4$. The input tensor is $v_1 \otimes v_2 \otimes v_3 \otimes v_4$ and the output is $z = Q^4(v_1 \otimes v_2 \otimes v_3 \otimes v_4)$. The intermediate nodes sketch the tensors $w_1 = S_1^4(v_1 \otimes v_2)$ and $w_1 = S_2^4(v_3 \otimes v_4)$.

Figure 5.2 illustrates the construction of a degree 4 sketch, Q^4.

For every integer power of two q, by definition of S^q in (5.2) of Definition 5.3.1, and claim 5.2.1,

$$S^q = S_1^q \times \cdots \times S_{q/2}^q = \left(S_1^q \times \cdots \times S_{q/2-1}^q \times I_m\right) \cdot \left(I_{m^{q-2}} \times S_{q/2}^q\right).$$

By multiple applications of Claim 5.2.1 we have the following claim,

Claim 5.3.1. *For every power of two integer q and any positive integer m, if S^q is defined as in* (5.2) *of Definition 5.3.1, then $S^q \equiv M_{q/2} M_{q/2-1} \cdots M_1$, where $M_j = I_{m^{q-2j}} \times S_{q/2-j+1}^q \times I_{m^{j-1}}$ for every $j \in [q/2]$.*

Embedding $\mathbb{R}^{d^q}$: So far we have constructed a sketch Q^q for sketching tensor product of vectors in $\mathbb{R}^m$. However, in general the data points can be in a space $\mathbb{R}^d$ of arbitrary dimension. A natural idea is to reduce the dimension of the vectors by a linear mapping from $\mathbb{R}^d$ to $\mathbb{R}^m$ and then apply Q^q on the tensor product of reduced data points. The dimensionality reduction defines a linear mapping from $\mathbb{R}^{d^q}$ to $\mathbb{R}^{m^d}$ which is denoted by T^q and formally defined as:

Definition 5.3.2 (Sketch T^q)**.** Let m, d be positive integers and let $T_{\text{base}} : \mathbb{R}^d \to \mathbb{R}^m$ be a linear map that specifies some base sketch. Then for any integer $q \geq 1$ we define T^q to be the linear map specified as follows,

$$T^q \stackrel{\text{def}}{=} T_1 \times T_2 \times \cdots \times T_q,$$

where the matrices $T_1, \cdots, T_q$ are drawn independently from T_{base}.
Discussion: Similar to Claim 5.3.1, the transform T^q can be expressed as the product of q matrices, $T^q \equiv M_q M_{q-1} \cdots M_1$, where $M_j = I_{d^{q-j}} \times T_{q-j+1} \times I_{m^{j-1}}$ for every $j \in [q]$.

Now we define the final sketch $\Pi^q : \mathbb{R}^{d^q} \to \mathbb{R}^m$ for arbitrary d as the composition of $Q^q \cdot T^q$.

Algorithm 15 Sketch for the Tensor $x^{\otimes p}$

input: Vector $x \in \mathbb{R}^d$, degree p, base sketches $S_{\text{base}} \in \mathbb{R}^{m \times m^2}$ and $T_{\text{base}} \in \mathbb{R}^{m \times d}$
output: Sketched vector $z \in \mathbb{R}^m$

1: Let $q = 2^{\lceil \log_2 p \rceil}$
2: Let $T_1, \cdots T_q$ be independent instances of the base sketch $T_{\text{base}} : \mathbb{R}^d \to \mathbb{R}^m$
3: For every $j \in \{1, 2, \cdots, p\}$, let $Y_j^0 = T_j \cdot x$
4: For every $j \in \{p+1, \cdots, q\}$, let $Y_j^0 = T_j \cdot \mathbf{e}_1$, where $\mathbf{e}_1$ is the standard basis vector in $\mathbb{R}^d$ with value 1 in the first coordinate and zero elsewhere
5: **for** $l = 1$ to $\log_2 q$ **do**
6: Let $S_1^{q/2^{l-1}}, \cdots, S_{q/2^l}^{q/2^{l-1}}$ be independent instances of the base sketch $S_{\text{base}} : \mathbb{R}^{m^2} \to \mathbb{R}^m$
7: For every $j \in \left\{1, \cdots, q/2^l\right\}$ let $Y_j^l = S_j^{q/2^{l-1}} \left(Y_{2j-1}^{l-1} \otimes Y_{2j}^{l-1} \right)$
8: **return** $z = Y_1^{\log_2 q}$

Moreover, to extend the definition to arbitrary q which is not necessarily a power of two we tensor the input vector with a standard basis vector a number of times to make the input size compatible with the sketch matrices. The sketch Π^p is formally defined below,

Definition 5.3.3 (Sketch Π^p)**.** Let m, d be positive integers and let $S_{\text{base}} : \mathbb{R}^{m^2} \to \mathbb{R}^m$ and $T_{\text{base}} : \mathbb{R}^d \to \mathbb{R}^m$ be linear maps that specify some base sketches. Then, for any integer $p \geq 2$ we define $\Pi^p : \mathbb{R}^{d^p} \to \mathbb{R}^m$ to be the linear map specified as follows:

1. If p is a power of two, then Π^p is defined as $\Pi^p \stackrel{\text{def}}{=} Q^p \cdot T^p$, where $Q^p \in \mathbb{R}^{m \times m^p}$ and $T^p \in \mathbb{R}^{m^p \times d^p}$ are sketches as in Definitions 5.3.1 and 5.3.2 respectively.

2. If p is not a power of two, then let $q = 2^{\lceil \log_2 p \rceil}$ be the smallest power of two integer that is greater than p. We define Π^p as the linear map,

$$\Pi^p(v) = \Pi^q \left(v \otimes \mathbf{e}_1^{\otimes(q-p)} \right),$$

for every $v \in \mathbb{R}^{d^p}$, where $\mathbf{e}_1 \in \mathbb{R}^d$ is the standard basis vector with value 1 in the first coordinate and zeros elsewhere, and Π^q is defined as in the first part of this definition.

Algorithm 15 sketches $x^{\otimes p}$ for any integer p and any input vector $x \in \mathbb{R}^d$ using the sketch Π^p as in Definition 5.3.3, i.e., computes $\Pi^p(x^{\otimes p})$. We show the correctness of Algorithm 15 in the next lemma.

Lemma 5.3.1. *For any positive integers d, m, and p, any distribution on matrices $S_{\text{base}} : \mathbb{R}^{m^2} \to \mathbb{R}^m$ and $T_{base} : \mathbb{R}^d \to \mathbb{R}^m$ which specify some base sketches, any vector $x \in \mathbb{R}^d$, Algorithm 15 computes $\Pi^p \left(x^{\otimes p} \right)$ as in Definition 5.3.3.*

Proof. Let $Y_1^0, \cdots, Y_p^0$ be the vectors that are computed in lines 3 and 4 of Algorithm 15. Then, as shown in Definition 5.2.2, $Y_1^0 \otimes \cdots \otimes Y_p^0 = T_1 \times \cdots \times T_q \cdot \left(x^{\otimes p} \otimes \mathbf{e}_1^{\otimes(q-p)} \right)$. Let T^q be the sketch

as in Definition 5.3.2. Then it follows that,

$$Y_1^0 \otimes \cdots \otimes Y_q^0 = T^q \cdot \left(x^{\otimes p} \otimes \mathbf{e}_1^{\otimes(q-p)}\right). \tag{5.3}$$

The algorithm computes $Y_1^l, \cdots Y_{q/2^l}^l$ in line 7 as, $Y_j^l = S_j^{q/2^{l-1}}\left(Y_{2j-1}^{l-1} \otimes Y_{2j}^{l-1}\right)$, for every $j \in \{1, \cdots, q/2^l\}$ and every $l \in \{1, \cdots, \log_2 q\}$. Therefore, by Claim 5.2.1,

$$Y_1^l \otimes \cdots \otimes Y_{q/2^l}^l = \left(S_1^{q/2^{l-1}} \times \cdots \times S_{q/2^l}^{q/2^{l-1}}\right) \cdot \left(Y_1^{l-1} \otimes \cdots \otimes Y_{q/2^{l-1}}^{l-1}\right).$$

By definition of the sketch $S^{q/2^{l-1}}$ in (5.2) of Definition 5.3.1, for every $l \in \{1, \cdots, \log_2 q\}$,

$$Y_1^l \otimes \cdots \otimes Y_{q/2^l}^l = S^{q/2^{l-1}} \cdot Y_1^{l-1} \otimes \cdots \otimes Y_{q/2^{l-1}}^{l-1}.$$

Therefore, by recursive application of the above identity we get that,

$$Y_1^{\log_2 q} = S^2 \cdot S^4 \cdots S^{q/2} \cdot S^q \cdot Y_1^0 \otimes \cdots \otimes Y_q^0.$$

By Definition 5.3.1 (sketch Q^q) it follows that,

$$Y_1^{\log_2 q} = Q^q \cdot Y_1^0 \otimes \cdots \otimes Y_q^0.$$

Substituting $Y_1^0 \otimes \cdots \otimes Y_q^0$ from (5.3) in the above gives, $z = (Q^q \cdot T^q) \cdot \left(x^{\otimes p} \otimes e_1^{\otimes(q-p)}\right)$, hence, by Definition 5.3.3, $z = \Pi^p\left(x^{\otimes p}\right)$. □

Choices of the Base Sketches S_{base} and T_{base}: We present formal definitions of various base sketches that will be used in our sketch construction. We start by briefly recalling the CountSketch (Charikar et al., 2002).

Definition 5.3.4 (CountSketch)**.** Let $h : [d] \to [m]$ be a 3-wise independent hash function, also let $\sigma : [d] \to \{-1, +1\}$ be a 4-wise independent random sign function. Then, the CountSketch transform, $S : \mathbb{R}^d \to \mathbb{R}^m$, is defined as $S_{r,i} \stackrel{\text{def}}{=} \sigma(i) \cdot \mathbb{1}[h(i) = r]$, for every $i \in [d]$ and every $r \in [m]$.

Another base sketch that we consider is the degree-2 TensorSketch (Pagh, 2013) defined as:

Definition 5.3.5 (Degree-2 TensorSketch transform)**.** Let $h_1, h_2 : [d] \to [m]$ be 3-wise independent hash functions, and also let $\sigma_1, \sigma_2 : [d] \to \{-1, +1\}$ be 4-wise independent random sign functions. Then, the degree-2 TensorSketch transform, $S : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^m$, is defined as $S_{r,(i,j)} \stackrel{\text{def}}{=} \sigma_1(i) \cdot \sigma_2(j) \cdot \mathbb{1}\left[h_1(i) + h_2(j) = r \pmod m\right]$, for every $i, j \in [d]$ and every $r \in [m]$,

Remark: $S\left(x^{\otimes 2}\right)$ can be computed in $O(m \log m + \text{nnz}(x))$ time using the FFT algorithm.

Now let us briefly recall the SRHT (Ailon and Chazelle, 2006).

Definition 5.3.6 (SRHT)**.** Let D be a $d \times d$ diagonal matrix with i.i.d. Rademacher diagonal entries. Also, let $P \in \{0,1\}^{m \times d}$ be a uniform random sampling matrix, and let H be a $d \times d$ Hadamard matrix. Then, the SRHT, $S \in \mathbb{R}^{m \times d}$, is defined as $S \stackrel{\text{def}}{=} \frac{1}{\sqrt{m}} PHD$.

We now define a new variant of the SRHT that is very efficient for sketching tensor product vectors. We call this sketch the TensorSRHT.

Definition 5.3.7 (TensorSRHT)**.** Let D_1 and D_2 be two independent $d \times d$ diagonal matrices, each with i.i.d. Rademacher diagonal entries. Also let $P \in \{0,1\}^{m \times d^2}$ be a uniform random sampling matrix, and let H be a $d \times d$ Hadamard matrix. Then, the TensorSRHT is defined to be a linear map $S : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^m$ given by $S \stackrel{\text{def}}{=} \frac{1}{\sqrt{m}} P \cdot (HD_1 \times HD_2)$.

Remark: $S\left(x^{\otimes 2}\right)$ can be computed in time $O(d \log d + m)$ using the FFT algorithm.

Another sketch which is particularly efficient for sketching sparse vectors with high probability is the OSNAP transform (Nelson and Nguyên, 2013), defined as follows.

Definition 5.3.8 (OSNAP)**.** For every sparsity parameter s, target dimension m, and positive integer d, the OSNAP transform with sparsity parameter s is defined as, $S_{r,j} \stackrel{\text{def}}{=} \sqrt{\frac{1}{s}} \cdot \delta_{r,j} \cdot \sigma_{r,j}$, for all $r \in [m]$ and all $j \in [d]$, where $\sigma_{r,j} \in \{-1,+1\}$ are independent and uniform Rademacher random variables and $\delta_{r,j}$ are Bernoulli random variables satisfying,

1. For every $i \in [d]$, $\sum_{r \in [m]} \delta_{r,i} = s$. That is, each column of S has exactly s non-zero entries.
2. For all $r \in [m]$ and all $i \in [d]$, $\mathbb{E}\left[\delta_{r,i}\right] = s/m$.
3. $\delta_{r,i}$'s are negatively correlated: $\forall T \subset [m] \times [d]$, $\mathbb{E}\left[\prod_{(r,i) \in T} \delta_{r,i}\right] \leq \prod_{(r,i) \in T} \mathbb{E}\left[\delta_{r,i}\right] = (\frac{s}{m})^{|T|}$.

5.4 Linear Dependence on the Tensoring Degree p

There are various desirable properties that one would wish for a linear sketch to satisfy. One such property that is central to our main results is the *JL Moment Property*. In this section we prove Theorem 5.1.1 by propagating the JL moment property through our recursive construction from Section 5.3. The JL moment property captures a bound on the moments of the difference between the Euclidean norm of a vector and its Euclidean norm after applying the sketch on it. This proves to be a powerful property which implies the *Oblivious Subspace Embedding* as well as the *Approximate Matrix Product* for linear sketches.

In this section we choose S_{base} and T_{base} to be TensorSketch and CountSketch respectively. Then we propagate the second JL moment through the sketch construction Π^p and thereby prove Theorem 5.1.1.

Throughout this section we use $\|X\|_{L^t}$ for the t^{th} moment of X, formally defined as:

Definition 5.4.1 (Moments of a Random Variable)**.** For every integer $t \geq 1$ and any random variable $X \in \mathbb{R}$, we define $\|X\|_{L^t} \stackrel{\text{def}}{=} \left(E\left[|X|^t\right]\right)^{1/t}$. Note that $\|X+Y\|_{L^t} \leq \|X\|_{L^t} + \|Y\|_{L^t}$ for any random variables X, Y by the Minkowski's Inequality.

We now formally define the JL Moment Property of sketches.

Definition 5.4.2 (JL Moment Property)**.** For every positive integer t and every $\delta, \varepsilon \geq 0$, we say a distribution over random matrices $S \in \mathbb{R}^{m \times d}$ has the (ϵ, δ, t)-JL-moment property, when

$$\left\| \|Sx\|_2^2 - 1 \right\|_{L^t} \leq \epsilon \delta^{1/t} \quad \text{and} \quad \mathbb{E}\left[\|Sx\|_2^2\right] = 1$$

for all $x \in \mathbb{R}^d$ such that $\|x\| = 1$.

The following two lemmas together show that to prove that Π^p is an OSE and that Π^p has the Approximate Matrix Multiplication property, it suffices to prove that Π^q has the JL Moment Property, for q which is the smallest power of two integer such that $q \geq p$, as in Definition 5.3.3. This reduction will be the main component of the proof of Theorem 5.1.1.

Lemma 5.4.1. *For every positive integers n, p, d, every $\varepsilon, \delta \in [0,1]$, and every $\mu \geq 0$. Let $q = 2^{\lceil \log_2(p) \rceil}$ and let $\Pi^p \in \mathbb{R}^{m \times d^p}$ and $\Pi^q \in \mathbb{R}^{m \times d^q}$ be defined as in Definition 5.3.3, for some base sketches $S_{\text{base}} \in \mathbb{R}^{m \times m^2}$ and $T_{\text{base}} \in \mathbb{R}^{d \times d}$.*
If Π^q is an $(\varepsilon, \delta, \mu, d^q, n)$-Oblivious Subspace Embedding then Π^p is an $(\varepsilon, \delta, \mu, d^p, n)$-Oblivious Subspace Embedding. Also if Π^q has the (ε, δ)-Approximate Matrix Multiplication Property then Π^p has the (ε, δ)-Approximate Matrix Multiplication Property.

The proof of this lemma can be found in Appendix D.1.

Lemma 5.4.2. *For any $\varepsilon, \delta \in [0,1]$, $t \geq 1$, if $M \in \mathbb{R}^{m \times d}$ is a random matrix with (ε, δ, t)-JL Moment Property then M has the (ε, δ)-Approximate Matrix Multiplication Property.*
Furthermore, for any $\mu > 0$, if $M \in \mathbb{R}^{m \times d}$ is a random matrix with $(\varepsilon/\mu, \delta, t)$-JL Moment Property then for every positive integer n, M is a $(\varepsilon, \delta, \mu, d, n)$-OSE.

This lemma is proved in Appendix D.1.
Our next important observation is that Π^q can be written as the product of $2q-1$ independent random matrices, which all have a special structure which makes them easy to analyse.

Lemma 5.4.3. *For any integer power of two q, if $\Pi^q : \mathbb{R}^{m^q} \to \mathbb{R}^m$ is defined as in Definition 5.3.3 for some base sketches $S_{\text{base}} : \mathbb{R}^{m^2} \to \mathbb{R}^m$ and $T_{\text{base}} : \mathbb{R}^d \to \mathbb{R}^m$, then there exist matrices $\left\{M^{(i)}\right\}_{i \in [q-1]}, \left\{M'^{(j)}\right\}_{j \in [q]}$ and integers $\left\{k_i, k_i' \leq m^{q-1}\right\}_{i \in [q-1]}, \left\{l_j, l_j' \leq d^{q-1}\right\}_{j \in [q]}$, such that,*

$$\Pi^q = M^{(q-1)} \cdot \ldots M^{(1)} \cdot M'^{(q)} \cdot \ldots \cdot M'^{(1)},$$

and $M^{(i)} = I_{k_i} \times S_{\text{base}}^{(i)} \times I_{k_i'}$, $M'^{(j)} = I_{l_j} \times T_{\text{base}}^{(j)} \times I_{l_j'}$, where $S_{\text{base}}^{(i)}$ and $T_{\text{base}}^{(j)}$ are independent instances of S_{base} and T_{base}, for every $i \in [q-1]$, $j \in [q]$.

Proof. We have that $\Pi^q = Q^q T^q$ by Definition 5.3.3. By Definition 5.3.1, $Q^q = S^2 S^4 \cdots S^q$. Claim 5.3.1 shows that for every $l \in \{2, 4, \cdots q\}$ we can write,

$$S^l = M^l_{l/2} M^l_{l/2-1} \cdot \ldots \cdot M^l_1, \tag{5.4}$$

where $M^l_j = I_{m^{l-2j}} \times S^l_{l/2-j+1} \times I_{m^{j-1}}$ for every $j \in [l/2]$. From the discussion in Definition 5.3.2 it follows that,

$$T^q = M'^{(q)} \cdot \ldots \cdot M'^{(1)}, \tag{5.5}$$

where $M'^{(j)} = I_{d^{q-j}} \times T_{q-j+1} \times I_{m^{j-1}}$ for every $j \in [q]$. Therefore by combining (5.4) and (5.5) we get the result. □

We want to show that $I_k \times M \times I_{k'}$ inherits the JL moment properties of M. The following Lemma, which follows from the simple fact stated in Lemma D.1.2 in Appendix D.1, does that.

Lemma 5.4.4. *If the matrix S has the (ε, δ, t)-JL Moment Property, then for any positive integers k, k', the matrix $M = I_k \times S \times I_{k'}$ has the (ε, δ, t)-JL Moment Property.*
Similarly, if the matrix S has the Strong (ε, δ)-JL Moment Property, then for any positive integers k, k', the matrix $M = I_k \times S \times I_{k'}$ has the Strong (ε, δ)-JL Moment Property.

Consequently, if we can prove that the product of matrices with the JL moment property inherits the JL moment property, then Lemma 5.4.4 and Lemma 5.4.3 will imply that Π^q has the JL moment property, which in turn implies that Π^p is an OSE and has the Approximate Matrix Multiplication property, by Lemma 5.4.2 and Lemma 5.4.1. This is exactly what we will do: in Section 5.4.1 we prove that the product of k independent matrices with the $\left(\frac{\varepsilon}{\sqrt{2k}}, \delta, 2\right)$-JL Moment Property results in a matrix with the $(\varepsilon, \delta, 2)$-JL Moment Property, thereby giving the proof of Theorem 5.1.1.

5.4.1 Second moment of Π^q (analysis for T_{base} : CountSketch and S_{base} : TensorSketch)

In this section we prove Theorem 5.1.1 by instantiating our recursive construction with CountSketch at the leaves and TensorSketch at the internal nodes of the tree. The proof proceeds by showing bounding the second moment of our recursive construction. More precisely, we prove that our sketch Π^q satisfies the $(\varepsilon, \delta, 2)$-JL Moment Property as per Definition 5.4.2 as long as the base sketches $S_{\text{base}}, T_{\text{base}}$ are chosen from distributions which satisfy such moment property. We show that this is the case for CountSketch and TensorSketch. Lemma 5.4.4 and Lemma 5.4.3 show that Π^q is the product of $2q-1$ independent random matrices, therefore, understanding how matrices with the JL Moment Property compose is crucial. The following lemma shows that composing independent random matrices which have the JL moment property results in a matrix with the JL moment property with a small loss in parameters.

Lemma 5.4.5 (Composition lemma for the second moment)**.** *For any $\varepsilon, \delta \geq 0$ and any integer*

k if $M^{(1)} \in \mathbb{R}^{d_2 \times d_1}, \cdots M^{(k)} \in \mathbb{R}^{d_{k+1} \times d_k}$ are independent random matrices with $\left(\frac{\varepsilon}{\sqrt{2k}}, \delta, 2\right)$-JL moment property then the product matrix $M = M^{(k)} \cdots M^{(1)}$ satisfies $(\varepsilon, \delta, 2)$-JL moment property.

Proof. Let $x \in \mathbb{R}^{d_1}$ be a fixed unit norm vector. We note that for any $i \in [k]$,

$$\mathbb{E}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2 \,\middle|\, M^{(1)}, \ldots, M^{(i-1)}\right] = \left\|M^{(i-1)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2. \tag{5.6}$$

We proceed to prove by induction on $i \in [k]$ that,

$$\mathrm{Var}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2\right] \leq \left(1 + \frac{\varepsilon^2 \delta}{2k}\right)^i - 1. \tag{5.7}$$

For $i = 1$ the result follows from the fact that $M^{(1)}$ has the $\left(\varepsilon/\sqrt{2k}, \delta, 2\right)$-JL moment property. Now assume that (5.7) is true for $i-1$. By the law of total variance we can write

$$\begin{aligned}\mathrm{Var}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2\right] &= \mathbb{E}\left[\mathrm{Var}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2 \,\middle|\, M^{(1)}, \ldots, M^{(i-1)}\right]\right] \\ &\quad + \mathrm{Var}\left[\mathbb{E}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2 \,\middle|\, M^{(1)}, \ldots, M^{(i-1)}\right]\right]\end{aligned} \tag{5.8}$$

Using (5.6) and the induction hypothesis we get that,

$$\begin{aligned}\mathrm{Var}\left[\mathbb{E}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2 \,\middle|\, M^{(1)}, \ldots, M^{(i-1)}\right]\right] &= \mathrm{Var}\left[\left\|M^{(i-1)} \cdot \ldots M^{(1)} x\right\|_2^2\right] \\ &\leq \left(1 + \frac{\varepsilon^2 \delta}{2k}\right)^{i-1} - 1.\end{aligned} \tag{5.9}$$

Since $M^{(i)}$ has the $\left(\varepsilon/\sqrt{2k}, \delta, 2\right)$-JL moment property, (5.6) and the induction hypothesis together imply that,

$$\begin{aligned}&\mathbb{E}\left[\mathrm{Var}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2 \,\middle|\, M^{(1)}, \ldots, M^{(i-1)}\right]\right] \\ &\leq \mathbb{E}\left[\frac{\varepsilon^2}{2k}\delta \left\|M^{(i-1)} \cdot \ldots M^{(1)} x\right\|_2^4\right] \\ &= \frac{\varepsilon^2 \delta}{2k}\left(\mathrm{Var}\left[\left\|M^{(i-1)} \cdot \ldots M^{(1)} x\right\|_2^2\right] + \mathbb{E}\left[\left\|M^{(i-1)} \cdot \ldots M^{(1)} x\right\|_2^2\right]^2\right) \\ &\leq \frac{\varepsilon^2 \delta}{2k}\left(\left(1 + \frac{\varepsilon^2 \delta}{2k}\right)^{i-1} - 1 + 1\right) = \frac{\varepsilon^2 \delta}{2k}\left(1 + \frac{\varepsilon^2 \delta}{2k}\right)^{i-1}.\end{aligned} \tag{5.10}$$

Plugging (5.9) and (5.10) into (5.8) gives,

$$\mathrm{Var}\left[\left\|M^{(i)} \cdot \ldots \cdot M^{(1)} x\right\|_2^2\right] \leq \frac{\varepsilon^2 \delta}{2k}\left(1 + \frac{\varepsilon^2 \delta}{2k}\right)^{i-1} + \left(1 + \frac{\varepsilon^2 \delta}{2k}\right)^{i-1} - 1 = \left(1 + \frac{\varepsilon^2 \delta}{2k}\right)^i - 1.$$

Hence, $\mathrm{Var}\left[\|Mx\|_2^2\right] \le \left(1+\frac{\varepsilon^2\delta}{2k}\right)^k - 1 \le \varepsilon^2\delta$. Moreover, it is easy to verify that $\mathbb{E}\left[\|Mx\|_2^2\right] = 1$, by law of total expectation, which proves that M has the $(\varepsilon,\delta,2)$-JL moment property. □

Equipped with the composition lemma for the second moment, we now establish the second moment property for our recursive sketch Π^q:

Corollary 5.4.1 (Second moment property of Π^q)**.** *For any power of two integer q let $\Pi^q : \mathbb{R}^{m^q} \to \mathbb{R}^m$ be defined as in Definition 5.3.3, where both of the common distributions $S_{\text{base}} : \mathbb{R}^{m^2} \to \mathbb{R}^m$ and $T_{\text{base}} : \mathbb{R}^d \to \mathbb{R}^m$, satisfy the $\left(\frac{\varepsilon}{\sqrt{4q-2}},\delta,2\right)$-JL moment property. Then it follows that Π^q satisfies the $(\varepsilon,\delta,2)$-JL moment property.*

Proof. This immediately follows from Lemma 5.4.3, Lemma 5.4.4, and Lemma 5.4.5. □

Now we are in a position to prove Theorem 5.1.1. Recall that $k(x,y) = \langle x,y\rangle^q = \langle x^{\otimes q}, y^{\otimes q}\rangle$ is the polynomial kernel of degree q. Let $x_1, x_2, \cdots x_n \in \mathbb{R}^m$ be an arbitrary dataset of n points in $\mathbb{R}^m$. We represent the data points by matrix $X \in \mathbb{R}^{m\times n}$ whose i^{th} column is the vector x_i. Also, let $A \in \mathbb{R}^{m^q\times n}$ be the matrix whose i^{th} column is $x_i^{\otimes q}$ for every $i \in [n]$.

Theorem 5.1.1. *For every positive integers n, p, d, every $\varepsilon, s_\lambda > 0$, there exists a distribution on linear sketches $\Pi^p \in \mathbb{R}^{m\times d^p}$ such that:* **(1)** *If $m = \Omega\left(ps_\lambda^2\epsilon^{-2}\right)$, then Π^p is an $(\varepsilon, 1/10, s_\lambda, d^p, n)$-oblivious subspace embedding as in Definition 5.1.2.* **(2)** *If $m = \Omega\left(p\varepsilon^{-2}\right)$, then Π^p has the $(\varepsilon, 1/10)$-approximate matrix product property as in Definition 5.1.3.*
Moreover, for any $X \in \mathbb{R}^{d\times n}$, if $A \in \mathbb{R}^{d^p\times n}$ is the matrix whose columns are obtained by the p-fold self-tensoring of each column of X then $\Pi^p A$ can be computed in time $\widetilde{O}\left(pnm + p\,\mathrm{nnz}(X)\right)$.

Proof. Throughout the proof, suppose $\delta = \frac{1}{10}$ denotes the failure probability, $q = 2^{\lceil \log_2 p\rceil}$, and $\mathbf{e}_1 \in \mathbb{R}^d$ is the column vector with a 1 in the first coordinate and zeros elsewhere. Let $\Pi^p \in \mathbb{R}^{m\times d^p}$ be the sketch defined in Definition 5.3.3, where the base distributions $S_{\text{base}} \in \mathbb{R}^{m\times m^2}$ and $T_{\text{base}} \in \mathbb{R}^{m\times d}$ are respectively the standard degree-2 TensorSketch (Definition 5.3.5) and standard CountSketch (Definition 5.3.4). It is shown in (Avron et al., 2014; Clarkson and Woodruff, 2017) that for these choices of base sketches, S_{base} and T_{base} are both unbiased and satisfy the $\left(\frac{\varepsilon}{\sqrt{4q-2}},\delta,2\right)$-JL moment property as long as $m = \Omega\left(\frac{q}{\varepsilon^2\delta}\right)$ (see Definition 5.4.2).

Oblivious Subspace Embedding. Let $m = \Omega\left(\frac{qs_\lambda^2}{\delta\epsilon^2}\right)$ be a large enough integer. Then S_{base} and T_{base} have the $\left(\frac{\varepsilon}{s_\lambda\cdot\sqrt{4q-2}},\delta,2\right)$-JL Moment Property. Thus using Corollary 5.4.1 we conclude that Π^q has the $\left(\frac{\varepsilon}{s_\lambda},\delta,2\right)$-JL Moment Property. Hence, Lemma 5.4.2 implies that Π^q is an $(\varepsilon,\delta,s_\lambda,d^q,n)$-Oblivious Subspace Embedding, which in turn, by Lemma 5.4.1, implies that Π^p is an $(\varepsilon,\delta,s_\lambda,d^p,n)$-Oblivious Subspace Embedding.

Approximate Matrix Multiplication. Let $m = \Omega\left(\frac{q}{\delta\epsilon^2}\right)$ be a large enough integer. Then S_{base} and T_{base} have the $\left(\frac{\varepsilon}{\sqrt{4q-2}}, \delta, 2\right)$-JL Moment Property. Thus, using Corollary 5.4.1 we conclude that Π^q has the $(\varepsilon, \delta, 2)$-JL Moment Property. Thus, Lemma 5.4.2 implies that Π^q has the (ε, δ)-Approximate Matrix Multiplication Property, and by Lemma 5.4.1 we find that Π^p has the (ε, δ)-Approximate Matrix Multiplication Property.

Runtime of Algorithm 15 when the base sketch S_{base} is degree-2 TensorSketch and T_{base} is CountSketch: We bound the time of running Algorithm 15 on a vector x. Computing Y_j^0 for each j in lines 3 and 4 of algorithm requires applying a CountSketch on either x or $\mathbf{e}_1$ which requires $O(\text{nnz}(x))$ operations. Therefore computing all Y_j^0's takes time $O(q \cdot \text{nnz}(x))$.
Computing each of Y_j^l's for $l \geq 1$ in line 7 of Algorithm 15 amounts to applying a degree-2 TensorSketch with input dimension m^2 and target dimension m on $Y_{2j-1}^{l-1} \otimes Y_{2j}^{l-1}$. This takes time $O(m \log m)$. Therefore computing Y_j^l for all $l, j \geq 1$ takes time $O(q \cdot m \log m)$. Note that $q \leq 2p$ and hence the total running time of Algorithm 15 on a vector x is $O(pm \log m + p\text{nnz}(x))$. Sketching n columns of a matrix $X \in \mathbb{R}^{d \times n}$ takes time $O\left(pnm \log m + p\text{nnz}(X)\right)$. □

5.5 Linear Dependence on the Statistical Dimension s_λ

In this section, we show that by choosing the internal nodes and the leaves of our recursive construction in Section 5.3 to be TensorSRHT and OSNAP transforms respectively, then the sketch Π^q as in Definition 5.3.3 yields a high probability OSE with target dimension $\widetilde{O}\left(p^4 s_\lambda\right)$. Thus, we prove Theorem 5.1.2. This sketch is highly efficient to compute because the OSNAP transform is computable in input sparsity time and the TensorSRHT supports fast matrix vector multiplication for tensor inputs.
We start by defining the *Spectral Property* of a random matrix. We use the notation $\|\cdot\|_{\text{op}}$ to denote the operator norm of matrices.

Definition 5.5.1 (Spectral Property)**.** For any positive integers m, n, d and any $\varepsilon, \delta, \mu_F, \mu_2 \geq 0$ we say that a random matrix $S \in \mathbb{R}^{m \times d}$ satisfies the $\left(\mu_F, \mu_2, \epsilon, \delta, n\right)$-*Spectral Property* if, for every fixed matrix $U \in \mathbb{R}^{d \times n}$ with $\|U\|_F^2 \leq \mu_F$ and $\|U\|_{\text{op}}^2 \leq \mu_2$,

$$\Pr_S\left[\left\|U^\top S^\top S U - U^\top U\right\|_{\text{op}} \leq \epsilon\right] \geq 1 - \delta.$$

The *spectral property* is a central property of our sketch construction in Section 5.3 when leaves are OSNAP and internal nodes are TensorSRHT. This is a powerful property which implies that our sketch is an *Oblivious Subspace Embedding*. The SRHT, TensorSRHT, as well as OSNAP sketches (Definitions 5.3.6, 5.3.7, 5.3.8 respectively) with target dimension $m = \Omega\left(\frac{\mu_F \mu_2}{\epsilon^2} \cdot \text{polylog}\frac{nd}{\delta}\right)$ and sparsity parameter $s = \Omega\left(\frac{\log \frac{nd}{\delta}}{\varepsilon}\right)$, all satisfy the above-mentioned spectral property (Sarlos, 2006; Tropp, 2011; Nelson and Nguyên, 2013).

We start by showing in section 5.5.1 that our recursive construction of Π^p satisfies the Spectral

Property as per Definition 5.5.1 as long as $I_{d^q} \times T_{\text{base}}$ and $I_{m^q} \times S_{\text{base}}$ have such property. Therefore, we proceed by analyzing the Spectral Property of $I_{d^q} \times$ OSNAP and $I_{m^q} \times$ TensorSRHT in section 5.5.2. Finally, we put everything together in section 5.5.3 and prove that when the leaves are OSNAP and the internal nodes are TensorSRHT in our recursive construction of Π^p, the resulting sketch satisfies the Spectral Property thereby proving Theorem 5.1.2.

5.5.1 Spectral property of the sketch Π^q

In this section we show that the sketch Π^q presented in Definition 5.3.3 inherits the spectral property (see Definition 5.5.1) from the base sketches S_{base} and T_{base}. We start by proving that composing independent random matrices inherits the spectral property from the original matrices a with small loss.

Lemma 5.5.1. *For any $\varepsilon, \delta, \mu_F, \mu_2 > 0$ and every positive integers k, n, if $M^{(1)} \in \mathbb{R}^{d_2 \times d_1}, \cdots M^{(k)} \in \mathbb{R}^{d_{k+1} \times d_k}$ are independent random matrices satisfying $\left(2\mu_F + 2, \mu_2 + 2, \frac{\epsilon}{3k}, \frac{\delta}{4nk}, n\right)$-spectral property then the product matrix $M = M^{(k)} \cdots M^{(1)}$ has $(\mu_F + 1, \mu_2 + 1, \epsilon, \delta, n)$-spectral property.*

Proof. Consider a matrix $U \in \mathbb{R}^{d_1 \times n}$ which satisfies $\|U\|_F^2 \leq \mu_F + 1$ and $\|U\|_{op}^2 \leq \mu_2 + 1$. We aim to prove that for every such U,$\Pr\left[\left\|U^\top M^\top M U - U^\top U\right\|_{\text{op}} \leq \epsilon\right] \geq 1 - \delta$, where $M \equiv M^{(k)} \cdots M^{(1)}$.

For every $j \in [k]$, let us define the set $\mathscr{E}_j$ as follows,

$$\mathscr{E}_j := \left\{ \left(M^{(1)}, \ldots, M^{(j)}\right) : \begin{cases} 1. \left\|\left[M^{(j)} \cdots M^{(1)}\right] U\right\|_F^2 \leq \left(1 + \frac{\epsilon}{3k}\right)^j \|U\|_F^2 \\ 2. \left\|U^\top \left[M^{(j)} \cdots M^{(1)}\right]^\top \left[M^{(j)} \cdots M^{(1)}\right] U - U^\top U\right\|_{\text{op}} \leq \frac{\epsilon j}{3k} \end{cases} \right\}.$$

First we prove that for every $j \in \{1, \cdots, k-1\}$,

$$\Pr\left[\left(M^{(1)}, \cdots, M^{(j+1)}\right) \in \mathscr{E}_{j+1} \middle| \left(M^{(1)}, \cdots, M^{(j)}\right) \in \mathscr{E}_j\right] \geq 1 - \frac{\delta}{2k}. \tag{5.11}$$

We proceed to prove (5.11) in an inductive fashion. Suppose that $\left(M^{(1)}, \cdots, M^{(j)}\right) \in \mathscr{E}_j$ for some $j \geq 1$. Let us denote $\left[M^{(j)} \cdots M^{(1)}\right] U$ by U'. The assumption $\left(M^{(1)}, \cdots, M^{(j)}\right) \in \mathscr{E}_j$ implies that, $\|U'\|_F^2 \leq \left(1 + \frac{\epsilon}{3k}\right)^j \|U\|_F^2$ and $\left\|U'^\top U' - U^\top U\right\|_{\text{op}} \leq \frac{\epsilon j}{3k}$ and therefore by triangle inequality we have $\|U'\|_{\text{op}}^2 \leq \|U\|_{\text{op}}^2 + \frac{\epsilon j}{3k}$. The assumptions $\|U\|_F^2 \leq \mu_F + 1$ and $\|U\|_{op}^2 \leq \mu_2 + 1$ together with $j \leq k-1$ imply that $\|U'\|_F^2 \leq 2\mu_F + 2$ and $\|U'\|_{\text{op}}^2 \leq \mu_2 + 2$. Now note that by the assumption of lemma, $M^{(j+1)}$ satisfies $\left(2\mu_F + 2, \mu_2 + 2, \frac{\epsilon}{3k}, \frac{\delta}{4nk}, n\right)$-spectral property. Therefore,

$$\Pr\left[\left\|\left(M^{(j+1)} U'\right)^\top M^{(j+1)} U' - U'^\top U'\right\|_{\text{op}} \leq \frac{\epsilon}{3k} \middle| \left(M^{(1)}, \cdots, M^{(j)}\right) \in \mathscr{E}_j\right] \geq 1 - \frac{\delta}{4nk}.$$

Combining the above with $\|{U'}^\top U' - U^\top U\|_2 \le \frac{\epsilon j}{3k}$ gives,

$$\Pr\left[\left\|\left(M^{(j+1)}U'\right)^\top M^{(j+1)}U' - U^\top U\right\|_{op} \le \epsilon\frac{j+1}{3k}\middle|\left(M^{(1)},\cdots,M^{(j)}\right)\in\mathscr{E}_j\right] \ge 1-\frac{\delta}{4nk}. \tag{5.12}$$

Furthermore, the spectral property of $M^{(j+1)}$ implies that for every column ${U'}^i$ of matrix U',

$$\left\|M^{(j+1)}{U'}^i\right\|_2^2 = \left(1\pm\frac{\epsilon}{3k}\right)\left\|{U'}^i\right\|_2^2,$$

with probability $1-\frac{\delta}{4nk}$. By a union bound over all $i\in[n]$, we have the following,

$$\Pr\left[\left\|M^{(j+1)}\cdot U'\right\|_F^2 \le \left(1+\frac{\epsilon}{3k}\right)\|U'\|_F^2\middle|\left(M^{(1)},\cdots,M^{(j)}\right)\in\mathscr{E}_j\right] \ge 1-\frac{\delta}{4k}.$$

Combining the above with $\|U'\|_F^2 \le \left(1+\frac{\epsilon}{3k}\right)^j\|U\|_F^2$, we find that

$$\Pr\left[\left\|M^{(j+1)}\cdot U'\right\|_F^2 \le \left(1+\frac{\epsilon}{3k}\right)^{j+1}\|U\|_F^2\middle|\left(M^{(1)},\cdots,M^{(j)}\right)\in\mathscr{E}_j\right] \ge 1-\frac{\delta}{4k}. \tag{5.13}$$

A union bound on (5.12) and (5.13) gives,

$$\Pr\left[\left(M^{(1)},\cdots,M^{(j+1)}\right)\in\mathscr{E}_{j+1}\middle|\left(M^{(1)},\cdots,M^{(j)}\right)\in\mathscr{E}_j\right] \ge 1-\frac{\delta}{4nk}-\frac{\delta}{4k} \ge 1-\frac{\delta}{2k}.$$

We also show that $\Pr\left[M^{(1)}\in\mathscr{E}_1\right]\ge 1-\delta/2k$. By the assumption of lemma we know that $M^{(1)}$ satisfies the $\left(2\mu_F+2,\mu_2+2,\frac{\epsilon}{3k},\frac{\delta}{4nk},n\right)$-spectral property. Therefore,

$$\Pr_{M^{(1)}}\left[\left\|\left(M^{(1)}U\right)^\top M^{(1)}U - U^\top U\right\|_{\text{op}} \le \frac{\epsilon}{3k}\right] \ge 1-\frac{\delta}{4nk}. \tag{5.14}$$

Additionally, for every column U^i of matrix U,

$$\left\|M^{(1)}U^i\right\|_2^2 = \left(1\pm\frac{\epsilon}{3k}\right)\left\|U^i\right\|_2^2,$$

with probability $1-\frac{\delta}{4nk}$. By a union bound over all $i\in[n]$, we find the following,

$$\Pr\left[\left\|M^{(1)}\cdot U\right\|_F^2 \le \left(1+\frac{\epsilon}{3k}\right)\|U\|_F^2\right] \ge 1-\frac{\delta}{4k}. \tag{5.15}$$

A union bound on (5.14) and (5.15) gives,

$$\Pr\left[M^{(1)}\in\mathscr{E}_1\right] \ge 1-\frac{\delta}{4nk}-\frac{\delta}{4k} \ge 1-\frac{\delta}{2k}.$$

By the chain rule we have,

$$\Pr\left[\left(M^{(1)},\cdots,M^{(k)}\right)\in\mathscr{E}_k\right]=\prod_{j=1}^{k}\Pr\left[\left(M^{(1)},\cdots M^{(j)}\right)\in\mathscr{E}_j\middle|\left(M^{(1)},\cdots M^{(j-1)}\right)\in\mathscr{E}_{j-1}\right]$$
$$\geq\left(1-\frac{\delta}{2k}\right)^k\geq 1-\delta,$$

which completes the proof of the lemma.

□

Before stating the main result of this section, we need to first present a claim that follows fom basic properties of tensor products and definition of the spectral property.

Claim 5.5.1. *For every $\epsilon,\delta>0$ and any sketch $S\in\mathbb{R}^{m\times d}$ such that $I_k\times S$ satisfies $(\mu_F,\mu_2,\epsilon,\delta,n)$-spectral property, the sketch $S\times I_k$ also satisfies the $(\mu_F,\mu_2,\epsilon,\delta,n)$-spectral property.*

Proof. Suppose $U\in\mathbb{R}^{dk\times n}$. Then, note that there exists $U'\in\mathbb{R}^{dk\times n}$ formed by permuting the rows of U such that $(S\times I_k)U$ and $(I_k\times S)U'$ are identical up to a permutation of the rows. (In particular, the (d,k)-reshaping of any column U^j of U is the transpose of the (k,d)-reshaping of the corresponding column U'^j of U'.) Then, observe that

$$U^\top U=U'^\top U',\ \text{ and }\ U^\top(S\times I_k)^\top(S\times I_k)U=U'^\top(I_k\times S)^\top(I_k\times S)U'.$$

Therefore,

$$\left\|U^\top(S\times I_k)^\top(S\times I_k)U-U^\top U\right\|_{\mathrm{op}}=\left\|U'^\top(I_k\times S)^\top(I_k\times S)U'-U'^\top U'\right\|_{\mathrm{op}}.$$

Moreover, since U and U' are identical up to a permutation of the rows, we have $\|U\|_{op}=\|U'\|_{op}$ and $\|U\|_F=\|U'\|_F$. The desired claim now follows easily. □

The following lemma shows that the sketch Π^q presented in definition 5.3.3 inherits the spectral property of Definition 5.5.1 from the base sketches. That is, if S_{base} and T_{base} are such that $I_{m^{q-2}}\times S_{\text{base}}$ and $I_{d^{q-1}}\times T_{\text{base}}$ satisfy the spectral property, then the sketch Π^q satisfies the spectral property.

Lemma 5.5.2. *For every positive integers n,d,m, any power of two integer q, any base sketches $S_{\text{base}}:\mathbb{R}^{m^2}\to\mathbb{R}^m$ and $T_{\text{base}}:\mathbb{R}^d\to\mathbb{R}^m$ such that both $I_{m^{q-2}}\times S_{\text{base}}$ and $I_{d^{q-1}}\times T_{\text{base}}$ satisfy the $\left(2\mu_F+2,\mu_2+2,\frac{\epsilon}{6q},\frac{\delta}{8nq},n\right)$-spectral property, then the sketch Π^q as defined in Definition 5.3.3 satisfies the $\left(\mu_F+1,\mu_2+1,\varepsilon,\delta,n\right)$-spectral property.*

Proof. We wish to show that the sketch $\Pi^q\equiv Q^qT^q$ as per Definition 5.3.3, satisfies the

$(\mu_F+1,\mu_2+1,\varepsilon,\delta,n)$-spectral property. By Lemma 5.4.3,

$$\Pi^q = M^{(2q-1)}M^{(2q)}\cdots M^{(1)},$$

where $M^{(i)}$ are independent matrices that satisfy $\left(2\mu_F+2,\mu_2+2,\frac{\varepsilon}{6q},\frac{\delta}{8nq},n\right)$-spectral property. That is by the assumption of the lemma about the spectral property of $I_{m^{q-2}}\times S_{\text{base}}$ and $I_{d^{q-1}}\times T_{\text{base}}$ together with Claim 5.5.1. Therefore, the Lemma readily follows by invoking Lemma 5.5.1 with $k=2q+1$. □

5.5.2 Spectral property of Identity×TensorSRHT and Identity×OSNAP

In this section, we show that tensoring an identity operator with our base sketches results in a matrix that satisfies the spectral property (Definition 5.5.1). We show that using either of TensorSRHT or OSNAP as the base sketch yields a transform that has the spectral property. The following lemma proves that tensoring identity with TensorSRHT (Definition 5.3.7) results in a transform that satisfies the spectral property with nearly optimal target dimension.

Lemma 5.5.3. *For any $\epsilon,\delta,\mu_2,\mu_F>0$ and any integers $n,k>0$, if $m=\Omega\left(\log\frac{n}{\delta}\log^2(\frac{ndk}{\epsilon\delta})\cdot\frac{\mu_F\mu_2}{\epsilon^2}\right)$ and $S\in\mathbb{R}^{m\times d}$ is a* TensorSRHT*, then the sketch $I_k\times S$ satisfies $(\mu_F,\mu_2,\epsilon,\delta,n)$-spectral property.*

The above lemma is proved in Appendix D.2.1. In the next lemma, we show that tensoring identity operator with OSNAP results in a sketch that satisfies the spectral property with nearly optimal target dimension as well as nearly optimal application time. This sketch is particularly efficient for sketching sparse vectors. We use a slightly different version than the original OSNAP to simplify the analysis, defined as follows.

Definition 5.5.2 (OSNAP transform)**.** For every positive integers s, m, and d, the OSNAP transform with sparsity parameter s and target dimension m is defined as,

$$S_{r,j}\overset{\text{def}}{=}\sqrt{\frac{1}{s}}\cdot\delta_{r,j}\cdot\sigma_{r,j},$$

for all $r\in[m]$ and all $j\in[d]$, where $\sigma_{r,j}\in\{-1,+1\}$ are i.i.d. uniform Rademacher random variables and $\delta_{r,j}$ are i.i.d. Bernoulli random variables satisfying, $\mathbb{E}[\delta_{r,i}]=s/m$.

Now we state the spectral property result for OSNAP which is proved in Appendix D.2.2.

Lemma 5.5.4. *For every $\epsilon,\delta,\mu_2,\mu_F>0$ and positive integer n, if $S\in\mathbb{R}^{m\times d}$ is an* OSNAP *sketch with sparsity parameter s, then the sketch $I_k\times S$ satisfies the $(\mu_F,\mu_2,\epsilon,\delta,n)$-spectral property, provided that $s=\Omega\left(\log^2\left(\frac{ndk}{\epsilon\delta}\right)\log\frac{n}{\delta}\cdot\frac{\mu_2^2}{\epsilon^2}\right)$ and $m=\Omega\left(\log\left(\frac{ndk\mu_F}{\epsilon\delta}\right)\log\frac{nd}{\delta}\cdot\frac{\mu_2\mu_F}{\epsilon^2}\right)$.*

5.5.3 High probability OSE with linear dependence on s_λ

We are ready to prove Theorem 5.1.2. We prove that if we instantiate Π^p (Definition 5.3.3) with T_{base} : OSNAP and S_{base} : TensorSRHT, it satisfies the Oblivious Subspace Embedding

guarantee.

Theorem 5.1.2. *For every positive integers p, d, n, every $\varepsilon, s_\lambda > 0$, there exists a distribution on linear sketches $\Pi^p \in \mathbb{R}^{m \times d^p}$ which is an $(\varepsilon, 1/\operatorname{poly} n, s_\lambda, d^p, n)$-oblivious subspace embedding as in Definition 5.1.2, provided that the integer m satisfies $m = \widetilde{\Omega}\left(p^4 s_\lambda/\epsilon^2\right)$.*
Moreover, for any $X \in \mathbb{R}^{d \times n}$, if $A \in \mathbb{R}^{d^p \times n}$ is the matrix whose columns are obtained by the p-fold self-tensoring of each column of X then $\Pi^p A$ can be computed in time $\widetilde{O}\left(pnm + p^5\epsilon^{-2}\operatorname{nnz}(X)\right)$.

Proof. Let $\delta = \frac{1}{\operatorname{poly}(n)}$ denote the failure probability. Let $m \approx \frac{p^4 s_\lambda}{\varepsilon^2}\log^3\left(\frac{nd}{\varepsilon\delta}\right)$ and $s \approx \frac{p^4}{\varepsilon^2}\log^3\left(\frac{nd}{\varepsilon\delta}\right)$. Let $\Pi^p \in \mathbb{R}^{m \times m^p}$ be the sketch defined in Definition 5.3.3, where $S_{\text{base}} \in \mathbb{R}^{m \times m^2}$ and $T_{\text{base}} \in \mathbb{R}^{m \times d}$ are TensorSRHT and OSNAP with sparsity parameter s, respectively. Let $q = 2^{\lceil \log_2(p) \rceil}$. By Lemma 5.4.1, it is sufficient to show that Π^q is an $(\varepsilon, \delta, s_\lambda, d^q, n)$-Oblivious Subspace Embedding. Consider arbitrary $A \in \mathbb{R}^{d^q \times n}$ and $\lambda \geq 0$ and let $U = A\left(A^\top A + \lambda I_n\right)^{-1/2}$. Let us denote the statistical dimension of A by $s_\lambda = s_\lambda(A^\top A)$. Therefore, $\|U\|_2 \leq 1$ and $\|U\|_F^2 = s_\lambda$. Since $q < 2p$, by Lemma 5.5.4, the transform $I_{d^{q-1}} \times T_{\text{base}}$, satisfies $\left(2s_\lambda + 2, 2, \frac{\varepsilon}{6q}, \frac{\delta}{8nq}, n\right)$-spectral property. Moreover, by Lemma 5.5.3, $I_{m^{q-2}} \times S_{\text{base}}$ satisfies $\left(2s_\lambda + 2, 2, \frac{\varepsilon}{6q}, \frac{\delta}{8nq}, n\right)$-spectral property. Therefore, by Lemma 5.5.2, Π^q satisfies $(s_\lambda + 1, 1, \varepsilon, \delta, n)$-spectral property. Hence,

$$\Pr\left[\left\|(\Pi^q U)^\top \Pi^q U - U^\top U\right\|_{\text{op}} \leq \varepsilon\right] \geq 1 - \delta.$$

Since $U^\top U = (A^\top A + \lambda I_n)^{-1/2} A^\top A (A^\top A + \lambda I_n)^{-1/2}$ and $\Pi^q U = \Pi^q A (A^\top A + \lambda I_n)^{-1/2}$ we find that $\Pr\left[(1-\epsilon)(A^\top A + \lambda I_n) \preceq (\Pi^q A)^\top \Pi^q A + \lambda I_n \preceq (1+\epsilon)(A^\top A + \lambda I_n)\right] \geq 1 - \delta$, which proves that Π^q is an $(\varepsilon, \delta, s_\lambda, d^q, n)$-Oblivious Subspace Embedding.

Runtime: By Lemma 5.3.1, if A is the matrix whose columns are obtained by p-fold self-tensoring of each column of $X \in \mathbb{R}^{d \times n}$ then $\Pi^p A$ can be computed using Algorithm 15. We bound the time of running Algorithm 15 on a vector x when S_{base} is TensorSRHT and T_{base} is OSNAP. Computing Y_j^0's for each j in lines 3 and 4 of algorithm requires applying an OSNAP sketch on either x or $\mathbf{e}_1$ which takes time $O(s \cdot \operatorname{nnz}(x))$. Therefore computing Y_j^0 for all j takes time $O(qs \cdot \operatorname{nnz}(x))$. Computing each of Y_j^l's in line 7 of algorithm amounts to applying a TensorSRHT with input dimension m^2 and target dimension m on $Y_{2j-1}^{l-1} \otimes Y_{2j}^{l-1}$. This takes time $O(m \log m)$. Therefore computing all Y_j^l's takes time $O(qm \log m)$. Note that $q \leq 2p$ hence the total time of running Algorithm 15 on a vector x is $O(pm \log m + ps \cdot \operatorname{nnz}(x))$. Therefore, sketching n columns of a matrix $X \in \mathbb{R}^{d \times n}$ takes time $O\left(pnm \log m + ps \cdot \operatorname{nnz}(X)\right)$. □

5.6 Oblivious Embedding of the Gaussian Kernel

In this section we show how to sketch the Gaussian kernel matrix by polynomial expansion and then applying our proposed sketch for the polynomial kernels.

Data-points with bounded ℓ_2 radius: Suppose that we are given a dataset of points $x_1, \cdots x_n \in \mathbb{R}^d$ such that for all $i \in [n]$, $\|x_i\|_2^2 \leq r$ for some positive value r. Consider the Gaussian kernel matrix $G \in \mathbb{R}^{n \times n}$ defined as $G_{i,j} \stackrel{\text{def}}{=} e^{-\|x_i - x_j\|_2^2/2}$ for all $i, j \in [n]$. We are interested in sketching the data points matrix X using a sketch $S_g : \mathbb{R}^d \to \mathbb{R}^m$ such that the following holds with probability $1 - \delta$,

$$(1-\epsilon)(G + \lambda I_n) \preceq \left(S_g(X)\right)^\top S_g(X) + \lambda I_n \preceq (1+\epsilon)(G + \lambda I_n).$$

Theorem 5.1.3. *For every $r > 0$, every positive integers n, d, and every $X \in \mathbb{R}^{d \times n}$ such that $\|x_i\|_2 \leq r$ for all $i \in [n]$, where x_i is the i^{th} column of X, suppose $G \in \mathbb{R}^{n \times n}$ is the Gaussian kernel matrix – i.e., $G_{j,k} = e^{-\|x_j - x_k\|_2^2/2}$ for all $j, k \in [n]$. There exists an algorithm that computes $S_g(X) \in \mathbb{R}^{m \times n}$ in time $\widetilde{O}\left(q^6 \epsilon^{-2} n s_\lambda + q^6 \epsilon^{-2} \operatorname{nnz}(X)\right)$ such that for every $\varepsilon, \lambda > 0$,*

$$\Pr_{S_g}\left[(1-\epsilon)(G + \lambda I_n) \preceq \left(S_g(X)\right)^\top S_g(X) + \lambda I_n \preceq (1+\epsilon)(G + \lambda I_n)\right] \geq 1 - \frac{1}{\operatorname{poly}(n)},$$

where $m = \widetilde{\Theta}\left(q^5 s_\lambda / \epsilon^2\right)$ and $q = \Theta(r^2 + \log \frac{n}{\epsilon\lambda})$ and s_λ is λ-statistical dimension of G.

Proof. Let $\delta = \frac{1}{\operatorname{poly}(n)}$ denote the failure probability. Note that $G_{i,j} = e^{-\|x_i\|_2^2/2} \cdot e^{x_i^\top x_j} \cdot e^{-\|x_j\|_2^2/2}$ for every $i, j \in [n]$. Let D be an $n \times n$ diagonal matrix with i^{th} diagonal entry $e^{-\|x_i\|_2^2/2}$ and let $K \in \mathbb{R}^{n \times n}$ be defined as $K_{i,j} = e^{x_i^\top x_j}$. Note that $DKD \equiv G$ and K is a positive definite kernel matrix. The Taylor series expansion of kernel K is $K \equiv \sum_{l=0}^{\infty} \frac{\left[X^{\otimes l}\right]^\top X^{\otimes l}}{l!}$. Therefore, G can be written as the following series,

$$G \equiv \sum_{l=0}^{\infty} \frac{\left[X^{\otimes l} D\right]^\top X^{\otimes l} D}{l!}.$$

Note that each term $\left[X^{\otimes l} D\right]^\top X^{\otimes l} D = D(X^{\otimes l})^\top X^{\otimes l} D$ is a positive definite kernel matrixand the statistical dimension of $\left[X^{\otimes l} D\right]^\top X^{\otimes l} D$ for every $l \geq 0$ is upper bounded by the statistical dimension of G through the following claim.

Claim 5.6.1. *For every $\mu \geq 0$ and every integer l, $s_\mu\left(\left[X^{\otimes l} D\right]^\top X^{\otimes l} D\right) \leq s_\mu(G)$.*

Proof. Using the fact that a polynomial kernel of any degree is positive definite, we find that $\left[X^{\otimes l} D\right]^\top X^{\otimes l} D \preceq G$. By Courant-Fischer's min-max theorem we have,

$$\lambda_j\left(\left[X^{\otimes l} D\right]^\top X^{\otimes l} D\right) = \max_{U \in \mathbb{R}^{(n-j+1) \times n}} \min_{\substack{\alpha \neq 0 \\ U\alpha = 0}} \frac{\alpha^\top \left[X^{\otimes l} D\right]^\top X^{\otimes l} D \alpha}{\|\alpha\|_2^2}.$$

Let U^* be the maximizer of the expression above. Then we have,

$$\begin{aligned}\lambda_j(G) &= \max_{U\in\mathbb{R}^{(n-j+1)\times n}} \min_{\substack{\alpha\neq 0\\ U\alpha=0}} \frac{\alpha^\top G\alpha}{\|\alpha\|_2^2}\\ &\geq \min_{\substack{\alpha\neq 0\\ U^*\alpha=0}} \frac{\alpha^\top G\alpha}{\|\alpha\|_2^2}\\ &\geq \min_{\substack{\alpha\neq 0\\ U^*\alpha=0}} \frac{\alpha^\top \left[X^{\otimes l}D\right]^\top X^{\otimes l}D\alpha}{\|\alpha\|_2^2}\\ &= \lambda_j\left(\left[X^{\otimes l}D\right]^\top X^{\otimes l}D\right).\end{aligned}$$

for every j. Therefore, the claim follows from the definition of statistical dimension,

$$s_\mu(G) = \sum_{j=1}^n \frac{\lambda_j(G)}{\lambda_j(G)+\mu} \geq \sum_{j=1}^n \frac{\lambda_j\left(\left[X^{\otimes l}D\right]^\top X^{\otimes l}D\right)}{\lambda_j\left(\left[X^{\otimes l}D\right]^\top X^{\otimes l}D\right)+\mu} = s_\mu\left(\left[X^{\otimes l}D\right]^\top X^{\otimes l}D\right).$$

□

If we let $P = \sum_{l=0}^q \frac{(X^{\otimes l})^\top X^{\otimes l}}{l!}$ and $q = 8r^2 + \ln\frac{n}{\epsilon\lambda}$, then by the triangle inequality we have,

$$\|K-P\|_{\mathrm{op}} \leq \sum_{l>q} \left\|\frac{\left[X^{\otimes l}\right]^\top X^{\otimes l}}{l!}\right\|_F \leq \sum_{l>q}\frac{n\cdot r^{2l}}{l!} \leq \epsilon\lambda/2.$$

Since P is a positive definite kernel matrix and all eigenvalues of D are bounded by 1, in order to prove the theorem it is sufficient to satisfy the following with probability $1-\delta$,

$$(1-\epsilon/3)(DPD+\lambda I_n) \preceq \left(S_g(X)\right)^\top S_g(X) + \lambda I_n \preceq (1+\epsilon/3)(DPD+\lambda I_n).$$

Let $\Pi^l \in \mathbb{R}^{m_l\times d^l}$ be the sketch defined as per Theorem 5.1.2 with $m_l = \Theta\left(l^4\log^3\frac{nd}{\delta}\cdot\frac{s_\lambda}{\epsilon^2}\right)$, where s_λ is the λ-statistical dimension of G. Therefore by Claim 5.6.1, with probability $1-\frac{\delta}{q+1}$:

$$\left(1-\frac{\epsilon}{9}\right)\left(\left[X^{\otimes l}D\right]^\top X^{\otimes l}D+\lambda I_n\right) \preceq \left[\Pi^l X^{\otimes l}D\right]^\top \Pi^l X^{\otimes l}D+\lambda I_n \preceq \left(1+\frac{\epsilon}{9}\right)\left(\left[X^{\otimes l}D\right]^\top X^{\otimes l}D+\lambda I_n\right). \quad (5.16)$$

Moreover, $\Pi^l X^{\otimes l}D$ can be computed using $O\left(lnm_l\log m_l + \frac{l^5}{\epsilon^2}\log^3\frac{nd}{\delta}\cdot\mathrm{nnz}(X)\right)$ runtime.

We let S_P be the linear mapping from $S_P : \mathbb{R}^{\sum_{l=0}^q d^l} \to \mathbb{R}^m$ defined as

$$S_P \stackrel{\mathrm{def}}{=} \left[\frac{1}{\sqrt{0!}}\Pi^0\right] \oplus \left[\frac{1}{\sqrt{1!}}\Pi^1\right] \oplus \left[\frac{1}{\sqrt{2!}}\Pi^2\right] \cdots \left[\frac{1}{\sqrt{q!}}\Pi^q\right].$$

Let Z be the matrix of size $\left(\sum_{l=0}^q d^l\right)\times n$ whose i^{th} column is $z_i = x_i^{\otimes 0} \oplus x_i^{\otimes 1} \oplus x_i^{\otimes 2} \cdots x_i^{\otimes q}$, where

x_i is the i^{th} column of X. The following holds for $[S_P Z]^\top S_P Z$,

$$[S_P Z]^\top S_P Z = \sum_{l=0}^{q} \frac{\left[\Pi^l X^{\otimes l}\right]^\top \Pi^l X^{\otimes l}}{l!}.$$

This trivially implies that,

$$[S_P ZD]^\top S_P ZD = \sum_{l=0}^{q} \frac{\left[\Pi^l X^{\otimes l} D\right]^\top \Pi^l X^{\otimes l} D}{l!}.$$

Therefore, by union bound, we find that (5.16) holds for all $0 \leq l \leq q$ with probability $1-\delta$, giving the following,

$$(1-\epsilon/3)(DPD+\lambda I_n) \preceq [S_P ZD]^\top S_P ZD + \lambda I_n \preceq (1+\epsilon/3)(DPD+\lambda I_n).$$

Now we define non-linear transformation $S_g : \mathbb{R}^d \to \mathbb{R}^m$ as follows,

$$S_g(x) \stackrel{\text{def}}{=} e^{-\|x\|_2^2/2}\left(\left[\frac{1}{\sqrt{0!}}\Pi^0\left(x^{\otimes 0}\right)\right] \oplus \left[\frac{1}{\sqrt{1!}}\Pi^1\left(x^{\otimes 1}\right)\right] \oplus \left[\frac{1}{\sqrt{2!}}\Pi^2\left(x^{\otimes 2}\right)\right] \cdots \left[\frac{1}{\sqrt{q!}}\Pi^q\left(x^{\otimes q}\right)\right]\right).$$

We have that $S_g(X) = S_P ZD$, therefore with probability $1-\delta$, the following holds,

$$(1-\epsilon)(G+\lambda I_n) \preceq \left(S_g(X)\right)^\top S_g(X) + \lambda I_n \preceq (1+\epsilon)(G+\lambda I_n).$$

Note that the target dimension of S_g is $m = m_0 + m_1 + \cdots + m_q \approx q^5 \log^3 \frac{nd}{\delta} \cdot \frac{s_\lambda}{\epsilon^2}$. Also, by Theorem 5.1.2, time to compute $S_g(X)$ is $O\left(\frac{q^6 n}{\epsilon^2}\log^4 \frac{nd}{\delta} \cdot s_\lambda + \frac{q^6}{\epsilon^2}\log^3 \frac{nd}{\delta} \cdot \text{nnz}(X)\right)$.

□

6 Conclusion

The research results mentioned in this thesis have shed light on–and sometimes fully resolved–a number of important questions in the theoretical foundations of data science while, at the same time, opening several promising avenues for further progress. We outline these directions below.

Sparse FFT: Theory vs Practice. In this thesis, we devised sparse FFT algorithms that achieve theoretically optimal sample complexity in sublinear time by exploiting *structure beyond sparsity*. Nonetheless, the theoretical guarantees ignore constant factors and in practice, the compressive sensing algorithms typically outperform the sparse FFT algorithms in terms of sample complexity at the expense of incurring a slow superlinear runtime. The practical sample complexity of learning sparse functions is crucially important because in applications like hyperparameter tuning of massive neural networks (Hazan et al., 2018) every data sample requires running a computationally expensive training process. On the other hand, the compressive sensing methods are both theoretically and practically slow with (at least) linear runtime which makes them intractable for the hyperparameter tuning applications because their runtime scales exponentially in the number of hyperparameter of the neural network. As of now, it is not known how to achieve the practically optimal sample complexity of compressive sensing methods and fast runtime at the same time, so, a very interesting open question is *unifying theory and practice of sparse FFT algorithms*.

Optimal Kernel Embeddings: a Unified Method. The results we presented in this thesis make exciting progress in the problem of spectrally approximating kernel matrices, by giving near-optimal oblivious sketching of the Polynomial kernel as well as near-optimal characterization of the Gaussian kernel using Fourier sampling. However, these methods are tailored to the specific choice of the kernel function and do not extend to for instance, non-smooth kernels such as Laplacian kernel or kernels with slowly decaying tails such as Cauchy. As a result, there is no overarching kernel approximation method that works optimally for a wide class of kernel problems. So, a very interesting open problem is to design a *unified* kernel embedding tech-

nique that works optimally for a wide range of kernel functions including Cauchy, Laplacian, Gaussian, and Polynomial.

A Supplementary Materials for Chapter 1

A.1 Fourier Downsampling via Compactly Supported Flat Filters

A.1.1 Flat filters with compact support

Our filter construction is similar to (Indyk and Kapralov, 2014), but we prove and utilize different properties, and hence provide the details for completeness.

Definition A.1.1 (Rectangular pulse)**.** For an even integer B', let $\mathrm{rect}_{B'}$ denote the rectangular pulse of width $B'-1$, i.e.,

$$\mathrm{rect}_{B'}(t) = \begin{cases} 1, & \text{if } |t| < \frac{B'}{2} \\ 0 & \text{otherwise.} \end{cases}$$

For an integer $B' > 0$ a power of 2, define the length-n signal,

$$W(\cdot) = \left(\frac{n}{B'-1} \cdot \mathrm{rect}_{B'}\right) \star \cdots \star \left(\frac{n}{B'-1} \cdot \mathrm{rect}_{B'}\right), \tag{A.1}$$

where the convolution is performed F times. As noted in (Indyk and Kapralov, 2014), we have $\mathrm{supp}\,(W) \subseteq [-FB', FB']$, and the Fourier transform is given by,

$$\widehat{W}_f = \left(\frac{1}{B'-1} \sum_{|f'|<\frac{B'}{2}} \omega_n^{ff'}\right)^F = \left(\frac{\sin(\pi(B'-1)f/n)}{(B'-1)\sin(\pi f/n)}\right)^F \tag{A.2}$$

for $f \neq 0$, and $W_0 = 1$.

Lemma A.1.1. (Properties of W) *For every even $F \geq 2$, the following hold for the signal W defined in* (A.1)–(A.2)*:*

1 $\widehat{W}_f \in [0,1]$ *for all* $f \in [n]$*;*

2 *There exists an absolute constant $C \geq 0$ such that for every $\lambda > 1$,*

$$\sum_{f \in [n], |f| \geq \frac{\lambda \cdot n}{2B'}} \widehat{W}_f \leq (C/\lambda)^{F-1} \sum_{f \in [n]} \widehat{W}_f.$$

Proof. First note that the maximum of $\widehat{W}_f$ is achieved at 0 and equals 1. Since F is even by assumption, we have from (A.2) that $\widehat{W}_f \geq 0$ for all f. These two facts establish the first claim.

To prove the second claim, note that for all $f \in [n]$, we have

$$\begin{aligned} \widehat{W}_f = \left| \frac{\sin(\pi(B'-1)f/n)}{(B'-1)\sin(\pi f/n)} \right|^F &\leq \left| \frac{1}{(B'-1)\sin(\pi f/n)} \right|^F \quad (\text{since } |\sin(\pi x)| \leq 1) \\ &\leq \left| \frac{1}{(B'-1)2|f|/n} \right|^F \quad (\text{since } |\sin(\pi x)| \geq 2|x| \text{ for } |x| \leq 1/2). \end{aligned} \tag{A.3}$$

We claim that this can be weakened to

$$\widehat{W}_f \leq \left(\frac{n}{B'|f|} \right)^F. \tag{A.4}$$

For $f \in [-n/B', n/B']$ the right-hand side is at least one, and hence this claim follows directly from the first claim above. On the other hand, if $|f| \geq n/B'$, we have

$$\begin{aligned} 2(B'-1)|f|/n &= 2B'|f|/n - 2|f|/n \\ &\geq 2B'|f|/n - 1 \quad (\text{since } |f| \leq n/2) \\ &\geq B'|f|/n \quad (\text{since } |f| \geq n/B'), \end{aligned}$$

and hence (A.4) follows from (A.3).

Using (A.4), we have

$$\sum_{|f| \geq \frac{\lambda \cdot n}{2B'}} \widehat{W}_f \leq \sum_{|f| \geq \frac{\lambda \cdot n}{2B'}} \left(\frac{n}{B'|f|} \right)^F = O(\lambda)^{-F+1} \cdot \frac{n}{B'}. \tag{A.5}$$

At the same time, for any $f \in [-\frac{n}{2B'}, \frac{n}{2B'}]$, we have

$$\begin{aligned} \widehat{W}_f &= \left| \frac{\sin(\pi(B'-1)f/n)}{(B'-1)\sin(\pi f/n)} \right|^F \\ &\geq \left| \frac{2(B'-1)f/n}{(B'-1)\sin(\pi f/n)} \right|^F \quad (\text{since } |\sin(\pi x)| \geq 2|x| \text{ for } |x| \leq 1/2) \\ &\geq \left| \frac{2(B'-1)f/n}{(B'-1)\pi(f/n)} \right|^F \quad (\text{since } |\sin(\pi x)| \leq \pi|x|) \\ &= \left(\frac{2}{\pi} \right)^F. \end{aligned}$$

This means that

$$\sum_{f\in[n]} \widehat{W}_f \geq \sum_{f\in[-\frac{n}{2B'},\frac{n}{2B'}]} \widehat{W}_f \geq \left(\frac{2}{\pi}\right)^F \cdot \frac{n}{B'}. \tag{A.6}$$

Putting (A.5) together with (A.6), we get

$$\sum_{f\in[n],|f|\geq\frac{\lambda\cdot n}{2B'}} \widehat{W}_f = O(\lambda)^{-F+1}\cdot\frac{n}{B'} \leq \left(\frac{\pi}{2}\right)^F (C'\cdot\lambda)^{-F+1}\cdot\sum_{f\in[n]} W_f$$

for an absolute constant $C'>0$. The desired claim follows by setting $C = C'(\pi/2)$. □

We now fix an integer B, and define $\widehat{G}$ by

$$\widehat{G}_f = \frac{1}{Z}\sum_{\Delta=-\frac{3n}{4B}}^{\frac{3n}{4B}} \widehat{W}_{f-\Delta}.$$

where $Z = \sum_{f\in[n]}\widehat{W}_f$. By interpreting this as a convolution with a rectangle, we obtain that the inverse Fourier transform G_t is obtained via the multiplication of W_t with a sinc pulse.

We proceed by showing that, upon identifying $B' = 8CB$ (where B' was used in defining $\widehat{W}$, and C is the implied constant in Lemma A.1.1), this filter satisfies the claims of Lemma 1.2.1. We start with the three properties in Definition 1.2.1.

Proof of Lemma 1.2.1: (filter property 1) For every f, we have,

$$\widehat{G}_f = \frac{1}{Z}\cdot\sum_{\Delta=-\frac{3n}{4B}}^{\frac{3n}{4B}}\widehat{W}_{f-\Delta} \leq \frac{1}{Z}\sum_{\Delta\in[n]}\widehat{W}_{f-\Delta} = 1.$$

Similarly, the non-negativity of $\widehat{G}$ follows directly from that of $\widehat{W}$.

(filter property 2) For every $f\in[n]$ with $|f|\leq\frac{n}{2B}$, we have,

$$\begin{aligned}
\widehat{G}_f &= \frac{1}{Z}\cdot\sum_{\Delta=-\frac{3n}{4B}}^{\frac{3n}{4B}}\widehat{W}_{f-\Delta}\\
&= 1-\frac{1}{Z}\sum_{|\Delta|>\frac{3n}{4B}}\widehat{W}_{f-\Delta}\\
&\geq 1-\frac{2}{Z}\sum_{f'>\frac{n}{4B}}\widehat{W}_{f'} \quad \text{(since } |f|\leq\frac{n}{2B} \text{ and } W \text{ is symmetric)}\\
&= 1-\frac{2}{Z}\sum_{f'>\frac{B'}{2B}\cdot\frac{n}{2B'}}\widehat{W}_{f'}\\
&\geq 1-\left(2C\frac{B}{B'}\right)^{F-1}. \quad \text{(by Lemma A.1.1)}
\end{aligned}$$

Since $B'/B = 8C$ by our choice of B' above, we get $\widehat{G}_f \geq 1 - (1/4)^{F-1}$, as required.

(filter property 3) For every $f \in [n]$ with $|f| \geq \frac{n}{B}$, we have,

$$\begin{aligned}\widehat{G}_f &= \frac{1}{Z} \cdot \sum_{\Delta=-\frac{3n}{4B}}^{\frac{3n}{4B}} \widehat{W}_{f-\Delta} \\ &\leq \frac{1}{Z} \cdot \sum_{f':|f'|\geq|f|-\frac{3n}{4B}} \widehat{W}_{f'} \quad (\text{by } |f| \geq \frac{n}{B}).\end{aligned}$$

Defining $\zeta \geq 1$ such that $|f| = (3+\zeta)\frac{n}{4B}$, this becomes,

$$\begin{aligned}\widehat{G}_f &\leq \frac{1}{Z} \cdot \sum_{f':|f'|\geq\frac{\zeta n}{4B}} \widehat{W}_{f'} \\ &= \frac{1}{Z} \cdot \sum_{f':|f'|\geq\frac{\zeta B'}{2B}\cdot\frac{n}{2B'}} \widehat{W}_{f'} \\ &\leq \left(\frac{2CB}{\zeta B'}\right)^{F-1} \quad (\text{by Lemma A.1.1}) \\ &= \left(\frac{1}{4\zeta}\right)^{F-1} \quad (\text{since } B' = 8CB).\end{aligned}$$

Rearranging the definition of ζ, we obtain $\zeta = \frac{4B|f|}{n} - 3$, and hence $\zeta \geq \frac{B|f|}{n}$ due to the fact that $|f| \geq \frac{n}{B}$. Therefore, $\widehat{G}_f \leq \left(\frac{n}{4B|f|}\right)^{F-1}$. □

Proof of Lemma 1.2.1:(additional property 1) We have already shown that W is supported on a window of length $O(FB') = O(FB)$ centered at zero. The same holds for G since it is obtained via a pointwise multiplication of W with a sinc pulse.

(additional property 2) Since $\widehat{G}_f \in [0,1]$, the total energy across $|f| < \frac{n}{B}$ is at most $\frac{2n}{B}$. On the other hand, we have from the third property in Definition 1.2.1 that

$$\begin{aligned}\sum_{|f|\geq\frac{n}{B}} |\widehat{G}_f|^2 &\leq 2\sum_{f\geq\frac{n}{B}} \left(\frac{1}{4}\right)^{2(F-1)} \left(\frac{n}{Bf}\right)^{2(F-1)} \\ &\leq \frac{1}{8}\sum_{f\geq\frac{n}{B}} \left(\frac{n}{Bf}\right)^2 \quad (\text{since } F \geq 2) \\ &\leq \frac{1}{8}\cdot\frac{n}{B}\sum_{f=1}^{\infty}\frac{1}{f^2} \\ &\leq \frac{n}{B} \quad \left(\text{since } \sum_{f=1}^{\infty}\frac{1}{f^2} < 8\right).\end{aligned}$$

Combining this with the contribution from $|f| < \frac{n}{B}$ concludes the proof. □

Proof of Lemma 1.4.3:

For brevity, let $\Psi = \sum_{f' \neq f} |\widehat{X}_{f'}|^2 \mathbb{E}_\pi\big[|G_{o_f(f')}|^2\big]$ denote the left-hand side of (1.14). Following the approach of (Indyk and Kapralov, 2014, Lemma 3.3), we define the intervals $\mathscr{F}_t = \big(\pi(f) - \frac{n}{B}2^t, \pi(f) + \frac{n}{B}2^t\big]$ for $t = 1, \ldots, \log_2 b$, and write

$$\begin{aligned}\Psi &\leq \sum_{f' \neq f} |\widehat{X}_{f'}|^2 \sum_{t=1}^{\log_2 B} \Pr[\pi(f') \in \mathscr{F}_t \backslash \mathscr{F}_{t-1}] \max_{f'' : \pi(f'') \in \mathscr{F}_t \backslash \mathscr{F}_{t-1}} |\widehat{G}_{o_f(f'')}|^2 \\ &\leq \frac{4}{B} \sum_{f' \neq f} |\widehat{X}_{f'}|^2 \Big(2 + \sum_{t=2}^{\log_2 B} 2^t \max_{f'' : \pi(f'') \in \mathscr{F}_t \backslash \mathscr{F}_{t-1}} |\widehat{G}_{o_f(f'')}|^2\Big), \end{aligned} \tag{A.7}$$

where the second line follows by (i) upper bounding $\Pr[\pi(f') \in \mathscr{F}_t \backslash \mathscr{F}_{t-1}] \leq \Pr[\pi(f') \in \mathscr{F}_t]$ and applying the approximate pairwise independence property (*cf.*, Definition 1.4.1); (ii) using the fact that there are at most $\frac{n}{B} \cdot 2^{t+1}$ integers within $\mathscr{F}_t$, and applying $|\widehat{G}_f| \leq 1$ for the case $t = 1$.

To handle the term containing $|\widehat{G}_{o_f(f'')}|^2$, we use the triangle inequality to write

$$\begin{aligned}|o_f(f'')| &\geq |\pi(f) - \pi(f'')| - \Big|\pi(f) - \frac{n}{b}\mathrm{round}\big(\pi(f)\frac{B}{n}\big)\Big| \\ &\geq |\pi(f) - \pi(f'')| - \frac{n}{B}.\end{aligned}$$

For any f'' with $\pi(f'') \notin \mathscr{F}_{t-1}$, we have $|\pi(f) - \pi(f'')| \geq \frac{n}{B}2^{t-1}$, and hence $|o_f(f'')| \geq \frac{n}{B}(2^{t-1} - 1)$. As a result, for $t \geq 2$, the third property in Definition 1.2.1 gives

$$\widehat{G}_{o_f(f'')} \leq \Big(\frac{1}{4}\Big)^{F-1}\Big(\frac{1}{2^{t-1}-1}\Big)^{F-1} \leq \Big(\frac{1}{4}\Big)^{F-1}\Big(\frac{1}{2^{t-2}}\Big)^{F-1} = \Big(\frac{1}{2^t}\Big)^{F-1},$$

and hence

$$\sum_{t=2}^{\log_2 B} 2^t \max_{f'' : \pi(f'') \in \mathscr{F}_t \backslash \mathscr{F}_{t-1}} |\widehat{G}_{o_f(f'')}|^2 \leq \sum_{t=2}^{\infty} \Big(\frac{1}{2^t}\Big)^{2F-1}.$$

This sum is less than $\frac{1}{2}$ for all $F \geq 2$, and hence substitution into (A.7) gives $\Psi \leq \frac{10}{B}\|\widehat{X}\|_2^2$, as desired. □

A.1.2 Optimal downsampling

We are interested in the behavior of $\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2$ for each j (first part), and summed over all j (second part). We therefore begin with the following lemma, bounding this summation in terms of the signal X and the filter G.

Lemma A.1.2. (Initial downsampling bound) *For any integers (n, k_1), parameter $\delta \in \big(0, \frac{1}{20}\big)$, signal $X \in \mathbb{C}^n$ and its corresponding (k_1, δ)-downsampling $\{Z^r\}_{r \in [2k_1]}$, the following holds for all $j \in \Big[\frac{n}{k_1}\Big]$:*

$$\Big|\frac{1}{2k_1}\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2 - \sum_{f=1}^{n} |\widehat{G}_{f - k_1 j}|^2 \cdot |\widehat{X}_f|^2\Big| \leq 3\delta \sum_{f=1}^{n} |\widehat{G}_{f-k_1 j}| \cdot |\widehat{X}_f|^2.$$

Proof. **Directly evaluating the sum:** Using the definition of the signals $\widehat{Z}^r$ in (1.2), we write,

$$\sum_{r\in[2k_1]}|\widehat{Z}_j^r|^2=\sum_{r\in[2k_1]}\Big(\sum_{f=1}^{n}\widehat{G}_{f-k_1j}\cdot\widehat{X}_f\cdot\omega_n^{a_rf}\Big)^*\Big(\sum_{f'=1}^{n}\widehat{G}_{f'-k_1j}\cdot\widehat{X}_{f'}\cdot\omega_n^{a_rf'}\Big)$$
$$=\sum_{f=1}^{n}\sum_{f'=1}^{n}\widehat{G}^*_{f-k_1j}\cdot\widehat{X}_f^*\cdot\widehat{G}_{f'-k_1j}\cdot\widehat{X}_{f'}\cdot\Big(\sum_{r\in[2k_1]}\omega_n^{a_r(f'-f)}\Big)$$

where $(\cdot)^*$ denotes the complex conjugate. Since $a_r=\frac{nr}{2k_1}$, the term $\sum_{r\in[2k_1]}\omega_n^{a_r(f'-f)}$ is equal to $2k_1$ if $f-f'$ is a multiple of $2k_1$ (including $f=f'$) and zero otherwise, yielding,

$$\sum_{r\in[2k_1]}|\widehat{Z}_j^r|^2=2k_1\cdot\sum_{f=1}^{n}\Bigg(|\widehat{G}_{f-k_1j}|^2\cdot|\widehat{X}_f|^2+\sum_{\substack{j'\in[\frac{n}{2k_1}]\\ j'\neq0}}\widehat{G}^*_{f-k_1j}\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'-k_1j}\cdot\widehat{X}_{f-2k_1j'}\Bigg).\tag{A.8}$$

Without loss of generality, we can assume that $j=0$; otherwise, we can simply consider a version of X shifted in frequency domain by k_1j. Setting $j=0$ in (A.8) and applying the triangle inequality, we obtain,

$$\Big|\sum_{r\in[2k_1]}|\widehat{Z}_0^r|^2-2k_1\cdot\sum_{f=1}^{n}|\widehat{G}_f|^2\cdot|\widehat{X}_f|^2\Big|\leq2k_1\cdot\sum_{\substack{|j'|\leq\frac{n}{2(2k_1)}\\ j'\neq0}}\sum_{f=1}^{n}\Big|\widehat{G}_f^*\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'}\cdot\widehat{X}_{f-2k_1j'}\Big|.\tag{A.9}$$

Bounding the right-hand side of (A.9)**:** We write,

$$\sum_{\substack{|j'|\leq\frac{n}{2(2k_1)}\\ j'\neq0}}\sum_{f=1}^{n}\Big|\widehat{G}_f^*\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'}\cdot\widehat{X}_{f-2k_1j'}\Big|$$
$$=\sum_{\substack{|j'|\leq\frac{n}{2(2k_1)}\\ j'\neq0}}\sum_{f=1}^{n}\Big(|\widehat{G}_f|^{1/2}\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\Big)\cdot|\widehat{G}_f|^{1/2}\cdot|\widehat{X}_f^*|\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\cdot|\widehat{X}_{f-2k_1j'}|.\tag{A.10}$$

In Lemma A.1.3 below, we show that,

$$|\widehat{G}_f|^{1/2}\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\leq\frac{(\frac{1}{2})^{F-1}}{|j'|^{(F-1)/2}}$$

for all $f\in[n]$ and all $|j'|\leq\frac{n}{2(2k_1)}$ with $j'\neq0$. Definition 1.2.2 ensures that $\big(\frac{1}{2}\big)^{F-1}\leq\delta$, and substitution into (A.10) gives,

$$\sum_{\substack{|j'|\leq\frac{n}{2(2k_1)}\\ j'\neq0}}\sum_{f=1}^{n}\Big|\widehat{G}_f^*\cdot\widehat{X}_f^*\cdot\widehat{G}_{f-2k_1j'}\cdot\widehat{X}_{f-2k_1j'}\Big|$$
$$\leq\delta\cdot\sum_{\substack{|j'|\leq\frac{n}{2(2k_1)}\\ j'\neq0}}\frac{1}{|j'|^{(F-1)/2}}\sum_{f=1}^{n}|\widehat{G}_f|^{1/2}\cdot|\widehat{X}_f^*|\cdot|\widehat{G}_{f-2k_1j'}|^{1/2}\cdot|\widehat{X}_{f-2k_1j'}|.\tag{A.11}$$

Next, we apply Cauchy-Schwarz inequality to upper bound the inner summation over f above for any fixed $j' \in [\frac{n}{2k_1}]$, yielding,

$$\sum_{f=1}^{n} |\widehat{G}_f|^{1/2} \cdot |\widehat{X}_f^*| \cdot |\widehat{G}_{f-2k_1 j'}|^{1/2} \cdot |\widehat{X}_{f-2k_1 j'}| \leq \sqrt{\sum_{f=1}^{n} |\widehat{G}_f| \cdot |\widehat{X}_f^*|^2} \cdot \sqrt{\sum_{f=1}^{n} |\widehat{G}_{f-2k_1 j'}| \cdot |\widehat{X}_{f-2k_1 j'}|^2}$$
$$= \sum_{f=1}^{n} |\widehat{G}_f| \cdot |\widehat{X}_f^*|^2, \tag{A.12}$$

where we used the fact that $\{|\widehat{G}_{f-2k_1 j'}| \cdot |\widehat{X}_{f-2k_1 j'}|^2\}_{f=1}^n$ is a permutation of $\{|\widehat{G}_f| \cdot |\widehat{X}_f^*|^2\}_{f=1}^n$.

Wrapping up: Substituting (A.12) into (A.11) gives,

$$\sum_{\substack{|j'| \leq \frac{n}{2(2k_1)} \\ j' \neq 0}} \sum_{f=1}^{n} \left| \widehat{G}_f^* \cdot \widehat{X}_f^* \cdot \widehat{G}_{f-2k_1 j'} \cdot \widehat{X}_{f-2k_1 j'} \right|$$
$$\leq \delta \cdot \sum_{\substack{|j'| \leq \frac{n}{2(2k_1)} \\ j' \neq 0}} \frac{1}{|j'|^{(F-1)/2}} \sum_{f=1}^{n} |\widehat{G}_f| \cdot |\widehat{X}_f|^2$$
$$\leq 3\delta \sum_{f=1}^{n} |\widehat{G}_f| \cdot |\widehat{X}_f|^2,$$

where the last inequality follows from the fact that $\sum_{\substack{|j'| \leq \frac{n}{2(2k_1)} \\ j' \neq 0}} \frac{1}{|j'|^{(F-1)/2}} \leq 2\sum_{j'=1}^{\infty} \frac{1}{|j'|^{(F-1)/2}}$, which is upper bounded by 3 for $F \geq 8$, a condition guaranteed by Definition 1.2.2. We therefore obtain the following bound from (A.9),

$$\left| \frac{1}{2k_1} \cdot \sum_{r \in [2k_1]} |\widehat{Z}_0^r|^2 - \sum_{f=1}^{n} |\widehat{G}_f|^2 \cdot |\widehat{X}_f|^2 \right| \leq 3\delta \sum_{f=1}^{n} |\widehat{G}_f| \cdot |\widehat{X}_f|^2.$$

The lemma follows by recalling that the choice $j = 0$ was without loss of generality, with the general case amounting to replacing $\widehat{Z}_0$ by $\widehat{Z}_j$ and $\widehat{G}_f$ by $\widehat{G}_{f-k_1 j}$. □

In the preceding proof, we made use of the following technical result bounding the product of the filter G evaluated at two locations separated by some multiple of $2k_1$.

Lemma A.1.3. (Additional filter property) *Given n, k_1 and a parameter $F \geq 2$, if G is an $\left(n, \frac{n}{k_1}, F\right)$-flat filter, then for all $f \in [n]$ and all $j' \in \left[\frac{n}{k_1}\right]$ with $|j'| \leq \frac{n}{2(2k_1)}$ and $j' \neq 0$, one has $|\widehat{G}_f|^{1/2} \cdot |\widehat{G}_{f-2k_1 j'}|^{1/2} \leq \frac{(\frac{1}{2})^{F-1}}{|j'|^{(F-1)/2}}$.*

Proof. For clarity, let f_1 and f_2 denote the frequencies corresponding to f and $f - 2k_1 j'$ respectively, defined in the range $(-n/2, n/2]$ according to modulo-n arithmetic. By definition, $f_1 - f_2$ is equal to $2k_1 j'$ modulo-n, and since $|j'| \leq \frac{n}{2(2k_1)}$, we have $|2k_1 j'| \leq \frac{n}{2}$. This immediately implies that the distance $\Delta = |f_1 - f_2|$ according to *regular arithmetic* is lower bounded by

the distance according to modulo-n arithmetic: $\Delta \geq 2k_1|j'|$. Since f_1 and f_2 are at distance Δ according to regular arithmetic, it must be the case that either $|f_1| \geq \frac{\Delta}{2}$ or $|f_2| \geq \frac{\Delta}{2}$. Moreover, since $j' \neq 0$, we have, from the above-established fact $\Delta \geq 2k_1|j'|$, that $\frac{\Delta}{2} \geq k_1$, and hence we can apply the third filter property in Definition 1.2.1 to conclude that $|G_{f_v}| \leq \left(\frac{1}{4}\right)^{F-1}\left(\frac{2k_1}{\Delta}\right)^{F-1}$ for either $v = 1$ or $v = 2$. Substituting $\Delta \geq 2k_1|j'|$, upper bounding $G_{f_{v'}} \leq 1$ (*cf.*, Definition 1.2.1) for the index $v' \in \{1,2\}$ differing from v, and taking the square root, we obtain the desired result. □

We are now in a position to prove Lemma 1.2.3

Proof of the first part of Lemma 1.2.3: Recall from Lemma A.1.2 that

$$\left|\frac{1}{2k_1}\sum_{r\in[2k_1]}|\widehat{Z}_j^r|^2 - \sum_{f=1}^{n}|\widehat{G}_{f-k_1 j}|^2\cdot|\widehat{X}_f|^2\right| \leq 3\delta\sum_{f=1}^{n}|\widehat{G}_{f-k_1 j}|\cdot|\widehat{X}_f|^2. \tag{A.13}$$

We proceed by lower bounding $\sum_{f=1}^{n}|\widehat{G}_{f-k_1 j}|^2\cdot|\widehat{X}_f|^2$ and upper bounding $\sum_{f=1}^{n}|\widehat{G}_{f-k_1 j}|\cdot|\widehat{X}_f|^2$. Starting with the former, recalling that $I_j = \big((j-1/2)k_1, (j+1/2)k_1\big] \cap \mathbb{Z}$, we have,

$$\begin{aligned}\sum_{f=1}^{n}|\widehat{G}_{f-k_1 j}|^2\cdot|\widehat{X}_f|^2 &\geq \sum_{f\in I_j}|\widehat{G}_{f-k_1 j}|^2\cdot|\widehat{X}_f|^2 \\ &\geq \left(1-\left(\frac{1}{4}\right)^{F-1}\right)^2\left\|\widehat{X}_{I_j}\right\|_2^2 \geq (1-\delta)\left\|\widehat{X}_{I_j}\right\|_2^2,\end{aligned} \tag{A.14}$$

where the second line is by the second filter property given in Definition 1.2.1, and the third line is by the choice of F made in Definition 1.2.2. Next, we upper bound $\sum_{f=1}^{n}|\widehat{G}_{f-k_1 j}|\cdot|\widehat{X}_f|^2$. We can write,

$$\sum_{f=1}^{n}|\widehat{G}_{f-k_1 j}|\cdot|\widehat{X}_f|^2 = \sum_{f\in I_j\cup I_{j-1}\cup I_{j+1}}|\widehat{G}_{f-k_1 j}|\cdot|\widehat{X}_f|^2 + \sum_{f\in[n]:|f-k_1 j|\geq\frac{3k_1}{2}}|\widehat{G}_{f-k_1 j}|\cdot|\widehat{X}_f|^2. \tag{A.15}$$

By the third property in Definition 1.2.1, the filter decays as $|\widehat{G}_f| \leq (\frac{1}{4})^{F-1}(\frac{k_1}{|f|})^{F-1}$ for $|f| \geq k_1$, and therefore the second term in (A.15) is bounded by,

$$\begin{aligned}\sum_{f\in[n]:|f-k_1 j|\geq\frac{3k_1}{2}}|\widehat{G}_{f-k_1 j}|\cdot|\widehat{X}_f|^2 &\leq \left(\frac{1}{4}\right)^{F-1}\cdot\sum_{j'\in[\frac{n}{k_1}]:|j'-j|\geq 2}\frac{\left\|\widehat{X}_{I_{j'}}\right\|_2^2}{(|j'-j|-1)^{F-1}} \\ &\leq \left(\frac{1}{2}\right)^{F-1}\cdot\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\left\|\widehat{X}_{I_{j'}}\right\|_2^2}{|j'-j|^{F-1}} \\ &\leq \delta\cdot\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\left\|\widehat{X}_{I_{j'}}\right\|_2^2}{|j'-j|^{F-1}},\end{aligned} \tag{A.16}$$

where the second line follows from $|j'-j|-1 \geq \frac{|j'-j|}{2}$, and the third line follows since the choice of F in Definition 1.2.2 ensures that $\left(\frac{1}{2}\right)^{F-1} \leq \delta$. We bound the term $\sum_{f \in I_j \cup I_{j-1} \cup I_{j+1}} |\widehat{G}_{f-k_1 j}| \cdot |\widehat{X}_f|^2$ in (A.15) using the first filter property in Definition 1.2.1, namely, $\widehat{G}_f \leq 1$,

$$\sum_{f \in I_j \cup I_{j-1} \cup I_{j+1}} |\widehat{G}_{f-k_1 j}| \cdot |\widehat{X}_f|^2 \leq \left\| \widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}} \right\|_2^2. \tag{A.17}$$

Hence, combining (A.15)–(A.17), we obtain,

$$\sum_{f=1}^{n} |\widehat{G}_{f-k_1 j}| \cdot |\widehat{X}_f|^2 \leq \left\| \widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}} \right\|_2^2 + \delta \cdot \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\left\| \widehat{X}_{I_{j'}} \right\|_2^2}{|j'-j|^{F-1}}. \tag{A.18}$$

The first claim of the lemma follows by combining (A.13), (A.14), and (A.18). □

Proof of the second part of Lemma 1.2.3: By following the same steps as those used to handle (A.15), we obtain the following analog of (A.18) with $|\widehat{G}_f|^2$ in place of $|\widehat{G}_f|$:

$$\sum_{f=1}^{n} |\widehat{G}_{f-k_1 j}|^2 \cdot |\widehat{X}_f|^2 \leq \left\| \widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}} \right\|_2^2 + \delta \cdot \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\left\| \widehat{X}_{I_{j'}} \right\|_2^2}{|j'-j|^{2(F-1)}}. \tag{A.19}$$

Combining (A.13), (A.18), and (A.19), we obtain,

$$\begin{aligned} \frac{\sum_{r \in [2k_1]} |\widehat{Z}_j^r|^2}{2k_1} &\leq \left\| \widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}} \right\|_2^2 + \delta \cdot \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\left\| \widehat{X}_{I_{j'}} \right\|_2^2}{|j'|^{2(F-1)}} \\ &\quad + 3\delta \cdot \left(\left\| \widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}} \right\|_2^2 + \delta \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\left\| \widehat{X}_{I_{j'}} \right\|_2^2}{|j'-j|^{F-1}} \right). \end{aligned}$$

Summing over all $j \in [n]$ gives,

$$\begin{aligned} \frac{1}{2k_1} \sum_{r \in [2k_1]} \|\widehat{Z}^r\|^2 &\leq \sum_{j \in [n]} \left((1+3\delta) \left\| \widehat{X}_{I_j \cup I_{j-1} \cup I_{j+1}} \right\|_2^2 + (3\delta^2+\delta) \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \right) \\ &= 3(1+3\delta)\|\widehat{X}\|_2^2 + (3\delta^2+\delta) \sum_{j \in [n]} \sum_{j' \in [\frac{n}{k_1}] \setminus \{j\}} \frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}} \end{aligned} \tag{A.20}$$

The double summation is upper bounded by $\sum_{j' \in [n]} \|\widehat{X}_{I_{j'}}\|_2^2 \cdot 2\sum_{\Delta=1}^{\infty} \frac{1}{\Delta^{F-1}} = 2\|\widehat{X}\|_2^2 \cdot \sum_{\Delta=1}^{\infty} \frac{1}{\Delta^{F-1}}$, which in turn is upper bounded by $3\|\widehat{X}\|_2^2$ for $F \geq 4$, a condition guaranteed by Definition 1.2.2. We can therefore upper bound (A.20) by $\|\widehat{X}\|_2^2(3(1+3\delta)+3(3\delta^2+\delta))$, which is further upper bounded by $6\|\widehat{X}\|_2^2$ for $\delta \leq \frac{1}{20}$, as is assumed in Definition 1.2.2.

For the lower bound, we sum the bound in the first part of the lemma over all j, yielding,

$$\sum_{r\in[2k_1]}\|\widehat{Z}^r\|^2 \geq \frac{\sum_{r\in[2k_1]}|\widehat{Z}^r_j|^2}{2k_1} \geq (1-\delta)\|\widehat{X}\|_2^2 - 3\delta\cdot\left(3\|\widehat{X}\|_2^2 + \delta\sum_{j\in[\frac{n}{k_1}]}\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}\right).$$

We showed above that the double summation is upper bounded by $3\|\widehat{X}\|_2^2$, yielding an lower bound of $(1-\delta-9\delta-3\delta^2)\|\widehat{X}\|_2^2$. This is further lower bounded by $(1-12\delta)\|\widehat{X}\|_2^2$ for $\delta\leq\frac{1}{20}$. □

A.2 Properties of Active Frequencies

Proof of Lemma 1.3.2: Note that for any $j\in\left[\frac{n}{k_1}\right]$, solving the first part of Lemma 1.2.3 for $\|\widehat{X}_{I_j}\|_2^2$ gives

$$\|\widehat{X}_{I_j}\|_2^2 \leq \frac{1}{1-\delta}\left(\frac{1}{2k_1}\sum_{r\in[2k_1]}|\widehat{Z}^r_j|^2 + 3\delta\cdot\left(\|\widehat{X}_{I_j\cup I_{j-1}\cup I_{j+1}}\|_2^2 + \delta\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}\right)\right). \tag{A.21}$$

We will sum both sides over $j\in S^*\setminus\tilde{S}$; we proceed by analyzing the resulting terms.

Second term in (A.21) **summed over** $j\in S^*\setminus\tilde{S}$**:** We have

$$\begin{aligned}
&\sum_{j\in S^*\setminus\tilde{S}} 3\delta\cdot\left(\|\widehat{X}_{I_j\cup I_{j-1}\cup I_{j+1}}\|_2^2 + \delta\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}\right)\\
&\leq 9\delta\|\widehat{X}\|_2^2 + 3\delta^2\sum_{j\in S^*\setminus\tilde{S}}\sum_{j'\in[\frac{n}{k_1}]\setminus\{j\}}\frac{\|\widehat{X}_{I_{j'}}\|_2^2}{|j'-j|^{F-1}}\\
&\leq 9\delta\|\widehat{X}\|_2^2 + 10\delta^2\|\widehat{X}\|_2^2 \leq 10\delta\|\widehat{X}\|_2^2,
\end{aligned} \tag{A.22}$$

where the last line follows by expanding the double summation to all $j,j'\in\left[\frac{n}{k_1}\right]$ with $j\neq j'$, noting that $2\sum_{\Delta=1}^{\infty}\frac{1}{\Delta^{F-1}}\leq 2.5$ for $F\geq 4$ (a condition guaranteed by Definition 1.2.2), and then applying the assumption $\delta\leq\frac{1}{20}$.

First term in (A.21) **summed over** $j\in S^*\setminus\tilde{S}$**:** We first rewrite the sum of squares in terms of a weighted sum of fourth moments:

$$\begin{aligned}
\sum_{j\in S^*\setminus\tilde{S}}\frac{1}{2k_1}\sum_{r\in[2k_1]}|\widehat{Z}^r_j|^2 &= \frac{1}{2k_1}\sum_{r\in[2k_1]}\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2 = \sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2\cdot\frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2}\\
&\leq \frac{1}{2k_1}\sqrt{\left(\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2\right)\left(\sum_{r\in[2k_1]}\frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2}\right)},
\end{aligned} \tag{A.23}$$

by Cauchy-Schwarz applied to the length-$2k_1$ vectors containing entries $\|\widehat{Z}^r\|_2$ and $\frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2}$.

The second summation inside the square root is upper bounded as

$$\sum_{r\in[2k_1]} \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2} \le \sum_{r\in[2k_1]} \|\widehat{Z}^r_{S^*}\|_2^2 \cdot \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2}$$
$$\le \sum_{r\in[2k_1]} \gamma^r \cdot \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2} + \sum_{r\in[2k_1]} \Big|\|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r\Big|_+ \cdot \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2}, \tag{A.24}$$

where the first inequality follows since $\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2 \le \|\widehat{Z}^r_{S^*}\|_2^2$ and the second inequality uses $\|\widehat{Z}^r_{S^*}\|_2^2 \le \gamma^r + \big|\|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r\big|_+$.

Now observe that by definition of $\tilde{S}$ (Definition 1.3.1), for every $j \notin \tilde{S}$, we have

$$\sum_{r\in[2k_1]} \Big(|\widehat{Z}^r_j|^2 \cdot \frac{\gamma^r}{\|\widehat{Z}^r\|_2^2}\Big) \le \delta \cdot \frac{\sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2}{k_0},$$

and summing both sides over all $j \in S^*\setminus\tilde{S}$ gives

$$\sum_{r\in[2k_1]} \gamma^r \cdot \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2} \le \frac{\delta|S^*\setminus\tilde{S}|}{k_0} \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 \le 10\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2,$$

since $|S^*| \le 10k_0$ by assumption. Applying this to the first term in (A.24), as well as $\frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^2}{\|\widehat{Z}^r\|_2^2} \le 1$ for the second term, we obtain

$$\sum_{r\in[2k_1]} \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2} \le 10\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 + \sum_{r\in[2k_1]} \Big|\|\widehat{Z}^r_{S^*}\|_2^2 - \gamma^r\Big|_+$$
$$\le 50\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2, \tag{A.25}$$

where we have applied the assumption (*) of the lemma.

Finally, substituting (A.25) into (A.23) yields

$$\sum_{j\in S^*\setminus\tilde{S}} \frac{1}{2k_1} \sum_{r\in[2k_1]} |\widehat{Z}^r_j|^2 \le \frac{1}{2k_1}\sqrt{\Big(\sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2\Big)\Big(\sum_{r\in[2k_1]} \frac{\|\widehat{Z}^r_{S^*\setminus\tilde{S}}\|_2^4}{\|\widehat{Z}^r\|_2^2}\Big)}$$
$$\le \frac{1}{2k_1}\sqrt{\Big(\sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2\Big)\Big(50\delta \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2\Big)} \quad \text{(by (A.25))}$$
$$\le \frac{\sqrt{50\delta}}{2k_1} \sum_{r\in[2k_1]} \|\widehat{Z}^r\|_2^2 \le 43\sqrt{\delta}\|\widehat{X}\|_2^2, \tag{A.26}$$

where the last inequality uses the fact that $\frac{\sum_{r\in[2k_1]}\|\widehat{Z}^r\|_2^2}{2k_1} \le 6\|X\|_2^2$ by the second part of Lemma 1.2.3. The proof is concluded by substituting (A.22) and (A.26) into (A.21), and using the assumption $\delta \le \frac{1}{20}$ to deduce that $\frac{1}{1-\delta}\big(43\sqrt{\delta}+10\delta\big) \le 100\sqrt{\delta}$. □

A.3 Hashing the Fourier Domain via Filtering and Subsampling

Proof of Lemma 1.4.2: The (exact) Fourier transform of U, denoted by $\widehat{U}^*$, is given by

$$\begin{aligned}\widehat{U}^*_j &= \frac{1}{B}\sum_{b\in[B]} U_b \omega_B^{-bj} \\ &= \frac{1}{n}\sum_{b\in[B]}\sum_{i'\in[\frac{n}{B}]} X_{\sigma(\Delta+b+B\cdot i')} G_{b+B\cdot i'} \omega_B^{-bj} \\ &= \frac{1}{n}\sum_{i\in[n]} X_{\sigma(\Delta+i)} G_i \omega_n^{-ijn/B}, \end{aligned} \tag{A.27}$$

where we used the fact that $\omega_B^{(\cdot)}$ is periodic with period B, and then applied $\omega_B = \omega_n^{n/B}$. We see that (A.27) is the Fourier transform of the signal $\{X_{\sigma(\Delta+i)}G_i\}_{i\in[n]}$ evaluated at frequency jn/B, and hence, since multiplication and convolution are dual under the Fourier transform, we obtain

$$\widehat{U}^*_j = (\widehat{Y} \star \widehat{G})_{jn/B}, \tag{A.28}$$

where $Y_i = X_{\sigma(\Delta+i)}$. Now, by standard Fourier transform properties, we have $\widehat{Y}_f = \widehat{X}_{\sigma^{-1}f}\omega_n^{\Delta f}$, and substitution into (A.28) gives

$$\begin{aligned}\widehat{U}^*_j &= \sum_{f\in[n]} \widehat{X}_{\sigma^{-1}f}\widehat{G}_{j\frac{n}{B}-f}\omega_n^{\Delta f} \\ &= \sum_{f\in[n]} \widehat{X}_f \widehat{G}_{\sigma f - \frac{n}{B}j}\omega_n^{\sigma\Delta f},\end{aligned}$$

where we have used the assumed symmetry of G about zero. □

A.3.1 Proof of Lemma 1.4.5

We use techniques resembling those used for a (k_1,ϵ)-downsampling in Section 1.2, but with the notable difference of using a more rapidly-decaying filter with bounded support in *frequency* domain.

Choice of filter: We let $G\in\mathbb{R}^n$ be the filter used in (Indyk et al., 2014) (as opposed to that used in Definition 1.2.1), satisfying the following:

- There exists an ideal filter G' satisfying $G'_f \in [0,1]$ for all f, and

$$G'_f = \begin{cases} 1 & |f| \le \frac{n}{2k_1} \\ 0 & |f| \ge \frac{n}{k_1}, \end{cases} \tag{A.29}$$

  such that $\|G - G'\|_2 \le n^{-c}$;

- $\widehat{G}$ is supported on a window of length $O(ck_1\log n)$ centered at zero;

- Each entry of $\widehat{G}$ can be computed in time $O(1)$.

Intuition behind the proof. Before giving the details, we provide the intuition for the proof. Recall that our goal is to compute X_j^r for $|j| \le k_0/2$, for all $r \in [2k_1]$. To do this, we first note that X_j^0, for $|j| \le k_0/2$ (i.e., for only one value of r, namely 0), can be computed via a reduction to standard semi-equispaced FFT (Lemma 1.4.4) on an input signal of length $2n/k_1$. To achieve this, consider the signal $X \cdot G$ aliased to length $2n/k_1$, which is close to X on all points j such that $|j| \le n/(2k_1)$. In order to compute X_j^0 for $|j| \le k_0/2$, it essentially suffices (modulo boundary issues; see below) to calculate $(X \cdot G)_j$ for $|j| \le k_0/2$. We show below that this can be achieved using Lemma 1.4.4, because multiplication followed by aliasing are dual to convolution and subsampling: the input (k_0, k_1)-block sparse signal of length n can be naturally mapped to an $O(k_0 \log n)$-sparse signal in a reduced space with $\approx n/k_1$ points, in which standard techniques (Lemma 1.4.4) can be applied.

This intuition only shows how to compute the values of X_j^r for $r = 0$ and $|j| \le k_0/2$, but we need the values for all $r \in [2k_1]$. As we show below, the regular structure of the set of shifts that we are interested in allows us use the standard FFT on a suitably defined set of length-$2k_1$ signals, without increasing the runtime by a k_1 factor. It is interesting to note that our runtime is $O(\log n)$ worse than the runtime of Lemma 1.4.4 due to the two-level nature of our scheme; this is for reasons similar to the $\log^d n$ scaling of runtime of high-dimensional semi-equispaced FFT, e.g. (Ghazi et al., 2013; Kapralov, 2016).

We now give the formal proof of the lemma.

Computing a convolved signal: Here we show that we can efficiently compute the values $\widehat{Y}_j^r = (\widehat{X}^r \star \widehat{G})_{\frac{k_1}{2} j}$ at all $j \in \left[\frac{2n}{k_1}\right]$ where it is non-zero, for all values of $r \in [2k_1]$. We will later show that applying Lemma 1.4.4 to these signals (as a function of j) gives accurate estimates of the desired values of X.

Note that in the definition of $\widehat{Y}_j^r$, each non-zero block is convolved with a filter of support $O(ck_1 \log n)$, and so contributes to at most $O(c \log n)$ values of j. Since there are k_0 non-zero blocks, there are $O(ck_0 \log n)$ values of j for which the result is non-zero.

The procedure is as follows:

1. For all j such that $\widehat{Y}_j^r$ may be non-zero ($O(ck_0 \log n)$ in total), compute,

$$\widetilde{Y}_j^b = \frac{k_1}{2} \sum_{l=1}^{\frac{n}{2k_1}} \widehat{X}_{b+2k_1 l} \widehat{G}_{\frac{k_1}{2} j - (b+2k_1 l)}$$

for $b \in [2k_1]$. That is, alias the signal $\left\{\widehat{X}_f \widehat{G}_{\frac{k_1}{2} j - f}\right\}$ down to length $2k_1$, and normalize by $\frac{2}{k_1}$ (for later convenience). Since $\widehat{G}$ is supported on an interval of length $O(ck_1 \log n)$, this can be done in worst-case time $O(c \log n)$ per entry, for a total of $O(ck_1 \log n)$ per j

value, or $O(c^2 k_0 k_1 \log^2 n)$ overall (note that this bound is loose and one can show that the total time to compute $\widehat{Y}_j^r$'s is in fact $O(ck_0 k_1 \log n)$).

2. Compute the length-$2k_1$ inverse FFT of $\widetilde{Y}_j = (\widetilde{Y}_j^1, \ldots, \widetilde{Y}_j^{2k_1})$ to obtain $\widehat{Y}_j \in \mathbb{C}^{2k_1}$. This can be done in time $O(k_1 \log(1+k_1))$ per j value, or $O(ck_0 k_1 \log(1+k_1)\log n)$ overall.

We now show that $\widehat{Y}_j^r = \frac{k_1}{2}(X^r \star G)_{\frac{k_1}{2}j}$ for $r = 1, \ldots, 2k_1$. By the definition of the inverse Fourier transform, we have,

$$\begin{aligned}
\widehat{Y}_j^r &= \sum_{b=1}^{2k_1} \widetilde{Y}_j^b \omega_{2k_1}^{rb} \\
&= \frac{k_1}{2} \sum_{b=1}^{2k_1} \sum_{l=1}^{\frac{n}{2k_1}} \widehat{X}_{b+2k_1 l} \widehat{G}_{\frac{k_1}{2}j-(b+2k_1 l)} \omega_{2k_1}^{rb} \\
&= \frac{k_1}{2} \sum_{f=1}^{n} \widehat{X}_f \widehat{G}_{\frac{k_1}{2}j-f} \omega_{2k_1}^{rf} \\
&= \frac{k_1}{2} \sum_{f=1}^{n} \widehat{X}_f \widehat{G}_{\frac{k_1}{2}j-f} \omega_n^{rf \cdot \frac{n}{2k_1}} \\
&= \frac{k_1}{2} \sum_{f=1}^{n} \widehat{X}_f^r \widehat{G}_{\frac{k_1}{2}j-f} \quad \text{(since } X_{(\cdot)}^r = X_{(\cdot)+\frac{nr}{2k_1}} \text{ by Definition 1.2.2)},
\end{aligned}$$

where the second line is by the definition of $\widetilde{Y}^b$, the third by the periodicity of ω_{2k_1}, and the fifth since translation and phase shifting are dual under the Fourier transform. Hence, $\widehat{Y}_j^r = \frac{k_1}{2}(\widehat{X}^r \star \widehat{G})_{\frac{k_1}{2}j}$.

Applying the standard semi-equispaced FFT: For $r \in [2k_1]$, define $\widehat{Y}^r = (\widehat{Y}_1^r, \ldots, \widehat{Y}_{n/k_1}^r)$. We have already established that the support of each $\widehat{Y}^r$ is a subset of a set having size at most $k' = O(ck_0 \log n)$. We can therefore apply Lemma 1.4.4 with $\zeta = n^{-(c+1)}$ to conclude that we can evaluate Y_j^r for $|j| \le \frac{k'}{2}$ satisfying,

$$|Y_j^r - Y_j^{*r}| \le n^{-(c+1)} \|Y^r\|_2, \tag{A.30}$$

where Y^{*r} is the exact inverse Fourier transform of $\widehat{Y}^r$. Moreover, this can be done in time $O\big(k' \log \frac{n/k_1}{n^{-(c+1)}}\big) = O(c^2 k_0 \log^2 n)$ per r value, or $O(c^2 k_0 k_1 \log^2 n)$ overall.

Proof of the first part of lemma: It remains to show that the above procedure produces estimates of the desired X values of the form (1.15).

Recall that $\widehat{Y}_j^r = \frac{k_1}{2}(\widehat{X}^r \star \widehat{G})_{\frac{k_1}{2}j}$. By the convolution theorem and the fact that subsampling and aliasing are dual (e.g., see Appendix A.3), the inverse Fourier transform of $\widehat{Y}^r$ satisfies the

following when $|j| \leq \frac{n}{k_1}$,

$$\begin{aligned} Y_j^r &= \sum_{i \in [\frac{k_1}{2}]} (G \cdot X^r)_{j+\frac{2n}{k_1} i} \\ &= G_j X_j^r + \sum_{i \in [\frac{k_1}{2}], i \neq 0} (G \cdot X^r)_{j+\frac{2n}{k_1} i} \\ &= \left(G'_j X_j^r + \sum_{i \in [\frac{k_1}{2}], i \neq 0} (G' \cdot X^r)_{j+\frac{2n}{k_1} i} \right) \pm \|G - G'\|_2 \|X\|_2 \\ &= X_j^r \pm n^{-c} \|X\|_2, \end{aligned} \tag{A.31}$$

where the last line follows from the definition of G' in (A.29) and the assumption $\|G - G'\|_2 \leq n^{-c}$.

Combining (A.30) and (A.31) and using the triangle inequality, we obtain

$$|Y_j^r - X_j^r| \leq n^{-(c+1)} \|Y^r\|_2 + n^{-c} \|X\|_2.$$

Since we have already shown that we can efficiently compute Y_j^r for $|j| \leq \frac{k'}{2}$ with $k' = O(ck_0 \log n)$, it only remains to show that $\|Y^r\|_2 \leq n\|X\|^2$. To do this, we use the first line of (A.31) to write

$$\begin{aligned} |Y_j^r| &\leq \sum_{i \in [\frac{k_1}{2}]} |G_{j+\frac{2n}{k_1} i}| \cdot |X^r_{j+\frac{2n}{k_1} i}| \\ &\leq 2 \sum_{i \in [\frac{k_1}{2}]} |X^r_{j+\frac{2n}{k_1} i}| \\ &\leq \sqrt{2k_1 \sum_{i \in [\frac{k_1}{2}]} |X^r_{j+\frac{2n}{k_1} i}|^2}, \end{aligned} \tag{A.32}$$

where the first line is the triangle inequality, the second line follows since the first filter assumption above ensures that $|G_j| \leq 2$ for all j, and the third line follows since the squared ℓ_1-norm is upper bounded by the squared ℓ_2-norm times the vector length.

Squaring both sides of (A.32) and summing over all j gives $\|Y^r\|_2^2 \leq 2k_1 \|X\|^2 \leq n^2 \|X\|^2$ (under the trivial assumption $n \geq 2$), thus completing the proof.

Proof of the second part of lemma: In the proof of the first part, we applied Lemma 1.4.4 to signals of length $\frac{2n}{k_1}$. It follows directly from the arguments in (Indyk et al., 2014, Cor. 12.2) that since we can approximate the entries of X_j^r for all $|j| \leq \frac{k_0}{2}$, we can do the same for all j equaling $\sigma j' + b$ modulo-$\frac{2n}{k_1}$ for some $|j'| \leq \frac{k_0}{2}$. Specifically, this follows since the multiplication by σ and shift by b simply amounts to a phase shift and a linear change of variables $f \to \sigma^{-1} f$ in frequency domain, both of which can be done in constant time.

However, the second part of the lemma regards indices modulo-$\frac{n}{k_1}$, as opposed to modulo-$\frac{2n}{k_1}$.

To handle the former, we note that for any integer a, we either have $a \bmod \frac{n}{k_1} = a \bmod \frac{2n}{k_1}$ or $a \bmod \frac{n}{k_1} = \left(a + \frac{n}{k_1}\right) \bmod \frac{2n}{k_1}$. Hence, we obtain the desired result by simply performing two calls to the first part, one with a universal shift of $\frac{n}{k_1}$.

A.3.2 Proof of Lemma 1.4.6

First Part: Since U_X is computed according to X itself in Algorithm 4, we only need to compute the error in U_χ.

In the definition of hashing in Definition 1.4.2, since G has support $O(FB)$, we see that the values of X used correspond to a permutation of an interval having length $k' = O(FB)$. We can therefore apply the second part of Lemma 1.4.4 with sparsity k' and parameter $\zeta = n^{-c'}$ for some $c' > 0$, ensuring an ℓ_∞-guarantee of $n^{-c'}\|\chi\|_2$ for the signal χ.

Since $\widehat{U}$ is computed from these values using (2.2) followed by the FFT, we readily obtain via the relation $\|v\|_\infty \le \|v\|_2 \le \sqrt{m}\|v\|_\infty$ (for $v \in \mathbb{C}^m$) and Parseval's theorem that $\widehat{U}$ has an ℓ_∞-guarantee of $n^{-(c'-O(1))}\|\chi\|_2$, which can be made to equal $n^{-c}\|\widehat{\chi}\|_2$ by choosing $c' = c + O(1)$.

Sample complexity and runtime: The only operation that consumes samples from the signal X is the hashing operation applied to X. From Definition 1.4.2, and the fact that the filter G has support $O(FB)$, we find that the sample complexity is also $O(FB)$.

The runtime is dominated by the application of the semi-equispaced FFT, which, by Lemma 1.4.4, uses $O(cF(\|\widehat{\chi}\|_0 + B)\log n)$ operations. In particular, this dominates the $O(B\log B)$ time to perform the FFT in Algorithm 4, and the hashing operation, whose time complexity is the same as the sample complexity.

Second Part: Recall the definition of a (k_1, δ)-downsampling of a signal X from (1.2):

$$Z_j^r = \frac{1}{k_1}\sum_{i\in[k_1]} (G\cdot X^r)_{j+\frac{n}{k_1}\cdot i}, \quad j \in \left[\frac{n}{k_1}\right], r \in [2k_1].$$

In order to compute the $\left(\frac{n}{k_1}, B^r, G^r, \sigma, \Delta\right)$-hashing of $\widehat{Z}^r$ (*cf.*, Definition 1.4.2), we use the samples of Z_j^r at the locations $j = \sigma(j' + \Delta) \mod \frac{n}{k_1}$ for $|j'| \le FB^r$; this is because G^r is supported on $[-FB^r, +FB^r]$. Note that FB^r is further upper bounded by $O(FB_{\max})$.

We claim that in the second part of Lemma 1.4.5, it suffices to set the sparsity level to $O(FB_{\max} + k_0)$. To see this, first note that k_0 is added in accordance with Remark 1.4.2 and the fact that $\widehat{\chi}$ is (k_0, k_1)-block sparse. Moreover, note that $\widehat{Z}^r$ has length $\frac{n}{k_1}$, and one sample of Z_j^r can be computed from $X^r_{j+i\frac{n}{k_1}} = X_j^{r+2i}$ for $|i| \le F$ as per Definition 1.2.2 and the fact that the filter G is supported on $[-F\frac{n}{k_1}, +F\frac{n}{k_1}]$. Therefore, all we need is $X_{j'}^{r'}$ for each $r' \in [2k_1]$ and for all $j' = \sigma(j + \Delta) \mod \frac{n}{k_1}$ with $|j| \le FB_{\max}$.

Applying the second part of Lemma 1.4.5 with sparsity $O(FB_{\max}+k_0)$ and parameter $\zeta = n^{-c'}$ for some $c'>0$, ensuring an ℓ_∞-guarantee of $2n^{-c'}\|\chi\|_2$ on the computed values of χ. By an analogous argument to the first case, this implies an ℓ_∞-guarantee of $n^{-c}\|\chi\|_2$ on the FFT $\widehat{U}^r$ of the hashing of Z^r_χ, with $c = c' + O(1)$.

Sample complexity and runtime: We take $O(FB^r)$ samples of the r-th downsampled signal each time we do the hashing, separately for each $r \in [2k_1]$. By Lemma 1.2.2, accessing a single sample of Z^r_X costs us $O(\log\frac{1}{\delta})$ samples of X. Hence, the sample complexity is $O\big(F\sum_{r\in[2k_1]} B^r \log\frac{1}{\delta}\big)$.

We now turn to the runtime. By Lemma 1.4.5, the call to SEMIEQUIINVERSEBLOCKFFT with $O(FB_{\max}+k_0)$ in place of k_0 takes time $O\big(c^2(FB_{\max}+k_0)k_1\log^2 n\big)$. The hashing operation's runtime matches its sample complexity, and since we have assumed $\delta \geq \frac{1}{n}$, its contribution is dominated by the preceding term.

A.3.3 Proof of Lemma 1.4.7

First part of lemma: We start with the following upper bound on the expression inside the expectation:

$$\Big|\|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*\|_2^2\Big|_+ \leq \Big|\|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*_{h(S\setminus S_{coll})}\|_2^2\Big|_+$$

where $h(S) = \{h(j) : j \in S\}$ with $h(j) = \text{round}\big(\pi(j)\frac{B}{m}\big)$, denoting the bucket into which element j hashes. We define S_{coll} to be a subset of S containing the elements that collide with each other, i.e., $S_{\text{coll}} = \{j \in S \,|\, h(j) \cap h(S\setminus\{j\}) \neq \emptyset\}$, yielding

$$\begin{aligned}
\Big|\|\widehat{Y}_S\|_2^2 - \|\widehat{U}^*\|_2^2\Big|_+ &\leq \Big|\sum_{j\in S}|\widehat{Y}_j|^2 - \sum_{b\in h(S\setminus S_{\text{coll}})}|\widehat{U}^*_b|^2\Big|_+ \\
&= \Big|\sum_{j\in S_{\text{coll}}}|\widehat{Y}_j|^2 + \sum_{j\in S\setminus S_{\text{coll}}}\Big(|\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2\Big)\Big|_+ \\
&\leq \sum_{j\in S_{\text{coll}}}|\widehat{Y}_j|^2 + \sum_{j\in S}\Big||\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2\Big|_+, \qquad \text{(A.33)}
\end{aligned}$$

where the final line follows from the inequality $[a+b]_+ \leq |a| + [b]_+$.

Bounding the first term in (A.33)**:** We start by evaluating the expected value of the term corresponding to S_{coll} over the random permutation π:

$$\begin{aligned}
\mathbb{E}_\pi\left[\sum_{j\in S_{\text{coll}}}|\widehat{Y}_j|^2\right] &\leq \mathbb{E}_\pi\left[\sum_{j\in S}|\widehat{Y}_j|^2\cdot\mathbb{1}\left[j\in S_{\text{coll}}\right]\right] \\
&\leq \sum_{j\in S}|\widehat{Y}_j|^2\sum_{j'\in S\setminus\{j\}}\Pr[h(j)=h(j')] \\
&\leq \sum_{j\in S}\sum_{j'\in S}|\widehat{Y}_j|^2\frac{4}{B} = \frac{4|S|}{B}\sum_{j\in S}|\widehat{Y}_j|^2, \qquad \text{(A.34)}
\end{aligned}$$

where the second line follows from the union bound, and the third line follows since π is

approximately pairwise independent as per Definition 1.4.1.

Bounding the second term in (A.33)**:** We apply Lemma 1.4.2 to obtain $\widehat{U}^*_{h(j)} = \sum_{j' \in [m]} \widehat{Y}_{j'} \widehat{H}_{o_j(j')} \omega_m^{\sigma \Delta j'}$, where $o_j(j') = \pi(j') - h(j)\frac{m}{B}$. We write this as $\widehat{U}^*_{h(j)} = \widehat{Y}_j \widehat{H}_{o_j(j)} \omega_m^{\sigma \Delta j} + \mathrm{err}_j$, where $\mathrm{err}_j := \sum_{j' \in [m] \setminus \{j\}} \widehat{Y}_{j'} \widehat{H}_{o_j(j')} \omega_m^{\sigma \Delta j'}$, yielding,

$$\begin{aligned}\sum_{j \in S} \left| |\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2 \right|_+ &\leq \sum_{j \in S} \left| |\widehat{Y}_j|^2 - |\widehat{Y}_j \widehat{H}_{o_j(j)} \omega_m^{\sigma \Delta j} + \mathrm{err}_j|^2 \right| \\ &\leq \sum_{j \in S} \left(\left| |\widehat{Y}_j|^2 - |\widehat{Y}_j H_{o_j(j)}|^2 \right| + |\mathrm{err}_j|^2 + 2|\mathrm{err}_j| \cdot |\widehat{Y}_j \widehat{H}_{o_j(j)}| \right) \end{aligned} \tag{A.35}$$

by $|\xi|_+ \leq |\xi|$ and the triangle inequality. We have by definition that $|o_j(j)| \leq \frac{m}{2B}$, and hence item 2 in Definition 1.2.1 yields $H_{o_j(j)} \geq 1 - \left(\frac{1}{4}\right)^{F'-1}$, which in turn implies $H^2_{o_j(j)} \geq 1 - 2\left(\frac{1}{4}\right)^{F'-1}$. Combining this with $H_f \leq 1$ from item 1 in Definition 1.2.1, we can weaken (A.35) to,

$$\sum_{j \in S} \left| |\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2 \right|_+ \leq \sum_{j \in S} \left(2|\mathrm{err}_j| \cdot |\widehat{Y}_j| + |\mathrm{err}_j|^2 + 2\left(\frac{1}{4}\right)^{F'-1} |\widehat{Y}_j|^2 \right). \tag{A.36}$$

We proceed by bounding the expected value of $|\mathrm{err}_j|^2$. We first take the expectation over Δ, using Parseval's theorem to write,

$$\mathbb{E}_\Delta[|\mathrm{err}_j|^2] = \sum_{j' \in [m] \setminus \{j\}} |\widehat{Y}_{j'}|^2 |\widehat{H}_{o_j(j')}|^2.$$

Taking the expectation over π, we obtain

$$\begin{aligned}\mathbb{E}_{\Delta,\pi}[|\mathrm{err}_j|^2] &= \mathbb{E}_\pi \Big[\sum_{j' \in [m] \setminus \{j\}} |\widehat{Y}_{j'}|^2 |\widehat{H}_{o_j(j')}|^2 \Big] \\ &= \sum_{j' \in [m] \setminus \{j\}} |\widehat{Y}_{j'}|^2 \cdot \mathbb{E}_\pi[|\widehat{H}_{o_j(j')}|^2] \\ &\leq \frac{10}{B} \sum_{j' \in [m] \setminus \{j\}} |\widehat{Y}_{j'}|^2 \leq \frac{10}{B} \|\widehat{Y}\|_2^2. \end{aligned}$$

where the final line follows from Lemma 1.4.3. Substituting into (A.36), and using Jensen's inequality to write $\mathbb{E}[|\mathrm{err}_j|] \leq \sqrt{\mathbb{E}[|\mathrm{err}_j|^2]}$, we obtain,

$$\begin{aligned}\mathbb{E}_{\Delta,\pi}\Big[\sum_{j \in S} \left| |\widehat{Y}_j|^2 - |\widehat{U}^*_{h(j)}|^2 \right|_+ \Big] &\leq 2 \sum_{j \in S} \sqrt{\frac{10}{B} \|\widehat{Y}\|_2^2} \cdot |\widehat{Y}_j| + \frac{10}{B} \sum_{j \in S} \|\widehat{Y}\|_2^2 + 2\left(\frac{1}{4}\right)^{F'-1} \sum_{j \in S} |\widehat{Y}_j|^2 \\ &\leq 10\sqrt{\frac{|S|}{B}} \|\widehat{Y}\|_2^2 + \left(10\frac{|S|}{B} + 2\delta^2 \right) \|\widehat{Y}\|_2^2, \end{aligned} \tag{A.37}$$

where the second line follows from the fact that $\|v\|_1 \leq \sqrt{|S|}\|v\|_2$ for any $v \in \mathbb{C}^{|S|}$, as well as $\left(\frac{1}{4}\right)^{F'-1} \leq \delta^2$ by the choice of F'. The claim follows by substituting (A.34) and (A.37) into (A.33).

Algorithm 16 Prune a location list via hashing and thresholding techniques.

procedure PRUNELOCATION($X, \widehat{\chi}, L, n, k_0, k_1, \delta, p, \theta$)
  $B \leftarrow 160\frac{k_0 k_1}{\delta}$
  $F \leftarrow 10\log\frac{1}{\delta}$
  $G \leftarrow (n, B, F)$-flat filter
  $T \leftarrow 10\log\frac{1}{\delta p}$
  for $t \in \{1, \ldots, T\}$ **do**
    $\Delta \leftarrow$ uniform random sample from $[n]$
    $\sigma \leftarrow$ uniform random sample from odd numbers in $[n]$
    $\widehat{U} \leftarrow$ HASHTOBINS($X, \widehat{\chi}, G, n, B, \sigma, \Delta$)
    $W_j^{(t)} \leftarrow \sum_{f \in I_j} \left|\widehat{G}^{-1}_{o_f(f)} \widehat{U}_{h(f)} \omega_n^{-\sigma\Delta f}\right|^2$ for all $j \in L$ ▷ $h(f), o_f(f)$ in Definition 1.4.2
  $W_j \leftarrow \text{Median}_t\left\{W_j^{(t)}\right\}$ for all $j \in L$
  $L' \leftarrow \left\{j \in L : W_j \geq \theta\right\}$
  return L'

Second part of lemma: By the definition of $\widehat{U}^*$ (*cf.*, Definition 1.4.2), we have,

$$\begin{aligned}\mathbb{E}_\Delta\left[\|\widehat{U}^*\|_2^2\right] &= \mathbb{E}_\Delta\Big[\sum_{b\in[B]}\Big|\sum_{j\in[m]}\widehat{Y}_j\widehat{H}_{\pi(j)-b\frac{m}{B}}\omega_m^{\Delta j}\Big|^2\Big] \\ &= \sum_{b\in[B]}\sum_{j\in[m]}|\widehat{Y}_j|^2|\widehat{H}_{\pi(j)-b\frac{m}{B}}|^2,\end{aligned}$$

by Parseval. Taking the expectation with respect to π, we obtain,

$$\begin{aligned}\mathbb{E}_{\Delta,\pi}\left[\|\widehat{U}^*\|_2^2\right] &= \sum_{b\in[B]}\mathbb{E}_\pi\left[\sum_{j\in[m]}|\widehat{Y}_j|^2|\widehat{H}_{\pi(j)-b\frac{m}{B}}|^2\right] \\ &= \sum_{b\in[B]}\sum_{j\in[m]}|\widehat{Y}_j|^2\cdot\mathbb{E}_\pi\left[|\widehat{H}_{\pi(j)-b\frac{m}{B}}|^2\right] \\ &\leq \sum_{b\in[B]}\frac{3}{B}\sum_{j\in[m]}|\widehat{Y}_j|^2 = 3\|\widehat{Y}\|_2^2,\end{aligned}$$

where the final line follows by noting that $\pi(j) - b\frac{m}{B}$ is uniformly distributed over $[m]$, and applying the second part of Lemma 1.2.1.

A.4 Pruning the Location List

The pruning procedure is given in Algorithm 16. Its goal is essentially to reduce the size of the list returned by MULTIBLOCKLOCATE (*cf.*, Algorithm 1) from $O(k_0\log(1+k_0))$ to $O(k_0)$. More formally, the following lemma shows that with high probability, the pruning algorithm retains most of the energy in the head elements, while removing most tail elements.

Lemma 1.5.1 (PRUNELOCATION guarantees – restated from Section 1.5.1)**.** *Given integers n, k_0, k_1, a list of block indices $L \subseteq \left[\frac{n}{k_1}\right]$, parameters $\theta > 0$, $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$ and $p \in (0,1)$, and signals*

$X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X}-\widehat{\chi}\|_2 \geq \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$, the output L' of PRUNELOCATION$(X, \widehat{\chi}, L, n, k_0, k_1, \delta, p, \theta)$ *has the following properties:*

a. *Let S_{tail} denote the tail elements in signal $\widehat{X}-\widehat{\chi}$, defined as,*

$$S_{\mathrm{tail}} = \left\{ j \in \left[\frac{n}{k_1}\right] : \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2 \leq \sqrt{\theta} - \sqrt{\frac{\delta}{k_0}}\|\widehat{X}-\widehat{\chi}\|_2 \right\},$$

where I_j is defined in Definition 1.1.1. Then, we have,

$$\mathbb{E}\left[\left|L' \cap S_{\mathrm{tail}}\right|\right] \leq \delta p \cdot |L \cap S_{\mathrm{tail}}|.$$

b. *Let S_{head} denote the head elements in signal $\widehat{X}-\widehat{\chi}$, defined as,*

$$S_{\mathrm{head}} = \left\{ j \in \left[\frac{n}{k_1}\right] : \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2 \geq \sqrt{\theta} + \sqrt{\frac{\delta}{k_0}}\|\widehat{X}-\widehat{\chi}\|_2 \right\}.$$

Then, we have,

$$\mathbb{E}\left[\sum_{j \in (L \cap S_{\mathrm{head}}) \setminus L'} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2\right] \leq \delta p \sum_{j \in L \cap S_{\mathrm{head}}} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2.$$

Moreover, provided that $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, the sample complexity is $O\left(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta p}\log\frac{1}{\delta}\right)$, and the runtime is $O\left(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta p}\log\frac{1}{\delta}\log n + k_1 \cdot |L|\log\frac{1}{\delta p}\right)$.

Proof. We begin by analyzing the properties of the random variables $W_j^{(t)}$ used in the threshold test. We define $X' = X - \chi$, let $\widehat{U}$ be the output of HASHTOBINS, and let $\widehat{U}^*$ be its exact counterpart as defined in Lemma 1.4.6. It follows that we can write the random variable $W_j^{(t)}$ (*cf.*, Algorithm 16) as

$$\begin{aligned} W_j^{(t)} &= \sum_{f \in I_j} \left| \widehat{G}^{-1}_{o_f(f)} \widehat{U}_{h(f)} \omega_n^{-\sigma\Delta f} \right|^2 \\ &= \sum_{f \in I_j} \left| \widehat{G}^{-1}_{o_f(f)} \widehat{U}^*_{h(f)} \omega_n^{-\sigma\Delta f} + \widehat{G}^{-1}_{o_f(f)} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega_n^{-\sigma\Delta f} \right|^2 \\ &= \sum_{f \in I_j} \left| \widehat{X}'_f + \mathrm{err}_f^{(t)} + \widetilde{\mathrm{err}}_f^{(t)} \right|^2, \end{aligned} \tag{A.38}$$

where (i) $\mathrm{err}_f^{(t)} = \widehat{G}^{-1}_{o_f(f)} \sum_{f' \in [n]\setminus\{f\}} \widehat{X}'_{f'} \widehat{G}_{o_f(f')} \omega_n^{\sigma\Delta(f'-f)}$, with (σ, Δ) implicitly depending on t; this follows directly from Lemma 1.4.2, along with the definitions $\pi(f) = \sigma f$ and $o_f(f') = \pi(f') - \frac{n}{B}h(f)$. (ii) $\widetilde{\mathrm{err}}_f^{(t)} = \widehat{G}^{-1}_{o_f(f)} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega^{-\sigma\Delta f}$, a polynomially small error term (*cf.*, Lemma 1.4.6).

Bounding $\mathrm{err}_f^{(t)}$ and $\widetilde{\mathrm{err}}_f^{(t)}$: In Lemma A.4.1 below, we show that,

$$\mathbb{E}_{\Delta,\pi}\left[\left|\mathrm{err}_f^{(t)}\right|^2\right] \le \frac{20}{B}\|\widehat{X}'\|_2^2 \tag{A.39}$$

$$\left|\widetilde{\mathrm{err}}_f^{(t)}\right| \le 2n^{-c+c'}\|\widehat{X}'\|_2, \tag{A.40}$$

where c is used in HASHTOBINS, and c' is a value such that $\|\widehat{X}-\widehat{\chi}\|_2 \ge \frac{1}{n^{c'}}\|\widehat{\chi}\|_2$. For (A.40), we upper bound the ℓ_2 norm by the square root of the vector length times the ℓ_∞ norm, yielding

$$\sqrt{\sum_{f\in[n]}\left|\widetilde{\mathrm{err}}_f^{(t)}\right|^2} \le \sqrt{n}\max_{f\in[n]}\left|\widetilde{\mathrm{err}}_f^{(t)}\right| \le 2n^{-c+c'+1/2}\|\widehat{X}'\|_2. \tag{A.41}$$

We now calculate the probability of a given block j passing the threshold test, considering two separate cases.

If j is in the tail: The probability for j to pass the threshold is closely related to $\Pr\left[W_j^{(t)} \ge \theta\right] = \Pr\left[\sqrt{W_j^{(t)}} \ge \sqrt{\theta}\right]$. From (A.38), $\sqrt{W_j^{(t)}}$ is the ℓ_2-norm of a sum of three signals, and hence we can apply the triangle inequality to obtain

$$\Pr\left[W_j^{(t)} \ge \theta\right] \le \Pr\left[\sum_{f\in I_j}\left|\mathrm{err}_f^{(t)}\right|^2 \ge \left(\sqrt{\theta} - \sqrt{\sum_{f\in I_j}|\widehat{X}'_f|^2} - 2n^{-c+c'+1/2}\|\widehat{X}'\|_2\right)^2\right],$$

where we have applied (A.41). By definition, for any $j \in S_{\text{tail}}$, we have $\sqrt{\theta} - \|\widehat{X}'_{I_j}\|_2 \ge \sqrt{\frac{\delta}{k_0}}\|\widehat{X}'\|_2$. Hence, and recalling that $\delta \ge \frac{1}{n}$, if c if sufficiently large so that $\sqrt{\frac{\delta}{k_0}} - 2n^{-c+c'+1/2} \ge \sqrt{\frac{0.9\delta}{k_0}}$, then Markov's inequality yields,

$$\begin{aligned}
\Pr\left[W_j^{(t)} \ge \theta\right] &\le \Pr\left[\sum_{f\in I_j}\left|\mathrm{err}_f^{(t)}\right|^2 \ge \frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2\right] \\
&\le \frac{\mathbb{E}_{\Delta,\pi}\left[\sum_{f\in I_j}|\mathrm{err}_f^{(t)}|^2\right]}{\frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2} \\
&\le \frac{\frac{20k_1}{B}\|\widehat{X}'\|_2^2}{\frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2} \le \frac{1}{6}
\end{aligned}$$

where the third inequality follows form (A.39) and $|I_j| = k_1$, and the final inequality follows from the choice $B = 160\frac{k_0k_1}{\delta}$. Since W_j is the median of T independent such random variables, it can only exceed θ if there exists a subset of t values of size $\frac{T}{2}$ with $W_j^{(t)} \ge \theta$. Hence,

$$\Pr\left[W_j \ge \theta\right] \le \binom{T}{T/2}\left(\frac{1}{6}\right)^{T/2} \le \left(\frac{2}{3}\right)^{T/2} \le \delta p,$$

where we applied $\binom{T}{T/2} \le 2^T$, followed by $T = 10\log\frac{1}{\delta p}$ (*cf.*, Algorithm 16).

If j is in the head: We proceed similarly to the tail case, but instead use the triangle inequality in the form of a lower bound (i.e., $\|a+b\|_2 \ge \|a\|_2 - \|b\|_2$), yielding

$$\Pr\left[W_j^{(t)} \le \theta\right] \le \Pr\left[\sum_{f\in I_j}\left|\mathrm{err}_f^{(t)}\right|^2 \ge \left(\sqrt{\sum_{f\in I_j}|\widehat{X}'_f|^2} - \sqrt{\theta} - 2n^{-c+c'+1/2}\|\widehat{X}'\|_2\right)^2\right].$$

By definition, for any $j \in S_{\text{head}}$, we have $\|\widehat{X}'_{I_j}\|_2 - \sqrt{\theta} \ge \sqrt{\frac{\delta}{k_0}}\|\widehat{X}'\|_2$. Hence, if c if sufficiently large so that $\sqrt{\frac{\delta}{k_0}} - 2n^{-c+c'+1/2} \ge \sqrt{\frac{0.9\delta}{k_0}}$, then analogously to the tail case above, we have,

$$\Pr\big[W_j^{(t)} \le \theta\big] \le \Pr\left[\sum_{f\in I_j}\left|\mathrm{err}_f^{(t)}\right|^2 \ge \frac{0.9\delta}{k_0}\|\widehat{X}'\|_2^2\right] \le \frac{1}{6},$$

and consequently $\Pr\big[W_j \le \theta\big] \le \delta p$.

First claim of the lemma: Since $L' \subset L$, we have,

$$\mathbb{E}\Big[\big|L' \cap S_{\text{tail}}\big|\Big] = \sum_{j\in L\cap S_{\text{tail}}} \Pr\big[j \in L'\big] = \sum_{j\in L\cap S_{\text{tail}}} \Pr\big[W_j \ge \theta\big].$$

Since we established that $\Pr\big[W_j \ge \theta\big]$ is at most δp, we obtain,

$$\mathbb{E}\Big[\big|L' \cap S_{\text{tail}}\big|\Big] \le \sum_{j\in L\cap S_{\text{tail}}} \delta p = \delta p \cdot |L \cap S_{\text{tail}}|.$$

Second claim of the lemma: In order to upper bound $\sum_{j\in(L\cap S_{\text{head}})\setminus L'}\|\widehat{X}'_{I_j}\|_2^2$, we first calculate its expected value as follows:

$$\begin{aligned}\mathbb{E}\left[\sum_{j\in(L\cap S_{\text{head}})\setminus L'}\|\widehat{X}'_{I_j}\|_2^2\right] &= \mathbb{E}\left[\sum_{j\in L\cap S_{\text{head}}}\|\widehat{X}'_{I_j}\|_2^2 \cdot \mathbb{1}\left[j \notin L'\right]\right] \\ &= \sum_{j\in L\cap S_{\text{head}}}\|\widehat{X}'_{I_j}\|_2^2 \cdot \Pr\big[j \notin L'\big] \\ &= \sum_{j\in L\cap S_{\text{head}}}\|\widehat{X}'_{I_j}\|_2^2 \cdot \Pr\big[W_j \le \theta\big].\end{aligned}$$

The probability $\Pr\big[W_j \le \theta\big]$ for $j \in L \cap S_{\text{head}}$ is at most δp, and hence,

$$\mathbb{E}\left[\sum_{j\in(L\cap S_{\text{head}})\setminus L'}\|\widehat{X}'_{I_j}\|_2^2\right] \le \delta p \sum_{j\in L\cap S_{\text{head}}}\|\widehat{X}'_{I_j}\|_2^2.$$

Sample complexity and runtime For the sample complexity, note that the algorithm only uses samples via its call to HashToBins. By part (i) of Lemma 1.4.6 and the choices $B = 160\frac{k_0 k_1}{\delta}$

and $F = 10\log\frac{1}{\delta}$, the sample complexity is $O(FB) = O(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta})$ per hashing operation. Since we run the hashing in a loop $10\log\frac{1}{\delta p}$ times, the sample complexity is $O(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta}\log\frac{1}{\delta p})$.

The runtime depends on three operations. The first is calling HASHTOBINS, for which an analogous argument as that for the sample complexity holds, with the extra $\log n$ factor arising from Lemma 1.4.6. The second operation is the computation of $W_j^{(t)}$, which takes $|I_j| = O(k_1)$ time for each $j \in L$. Hence, the total contribution from the loop is $O(k_1 \cdot |L|\log\frac{1}{\delta p})$. Finally, since the median can be computed in linear time, computing the medians for every $j \in L$ costs $O(|L|\log\frac{1}{\delta p})$ time, which is dominated by the computation of $W_j^{(t)}$. □

In the preceding proof, we made use of the following lemma.

Lemma A.4.1. *Fix integers n, k_0, k_1, B, signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{n^{c'}}\|\widehat{\chi}\|_2$, and uniformly random parameters $\sigma, \Delta \in [n]$ with σ odd, and let $\widehat{U}$ be the output of* HASHTOBINS *$(X, \widehat{\chi}, G, n, B, \sigma, \Delta)$ and $\widehat{U}^*$ its exact counterpart. Then, defining,*

$$\mathrm{err}_f := \widehat{G}^{-1}_{o_f(f)} \sum_{f' \in [n]\setminus\{f\}} \widehat{X}'_{f'} \widehat{G}_{o_f(f')} \omega_n^{\sigma\Delta(f'-f)} \text{ , and } \widetilde{\mathrm{err}}_f := \widehat{G}^{-1}_{o_f(f)} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega^{-\sigma\Delta f}$$

(for h and σ_f in Definition 1.4.2), we have,

$$\mathbb{E}_{\Delta,\pi}\left[\left|\mathrm{err}_f\right|^2\right] \leq \frac{20}{B}\|\widehat{X}'\|_2^2 \tag{A.42}$$

$$|\widetilde{\mathrm{err}}_f| \leq 2n^{-c+c'}\|\widehat{X}'\|_2 \tag{A.43}$$

for c used in HASHTOBINS.

Proof. We take the expectation of $|\mathrm{err}_f|^2$, first over Δ:

$$\mathbb{E}_\Delta\left[|\mathrm{err}_f|^2\right] = |\widehat{G}_{o_f(f)}|^{-2} \sum_{f' \in [n]\setminus\{f\}} |\widehat{X}'_{f'}|^2 |\widehat{G}_{o_f(f')}|^2$$

by Parseval. By Definition 1.2.1 and the definition of $o_f(\cdot)$, we can upper bound $|\widehat{G}_{o_f(f)}|^{-2} \leq 2$. Continuing, we take the expectation with respect to the random permutation π:

$$\begin{aligned}\mathbb{E}_{\Delta,\pi}\left[|\mathrm{err}_f|^2\right] &\leq \mathbb{E}_\pi\left[2\sum_{f' \in [n]\setminus\{f\}} |\widehat{X}'_{f'}|^2 |\widehat{G}_{o_f(f')}|^2\right] \\ &= 2\sum_{f' \in [n]\setminus\{f\}} |\widehat{X}'_{f'}|^2 \mathbb{E}_\pi\left[|\widehat{G}_{o_f(f')}|^2\right] \leq \frac{20}{B}\|\widehat{X}'\|_2^2. \end{aligned} \tag{A.44}$$

by Lemma 1.4.3.

We now turn to $\widetilde{\mathrm{err}}_f$. We know from Lemma 1.4.6 that $\left|\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}\right| \leq \|\widehat{U} - \widehat{U}^*\|_\infty \leq n^{-c}\|\widehat{\chi}\|_2$. Hence, and again using $|\widehat{G}_{o_f(f)}|^{-2} \leq 2$, we find that,

$$|\widetilde{\mathrm{err}}_f| \leq 2n^{-c}\|\widehat{\chi}\|_2 \leq 2n^{-c+c'}\|\widehat{X}'\|_2, \tag{A.45}$$

Algorithm 17 Estimation procedure for individual frequencies

procedure EstimateValues($X, \widehat{\chi}, L, n, k_0, k_1, \delta, p$)
  $B \leftarrow \frac{1200}{\delta} k_0 k_1$
  $F \leftarrow 10 \log \frac{1}{\delta}$
  $G \leftarrow (n, B, F)$-flat filter ▷ See Definition 1.2.1
  $\mathscr{F} \leftarrow \left\{ f \in [n] : \text{round}\left(\frac{f}{k_1}\right) \in L \right\}$
  $T \leftarrow 10 \log \frac{2}{p}$
  for $t \in \{1, \ldots, T\}$ **do**
    $\Delta \leftarrow$ uniform random sample from $[n]$
    $\sigma \leftarrow$ uniform random sample from odd numbers in $[n]$
    $\widehat{U} \leftarrow$ HashToBins$(X, \widehat{\chi}, G, n, B, \sigma, \Delta)$ ▷ $o_f(f), h(f)$ as in Definition 1.4.2
    $W_f^{(t)} \leftarrow \widehat{G}_{o_f(f)}^{-1} \widehat{U}_{h(f)} \omega^{-\sigma \Delta f}$ for every $f \in \mathscr{F}$
  $W_f \leftarrow \text{Median}_t \left\{ W_f^{(t)} \right\}$ for every $f \in \mathscr{F}$ ▷ Separately for the real and imaginary parts
  return W

where the second inequality follows since $\|\widehat{\chi}\|_2 \le n^{c'} \|\widehat{X}'\|_2$ for some $c' > 0$, by assumption. □

A.5 Estimating Individual Frequency Values

Once we have located the blocks, we need to estimate the frequency values with them. The function EstimateValues in Algorithm 17 performs this task for us via basic hashing techniques. The following lemma characterizes the guarantee on the output.

Lemma 1.5.2 (EstimateValues guarantees – restated from Section 1.5.1)**.** *For any integers n, k_0, k_1, any list of block indices $L \subseteq \left[\frac{n}{k_1}\right]$, parameters $\delta \in \left(\frac{1}{n}, \frac{1}{20}\right)$ and $p \in (0, 1/2)$, and signals $X \in \mathbb{C}^n$ and $\widehat{\chi} \in \mathbb{C}^n$ with $\|\widehat{X} - \widehat{\chi}\|_2 \ge \frac{1}{\text{poly}(n)} \|\widehat{\chi}\|_2$, with probability at least $1 - p$, the output W of the function* EstimateValues *$(X, \widehat{\chi}, L, n, k_0, k_1, \delta, p)$ (Algorithm 17) has the following property:*

$$\sum_{f \in \bigcup_{j \in L} I_j} |W_f - (\widehat{X} - \widehat{\chi})_f|^2 \le \delta \frac{|L|}{3k_0} \|\widehat{X} - \widehat{\chi}\|_2^2,$$

where I_j is the j-th block. Moreover, provided that $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, the sample complexity is $O\left(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta}\right)$, and the runtime is $O\left(\frac{k_0 k_1}{\delta} \log \frac{1}{p} \log \frac{1}{\delta} \log n + k_1 \cdot |L| \log \frac{1}{p}\right)$.

Proof. Let $X' = X - \chi$, and let $\widehat{U}$ be the output of HashToBins and $\widehat{U}^*$ its exact counterpart. We start by calculating $W_f^{(t)}$ for an arbitrary $f \in \mathscr{F}$:

$$\begin{aligned} W_f^{(t)} &= \widehat{G}_{o_f(f)}^{-1} \widehat{U}_{h(f)} \omega^{-\sigma \Delta f} \\ &= \widehat{G}_{o_f(f)}^{-1} \widehat{U}^*_{h(f)} \omega^{-\sigma \Delta f} + \widehat{G}_{o_f(f)}^{-1} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega^{-\sigma \Delta f} \\ &= \widehat{X}'_f + \text{err}_f^{(t)} + \widetilde{\text{err}}_f^{(t)} \quad \text{(by Lemma 1.4.2),} \end{aligned} \tag{A.46}$$

where $\mathrm{err}_f^{(t)} = \widehat{G}_{o_f(f)}^{-1} \sum_{f' \in [n] \setminus \{f\}} \widehat{X}'_{f'} \widehat{G}_{o_f(f')} \omega^{\sigma\Delta(f'-f)}$, and $\widetilde{\mathrm{err}}_f^{(t)} = \widehat{G}_{o_f(f)}^{-1} (\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}) \omega^{-\sigma\Delta f}$, for (σ, Δ) implicitly depending on t.

Bounding $\mathrm{err}_f^{(t)}$ and $\widetilde{\mathrm{err}}_f^{(t)}$: Using Lemma A.4.1 in Appendix A.4, we have

$$\mathbb{E}_{\Delta,\pi}\left[\left|\mathrm{err}_f^{(t)}\right|^2\right] \le \frac{20}{B}\|\widehat{X}'\|_2^2 \tag{A.47}$$

$$\left|\widetilde{\mathrm{err}}_f^{(t)}\right| \le 2n^{-c+c'}\|\widehat{X}'\|_2, \tag{A.48}$$

where c is used in HashToBins, and c' is the exponent in the poly(n) notation of the assumption $\|\widehat{X} - \widehat{\chi}\|_2 \ge \frac{1}{\mathrm{poly}(n)}\|\widehat{\chi}\|_2$.

In order to characterize $|W_f^{(t)} - \widehat{X}'_f|^2$, we use the following:

$$|\widetilde{\mathrm{err}}_f|^2 + 2|\,\mathrm{err}_f^{(t)}\,| \cdot |\widetilde{\mathrm{err}}_f^{(t)}| \le 4n^{2(-c+c')}\|\widehat{X}'\|_2^2 + 4n^{-c+c'}\|\widehat{X}'\|_2 \cdot |\,\mathrm{err}_f^{(t)}\,|. \tag{A.49}$$

which follows directly from (A.48).

Characterizing $|W_f^{(t)} - \widehat{X}'_f|^2$: We have from (A.46), (A.47), and (A.49) that

$$\begin{aligned}
\mathbb{E}\left[|W_f^{(t)} - \widehat{X}'_f|^2\right] &\le \mathbb{E}\left[|\,\mathrm{err}_f^{(t)}\,|^2 + 2|\,\mathrm{err}_f^{(t)}\,| \cdot |\widetilde{\mathrm{err}}_f^{(t)}| + |\widetilde{\mathrm{err}}_f^{(t)}|^2\right] \\
&\le \frac{20}{B}\|\widehat{X}'\|_2^2 + 4n^{2(-c+c')}\|\widehat{X}'\|_2^2 + 4n^{-c+c'}\|\widehat{X}'\|_2 \mathbb{E}\left[|\,\mathrm{err}_f^{(t)}\,|\right] \\
&\le \frac{20}{B}\|\widehat{X}'\|_2^2 + 4n^{2(-c+c')}\|\widehat{X}'\|_2^2 + 4\sqrt{\frac{20}{B}}n^{-c+c'}\|\widehat{X}'\|_2^2,
\end{aligned} \tag{A.50}$$

where the last line follows from $\mathbb{E}\left[\left|\mathrm{err}_f^{(t)}\right|\right] \le \sqrt{\mathbb{E}\left[\left|\mathrm{err}_f^{(t)}\right|^2\right]}$ via Jensen's inequality, and then applying (A.47).

Since $B = \frac{1200k_0k_1}{\delta}$ and we have assumed $\delta \ge \frac{1}{n}$, we have $B \le 1200n^3$, and hence we have for sufficiently large c that (A.50) simplifies to $\mathbb{E}\left[\left|W_f^{(t)} - \widehat{X}'_f\right|^2\right] \le \frac{25}{B}\|\widehat{X}'\|_2^2$. This means that for any $\nu > 0$,

$$\Pr_{\Delta,\pi}\left[\left|W_f^{(t)} - \widehat{X}'_f\right|^2 \ge \frac{160\nu}{B}\|\widehat{X}'\|_2^2\right] \le \frac{\mathbb{E}_{\Delta,\pi}\left[\left|W_f^{(t)} - \widehat{X}'_f\right|^2\right]}{\frac{160\nu}{B}\|\widehat{X}'\|_2^2} \le \frac{1}{6\nu}. \tag{A.51}$$

by Markov's inequality.

Taking the median: Recall that W_f is the median of T independent random variables, with the median taken separately for the real and imaginary parts. Since $|W|^2 = |\mathrm{Re}(W)|^2 + |\mathrm{Im}(W)|^2$, we find that (A.51) holds true when $W_f^{(t)} - \widehat{X}'_f$ is replaced by its real or imaginary part. Hence, with probability at least $1 - \binom{T}{T/2}\left(\frac{1}{6\nu}\right)^{T/2}$, we have $|\mathrm{Re}(W_f^{(t)} - \widehat{X}'_f)|^2 < \frac{160\nu}{B}\|\widehat{X}'\|_2^2$, and analogously for

the imaginary part. Combining these and applying the union bound, we obtain

$$\Pr\left[|W_f - \widehat{X}'_f|^2 \geq \frac{320v}{B}\|\widehat{X}'\|_2^2\right] \leq 2\binom{T}{T/2}\left(\frac{1}{6v}\right)^{T/2} \leq 2\left(\frac{2}{3t}\right)^{T/2} \leq \frac{p}{v^{T/2}}, \tag{A.52}$$

where we first applied $\binom{T}{T/2} \leq 2^T$, and then the choice $T = 10\log\frac{2}{p}$ from Algorithm 17 and the choice of $p \leq 1/2$.

We now bound the error as follows:

$$\left|W_f - \widehat{X}'_f\right|^2 \leq \frac{320}{B}\|\widehat{X}'\|_2^2 + \left|\left|W_f - \widehat{X}'_f\right|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2\right|_+. \tag{A.53}$$

We write the expected value of the second term as,

$$\begin{aligned}
\mathbb{E}\left[\left||W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2\right|_+\right] &= \int_0^\infty \Pr\left[\left||W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2\right|_+ \geq u\right] du \\
&= \int_0^\infty \Pr\left[|W_f - \widehat{X}'_f|^2 \geq \frac{320}{B}\|\widehat{X}'\|_2^2 + u\right] du \\
&= \int_1^\infty \frac{320}{B}\|\widehat{X}'\|_2^2 \Pr\left[|W_f - \widehat{X}'_f|^2 \geq \frac{320v}{B}\|\widehat{X}'\|_2^2\right] dv
\end{aligned}$$

where we applied the change of variable $v = 1 + \frac{u}{\frac{320}{B}\|\widehat{X}'\|_2^2}$. By incorporating (A.52) into this integral, we obtain,

$$\begin{aligned}
\mathbb{E}\left[\left||W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2\right|_+\right] &\leq \frac{320}{B}\|\widehat{X}'\|_2^2 \int_1^\infty \frac{p}{v^{T/2}} dv \\
&\leq \frac{320}{B}\|\widehat{X}'\|_2^2 \cdot \frac{p}{T/2-1} \\
&\leq \frac{80}{B}\|\widehat{X}'\|_2^2 \cdot p,
\end{aligned} \tag{A.54}$$

where the second line is by explicitly evaluating the integral (with $T > 2$), and the third by $T/2 - 1 \geq 4$ (*cf.*, Algorithm 17).

Summing (A.53) over $\mathscr{F} = \cup_{j\in L} I_j$, we find that the total error is upper bounded as follows:

$$\sum_{f\in\mathscr{F}} |W_f - \widehat{X}'_f|^2 \leq \frac{320|\mathscr{F}|}{B}\|\widehat{X}'\|_2^2 + \sum_{f\in\mathscr{F}} \left||W_f - \widehat{X}'_f|^2 - \frac{320}{B}\|\widehat{X}'\|_2^2\right|_+.$$

From (A.54), the expected value of the second term above is at most $p \cdot \frac{80|\mathscr{F}|}{B}\|\widehat{X}'\|_2^2$, and hence,

$$\sum_{f\in\mathscr{F}} |W_f - \widehat{X}'_f|^2 \leq \frac{400|\mathscr{F}|}{B}\|\widehat{X}'\|_2^2,$$

with probability at least $1 - p$, by Markov's inequality. The lemma follows since $B = \frac{1200}{\delta}k_0k_1$ in Algorithm 17.

Sample complexity and runtime: To calculate the sample complexity, note that the only operation in the algorithm that takes samples is the call to HASHTOBINS. By Lemma 1.4.6, and the choices $B = 1200\frac{k_0 k_1}{\delta}$ and $F = 10\log\frac{1}{\delta}$, the sample complexity is $O\left(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta}\right)$ per performed hashing. Since we run the hashing in a loop $10\log\frac{2}{p}$ times, this amounts to a total of $O\left(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta}\log\frac{1}{p}\right)$.

The runtime depends on two operations. The first one is calling HASHTOBINS, whose analysis follows similarly to the aforementioned sample complexity analysis using the assumption $\|\widehat{\chi}\|_0 = O(k_0 k_1)$, but with an extra $\log n$ factor compared to the sample complexity, as per Lemma 1.4.6. The other operation is computation of $W_f^{(t)}$, which takes unit time for each $f \in \mathscr{F}$. Since the size of $|\mathscr{F}| = k_1 \cdot |L|$, running it in a loop costs $O\left(k_1 \cdot |L|\log\frac{1}{p}\right)$. Computing the median is done in linear time which consequently results in $|\mathscr{F}|T = O\left(k_1 \cdot |L|\log\frac{1}{p}\right)$. □

A.6 Analysis of REDUCESNR and RECOVERATCONSTSNR

Note on $\frac{1}{\text{poly}(n)}$ assumptions in lemmas: Throughout the proofs of Lemmas 1.5.3 and 1.5.4, we apply Lemmas 1.3.5, 1.5.1, and 1.5.2. The first of these assumes that $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\Omega\left(\frac{\|\widehat{\chi}\|_2}{\text{poly}(n)}\right)$ interval, and the latter two use the assumption $\|\widehat{X} - \widehat{\chi}\|_2 \geq \frac{1}{\text{poly}(n)}\|\widehat{\chi}\|_2$.

We argue that these assumptions are trivial and can be ignored. To see this, we apply a minor technical modification to the algorithm as follows. Suppose the implied exponent to the poly(n) notation is c'. By adding a noise term to $\widehat{\chi}_0$ on each iteration uniform in $\left[-n^{-c'+10}\|\widehat{\chi}\|_2, n^{-c'+10}\|\widehat{\chi}\|_2\right]$, we immediately satisfy the first assumption above, and we also find that the probability of $\|\widehat{X} - \widehat{\chi}\|_2 < \frac{1}{n^{c'}}\|\widehat{\chi}\|_2$ is at most n^{-10}, and the additional error in the estimate is $O\left(n^{-c'+10}\|\widehat{\chi}\|_2\right)$. Since we only do $O(\log\text{SNR}') = O(\log n)$ iterations (by the assumption $\text{SNR}' \leq \text{poly}(n)$), this does not affect the result because the accumulated noise added to $\widehat{\chi}_0$ which we denote by $\text{err}(\widehat{\chi}_0)$, does not exceed $\frac{\|\widehat{X}\|_2^2}{\text{poly}(n)}$ which by the final assumption of the lemma implies that $\text{err}(\widehat{\chi}_0) \leq \nu^2$.

A.6.1 Proof of Lemma 1.5.3

Overview of the proof: We introduce the *approximate support* set of the input signal $\widehat{X}$, given by the union of the top k_0 blocks of the signal and the blocks whose energy is more than the tail noise level:

$$S_0 := \left\{ j \in \left[\frac{n}{k_1}\right] : \|\widehat{X}_{I_j}\|_2^2 \geq \mu^2 \right\} \cup \left(\text{argmin}_{S \subset [\frac{n}{k_1}] \atop |S| = k_0} \sum_{j \in [\frac{n}{k_1}] \setminus S} \|\widehat{X}_{I_j}\|_2^2 \right). \tag{A.55}$$

From the definition of μ^2 in Definition 1.1.2, we readily obtain $|S_0| \leq 2k_0$. For each $t = 1, 2, ..., T$, define the set S_t as,

$$S_t = S_{t-1} \cup L'_t,$$

where L'_t is the output of PRUNELOCATION at iteration t of REDUCESNR. S_t contains the set of the head elements of $\widehat{X}$, plus every element that is modified by the algorithm so far.

We prove by induction on the iteration number $t = 1, \ldots, T$ that there exist events $\mathscr{E}_0 \supseteq \mathscr{E}_1 \supseteq ... \supseteq \mathscr{E}_T$ such that conditioned on $\mathscr{E}_t$, the followings hold true:

a. $|S_t| \leq 2k_0 + \frac{tk_0}{T}$;

b. $\|\widehat{\chi}^{(t)}_{I_j}\|_2^2 = 0$ for all $j \in \left[\frac{n}{k_1}\right] \setminus S_t$;

c. $\|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 \leq 99 \cdot \mathrm{SNR}'(k_0 v^2)/2^t$;

We prove using induction that for each $t \leq T$, $\Pr[\mathscr{E}_{t+1}|\mathscr{E}_t] \geq 1 - \frac{1}{10T}$.

Base case of the induction: We have already deduced that $|S_0| \leq 2k_0$. Furthermore, $\widehat{\chi}^{(0)} = 0$ by definition, and we have $\|\widehat{X} - \widehat{\chi}^{(0)}\|_2^2 = \|\widehat{X}\|_2^2 \leq \mathrm{SNR}' \cdot (k_0\mu^2)/2^0$ by assumption (2) of the lemma. Hence, we can let $\mathscr{E}_0$ be the trivial event satisfying $\Pr[\mathscr{E}_0] = 1$.

Inductive step: We seek to define an event $\mathscr{E}_{t+1}$ that occurs with probability at least $1 - \frac{1}{10T}$ conditioned on $\mathscr{E}_t$, and such that the induction hypotheses **a**, **b**, and **c** are satisfied conditioned on $\mathscr{E}_{t+1}$. To this end, we introduce three events $\mathscr{E}_{\mathrm{loc},t}$, $\mathscr{E}_{\mathrm{prune},t}$, and $\mathscr{E}_{\mathrm{est},t}$, and set $\mathscr{E}_{t+1} = \mathscr{E}_{\mathrm{loc},t} \cap \mathscr{E}_{\mathrm{prune},t} \cap \mathscr{E}_{\mathrm{est},t} \cap \mathscr{E}_t$. In what follows, we let δ, θ, and p be chosen as in Algorithm 6

Success event associated with MULTIBLOCKLOCATE: Let $\mathscr{E}_{\mathrm{loc},t}$ be the event of having a successful run of MULTIBLOCKLOCATE$(X, \widehat{\chi}^{(t)}, n, k_1, k_0, \delta, p)$ at iteration $t+1$ of the algorithm. More precisely, $\mathscr{E}_{\mathrm{loc},t}$ corresponds to having the following conditions on the output list L:

$$|L| \leq C \cdot \frac{k_0}{\delta} \log \frac{k_0}{\delta} \log^3 \frac{1}{\delta p} \tag{A.56}$$

$$\sum_{j \in S_t \setminus L} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq 0.1 \|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2, \tag{A.57}$$

where C is a constant to be specified shortly. We proceed by invoking Lemma 1.3.5 with $S^* = S_t$. Note that inductive hypothesis (**a**) implies $|S_t| \leq (2 + \frac{t}{\log \mathrm{SNR}'})k_0 \leq 3k_0$. By Lemma 1.3.5 both (A.56) and (A.57) hold with probability at least $1 - p$: the first part of Lemma 1.3.5, implies that $|L| \leq C\frac{k_0}{\delta} \log \frac{k_0}{\delta} \log \frac{1}{p} \log^2 \frac{1}{\delta p}$, for an absolute constant C; the second part of Lemma 1.3.5, implies (A.57) provided that $\delta \leq \frac{1}{2000^2}$. So, the event $\mathscr{E}_{\mathrm{loc},t}$ occurs with probability $\Pr\left[\mathscr{E}_{\mathrm{loc},t}|\mathscr{E}_t\right] \geq 1 - p$.

Success event associated with PRUNELOCATION: Let $\mathscr{E}_{\mathrm{prune},t}$ be the event of having a successful run of PRUNELOCATION$(X, \widehat{\chi}^{(t)}, L, k_0, k_1, \delta, p, n, \theta)$ at iteration $t+1$ of the algorithm. More

precisely, $\mathscr{E}_{\text{prune},t}$ corresponds to the following conditions on the output list L':

$$|L' \setminus S_t| \le \frac{k_0}{T} \tag{A.58}$$

$$\sum_{j \in [\frac{n}{k_1}] \setminus L'} \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 0.2 \|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 + k_0 \left(\mu^2 + 33\nu^2 \text{SNR}'/2^{t+1}\right). \tag{A.59}$$

The probability of (A.58) **holding:** In order to bound $|L' \setminus S_t|$, first recall that the set S_{tail}, defined in Lemma 1.5.1 part (**a**), has the following form:

$$S_{\text{tail}} = \left\{ j \in \left[\frac{n}{k_1}\right] : \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2 \le \sqrt{\theta} - \sqrt{\frac{\delta}{k_0}} \|\widehat{X} - \widehat{\chi}^{(t)}\|_2 \right\}.$$

By substituting $\theta = 10 \cdot 2^{-(t+1)} \cdot \nu^2(\text{SNR}')$ and using $\|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 \le 99 \cdot \text{SNR}'(k_0\nu^2)/2^t$ from part **c** of the inductive hypothesis, we have

$$\begin{aligned}
&\sqrt{\theta} - \sqrt{\frac{\delta}{k_0}} \|\widehat{X} - \widehat{\chi}^{(t)}\|_2 \\
&\quad \ge \sqrt{10 \cdot 2^{-(t+1)} \cdot \nu^2(\text{SNR}')} - \sqrt{\frac{\delta}{k_0}} \sqrt{99 \cdot \text{SNR}'(k_0\nu^2)/2^t} \\
&\quad \ge \sqrt{9 \cdot \nu^2(\text{SNR}')/2^{t+1}},
\end{aligned}$$

where the last inequality holds when δ is sufficiently small. Hence,

$$S_{\text{tail}} \supseteq \left\{ j \in \left[\frac{n}{k_1}\right] : \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 9 \cdot \nu^2(\text{SNR}')/2^{t+1} \right\}. \tag{A.60}$$

Now, to prove that (A.58) holds with high probability, we write

$$|L' \setminus S_t| = |(L' \cap S_{\text{tail}}) \setminus S_t| + |L' \setminus (S_{\text{tail}} \cup S_t)|. \tag{A.61}$$

To upper bound the first term, note that by the first part of Lemma 1.5.1, we have

$$\mathbb{E}\left[\left|L' \cap S_{\text{tail}}\right|\right] \le \delta p \cdot |L|,$$

and hence by Markov's inequality, the following holds with probability at least $1 - \frac{1}{100T}$:

$$\begin{aligned}
\left|(L' \cap S_{\text{tail}}) \setminus S_t\right| &\le \left|L' \cap S_{\text{tail}}\right| \\
&\le 100T\delta p \cdot |L| \\
&\le 100T\delta p \cdot CT \frac{k_0}{\delta} \log \frac{k_0}{\delta} \log^3 \frac{1}{\delta p} \\
&= \frac{100C\delta \cdot k_0 \log^3 \frac{1}{\delta p}}{\log \frac{k_0}{\delta} \cdot \log^2 \text{SNR}'},
\end{aligned}$$

where the third line follows from (A.56) (we condition on $\mathscr{E}_{\mathrm{loc},t}$), and fourth line follows from the choices $T = \log \mathrm{SNR}'$ and $p = \frac{\delta}{\log^2 \frac{k_0}{\delta} \log^4 \mathrm{SNR}'}$ in Algorithm 6. Again using this choice of p, we claim that $\frac{100C\delta \log^3 \frac{1}{\delta p}}{\log \frac{k_0}{\delta} \cdot \log \mathrm{SNR}'} \leq 1$ for sufficiently small δ regardless of the values (k_0, SNR'); this is because the dependence of $1/p$ on k_0 and SNR' is logarithmic, so in the numerator contains $\log^3 \log k_0$ and $\log^3 \log \mathrm{SNR}'$ while the denominator contains $\log k_0$ and $\log \mathrm{SNR}'$. Hence,

$$\left|(L' \cap S_{\mathrm{tail}}) \backslash S_t\right| \leq \frac{k_0}{\log \mathrm{SNR}'}$$

with probability at least $1 - \frac{1}{100T}$.

We now show that the second term in (A.61) is zero, by showing that $S_{\mathrm{tail}} \cup S_t = [\frac{n}{k_1}]$. To see this, note that the term $v^2(\mathrm{SNR}')/2^{t+1}$ in the bound on S_{tail} in (A.60) satisfies

$$v^2(\mathrm{SNR}')/2^{t+1} \geq \frac{1}{2} v^2 \geq \frac{1}{2} \mu^2, \tag{A.62}$$

by applying $t \leq T = \log \mathrm{SNR}'$, followed by the first assumption of the lemma. Hence,

$$S_{\mathrm{tail}} \backslash S_t \supseteq \left\{ j \in \left[\frac{n}{k_1} \right] \backslash S_t : \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq 4\mu^2 \right\}.$$

By part **b** of the inductive hypothesis, we have $\|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 = \|\widehat{X}_{I_j}\|_2^2$ for all $j \notin S_t$, and hence,

$$S_{\mathrm{tail}} \backslash S_t \supseteq \left\{ j \in \left[\frac{n}{k_1} \right] \backslash S_t : \|\widehat{X}_{I_j}\|_2^2 \leq 4\mu^2 \right\}.$$

But from (A.55), we know that S_0 (and hence S_t) contains all $j \in \left[\frac{n}{k_1} \right]$ with $\|\widehat{X}_{I_j}\|_2^2 > 4\mu^2$, so we obtain $S_{\mathrm{tail}} \backslash S_t \supset \left[\frac{n}{k_1} \right] \backslash S_t$, and hence $S_{\mathrm{tail}} \cup S_t = (S_{\mathrm{tail}} \backslash S_t) \cup S_t = \left[\frac{n}{k_1} \right]$, as required.

Therefore, the probability of (A.58) holding conditioned on $\mathscr{E}_t$ and $\mathscr{E}_{\mathrm{loc},t}$ is at least $1 - \frac{1}{100T}$.

The probability of (A.59) **holding:** To show (A.59), we use the second part of Lemma 1.5.1. The set S_{head} therein is defined as,

$$S_{\mathrm{head}} = \left\{ j \in \left[\frac{n}{k_1} \right] : \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2 \geq \sqrt{\theta} + \sqrt{\frac{\delta}{k_0}} \|\widehat{X} - \widehat{\chi}^{(t)}\|_2 \right\}.$$

By substituting $\theta = 10 \cdot 2^{-(t+1)} \cdot v^2(\mathrm{SNR}')$ and using $\|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 \leq 99 \cdot \mathrm{SNR}'(k_0 v^2)/2^t$ from part **c**

of the inductive hypothesis, we have

$$\begin{aligned}
&\sqrt{\theta}+\sqrt{\frac{\delta}{k_0}}\|\widehat{X}-\widehat{\chi}\|_2 \\
&\quad=\sqrt{10\cdot 2^{-(t+1)}\cdot \nu^2(\mathrm{SNR}')}+\sqrt{\frac{\delta}{k_0}}\sqrt{99\cdot \mathrm{SNR}'(k_0\nu^2)/2^t} \\
&\quad\le \sqrt{11\cdot \nu^2(\mathrm{SNR}')/2^{t+1}}
\end{aligned}$$

for sufficiently small δ. Therefore,

$$S_{\mathrm{head}}\supseteq\left\{j\in\left[\frac{n}{k_1}\right]:\|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2\ge 11\cdot\nu^2(\mathrm{SNR}')/2^{t+1}\right\}. \tag{A.63}$$

Next, we write,

$$\begin{aligned}
\sum_{j\in[\frac{n}{k_1}]\setminus L'}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 = &\sum_{j\in(S_t\cap S_{\mathrm{head}}\cap L)\setminus L'}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2+\sum_{j\in(S_t\cap S_{\mathrm{head}})\setminus(L'\cup L)}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 \\
&+\sum_{j\in S_t\setminus(S_{\mathrm{head}}\cup L')}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2+\sum_{j\in[\frac{n}{k_1}]\setminus(L'\cup S_t)}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2,
\end{aligned} \tag{A.64}$$

and we proceed by upper bounding each and every one of the four terms.

Bounding the first term in (A.64)**:** By the second part of Lemma 1.5.1 and use of Markov's inequality, we have

$$\sum_{j\in(S_t\cap S_{\mathrm{head}}\cap L)\setminus L'}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2\le\delta\sum_{j\in L\cap S_{\mathrm{head}}}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2\le\delta\|\widehat{X}-\widehat{\chi}^{(t)}\|_2^2$$

with probability at least $1-p$.

Bounding the second term in (A.64)**:** Conditioned on $\mathscr{E}_{\mathrm{loc},t}$, we have

$$\sum_{j\in(S_t\cap S_{\mathrm{head}})\setminus(L\cup L')}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2\le\sum_{j\in S_t\setminus L}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2\le 0.1\|\widehat{X}-\widehat{\chi}^{(t)}\|_2^2,$$

where we have applied (A.57).

Bounding the third term in (A.64)**:** We have

$$\begin{aligned}
\sum_{j\in S_t\setminus(S_{\mathrm{head}}\cup L')}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 &\le\sum_{j\in S_t\setminus S_{\mathrm{head}}}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 \\
&\le|S_t\setminus S_{\mathrm{head}}|\cdot\max_{j\in S_t\setminus S_{\mathrm{head}}}\|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 \\
&\le|S_t|(11\cdot\nu^2\mathrm{SNR}'/2^{t+1})
\end{aligned}$$

by (A.63). Part **a** of the inductive hypothesis implies that $|S_t| \le 3k_0$, and hence

$$\sum_{j\in(L\cap S_t)\setminus(S_{\text{head}}\cup L')} \|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 \le 33k_0\cdot\nu^2\text{SNR}'/2^{t+1}.$$

Bounding the fourth term in (A.64)**:**

$$\begin{aligned}\sum_{j\in[\frac{n}{k_1}]\setminus(L'\cup S_t)} \|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2 &\le \sum_{[\frac{n}{k_1}]\setminus S_t} \|(\widehat{X}-\widehat{\chi}^{(t)})_{I_j}\|_2^2\\ &= \sum_{[\frac{n}{k_1}]\setminus S_t} \|\widehat{X}_{I_j}\|_2^2 \le k_0\mu^2,\end{aligned}$$

where the equality follows from part **b** of the inductive hypothesis, and the final step holds since S_t contains all top k_0 blocks of $\widehat{X}$ (*cf.*, (A.55)).

Adding the above four contributions and applying union bound, we find that conditioned on $\mathscr{E}_t$ and $\mathscr{E}_{\text{loc},t}$, (A.59) holds with probability at least $\Pr\left[\mathscr{E}_{\text{prune},t}|\mathscr{E}_t\cap\mathscr{E}_{\text{loc},t}\right] \ge 1-p-\frac{1}{100T}$, provided that δ is a sufficiently small constant ($\delta \le 0.1$).

Success event associated with ESTIMATEVALUES: Let $\mathscr{E}_{\text{est},t}$ be the event of having a successful run of ESTIMATEVALUES$(X,\widehat{\chi}^{(t)},L',k_0,k_1,\delta,p)$ at iteration $t+1$ of the algorithm conditioned on $\mathscr{E}_t$. More precisely, $\mathscr{E}_{\text{est},t}$ corresponds to having the following conditions on the output signal W:

$$\begin{aligned}W_f = 0 \text{ for all } f\notin\mathscr{F}&\\ \sum_{j\in L'} \|(\widehat{X}-\widehat{\chi}^{(t)}-W)_{I_j}\|_2^2 \le \delta\|\widehat{X}-\widehat{\chi}^{(t)}\|_2^2,&\end{aligned} \tag{A.65}$$

where $\mathscr{F}$ contains the frequencies within the blocks indexed by L'. By Lemma 1.5.2 and the fact that $|L'| \le 3k_0$ conditioned on $\mathscr{E}_{\text{prune},t}$, $\mathscr{E}_{\text{loc},t}$, and $\mathscr{E}_t$, it immediately follows that $\mathscr{E}_{\text{est},t}$ occurs with probability at least $\Pr\left[\mathscr{E}_{\text{est},t}|\mathscr{E}_{\text{prune},t}\cap\mathscr{E}_{\text{loc},t}\cap\mathscr{E}_t\right] \ge 1-p$.

Combining the events: We can now wrap everything up as follows:

$$\begin{aligned}\Pr\left[\mathscr{E}_{t+1}\middle|\mathscr{E}_t\right] &= \Pr\left[\mathscr{E}_{\text{loc},t}\cap\mathscr{E}_{\text{prune},t}\cap\mathscr{E}_{\text{est},t}\cap\mathscr{E}_t\middle|\mathscr{E}_t\right]\\ &= \Pr\left[\mathscr{E}_{\text{est},t}\middle|\mathscr{E}_{\text{loc},t}\cap\mathscr{E}_{\text{prune},t}\cap\mathscr{E}_t\right]\Pr\left[\mathscr{E}_{\text{prune},t}\middle|\mathscr{E}_{\text{loc},t}\cap\mathscr{E}_t\right]\Pr\left[\mathscr{E}_{\text{loc},t}\middle|\mathscr{E}_t\right].\end{aligned}$$

Substituting the probability bounds into the above equation, we have

$$\Pr\left[\mathscr{E}_{t+1}\middle|\mathscr{E}_t\right] \ge 1-3p-\frac{2}{100T} \ge 1-\frac{1}{20T},$$

by the choice of p in Algorithm 6 along with $T=\log\text{SNR}$.

Now we show that the event $\mathscr{E}_{t+1}=\mathscr{E}_{\text{loc},t}\cap\mathscr{E}_{\text{prune},t}\cap\mathscr{E}_{\text{est},t}\cap\mathscr{E}_t$ implies the induction hypothesis. Conditioned on $\mathscr{E}_{\text{prune},t}\cap\mathscr{E}_t$, we have (A.58), which immediately gives part **a**. Conditioned on $\mathscr{E}_{\text{loc},t}\cap\mathscr{E}_{\text{est},t}$, from the definition $S_{t+1}=S_t\cup L'$, part **b** of the inductive hypothesis follows from

the fact that only elements in L' are updated. Finally, conditioned on $\mathscr{E}_t \cap \mathscr{E}_{\text{prune},t} \cap \mathscr{E}_{\text{est},t}$, we have

$$\begin{aligned}
\|\widehat{X} - \widehat{\chi}^{(t+1)}\|_2^2 &= \sum_{j \in L'} \|(\widehat{X} - \widehat{\chi}^{(t+1)})_{I_j}\|_2^2 + \sum_{j \in [\frac{n}{k_1}] \setminus L'} \|(\widehat{X} - \widehat{\chi}^{(t+1)})_{I_j}\|_2^2 \\
&= \sum_{j \in L'} \|(\widehat{X} - \widehat{\chi}^{(t)} - W)_{I_j}\|_2^2 + \sum_{j \in [\frac{n}{k_1}] \setminus L'} \|(\widehat{X} - \widehat{\chi}^{(t+1)})_{I_j}\|_2^2 \\
&\le (0.2 + \delta)\|\widehat{X} - \widehat{\chi}^{(t)}\|_2^2 + k_0(\mu^2 + 33\nu^2 \text{SNR}'/2^{t+1}) \\
&\le 99\nu^2 k_0 \text{SNR}'/2^{t+1},
\end{aligned}$$

where the second line holds since W is non-zero only for the blocks indexed by L', the third line follows from (A.59) and (A.65), and the last line holds for sufficiently small δ from part **c** of the induction hypothesis, and the upper bound $\mu^2 \le 2\nu^2(\text{SNR}')/2^{t+1}$ given in (A.62).

The first part of the lemma now follows from a union bound over the T iterations and the fact that the accumulated error $\text{err}(\widehat{\chi}_0) \le \nu^2$, and by noting that the three parts of the induction hypothesis immediately yield the two claims therein. We conclude by analyzing the sample complexity and runtime.

Sample complexity: By Lemma 1.3.5, the sample complexity of MULTIBLOCKLOCATE in a given iteration is $O^*\big(\frac{k_0}{\delta}\log(1+k_0)\log n + \frac{k_0 k_1}{\delta^2}\big)$, and multiplying by the number $T = O(\log \text{SNR}')$ of iterations gives a total of $O^*\big(\log\text{SNR}'\big(\frac{k_0}{\delta}\log(1+k_0)\log n + \frac{k_0 k_1}{\delta^2}\big)\big)$. Since $\delta = \Omega(1)$, the above sample complexity simplifies to $O^*\left(k_0 \log(1+k_0)\log\text{SNR}'\log n + k_0 k_1 \log \text{SNR}'\right)$.

By Lemma 1.5.1, the sample complexity of PRUNELOCATION is $O\Big(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta p}\log\frac{1}{\delta}\Big)$, and by Lemma 1.5.2, the sample complexity of ESTIMATEVALUES is $O\Big(\frac{k_0 k_1}{\delta}\log\frac{1}{p}\log\frac{1}{\delta}\Big)$. Substituting the choices of δ and p, these behave as $O^*(k_0 k_1)$ per iteration, or $O^*\left(k_0 k_1 \log \text{SNR}'\right)$ overall.

Runtime: By Lemma 1.3.5, the runtime of MULTIBLOCKLOCATE in a given iteration t is $O^*\big(\frac{k_0}{\delta}\log(1+k_0)\log^2 n + \frac{k_0 k_1}{\delta^2}\log^2 n + \frac{k_0 k_1}{\delta}\log^3 n\big)$. Moreover, by Lemma 1.5.1, the runtime of PRUNELOCATION as a function of $|L|$ is $O\Big(\frac{k_0 k_1}{\delta}\log\frac{1}{\delta p}\log\frac{1}{\delta}\log n + k_1 \cdot |L|\log\frac{1}{\delta p}\Big)$. Conditioned on $\mathscr{E}_{\text{loc},t}$ it holds that $|L| = O\big(\frac{k_0}{\delta}\log\frac{k_0}{\delta}\log^3\frac{1}{\delta p}\big)$ (see (A.56)), and substituting this in the runtime of PRUNELOCATION we get $O^*\Big(\frac{k_0 k_1}{\delta}\log n\Big)$, by absorbing the $\log\frac{1}{\delta}$ and $\log\frac{1}{p}$ factors into the $O^*(\cdot)$ notation.

Summing the preceding per-iteration expected runtimes, multiplying by the number of iterations T, and substituting the choices of T, p and δ, we find that the combined runtime across all calls to MULTIBLOCKLOCATE and PRUNELOCATION is $O^*(k_0 \log k_0 \log\text{SNR}'\log^2 n + k_0 k_1 \log\text{SNR}'\log^3 n)$.

By Lemma 1.5.2, the runtime of ESTIMATEVALUES is $O\Big(\frac{k_0 k_1}{\delta}\log\frac{1}{p}\log\frac{1}{\delta}\log n + k_1 \cdot |L'|\log\frac{1}{p}\Big)$,

which behaves as $O\left(\frac{k_0 k_1}{\delta}\log\frac{1}{p}\log\frac{1}{\delta}\log n\right)$ conditioned on $\mathscr{E}_{\text{prune},t}\cap\mathscr{E}_t$ (see (A.58) and recall that $|S_t|\le 3k_0$). By our choices of p and δ, this simplifies to $O^*(k_0k_1\log n)$ per iteration, or $O^*\left(k_0k_1\log\text{SNR}'\log n\right)$ overall.

A.6.2 Proof of Lemma 1.5.4

The proof resembles that of Lemma 1.5.3, but is generally simpler, and has some differing details. We provide the details for completeness.

Overview of the proof: We first introduce the *approximate support* set of the input signal $\widehat{X}-\widehat{\chi}$, given by the top $10k_0$ blocks of the signal:

$$S_0 := \operatorname{argmin}_{\substack{S\subset[\frac{n}{k_1}]\\ |S|=10k_0}} \sum_{j\notin S}\|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \tag{A.66}$$

We also introduce another set indexing blocks whose energy is sufficiently large:

$$S_\epsilon = \left\{ j\in\left[\frac{n}{k_1}\right] : \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \ge \epsilon\frac{\text{Err}^2(\widehat{X}-\widehat{\chi},10k_0,k_1)}{k_0}\right\}\bigcup S_0. \tag{A.67}$$

It readily follows from the above definition and Definition 1.1.2 that $|S_\epsilon\backslash S_0|\le k_0/\epsilon$.

The function calls three other primitives, and below, we show that each of them succeeds with high probability by introducing suitable success events. Throughout, we let θ, p, and η be as chosen in Algorithm 6.

Success event of the location primitive: Let $\mathscr{E}_{\text{loc}}$ be the event of having a successful run of MULTIBLOCKLOCATE$(X,\widehat{\chi},k_1,k_0,n,\epsilon^2,p)$, defined as the following conditions on the list L:

$$|L|\le C\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon^2}\log^3\frac{1}{\epsilon^2 p} \tag{A.68}$$

$$\sum_{j\in S_0\backslash L}\|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \le 200\epsilon\|\widehat{X}-\widehat{\chi}\|_2^2, \tag{A.69}$$

where C is a constant to be specified shortly. To verify these conditions, we invoke Lemma 1.3.5 with $S^*=S_0$. Lemma 1.3.5, both (A.68) and (A.69) hold with probability at least $1-p$. By the first part of Lemma 1.3.5, we have $|L|\le C\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon}\log^3\frac{1}{\epsilon p}$ for an absolute constant C. The second part of Lemma 1.3.5 with $\delta=\epsilon^2$, implies (A.69). So, the event $\mathscr{E}_{\text{loc}}$ occurs with probability at least $1-p$.

Success event of the pruning primitive: Let $\mathscr{E}_{\text{prune}}$ be the event of having a successful run of

PRUNELOCATION$(X, \widehat{\chi}, L, n, k_0, k_1, \epsilon, p, \theta)$, meaning the following conditions on the output L':

$$|L' \setminus S_0| \leq \frac{2k_0}{\epsilon} \tag{A.70}$$

$$\sum_{j \in [\frac{n}{k_1}] \setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \leq 300\epsilon \|\widehat{X} - \widehat{\chi}\|_2^2 + 6000\epsilon\nu^2 k_0 + \mathrm{Err}^2(\widehat{X} - \widehat{\chi}, 10k_0, k_1). \tag{A.71}$$

The probability of (A.70)**:** In order to bound $|L' \setminus S_0|$, first note that the set S_{tail}, defined in Lemma 1.5.1 part (**a**), has the following form:

$$S_{\text{tail}} = \left\{ j \in \left[\frac{n}{k_1} \right] : \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2 \leq \sqrt{\theta} - \sqrt{\frac{\epsilon}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 \right\}.$$

By substituting $\theta = 200 \cdot \epsilon\nu^2$ (*cf.*, Algorithm 6) and using the assumption $\|\widehat{X} - \widehat{\chi}\|_2^2 \leq 100k_0\nu^2$ in the lemma, we have,

$$\begin{aligned} \sqrt{\theta} - \sqrt{\frac{\epsilon}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 &= \sqrt{200 \cdot \epsilon\nu^2} - \sqrt{\frac{\epsilon}{k_0}} \|\widehat{X} - \widehat{\chi}\|_2 \\ &\geq \sqrt{200 \cdot \epsilon\nu^2} - \sqrt{100 \cdot \epsilon\nu^2} \\ &\geq \sqrt{16 \cdot \epsilon\nu^2}. \end{aligned} \tag{A.72}$$

Hence,

$$S_{\text{tail}} \supseteq \left\{ j \in \left[\frac{n}{k_1} \right] : \|(\widehat{X} - \widehat{\chi}^{(t)})_{I_j}\|_2^2 \leq 16\epsilon \cdot \nu^2 \right\}.$$

Now, to prove that (A.70) holds, we write,

$$\begin{aligned} |L' \setminus S_0| &= |(L' \cap S_\epsilon) \setminus S_0| + |L' \setminus (S_0 \cup S_\epsilon)| \\ &\leq |(L' \cap S_\epsilon) \setminus S_0| + |(L' \cap S_{\text{tail}}) \setminus S_\epsilon| + |L' \setminus (S_{\text{tail}} \cup S_\epsilon)|. \end{aligned} \tag{A.73}$$

We first upper bound the first term as follows:

$$|(L' \cap S_\epsilon) \setminus S_0| \leq |S_\epsilon \setminus S_0| \leq k_0/\epsilon,$$

which follows directly from the definition of S_ϵ. To upper bound the second term in (A.73), note that by Lemma 1.5.1 part (**a**) with $\delta = \epsilon$,

$$\mathbb{E}\left[\left| L' \cap S_{\text{tail}} \right| \right] \leq \epsilon p \cdot |L|,$$

and hence by Markov's inequality, the following holds with probability at least $1 - \frac{1}{100}$:

$$\begin{aligned}\left|(L' \cap S_{\text{tail}}) \backslash S_\epsilon\right| &\le \left|L' \cap S_{\text{tail}}\right| \\ &\le 100\epsilon p \cdot |L| \\ &\le 100\epsilon p \cdot C\frac{k_0}{\epsilon}\log\frac{k_0}{\epsilon}\log^3\frac{1}{\epsilon p} \\ &= 100Cp \cdot k_0 \log\frac{k_0}{\epsilon}\log^3\frac{1}{\epsilon p} \\ &= \frac{100C\eta\epsilon \cdot k_0 \log^3\frac{1}{\epsilon p}}{\log\frac{k_0}{\epsilon}},\end{aligned}$$

where the third line follows from (A.68) (we condition on $\mathscr{E}_{\text{loc}}$), and the fifth line follows from and the choice $p = \frac{\eta\epsilon}{\log^2\frac{k_0}{\epsilon}}$ in Algorithm 6. Again using this choice of p, we claim that $\frac{100C\eta\epsilon\log^3\frac{1}{\epsilon p}}{\log\frac{k_0}{\delta}} \le 1$ for sufficiently small η regardless of the value of k_0; this is because the dependence of $1/p$ on k_0 is logarithmic, so the numerator contains $\log^3\log k_0$, while the denominator contains $\log k_0$ which means that the ratio is upper bounded and can be made arbitrarily small by choosing a small enough constant η. Hence

$$\left|(L' \cap S_{\text{tail}}) \backslash S_\epsilon\right| \le k_0$$

with probability at least $1 - \frac{1}{100}$.

We now show that the third term in (A.73) is zero, by showing that $S_{\text{tail}} \cup S_\epsilon = [\frac{n}{k_1}]$. To see this, note that the term v^2 in the definition of S_{tail} is more than $\frac{\text{Err}^2(\widehat{X}-\widehat{\chi},10k_0,k_1)}{k_0}$ by the first assumption of the lemma, and hence

$$S_{\text{tail}} \backslash S_\epsilon \supset \left\{ j \in \left[\frac{n}{k_1}\right] \backslash S_\epsilon : \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \le 16\epsilon\frac{\text{Err}^2(\widehat{X}-\widehat{\chi},10k_0,k_1)}{k_0} \right\}.$$

However, the definition of S_ϵ in (A.67) reveals that the condition upper bounding $\|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2$ is redundant, and $S_{\text{tail}} \backslash S_\epsilon \supset \left[\frac{n}{k_1}\right] \backslash S_\epsilon$, and hence $S_{\text{tail}} \cup S_\epsilon = (S_{\text{tail}} \backslash S_\epsilon) \cup S_\epsilon = \left[\frac{n}{k_1}\right]$.

Bounding the probability of (A.71)**:** To show (A.71), we use the second part of Lemma 1.5.1. The set S_{head} therein is defined as

$$S_{\text{head}} = \left\{ j \in \left[\frac{n}{k_1}\right] : \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2 \ge \sqrt{\theta} + \sqrt{\frac{\epsilon}{k_0}}\|\widehat{X}-\widehat{\chi}\|_2 \right\}.$$

By substituting $\theta = 200\epsilon \cdot v^2$ (*cf.*, Algorithm 6) and using the assumption $\|\widehat{X}-\widehat{\chi}\|_2^2 \le 100k_0v^2$ in the lemma, we have

$$\sqrt{\theta} + \sqrt{\frac{\epsilon}{k_0}}\|\widehat{X}-\widehat{\chi}\|_2 = \sqrt{200\epsilon \cdot v^2} + \sqrt{\frac{\epsilon}{k_0}}\|\widehat{X}-\widehat{\chi}\|_2 \le \sqrt{600\epsilon \cdot v^2},$$

and hence

$$S_{\text{head}} \supseteq \left\{ j \in \left[\frac{n}{k_1} \right] : \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \geq 600\epsilon \cdot \nu^2 \right\}. \tag{A.74}$$

Next, we write

$$\begin{aligned} \sum_{j \in [\frac{n}{k_1}] \setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 = & \sum_{j \in (S_0 \cap S_{\text{head}} \cap L) \setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 + \sum_{j \in (S_0 \cap S_{\text{head}}) \setminus (L' \cup L)} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \\ & + \sum_{j \in S_0 \setminus (S_{\text{head}} \cup L')} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 + \sum_{j \in [\frac{n}{k_1}] \setminus (L' \cup S_0)} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2, \end{aligned} \tag{A.75}$$

and we proceed by upper bounding the four terms.

Bounding the first term in (A.75)**:** By part **b** of Lemma 1.5.1, the choice $\delta = \epsilon$, and the use of Markov, we have

$$\sum_{j \in (S_0 \cap S_{\text{head}} \cap L) \setminus L'} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \leq \epsilon \sum_{j \in L \cap S_{\text{head}}} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \leq \epsilon \|\widehat{X} - \widehat{\chi}\|_2^2$$

with probability at least $1 - p$.

Bounding the second term in (A.75)**:** Conditioned on $\mathscr{E}_{\text{loc}}$, we have

$$\begin{aligned} \sum_{j \in (S_0 \cap S_{\text{head}}) \setminus (L \cup L')} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 &\leq \sum_{j \in S_0 \setminus L} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \\ &\leq 200\epsilon \|\widehat{X} - \widehat{\chi}\|_2^2, \end{aligned}$$

where we have applied (A.69).

Bounding the third term in (A.75)**:** We have

$$\begin{aligned} \sum_{j \in S_0 \setminus (S_{\text{head}} \cup L')} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 &\leq \sum_{j \in S_0 \setminus S_{\text{head}}} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \\ &\leq |S_0 \setminus S_{\text{head}}| \cdot \max_{j \in S_0 \setminus S_{\text{head}}} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \\ &\leq |S_0| (600\epsilon \cdot \nu^2) \end{aligned}$$

by (A.74). We have by definition that $|S_0| = 10k_0$ (*cf.*, (A.66)), and hence

$$\sum_{j \in (L \cap S_0) \setminus (S_{\text{head}} \cup L')} \|(\widehat{X} - \widehat{\chi})_{I_j}\|_2^2 \leq 6000 k_0 \epsilon \cdot \nu^2.$$

Bounding the fourth term in (A.75)**:** We have

$$\sum_{j\in[\frac{n}{k_1}]\setminus(L'\cup S_0)} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 \le \sum_{j\in[\frac{n}{k_1}]\setminus S_0} \|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 = \mathrm{Err}^2(\widehat{X}-\widehat{\chi}, 10k_0, k_1),$$

which follows from the definition of S_0 in (A.66), along with Definition 1.1.2.

Hence by the union bound, it follows that $\mathscr{E}_{\text{prune}}$ holds with probability at least $1-p-\frac{1}{1000}$ conditioned on $\mathscr{E}_{\text{loc}}$.

Success event of estimation primitive: Let $\mathscr{E}_{\text{est}}$ be the event of having a successful run of EstimateValues$(X,\widehat{\chi},L,n,3k_0/\epsilon,k_1,\epsilon,p)$, meaning the following conditions on the output, W:

$$W_f = 0 \text{ for all } f\notin\mathscr{F}$$
$$\sum_{j\in L'} \|(\widehat{X}-\widehat{\chi}-W)_{I_j}\|_2^2 \le \epsilon\|\widehat{X}-\widehat{\chi}\|_2^2, \tag{A.76}$$

where $\mathscr{F}$ contains the frequencies within the blocks indexed by L'. Since the assumption of the theorem implies that $\|\widehat{\chi}\|_0 = O(k_0k_1)$, by Lemma 1.5.2 (with $\delta=\epsilon$ and $3k_0/\epsilon$ in place of k_0) and the fact that conditioned on $\mathscr{E}_{\text{prune}}$ we have $|L'|\le 3k_0/\epsilon$ (*cf.*, (A.70)), it follows that $\mathscr{E}_{\text{est}}$ occurs with probability at least $1-p$.

Combining the events: We can now can wrap everything up.

Letting $\mathscr{E}$ denote the overall success event corresponding to the claim of the lemma, we have

$$\begin{aligned}\Pr[\mathscr{E}] &= \Pr\big[\mathscr{E}_{\text{loc}}\cap\mathscr{E}_{\text{prune}}\cap\mathscr{E}_{\text{est}}\big]\\ &= \Pr\big[\mathscr{E}_{\text{est}}\big|\mathscr{E}_{\text{loc}}\cap\mathscr{E}_{\text{prune}}\big]\Pr\big[\mathscr{E}_{\text{prune}}\big|\mathscr{E}_{\text{loc}}\big]\Pr\big[\mathscr{E}_{\text{loc}}\big].\end{aligned}$$

By the results that we have above, along with the union bound, it follows that

$$\Pr\big[\mathscr{E}\big]\ge 1-2/100-3p\ge 0.95$$

for sufficiently small η in Algorithm 6. A union bound over $\bar{\mathscr{E}}$ and the $1/\operatorname{poly}(n)$ probability failure event arising from random perturbations of $\widehat{\chi}_0$ yields the required bound of 0.9 on the success probability.

What remains is to first show that the statement of the lemma follows from $\mathscr{E}_{\text{loc}}\cap\mathscr{E}_{\text{prune}}\cap\mathscr{E}_{\text{est}}$.

To do this, we observe that, conditioned on these events,

$$\begin{aligned}\|\widehat{X}-\widehat{\chi}'\|_2^2 &= \sum_{j\in L'}\|(\widehat{X}-\widehat{\chi}')_{I_j}\|_2^2 + \sum_{j\in[\frac{n}{k_1}]\setminus L'}\|(\widehat{X}-\widehat{\chi}')_{I_j}\|_2^2 + \mathrm{err}(\widehat{\chi}_0)\\ &= \sum_{j\in L'}\|(\widehat{X}-\widehat{\chi}-W)_{I_j}\|_2^2 + \sum_{j\in[\frac{n}{k_1}]\setminus L'}\|(\widehat{X}-\widehat{\chi})_{I_j}\|_2^2 + \mathrm{err}(\widehat{\chi}_0)\\ &\le \epsilon\|\widehat{X}-\widehat{\chi}\|_2^2 + 300\epsilon\|\widehat{X}-\widehat{\chi}\|_2^2 + 6000\epsilon v^2 k_0 + \mathrm{Err}^2(\widehat{X}-\widehat{\chi}, 10k_0, k_1) + \epsilon v^2\\ &\le (4\cdot 10^5)\epsilon v^2 k_0 + \mathrm{Err}^2(\widehat{X}-\widehat{\chi}, 10k_0, k_1),\end{aligned}$$

where the second line follows since $\widehat{\chi}' = \widehat{\chi} + W$ and W is non-zero only within the blocks indexed by L', the third line follows from (A.71) and (A.76), and the final line follows from the assumption $\|\widehat{X}-\widehat{\chi}\|_2^2 \le 100k_0v^2$ in the lemma.

Sample complexity: By Lemma 1.3.5 with $\delta = \epsilon^2$, the sample complexity of MULTIBLOCKLOCATE is $O^*\left(\frac{k_0}{\epsilon^2}\log(1+k_0)\log n + \frac{k_0k_1}{\epsilon^4}\log\frac{1}{p}\right)$. By Lemma 1.5.1 with $\delta=\epsilon$, the sample complexity of PRUNELOCATION is $O\left(\frac{k_0k_1}{\epsilon}\log\frac{1}{\epsilon p}\log\frac{1}{\epsilon}\right)$. In addition, the sample complexity of ESTIMATEVALUES is $O\left(\frac{k_0k_1}{\epsilon^2}\log\frac{1}{p}\log\frac{1}{\epsilon}\right)$, by Lemma 1.5.2 with $\delta=\epsilon$. By summing the three terms, and noting from the choice of p in Algorithm 6 that, up to $\log\log\frac{k_0}{\epsilon}$ factors, we can replace p by ϵ in the above calculations, the total sample complexity of this procedure is $O^*\left(\frac{k_0}{\epsilon^2}\log(1+k_0)\log n + \frac{k_0k_1}{\epsilon^4}\right)$.

Runtime: By Lemma 1.3.5 with $\delta=\epsilon^2$, and because by assumption, $\widehat{\chi}$ is $(O(k_0),k_1)$-block sparse, the runtime of MULTIBLOCKLOCATE is $O^*\left(\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon}\cdot\log^2 n + \frac{k_0k_1}{\epsilon^4}\log^2 n + \frac{k_0k_1}{\epsilon^2}\log^3 n\right)$. By Lemma 1.5.1 with $\delta=\epsilon$, the runtime of PRUNELOCATION as a function of $|L|$ is $O^*\left(\frac{k_0k_1}{\epsilon}\log n + k_1\cdot|L|\right)$, which behaves as $O\left(\frac{k_0k_1}{\epsilon}\log n + k_1\cdot\frac{k_0}{\epsilon^2}\log\frac{k_0}{\epsilon}\right)$ conditioned on $\mathscr{E}_{\mathrm{loc}}$ (see (A.68)). Moreover, by Lemma 1.5.2 with $\delta=\epsilon$, the runtime of the primitive ESTIMATEVALUES is $O\left(\frac{k_0k_1}{\epsilon}\log\frac{1}{p}\log\frac{1}{\epsilon}\log n + k_1\cdot|L'|\log\frac{1}{p}\right)$, which behaves as $O\left(\frac{k_0k_1}{\epsilon}\log\frac{1}{p}\log\frac{1}{\epsilon}\log n\right)$ conditioned on $\mathscr{E}_{\mathrm{prune}}$ (see (A.70) and recall that $|S_0|=10k_0$). The total runtime follows by summing the above terms and replacing p by ϵ, with the $\log\log n$, $\log\log\mathrm{SNR}'$ and $\log\frac{1}{\epsilon}$ terms absorbed into the $O^*(\cdot)$ notation and applying Markov's inequality.

A.7 Discussion on Energy-based Importance Sampling

Here we provide further discussing on the adaptive energy-based importance sampling scheme described in Sections 1.2–1.3. Recall from Definition 1.2.2 that given the signal X and filter G, we are considering downsampled signals of the form $\widehat{Z}_j^r = (\widehat{X}^r \star \widehat{G})_{jk_1}$ with $X_i^r = X_{i+\frac{nr}{2k_1}}$ for $r\in[2k_1]$, and recall from (1.3) that the goal of energy-based importance sampling is to

approximately solve the covering problem

$$\text{Minimize}_{\{s^r\}_{r\in[2k_1]}} \sum_{r\in[2k_1]} s^r \quad \text{subject to} \quad \sum_{\substack{j:|\widehat{Z}_j^r|^2\geq\frac{\|\widehat{Z}^r\|_2^2}{s^r}\\ \text{for some } r\in[2k_1]}} \|\widehat{X}_{I_j}\|_2^2 \geq (1-\alpha)\|\widehat{X}^*\|_2^2 \tag{A.77}$$

for suitable $\alpha \in (0,1)$, where $\widehat{X}^*$ is the best $(k_0.k_1)$-block sparse approximation of $\widehat{X}$.

To ease the discussion, we assume throughout this appendix that the filter G is a width-$\frac{n}{k_1}$ rectangle in time domain, corresponding to a sinc pulse of "width" k_1 in frequency domain. Such a filter is less tight than the one we use (see the proof of Lemma 1.2.1), but similar enough for the purposes of the discussion.

A.7.1 Examples – Flat vs. Spiky Energies

We begin by providing two examples for the 1-block sparse case, demonstrating how the energies can vary with r. An illustration of the energy in each $\widehat{Z}^r$ is illustrated in Figure A.1 in two different cases – one in which X is a sinc pulse (i.e., rectangular in frequency domain), and one in which X is constant (i.e., a delta function in frequency domain). Both of the signals are $(1,k_1)$-block sparse with $k_1 = 16$, and the signal energy is the same in both cases. However, the sinc pulse gives significantly greater variations in $|\widehat{Z}_j^r|^2$ as a function of r. In fact, these examples demonstrate two extremes that can occur – in one case, the energy exhibits no variations, and in the other case, the energy is $O(k_1)$ times its expected value for an $O\big(\frac{1}{k_1}\big)$ fraction of the r values.

The second example above is, of course, an extreme case of a $(1,k_1)$-block sparse signal, because it is also $(1,1)$-block sparse. Nevertheless, one also observes a similar flatness in time domain for other signals; e.g., one could take the first example above and randomize the signs, as opposed to letting them all be positive.

A.7.2 The $\log(1+k_0)$ factor

Here we provide an example demonstrating that, as long as we rely solely on frequencies being covered according to Definition 1.2.3, after performing the budget allocation, the extra $\log(1+k_0)$ factor in our analysis is unavoidable. Specifically, we argue that for a certain signal X, the *optimal* solution to (1.3) satisfies $\sum_{r\in[2k_1]} s^r = \Omega(k_0\log(1+k_0))$. However, we do not claim that this $\log(1+k_0)$ factor is unavoidable for *arbitrary* sparse FFT algorithms.
We consider a scenario where $k_0 = \Theta(k_1) = o(n)$, and for concreteness, we let both k_0 and k_1 behave as $\Theta(n^{0.1})$; hence, $\log(1+k_0) = O(\log k_0)$

Constructing a base signal: We first specify a base signal $W \in \mathbb{C}^n$ that will be used to construct

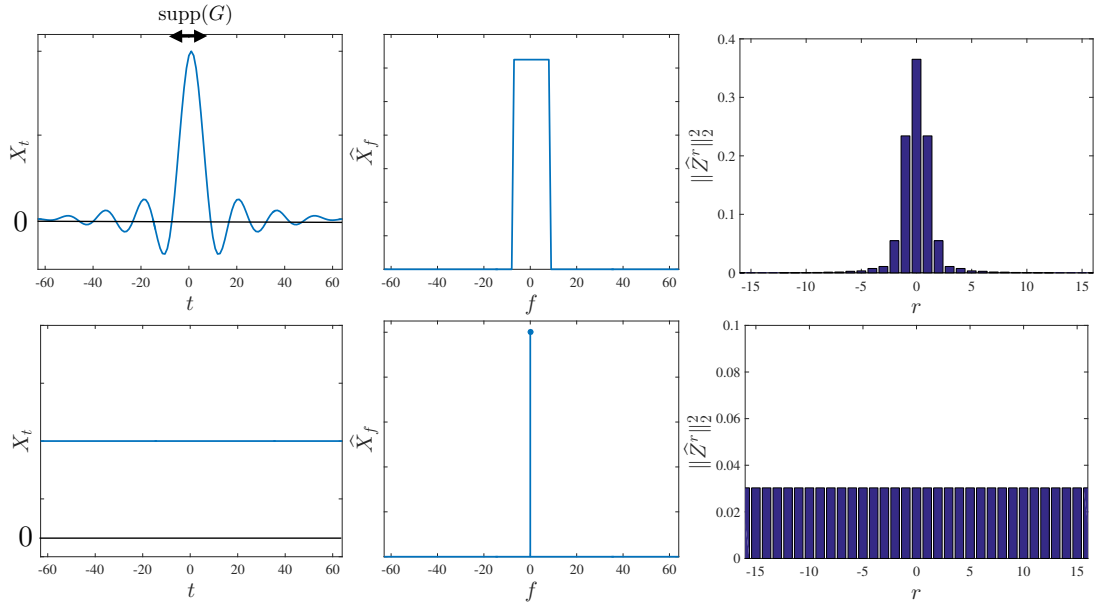


Figure A.1 – Behavior of $\|\widehat{Z}^r\|_2^2$ as a function of r for a sinc function (top) and a rectangular function (bottom), both of which are $(1,16)$-block sparse.

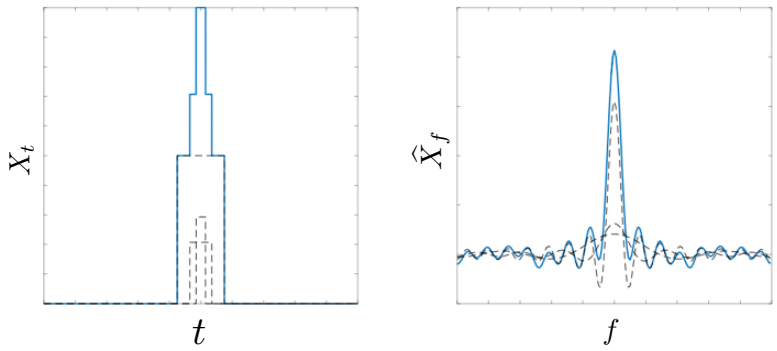


Figure A.2 – Base signal and its Fourier transform, for constructing a signal where a $\log k_0$ loss is unavoidable with our techniques.

the approximately (k_0, k_1)-block sparse signal. Specifically, we fix the integers C and L, and set

$$W_t = \begin{cases} \sqrt{2^{L-1}} & |t| \le \frac{Cn}{2k_1} \\ \sqrt{2^{L-2}} & \frac{Cn}{2k_1} < |t| \le \frac{3Cn}{2k_1} \\ \vdots & \vdots \\ \sqrt{2^{L-\ell}} & \frac{(2^{\ell-1}-1)Cn}{2k_1} < |t| \le \frac{(2^{\ell}-1)Cn}{2k_1} \\ \vdots & \vdots \\ 1 & \frac{(2^{L-1}-1)Cn}{2k_1} < |t| \le \frac{(2^{L}-1)Cn}{2k_1} \\ 0 & |f| > \frac{(2^{L+1}-1)Cn}{2k_1}. \end{cases} \tag{A.78}$$

Hence, the signal contains L regions of exponentially increasing width but exponentially decreasing magnitude. See Figure A.2 for an illustration ($L = 3$), and observe that we can express this function as a sum of rectangles having geometrically decreasing magnitudes. Hence, we can specify its Fourier transform as a sum of sinc functions.

The narrowest of the rectangles has width $\frac{Cn}{k_1}$, and hence the widest of the sinc pulses has width $\frac{k_1}{C}$. This means that by choosing C to be sufficiently large, we can ensure that an arbitrarily high proportion of the energy lies in a window of length k_1 in frequency domain, meaning W is approximately 1-block sparse.

Constructing a block-spare signal: We construct a k_0-block sparse signal by adding multiple copies of W together, each shifted by a different amount in time domain, and also modulated by a different frequency (i.e., shifted by a different amount in frequency domain). We choose L such that the cases in (A.78) collectively occupy the whole time domain, yielding $L = \Theta(\log k_1) = \Theta(\log k_0)$.

Then, we set $k_0 = \frac{k_1}{C}$ and let each copy of W be shifted by a multiple of $\frac{Cn}{k_1}$, so that the copies are separated by a distance equal to the length of the thinnest segment of W, and collectively these thin segments cover the whole space $[n]$. As for the modulation, we choose these so that the resulting peaks in frequency domain are separated by $\Omega(k_1^4)$, so that the tail of the copy of $\widehat{W}$ corresponding to one block has a negligible effect on the other blocks. This is possible within n coefficients, since we have chosen $k_1 = O(n^{0.1})$.

Evaluating the values of $|\widehat{Z}_j^r|^2$: Recall that we are considering G in (A.77) equaling a rectangle of width $\frac{n}{k_1}$. Because of the above-mentioned separation of the blocks in frequency domain, each copy of W can essentially be treated separately. By construction, within a window of length $\frac{n}{k_1}$, we have one copy of W at magnitude $\sqrt{2^{L-1}}$, two copies at magnitude $\sqrt{2^{L-2}}$, and so on. Upon subsampling by a factor of k_1, the relative magnitudes remain the same; there is no aliasing, since we let G be rectangular. Hence, the dominant coefficients in the spectrum of the subsampled signal exhibit this same structure, having energies of a form such as $(8,4,4,2,2,2,2,1,1,1,1,1,1,1,1)$ when sorted and scaled (up to negligible leakage effects). Moreover, the matrix of $|\widehat{Z}_j^r|^2$ values (*cf.*, Figure 1.1) essentially amounts to circular shifts of

a vector of this form – the structure of any given $\widehat{Z}^r$ maintains this geometric structure, but possibly in a different order.

Lower bounding the sum of budgets allocated: We now turn to the allocation problem in (A.77). Allocating a sparsity budget s to a signal Z^r covers all coefficients j for which $|\widehat{Z}^r_j|^2 \geq \frac{\|\widehat{Z}\|^2}{s}$. For the signal we have constructed, the total energy E is equally spread among the L geometric levels: The ℓ-th level consists of $2^{\ell-1}$ coefficients of energy $2^{1-\ell}\frac{E}{L}$, and hence covering that level requires $s \geq L \cdot 2^{\ell}$.

Hence, setting $s = L \cdot 2^{\ell-1}$ covers the top ℓ levels, for a total of $2^{\ell} - 1$ coefficients. That is, covering some number of coefficients requires letting s be $\Omega(L)$ times that number, and hence covering a constant fraction of the k_0 coefficients requires the sum of sparsity budgets to be $\Omega(Lk_0)$. Moreover, we have designed every block to have the same energy, so accounting for a constant fraction of the energy amounts to covering a constant fraction of the k_0 coefficients.

Since we selected $L = \Theta(\log k_0)$, this means that the sum of sparsity budgets is $\Omega(k_0 \log k_0)$, so that the $\log k_0$ factor must be present in any solution to (A.77).

A.8 Location of Reduced Signals

In Algorithm 18, we provide a location primitive that, given a sequence of budgets s^r, locates dominant frequencies in the sequence of reduced signals Z^r using $O\big(\sum_{r\in[2k_1]} s^r \log n\big)$ samples. The core of the primitive is a simple k-sparse recovery scheme, where k frequencies are hashed into $B = Ck$ buckets for a large constant $C > 1$, and then each bucket is decoded individually. Specifically, for each bucket that is approximately 1-sparse (i.e., dominated by a single frequency that hashed into it) the algorithm accesses the signal at about a logarithmic number of locations and decodes the bit representation of the dominant frequency bit by bit. More precisely, to achieve the right sample complexity we decode the frequencies in blocks of $O(\log\log n)$ bits. Such schemes or versions thereof have been used in the literature (e.g., Gilbert et al. (2005); Hassanieh et al. (2012c); Kapralov (2016)).

A novel aspect of our decoding scheme is that it receives access to the input signal X, but must run a basic sparse recovery scheme as above on each reduced signal Z^r. Specifically, for each r it must hash Z^r into s^r buckets (the *budget* computed in MULTIBLOCKLOCATE and passed to LOCATESIGNAL as input). This would be trivial since Z^r can be easily accessed given access to X (*cf.*, Lemma 1.2.2), but the fact that we need to operate on the *residual signal* $X - \chi$ (where $\widehat{\chi}$ is block sparse and given explicitly as input) introduces difficulties.

The difficulty is that we would like to compute χ on the samples that individual invocations of sparse recovery use, for each $r \in [2k_1]$, but computing this directly would be very costly. Our solution consists of ensuring that all invocations of sparse recovery use the same random permutation π, and therefore all need to access $X - \chi$ on a set of shifted intervals after a change of variables given by π (crucially, this change of variables is shared across all r). The

lengths of the intervals are different, and given by s^r, but it suffices to compute the values of χ on the shifts of the largest of these intervals, which is done in HASHTOBINSREDUCED (see Lemma 1.4.6). We present the details below in Algorithm 18.

For convenience, throughout this section, we use m to denote the reduced signal length n/k_1.

Lemma 1.2.4 (LOCATEREDUCEDSIGNAL guarantees – formal version)**.** *Fix integers n, k_0, k_1, signals $X, \widehat{\chi} \in \mathbb{C}^n$ with $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\frac{\|\widehat{\chi}\|_2}{\mathrm{poly}(n)}$ interval, sparsity budgets $\{s^r\}_{r\in[2k_1]}$ with $s^r = O\big(\frac{k_0}{\delta}\big)$ for all $r \in [2k_1]$, and parameters $\delta \in \big(\frac{1}{n}, \frac{1}{20}\big)$ and $p \in \big(\frac{1}{n^3}, \frac{1}{2}\big)$, and let $\{Z^r\}_{r\in[2k_1]}$ be the (k_1,δ)-downsampling of $X - \chi$.*

If L denotes the output of LOCATEREDUCEDSIGNALS$(X, \widehat{\chi}, n, k_0, k_1, \{s^r\}_{r\in[2k_1]}, \delta, p)$*, then for any $j \in \big[\frac{n}{k_1}\big]$ such that $|Z^r_j|^2 \geq \frac{\|Z^r\|_2^2}{s^r}$ for some $r \in [2k_1]$, one has $j \in L$ with probability at least $1-p$. Additionally, the list size satisfies $|L| = O\big(\sum_{r\in[2k_1]} s^r \log\frac{1}{p}\big)$ with probability 1.*
Moreover, if $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse, the sample complexity is $O\Big(\sum_{r\in[2k_1]} s^r \log\frac{1}{p}\log\frac{1}{\delta}\log n\Big)$, and the runtime is $O\Big(\sum_{r\in[2k_1]} s^r \log\frac{1}{p}\log\frac{1}{\delta}\log^2 n + \frac{k_0k_1}{\delta}\log\frac{1}{p}\log^3 n\Big)$.

Proof. We first note that the claim on the list size follows immediately from the fact that $B = O(s^r)$ entries are added to the list for each t and r, and the loop over t is of length $O\big(\log\frac{1}{p}\big)$.

In order to prove the main claim of the lemma, it suffices to show that for any single value of r, if we replace the loop over r by that single value, then L contains any given $j \in \big[\frac{n}{k_1}\big]$ such that $|Z^r_j|^2 \geq \|Z^r\|_2^2/s^r$, with probability at least $1-p$. Since this essentially corresponds to a standard sparse recovery problem, we switch to simpler notation throughout the proof: We let Y denote a generic signal Z^r, we write its length as $m = n/k_1$, we index its entries in frequency domain as $\widehat{Y}_f$, and we define $k = s^r$.

The proof now consists of two steps. First, we show correctness of the location algorithm assuming that the SEMIEQUIINVERSEBLOCKFFT computation in line 11 computes all the required values for the computation of $\widehat{U}$ in line 19. We then prove that SEMIEQUIINVERSEBLOCKFFT indeed computes all the required values of χ, and conclude with sample complexity and runtime bounds.

Proving correctness of the location process We show that each element f with $|\widehat{Y}_f|^2 \geq \|\widehat{Y}\|_2^2/k$ is reported in a given iteration of the outer loop over $t = 1, \ldots, C_1\log(2/p)$, with probability at least $9/10$. Since the loops use independent randomness, the probability of f not being reported in any of the iterations is bounded by $(1/10)^{C_1\log(2/p)} \leq p/2$ if C_1 is sufficiently large.

Fix an iteration t. We first show that the random set $\mathscr{A}$ chosen in LOCATEREDUCEDSIGNALS has useful error-correcting properties with high probability. Specifically, we let $\mathscr{E}_{\text{balanced}}$ denote the event that for every $\lambda \in [\Lambda], \lambda \neq 0$ at least a fraction $49/100$ of the numbers $\{\omega_\Lambda^{\lambda\cdot\beta}\}_{(\alpha,\beta)\in\mathscr{A}}$ have non-positive real part; in that case, we say that $\mathscr{A}$ is *balanced*. We have for fixed $\lambda \in [\Lambda], \lambda \neq 0$ that since the pair (α,β) was chosen uniformly at random from $[m]\times[m]$, the quantity $\omega_\Lambda^{\lambda\cdot\beta}$

is uniformly distributed on the set of roots of unity of order 2^s for some $s > 0$ (since $\lambda \neq 0$). At least half of these roots have non-positive real part, so for every fixed $\lambda \in [\Lambda], \lambda \neq 0$ one has $\Pr_\beta[\mathrm{Re}(\omega_\Lambda^{\lambda\cdot\beta}) \leq 0] \geq 1/2$. It thus follows by standard concentration inequalities that for every fixed λ at least 49/100 of the numbers $\{\omega_\Lambda^{\lambda\cdot\beta}\}_{(\alpha,\beta)\in\mathscr{A}}$ have non-positive real part with probability at least $1 - e^{-\Omega(|\mathscr{A}|)} = 1 - \exp(-\Omega(C_3 \log\log m))) \geq 1 - 1/(100\log_2 m)$ as long as C_3 is larger than an absolute constant. A union bound over $\Lambda \leq \log_2 m$ values of λ shows that $\Pr[\mathscr{E}_{\text{balanced}}] \geq 1 - (\log_2 m)\cdot/(100\log_2 m) = 1 - 1/100$ for sufficiently large m (recall from Section 1.1 that $\frac{n}{k_1}$ exceeds a large absolute constant by assumption). We henceforth condition on $\mathscr{E}_{\text{balanced}}$.

Fix any f such that $|\widehat{Y}_f|^2 \geq ||\widehat{Y}||_2^2/k$, and let $q = \sigma i$ for convenience. We show by induction on $g = 1, \ldots, \log_\Lambda N$ that before the g-th iteration of lines 23–26 of Algorithm 18, we have that $\mathbf{f}$ coincides with $\mathbf{q}$ on the bottom $g\cdot\log_2\Lambda$ bits, i.e., $\mathbf{f} - \mathbf{q} = 0 \mod \Lambda^{g-1}$.

The **base** of the induction is trivial and is provided by $g = 1$. We now show the **inductive step**. Assume by the inductive hypothesis that $\mathbf{f} - \mathbf{q} = 0 \mod \Lambda^{g-1}$, so that $\mathbf{q} = \mathbf{f} + \Lambda^{g-1}(\lambda_0 + \Lambda\lambda_1 + \Lambda^2\lambda_2 + \ldots)$. Thus, $(\lambda_0, \lambda_1, \ldots)$ is the expansion of $(\mathbf{q} - \mathbf{f})/\Lambda^{g-1}$ base Λ, and λ_0 is the least significant digit. We now show that λ_0 is the unique value of λ that satisfies the condition of line 24 of Algorithm 18, with high probability

In the following, we use the definitions of $\pi(f)$, $h(f)$, and $o_f(f')$ from Definition 1.4.2 with $\Delta = 0$. First, we have for each $a = (\alpha, \beta) \in \mathscr{A}$ and $\mathbf{w} \in \mathbf{W}$ that

$$\begin{aligned}\widehat{H}_{o_f(f)}^{-1}\widehat{U}_{h(f)}(\alpha + \mathbf{w}\cdot\beta) - \widehat{Y}_f\omega_N^{(\alpha+\mathbf{w}\cdot\beta)\mathbf{q}} &= \widehat{H}_{o_f(f)}^{-1}\widehat{U}^*_{h(f)}(\alpha + \mathbf{w}\cdot\beta) - \widehat{Y}_f\omega_N^{(\alpha+\mathbf{w}\cdot\beta)\mathbf{q}} + \mathrm{err}_w \\ &= \widehat{H}_{o_f(f)}^{-1}\sum_{f'\in[m]\setminus\{f\}}\widehat{H}_{o_f(f')}\widehat{Y}_{f'}\omega_N^{\sigma f'\cdot(\alpha+\mathbf{w}\cdot\beta)} + \mathrm{err}_w =: E'(\mathbf{w}),\end{aligned}$$

where $\mathrm{err}_w = \widehat{H}_{o_f(f)}^{-1}(\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)})(\alpha + \mathbf{w}\cdot\beta)$.

And similarly

$$\begin{aligned}\widehat{H}_{o_f(f)}^{-1}\widehat{U}_{h(f)}(\alpha) - \widehat{Y}_f\omega_N^{\alpha\mathbf{q}} &= \widehat{H}_{o_f(f)}^{-1}\widehat{U}^*_{h(f)}(\alpha) - \widehat{Y}_f\omega_N^{\alpha\mathbf{q}} + \mathrm{err} \\ &= \widehat{H}_{o_f(f)}^{-1}\sum_{f'\in[m]\setminus\{f\}}\widehat{H}_{o_f(f')}\widehat{Y}'_{f'}\omega_N^{\sigma f'\cdot\alpha} + \mathrm{err} =: E''.\end{aligned}$$

where $\mathrm{err} = \widehat{H}_{o_f(f)}^{-1}(\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)})(\alpha)$.

We will show that f is recovered from bucket $h(f)$ with high (constant) probability. The bounds above imply that

$$\frac{\widehat{U}_{h(f)}(\alpha + \mathbf{w}\beta))}{\widehat{U}_{h(f)}(\alpha)} = \frac{\widehat{Y}_f\omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q}} + E'(\mathbf{w})}{\widehat{Y}_f\omega_N^{\alpha\mathbf{q}} + E''}. \tag{A.79}$$

The rest of the proof consists of two parts. We first show that with high probability over the

choice of π, the error terms $E'(\mathbf{w})$ and E'' are small in absolute value for most $a = (\alpha, \beta) \in \mathscr{A}$ with extremely high probability. We then use this assumption to argue that f is recovered.

Bounding the error terms $E'(\mathbf{w})$ and E'' (part (i)). We have by Parseval's theorem that

$$\mathbb{E}_a[|E'(\mathbf{w})|^2] \leq \widehat{H}_{o_f(f)}^{-2} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}^2 |Y_{f'}|^2 + |\operatorname{err}_w|^2 + 2|\operatorname{err}_w| \widehat{H}_{o_f(f)}^{-1} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')} |Y_{f'}|, \tag{A.80}$$

and

$$\mathbb{E}_a[|E''|^2] \leq \widehat{H}_{o_f(f)}^{-2} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}^2 |\widehat{Y}_{f'}|^2 + |\operatorname{err}|^2 + 2|\operatorname{err}| \widehat{H}_{o_f(f)}^{-1} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')} |\widehat{Y}_{f'}|,$$

where we used the fact that $\alpha + \mathbf{w}\beta$ is uniformly random in $[m]$ (due to α being uniformly random in $[m]$ and independent of β by definition of $\mathscr{A}$ in line 6 of Algorithm 18).

Taking the expectation of the term $\widehat{H}_{o_f(f)}^{-2} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}^2 |Y_{f'}|^2$ with respect to π, we obtain

$$\mathbb{E}_\pi \left[\widehat{H}_{o_f(f)}^{-2} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}^2 |Y_{f'}|^2 \right] = O(\|Y\|_2^2 / B) = O(||X'||_2^2 / (C_2 k))$$

by Lemma 1.4.3 (note that $F' \geq 2$, so the lemma applies) and the choice $B = C_2 \cdot k$ (line **??** of Algorithm 18). We thus have by Markov's inequality together with the assumption that $|\widehat{Y}_f|^2 \geq ||\widehat{Y}||_2^2 / k$ that

$$\Pr_\pi \left[\widehat{H}_{o_f(f)}^{-2} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}^2 |\widehat{Y}_{f'}|^2 > |\widehat{Y}_f|^2 / 1700 \right] < O(1/C_2) < 1/40$$

and

$$\Pr_\pi \left[\widehat{H}_{o_f(f)}^{-2} \sum_{f' \in [m] \setminus \{f\}} \widehat{H}_{o_f(f')}^2 |\widehat{Y}_{f'}|^2 > |\widehat{Y}_f|^2 / 1700 \right] < O(1/C_2) < 1/40$$

since C_2 is larger than an absolute constant by assumption.

Bounding err **and** err_w **(numerical errors from semi-equispaced FFT computation):** Recall that we have by assumption that $\widehat{\chi}_0$ uniformly distributed over an arbitrarily length-$\frac{\|\widehat{\chi}\|^2}{\operatorname{poly}(n)}$ interval, and that $\widehat{Y} = \widehat{Z}^r$ for some $\widehat{Z}^r$ in the (k_1, δ)-downsampling of $X - \chi$. By decomposing $\widehat{Z}_j^r = ((\widehat{X}^r - \widehat{\chi}^r) \star \widehat{G})_{jk_1}$ into a deterministic part and a random part (in terms of the above-mentioned uniform distribution), we readily obtain for some $c' > 0$ that

$$\|\widehat{Y}\|^2 \geq \frac{\|\widehat{\chi}\|_2^2}{n^{c'}} \tag{A.81}$$

with probability at least $1 - \frac{1}{n^4}$. Since $p \geq \frac{1}{n^3}$ by assumption, we deduce that this also holds with probability at least $1 - p/2$. By the the accuracy of the $\widehat{\chi}$ values stated in Lemma 1.4.5, along with the argument used in (A.31)–(A.32) its proof in Appendix A.3.1 to convert (A.81) to

accuracy on hashed values, we know that $|\widehat{U}_{h(f)} - \widehat{U}^*_{h(f)}| \le \|\widehat{U} - \widehat{U}^*\|_\infty \le n^{-c+1}\|\widehat{\chi}\|_2$. Hence, by using $|\widehat{H}_{o_f(f)}|^{-2} \le 2$ and $\alpha, \alpha + \mathbf{w}\cdot\beta \le m$, we find that

$$|\operatorname{err}| \le 2n^{-c+1}\|\widehat{\chi}\|_2 \le 2n^{-c+c'+1}\|\widehat{Y}\|_2$$
$$|\operatorname{err}_w| \le 2n^{-c+1}\|\widehat{\chi}\|_2 \le 2n^{-c+c'+1}\|\widehat{Y}\|_2,$$

where the second inequality in each equation holds for some $c' > 0$ by (A.81).

Note also that $\widehat{H}^{-1}_{o_f(f)} \sum_{f' \in [m]\setminus\{f\}} \widehat{H}_{o_f(f')}|\widehat{Y}_{f'}| \le 2\|\widehat{Y}\|_1 \le 2\sqrt{m}\|\widehat{Y}\|_2$, since we have $\widehat{H}^{-1}_{o_f(f)} \le 2$ and $\widehat{H}_{f'}| \le 1$ for all f'. We can thus write

$$\begin{aligned}|\operatorname{err}|^2 + 2|\operatorname{err}| \cdot \widehat{H}^{-1}_{o_f(f)} \sum_{f' \in [m]\setminus\{f\}} \widehat{H}_{o_f(f')}|\widehat{Y}_{f'}| &\le |\operatorname{err}|^2 + 4\sqrt{m}|\operatorname{err}| \cdot \|\widehat{Y}\|_2 \\ &\le 4n^{-2c+2c'+2}\|\widehat{Y}\|_2^2 + 8n^{-c+c'+3/2}\|\widehat{Y}\|_2^2 \\ &= n^{\Omega(-c+c')}\|\widehat{Y}\|_2^2, \end{aligned} \tag{A.82}$$

which can thus be made to behave as $\frac{1}{\operatorname{poly}(n)}\|\widehat{Y}\|_2^2$ by a suitable choice of c.

Bounding the error terms $E'(\mathbf{w})$ and E'' (part (ii)). By union bound, we have $|E'(\mathbf{w})|^2 \le |\widehat{Y}_f|^2/1600$ and $|E''|^2 \le |\widehat{Y}_f|^2/1600$ simultaneously with probability at least $1 - 1/20$ – denote the success event by $\mathscr{E}^t_{f,\pi}(\mathbf{w})$. Conditioned on $\mathscr{E}^t_{f,\pi}(\mathbf{w})$, we thus have by (A.80) and (A.82), along with the fact that $\mathscr{A}$ is independent of π, that

$$\mathbb{E}_a[|E'(\mathbf{w})|^2|] \le |Y_f|^2/1600 \quad \text{and} \quad \mathbb{E}_a[|E''|^2] \le |Y_f|^2/1600.$$

Another application of Markov's inequality gives

$$\Pr_a[|E'(\mathbf{w})|^2 \ge |\widehat{Y}_f|^2/40] \le 1/40 \quad \text{and} \quad \Pr_a[|E''|^2 \ge |\widehat{Y}_f|^2/40] \le 1/40.$$

This means that conditioned on $\mathscr{E}^t_{f,\pi}(\mathbf{w})$, with probability at least $1 - e^{-\Omega(|\mathscr{A}|)} \ge 1 - 1/(100\log_2 m)$ over the choice of $\mathscr{A}$, both events occur for all but $2/5$ fraction of $a \in \mathscr{A}$; denote this success event by $\mathscr{E}^t_{f,\mathscr{A}}(\mathbf{w})$. We condition on this event in what follows. Let $\mathscr{A}^*(\mathbf{w}) \subseteq \mathscr{A}$ denote the set of values of $a \in \mathscr{A}$ that satisfy the bounds above.

In particular, we can rewrite (A.79) as

$$\begin{aligned}\frac{\widehat{U}_{h(f)}(\alpha + \mathbf{w}\beta)}{\widehat{U}_{h(f)}(\alpha)} &= \frac{\widehat{Y}_f \omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q}} + E'(\mathbf{w})}{\widehat{Y}_f \omega_N^{\alpha\mathbf{q}} + E''} \\ &= \frac{\omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q}}}{\omega_N^{\alpha\mathbf{q}}} \cdot \xi \quad \left(\text{where } \xi = \frac{1 + \omega^{-(\alpha+\mathbf{w}\beta)\mathbf{q}} E'(\mathbf{w})/\widehat{Y}'_f}{1 + \omega_N^{-\alpha\mathbf{q}} E''/\widehat{Y}'_f}\right) \\ &= \omega_N^{(\alpha+\mathbf{w}\beta)\mathbf{q} - \alpha\mathbf{q}} \cdot \xi = \omega_N^{\mathbf{w}\beta\mathbf{q}} \cdot \xi. \end{aligned}$$

We thus have for $a \in \mathscr{A}^*(\mathbf{w})$ that

$$|E'(\mathbf{w})|/|\widehat{Y}'_f| \le 1/40 \quad \text{and} \quad |E''|/|\widehat{Y}'_f| \le 1/40. \tag{A.83}$$

Showing that $\mathscr{A}^*(\mathbf{w}) \subseteq \mathscr{A}$ suffices for recovery. By the above calculations, we get

$$\frac{\widehat{U}_{h(f)}(\alpha + \mathbf{w}\beta)}{\widehat{U}_{h(f)}(\alpha)} = \omega_N^{\mathbf{w}\beta\mathbf{q}} \cdot \xi = \omega_N^{N\Lambda^{-g}\beta\mathbf{q}} \cdot \xi = \omega_N^{N\Lambda^{-g}\beta\mathbf{q}} + \omega_N^{N\Lambda^{-g}\beta\mathbf{q}}(\xi - 1).$$

We proceed by analyzing the first term, and we will later show that the second term is small. Since $\mathbf{q} = \mathbf{f} + \Lambda^{g-1}(\lambda_0 + \Lambda\lambda_1 + \Lambda^2\lambda_2 + \ldots)$, by the inductive hypothesis, we have

$$\begin{aligned}
\omega_\Lambda^{-\lambda\cdot\beta} \cdot \omega_N^{-N\Lambda^{-g}\mathbf{f}\cdot\beta} \cdot \omega_N^{N\Lambda^{-g}\beta\mathbf{q}} &= \omega_\Lambda^{-\lambda\cdot\beta} \cdot \omega_N^{N\Lambda^{-g}(\mathbf{q}-\mathbf{f})\cdot\beta} \\
&= \omega_\Lambda^{-\lambda\cdot\beta} \cdot \omega_N^{N\Lambda^{-g}(\Lambda^{g-1}(\lambda_0+\Lambda\lambda_1+\Lambda^2\lambda_2+\ldots))\cdot\beta} \\
&= \omega_\Lambda^{-\lambda\cdot\beta} \cdot \omega_N^{(N/\Lambda)\cdot(\lambda_0+\Lambda\lambda_1+\Lambda^2\lambda_2+\ldots)\cdot\beta} \\
&= \omega_\Lambda^{-\lambda\cdot\beta} \cdot \omega_\Lambda^{\lambda_0\cdot\beta} \\
&= \omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta},
\end{aligned}$$

where we used the fact that $\omega_N^{N/\Lambda} = e^{2\pi f(N/\Lambda)/N} = e^{2\pi f/\Lambda} = \omega_\Lambda$. Thus, we have

$$\omega_\Lambda^{-\lambda\cdot\beta}\omega^{-(N\Lambda^{-g}\mathbf{f})\cdot\beta}\frac{\widehat{U}_{h(f)}(\alpha + \mathbf{w}\beta)}{\widehat{U}_{h(f)}(\alpha)} = \omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta}\xi.$$

We now consider two cases. First suppose that $\lambda = \lambda_0$. Then $\omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta} = 1$, and it remains to note that by (A.83) we have $|\xi - 1| \le \frac{1+1/40}{1-1/40} - 1 < 1/3$. Thus, every $a \in \mathscr{A}^*(\mathbf{w})$ passes the test in line 24 of Algorithm 18. Since $|\mathscr{A}^*(\mathbf{w})| \ge (3/5)|\mathscr{A}|$ by the argument above, we have that λ_0 passes the test in line 24. It remains to show that λ_0 is the unique element in $0, \ldots, \Lambda - 1$ that passes this test.

Suppose that $\lambda \ne \lambda_0$. Then, by conditioning on $\mathscr{E}_{\text{balanced}}$, at least a $49/100$ fraction of $\omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta}$ have negative real part. This means that for at least $49/100$ of $a \in \mathscr{A}$, we have

$$|\omega_\Lambda^{(-\lambda+\lambda_0)\cdot\beta}\xi - 1| \ge |\mathbf{i}\cdot|\xi| - 1| \ge |(7/9)\mathbf{i} - 1| > 1/3,$$

and hence the condition in line 16 of Algorithm 18 is not satisfied for any $\lambda \ne \lambda_0$.

We thus get that conditioned on $\mathscr{E}_{\text{balanced}}$ and the intersection of $\mathscr{E}^t_{f,\pi}(\mathbf{w})$ for all $\mathbf{w} \in \mathbf{W}$ and $\mathscr{E}^t_{f,\mathscr{A}}$, recovery succeeds for all values of $g = 1, \ldots, \log_\Lambda N$. By a union bound over the failure events, we get that

$$\Pr\left[\mathscr{E}_{\text{balanced}} \cap \mathscr{E}^t_{f,\mathscr{A}} \cap \left(\bigcap_{\mathbf{w}\in\mathbf{W}} \mathscr{E}^t_{f,\pi}(\mathbf{w})\right)\right] \ge 1 - 1/100 - (\log_\Lambda N) \cdot \frac{1}{100\log_2 m} \ge 98/100.$$

This shows that location is successful for f in a single iteration t with probability at least $98/100 \geq 9/10$, as required.

Sample complexity and runtime.

We first consider the calls to HASHTOBINSREDUCED. This is called for $\log\log m$ values of (α, β) and $\log_\Lambda N = O\big(\frac{\log m}{\log\log m}\big)$ values of g in each iteration, the product of which is $O(\log m) = O(\log n)$. Moreover, the number of iterations is $O\big(\log \frac{1}{p}\big)$. Hence, using Lemma 1.4.6, we find that the combination of all of these calls costs $O\big(F \sum_{r \in [2k_1]} B^r \log \frac{1}{\delta} \log n\big)$ samples, with a runtime of $O\big((B_{\max} F + k_0) k_1 \log^3 n\big)$, where $B_{\max} = O(\max_r s^r)$, and k_0 is such that $\widehat{\chi}$ is $(O(k_0), k_1)$-block sparse. By the assumption $\max_r s^r = O\big(\frac{k_0}{\delta}\big)$, the runtime simplifies to $O\big(\frac{k_0 k_1}{\delta} \log^3 n\big)$

□

Algorithm 18 Location primitive: Given access to the input signal X, a partially recovered signal $\widehat{\chi}$, a budget k and bound on failure probability p, recovers any given $j \in [n/k_1]$ with $|\widehat{Z}^r_j|^2 \geq \|\widehat{Z}^r\|_2^2/k$ for some $r \in [2k_1]$, in the (k_1,δ)-downsampling of $X - \chi$.

procedure LOCATEREDUCEDSIGNALS($X, \widehat{\chi}, n, k_0, k_1, \{s^r\}_{r\in[2k_1]}, \delta, p$)
  ▷ Uses large absolute constants $C_1, C_2, C_3 > 0$
  $B^r \leftarrow C_2 s^r$ for each $r \in [2k_1]$ ▷ Rounded up to a power of two
  $H^r \leftarrow (m, B, F')$-flat filter for each $r \in [2k_1]$, for sufficiently large $F' \geq 2$
  $B_{\max} \leftarrow B^r$
  $\{Z^r_X\}_{r\in[2k_1]} \leftarrow (k_1,\delta)$-downsampling of X ▷ See Definition 1.2.2
  $m \leftarrow n/k_1$
  $L \leftarrow \emptyset$
  for $t = \{1, \ldots, C_1 \log(2/p)\}$ **do**
    $\sigma \leftarrow$ uniformly random odd integer in $[m]$
    $\mathscr{A} \leftarrow C_3 \log\log m$ uniformly random elements in $[m] \times [m]$
    $\Lambda \leftarrow 2^{\lfloor \frac{1}{2}\log_2\log_2 m \rfloor}$, $N \leftarrow \Lambda^{\lceil \log_\Lambda m \rceil}$ ▷ Implicitly extend X to an m-periodic length-N signal
    for $(\alpha,\beta) \in \mathscr{A}$ **do**
      $\mathbf{w} \leftarrow N\Lambda^{-g}$
      $\Delta \leftarrow \alpha + \mathbf{w}\cdot\beta$
      $\mathbf{H} \leftarrow \{H^r\}_{r\in[2k_1]}$
      $\mathbf{B} \leftarrow \{B^r\}_{r\in[2k_1]}$
      $\widehat{U}^r(\alpha + \mathbf{w}\cdot\beta) \leftarrow$ HASHTOBINSREDUCED($\{Z^r_X\}_{r\in[2k_1]}, \widehat{\chi}, \mathbf{H}, n, k_1, \mathbf{B}, \sigma, \Delta$)
    for $r \in [2k_1]$ **do**
      for $b \in [B^r]$ **do** ▷ Loop over all hash buckets
        $\mathbf{f} \leftarrow \mathbf{0}$
        for $g = \{1, \ldots, \log_\Lambda N\}$ **do**
          $\mathbf{w} \leftarrow N\Lambda^{-g}$
          If there exists a unique $\lambda \in \{0, 1, \ldots, \Lambda - 1\}$ such that
          $\left|\omega_\Lambda^{-\lambda\cdot\beta} \cdot \omega^{-(N\cdot\Lambda^{-g}\mathbf{f})\cdot\beta} \cdot \frac{\widehat{U}^r_b(\alpha+\mathbf{w}\cdot\beta)}{\widehat{U}^r_b(\alpha)} - 1\right| < \frac{1}{3}$ for at least $\frac{3}{5}$ fraction of $(\alpha,\beta) \in \mathscr{A}$
          then $\mathbf{f} \leftarrow \mathbf{f} + \Lambda^{g-1}\cdot\lambda$
        $L \leftarrow L \cup \{\sigma^{-1}\mathbf{f}\cdot\frac{m}{N}\}$ ▷ Add recovered element to output list
  return L

B Tight Leverage Scores Characterization of Constrained Signal Classes

B.1 Operator Theory Preliminaries

Throughout the book, we use the term *operator* for linear transformation between two Hilbert spaces. In this section we discuss and prove basic results on operators that we use throughout the book.

B.1.1 Basic definitions and the Loewner partial ordering

Consider two Hilbert spaces $\mathscr{H}_1$ and $\mathscr{H}_2$ with inner products $\langle\cdot,\cdot\rangle_{\mathscr{H}_1}$ and $\langle\cdot,\cdot\rangle_{\mathscr{H}_2}$. We denote by $\mathbb{B}(\mathscr{H}_1,\mathscr{H}_2)$ the set of bounded operators from $\mathscr{H}_1$ to $\mathscr{H}_2$, and abbreviate $\mathbb{B}(\mathscr{H})$ if $\mathscr{H}_1=\mathscr{H}_2=\mathscr{H}$. We denote by $\mathbb{B}_{TC}(\mathscr{H})$ and $\mathbb{B}_{HS}(\mathscr{H})$ the set of trace-class and Hilbert-Schmidt operators (respectively) on $\mathscr{H}$ (i.e. from $\mathscr{H}$ to $\mathscr{H}$). Note that $\mathbb{B}_{TC}(\mathscr{H})\subset\mathbb{B}_{HS}(\mathscr{H})\subset\mathbb{B}(\mathscr{H})$. Recall that for operators, boundedness is equivalent to continuity. The open mapping theorem states that if $\mathscr{A}$ is invertible, then $\mathscr{A}^{-1}$ is bounded. This implies that a compact operator is not invertible. If $\mathscr{A}\in\mathbb{B}(\mathscr{H})$ and $\mathscr{B}\in\mathbb{B}_{TC}(\mathscr{H})$ then $\mathscr{A}\mathscr{B},\mathscr{B}\mathscr{A}\in\mathbb{B}_{TC}(\mathscr{H})$ and $\mathrm{tr}(\mathscr{A}\mathscr{B})=\mathrm{tr}(\mathscr{B}\mathscr{A})$.

We call self-adjoint $\mathscr{A}$ *positive semidefinite* (or simply *positive*) and write $\mathscr{A}\succeq 0$ if $\langle x,\mathscr{A}x\rangle_{\mathscr{H}}\geq 0$ for all $x\in\mathscr{H}$. We write $\mathscr{A}\succnsim 0$ if $\mathscr{A}$ is *positive definite*, i.e. $\langle x,\mathscr{A}x\rangle>0$ for all $x\in\mathscr{H}$. We denote $\mathscr{A}\succ 0$ if $\mathscr{A}$ is *strictly positive*, i.e. there exist a $c>0$ such that $\mathscr{A}\succnsim c\cdot\mathscr{I}_{\mathscr{H}}$ where $\mathscr{I}_{\mathscr{H}}$ is the identity operator on $\mathscr{H}$. Note that for operators on finite dimensional Hilbert spaces, $\mathscr{A}\succnsim 0$ if and only if $\mathscr{A}\succ 0$, but this is not always the case for infinite dimensional Hilbert spaces. The notation for $\mathscr{A}\succeq\mathscr{B}$, $\mathscr{A}\succnsim\mathscr{B}$, and $\mathscr{A}\succ\mathscr{B}$ follow in the standard way.

If $\mathscr{A}\succeq 0$ is self-adjoint and bounded, then it possesses a unique self-adjoint bounded square root $\mathscr{A}^{1/2}\succeq 0$ (Wouk, 1966). Furthermore, if $\mathscr{A}$ is strictly positive then so is $\mathscr{A}^{1/2}$. This implies that if $\mathscr{A}$ is strictly positive and bounded, then $\mathscr{A}$ is bounded below and that the inverse of the square root of $\mathscr{A}$ is $\mathscr{A}^{-1/2}\stackrel{\mathrm{def}}{=}(\mathscr{A}^{-1})^{1/2}$. Lidskii's theorem states the that trace of a trace-class operator is the sum of its eigenvalues.

Many of the following claims are well known of matrices, and the proofs in most cases, but not

all, mirror the matrix case. However, for the operator case we need to be more careful with the conditions due to the aforementioned distinction between $\succsim$ and $\succ$.

Claim B.1.1. *Suppose that $\mathscr{A}$ is a self-adjoint bounded positive semidefinite operator on an Hilbert space $\mathscr{H}$. For every $\epsilon > 0$, the operator $\mathscr{A} + \epsilon \mathscr{I}_{\mathscr{H}}$ is bounded, strictly positive and invertible, and the inverse is bounded.*

Proof. The operator $\mathscr{A} + \epsilon \mathscr{I}_{\mathscr{H}}$ is the sum of two bounded operators, and so it is bounded. It is also clearly bounded below, since $\mathscr{A} + \epsilon \mathscr{I}_{\mathscr{H}} \succeq \epsilon \mathscr{I}_{\mathscr{H}} \succ 0$. A continuous (i.e., bounded) bounded-below operator is invertible, so $\mathscr{A} + \epsilon \mathscr{I}_{\mathscr{H}}$ is invertible. The inverse is bounded due to the open mapping theorem. □

Claim B.1.2. *Suppose that $0 \prec \mathscr{A} \preceq \mathscr{I}_{\mathscr{H}}$ for a self-adjoint operator $\mathscr{A}$. Then, $\mathscr{A}^{-1} \succeq \mathscr{I}_{\mathscr{H}}$.*

Proof. For every $x \in \mathscr{H}$ we have $\langle x, \mathscr{A} x \rangle_{\mathscr{H}} \leq \langle x, x \rangle_{\mathscr{H}}$. Given y, let $x = \mathscr{A}^{-1/2} y$. Then $\langle y, y \rangle_{\mathscr{H}} = \langle \mathscr{A}^{1/2} x, \mathscr{A}^{1/2} x \rangle_{\mathscr{H}} = \langle x, \mathscr{A} x \rangle_{\mathscr{H}} \leq \langle x, x \rangle_{\mathscr{H}} = \langle \mathscr{A}^{-1/2} y, \mathscr{A}^{-1/2} y \rangle_{\mathscr{H}} = \langle y, \mathscr{A}^{-1} y \rangle_{\mathscr{H}}$ so $\mathscr{A}^{-1} \succeq \mathscr{I}_{\mathscr{H}}$. □

Claim B.1.3. *Suppose that $\mathscr{A} \in \mathbb{B}(\mathscr{H})$ and that $\mathscr{B} \succeq 0$ is self-adjoint trace-class operator. Then, $\mathscr{B}^{1/2} \mathscr{A} \mathscr{B}^{1/2}$ is trace-class, and* $\operatorname{tr}(\mathscr{B}^{1/2} \mathscr{A} \mathscr{B}^{1/2}) = \operatorname{tr}(\mathscr{A} \mathscr{B})$.

Proof. Since $\mathscr{B}$ is trace-class, $\mathscr{B}^{1/2} \in \mathbb{B}_{HS}(\mathscr{H})$. This implies that $\mathscr{A} \mathscr{B}^{1/2}$ is also Hilbert-Schmidt. Thus, $\mathscr{B}^{1/2} \mathscr{A} \mathscr{B}^{1/2}$ is the product of two Hilbert-Schmidt operators, so it is trace-class. The trace equality follows from the cyclic property of the trace. □

Claim B.1.4. *Suppose that $\mathscr{A} \succ 0$ is a self-adjoint bounded operator, and that $\mathscr{B} \succeq 0$ is self-adjoint trace-class operator, both on a separable Hilbert space $\mathscr{H}$. Suppose we have* $\operatorname{tr}(\mathscr{A} \mathscr{B}) \leq 1$. *Then, $\mathscr{B} \preceq \mathscr{A}^{-1}$.*

Proof. Due to the cyclicity of the trace $\operatorname{tr}(\mathscr{A}^{1/2} \mathscr{B} \mathscr{A}^{1/2}) \leq 1$. The operator $\mathscr{A}^{1/2} \mathscr{B} \mathscr{A}^{1/2}$ is positive semidefinite, so due to Lidskii's theorem it's largest eigenvalue ≤ 1. For $\mathscr{A}^{1/2} \mathscr{B} \mathscr{A}^{1/2}$, the largest eigenvalue is equal to the operator norm , so for any y,

$$\langle y, \mathscr{A}^{1/2} \mathscr{B} \mathscr{A}^{1/2} y \rangle_{\mathscr{H}} \leq \langle y, y \rangle_{\mathscr{H}}.$$

Since $\mathscr{A}^{1/2}$ is invertible, with inverse $\mathscr{A}^{-1/2}$, the conclusion of the claim follows. □

Claim B.1.5. *Let $\mathscr{A}, \mathscr{B}$ be self-adjoint, bounded, strictly positive operators. If $\mathscr{A} \preceq \mathscr{B}$ then $\mathscr{A}^{-1} \succeq \mathscr{B}^{-1}$.*

Proof. Since $\mathscr{B}$ is bounded and strictly positive, then it is invertible and has an invertible square root. For any $y \in \mathscr{H}$ let $x = \mathscr{B}^{-1/2} y$. We have

$$\begin{aligned} \langle y, \mathscr{B}^{-1/2} \mathscr{A} \mathscr{B}^{-1/2} y \rangle_{\mathscr{H}} &= \langle \mathscr{B}^{-1/2} y, \mathscr{A} \mathscr{B}^{-1/2} y \rangle_{\mathscr{H}} \\ &= \langle x, \mathscr{A} x \rangle_{\mathscr{H}} \\ &\leq \langle x, \mathscr{B} x \rangle_{\mathscr{H}} \\ &= \langle y, y \rangle_{\mathscr{H}}. \end{aligned}$$

So $\mathscr{B}^{-1/2} \mathscr{A} \mathscr{B}^{-1/2} \preceq \mathscr{I}_{\mathscr{H}}$. Since both $\mathscr{A}$ and $\mathscr{B}$ are strictly positive, $\mathscr{B}^{-1/2} \mathscr{A} \mathscr{B}^{-1/2}$ is also strictly positive. Thus, by Claim B.1.2, $\mathscr{B}^{1/2} \mathscr{A}^{-1} \mathscr{B}^{1/2} \succeq \mathscr{I}_{\mathscr{H}}$, from which the claim follows. □

Claim B.1.6. *Suppose that $\mathscr{A} \succ 0$ and $\mathscr{A} \succeq \mathscr{B}$. Then for any $0 \leq c < 1$ we have $\mathscr{A} - c\mathscr{B} \succ 0$.*

Proof. Suppose by contradiction that $\mathscr{A} - c\mathscr{B} \not\succ 0$. Then for any $\epsilon > 0$ there exists an x with unit norm ($\langle x, x \rangle_{\mathscr{H}} = 1$) such that $\langle x, (\mathscr{A} - c\mathscr{B}) x \rangle_{\mathscr{H}} \leq \epsilon$. We have $\langle x, \mathscr{B} x \rangle_{\mathscr{H}} \geq (\langle x, \mathscr{A} x \rangle_{\mathscr{H}} - \epsilon)/c$, and since $\langle x, \mathscr{A} x \rangle_{\mathscr{H}}$ is bounded away from zero and $c < 1$, for small enough ϵ we have $\langle x, \mathscr{B} x \rangle_{\mathscr{H}} > \langle x, \mathscr{A} x \rangle_{\mathscr{H}}$ so $\langle x, (\mathscr{A} - \mathscr{B}) x \rangle_{\mathscr{H}} < 0$ which contradicts the assumption that $\mathscr{A} \succeq \mathscr{B}$. □

Definition B.1.1. Given $x \in \mathscr{H}_1$ and $y \in \mathscr{H}_2$, we define the operator $x \otimes y : \mathscr{H}_2 \to \mathscr{H}_1$ by,

$$(x \otimes y) z \stackrel{\text{def}}{=} \langle y, z \rangle_{\mathscr{H}_2} x.$$

Claim B.1.7. *Let $\mathscr{H}$ be a separable Hilbert space, and assume that $\mathscr{A} \in \mathbb{B}(\mathscr{H})$ and $v \in \mathscr{H}$. Then, $\langle v, \mathscr{A} v \rangle_{\mathscr{H}} = \operatorname{tr}(\mathscr{A}(v \otimes v))$. (We remark that $\mathscr{A}(v \otimes v)$ is trace-class since $v \otimes v$ has finite-rank and $\mathscr{A}$ is bounded.)*

Proof. Let $e_1, e_2, \ldots$ be an orthonormal basis for $\mathscr{H}$. Write $v = \sum_{i=1}^{\infty} \alpha_i e_i$. On one hand we have

$$\begin{aligned} \langle v, \mathscr{A} v \rangle_{\mathscr{H}} &= \left\langle \sum_{i=1}^{\infty} \alpha_i e_i, \mathscr{A} v \right\rangle_{\mathscr{H}} \\ &= \sum_{i=1}^{\infty} \alpha_i^* \langle e_i, \mathscr{A} v \rangle_{\mathscr{H}}. \end{aligned}$$

On the other hand we have,

$$\begin{aligned} \operatorname{tr}(\mathscr{A}(v \otimes v)) &= \sum_{i=1}^{\infty} \langle e_i, \mathscr{A}(v \otimes v) e_i \rangle_{\mathscr{H}} \\ &= \sum_{i=1}^{\infty} \langle e_i, \mathscr{A} \langle v, e_i \rangle_{\mathscr{H}} v \rangle_{\mathscr{H}} \\ &= \sum_{i=1}^{\infty} \langle v, e_i \rangle_{\mathscr{H}} \langle e_i, \mathscr{A} v \rangle_{\mathscr{H}} \\ &= \sum_{i=1}^{\infty} \alpha_i^* \langle e_i, \mathscr{A} v \rangle_{\mathscr{H}}, \end{aligned}$$

so the two terms are equal. □

B.1.2 Weak integrals of operators

We are going to work with operator-valued random variables. To reason about the expected value, we need a notion of an integral of operator-valued functions. We use a generalization of the concept of weak integrals (a.k.a., Pettis integral) of vector-valued functions (Pettis, 1938).

Definition B.1.2. Let $\mathscr{H}_1, \mathscr{H}_2$ be two separable Hilbert spaces, G a measurable space and μ a measure on G, and consider a mapping $\mathscr{A} : G \to \mathbb{B}(\mathscr{H}_1, \mathscr{H}_2)$. If the mapping $(x,z) \mapsto \int_G \langle x, \mathscr{A}(\xi) z\rangle_{\mathscr{H}_2} \, d\mu(\xi)$ is a bounded sesquilinear map in x, z, then we say that $\mathscr{A}$ is a *weakly integrable operator valued function* and the *weak operator integral* is defined to be the unique bounded operator

$$\int_G \mathscr{A}(\xi) \, d\mu(\xi) \in \mathbb{B}(\mathscr{H}_1, \mathscr{H}_2)$$

such that for all x and z,

$$\left\langle x, \left(\int_G \mathscr{A}(\xi) \, d\mu(\xi) \right) z \right\rangle_{\mathscr{H}_2} \stackrel{\text{def}}{=} \int_G \langle x, \mathscr{A}(\xi) z\rangle_{\mathscr{H}_2} \, d\mu(\xi).$$

The existence and uniqueness of such an operator is guaranteed by the Riesz representation theorem for sesquilinear maps (Helmberg, 2008, Page 92, Theorem 5).[1]

Claim B.1.8. *Suppose that $\mathscr{A} : G \to \mathbb{B}(\mathscr{H}_1, \mathscr{H}_2)$ is weakly integrable operator valued function, and $\mathscr{S} \in \mathbb{B}(\mathscr{H}_1), \mathscr{T} \in \mathbb{B}(\mathscr{H}_2)$. Then $\xi \mapsto \mathscr{T}\mathscr{A}(\xi)\mathscr{S}$ is also a weakly integrable operator valued function and*

$$\int_G \mathscr{T}\mathscr{A}(\xi)\mathscr{S} \, d\mu(\xi) = \mathscr{T} \left(\int_G \mathscr{A}(\xi) \, d\mu(\xi) \right) \mathscr{S}.$$

Proof. Recall that $(x,z) \mapsto \int_G \langle x, \mathscr{A}(\xi) z\rangle_{\mathscr{H}_2} \, d\mu(\xi)$ is bounded, so there exists a γ such that for every $x \in \mathscr{H}_2, z \in \mathscr{H}_1$,

$$\left| \int_G \langle x, \mathscr{A}(\xi) z\rangle_{\mathscr{H}_2} \, d\mu(\xi) \right| \leq \gamma \|x\|_{\mathscr{H}_2} \|z\|_{\mathscr{H}_1}$$

We have

$$\begin{aligned} \left| \int_G \langle x, \mathscr{T}\mathscr{A}(\xi)\mathscr{S} z\rangle_{\mathscr{H}_2} d\mu(\xi) \right| &= \left| \int_G \langle \mathscr{T}^* x, \mathscr{A}(\xi)\mathscr{S} z\rangle_{\mathscr{H}_2} d\mu(\xi) \right| \\ &\leq \gamma \|\mathscr{T}^* x\|_{\mathscr{H}_2} \|\mathscr{S} z\|_{\mathscr{H}_1} \leq \gamma \, \|\mathscr{T}\|_{\text{op}} \, \|\mathscr{S}\|_{\text{op}} \, \|x\|_{\mathscr{H}_2} \|z\|_{\mathscr{H}_1}, \end{aligned}$$

where we used the fact that both $\mathscr{S}$ and $\mathscr{T}$ are bounded. Therefore, the mapping $(x,z) \mapsto \int_G \langle x, \mathscr{T}\mathscr{A}(\xi)\mathscr{S} z\rangle_{\mathscr{H}_2} d\mu(\xi)$ is bounded and $\xi \mapsto \mathscr{T}\mathscr{A}(\xi)\mathscr{S}$ is weakly integrable.

[1] We remark that (Helmberg, 2008, Page 92, Theorem 5) is stated and proved only for sesquilinear forms on the same Hilbert space (i.e., $\mathscr{H}_1 = \mathscr{H}_2$). However, it is easy to verify that the result also holds for sesquilinear forms between two Hilbert spaces.

We now show that the value of the integral is $\mathscr{T}\left(\int_G \mathscr{A}(\xi)d\mu(\xi)\right)\mathscr{S}$. For any $x \in \mathscr{H}_2, z \in \mathscr{H}_1$:

$$\left\langle x, \mathscr{T}\left(\int_G \mathscr{A}(\xi)d\mu(\xi)\right)\mathscr{S}z\right\rangle_{\mathscr{H}_2} = \left\langle \mathscr{T}^* x, \left(\int_G \mathscr{A}(\xi)d\mu(\xi)\right)\mathscr{S}z\right\rangle_{\mathscr{H}_2}$$

By definition of $\int_G \mathscr{A}(\xi)d\mu(\xi)$,

$$\left\langle \mathscr{T}^* x, \left(\int_G \mathscr{A}(\xi)d\mu(\xi)\right)\mathscr{S}z\right\rangle_{\mathscr{H}_2} = \int_G \langle \mathscr{T}^* x, \mathscr{A}(\xi)\mathscr{S}z\rangle_{\mathscr{H}_2} d\mu(\xi) = \int_G \langle x, \mathscr{T}\mathscr{A}(\xi)\mathscr{S}z\rangle_{\mathscr{H}_2} d\mu(\xi)$$

so indeed $\int_G \mathscr{T}\mathscr{A}(\xi)\mathscr{S}d\mu(\xi) = \mathscr{T}\left(\int_G \mathscr{A}(\xi)d\mu(\xi)\right)\mathscr{S}$. □

Claim B.1.9. *Let ρ, μ be two, possibly different, probability measures, on $\mathbb{R}$, and let $\mathscr{A} \in \mathbb{B}(L_2(\rho))$ be self-adjoint and positive semi-definite, and let $\mathscr{B} \in \mathbb{B}_{TC}(L_2(\rho))$. Assume that there exists an orthonormal basis for $L_2(\rho)$ consisting of eigenvectors of $\mathscr{A}$. Given a mapping $\eta \in \mathbb{R} \mapsto v_\eta \in L_2(\rho)$ such that $\mathscr{B} = \int_{\mathbb{R}}(v_\eta \otimes v_\eta)d\mu(\eta)$ we have:*

$$\int_{\mathbb{R}} \langle v_\eta, \mathscr{A} v_\eta\rangle_\rho d\mu(\eta) = \mathrm{tr}(\mathscr{A}\mathscr{B})$$

Proof. Let $e_1, e_2, \ldots$ be an orthonormal basis for $L_2(\rho)$ consisting of eigenvectors of $\mathscr{A}$. Using Claim B.1.7, we have

$$\begin{aligned}
\int_{\mathbb{R}} \langle v_\eta, \mathscr{A} v_\eta\rangle_\rho d\mu(\eta) &= \int_{\mathbb{R}} \mathrm{tr}(\mathscr{A}(v_\eta \otimes v_\eta))d\mu(\eta) \\
&= \int_{\mathbb{R}} \sum_{i=1}^{\infty} \langle e_i, \mathscr{A}(v_\eta \otimes v_\eta)e_i\rangle_\rho d\mu(\eta) \\
&= \sum_{i=1}^{\infty} \int_{\mathbb{R}} \langle e_i, \mathscr{A}(v_\eta \otimes v_\eta)e_i\rangle_\rho d\mu(\eta) \\
&= \sum_{i=1}^{\infty} \langle e_i, \int_{\mathbb{R}} \mathscr{A}(v_\eta \otimes v_\eta)d\mu(\eta)e_i\rangle_\rho \\
&= \sum_{i=1}^{\infty} \langle e_i, \mathscr{A}\int_{\mathbb{R}} (v_\eta \otimes v_\eta)d\mu(\eta)e_i\rangle_\rho \\
&= \sum_{i=1}^{\infty} \langle e_i, \mathscr{A}\mathscr{B}e_i\rangle_\mu \\
&= \mathrm{tr}(\mathscr{A}\mathscr{B})
\end{aligned}$$

where the exchange of the integral and infinite sum in the third equality is justified by Tonelli's Theorem. In order to apply Tonelli's theorem we need to show that $\langle e_i, \mathscr{A}(v_\eta \otimes v_\eta)e_i\rangle_\rho \geq 0$ for every i and η. This is indeed the case since $\langle e_i, \mathscr{A}(v_\eta \otimes v_\eta)e_i\rangle_\rho = \langle \mathscr{A}e_i, (v_\eta \otimes v_\eta)e_i\rangle_\rho = \lambda_i\langle e_i, (v_\eta \otimes v_\eta)e_i\rangle_\rho \geq 0$ where λ_i is the eigenvalue corresponding to e_i. Note that since $\mathscr{A}$ is self-adjoint and positive semi-definite, λ_i is real and non-negative. We also used the immediate fact that $v_\eta \otimes v_\eta$ is positive semi-definite. □

Remark: One way to guarantee that there exists an orthonormal basis of eigenvectors for $\mathscr{A}$ is to require $\mathscr{A}$ to be compact. However, it is quite possible for $\mathscr{A}$ not to be compact, and still have an orthonormal basis of eigenvectors. In fact, we primarily apply Claim B.1.9 to operators of the form $(\mathscr{C}+\epsilon\mathscr{I})^{-1}$ where $\mathscr{C}$ is compact, and such operators have an orthonormal basis of eigenvectors (since they share eigenvectors with $\mathscr{C}$).

We say that a weakly integrable $\mathscr{A}(\cdot)$ is *self-adjoint* if $\mathscr{A}(\xi)$ is self-adjoint for all ξ. It is easy to verify that if $\mathscr{A}(\cdot)$ is self-adjoint, then $\int_G A(\xi)d\mu(\xi)$ is self-adjoint as well.

Claim B.1.10. *Suppose that $\mathscr{A},\mathscr{B}: G \to \mathbb{B}(\mathscr{H})$ are two self-adjoint weakly integrable operator valued functions. If, with respect to a measure μ on G, $\mathscr{A}(\xi) \preceq \mathscr{B}(\xi)$ almost everywhere, then $\int_G \mathscr{A}(\xi)d\mu(\xi) \preceq \int_G \mathscr{B}(\xi)d\mu(\xi)$.*

Proof. For every $x \in \mathscr{H}$,

$$\left\langle x, \int_G \mathscr{A}(\xi)d\mu(\xi)x \right\rangle_{\mathscr{H}} = \int_G \langle x, \mathscr{A}(\xi)x\rangle_{\mathscr{H}} d\mu(\xi) \leq \int_G \langle x, \mathscr{B}(\xi)x\rangle_{\mathscr{H}} d\mu(\xi) = \left\langle x, \int_G \mathscr{B}(\xi)d\mu(\xi)x \right\rangle_{\mathscr{H}}$$

so indeed $\int_G \mathscr{A}(\xi)d\mu(\xi) \preceq \int_G \mathscr{B}(\xi)d\mu(\xi)$. □

Claim B.1.11. *Suppose that $\mathscr{B}: G \to \mathbb{B}(\mathscr{H})$ is a self-adjoint weakly integrable operator valued function. Consider another self-adjoint operator valued function $\mathscr{A}: G \to \mathbb{B}(\mathscr{H})$. If for every $\xi \in G$ we have $0 \preceq \mathscr{A}(\xi) \preceq \mathscr{B}(\xi)$, then $\mathscr{A}$ is weakly integrable and $\int_G \mathscr{A}(\xi)d\mu(\xi) \preceq \int_G \mathscr{B}(\xi)d\mu(\xi)$.*

Proof. We need to prove only that $\mathscr{A}$ is weakly integrable, since the integral bound follows from Claim B.1.10. A sesquilinear form is bounded if and only if the associated quadratic form is bounded (Helmberg, 2008, Page 92, Theorem 3), so we need to show that the integral of the quadratic form associated with $\mathscr{A}$ is bounded. Since $\mathscr{A}(\xi)$ is positive semidefinite for every $\xi \in G$, for any x

$$\left|\int_G \langle x, \mathscr{A}(\xi)x\rangle_{\mathscr{H}} d\mu(\xi)\right| = \int_G \langle x, \mathscr{A}(\xi)x\rangle_{\mathscr{H}} d\mu(\xi) \leq \int_G \langle x, \mathscr{B}(\xi)z\rangle_{\mathscr{H}} d\mu(\xi) = \left|\int_G \langle x, \mathscr{B}(\xi)x\rangle_{\mathscr{H}} d\mu(\xi)\right|$$

and since the integral of the quadratic form associated with $\mathscr{B}$ is bounded (since $\mathscr{B}$ is weakly integrable) we conclude that integral quadratic form associated with $\mathscr{A}$ is bounded, so indeed $\mathscr{A}$ is weakly integrable. □

B.1.3 Concentration of random operators

Let $\mathscr{A}: G \to \mathbb{B}(\mathscr{H})$ be a weakly integrable operator valued function. If the underlying measure μ is a probability measure, then we shall call $\mathscr{A}$ a *random operator*, and write

$$\mathbb{E}[\mathscr{A}] = \int_G \mathscr{A}(\xi)d\mu(\xi).$$

Certain matrix concentration results can be generalized to the case that $\mathscr{A}$ is a random operator which takes only self-adjoint Hilbert-Schmidt values. The underlying reason is that Hilbert-Schmidt operators can be well-approximated using finite rank operators. The basic technique is outlined in (Minsker, 2017, Section 3.2). We use this technique to prove the following lemma.

Lemma B.1.1. *Suppose that $\mathscr{H}$ is a separable Hilbert space, and $\mathscr{B}$ is a fixed self-adjoint Hilbert-Schmidt operator on $\mathscr{H}$. Let $\mathscr{R}$ be a self-adjoint Hilber-Schmidt random operator satisfying*

$$\mathbb{E}[\mathscr{R}] = \mathscr{B} \quad and \quad \|\mathscr{R}\|_{\text{op}} \leq L.$$

Let $\mathscr{M}$ be another self-adjoint trace-class operator such that $\mathbb{E}[\mathscr{R}^2] \preceq \mathscr{M}$. Form the operator sampling estimator

$$\bar{\mathscr{R}}_n = \frac{1}{n}\sum_{k=1}^{n} \mathscr{R}_k,$$

where each $\mathscr{R}_k$ is an independent copy of $\mathscr{R}$. Then, for all $t > \sqrt{\|\mathscr{M}\|_{\text{op}}/n} + 2L/3n$,

$$\Pr\left(\left\|\bar{\mathscr{R}}_n - \mathscr{B}\right\|_{\text{op}} > t\right) \leq \frac{8\,\text{tr}(\mathscr{M})}{\|\mathscr{M}\|_{\text{op}}} \exp\left(\frac{-nt^2/2}{\|\mathscr{M}\|_{\text{op}} + 2Lt/3}\right). \tag{B.1}$$

Proof. Let $e_1, e_2, \ldots$ be the eigenvectors of $\mathscr{M}$, ordered according to the magnitude of the corresponding eigenvalue, and let $\mathscr{P}_j$ be the orthogonal projector on the span of $e_1, e_2, \ldots, e_j$. Consider the finite-rank operators $\mathscr{R}^{(j)} = \mathscr{P}_j\mathscr{R}\mathscr{P}_j$, $\mathscr{R}_k^{(j)} = \mathscr{P}_j\mathscr{R}_k\mathscr{P}_j$, $\bar{\mathscr{R}}_n^{(j)} = \mathscr{P}_j\bar{\mathscr{R}}_n\mathscr{P}_j$, $\mathscr{B}^{(j)} = \mathscr{P}_j\mathscr{B}\mathscr{P}_j$ and $\mathscr{M}^{(j)} = \mathscr{P}_j\mathscr{M}\mathscr{P}_j$. We will apply on these operator sequences the matrix version of the current lemma (Avron et al., 2017c)[2]

Due to linearity of weak operator integrals we have $\mathbb{E}(\mathscr{R}^{(j)}) = \mathscr{P}_j\mathscr{B}^{(j)}\mathscr{P}_j$. We can bound the operator norm of $\mathscr{R}^{(j)}$: $\left\|\mathscr{R}^{(j)}\right\|_{\text{op}} \leq \left\|\mathscr{P}_j\mathscr{R}\mathscr{P}_j\right\|_{\text{op}} \leq \left\|\mathscr{P}_j\right\|_{\text{op}}^2 \|\mathscr{R}\|_{\text{op}} \leq L$ since the operator norm of a projection operator is 1. Using the fact that $\mathscr{P}_j \preceq \mathscr{I}_{\mathscr{H}}$ and so $\mathscr{R}\mathscr{P}_j\mathscr{R} \preceq \mathscr{R}^2$ we have

$$\mathbb{E}\left[(\mathscr{R}^{(j)})^2\right] = \mathscr{P}_j\mathbb{E}[\mathscr{R}\mathscr{P}_j\mathscr{R}]\mathscr{P}_j \preceq \mathscr{P}_j\mathbb{E}[\mathscr{R}^2]\mathscr{P}_j \preceq \mathscr{M}^{(j)}$$

Now applying the aforementioned matrix version of the current lemma[3] we find that

$$\Pr\left(\left\|\bar{\mathscr{R}}_n^{(j)} - \mathscr{B}^{(j)}\right\|_{\text{op}} \geq t\right) \leq \frac{8\,\text{tr}(\mathscr{M}^{(j)})}{\left\|\mathscr{M}^{(j)}\right\|_{\text{op}}} \exp\left(\frac{-nt^2/2}{\left\|\mathscr{M}^{(j)}\right\|_{\text{op}} + 2Lt/3}\right). \tag{B.2}$$

Due to the way we constructed $\mathscr{P}_j$, and $\mathscr{M}$ being trace-class, we have $\text{tr}(\mathscr{M}^{(j)}) \to \text{tr}(\mathscr{M})$ as $j \to \infty$. Furthermore, since $\mathscr{M}$ is trace-class, $\mathscr{P}_j\mathscr{M} \to \mathscr{M}$ uniformly (Hunter and Nachtergaele,

[2] The lemma in Avron et al. (2017c) is stated as a bound on $\Pr\left(\left\|\bar{\mathscr{R}}_n - \mathscr{B}\right\|_{\text{op}} \geq t\right)$, while for operators strict inequality is necessary. It is easy to verify that the matrix version of the Lemma continues to hold for $\Pr\left(\left\|\bar{\mathscr{R}}_n - \mathscr{B}\right\|_{\text{op}} > t\right)$.

[3] Technically, the aforementioned concentration result is for *matrices*, while here we deal with abstract operators on finite dimensional subspaces. We can address this issue by using the corresponding transformation matrices, but we find that to be tedious details.

2001, Theorem 9.21), and so $\mathscr{M}^{(j)} \to \mathscr{M}$ implies that $\left\|\mathscr{M}^{(j)}\right\|_{\text{op}} \to \|\mathscr{M}\|_{\text{op}}$. Thus, the entire right side of (B.2) converges to the right side of (B.1), so

$$\liminf_{j\to\infty} \Pr\left(\left\|\bar{\mathscr{R}}_n^{(j)} - \mathscr{B}^{(j)}\right\|_{\text{op}} > t\right) \leq \frac{8\,\text{tr}(\mathscr{M})}{\|\mathscr{M}\|_{\text{op}}} \exp\left(\frac{-nt^2/2}{\|\mathscr{M}\|_{\text{op}} + 2Lt/3}\right).$$

Let G and μ denote the underlying probability space and probability measure. Let f_j now denote the indicator function for the event $\left\|\bar{\mathscr{R}}_n^{(j)} - \mathscr{B}^{(j)}\right\|_{\text{op}} > t$, and f the indicator for the event $\left\|\bar{\mathscr{R}}_n - \mathscr{B}\right\|_{\text{op}} > t$. Again, due to the fact that $\bar{\mathscr{R}}_n - \mathscr{B}$ is Hilbert-Schmidt we have $\bar{\mathscr{R}}_n^{(j)} - \mathscr{B}^{(j)} \to \bar{\mathscr{R}}_n - \mathscr{B}$, implies that for any $\xi \in G$, $f(\xi) = \liminf_{j\to\infty} f_j(\xi)$. Now due to Fatou's lemma:

$$\begin{aligned}
\Pr\left(\left\|\bar{\mathscr{R}}_n - \mathscr{B}\right\|_{\text{op}} > t\right) &= \int_G f(\xi)\,d\mu(\xi) \\
&= \int_G \liminf_{j\to\infty} f_j(\xi)\,d\mu(\xi) \\
&\leq \liminf_{j\to\infty} \int_G f_j(\xi)\,d\mu(\xi) \\
&= \liminf_{j\to\infty} \Pr\left(\left\|\bar{\mathscr{R}}_n^{(j)} - \mathscr{B}^{(j)}\right\|_{\text{op}} > t\right) \\
&\leq \frac{8\,\text{tr}(\mathscr{M})}{\|\mathscr{M}\|_{\text{op}}} \exp\left(\frac{-nt^2/2}{\|\mathscr{M}\|_{\text{op}} + 2Lt/3}\right).
\end{aligned}$$

□

B.2 Properties of the Ridge Leverage Scores

B.2.1 Basic facts about leverage scores

In this section we prove Theorem 3.4.1, giving fundamental properties of the ridge leverage scores that we use both in bounding these scores and proving that leverage score sampling can be used to solve the regularized regression problem of (3.10) (and hence Problem (3.2.1)).

Theorem 3.4.1 (Leverage Function Properties)**.** *Let $\tau_{\mu,\epsilon}(t)$ be the ridge leverage function of Definition 3.4.1, that is*

$$\tau_{\mu,\epsilon}(t) = \frac{1}{T} \cdot \max_{\{\alpha \in L_2(\mu):\, \|\alpha\|_\mu > 0\}} \frac{|[\mathscr{F}_\mu^* \alpha](t)|^2}{\left\|\mathscr{F}_\mu^* \alpha\right\|_T^2 + \epsilon \|\alpha\|_\mu^2}, \tag{B.3}$$

and let $\varphi_t \in L_2(\mu)$ be defined by $\varphi_t(\xi) = e^{-2\pi i t\xi}$, we have the following basic properties:

- *The leverage scores integrate to the statistical dimension:*

$$\int_0^T \tau_{\mu,\epsilon}(t)\,dt = s_{\mu,\epsilon} \overset{\text{def}}{=} \operatorname{tr}\left(\mathcal{K}_\mu(\mathcal{K}_\mu + \epsilon\mathcal{I}_T)^{-1}\right). \tag{B.4}$$

- *Inner Product characterization:*

$$\tau_{\mu,\epsilon}(t) = \frac{1}{T}\cdot\langle\varphi_t, (\mathcal{G}_\mu + \epsilon\mathcal{I}_\mu)^{-1}\varphi_t\rangle_\mu. \tag{B.5}$$

- *Minimization Characterization:*

$$\tau_{\mu,\epsilon}(t) = \frac{1}{T}\cdot\min_{\beta\in L_2(T)} \frac{\left\|\mathcal{F}_\mu\beta - \varphi_t\right\|_\mu^2}{\epsilon} + \left\|\beta\right\|_T^2. \tag{B.6}$$

Proof. Recall, that given $t\in[0,T]$, $\varphi_t(\xi) \overset{\text{def}}{=} e^{-2\pi i t\xi}$ ($\varphi_t \in L_2(\mu)$). It is easy to verify that:

$$\mathcal{G}_\mu = \frac{1}{T}\int_0^T (\varphi_t\otimes\varphi_t)\,dt. \tag{B.7}$$

To prove the equality between Equations (B.3), (B.5), and (B.6), we first show that the right hand side of (B.5) is equal to the right hand side of (B.6) and then show that the right hand side of (B.5) is equal to the right hand side of (B.3).

First, we need an auxiliary lemma regarding the solution of regularized least squares problems. If $\mathcal{A}$ is matrix with full column rank or a one-to-one linear operator between finite-dimensional Hilbert spaces, and b some vector, then $F(x) = \|\mathcal{A}x - b\|^2$ has a unique minimizer. In infinite dimension spaces, this remains true if only the co-domain of $\mathcal{A}$ is infinite dimensional. However, if both the domain and co-domain are infinite dimensional there might not be a minimizer even if the $\mathcal{A}$ is bounded: the range of $\mathcal{A}$ might not be closed, so it is possible that $\|\mathcal{A}x - b\| > 0$ for every x, but also that there exists a series $\{x_n\}$ such that $\|\mathcal{A}x_n - b\| \to 0$ as $n\to\infty$. However, once we introduce a ridge term (i.e., minimize $F(x) = \|\mathcal{A}x - b\|^2 + \lambda\|x\|^2$ for some $\lambda > 0$) there is always a unique minimizer (as long as $\mathcal{A}$ is bounded), due to the extreme value theorem (since we can bound the search domain). Furthermore, we can write an analytic expression for the minimizer in an analogous way to the finite dimensional case, as the following lemma shows.

Lemma B.2.1 (Regularized Least Squares Minimizer)**.** *Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two Hilbert spaces, and $\mathcal{A}:\mathcal{H}_1\to\mathcal{H}_2$ be a bounded linear operator. Let $b\in\mathcal{H}_2$ and $\lambda > 0$. The function*

$$F(x) = \|\mathcal{A}x - b\|_{\mathcal{H}_2}^2 + \lambda\|x\|_{\mathcal{H}_1}^2$$

has a unique minimizer which is $x^\star = \mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda\mathcal{I}_{\mathcal{H}_2})^{-1}b$.

Proof. Consider the Hilbert space $\mathcal{H}_1 \times \mathcal{H}_2$ equipped with the inner product

$$\langle (a_1, a_2), (b_1, b_2) \rangle_{\mathcal{H}_1 \times \mathcal{H}_2} \stackrel{\text{def}}{=} \langle a_1, b_1 \rangle_{\mathcal{H}_1} + \langle a_2, b_2 \rangle_{\mathcal{H}_2}.$$

Define the operator $\mathcal{T} : \mathcal{H}_1 \to \mathcal{H}_1 \times \mathcal{H}_2, \quad \mathcal{T}(x) = (\sqrt{\lambda}x, \mathcal{A}x)$. Let $y = (0, b) \in \mathcal{H}_1 \times \mathcal{H}_2$. We have $F(x) = \left\| \mathcal{T}x - y \right\|^2_{\mathcal{H}_1 \times \mathcal{H}_2}$. Thus, we need to show that there is a unique point $\tilde{y} \in \text{range}(\mathcal{T})$ that minimizes $\left\| \tilde{y} - y \right\|^2_{\mathcal{H}_1 \times \mathcal{H}_2}$ and that $\tilde{y} = \mathcal{T}x^\star$ for $x^\star = \mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda \mathcal{I}_{\mathcal{H}_2})^{-1} b$.

The operator $\mathcal{T}$ is a bounded linear operator, so it is continuous. We also have that for every $x \in \mathcal{H}_1$, $\|\mathcal{T}x\|^2_{\mathcal{H}_1 \times \mathcal{H}_2} \geq \lambda \|x\|^2_{\mathcal{H}_1}$ where $\lambda > 0$, so $\mathcal{T}$ is bounded from below. So $\mathcal{T}$ has a closed range (Abramovich et al., 2002, Theorem 2.5). Thus, there is a unique $\tilde{y} \in \text{range}(\mathcal{T})$ that minimizes $\left\| \tilde{y} - y \right\|^2_{\mathcal{H}_1 \times \mathcal{H}_2}$, and that $\tilde{y}$ is the unique element of $\text{range}(\mathcal{T})$ with the property $y - \tilde{y} \perp \text{range}(\mathcal{T})$ (Hunter and Nachtergaele, 2001, Theorem 6.13). So it suffices to show that for every $x \in \mathcal{H}_1$ we have $y - \mathcal{T}x^\star \perp \mathcal{T}x$. We compute:

$$\begin{aligned}
&\langle y - \mathcal{T}x^\star, \mathcal{T}x \rangle_{\mathcal{H}_1 \times \mathcal{H}_2} \\
&\quad = \langle (-\sqrt{\lambda}\mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda \mathcal{I}_{\mathcal{H}_2})^{-1} b, b - \mathcal{A}\mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda \mathcal{I}_{\mathcal{H}_2})^{-1} b), (\sqrt{\lambda}x, \mathcal{A}x) \rangle_{\mathcal{H}_1 \times \mathcal{H}_2} \\
&\quad = \langle (-\sqrt{\lambda}\mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda \mathcal{I}_{\mathcal{H}_2})^{-1} b, \lambda(\mathcal{A}\mathcal{A}^* + \lambda \mathcal{I}_{\mathcal{H}_2})^{-1} b), (\sqrt{\lambda}x, \mathcal{A}x) \rangle_{\mathcal{H}_1 \times \mathcal{H}_2} \\
&\quad = -\lambda \langle \mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda \mathcal{I}_{\mathcal{H}_2})^{-1} b, x \rangle_{\mathcal{H}_1} + \lambda \langle (\mathcal{A}\mathcal{A}^* + \lambda \mathcal{I}_{\mathcal{H}_2})^{-1} b, \mathcal{A}x \rangle_{\mathcal{H}_2} = 0.
\end{aligned}$$

So indeed, for every $x \in \mathcal{H}_1$ we have $y - \mathcal{T}x^\star \perp \mathcal{T}x$ and $x^\star$ is the unique minimizer. □

Using Lemma B.2.1 we now proceed with the proof of Theorem 3.4.1.

Corollary B.2.1. *Let*

$$\beta^\star = \mathcal{F}_\mu^* (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t.$$

Then,

$$\frac{1}{T} \cdot \left(\frac{\left\| \mathcal{F}_\mu \beta^\star - \varphi_t \right\|^2_\mu}{\epsilon} + \left\| \beta^\star \right\|^2_T \right) = \frac{1}{T} \cdot \min_{\beta \in L_2(T)} \frac{\left\| \mathcal{F}_\mu \beta - \varphi_t \right\|^2_\mu}{\epsilon} + \left\| \beta \right\|^2_T.$$

Claim B.2.1. *We have*

$$\langle \varphi_t, (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t \rangle_\mu = \frac{\left\| \mathcal{F}_\mu \beta^\star - \varphi_t \right\|^2_\mu}{\epsilon} + \left\| \beta^\star \right\|^2_T$$

so the right hand side of (B.5) *is equal to the right hand side of* (B.6).

Proof. We compute:

$$\begin{aligned}
\left\| \beta^\star \right\|^2_T &= \langle \mathcal{F}_\mu^* (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t, \mathcal{F}_\mu^* (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t \rangle_\mu \\
&= \langle (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t, \mathcal{G}_\mu (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t \rangle_\mu \\
&= \langle (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t, (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu - \epsilon \mathcal{I}_\mu)(\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t \rangle_\mu \\
&= \langle \varphi_t, (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-1} \varphi_t \rangle_\mu - \epsilon \langle \varphi_t, (\mathcal{G}_\mu + \epsilon \mathcal{I}_\mu)^{-2} \varphi_t \rangle_\mu,
\end{aligned}$$

and,

$$\begin{aligned}
\left\|\mathscr{F}_\mu\beta^\star-\varphi_t\right\|_\mu^2 &= \left\|\mathscr{F}_\mu\mathscr{F}_\mu^*(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}\varphi_t-\varphi_t\right\|_\mu^2\\
&= \left\|\left(\mathscr{G}_\mu(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}-\mathscr{I}_\mu\right)\varphi_t\right\|_\mu^2\\
&= \left\|\left((\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu-\epsilon\mathscr{I}_\mu)(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}-\mathscr{I}_\mu\right)\varphi_t\right\|_\mu^2\\
&= \left\|\epsilon(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}\varphi_t\right\|_\mu^2\\
&= \epsilon^2\langle\varphi_t,(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-2}\varphi_t\rangle_\mu
\end{aligned}$$

Summing the last equalities completes the proof. □

Claim B.2.2. *We have*

$$\langle\varphi_t,(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}\varphi_t\rangle_\mu=\max_{\{\alpha\in L_2(\mu):\|\alpha\|_\mu>0\}}\frac{|[\mathscr{F}_\mu^*\alpha](t)|^2}{\left\|\mathscr{F}_\mu^*\alpha\right\|_T^2+\epsilon\|\alpha\|_\mu^2}$$

so the right hand side of (B.5) *is equal to the right hand side of* (B.3).

Proof. We can reformulate the previous claim as :

$$\begin{aligned}
\langle\varphi_t,(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}\varphi_t\rangle_\mu=\text{minimum}\quad & \|u\|_\mu^2+\left\|\beta\right\|_T^2\\
\beta\in L_2(\mu);\quad & u\in L_2(T)\\
\text{subject to:}\quad & \mathscr{F}_\mu\beta+\sqrt{\epsilon}u=\varphi_t.
\end{aligned}$$

Let the optimal solution be $\beta^\star$ and $u^\star$. We have $\varphi_t=\mathscr{F}_\mu\beta^\star+\sqrt{\epsilon}u^\star$, hence for any $0\neq\alpha\in L_2(\mu)$:

$$\begin{aligned}
|[\mathscr{F}_\mu^*\alpha](t)| &= |\langle\varphi_t,\alpha\rangle_\mu|\\
&= |\langle\alpha,\varphi_t\rangle_\mu|\\
&= |\langle\alpha,\mathscr{F}_\mu\beta^\star+\sqrt{\epsilon}u^\star\rangle_\mu|\\
&\leq |\langle\alpha,\mathscr{F}_\mu\beta^\star\rangle_\mu|+|\langle\alpha,\sqrt{\epsilon}u^\star\rangle_\mu|\\
&= |\langle\mathscr{F}_\mu^*\alpha,\beta^\star\rangle_T|+|\langle\alpha,\sqrt{\epsilon}u^\star\rangle_\mu|\\
&\leq \left\|(\mathscr{F}_\mu^*\alpha)\right\|_T\cdot\left\|\beta^\star\right\|_T+\sqrt{\epsilon}\|\alpha\|_\mu\cdot\left\|u^\star\right\|_\mu,
\end{aligned}$$

where the last inequality follows from Cauchy-Schwarz inequality. Using Cauchy-Schwarz again:

$$\begin{aligned}
|[\mathscr{F}_\mu^*\alpha](t)|^2 &\leq \left(\left\|(\mathscr{F}_\mu^*\alpha)\right\|_T\cdot\left\|\beta^\star\right\|_T+\sqrt{\epsilon}\|\alpha\|_\mu\cdot\left\|u^\star\right\|_\mu\right)^2\\
&\leq \left(\left\|\mathscr{F}_\mu^*\alpha\right\|_T^2+\epsilon\|\alpha\|_\mu^2\right)\cdot\left(\left\|\beta^\star\right\|_T^2+\left\|u^\star\right\|_\mu^2\right)
\end{aligned}$$

So for every $0 \neq \alpha \in L_2(\mu)$:

$$\frac{|[\mathscr{F}_\mu^* \alpha](t)|^2}{\left\|\mathscr{F}_\mu^* \alpha\right\|_T^2 + \epsilon \|\alpha\|_\mu^2} \leq \left\|\beta^\star\right\|_T^2 + \left\|u^\star\right\|_\mu^2 = \langle \varphi_t, (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \varphi_t \rangle_\mu$$

We conclude by showing that the maximum value is attained. Let $\alpha^\star = (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \varphi_t$. Then,

$$\left\|\mathscr{F}_\mu^* \alpha^\star\right\|_T^2 + \epsilon \left\|\alpha^\star\right\|_\mu^2 = \langle \alpha^\star, (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu) \alpha^\star \rangle = \langle \varphi_t, (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \varphi_t \rangle_\mu$$

and finally,

$$\frac{|[\mathscr{F}_\mu^* \alpha^\star](t)|^2}{\left\|\mathscr{F}_\mu^* \alpha^\star\right\|_T^2 + \epsilon \|\alpha\|_\mu^2} = \frac{|\langle \varphi_t, \alpha^\star \rangle_\mu|^2}{\langle \varphi_t, (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \varphi_t \rangle_\mu} = \langle \varphi_t, (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \varphi_t \rangle_\mu.$$

□

We now turn to showing that the leverage function integrates to the statistical dimension.

Claim B.2.3.

$$\int_0^T \tau_{\mu,\epsilon}(t)\, dt = s_{\mu,\epsilon}.$$

Proof. It follows from Eq. (B.7) and Claim B.1.9 that $\int_0^T \tau_{\mu,\epsilon}(t) = \mathrm{tr}((\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \mathscr{G}_\mu)$. The claim follows by noting that $\mathscr{K}_\mu$ and $\mathscr{G}_\mu$ have the same eigenvalues (both operators are compact and self-adjoint, so their spectrum consists of only eigenvalues, and it is easy to verify that if x is an eigenvector of $\mathscr{K}_\mu$ then $\mathscr{F}_\mu x$ is eigenvector of $\mathscr{G}_\mu$).

□

We thus have completed the proof of Theorem 3.4.1. □

B.2.2 Operator Approximation via Leverage Score Sampling

Analogs of the following concentration result are well known for matrices. Accordingly, the proof is an adaptation of standard proofs for finite matrix approximation by leverage score sampling, where matrix concentration results are replaced with operator concentration results. A similar strategy was employed in Bach (2017).

Lemma B.2.2. *Consider the preconditions of Theorem 3.4.2, and denote* $\widehat{\mathscr{G}}_\mu = \mathbf{F}\mathbf{F}^*$. *Let* $\Delta \leq 1/2$ *and* $\epsilon \leq \left\|\mathscr{G}_\mu\right\|_{\mathrm{op}}$. *If* $s \geq \frac{8}{3}\Delta^{-2} \tilde{s}_{\mu,\epsilon} \ln(16 \tilde{s}_{\mu,\epsilon}^2/\delta)$, *then*

$$(1-\Delta)(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu) \preceq \widehat{\mathscr{G}}_\mu + \epsilon \mathscr{I}_\mu \preceq (1+\Delta)(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu) \tag{B.8}$$

with probability of at least $1-\delta$.

Proof. The condition (B.8) is equivalent to

$$\mathscr{G}_\mu - \Delta(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu) \preceq \widehat{\mathscr{G}}_\mu \preceq \mathscr{G}_\mu + \Delta(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)$$

By composing with $(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}$ on the left and right, we find that the condition is equivalent to

$$\left\|(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}(\widehat{\mathscr{G}}_\mu - \mathscr{G}_\mu)(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}\right\|_{\mathrm{op}} \le \Delta. \tag{B.9}$$

Noticing that $\mathbf{F}g = \sum_{j=1}^s w_j g(j)\varphi_{t_j}$ and that $[\mathbf{F}^* z](j) = w_j\langle\varphi_{t_j}, z\rangle_\mu$, we understand that $\widehat{\mathscr{G}}_\mu = \sum_{j=1}^s w_j^2(\varphi_{t_j}\otimes\varphi_{t_j})$. Let,

$$\mathscr{X}_j \overset{\text{def}}{=} s w_j^2(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}(\varphi_{t_j}\otimes\varphi_{t_j})(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}.$$

Note that $\mathscr{X}_j$ is self-adjoint and Hilbert-Schmidt (since it has finite rank). We have,

$$(\mathscr{G}_\mu + \mathscr{I}_\mu)^{-1/2}\widehat{\mathscr{G}}_\mu(\mathscr{G}_\mu + \mathscr{I}_\mu)^{-1/2} = \frac{1}{s}\sum_{j=1}^s \mathscr{X}_j.$$

Since the time samples are drawn randomly, $\mathscr{X}_1,\dots,\mathscr{X}_s$ are i.i.d. random operators. We also have, using Claim B.1.8,

$$\mathbb{E}_{t_j\propto\tilde\tau_{\mu,\epsilon}}[\mathscr{X}_j] = (\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}\mathbb{E}_{t_j\propto\tilde\tau_{\mu,\epsilon}}\left[s w_j^2(\varphi_{t_j}\otimes\varphi_{t_j})\right](\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}.$$

Write $w(t) = \sqrt{\frac{\tilde s_{\mu,\epsilon}}{T\cdot\tilde\tau_{\mu,\epsilon}(t)}}$. For every $x, z\in L_2(\mu)$,

$$\int_0^T \left\langle x, w(t)^2\cdot(\varphi_t\otimes\varphi_t)z\right\rangle_\mu \cdot(\tilde\tau_{\mu,\epsilon}(t)/\tilde s_{\mu,\epsilon})\,dt = \frac{1}{T}\int_0^T \left\langle x,(\varphi_t\otimes\varphi_t)z\right\rangle_\mu dt = \langle x,\mathscr{G}_\mu z\rangle_\mu$$

which shows that $\mathbb{E}_{t_j\propto\tilde\tau_{\mu,\epsilon}}\left[s w_j^2(\varphi_{t_j}\otimes\varphi_{t_j})\right] = \mathscr{G}_\mu$. Therefore,

$$\mathbb{E}_{t_j\propto\tilde\tau_{\mu,\epsilon}}[\mathscr{X}_j] = (\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}\mathscr{G}_\mu(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1/2}. \tag{B.10}$$

Next, we bound the operator norm of $\mathscr{X}_j$. The random operator only takes values that are both positive semidefinite and rank one, so the operator norm of $\mathscr{X}_j$ is equal to the trace of

the operator. Thus, we have

$$\begin{aligned}\left\|\mathscr{X}_j\right\|_{\mathrm{op}} &= s w_j^2 \operatorname{tr}\left((\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}(\varphi_{t_j}\otimes\varphi_{t_j})(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}\right)\\ &= \frac{\tilde{s}_{\mu,\epsilon}}{\tilde{\tau}_{\mu,\epsilon}(t_j)}\cdot\frac{1}{T}\operatorname{tr}\left((\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}(\varphi_{t_j}\otimes\varphi_{t_j})\right)\\ &= \frac{\tilde{s}_{\mu,\epsilon}}{\tilde{\tau}_{\mu,\epsilon}(t_j)}\cdot\tau_{\mu,\epsilon}(t_j) \qquad \text{(via Theorem 3.4.1, equation (B.5).)}\\ &\le \tilde{s}_{\mu,\epsilon}\end{aligned}$$

where the last line follows since $\tilde{\tau}_{\mu,\epsilon}(t_j)\ge\tau_{\mu,\epsilon}(t_j)$ by assumption.

The final ingredient for applying Lemma B.1.1 is to bound $\mathscr{X}_j^2$. Again, using the fact that $\tilde{\tau}_{\mu,\epsilon}(t_j)\ge\tau_{\mu,\epsilon}(t_j)$ we have:

$$\begin{aligned}\mathscr{X}_j^2 &= \frac{\tilde{s}_{\mu,\epsilon}^2}{T^2\cdot\tilde{\tau}_{\mu,\epsilon}(t_j)^2}(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}(\varphi_{t_j}\otimes\varphi_{t_j})(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}(\varphi_{t_j}\otimes\varphi_{t_j})(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}\\ &= \frac{\tilde{s}_{\mu,\epsilon}^2\cdot\langle\varphi_{t_j},(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1}\varphi_{t_j}\rangle_\mu}{T^2\cdot\tilde{\tau}_{\mu,\epsilon}(t_j)^2}(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}(\varphi_{t_j}\otimes\varphi_{t_j})(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}\\ &= \frac{\tilde{s}_{\mu,\epsilon}^2\cdot\tau_{\mu,\epsilon}(t_j)}{T\cdot\tilde{\tau}_{\mu,\epsilon}(t_j)^2}(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}(\varphi_{t_j}\otimes\varphi_{t_j})(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}\\ &= \frac{\tilde{s}_{\mu,\epsilon}\cdot\tau_{\mu,\epsilon}(t_j)}{\tilde{\tau}_{\mu,\epsilon}(t_j)}\mathscr{X}_j \preceq \tilde{s}_{\mu,\epsilon}\mathscr{X}_j.\end{aligned}$$

So, using Claim B.1.11,

$$\mathbb{E}_{t_j\propto\tilde{\tau}_{\mu,\epsilon}}\left[\mathscr{X}_j^2\right]\preceq\mathbb{E}_{t_j\propto\tilde{\tau}_{\mu,\epsilon}}\left[\tilde{s}_{\mu,\epsilon}\mathscr{X}_j\right]=\tilde{s}_{\mu,\epsilon}(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}\mathscr{G}_\mu(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}\overset{\text{def}}{=}\mathscr{M}.$$

Noticing that $\operatorname{tr}(\mathscr{M})=\tilde{s}_{\mu,\epsilon}\cdot s_{\mu,\epsilon}$ and $\|\mathscr{M}\|_{\mathrm{op}}=\tilde{s}_{\mu,\epsilon}\cdot\frac{\|\mathscr{G}_\mu\|_{\mathrm{op}}}{\|\mathscr{G}_\mu\|_{\mathrm{op}}+\epsilon}\ge\tilde{s}_{\mu,\epsilon}/2$ by our assumption that $\epsilon\le\left\|\mathscr{G}_\mu\right\|_{\mathrm{op}}$, and Lemma B.1.1 we have:

$$\begin{aligned}\Pr\left(\left\|(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}(\widehat{\mathscr{G}}_\mu-\mathscr{G}_\mu)(\mathscr{G}_\mu+\epsilon\mathscr{I}_\mu)^{-1/2}\right\|_{\mathrm{op}}\ge\Delta\right)&\le\frac{8\operatorname{tr}(\mathscr{M})}{\|\mathscr{M}\|_{\mathrm{op}}}\exp\left(\frac{-s\Delta^2/2}{\|\mathscr{M}\|_{\mathrm{op}}+2\tilde{s}_{\mu,\epsilon}\Delta/3}\right)\\ &\le 16 s_{\mu,\epsilon}\cdot\exp\left(\frac{-s\Delta^2}{2\tilde{s}_{\mu,\epsilon}(1+2\Delta/3)}\right)\\ &\le 16 s_{\mu,\epsilon}\cdot\exp\left(\frac{-3s\Delta^2}{8\tilde{s}_{\mu,\epsilon}}\right)\le\delta.\end{aligned}$$

□

B.2.3 Approximate Discretization via Leverage Score Sampling

With the operator approximation bound of Lemma B.2.2 in place, we can prove Theorem 3.4.2, which shows that we can approximately solve the regression problem of (3.10) (and by Claim 3.4.1 solve Problem 3.2.1) by sampling time domain points via over-approximations to their ridge leverage scores.

Theorem 3.4.2 (Approximate Regression via Leverage Score Sampling)**.** *Assume* $\epsilon \leq \left\|\mathcal{K}_\mu\right\|_{\text{op}}$ *and consider a measurable* $\tilde{\tau}_{\mu,\epsilon}(t)$ *with* $\tilde{\tau}_{\mu,\epsilon}(t) \geq \tau_{\mu,\epsilon}(t)$ *for all* t *and let* $\tilde{s}_{\mu,\epsilon} = \int_0^T \tilde{\tau}_{\mu,\epsilon}(t)\,dt$*. Let* $s = c\cdot\left(\tilde{s}_{\mu,\epsilon}\cdot[\log(\tilde{s}_{\mu,\epsilon})+1/\delta]\right)$ *for sufficiently large fixed constant* c *and let* $t_1,\ldots,t_s$ *be time points selected by drawing each randomly from* $[0,T]$ *with probability proportional to* $\tilde{\tau}_{\mu,\epsilon}(t)$*. For* $j\in 1,\ldots,s$ *let* $w_j = \sqrt{\frac{1}{sT}\cdot\frac{\tilde{s}_{\mu,\epsilon}}{\tilde{\tau}_{\mu,\epsilon}(t_j)}}$*. Let* $\mathbf{F}:\mathbb{C}^s\to L_2(\mu)$ *be the operator defined by:*

$$[\mathbf{F}\mathbf{x}](\xi) = \sum_{j=1}^{s} w_j\cdot\mathbf{x}(j)\cdot e^{-2\pi i\xi t_j} \tag{B.11}$$

and $\mathbf{y},\mathbf{n}\in\mathbb{R}^s$ *be the vector with* $\mathbf{y}(j) = w_j\cdot y(t_j)$ *and* $\mathbf{n}(j) = w_j\cdot n(t_j)$*.*
For any $\beta\geq 0$*, if* $\tilde{g}\in L_2(\mu)$ *satisfies:* [4]

$$\|\mathbf{F}^*\tilde{g}-(\mathbf{y}+\mathbf{n})\|_2^2+\epsilon\|\tilde{g}\|_\mu^2 \leq (1+\delta\beta)\cdot\min_{g\in L_2(\mu)}\left[\|\mathbf{F}^*g-(\mathbf{y}+\mathbf{n})\|_2^2+\epsilon\|g\|_\mu^2\right], \tag{B.12}$$

then with probability $\geq 1-\delta$*,*

$$\|\mathscr{F}_\mu^*\tilde{g}-(y+n)\|_T^2+\epsilon\|\tilde{g}\|_\mu^2 \leq 3(1+2\beta)\cdot\min_{g\in L_2(\mu)}\left[\|\mathscr{F}_\mu^*g-(y+n)\|_T^2+\epsilon\|g\|_\mu^2\right]. \tag{B.13}$$

So $\tilde{g}$ *provides an approximate solution to* (3.10) *and by Claim 3.4.1,* $\tilde{y} = \mathscr{F}_\mu^*\tilde{g}$ *solves Problem 3.2.1 with parameter* $\Theta(\epsilon)$*.*

Proof. Throughout the proof we will let $\bar{y} = y+n$ and $\bar{\mathbf{y}} = \mathbf{y}+\mathbf{n}$. Let

$$g^\star \stackrel{\text{def}}{=} \operatorname{argmin}_{g\in L_2(\mu)}\left[\|\mathscr{F}_\mu^*g-\bar{y}\|_T^2+\epsilon\|g\|_\mu^2\right].$$

By Lemma B.2.1, $g^\star = \mathscr{F}_\mu(\mathscr{K}_\mu+\lambda\mathscr{I}_T)^{-1}\bar{y}$. Denote the optimal error as $b^\star \stackrel{\text{def}}{=} \mathscr{F}_\mu^*g^\star-\bar{y}$ and the optimal cost as $B^\star \stackrel{\text{def}}{=} \|\mathscr{F}_\mu^*g^\star-\bar{y}\|_T^2+\epsilon\|g^\star\|_\mu^2$.

Reduction to Affine Embedding

We prove that, for all $g\in L_2(\mu)$, ridge leverage score sampling lets us approximate the value of the objective function of (3.10) when evaluated at g. In the randomized linear algebra literature, this is known as an *affine embedding guarantee*. Specifically, we show that, with

[4] We can see that the adjoint $\mathbf{F}^*:L_2(\mu)\to\mathbb{C}^s$ is given by $\left[\mathbf{F}^*g\right](j) = w_j\cdot\int_{\mathbb{R}} g(\xi)e^{2\pi i\xi t_j}\,d\mu(\xi)$.

probability $\geq 1-\delta$, for all $g \in L_2(\mu)$,

$$\frac{1}{2}\left(\left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon \left\|g\right\|_\mu^2\right) \leq \left\|\mathbf{F}^* g - \bar{\mathbf{y}}\right\|_2^2 + \epsilon \left\|g\right\|_\mu^2 + \alpha \leq \frac{3}{2}\left(\left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon \left\|g\right\|_\mu^2\right) \tag{B.14}$$

where α is some fixed value independent of g (but which depends on $\mathbf{F}$ and $\bar{\mathbf{y}}$) with $|\alpha| \leq \frac{1}{\delta}\cdot B^\star$.

It is not hard to see that (B.14) gives the theorem. For any $\tilde{g} \in L_2(\mu)$ satisfying:

$$\|\mathbf{F}^*\tilde{g} - \bar{\mathbf{y}}\|_2^2 + \epsilon\|\tilde{g}\|_\mu^2 \leq (1+\delta C)\cdot \min_{g\in L_2(\mu)}\left[\|\mathbf{F}^* g - \bar{\mathbf{y}}\|_2^2 + \epsilon\|g\|_\mu^2\right], \tag{B.15}$$

we can apply (B.14) to give the main claim of the theorem:

$$\begin{aligned}
\|\mathscr{F}_\mu^*\tilde{g} - \bar{y}\|_T^2 + \epsilon\|\tilde{g}\|_\mu^2 &\leq 2\left(\left\|\mathbf{F}^*\tilde{g} - \bar{\mathbf{y}}\right\|_2^2 + \epsilon\left\|\tilde{g}\right\|_\mu^2 + \alpha\right) && \text{(applying lower bound of (B.14))}\\
&\leq 2(1+\delta C)\cdot \min_{g\in L_2(\mu)}\left(\|\mathbf{F}^* g - \bar{\mathbf{y}}\|_2^2 + \epsilon\|g\|_\mu^2\right) + 2\alpha && \text{(by assumption of (B.15))}\\
&\leq 2(1+\delta C)\cdot\left(\|\mathbf{F}^* g^\star - \bar{\mathbf{y}}\|_2^2 + \epsilon\|g^\star\|_\mu^2\right) + 2\alpha\\
&= 2(1+\delta C)\cdot\left(\|\mathbf{F}^* g^\star - \bar{\mathbf{y}}\|_2^2 + \epsilon\|g^\star\|_\mu^2 + \alpha\right) - 2\delta C\alpha\\
&\leq 3(1+\delta C)\cdot\left(\|\mathscr{F}_\mu^* g^\star - \bar{y}\|_F^2 + \epsilon\|g^\star\|_\mu^2\right) - 2\delta C\alpha && \text{(upper bound of (B.14))}\\
&\leq [3(1+\delta C)+2C]\cdot\left(\|\mathscr{F}_\mu^* g^\star - \bar{y}\|_F^2 + \epsilon\|g^\star\|_\mu^2\right) && \text{(since } |\alpha| \leq \tfrac{B^\star}{\delta}\text{)}\\
&\leq 3(1+2C)\cdot\min_{g\in L_2(\mu)}\left[\|\mathscr{F}_\mu^* g - \bar{y}\|_T^2 + \epsilon\|g\|_\mu^2\right]. && (g^\star \text{ is optimum})
\end{aligned}$$

Thus, we focus our attention to proving that the affine embedding guarantee of (B.14) holds with probability $\geq 1-\delta$.

Expression of Error in Terms of $g - g^\star$

We begin by showing how, for any $g \in L_2(\mu)$, the cost $\left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon\left\|g\right\|_\mu^2$ can be written as a function of the deviation from the optimum: $g - g^\star$.

Claim B.2.4 (Expression for Excess Cost)**.** *For any $g \in L_2(\mu)$:*

$$\left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon\left\|g\right\|_\mu^2 = \left\|\mathscr{F}_\mu^*(g-g^\star)\right\|_T^2 + \epsilon\left\|g-g^\star\right\|_\mu^2 + B^\star,$$

recalling that $B^\star \stackrel{\text{def}}{=} \left\|\mathscr{F}_\mu^* g^\star - \bar{y}\right\|_T^2 + \epsilon\left\|g^\star\right\|_\mu^2$ is the optimum cost of the ridge regression problem.

Proof. Following Lemma B.2.1 we define $\mathscr{T} : L_2(\mu) \to L_2(\mu)\times L_2(T), \quad \mathscr{T} g = (\sqrt{\epsilon}g, \mathscr{F}_\mu^* g)$. For any g, $\left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon\left\|g\right\|_\mu^2 = \left\|\mathscr{T} g - (0,\bar{y})\right\|_{L_2(\mu)\times L_2(T)}^2$. Again, as in Lemma B.2.1 we know $g^\star$ is the unique minimizer of this function with the property that $(0,y) - \mathscr{T} g^\star \perp \text{range}(\mathscr{T})$ (Hunter

and Nachtergaele, 2001, Theorem 6.13). We can thus decompose:

$$\begin{aligned}\left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon \left\|g\right\|_\mu^2 &= \left\|\mathscr{T} g - (0,\bar{y})\right\|_{L_2(\mu)\times L_2(T)}^2 \\ &= \left\|\mathscr{T} g^\star - (0,\bar{y}) + (\mathscr{T} g - \mathscr{T} g^\star)\right\|_{L_2(\mu)\times L_2(T)}^2 \\ &= \left\|\mathscr{T} g^\star - (0,\bar{y})\right\|_{L_2(\mu)\times L_2(T)}^2 + \left\|\mathscr{T}(g - g^\star)\right\|_{L_2(\mu)\times L_2(T)}^2 \\ &= B^\star + \left\|\mathscr{F}_\mu^*(g - g^\star)\right\|_T^2 + \epsilon \left\|g - g^\star\right\|_\mu^2\end{aligned}$$

which gives the claim. □

Bounding The Sampling Error

We now show that Claim B.2.4 holds approximately, even after sampling. This almost immediately yields the affine embedding bound of (B.14).

Let $\tilde{B} \stackrel{\text{def}}{=} \left\|\mathbf{F}^* g^\star - \bar{\mathbf{y}}\right\|_2^2 + \epsilon \left\|g^\star\right\|_\mu^2$ be the error of the optimal solution in our randomly discretized regression problem. We can write the discretized objective function value for any $g \in L_2(\mu)$ as:

$$\begin{aligned}\left\|\mathbf{F}^* g - \bar{\mathbf{y}}\right\|_2^2 + \epsilon \left\|g\right\|_\mu^2 &= \left\|\mathbf{F}^*(g - g^\star) + \mathbf{F}^* g^\star - \bar{\mathbf{y}}\right\|_2^2 + \epsilon \left\|g^\star + (g - g^\star)\right\|_\mu^2 \\ &= \tilde{B} + \left\|\mathbf{F}^*(g - g^\star)\right\|_2^2 + \epsilon \left\|g - g^\star\right\|_\mu^2 \\ &\quad + 2\Re(\langle \mathbf{F}^*(g - g^\star), \mathbf{F}^* g^\star - \bar{\mathbf{y}}\rangle) + 2\epsilon\Re(\langle (g - g^\star), g^\star\rangle_\mu). \qquad \text{(B.16)}\end{aligned}$$

Let $\bar{\mathscr{F}}_\mu : L_2(T) \times L_2(\mu) \to L_2(\mu)$ be the operator $\bar{\mathscr{F}}_\mu(f,g) = \mathscr{F}_\mu f + \sqrt{\epsilon}\cdot g$. We can see that $\bar{\mathscr{F}}_\mu^* : L_2(\mu) \to L_2(T) \times L_2(\mu)$ is given by $\bar{\mathscr{F}}_\mu^* g = (\mathscr{F}_\mu^* g, \sqrt{\epsilon}\cdot g)$. Further, we see that $\bar{\mathscr{F}}_\mu \bar{\mathscr{F}}_\mu^* = \mathscr{G}_\mu + \epsilon\mathscr{I}_\mu$. We can write:

$$\bar{\mathscr{F}}_\mu^* = \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu) = \bar{\mathscr{P}}_\mu \bar{\mathscr{F}}_\mu^*$$

where $\bar{\mathscr{P}}_\mu = \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\bar{\mathscr{F}}_\mu$. Note that $\bar{\mathscr{P}}_\mu$ is self adjoint. Correspondingly, let $\bar{\mathbf{F}} : \mathbb{C}^s \times L_2(\mu) \to L_2(\mu)$ be given by $\bar{\mathbf{F}}(f,g) = \mathbf{F}f + \sqrt{\epsilon}\cdot g$. We have $\bar{\mathbf{F}}^* g = (\mathbf{F}^* g, \sqrt{\epsilon}\cdot g)$. We can also write $\bar{\mathbf{P}} = \bar{\mathbf{F}}^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\bar{\mathscr{F}}_\mu$, and observe that $\bar{\mathbf{F}}^* = \bar{\mathbf{P}}\bar{\mathscr{F}}_\mu^*$.

With this notation in place we can rewrite (B.16) as:

$$\begin{aligned}\langle \mathbf{F}^*(g - g^\star), \mathbf{F}^* g^\star - \bar{\mathbf{y}}\rangle + \epsilon\langle (g - g^\star), g^\star\rangle_\mu &= \langle \bar{\mathbf{F}}^*(g - g^\star), (\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\rangle_{\mathbb{C}^s \times L_2(\mu)} \\ &= \langle \bar{\mathbf{P}}\bar{\mathscr{F}}_\mu^*(g - g^\star), (\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\rangle_{\mathbb{C}^s \times L_2(\mu)} \\ &= \langle \bar{\mathscr{F}}_\mu^*(g - g^\star), \bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\rangle_{L_2(T)\times L_2(\mu)}.\end{aligned}$$

Using the fact that $\Re(z) \le |z|$ for all $z \in \mathbb{C}$, and applying Cauchy-Schwarz to the above and

plugging back into (B.16) we have:

$$\left\|\mathbf{F}^* g - \bar{\mathbf{y}}\right\|_2^2 + \epsilon \left\|g\right\|_\mu^2 \in \tilde{B} + \left\|\mathbf{F}^*(g-g^\star)\right\|_2^2 + \epsilon \left\|g-g^\star\right\|_\mu^2 \\ \pm 2\left(\left\|\mathscr{F}_\mu^*(g-g^\star)\right\|_T + \epsilon \left\|g-g^\star\right\|_\mu\right) \cdot \left\|\bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\right\|_{L_2(T)\times L_2(\mu)}. \tag{B.17}$$

We now bound $\left\|\bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\right\|_{L_2(T)\times L_2(\mu)}$. *If we had not sampled,* this would equal:

$$\left\|\bar{\mathscr{P}}_\mu(\mathscr{F}_\mu^* g^\star - \bar{y}, \sqrt{\epsilon} g^\star)\right\|_{L_2(T)\times L_2(\mu)} = \left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \bar{\mathscr{F}}_\mu \left[\bar{\mathscr{F}}_\mu^* g^\star - (\bar{y}, 0)\right]\right\|_{L_2(T)\times L_2(\mu)} = 0 \tag{B.18}$$

since $g^\star$ is the optimum of $\left\|\bar{\mathscr{F}}_\mu^* g - (\bar{y}, 0)\right\|_{L_2(T)\times L_2(\mu)}$ and thus $\bar{\mathscr{F}}_\mu^* g^\star - (\bar{y}, 0) \perp \text{range}(\bar{\mathscr{F}}_\mu^*)$. We will show that after sampling, while the norm no longer equals 0, it is still small. The bound we give is analogous to standard approximate matrix multiplication results for finite dimensional matrices. Specifically, our proof follows that of (Drineas et al., 2006a, Lemma 4).

Claim B.2.5 (Approximate Operator Application)**.** *With probability* $\geq 1-\delta$*:*

$$\left\|\bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\right\|_{L_2(T)\times L_2(\mu)} \leq \frac{1}{64} \cdot B^\star.$$

Proof. For conciseness let $\mathscr{H}$ denote the space $L_2(T) \times L_2(\mu)$. Let $\varphi_t \in L_2(\mu)$ be given by $\varphi_t(\xi) = e^{-2\pi i t \xi}$. Let $b^\star \overset{\text{def}}{=} \mathscr{F}_\mu^* g^\star - \bar{y}$ and $\mathbf{b}^\star \in \mathbb{C}^s$ be given by $\mathbf{b}^\star \overset{\text{def}}{=} \mathbf{F}^* g^\star - \bar{\mathbf{y}}$. We can see that $\mathbf{b}^\star(j) = w_j \cdot [\langle \varphi_{t_j}, g^\star \rangle_\mu - \bar{y}(t_j)]$. We have:

$$\begin{aligned}
\mathbb{E}\left[\left\|\bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\right\|_{\mathscr{H}}^2\right] &= \mathbb{E}\left[\left\|\bar{\mathbf{P}}^*(\mathbf{b}^\star, \sqrt{\epsilon} g^\star)\right\|_{\mathscr{H}}^2\right] \\
&= \mathbb{E}\left[\left\|\bar{\mathbf{P}}^*(\mathbf{b}^\star, \sqrt{\epsilon} g^\star) - \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \bar{\mathscr{F}}_\mu \left[\bar{\mathscr{F}}_\mu^* g^\star - (\bar{y}, 0)\right]\right\|_{\mathscr{H}}^2\right] \\
&\quad \text{(Since by (B.18), } \left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \bar{\mathscr{F}}_\mu \left[\bar{\mathscr{F}}_\mu^* g^\star - (\bar{y}, 0)\right]\right\|_{\mathscr{H}} = 0) \\
&= \mathbb{E}\left[\left\|\bar{\mathbf{P}}^*(\mathbf{b}^\star, \sqrt{\epsilon} g^\star) - \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \bar{\mathscr{F}}_\mu \left(b^\star, \sqrt{\epsilon} g^\star\right)\right\|_{\mathscr{H}}^2\right] \\
&\quad \text{(Since } \bar{\mathscr{F}}_\mu^* g^\star = (\mathscr{F}_\mu^* g, \sqrt{\epsilon} g) \text{ and } b^\star \overset{\text{def}}{=} \mathscr{F}_\mu^* g^\star - \bar{y}, \text{ giving } \left[\bar{\mathscr{F}}_\mu^* g^\star - (\bar{y}, 0)\right] = (b^\star, \sqrt{\epsilon} g^\star)) \\
&= \mathbb{E}\left[\left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \left(\bar{\mathbf{F}}\left(\mathbf{b}^\star, \sqrt{\epsilon} g^\star\right) - \bar{\mathscr{F}}_\mu\left(b^\star, \sqrt{\epsilon} g^\star\right)\right)\right\|_{\mathscr{H}}^2\right] \\
&\quad \text{(Factoring } \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \text{ out of } \bar{\mathbf{P}}^* = \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \bar{\mathbf{F}}) \\
&= \mathbb{E}\left[\left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \left(\mathbf{F}\mathbf{b}^\star - \mathscr{F}_\mu b^\star\right)\right\|_{\mathscr{H}}^2\right] \\
&\quad \text{(Recalling that } \bar{\mathbf{F}}(f, g) = \mathbf{F} f + \sqrt{\epsilon} g \text{ and similarly } \bar{\mathscr{F}}_\mu(f, g) = \mathscr{F}_\mu f + \sqrt{\epsilon} g) \\
&= \mathbb{E}\left[\left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \sum_{i=1}^{s} \left(\varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j) - \frac{1}{s} \mathscr{F}_\mu b^\star\right)\right\|_{\mathscr{H}}^2\right],
\end{aligned} \tag{B.19}$$

where the last equality follows since by (B.11), for any $\mathbf{x} \in \mathbb{C}^s$, $\mathbf{F}\mathbf{x} = \sum_{j=1}^{s} \varphi_{t_j} \cdot w_j \cdot \mathbf{x}(j)$. To simplify (B.19) we first bound, for an arbitrary $g \in L_2(\mu)$, $\mathbb{E}\left[\langle g, \varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j)\rangle_\mu\right]$, recalling that $\mathbf{b}^\star(j) = w_j \cdot [\langle \varphi_{t_j}, g^\star\rangle_\mu - \bar{y}(t_j)]$. Let $p(t) = \frac{\tilde{\tau}_{\mu,\epsilon}(t)}{\tilde{s}_{\mu,\epsilon}}$ be the density with which we sample our time points $t_1, \ldots, t_s$ and $w(t) = \sqrt{\frac{1}{sT \cdot p(t)}}$ be the reweighting factor we apply if we sample time t (so $w_j = w(t_j)$).

First we argue that we can apply Fubini's theorem to switch the order of the double integration in $\mathbb{E}\left[\langle g, \varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j)\rangle_\mu\right]$ (over random instantiations of $\varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j)$ and within the inner product). Letting for $z \in L_2(\mu)$, $|z| \in L_2(\mu)$ be given by $|z|(\eta) = |z(\eta)|$ we have:

$$\mathbb{E}\left[\langle |g|, |\varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j)|\rangle_\mu\right] \leq \left\|g\right\|_\mu \cdot \mathbb{E}\left[\left\|\varphi_{t_j} w_j \mathbf{b}^\star(j)\right\|_\mu\right],$$

which, noting that $\left\|\varphi_{t_j}\right\|_\mu = 1$ gives:

$$\begin{aligned}
\mathbb{E}\left[\langle |g|, |\varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j)|\rangle_\mu\right] &\leq \left\|g\right\|_\mu \cdot \mathbb{E}\left[|w_j \mathbf{b}^\star(j)|\right] \\
&= \left\|g\right\|_\mu \cdot \int_0^T |\langle \varphi_t, g^\star\rangle_\mu - \bar{y}(t)| w(t)^2 \cdot p(t)\, dt \\
&= \left\|g\right\|_\mu \cdot \frac{1}{sT} \int_0^T |\langle \varphi_t, g^\star\rangle_\mu - \bar{y}(t)|\, dt \\
&< \infty
\end{aligned}$$

where the last line follows since $g \in L_2(\mu)$ so $\left\|g\right\|_\mu < \infty$ and since $\frac{1}{T}\int_0^T |\langle \varphi_t, g^\star\rangle_\mu - \bar{y}(t)|\, dt \leq \frac{1}{T}\int_0^T \left(|\langle \varphi_t, g^\star\rangle_\mu - \bar{y}(t)|^2 + 1\right) dt = \left\|\mathscr{F}_\mu^* g^\star - \bar{y}\right\|_T^2 + 1 \leq \left\|\bar{y}\right\|_T^2 + 1 < \infty$. Since we have established that $\mathbb{E}\left[\langle |g|, |\varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j)|\rangle_\mu\right]$ is finite we can apply Fubini's theorem to compute:

$$\begin{aligned}
\mathbb{E}\left[\langle g, \varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j)\rangle_\mu\right] &= \int_0^T [\langle \varphi_t, g^\star\rangle_\mu - \bar{y}(t)]\, w(t)^2 \cdot \langle g, \varphi_t\rangle_\mu \cdot p(t)\, dt \\
&= \frac{1}{sT} \int_0^T \left(b^\star(t) \cdot \int_{\xi \in \mathbb{R}} g(\xi)^* e^{-2\pi i \xi t} d\mu(\xi)\right) dt \\
&= \frac{1}{s} \int_{\xi \in \mathbb{R}} \left(g(\xi)^* \cdot \frac{1}{T} \int_0^T e^{-2\pi i \xi t} b^\star(t) dt\right) d\mu(\xi) \\
&= \frac{1}{s} \langle g, \mathscr{F}_\mu b^\star\rangle_\mu. \qquad \text{(B.20)}
\end{aligned}$$

This in turn gives that $\mathbb{E}\left[\langle g, \varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\rangle_\mu\right] = 0$ and so for any $g \in L_2(\mu)$:

$$\begin{aligned}
&\mathbb{E}\left[\left\langle \bar{\mathscr{F}}_\mu^* (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} g, \bar{\mathscr{F}}_\mu^* (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \left(\varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\right)\right\rangle_{\mathscr{H}}\right] = \\
&\qquad \mathbb{E}\left[\left\langle (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} \bar{\mathscr{F}}_\mu \bar{\mathscr{F}}_\mu^* (\mathscr{G}_\mu + \epsilon \mathscr{I}_\mu)^{-1} g, \varphi_{t_j} \cdot w_j \cdot \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\right\rangle_\mu\right] = 0. \qquad \text{(B.21)}
\end{aligned}$$

Further, since $t_1, \ldots, t_s$ are independent, the above gives that for $j \neq k$:

$$\mathbb{E}\left[\left\langle \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_j}\cdot w_j \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\right), \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_k}\cdot w_k \mathbf{b}^\star(k) - \frac{1}{s}\mathscr{F}_\mu b^\star\right)\right\rangle_{\mathscr{H}}\right] = 0. \tag{B.22}$$

We can apply (B.21) and (B.22) to expand out (B.19), giving:

$$\begin{aligned}
&\mathbb{E}\left[\left\|\bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\right\|_{\mathscr{H}}^2\right] = \\
&\sum_{j,k\in[s]} \mathbb{E}\left[\left\langle \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\right), \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_k}\cdot w_k\cdot \mathbf{b}^\star(k) - \frac{1}{s}\mathscr{F}_\mu b^\star\right)\right\rangle_{\mathscr{H}}\right] \\
&= \sum_{j=1}^{s} \mathbb{E}\left[\left\langle \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\right), \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\right)\right\rangle_{\mathscr{H}}\right] \\
&\qquad\qquad \text{(since cross terms are 0 via (B.22))} \\
&= \sum_{j=1}^{s} \mathbb{E}\left[\left\langle \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j) - \frac{1}{s}\mathscr{F}_\mu b^\star\right), \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\left(\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j)\right)\right\rangle_{\mathscr{H}}\right] \\
&\qquad\qquad \text{(applying (B.21) to } g = -\tfrac{1}{s}\mathscr{F}_\mu b^\star) \\
&= \sum_{i=1}^{s} \mathbb{E}\left[\left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j)\right\|_{\mathscr{H}}^2 - \frac{1}{s}\left\langle \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\mathscr{F}_\mu b^\star, \bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j)\right\rangle_{\mathscr{H}}\right] \\
&= \sum_{i=1}^{s} \mathbb{E}\left[\left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j)\right\|_{\mathscr{H}}^2 - \frac{1}{s^2}\left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\mathscr{F}_\mu b^\star\right\|_{\mathscr{H}}^2\right] \\
&\leq \sum_{i=1}^{s} \mathbb{E}\left[\left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_{t_j}\cdot w_j\cdot \mathbf{b}^\star(j)\right\|_{\mathscr{H}}^2\right],
\end{aligned} \tag{B.23}$$

where the second to last line follows from (B.20).

Given the bound of (B.23) we can now expand out, using the fact that time t is sampled with probability proportional to $\tilde{\tau}_{\mu,\epsilon}(t)$:

$$\begin{aligned}
\mathbb{E}\left[\left\|\bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\right\|_{\mathscr{H}}^2\right] &\leq s\cdot \int_{t=0}^{T} \frac{\tilde{\tau}_{\mu,\epsilon}(t)}{\tilde{s}_{\mu,\epsilon}} \cdot \left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_t \cdot \frac{(\langle \varphi_t, g^\star\rangle_\mu - \bar{y}(t))\cdot \tilde{s}_{\mu,\epsilon}}{sT\cdot \tilde{\tau}_{\mu,\epsilon}(u)}\right\|_{\mathscr{H}}^2 dt \\
&= \frac{1}{sT^2}\cdot \int_{t=0}^{T} \frac{\tilde{s}_{\mu,\epsilon}\cdot b^\star(t)^2}{\tilde{\tau}_{\mu,\epsilon}(t)} \cdot \left\|\bar{\mathscr{F}}_\mu^*(\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_t\right\|_{\mathscr{H}}^2 dt \\
&= \frac{1}{sT^2}\cdot \int_{t=0}^{T} \frac{\tilde{s}_{\mu,\epsilon}\cdot b^\star(t)^2}{\tilde{\tau}_{\mu,\epsilon}(t)} \cdot \langle \varphi_t, (\mathscr{G}_\mu + \epsilon\mathscr{I}_\mu)^{-1}\varphi_t\rangle_\mu^2 dt \\
&\qquad\qquad \text{(since } \bar{\mathscr{F}}_\mu \bar{\mathscr{F}}_\mu^* = \mathscr{G}_\mu + \epsilon\mathscr{I}_\mu) \\
&= \frac{1}{sT}\cdot \int_{t=0}^{T} \frac{\tilde{s}_{\mu,\epsilon}\cdot b^\star(t)^2\cdot \tau_{\mu,\epsilon}(t)}{\tilde{\tau}_{\mu,\epsilon}(t)} dt \qquad \text{(Theorem 3.4.1, (3.29))} \\
&\leq \frac{\tilde{s}_{\mu,\epsilon}\cdot \left\|b^\star\right\|_T^2}{s}. \qquad \text{(since by assumption } \tilde{\tau}_{\mu,\epsilon}(t) \geq \tau_{\mu,\epsilon}(t))
\end{aligned}$$

Since $s = \Omega\left(\frac{\tilde{s}_{\mu,\epsilon}}{\delta}\right)$ we thus have via Markov's inequality, with probability $\geq 1-\delta$,

$$\left\|\bar{\mathbf{P}}^*(\mathbf{F}^* g^\star - \bar{\mathbf{y}}, \sqrt{\epsilon} g^\star)\right\|_{\mathscr{H}}^2 \leq \frac{1}{64}\cdot\left\|b^\star\right\|_T^2 \leq \frac{1}{64}\cdot B^\star$$

which completes the claim. Note that 64 is an arbitrarily chosen constant, which can be made as small as we want by increasing the sample size s by a constant factor. □

Plugging Claim B.2.5 back into (B.17) gives:

$$\begin{aligned}
\left\|\mathbf{F}^* g - \bar{\mathbf{y}}\right\|_2^2 + \epsilon\left\|g\right\|_\mu^2 &\in \tilde{B} + \left\|\mathbf{F}^*(g-g^\star)\right\|_2^2 + \epsilon\left\|g-g^\star\right\|_\mu^2 \pm \frac{\sqrt{B^\star}}{4}\left(\left\|\mathscr{F}_\mu^*(g-g^\star)\right\|_T + \epsilon\left\|g-g^\star\right\|_\mu\right) \\
&\in \tilde{B} + \left\|\mathbf{F}^*(g-g^\star)\right\|_2^2 + \epsilon\left\|g-g^\star\right\|_\mu^2 \pm \frac{1}{8}\left(\left\|\mathscr{F}_\mu^*(g-g^\star)\right\|_T + \epsilon\left\|g-g^\star\right\|_\mu\right)^2 \pm \frac{B^\star}{8} \\
&\in \tilde{B} + \left\|\mathbf{F}^*(g-g^\star)\right\|_2^2 + \epsilon\left\|g-g^\star\right\|_\mu^2 \pm \frac{1}{4}\left(\left\|\mathscr{F}_\mu^*(g-g^\star)\right\|_T^2 + \epsilon\left\|g-g^\star\right\|_\mu^2\right) \pm \frac{B^\star}{8}.
\end{aligned}$$

Applying the operator approximation bound of Lemma B.2.2 with error $\Delta = 1/4$ then gives:

$$\left\|\mathbf{F}^* g - \bar{\mathbf{y}}\right\|_2^2 + \epsilon\left\|g\right\|_\mu^2 \in \tilde{B} + \left(1 \pm \frac{1}{2}\right)\left(\left\|\mathscr{F}_\mu^*(g-g^\star)\right\|_2^2 + \epsilon\left\|g-g^\star\right\|_\mu^2\right) \pm \frac{1}{8}B^\star.$$

Finally, applying Claim B.2.4 gives:

$$\left\|\mathbf{F}^* g - \bar{\mathbf{y}}\right\|_2^2 + \epsilon\left\|g\right\|_\mu^2 \in (\tilde{B} - B^\star) + \left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon\left\|g\right\|_\mu^2 \pm \frac{1}{2}\left(\left\|\mathscr{F}_\mu^* g - \bar{y}\right\|_T^2 + \epsilon\left\|g\right\|_\mu^2\right).$$

Note that $\mathbb{E}[\tilde{B}] = B^\star$. So writing $\alpha = \tilde{B} - B^\star$ we have $|\alpha| \leq \frac{1}{\delta}\cdot B^\star$ with probability $1-\delta$. This completes the theorem. □

B.2.4 Frequency Subset Selection

We now prove the frequency subset selection guarantee Theorem 3.5.2 used in Section 3.5.1 to bound the leverage scores for general constraints μ, by showing that $\mathscr{F}_\mu^*$ can be well approximated by an operator whose columns are spanned by just $O(s_{\mu,\epsilon})$ frequencies.

Theorem 3.5.2 (Frequency Subset Selection)**.** *For some $s \leq \lceil 36 \cdot s_{\mu,\epsilon}\rceil$ there exists a set of distinct frequencies $\xi_1,\ldots,\xi_s \in \mathbb{R}$ such that, letting $\mathbf{C}_s : L_2(T) \to \mathbb{C}^s$ be defined by:*

$$[\mathbf{C}_s g](j) = \frac{1}{T}\int_0^T g(t) e^{-2\pi i \xi_j t}\, dt,$$

and $\mathbf{Z} = (\mathbf{C}_s\mathbf{C}_s^*)^{-1}\mathbf{C}_s\mathscr{F}_\mu^*$, for $\varphi_t \in L_2(\mu), \boldsymbol{\phi}_t \in \mathbb{C}^s$ with $\varphi_t(\xi) \overset{\text{def}}{=} e^{-2\pi i t\xi}$ and $\boldsymbol{\phi}_t(j) \overset{\text{def}}{=} \varphi_t(\xi_j)$:

$$\frac{1}{T}\int_{t\in[0,T]} \left\|\varphi_t - \mathbf{Z}^*\boldsymbol{\phi}_t\right\|_\mu^2 dt \leq 4\epsilon\cdot s_{\mu,\epsilon}. \tag{B.24}$$

Our proof relies on the following spectral error bound for weighted frequency subset selection:

Lemma B.2.3 (Frequency Subset Selection – Direct Spectral Approximation)**.** *For some $s \leq \lceil 36 \cdot s_{\mu,\epsilon} \rceil$ there exists a set of distinct frequencies $\xi_1, \ldots, \xi_s \in \mathbb{R}$ and positive weights $w_1, \ldots, w_s \in \mathbb{R}$ such that letting $\bar{\mathbf{C}}_s : L_2(T) \to \mathbb{C}^s$ be given by $[\bar{\mathbf{C}}_s g](j) \stackrel{\text{def}}{=} \frac{1}{T}\int_0^T g(t) w_j e^{-2\pi i \xi_j t} dt$, and letting $\widehat{\mathcal{K}_\mu} = \bar{\mathbf{C}}_s^* \bar{\mathbf{C}}_s$, we have:*

$$\frac{1}{2} \cdot (\mathcal{K}_\mu + \epsilon \mathcal{I}_T) \preceq \widehat{\mathcal{K}_\mu} + \epsilon \mathcal{I}_T \preceq \frac{3}{2} \cdot (\mathcal{K}_\mu + \epsilon \mathcal{I}_T). \tag{B.25}$$

Proof. We prove a more general statement, in which we are given $0 < \Delta < 1$ and we select $s = \lceil 9 s_{\mu,\epsilon}/\Delta^2 \rceil$ frequencies $\xi_1, \ldots, \xi_s \in \mathbb{R}$ and weights $w_1, \ldots, w_s \in \mathbb{R}$ such that

$$(1-\Delta)(\mathcal{K}_\mu + \epsilon \mathcal{I}_T) \preceq \widehat{\mathcal{K}_\mu} + \epsilon \mathcal{I}_T \preceq (1+\Delta)(\mathcal{K}_\mu + \epsilon \mathcal{I}_T).$$

The claim follows by setting $\Delta = 1/2$. We can assume that $\xi_1, \ldots, \xi_s$ are distinct, since if ξ_i, ξ_j are equal, we can simply remove ξ_j and update $w_i \leftarrow \sqrt{w_i^2 + w_j^2}$, leaving $\widehat{\mathcal{K}_\mu}$ unchanged and only decreasing s.

The last condition is equivalent to $\mathcal{K}_\mu - \Delta(\mathcal{K}_\mu + \epsilon \mathcal{I}_T) \preceq \widehat{\mathcal{K}_\mu} \preceq \mathcal{K}_\mu + \Delta(\mathcal{K}_\mu + \epsilon \mathcal{I}_T)$. Multiplying with $(\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2}$ on the left and right, we find that the condition is equivalent to:

$$-\Delta \mathcal{I}_T \preceq (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2} \widehat{\mathcal{K}_\mu} (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2} - (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2} \mathcal{K}_\mu (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2} \preceq \Delta \mathcal{I}_T.$$

To shorten the notation, we write $\mathcal{Z} = (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2} \mathcal{K}_\mu (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2}$ and $\widehat{\mathcal{Z}} = (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2} \widehat{\mathcal{K}_\mu} (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2}$. Given $\xi \in \mathbb{R}$, we define $\vartheta_\xi(t) \stackrel{\text{def}}{=} e^{2\pi i t \xi}$ ($\vartheta_\xi \in L_2(T)$). It is easy to verify that

$$\mathcal{K}_\mu = \int_{\mathbb{R}} (\vartheta_\xi \otimes \vartheta_\xi) d\mu(\xi), \quad \text{and} \quad \widehat{\mathcal{K}_\mu} = \sum_{i=1}^{s} w_i^2 (\vartheta_{\xi_i} \otimes \vartheta_{\xi_i}).$$

Further define $\bar{\vartheta}_\xi \stackrel{\text{def}}{=} (\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2} \vartheta_\xi$. Since $(\mathcal{K}_\mu + \epsilon \mathcal{I}_T)^{-1/2}$ is self-adjoint and bounded, we have

$$\mathcal{Z} = \int_{\mathbb{R}} (\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) d\mu(\xi), \quad \text{and} \quad \widehat{\mathcal{Z}} = \sum_{i=1}^{s} w_i^2 (\bar{\vartheta}_{\xi_i} \otimes \bar{\vartheta}_{\xi_i}).$$

We prove the existence of $\xi_1, \ldots, \xi_s$ and $w_1, \ldots, w_s$ using the deterministic selection process known as "BSS" (Batson et al., 2009).[5] In particular, we use a process that in essence is the same as the one described in (Cohen et al., 2016a, Theorem 5). Indeed, since $\|\mathcal{Z}\|_{\text{op}} \leq 1$ and $\text{tr}(\mathcal{Z}) = s_{\mu,\epsilon}$ the aforementioned results would suffice if we were dealing with matrices instead of operators. The rest of the proof extends these results to the operator case. Let

$$\delta_u \stackrel{\text{def}}{=} \Delta/3 + 2\Delta^2/9, \quad \delta_l \stackrel{\text{def}}{=} \Delta/3 - 2\Delta^2/9,$$

[5] We remark that unlike the process described in Batson et al. (2009), our existence proof does not trivially translate to an algorithm, since it involves a search over an infinite domain. Nevertheless, for our needs, existence suffices.

and for $j = 0, 1, \ldots, s$,

$$\mathscr{X}_l^{(j)} \stackrel{\text{def}}{=} j\delta_l \cdot \mathscr{Z} - s_{\mu,\epsilon} \cdot \mathscr{I}_T, \quad \mathscr{X}_u^{(j)} \stackrel{\text{def}}{=} j\delta_u \cdot \mathscr{Z} + s_{\mu,\epsilon} \cdot \mathscr{I}_T.$$

The process we shall describe iteratively selects $\xi_1, \xi_2, \ldots$ and unscaled weights $\tilde{w}_1, \tilde{w}_2, \ldots$ such that if we define $\widehat{\mathscr{Z}}^{(j)} \stackrel{\text{def}}{=} \sum_{i=1}^{j} \tilde{w}_i (\bar{\vartheta}_{\xi_i} \otimes \bar{\vartheta}_{\xi_i})$ the invariant

$$\mathscr{X}_l^{(j)} \prec \widehat{\mathscr{Z}}^{(j)} \prec \mathscr{X}_u^{(j)} \tag{B.26}$$

is held. Let us write $s = \lceil 9 s_{\mu,\epsilon}/\Delta^2 \rceil$, so $s = C s_{\mu,\epsilon}/\Delta^2$ for $C \geq 9$. If indeed we are able to select the frequencies and weights for s steps such that this invariant holds, we shall have

$$\frac{C s_{\mu,\epsilon}}{3\Delta} \cdot \mathscr{Z} - (1 + 2C/9) \cdot s_{\mu,\epsilon} \cdot \mathscr{I}_T \preceq \widehat{\mathscr{Z}}^{(s)} \preceq \frac{C s_{\mu,\epsilon}}{3\Delta} \cdot \mathscr{Z} + (1 + 2C/9) \cdot s_{\mu,\epsilon} \cdot \mathscr{I}_T$$

where we used the fact that $\mathscr{Z} \preceq \mathscr{I}_T$. Since $C \geq 9$ we have $-\Delta \cdot \mathscr{I}_T \preceq \frac{3\Delta}{C s_{\mu,\epsilon}} \widehat{\mathscr{Z}}^{(s)} - \mathscr{Z} \preceq \Delta \cdot \mathscr{I}_T$, so by defining $w_i = \sqrt{\frac{3\Delta}{C s_{\mu,\epsilon}}} \tilde{w}_i$ for $i = 1, \ldots, s$ we shall then have $\widehat{\mathscr{Z}} = \frac{3\Delta}{C s_{\mu,\epsilon}} \widehat{\mathscr{Z}}^{(s)}$ thereby establishing the desired bound.

Thus, it suffices to show that we can select frequencies and weights iteratively so that (B.26) is maintained. In fact, the iterative selection process will maintain two additional invariants:

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, (\mathscr{X}_u^{(j)} - \widehat{\mathscr{Z}}^{(j)})^{-1} \bar{\vartheta}_\xi \rangle_T \, d\mu(\xi) \leq 1$$
$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, (\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)})^{-1} \bar{\vartheta}_\xi \rangle_T \, d\mu(\xi) \leq 1$$

All the invariants hold for $j = 0$. Eq. (B.26) trivially holds for $j = 0$. As for the integral,

$$\begin{aligned}
\int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, (\mathscr{X}_u^{(0)} - \widehat{\mathscr{Z}}^{(0)})^{-1} \bar{\vartheta}_\xi \rangle_T \, d\mu(\xi) &= \int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, s_{\mu,\epsilon}^{-1} \bar{\vartheta}_\xi \rangle_T \, d\mu(\xi) \\
&= s_{\mu,\epsilon}^{-1} \int_{\mathbb{R}} \langle (\mathscr{K}_\mu + \epsilon \mathscr{I}_T)^{-1/2} \vartheta_\xi, (\mathscr{K}_\mu + \epsilon \mathscr{I}_T)^{-1/2} \vartheta_\xi \rangle_T \, d\mu(\xi) \\
&= s_{\mu,\epsilon}^{-1} \int_{\mathbb{R}} \langle \vartheta_\xi, (\mathscr{K}_\mu + \epsilon \mathscr{I}_T)^{-1} \vartheta_\xi \rangle_T \, d\mu(\xi) \\
&= s_{\mu,\epsilon}^{-1} \operatorname{tr}\left((\mathscr{K}_\mu + \epsilon \mathscr{I}_T)^{-1} \mathscr{K}_T \right) = 1
\end{aligned}$$

and similarly for the second invariant. In the above, the last equality is due to Claim B.1.9.

Suppose by induction that the invariants hold for j. We prove that it is possible to pick a frequency ξ and weight $w > 0$ such that if we set $\xi_{j+1} = \xi$ and $\tilde{w}_{j+1} = w$ then the invariants will hold for $j + 1$.

Fix j. For $t \geq 0$, let us denote

$$M_u(t) = \left(\mathscr{X}_u^{(j)} + t\mathscr{Z} - \widehat{\mathscr{Z}}^{(j)} \right)^{-1}, \quad \text{and} \quad M_l(t) = \left(\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)} - t\mathscr{Z} \right)^{-1},$$

where M_u is defined for any $t \geq 0$ (since the inverted operator is strictly positive and bounded, so invertible), and M_l is defined for $t < 1$. We can define M_l for $t < 1$ since $\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)} - t\mathscr{Z} \succ 0$ for $t < 1$ as we show now. Due to Claim B.1.9:

$$\mathrm{tr}\left((\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)})^{-1}\mathscr{Z}\right) = \int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, (\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)})^{-1}\bar{\vartheta}_\xi\rangle_T \, d\mu(\xi) \leq 1.$$

Since $\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)} \succ 0$ (induction assumption), $(\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)})^{-1}$ is bounded so according to Claim B.1.4, $\mathscr{Z} \preceq \widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)}$, and then Claim B.1.6 implies that $\widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)} - t\mathscr{Z} \succ 0$.

Consider some fixed ξ. We first claim that for $w < 1/\langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T$ we have $M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) \succ 0$. Obviously, the last statement holds for $w = 0$, and due to continuity of $w \mapsto \langle x, (M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi))x\rangle_T$ with respect to w, it will also hold for some interval around 0. Let $w^\star$ be the maximal value such that for all $w \in [0, w^\star)$ we have $M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) \succ 0$. Our goal is to show that $w^\star \geq 1/\langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T$. Assume by contradiction that $w^\star < 1/\langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T$. For every $w \in [0, w^\star)$, the operator $M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi)$ is invertible, and we can apply a operator pseudo-inversion lemma due to (Deng, 2011, Theorem 2.1) to find that

$$(M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_T))^{-1} = M_u(\delta_u) + \frac{w}{1 - w \cdot \langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T} M_u(\delta_u)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi)M_u(\delta_u).$$

Since we assumed $w^\star < 1/\langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T$, it is clear from the above equation that there exists a K such that for all $w \in [0, w^\star)$ we have:

$$(M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_T))^{-1} \preceq K \cdot \mathscr{I}_T.$$

Note that $M_u(\delta_u)^{-1} - w^\star(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi)$ is not strictly positive for otherwise due to continuity we could have extended the interval, so there exists a x with norm 1 such that $\langle x, (M_u(\delta_u)^{-1} - w^\star(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi))x\rangle < \frac{1}{2K}$. Let $w_1, w_2, \ldots$ be a sequence which converges to $w^\star$, and let $y_i = \left(M_u(\delta_u)^{-1} - w_i(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi)\right)^{1/2} x$. We now have $\langle y_i, y_i\rangle_T = \langle x, (M_u(\delta_u)^{-1} - w_i(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi))x\rangle_T \to \langle x, (M_u(\delta_u)^{-1} - w^\star(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi))x\rangle_T < \frac{1}{2K}$ as $i \to \infty$. However $\langle y_i, (M_u(\delta_u)^{-1} - w_i(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi))^{-1} y_i\rangle_T = \langle x, x\rangle_T = 1$ which contradicts the bound on $(M_u(\delta_u)^{-1} - w_i(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi))^{-1}$.

Thus, if we picked ξ and $w < 1/\langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T$ for the step, we shall have $\mathscr{X}_u^{(j+1)} - \widehat{\mathscr{Z}}^{(j+1)} = M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) \succ 0$ as required, and the upper invariant will translate to

$$\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, \left(M_u(\delta_u)^{-1} - w(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi)\right)^{-1} \bar{\vartheta}_\eta \right\rangle_T d\mu(\eta) \leq 1,$$

which is equivalent to

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(\delta_u)\bar{\vartheta}_\eta\rangle_T \, d\mu(\eta) + \frac{w \cdot \int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_u(\delta_u)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi)M_u(\delta_u)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta)}{1 - w \cdot \langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T} \leq 1.$$

The induction hypothesis is $\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(0)\bar{\vartheta}_\eta\rangle_T d\mu(\eta) \le 1$, so the upper invariant is held if

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(\delta_u)\bar{\vartheta}_\eta\rangle_T d\mu(\eta) - \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(0)\bar{\vartheta}_\eta\rangle_T d\mu(\eta) + \frac{w\cdot\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_u(\delta_u)(\bar{\vartheta}_\xi\otimes\bar{\vartheta}_\xi)M_u(\delta_u)\bar{\vartheta}_\eta\right\rangle_T d\mu(\eta)}{1 - w\cdot\langle\bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T} \le 0. \tag{B.27}$$

Consider any $\eta\in\mathbb{R}$, and let $f_\eta(y) \stackrel{\text{def}}{=} \langle \bar{\vartheta}_\eta, M_u(y)\bar{\vartheta}_\eta\rangle_T$. Using the operator inversion formula, we have for any $t_2 \ge t_1$:

$$M_u(t_2) = M_u(t_1) - (t_2-t_1)M_u(t_1)\mathcal{Z}^{1/2}\left(\mathcal{I}_T + (t_2-t_1)\mathcal{Z}^{1/2}M_u(t_1)\mathcal{Z}^{1/2}\right)^{-1}\mathcal{Z}^{1/2}M_u(t_1).$$

From this equation we see that

$$f'_\eta(y) = -\langle\bar{\vartheta}_\eta, M_u(y)\mathcal{Z}M_u(y)\bar{\vartheta}_\eta\rangle_T.$$

Furthermore, since for $t_2 > t_1$ we have $\mathcal{I}_T + t_2\mathcal{Z}^{1/2}M_u(t_1)\mathcal{Z}^{1/2} \succeq \mathcal{I}_T + t_1\mathcal{Z}^{1/2}M_u(t_1)\mathcal{Z}^{1/2}$ and both operators are strictly positive and bounded, then $(\mathcal{I}_T + t_1\mathcal{Z}^{1/2}M_u(t_1)\mathcal{Z}^{1/2})^{-1} \preceq (\mathcal{I}_T + t_2\mathcal{Z}^{1/2}M_u(t_1)\mathcal{Z}^{1/2})^{-1}$, and we can easily verify that f_η has a positive second derivative and hence is convex. Thus,

$$f_\eta(\delta_u) - f_\eta(0) \le -\delta_u\langle\bar{\vartheta}_\eta, M_u(y)\mathcal{Z}M_u(y)\bar{\vartheta}_\eta\rangle_T.$$

After integrating on both sides, we have the bound

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(\delta_u)\bar{\vartheta}_\eta\rangle_T d\mu(\eta) - \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(0)\bar{\vartheta}_\eta\rangle_T d\mu(\eta) \le -\delta_u\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(\delta_u)\mathcal{Z}M_u(\delta_u)\bar{\vartheta}_\eta\rangle_T d\mu(\eta).$$

Using this bound in (B.27) and rearranging, we find that for any ξ, the upper invariant is held if we select w such that

$$\frac{1}{w} > \frac{\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_u(\delta_u)(\bar{\vartheta}_\xi\otimes\bar{\vartheta}_\xi)M_u(\delta_u)\bar{\vartheta}_\eta\right\rangle_T d\mu(\eta)}{\delta_u\int_{\mathbb{R}}\langle\bar{\vartheta}_\eta, M_u(\delta_u)\mathcal{Z}M_u(\delta_u)\bar{\vartheta}_\eta\rangle_T d\mu(\eta)} + \langle\bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T. \tag{B.28}$$

Note that if this is held, we also have $w < 1/\langle\bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T$, as previously required.

We now consider the lower invariants. If we picked ξ and $w > 0$ for the step, then $\widehat{\mathcal{Z}}^{(j+1)} - \mathcal{X}_l^{(j+1)} = M_l(\delta_l)^{-1} + w(\bar{\vartheta}_\xi\otimes\bar{\vartheta}_\xi) \succeq M_l(\delta_l)^{-1} > 0$ as long $\delta_l < 1$ which holds for our choice of δ_l. So the left part of (B.26) will hold regardless of how we choose ξ and $w > 0$. As for the lower trace bound, it translates to:

$$\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, \left(M_l(\delta_l)^{-1} + w(\bar{\vartheta}_\xi\otimes\bar{\vartheta}_\xi)\right)^{-1}\bar{\vartheta}_\eta\right\rangle_T d\mu(\eta) \le 1.$$

Applying another variant of operator pseudo-inversion lemma (Ogawa, 1988, Theorem 2), we find that the last condition is equivalent to

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(\delta_l)\bar{\vartheta}_\eta\rangle_T d\mu(\eta) - \frac{w\cdot\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_l(\delta_l)(\bar{\vartheta}_\xi\otimes\bar{\vartheta}_\xi)M_l(\delta_l)\bar{\vartheta}_\eta\right\rangle_T d\mu(\eta)}{1 + w\cdot\langle\bar{\vartheta}_\xi, M_l(\delta_l)\bar{\vartheta}_\xi\rangle_T} \le 1.$$

The induction hypothesis is

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(0)\bar{\vartheta}_\eta \rangle_T d\mu(\eta) \leq 1$$

so the lower invariant is held if

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(\delta_l)\bar{\vartheta}_\eta \rangle_\mu d\mu(\eta) - \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(0)\bar{\vartheta}_\eta \rangle_T d\mu(\eta) - \frac{w \cdot \int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_l(\delta_l)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_l(\delta_l)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta)}{1 + w \cdot \langle \bar{\vartheta}_\xi, M_l(\delta_l)\bar{\vartheta}_\xi \rangle_T} \leq 0. \tag{B.29}$$

Similarly to before, by using the convexity of the first integrand, we can bound

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(\delta_l)\bar{\vartheta}_\eta \rangle_T d\mu(\eta) - \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(0)\bar{\vartheta}_\eta \rangle_T d\mu(\eta) \leq \delta_l \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(\delta_l)\mathcal{Z} M_l(\delta_l)\bar{\vartheta}_\eta \rangle_T d\mu(\eta).$$

Using this bound in (B.29) and rearranging, we find that for any ξ, the lower invariant is held if we select w such that

$$\frac{1}{w} \leq \frac{\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_l(\delta_l)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_l(\delta_l)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta)}{\delta_l \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(\delta_l)\mathcal{Z} M_l(\delta_l)\bar{\vartheta}_\eta \rangle_T d\mu(\eta)} - \langle \bar{\vartheta}_\xi, M_l(\delta_l)\bar{\vartheta}_\xi \rangle_T. \tag{B.30}$$

Thus, we need to show that there exists a ξ and w such that both (B.28) and (B.30) hold. However, for a given ξ, such a w will surely exist if

$$\begin{aligned}
&\frac{\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_u(\delta_u)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_u(\delta_u)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta)}{\delta_u \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(\delta_u)\mathcal{Z} M_u(\delta_u)\bar{\vartheta}_\eta \rangle_T d\mu(\eta)} + \langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi \rangle_T \\
&< \frac{\int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_l(\delta_l)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_l(\delta_l)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta)}{\delta_l \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(\delta_l)\mathcal{Z} M_l(\delta_l)\bar{\vartheta}_\eta \rangle_T d\mu(\eta)} - \langle \bar{\vartheta}_\xi, M_l(\delta_l)\bar{\vartheta}_\xi \rangle_T.
\end{aligned}$$

Thus, it suffices to show that there exists a ξ for which the above inequality holds. To show that such a ξ exists, we will show that the inequality holds for the integral of both sides with respect to μ measure. This will guarantee the existence of such a ξ since the Lebesgue integral is strictly positive for non-negative functions. We compute:

$$\begin{aligned}
&\int_{\mathbb{R}} \int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_u(\delta_u)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_u(\delta_u)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta) d\mu(\xi) \\
&= \int_{\mathbb{R}} \int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_u(\delta_u)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_u(\delta_u)\bar{\vartheta}_\eta \right\rangle_T d\mu(\xi) d\mu(\eta) \\
&= \int_{\mathbb{R}} \int_{\mathbb{R}} \left\langle M_u(\delta_u)\bar{\vartheta}_\eta, (\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_u(\delta_u)\bar{\vartheta}_\eta \right\rangle_T d\mu(\xi) d\mu(\eta) \\
&= \int_{\mathbb{R}} \left\langle M_u(\delta_u)\bar{\vartheta}_\eta, \mathcal{Z} M_u(\delta_u)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta) \\
&= \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_u(\delta_u)\mathcal{Z} M_u(\delta_u)\bar{\vartheta}_\eta \rangle_T d\mu(\eta).
\end{aligned}$$

Similarly,

$$\int_{\mathbb{R}} \int_{\mathbb{R}} \left\langle \bar{\vartheta}_\eta, M_l(\delta_l)(\bar{\vartheta}_\xi \otimes \bar{\vartheta}_\xi) M_l(\delta_l)\bar{\vartheta}_\eta \right\rangle_T d\mu(\eta) d\mu(\xi) = \int_{\mathbb{R}} \langle \bar{\vartheta}_\eta, M_l(\delta_l)\mathcal{Z} M_l(\delta_l)\bar{\vartheta}_\eta \rangle_T d\mu(\eta).$$

$\mathscr{Z}$ is self-adjoint and positive definite, so the operator pseudo-inversion lemma (Ogawa, 1988, Theorem 2) implies that $M_u(\delta_u) \preceq M_u(0)$, so by the induction hypothesis

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, M_u(\delta_u)\bar{\vartheta}_\xi\rangle_T d\mu(\xi) \leq \int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, M_u(0)\bar{\vartheta}_\xi\rangle_T d\mu(\xi) \leq 1.$$

We now consider the lower invariant. We already showed that $\mathscr{Z} \preceq \widehat{\mathscr{Z}}^{(j)} - \mathscr{X}_l^{(j)}$, so as long as $\delta_l \leq 1/2$ we will have:

$$\int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, M_l(\delta_l)\bar{\vartheta}_\xi\rangle_T d\mu(\xi) = \int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, (M_l(0)^{-1} - \delta_l\mathscr{Z})^{-1}\bar{\vartheta}_\xi\rangle_T d\mu(\xi) \leq 2\int_{\mathbb{R}} \langle \bar{\vartheta}_\xi, M_l(0)\bar{\vartheta}_\xi\rangle_T d\mu(\xi) \leq 2$$

where we used Claim B.1.5. So there will be a gap in the value of the integrals (as desired), if

$$\frac{1}{\delta_u} + 1 < \frac{1}{\delta_l} - 2,$$

which is the case for our selection of δ_l and δ_u. □

From Lemma B.2.3 we can prove a stronger spectral error bound for the projection onto the range of $\bar{\mathbf{C}}_s$.

Lemma B.2.4 (Frequency Subset Selection – Projection Based Spectral Approximation)**.** *For some $s \leq \lceil 36 \cdot s_{\mu,\epsilon}\rceil$ there exists a set of distinct frequencies $\xi_1,\ldots,\xi_s \in \mathbb{R}$ such that letting $\mathbf{C}_s : L_2(T) \to \mathbb{C}^s$ and $\mathbf{Z}: L_2(\mu) \to \mathbb{C}^s$ be defined as in Theorem 3.5.2 and $\widehat{\mathscr{G}}_\mu = \mathbf{Z}^*\mathbf{C}_s\mathbf{C}_s^*\mathbf{Z}$,*

$$\widehat{\mathscr{G}}_\mu \preceq \mathscr{G}_\mu \preceq \widehat{\mathscr{G}}_\mu + \epsilon\mathscr{I}_\mu. \tag{B.31}$$

Proof. Let $\xi_1,\ldots,\xi_s \in \mathbb{C}$ and $w_1,\ldots,w_s \in \mathbb{R}$ be the frequencies and weights shown to exist in Lemma B.2.3 and let $\bar{\mathbf{C}}_s$ be as defined in that lemma (note that $\bar{\mathbf{C}}_s$ is identical to $\mathbf{C}_s$ except that its rows are weighted by $w_1,\ldots,w_s$.) First note that for any $g \in L_2(\mu)$,

$$\langle g, \widehat{\mathscr{G}}_\mu g\rangle_\mu = \left\|\mathbf{C}_s^*\mathbf{Z}g\right\|_T^2 = \left\|\mathbf{C}_s^*(\mathbf{C}_s\mathbf{C}_s^*)^{-1}\mathbf{C}_s\mathscr{F}_\mu^* g\right\|_T^2 \leq \left\|\mathscr{F}_\mu^* g\right\|_T^2 = \langle g, \mathscr{G}_\mu g\rangle_\mu$$

where the inequality follows from observing that $\mathbf{C}_s^*(\mathbf{C}_s\mathbf{C}_s^*)^{-1}\mathbf{C}_s$ is an orthogonal projection. Thus $\widehat{\mathscr{G}}_\mu \preceq \mathscr{G}_\mu$. It remains to show that $\mathscr{G}_\mu \preceq \widehat{\mathscr{G}}_\mu + \epsilon\mathscr{I}_\mu$. Let $\bar{\mathscr{P}} = \mathscr{I}_T - \mathbf{C}_s^*(\mathbf{C}_s\mathbf{C}^*)^{-1}\mathbf{C}_s$ be the projection to the orthogonal complement of $\mathbf{C}_s^*$'s range and let $\widehat{\mathscr{K}_\mu} = \bar{\mathbf{C}}_s^*\bar{\mathbf{C}}_s$ be as defined in Lemma B.2.3. Rearranging the guarantee of Lemma B.2.3 gives $\mathscr{K}_\mu \preceq 2\cdot\widehat{\mathscr{K}_\mu} + \epsilon\mathscr{I}_T$ which immediately yields,

$$\bar{\mathscr{P}}\mathscr{K}_\mu\bar{\mathscr{P}} \preceq 2\cdot\bar{\mathscr{P}}\widehat{\mathscr{K}_\mu}\bar{\mathscr{P}} + \epsilon\bar{\mathscr{P}}\mathscr{I}_T\bar{\mathscr{P}}.$$

Note that $\bar{\mathbf{C}}_s\bar{\mathscr{P}} = 0$ (since $\bar{\mathscr{P}}$ is an orthogonal projection onto $\ker(\mathbf{C}_s) = \ker(\bar{\mathbf{C}}_s)$) and so $\bar{\mathscr{P}}\widehat{\mathscr{K}_\mu}\bar{\mathscr{P}} = 0$, giving:

$$\bar{\mathscr{P}}\mathscr{K}_\mu\bar{\mathscr{P}} \preceq \epsilon\bar{\mathscr{P}}\mathscr{I}_T\bar{\mathscr{P}} \preceq \epsilon\mathscr{I}_T. \tag{B.32}$$

Note that $\bar{\mathscr{P}}\mathscr{K}_\mu\bar{\mathscr{P}} = \bar{\mathscr{P}}\mathscr{F}_\mu^*\mathscr{F}_\mu\bar{\mathscr{P}}$ and

$$\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu} = \mathscr{F}_\mu\mathscr{F}_\mu^* - \mathbf{Z}^*\mathbf{C}_s\mathbf{C}_s^*\mathbf{Z} = \mathscr{F}_\mu\bar{\mathscr{P}}\mathscr{F}_\mu^*.$$

Thus by (B.32) we also have $\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu} \preceq \epsilon\mathscr{I}_\mu$ (since the norm of an operator and its adjoint are the same so $\bar{\mathscr{P}}\mathscr{K}_\mu\bar{\mathscr{P}} \preceq \epsilon\mathscr{I}_T \implies \mathscr{F}_\mu\bar{\mathscr{P}}\mathscr{F}_\mu^* \preceq \epsilon\mathscr{I}_\mu$), which completes the lemma. □

Finally, from Lemma B.2.4 we can prove the frequency subset selection guarantee of Theorem 3.5.2.

Proof of Theorem 3.5.2. We consider the same set of frequencies $\xi_1,\dots,\xi_s$ shown to exist in Lemma B.2.4 and the corresponding operators $\mathbf{C}_s$, $\mathbf{Z}$. We show that these frequencies satisfy the guarantee of Theorem 3.5.2. First, we define $\mathbf{K} \overset{\text{def}}{=} \mathbf{C}_s\mathbf{C}_s^* = \frac{1}{T}\int_0^T(\boldsymbol{\phi}_t\otimes\boldsymbol{\phi}_t)\,dt$ (we abuse the notation and use $\boldsymbol{\phi}_t$ to denote both the vector defined in the Theorem statement, and the operator $x\in\mathbb{C}\mapsto x\boldsymbol{\phi}_t$). From Claim B.1.9:

$$\begin{aligned}\frac{1}{T}\int_{t\in[0,T]}\left\|\varphi_t-\mathbf{Z}^*\boldsymbol{\phi}_t\right\|_\mu^2 dt &= \operatorname{tr}\left(\frac{1}{T}\int_{t\in[0,T]}(\varphi_t-\mathbf{Z}^*\boldsymbol{\phi}_t)\otimes(\varphi_t-\mathbf{Z}^*\boldsymbol{\phi}_t)\,dt\right)\\ &= \operatorname{tr}\left(\frac{1}{T}\int_{t\in[0,T]}\varphi_t\otimes\varphi_t\,dt\right)+\operatorname{tr}\left(\frac{1}{T}\int_{t\in[0,T]}\mathbf{Z}^*\boldsymbol{\phi}_t\otimes\mathbf{Z}^*\boldsymbol{\phi}_t\,dt\right)\\ &\quad-\operatorname{tr}\left(\frac{1}{T}\int_{t\in[0,T]}\mathbf{Z}^*\boldsymbol{\phi}_t\otimes\varphi_t\,dt\right)-\operatorname{tr}\left(\frac{1}{T}\int_{t\in[0,T]}\varphi_t\otimes\mathbf{Z}^*\boldsymbol{\phi}_t\,dt\right)\end{aligned}$$

We have,

$$\frac{1}{T}\int_{t\in[0,T]}\varphi_t\otimes\varphi_t\,dt = \mathscr{G}_\mu,$$

From Claim B.1.8:

$$\frac{1}{T}\int_{t\in[0,T]}\mathbf{Z}^*\boldsymbol{\phi}_t\otimes\mathbf{Z}^*\boldsymbol{\phi}_t\,dt = \mathbf{Z}^*\left(\frac{1}{T}\int_{t\in[0,T]}\boldsymbol{\phi}_t\otimes\boldsymbol{\phi}_t\,dt\right)\mathbf{Z} = \mathbf{Z}^*\mathbf{K}\mathbf{Z} = \widehat{\mathscr{G}_\mu}$$

Next, consider $\frac{1}{T}\int_0^T\boldsymbol{\phi}_t\otimes\varphi_t\,dt$. For any $\alpha\in L_2(\mu)$,

$$\frac{1}{T}\left(\int_0^T\boldsymbol{\phi}_t\otimes\varphi_t\,dt\right)\alpha = \frac{1}{T}\int_0^T\langle\varphi_t,\alpha\rangle_\mu\boldsymbol{\phi}_t\,dt$$

where the integral on the left is a weak vector integral. Since for every $g\in L_2(T)$,

$$\mathbf{C}_s g = \frac{1}{T}\int_0^T g(t)\boldsymbol{\phi}_t\,dt$$

and for every $\alpha\in L_2(\mu)$, $[\mathscr{F}_\mu^*\alpha](t) = \langle\varphi_t,\alpha\rangle_\mu$, we have $\frac{1}{T}\int_0^T\boldsymbol{\phi}_t\otimes\varphi_t\,dt = \mathbf{C}_s\mathscr{F}_\mu^*$, so

$$\frac{1}{T}\int_{t\in[0,T]}\mathbf{Z}^*\boldsymbol{\phi}_t\otimes\varphi_t\,dt = \mathbf{Z}^*\left(\frac{1}{T}\int_{t\in[0,T]}\boldsymbol{\phi}_t\otimes\varphi_t\,dt\right) = \mathbf{Z}^*\mathbf{C}_s\mathscr{F}_\mu^* = \mathbf{Z}^*\mathbf{K}\mathbf{Z} = \widehat{\mathscr{G}_\mu}.$$

Combining the previous observations, we find that

$$\frac{1}{T}\int_{t\in[0,T]} \left\|\varphi_t - \mathbf{Z}^*\boldsymbol{\phi}_t\right\|_\mu^2 dt = \operatorname{tr}(\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu}).$$

Let $v_1,\ldots,v_{2s_{\mu,\epsilon}} \in L_2(\mu)$ be the eigenfunctions of $\mathscr{G}_\mu$ corresponding to its top $2s_{\mu,\epsilon}$ eigenvalues. Define $\mathbf{X}\colon L_2(\mu)\to\mathbb{C}^{2s_{\mu,\epsilon}}$ as: for $g\in L_2(\mu)$, $[\mathbf{X}g](j) = \langle v_j, g\rangle_\mu$. Note that

$$\begin{aligned}\operatorname{tr}\left(\widehat{\mathscr{G}_\mu} - \mathbf{X}^*\mathbf{X}\widehat{\mathscr{G}_\mu}\mathbf{X}^*\mathbf{X}\right) &= \operatorname{tr}\left(\mathbf{Z}^*\mathbf{C}_s\mathbf{C}_s\mathbf{Z} - \mathbf{X}^*\mathbf{X}\mathbf{Z}^*\mathbf{C}_s\mathbf{C}_s\mathbf{Z}\mathbf{X}^*\mathbf{X}\right)\\ &= \operatorname{tr}\left(\mathbf{C}_s\mathbf{Z}\mathbf{Z}^*\mathbf{C}_s - \mathbf{C}_s\mathbf{Z}\mathbf{X}^*\mathbf{X}\mathbf{Z}^*\mathbf{C}_s\right) \geq 0\end{aligned}$$

since $\mathbf{C}_s\mathbf{Z}\mathbf{Z}^*\mathbf{C}_s \succeq \mathbf{C}_s\mathbf{Z}\mathbf{X}^*\mathbf{X}\mathbf{Z}^*\mathbf{C}_s$ ($\mathbf{X}^*\mathbf{X}$ is a projection, so $\mathbf{X}^*\mathbf{X}\preceq\mathscr{I}_\mu$). So we can bound:

$$\begin{aligned}\frac{1}{T}\int_{t\in[0,T]} \left\|\varphi_t - \mathbf{Z}^*\boldsymbol{\phi}_t\right\|_\mu^2 dt &= \operatorname{tr}\left(\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu}\right)\\ &\leq \operatorname{tr}\left(\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu}\right) + \operatorname{tr}\left(\widehat{\mathscr{G}_\mu} - \mathbf{X}^*\mathbf{X}\widehat{\mathscr{G}_\mu}\mathbf{X}^*\mathbf{X}\right)\\ &= \operatorname{tr}\left(\mathscr{G}_\mu - \mathbf{X}^*\mathbf{X}\mathscr{G}_\mu\mathbf{X}^*\mathbf{X}\right) + \operatorname{tr}\left(\mathbf{X}^*\mathbf{X}(\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu})\mathbf{X}^*\mathbf{X}\right).\end{aligned} \tag{B.33}$$

Let i_ϵ be the smallest i with $\lambda_i(\mathscr{G}_\mu)\leq\epsilon$. We have:

$$s_{\mu,\epsilon} = \sum_{i=1}^{\infty}\frac{\lambda_i(\mathscr{G}_\mu)}{\lambda_i(\mathscr{G}_\mu)+\epsilon} \geq \sum_{i=1}^{i+\epsilon}\frac{\lambda_i(\mathscr{G}_\mu)}{\lambda_i(\mathscr{G}_\mu)+\epsilon} \geq \frac{i_\epsilon}{2}.$$

Thus we can bound $\operatorname{tr}\left(\mathscr{G}_\mu - \mathbf{X}^*\mathbf{X}\mathscr{G}_\mu\mathbf{X}^*\mathbf{X}\right)$ as:

$$\operatorname{tr}\left(\mathscr{G}_\mu - \mathbf{X}^*\mathbf{X}\mathscr{G}_\mu\mathbf{X}^*\mathbf{X}\right) = \sum_{i=2s_{\mu,\epsilon}+1}^{\infty}\lambda_i(\mathscr{G}_\mu) \leq \sum_{i=i_\epsilon+1}^{\infty}\lambda_i(\mathscr{G}_\mu) \leq 2\epsilon s_{\mu,\epsilon}. \tag{B.34}$$

where the last bound follows from the fact that $s_{\mu,\epsilon}\geq\sum_{i=i_\epsilon+1}^{\infty}\frac{\lambda_i(\mathscr{G}_\mu)}{\lambda_i(\mathscr{G}_\mu)+\epsilon}\geq\sum_{i=i_\epsilon+1}^{\infty}\frac{\lambda_i(\mathscr{G}_\mu)}{2\epsilon}$.

We can also bound $\operatorname{tr}\left(\mathbf{X}^*\mathbf{X}(\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu})\mathbf{X}^*\mathbf{X}\right)$ using Lemma B.2.4. Since $\mathscr{G}_\mu\preceq\widehat{\mathscr{G}_\mu}+\epsilon\mathscr{I}_\mu$ we have:

$$\operatorname{tr}\left(\mathbf{X}^*\mathbf{X}(\mathscr{G}_\mu - \widehat{\mathscr{G}_\mu})\mathbf{X}^*\mathbf{X}\right) \leq \epsilon\operatorname{tr}(\mathbf{X}^*\mathbf{X}\mathbf{X}^*\mathbf{X}) = 2\epsilon s_{\mu,\epsilon}. \tag{B.35}$$

Plugging (B.34) and (B.35) back into (B.33) we have, $\frac{1}{T}\int_{t\in[0,T]}\left\|\varphi_t - \mathbf{Z}^*\boldsymbol{\phi}_t\right\|_\mu^2 dt \leq 4\epsilon\cdot s_{\mu,\epsilon}$, which completes the theorem. □

B.3 Tight Statistical Dimension Bound for Bandlimited Functions

In Section 3.5 we demonstrated, perhaps surprisingly, that a simple function $\tilde{\tau}_{\mu,\epsilon}(t)$ (defined in Theorem 3.5.6) exists for *any* μ that upper bounds $\tau_{\mu,\epsilon}(t)$ and has $\tilde{s}_{\mu,\epsilon} = \tilde{O}(s_{\mu,\epsilon})$. Combined with Theorem 3.4.3 this yields our main algorithmic result Theorem 3.2.3, which shows that

we can achieve $O\left(s_{\mu,\epsilon}\log^2(s_{\mu,\epsilon})\right)$ sample complexity with just $\tilde{O}\left(s_{\mu,\epsilon}^{\omega}\right)$ runtime.

Instantiating Theorem 3.2.3 using the approximate ridge leverage function of Theorem 3.5.6 requires an upper bound on $s_{\mu,\epsilon}$. In this section we show how to bound $s_{\mu,\epsilon}$ when μ is uniform measure on some interval – i.e., when our interpolation problem is over bandlimited functions. In Section B.4 we leverage this result to bound $s_{\mu,\epsilon}$ for a number of other important priors, including multiband, Gaussian, and Cauchy-Lorentz.

Beyond letting us upper bound $s_{\mu,\epsilon}$ to apply Theorem 3.2.3, our proof for bandlimited functions is constructive, giving a simple upper bound on $\tau_{\mu,\epsilon}(t)$ for any t. This upper bound can be plugged directly into Algorithm 13 and Theorem 3.4.3 to give a tightening of Theorem 3.2.3 by a logarithmic factor in the bandlimited case. Like our general result, the proof is based on the definition of leverage scores given in (3.11). This definition makes it clear that, to upper bound $\tau_{\mu,\epsilon}(t)$, it suffices to show that a function with Fourier support controlled by μ cannot "spike" too extremely at time t.

For bandlimited functions, we obtain a smoothness bound by introducing and applying a Bernstein type smoothness bound for low-degree polynomials and relying on the fact that any bandlimited function is well approximated by a low-degree polynomial. This approach mirrors the general proof in Section 3.5, which uses a more sophisticated smoothness bound for Fourier sparse functions.

Our result for bandlimited functions is as follows:

Theorem B.3.1. *Let μ be the uniform measure on $[-F,F]$. Let $q = \lceil 16\pi eFT + 2\log(1/\epsilon) + 11\rceil$. For all $t \in [0,T]$, let the approximate ridge leverage function $\tilde{\tau}_{\mu,\epsilon}$ equal:*

$$\tilde{\tau}_{\mu,\epsilon}(t) = \frac{1}{T}\left(4 + \frac{q}{\sqrt{\min(t, T-t)/T}}\right).$$

For any $\epsilon \le 1, F, T$, $\tilde{\tau}_{\mu,\epsilon}(t)$ satisfies:

1. $\tilde{\tau}_{\mu,\epsilon}(t) \ge \tau_{\mu,\epsilon}(t)$.
2. $\int_0^T \tilde{\tau}_{\mu,\epsilon}(t)\,dt \overset{\text{def}}{=} \tilde{s}_{\mu,\epsilon} = O\left(FT + \log(1/\epsilon)\right)$.

Thus we have $s_{\mu,\epsilon} \le \tilde{s}_{\mu,\epsilon} = O\left(FT + \log(1/\epsilon)\right)$.

Combined with Theorem 3.4.3, Theorem B.3.1 immediately gives:

Corollary B.3.1. *Let μ be the uniform measure on $[-F,F]$. Using $\tilde{\tau}_{\mu,\epsilon}$ as defined in Theorem B.3.1, Algorithm 13 returns $t_1,\ldots,t_s \in [0,T]$ and $\mathbf{z} \in \mathbb{C}^s$ such that $\tilde{y}(t) = \sum_{i=1}^{s} \mathbf{z}(i)\cdot k_\mu(t_i,t)$ satisfies with probability $\ge 1-\delta$:*

$$\left\|\tilde{y} - y\right\|_T^2 \le 6\epsilon\left\|x\right\|_\mu^2 + 7\left\|n\right\|_T^2.$$

The algorithm queries $y+n$ at s points and runs in $O(s^\omega)$ time and furthermore $\tilde{y}(t)$ can be evaluated using Algorithm 14 in $O(s)$ time, where $s = O\left([FT + \log(1/\epsilon)] \cdot [\log(FT + \log(1/\epsilon)) + 1/\delta]\right)$.

Proof. The corollary follows immediately from Theorem 3.4.3 after noting that

- $Z = O(1)$ since, as shown in Appendix B.5, $k_\mu(t_1, t_2) = \frac{\sin(2\pi F(t_1 - t_2))}{2\pi F(t_1 - t_2)}$ and so can be computed in $O(1)$ arithmetic operations.
- $W = O(1)$ since to sample points proportional to $\tilde{\tau}_{\mu,\epsilon}(t)$, we must sample a mixture of the uniform distribution and the distribution with density proportional to $\frac{1}{\sqrt{\min(t, T-t)/T}}$. It suffices to show that we can sample from the later in $O(1)$ time, and in fact that we can sample $t \in [0, 1/2]$ with probability proportional to $\frac{1}{\sqrt{t}}$ in $O(1)$ time, since we can then symmetrize and scale this distribution. We can accomplish this with inverse transform sampling. Our density is $p(t) = \frac{1}{2\sqrt{2t}}$ and so its cumulative distribution function is $C(t) = \sqrt{t/2}$. Thus we can sample z uniformly in $[0, 1]$ and return $C^{-1}(z) = 2z^2$, which will be a sample from the desired distribution. This can be done in $O(1)$ operations.

□

B.3.1 Smoothness bounds for polynomials

Our main technical tool is a Bernstein type smoothness bounds for low-degree polynomials. In general, low-degree polynomials are smoother than high-degree polynomials, and thus cannot spike as sharply. There are a number of ways to formalize this statement. The well known Markov brother's inequality and Bernstein inequality bound the maximum derivative of a polynomial by a function of the polynomial's degree and it's maximum value on an interval.

To bound leverage scores, we are interested in a slightly different metric of smoothness. In particular, we need to bound the maximum squared value of a polynomial by its average squared value on $[0, T]$. We can use standard properties of the Legendre polynomials to prove:

Claim B.3.1. *For any degree d polynomial $p(\cdot)$ with complex coefficients and any $t \in [0, T]$, if $r = \frac{\min(t, T-t)}{T}$, then:*

$$|p(t)|^2 \leq \frac{d+1}{\sqrt{r}} \cdot \frac{1}{T} \int_0^T |p(t)|^2 \, dt.$$

This bound is tighter for points near the center of the interval $[0, T]$ and goes to infinity near the edges. Using Markov brother's inequality, it is possible to obtain a fixed up bound of $O(d^2)$, which is tighter for small values of r. However, this won't be necessary for our purposes. We note that, when $t = T/2$, the upper bound on $p(t)^2$ improves to $O(d)$ times the average squared value of p. This improvement is nearly optimal: the upper bound of Claim B.3.1 is matched

up to a logarithmic factor by an appropriately scaled and shifted Chebyshev polynomial of the first kind applied to $[T/2-t]^2$ (see e.g. Frostig et al. (2016)).

Proof of Claim B.3.1. The claim follows from properties of the standard orthogonal Legendre polynomials, which are denoted by $P_0, P_1, \ldots$ and defined via the recurrence relation:

$$\begin{aligned} P_0(x) &= 1 \\ P_1(x) &= x \\ &\vdots \\ P_k(x) &= \frac{2k-1}{k} x \cdot P_{k-1}(x) - \frac{n-1}{n} \cdot P_{k-2}(x). \end{aligned}$$

The Legendre polynomials are orthogonal over the interval $[-1,1]$ with respect to the constant weight function. In particular, they satisfy

$$\int_{-1}^{1} P_j(x) P_k(x)\, dx = \frac{2}{2j+1} \delta_{j,k}, \tag{B.36}$$

where $\delta_{m,n}$ is the Kronecker delta function. Additionally, for all j and all $x \in [-1,1]$, $|P_j(x)| \leq 1$.

Using these facts we show that for any degree d polynomial $p(\cdot)$, interval $[a,b]$, and $x \in [a,b]$:

$$|p(x)|^2 \leq \frac{d+1}{\sqrt{r}} \cdot \frac{\int_a^b |p(t)|^2\, dt}{(b-a)},$$

where $r = \frac{\min(|a-x|,|b-x|)}{(b-a)}$. Setting $a=0$ and $b=T$ gives the claim.

We begin by noting that, without loss of generality, we can assume that $a=-1$ and $b=1$. In particular, shift and stretch $p(x)$ by defining $g(x) = p\left(\frac{2(x-a)}{b-a} - 1\right)$. g has degree d and the maximum of $|g(x)|^2$ for $x \in [-1,1]$ is the same as the maximum of $|p(x)|^2$ for $x \in [a,b]$. Additionally, $\frac{\int_{-1}^{1} |g(t)|^2\, dt}{2} = \frac{\int_a^b |p(t)|^2\, dt}{(b-a)}$. Accordingly, to prove the claim it suffices to prove that, for any degree d polynomial g,

$$\max_{x \in [-1,1]} |g(x)|^2 \leq \frac{d+1}{\sqrt{r}} \cdot \frac{\int_{-1}^{1} |g(t)|^2\, dt}{2}. \tag{B.37}$$

Our proof depends on a Bernstein type inequality for Legendre polynomials, which can be found in Lorch (1983). Specifically, for all $j = 0,1,2,\ldots$ and any $x \in [-1,1]$ it holds that:

$$P_j(x)^2 \leq \frac{2}{\pi(j+1/2)} \frac{1}{\sqrt{1-x^2}}. \tag{B.38}$$

Writing g in the Legendre basis:

$$g(x) = \sum_{j=0}^{d} c_j P_j(x),$$

we have from (B.38) that

$$|g(x)| \le \sum_{j=0}^{d} |c_j| \left(\frac{2}{\pi(j+1/2)} \frac{1}{\sqrt{1-x^2}} \right)^{1/2}$$

and thus

$$\begin{aligned} |g(x)|^2 &\le (d+1) \sum_{j=0}^{d} |c_j|^2 \frac{2}{\pi(j+1/2)} \frac{1}{\sqrt{1-x^2}} \\ &= \frac{2}{\pi} \frac{(d+1)}{\sqrt{1-x^2}} \sum_{j=0}^{d} |c_j|^2 \frac{2}{2j+1} \\ &= \frac{2}{\pi} \frac{(d+1)}{\sqrt{1-x^2}} \int_{-1}^{1} |g(t)|^2 \, dt. \end{aligned} \tag{B.39}$$

The last equality step follows from (B.36). Finally, let $q = \min(|-1-x|, |1-x|)$ and note that

$$\frac{1}{\sqrt{1-x^2}} = \frac{1}{\sqrt{1-(1-q)^2}} \le \frac{1}{\sqrt{q}}.$$

As defined, $r = q/2$ Plugging into (B.39) we have a final bound of

$$|g(x)|^2 \le \frac{4}{\pi} \frac{(d+1)}{\sqrt{2r}} \frac{\int_{-1}^{1} |g(t)|^2 \, dt}{2} < \frac{(d+1)}{\sqrt{r}} \frac{\int_{-1}^{1} |g(t)|^2 \, dt}{2},$$

which establishes (B.37) and thus the claim. □

B.3.2 Smoothness bounds for bandlimited functions

With Claim B.3.1 in place, we are now ready to prove our main result for bandlimited functions.

Proof of Theorem B.3.1. Following Definition 3.4.1, our goal is to choose $\tilde{\tau}_{\mu,\epsilon}$ to satisfy:

$$\tilde{\tau}_{\mu,\epsilon}(t) \ge \frac{1}{T} \cdot \frac{|[\mathscr{F}_\mu \alpha](t)|^2}{\|\mathscr{F}_\mu \alpha\|_T^2 + \epsilon \|\alpha\|_\mu^2}. \tag{B.40}$$

for any α. Let $z = \mathscr{F}_\mu \alpha$. Expanding $e^{-2i\pi\xi t}$ using its Maclaurin series and letting d be some

degree parameter that we will fix later, we write z as the sum of two functions, a and b:

$$\begin{aligned} z(t) &= \frac{1}{2F}\int_{-F}^{F} \alpha(\xi) e^{-2i\pi\xi t}\, d\xi \\ &= \sum_{j=0}^{\infty} \frac{1}{2F}\int_{-F}^{F} \alpha(\xi)\frac{(-2\pi i\xi)^j}{j!} t^j\, d\xi \\ &= \sum_{j=0}^{d}\left(\frac{1}{2F}\int_{-F}^{F}\alpha(\xi)\frac{(-2\pi i\xi)^j}{j!}\, d\xi\right) t^j + \sum_{j=d+1}^{\infty}\frac{1}{2F}\int_{-F}^{F}\alpha(\xi)\frac{(-2\pi i\xi)^j}{j!} t^j\, d\xi \\ &\stackrel{\text{def}}{=} a(t) + b(t). \end{aligned} \tag{B.41}$$

Note that a is a degree d polynomial with complex coefficients. So by Claim B.3.1,

$$|a(t)|^2 \le \frac{d+1}{\sqrt{\min(t, T-t)/T}} \cdot \|a\|_T^2. \tag{B.42}$$

Turning our attention to b, we see that:

$$\begin{aligned} |b(t)| &= \left|\sum_{j=d+1}^{\infty}\frac{1}{2F}\int_{-F}^{F}\alpha(\xi)\frac{(-2\pi i\xi)^j}{j!} t^j\, d\xi\right| \le \sum_{j=d+1}^{\infty}\frac{(2\pi FT)^j}{j!}\frac{1}{2F}\int_{-F}^{F}|\alpha(\xi)|\, d\xi \\ &\le \sum_{j=d+1}^{\infty}\frac{(2\pi FT)^j}{j!}\sqrt{\frac{1}{2F}\int_{-F}^{F} 1\, d\xi}\sqrt{\|\alpha\|_\mu^2} = \sum_{j=d+1}^{\infty}\frac{(2\pi FT)^j}{j!}\cdot\|\alpha\|_\mu. \end{aligned} \tag{B.43}$$

The second to last step uses Cauchy-Schwarz inequality. Finally using that for all j, $j! \ge (j/e)^j$, for any $d \ge 4\pi eFT$:

$$\begin{aligned} \sum_{j=d+1}^{\infty}\frac{(2\pi FT)^j}{j!} &\le \sum_{j=d+1}^{\infty}\left(\frac{2\pi eFT}{j}\right)^j \\ &\le \sum_{j=d+1}^{\infty}\left(\frac{2\pi eFT}{d+1}\right)^j \\ &\le \sum_{j=d+1}^{\infty}\left(\frac{1}{2}\right)^j = \frac{1}{2^d}. \end{aligned} \tag{B.44}$$

So, if we take $d = \lceil 4\pi eFT + \log(1/\epsilon)/2 + 1\rceil$, it follows from (B.43) and (B.44) that

$$|b(t)| \le \frac{1}{2^d}\cdot\|\alpha\|_\mu \le \frac{1}{2^{\lceil\log(1/\epsilon)/2\rceil+1}}\cdot\|\alpha\|_\mu \le \frac{\sqrt{\epsilon}}{2}\cdot\|\alpha\|_\mu.$$

Moreover, $\|b\|_T \le \frac{\sqrt{\epsilon}}{2}\|\alpha\|_\mu$. Using the decomposition of (B.41) and the fact that for any real

non-negative c, d, $c^2 + d^2 \le (c+d)^2$, and for any complex e, f, $|e+f|^2 \le 2|e|^2 + 2|f|^2$:

$$\begin{aligned}\frac{|z(t)|^2}{\|z\|_T^2 + \epsilon\|\alpha\|_\mu^2} &\le \frac{|a(t)+b(t)|^2}{(\|a\|_T - \|b\|_T)^2 + \epsilon\|\alpha\|_\mu^2} \\ &\le \frac{2|a(t)|^2 + 2|b(t)|^2}{\frac{1}{2}(\|a\|_T - \|b\|_T + \sqrt{\epsilon}\|\alpha\|_\mu)^2} \\ &\le \frac{4|a(t)|^2 + 4|b(t)|^2}{(\|a\|_T + \frac{\sqrt{\epsilon}}{2}\|\alpha\|_\mu)^2} \\ &\le \frac{4|a(t)|^2 + \epsilon\|\alpha\|_\mu^2}{\|a\|_T^2 + \frac{\epsilon}{4}\|\alpha\|_\mu^2}.\end{aligned}$$

It follows from (B.42) that:

$$\begin{aligned}\frac{|z(t)|^2}{\|z\|_T^2 + \epsilon\|\alpha\|_\mu^2} &\le \max\left(\frac{4|a(t)|^2}{\|a\|_T^2}, 4\right) \\ &\le \frac{4(d+1)}{\sqrt{\min(t, T-t)/T}} + 4.\end{aligned}$$

In Theorem B.3.1 we set $q = \lceil 16\pi eFT + 2\log(1/\epsilon) + 11\rceil$. We have $q \ge 4 \cdot \lceil 4\pi eFT + \log(1/\epsilon)/2 + 2\rceil = 4(d+1)$ since, for any x, $\lceil 4x+3\rceil \ge 4\lceil x\rceil$. Recalling that $z = \mathscr{F}_\mu \alpha$, it follows $\tilde{\tau}_{\mu,\epsilon}$ defined in that theorem satisfies (B.40) for any α. It remains to bound the total measure of our approximate ridge leverage function, $\tilde{s}_{\mu,\epsilon}$. To do so, note that:

$$\tilde{s}_{\mu,\epsilon} = \frac{2}{T}\int_0^{T/2} \frac{q}{\sqrt{t/T}} + 4\, dt.$$

We can compute:

$$\frac{2}{T}\int_0^{T/2} \frac{q}{\sqrt{t/T}} + 4\, dt = 2\int_0^{1/2} \frac{q}{\sqrt{t}} + 4\, dt = 2\sqrt{2}q + 4 = O(FT + \log(1/\epsilon)).$$

This bound establishes the theorem. □

B.4 Statistical Dimension of Common Fourier Constraints

In this section we leverage Theorem B.3.1 to give upper bounds on the statistical dimensions of a number common priors μ used for Fourier constrained interpolation, including multiband, Gaussian, and Cauchy-Lorentz priors. We start by giving two simple lemmas that we use to translate our bound for bandlimited functions to these more general priors.

Lemma B.4.1 (Statistical dimension of sum of measures)**.** *For any finite measures $\mu_1, \mu_2, \cdots \mu_s$*

on $\mathbb{R}$*, if* $\mu \stackrel{\text{def}}{=} \mu_1 + \mu_2 + \cdots + \mu_s$ *is a measure, then:*

$$s_{\mu,\epsilon} \leq \sum_{i=1}^{s} s_{\mu_i,\epsilon}.$$

Proof. We can see from Definition 3.2.2 that for $\mu = \mu_1 + \ldots + \mu_s$ the kernel operator $\mathcal{K}_\mu$ satisfies $\mathcal{K}_\mu = \sum_{i=1}^{s} \mathcal{K}_{\mu_i}$. We can thus bound:

$$\begin{aligned} s_{\mu,\epsilon} = \operatorname{tr}\left(\mathcal{K}_\mu(\mathcal{K}_\mu + \epsilon\mathcal{I}_T)^{-1}\right) &= \sum_{i=1}^{s} \operatorname{tr}\left(\mathcal{K}_{\mu_i}(\mathcal{K}_\mu + \epsilon\mathcal{I}_T)^{-1}\right) \\ &\leq \sum_{i=1}^{s} \operatorname{tr}\left(\mathcal{K}_{\mu_i}(\mathcal{K}_{\mu_i} + \epsilon\mathcal{I}_T)^{-1}\right) \\ &= \sum_{i=1}^{s} s_{\mu_i,\epsilon}. \end{aligned}$$

The second to last inequality follows since $0 \preceq \mathcal{K}_{\mu_i} \preceq \mathcal{K}_\mu$, so $0 \prec \mathcal{K}_{\mu_i} + \epsilon\mathcal{I}_T \preceq \mathcal{K}_\mu + \epsilon\mathcal{I}_T$ and $(\mathcal{K}_\mu + \epsilon\mathcal{I}_T)^{-1} \preceq (\mathcal{K}_{\mu_i} + \epsilon\mathcal{I}_T)^{-1}$ by Claim B.1.5. Letting e_1, e_2 be an orthonormal basis for $L_2(T)$, we thus have:

$$\begin{aligned} \operatorname{tr}\left(\mathcal{K}_{\mu_i}(\mathcal{K}_{\mu_i} + \epsilon\mathcal{I}_T)^{-1}\right) &= \operatorname{tr}\left(\mathcal{K}_{\mu_i}^{1/2}(\mathcal{K}_{\mu_i} + \epsilon\mathcal{I}_T)^{-1}\mathcal{K}_{\mu_i}^{1/2}\right) \\ &= \sum_{i=1}^{\infty} \langle \mathcal{K}_{\mu_i}^{1/2} e_i, (\mathcal{K}_{\mu_i} + \epsilon\mathcal{I}_T)^{-1}\mathcal{K}_{\mu_i}^{1/2} e_i\rangle_T \\ &\geq \sum_{i=1}^{\infty} \langle \mathcal{K}_{\mu_i}^{1/2} e_i, (\mathcal{K}_{\mu} + \epsilon\mathcal{I}_T)^{-1}\mathcal{K}_{\mu_i}^{1/2} e_i\rangle_T \\ &= \operatorname{tr}\left(\mathcal{K}_{\mu_i}^{1/2}(\mathcal{K}_{\mu} + \epsilon\mathcal{I}_T)^{-1}\mathcal{K}_{\mu_i}^{1/2}\right) \\ &= \operatorname{tr}\left(\mathcal{K}_{\mu_i}(\mathcal{K}_{\mu} + \epsilon\mathcal{I}_T)^{-1}\right). \end{aligned}$$

This completes the lemma. □

Lemma B.4.2 (Statistical dimension of scaled measures)**.** *For any measure* μ *on* $\mathbb{R}$ *and any parameter* $\gamma > 0$*, if* $\mu' = \mu/\gamma$ *and* $\epsilon' = \epsilon/\gamma$ *then:*

$$s_{\mu,\epsilon} = s_{\mu',\epsilon'}.$$

Proof. From Definition 3.2.2, we can see that $\mathcal{K}_{\mu'} = \frac{1}{\gamma}\mathcal{K}_\mu$ and thus has eigenvalues equal to

$\lambda_1/\gamma, \lambda_2/\gamma, \ldots$, where $\lambda_1, \lambda_2, \ldots$ are the eigenvalues of $\mathscr{K}_\mu$. Therefore,

$$\begin{aligned} s_{\mu',\epsilon'} &= \sum_{i=1}^{\infty} \frac{\lambda_i(\mathscr{K}_{\mu'})}{\lambda_i(\mathscr{K}_{\mu'}) + \epsilon'} \\ &= \sum_{i=1}^{\infty} \frac{\lambda_i/\gamma}{\lambda_i/\gamma + \epsilon/\gamma} \\ &= \sum_{i=1}^{\infty} \frac{\lambda_i}{\lambda_i + \epsilon} = s_{\mu,\epsilon}. \end{aligned}$$

□

We now use Lemmas B.4.1 and B.4.2 to prove our statistical dimension bounds. We first start with multiband Fourier constraints, showing that the statistical dimension is roughly proportional to the total length of all the frequency bands times the time domain window size, intuitively matching the Landau rate for asymptotic recovery of multiband functions Landau (1967).

Theorem B.4.1 (Multiband statistical dimension)**.** *Consider a set of s disjoint frequency bands, $I_1, I_2, \cdots, I_s$, and suppose that the length of the band I_i is denoted by F_i. Let μ be the measure which induces a uniform probability density on $I_1 \cup I_2 \cup \cdots \cup I_s$. We have:*

$$s_{\mu,\epsilon} = O\left(\sum_{i=1}^{s} F_i T + s\log(1/\epsilon)\right).$$

Proof. For every i, let μ_i be the measure defined by $\mu_i(A) = \mu(A \cap I_i)$. Note that we have $\mu = \sum_i \mu_i$ and so can invoke Lemma B.4.1, giving:

$$s_{\mu,\epsilon} \leq \sum_{i=1}^{s} s_{\mu_i,\epsilon}. \tag{B.45}$$

If μ_i gave a uniform probability measure on frequency band I_i (i.e., if we had $\mu_i(\mathbb{R}) = 1$), we could use the result of Theorem B.3.1 to obtain $s_{\mu_i,\epsilon} = O(F_i T + \log(1/\epsilon))$. This is not the case, but we can instead let $\gamma_i \stackrel{\text{def}}{=} \mu_i(\mathbb{R}) \leq 1$. By Lemma B.4.2,

$$s_{\mu_i,\epsilon} = s_{(\mu_i/\gamma_i),(\epsilon/\gamma_i)}.$$

Now μ_i/γ_i is a uniform probability measure on I_i, so we can invoke Theorem B.3.1 giving:

$$s_{\mu_i,\epsilon} = s_{(\mu_i/\gamma_i),(\epsilon/\gamma_i)} = O\big(F_i T + \log(\gamma_i/\epsilon)\big).$$

Plugging this bound in (B.45) and using that $\gamma_i \leq 1$ we obtain:

$$s_{\mu,\epsilon} = O\left(\sum_{i=1}^{s} F_i T + \log(\gamma_i/\epsilon)\right) = O\left(\sum_{i=1}^{s} F_i T + s\log(1/\epsilon)\right),$$

completing the theorem. □

We next bound the statistical dimension of Gaussian measure.

Theorem B.4.2 (Gaussian statistical dimension)**.** *Suppose that μ induces the Gaussian probability distribution with standard deviation F defined by $d\mu(\xi) = \frac{1}{\sqrt{2\pi F^2}} e^{-\xi^2/2F^2} d\xi$. We have:*

$$s_{\mu,\epsilon} = O\Big(FT\sqrt{\log(1/\epsilon)} + \log(1/\epsilon)\Big).$$

Proof. Let I_h be the interval defined by $I_h = \big\{\xi \in \mathbb{R} : |\xi| \le F\sqrt{\log(1/\epsilon)}\big\}$. We decompose μ into two measures μ_h and μ_t as follows:

$$\mu_h(A) = \mu(A \cap I_h), \quad \text{and} \quad \mu_t(A) = \mu(A - A \cap I_h).$$

We can see that $\mu = \mu_h + \mu_t$ and so by Lemma B.4.1, $s_{\mu,\epsilon} \le s_{\mu_t,\epsilon} + s_{\mu_h,\epsilon}$. For μ_t we have:

$$\begin{aligned}\operatorname{tr}(\mathscr{K}_{\mu_t}) = \mu_t(\mathbb{R}) &= \frac{1}{\sqrt{2\pi F^2}} \int_{|\xi| > F\sqrt{\log(1/\epsilon)}} e^{-\xi^2/2F^2}\, d\xi \\ &= 1 - \operatorname{erf}\Big(\sqrt{\log(1/\epsilon)}\Big) \le 2\epsilon,\end{aligned}$$

where the last bound follows from a Chernoff bound, giving $1 - \operatorname{erf}(x) \le 2e^{-x^2}$ (Wainwright, 2019). This lets us crudely bound:

$$s_{\mu_t,\epsilon} = \operatorname{tr}\left(\mathscr{K}_{\mu_t}(\mathscr{K}_{\mu_t} + \epsilon\mathscr{I}_T)^{-1}\right) \le \frac{\operatorname{tr}(\mathscr{K}_{\mu_t})}{\epsilon} \le 2, \tag{B.46}$$

where the first ineguality is because $\left\|(\mathscr{K}_{\mu_t} + \epsilon\mathscr{I}_T)^{-1}\right\|_{\text{op}} \le 1/\epsilon$.

We next bound the statistical dimension of μ_h. Let $\tilde{\mu}_h$ be a uniform measure on I_h, with $d\mu(\xi) = \frac{1}{\sqrt{2\pi F^2}} d\xi$ for all $\xi \in I_h$. Note that $d\tilde{\mu}_h(\xi) \ge d\mu_h(\xi)$ for all $\xi \in I_h$ which gives that $K_{\mu_h} \preceq K_{\tilde{\mu}_h}$ and so $s_{\mu_h,\epsilon} \le s_{\tilde{\mu}_h,\epsilon}$.

Let $\gamma \overset{\text{def}}{=} \tilde{\mu}_h(\mathbb{R}) = \sqrt{\frac{2\log(1/\epsilon)}{\pi}}$. By Lemma B.4.2, $s_{\tilde{\mu},\epsilon} = s_{(\tilde{\mu}/\gamma),(\epsilon/\gamma)}$. Since $\tilde{\mu}/\gamma$ is a uniform probability measure on I_h, invoking Theorem B.3.1 yields:

$$\begin{aligned}s_{\mu_h,\epsilon} \le s_{\tilde{\mu}_h,\epsilon} = s_{(\tilde{\mu}_h/\gamma),(\epsilon/\gamma)} &= O\Big(FT\sqrt{\log(1/\epsilon)} + \log(\gamma/\epsilon)\Big) \\ &= O\Big(FT\sqrt{\log(1/\epsilon)} + \log(1/\epsilon)\Big),\end{aligned} \tag{B.47}$$

where the last equality follows from the fact that $\gamma = O\big(\sqrt{\log(1/\epsilon)}\big)$. Combining (B.46) and (B.47) and applying Lemma B.4.1 we have:

$$\begin{aligned}s_{\mu,\epsilon} &\le s_{\mu_t,\epsilon} + s_{\mu_h,\epsilon} \\ &= 2 + O\Big(FT\sqrt{\log(1/\epsilon)} + \log(1/\epsilon)\Big) = O\Big(FT\sqrt{\log(1/\epsilon)} + \log(1/\epsilon)\Big),\end{aligned}$$

which completes the theorem. □

Finally, we bound the statistical dimension of the Cauchy-Lorentz measure.

Theorem B.4.3 (Cauchy-Lorentz statistical dimension)**.** *If μ induces the Cauchy-Lorentz probability distribution with scale parameter F defined by $d\mu(\xi) = \frac{1}{\pi F\left(1+(\xi/F)^2\right)} d\xi$, then:*

$$s_{\mu,\epsilon} = O\left(\frac{FT}{\sqrt{\epsilon}} + \frac{1}{\sqrt{\epsilon}}\right).$$

Proof. Similar to the proof of Theorem B.4.2, we define the interval $I_h = \left\{\xi \in \mathbb{R} : |\xi| \leq F/\sqrt{\epsilon}\right\}$. We decompose μ into two measures μ_h and μ_t as follows:

$$\mu_h(A) = \mu(A \cap I_h), \quad \text{and} \quad \mu_t(A) = \mu(A - A \cap I_h).$$

Since $\mu = \mu_h + \mu_t$, by Lemma B.4.1, $s_{\mu,\epsilon} \leq s_{\mu_t,\epsilon} + s_{\mu_h,\epsilon}$. For μ_t we have:

$$\begin{aligned}
\operatorname{tr}\left(\mathscr{K}_{\mu_t}\right) = \mu_t(\mathbb{R}) &= \frac{1}{\pi F} \int_{|\xi|>F/\sqrt{\epsilon}} \frac{1}{1+(\xi/F)^2}\, d\xi \\
&= \frac{2}{\pi} \int_{1/\sqrt{\epsilon}}^{\infty} \frac{1}{1+\xi^2}\, d\xi \\
&\leq \frac{2}{\pi} \int_{1/\sqrt{\epsilon}}^{\infty} \frac{1}{\xi^2}\, d\xi = \frac{2\sqrt{\epsilon}}{\pi}.
\end{aligned}$$

As in (B.46) we can thus bound:

$$s_{\mu_t,\epsilon} \leq \operatorname{tr}(\mathscr{K}_{\mu_t})/\epsilon = O\left(1/\sqrt{\epsilon}\right). \tag{B.48}$$

We next bound the statistical dimension of μ_h. Let $\tilde{\mu}_h$ be a uniform measure on I_h with $d\mu(\xi) = \frac{1}{\pi F}$ for all $\xi \in I_h$. As in the proof of Theorem B.4.2, $d\tilde{\mu}_h(\xi) \geq d\mu_h(\xi)$ for all $\xi \in I_h$ which gives that $K_{\mu_h} \preceq K_{\tilde{\mu}_h}$ and so $s_{\mu_h,\epsilon} < s_{\tilde{\mu}_h,\epsilon}$.

Let $\gamma \stackrel{\text{def}}{=} \tilde{\mu}_h(\mathbb{R}) = \frac{2}{\pi\sqrt{\epsilon}}$. By Lemma B.4.2, $s_{\tilde{\mu},\epsilon} = s_{(\tilde{\mu}/\gamma),(\epsilon/\gamma)}$. Since $\tilde{\mu}/\gamma$ is a uniform probability measure on I_h, we can invoke Theorem B.3.1 to give:

$$\begin{aligned}
s_{\mu_h,\epsilon} \leq s_{\tilde{\mu}_h,\epsilon} = s_{(\tilde{\mu}_h/\gamma),(\epsilon/\gamma)} &= O\left(\frac{FT}{\sqrt{\epsilon}} + \log(\gamma/\epsilon)\right) \\
&= O\left(\frac{FT}{\sqrt{\epsilon}} + \log(1/\epsilon)\right),
\end{aligned} \tag{B.49}$$

where the last equality follows from the fact that $\gamma = O(1/\sqrt{\epsilon})$. Combining (B.48) and (B.49) and applying Lemma B.4.1 we have:

$$s_{\mu,\epsilon} \leq s_{\mu_t,\epsilon} + s_{\mu_h,\epsilon} = O\left(\frac{1}{\sqrt{\epsilon}} + \frac{FT}{\sqrt{\epsilon}} + \log(1/\epsilon)\right) = O\left(\frac{FT}{\sqrt{\epsilon}} + \frac{1}{\sqrt{\epsilon}}\right),$$

which completes the theorem. □

B.5 Kernel Computation for Common Fourier Constraints

Algorithm 13 and the corresponding Theorem 3.2.3 assume the ability to compute the kernel function $k_\mu(t_1,t_2)=\int_{\xi\in\mathbb{R}} e^{-2\pi i(t_1-t_2)}d\mu(\xi)$. In this section we give closed form expressions for kernel functions of popular measures μ, including all those whose statistical dimension we bound in Appendix B.4.

Bandlimited Fourier Constraint: When μ is the uniform measure on frequencies in $[-F,F]$, k_μ is the *sinc* kernel:

$$k_\mu(t_1,t_2)=\frac{1}{2F}\int_{-F}^{F} e^{-2\pi i(t_1-t_2)\xi}d\xi=\frac{\sin\left(2\pi F(t_1-t_2)\right)}{2\pi F(t_1-t_2)}.$$

Multiband Fourier Constraint: Consider a set of s disjoint frequency bands, $I_1,I_2,\cdots,I_s$, where $I_j=[c_j-F_j,c_j+F_j]$. Let μ be the uniform measure on $I_1\cup I_2\cup\cdots\cup I_s$. Then we have:

$$\begin{aligned}k_\mu(t_1,t_2)&=\frac{1}{2\sum_{j=1}^{s}F_j}\cdot\sum_{j=1}^{s}e^{-2\pi i c_j(t_1-t_2)\xi}\int_{-F_j}^{F_j}e^{-2\pi i(t_1-t_2)\xi}d\xi\\&=\frac{\sum_{j=1}^{s}e^{-2\pi i c_j(t_1-t_2)}\cdot\sin\left(2\pi F_j(t_1-t_2)\right)}{2\pi\sum_{j=1}^{s}F_j(t_1-t_2)}.\end{aligned}$$

Gaussian Fourier Constraint: When μ induces the Gaussian probability distribution with standard deviation F defined by $d\mu(\xi)=\frac{1}{\sqrt{2\pi F^2}}e^{-\xi^2/2F^2}d\xi$, then k_μ is the *Gaussian* kernel:

$$\begin{aligned}k_\mu(t_1,t_2)&=\frac{1}{\sqrt{2\pi F^2}}\cdot\int_{\xi\in\mathbb{R}}e^{-2\pi i(t_1-t_2)\xi}e^{-\xi^2/2F^2}d\xi\\&=e^{-2\pi^2F^2(t_1-t_2)^2}.\end{aligned}$$

Cauchy-Lorentz Fourier Constraint: When μ induces the Cauchy-Lorentz probability density with scale parameter F defined by $d\mu(\xi)=\frac{1}{\pi F\left(1+(\xi/F)^2\right)}\cdot d\xi$, k_μ is the so-called *Laplacian* kernel:

$$\begin{aligned}k_\mu(t_1,t_2)&=\int_{\xi\in\mathbb{R}}e^{-2\pi i(t_1-t_2)}\frac{1}{\pi F\left(1+(\xi/F)^2\right)}d\xi\\&=e^{-2\pi F|t_1-t_2|}.\end{aligned}$$

C Tight Characterization of the Gaussian Kernel Leverage Scores

C.1 Properties of Fourier Transform and Gaussian Distribution

Our upper and lower bound analysis of the Gaussian kernel leverage function relies predominantly on Fourier analysis and properties of the Gaussian distribution. In this section we introduce some additional notation and state some useful facts about these.

C.1.1 Properties of Fourier transform

Definition C.1.1 (Fourier transform)**.** The *Fourier transform* of a continuous function $f : \mathbb{R}^d \to \mathbb{C}$ in $L_1(\mathbb{R}^n)$ is defined to be the function $\mathscr{F} f : \mathbb{R}^d \to \mathbb{C}$ as follows:

$$[\mathscr{F} f](\boldsymbol{\xi}) = \int_{\mathbb{R}^d} f(\mathbf{t}) e^{-2\pi i \mathbf{t}^T \boldsymbol{\xi}} \, d\mathbf{t}.$$

We also sometimes use the notation $\hat{f}$ for the Fourier transform of f. We often informally refer to f as representing the function in *time domain* and $\hat{f}$ as representing the function in *frequency domain*.
The original function f can also be obtained from $\hat{f}$ by the *inverse Fourier transform*:

$$f(\mathbf{t}) = \int_{\mathbb{R}^d} \hat{f}(\boldsymbol{\xi}) e^{2\pi i \boldsymbol{\xi}^T \mathbf{t}} \, d\boldsymbol{\xi}$$

Definition C.1.2 (Convolution)**.** The *convolution* of two functions $f : \mathbb{R}^d \to \mathbb{C}$ and $g : \mathbb{R}^d \to \mathbb{C}$ is defined to be the function $(f * g) : \mathbb{R}^d \to \mathbb{C}$ given by

$$(f * g)(\boldsymbol{\eta}) = \int_{\mathbb{R}^d} f(\mathbf{t}) g(\boldsymbol{\eta} - \mathbf{t}) \, d\mathbf{t}.$$

The convolution theorem shows that the Fourier transform of the convolution of two functions is simply the product of the individual Fourier transforms:

Claim C.1.1 (Convolution Theorem)**.** *Given functions $f : \mathbb{R}^d \to \mathbb{C}$ and $g : \mathbb{R}^d \to \mathbb{C}$ whose convo-*

lution is $h = f * g$, we have $\hat{h}(\boldsymbol{\xi}) = \hat{f}(\boldsymbol{\xi}) \cdot \hat{g}(\boldsymbol{\xi})$ for all $\boldsymbol{\xi} \in \mathbb{R}^d$.

We now define the *rectangle function* and *normalized sinc function*, which we use extensively in our analysis.

Definition C.1.3 (Rectangle Function)**.** We define the 1-dimensional *rectangle function* $\mathrm{rect}_{1,a} : \mathbb{R} \to \mathbb{C}$ as

$$\mathrm{rect}_{1,a}(x) = \begin{cases} 0 & \text{if } |x| > a/2 \\ \frac{1}{2} & \text{if } |x| = a/2 \\ 1 & \text{if } |x| < a/2 \end{cases}.$$

For any $d > 1$, we define the d-dimensional *rectangle function* $\mathrm{rect}_{d,a} : \mathbb{R}^d \to \mathbb{C}$ as

$$\mathrm{rect}_{d,a}(\mathbf{x}) = \prod_{j=1}^{d} \mathrm{rect}_{1,a}(x_j).$$

If d is understood from context, we often omit d and write rect_a. Moreover, if $a = 1$ (and d is understood from context), we often omit all subscripts and simply write rect.

Definition C.1.4 (Normalized Sinc Function)**.** We define the d-dimensional *normalized sinc function* $\mathrm{sinc}_d : \mathbb{R}^d \to \mathbb{C}$ as

$$\mathrm{sinc}_d(\mathbf{x}) = \prod_{j=1}^{d} \frac{\sin(\pi x_j)}{\pi x_j}.$$

We often omit the subscript and simply write sinc.

It is well known that the Fourier transform of the rectangle function (with $a = 1$) is the normalized sinc function:

$$\mathscr{F}[\mathrm{rect}_d] = \mathrm{sinc}_d.$$

We use δ_d to denote the d-dimensional *Dirac delta function*. The Dirac delta function satisfies the following useful property for any function f:

$$\int_{\mathbb{R}^d} f(\mathbf{x}) \delta_d(\mathbf{x} - \mathbf{a})\, d\mathbf{x} = f(\mathbf{a}),$$

i.e. the integral of a function multiplied by a shifted Dirac delta functions picks out the value of the function at a particular point. Therefore,

$$[\mathscr{F}\delta_d](\boldsymbol{\xi}) = \int_{\mathbb{R}^d} e^{-2\pi i \mathbf{t}^\top \boldsymbol{\xi}} \cdot \delta_d(\mathbf{t})\, d\mathbf{t} = e^{-2\pi i \cdot 0^\top \cdot \boldsymbol{\xi}} = 1$$

for all $\boldsymbol{\xi}$. Similarly, the Fourier transform of a shifted delta function is as follows:

$$[\mathscr{F}\delta(\cdot - \mathbf{a})](\boldsymbol{\xi}) = \int_{\mathbb{R}^d} e^{-2\pi i \mathbf{t}^\top \boldsymbol{\xi}} \cdot \delta_d(\mathbf{t} - \mathbf{a})\, d\mathbf{t} = e^{-2\pi i \mathbf{a}^\top \boldsymbol{\xi}}.$$

Moreover, convolving a function by a shifted delta function results in a shift of the original function:

$$[f * \delta_d(\cdot - \mathbf{a})](\mathbf{x}) = f(\mathbf{x} - \mathbf{a}).$$

Thus, by the convolution theorem, we obtain the following identity:

Claim C.1.2. *Given a function $f : \mathbb{R}^d \to \mathbb{C}$, we have*

$$[\mathscr{F} f(\cdot - \mathbf{a})](\boldsymbol{\xi}) = [\mathscr{F}(f * \delta_d(\cdot - \mathbf{a}))](\boldsymbol{\xi}) = \hat{f}(\boldsymbol{\xi}) \cdot e^{-2\pi i \mathbf{a}^T \boldsymbol{\xi}}.$$

Similarly,

Claim C.1.3. *Given a function $f : \mathbb{R}^d \to \mathbb{C}$, we have*

$$\left[\mathscr{F}\left(f(\mathbf{x}) \cdot e^{2\pi i \mathbf{a}^T \mathbf{x}}\right)\right](\boldsymbol{\xi}) = \hat{f}(\boldsymbol{\xi} - \mathbf{a}).$$

Finally, we introduce a useful function known as the *Dirac comb function*:

Definition C.1.5. The d-dimensional *Dirac comb function* with period T is defined as

$$f(\mathbf{x}) = \sum_{\mathbf{j} \in \mathbb{Z}^d} \delta(\mathbf{x} - \mathbf{j}T).$$

It is a standard fact that the Fourier transform of a Dirac comb function is another Dirac comb function which is scaled and has the inverse period:

Claim C.1.4. *Let $f(\mathbf{x}) = \sum_{\mathbf{j} \in \mathbb{Z}^d} \delta(\mathbf{x} - \mathbf{j}T)$ be the d-dimensional Dirac comb function with period T. Then,*

$$[\mathscr{F} f](\boldsymbol{\xi}) = \frac{1}{T^d} \sum_{\mathbf{j} \in \mathbb{Z}^d} \delta\left(\boldsymbol{\xi} - \frac{\mathbf{j}}{T}\right).$$

We use the Dirac comb function in our lower bound constructions.

Claim C.1.5. *Given a function $f : \mathbb{R}^d \to \mathbb{C}$, we have:*

$$\mathscr{F}\left[f(\cdot) \sum_{\mathbf{j} \in \mathbb{Z}^d} \delta_d(\cdot - T\mathbf{j})\right](\boldsymbol{\xi}) = \sum_{\mathbf{j} \in \mathbb{Z}^d} T^{-d} [\mathscr{F} f](\boldsymbol{\xi} - T^{-1}\mathbf{j}). \tag{C.1}$$

C.1.2 Properties of Gaussian distributions

The following is a standard fact about the cumulative distribution function of the standard Gaussian distribution:

Claim C.1.6 (Feller (2008))**.** *For any $x > 0$, we have*

$$\frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2}\, dt \leq \frac{e^{-x^2/2}}{x\sqrt{2\pi}}.$$

Moreover, as a direct consequence, for any $\sigma, x > 0$, we have that

$$\frac{1}{\sqrt{2\pi}\sigma}\int_x^\infty e^{-t^2/2\sigma^2}\,dt \le \frac{\sigma e^{-x^2/2\sigma^2}}{x\sqrt{2\pi}}.$$

Also, if $x \ge 1$, then

$$\left(\frac{1}{x}-\frac{1}{x^3}\right)\cdot\frac{1}{\sqrt{2\pi}}e^{-x^2/2} \le \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-t^2}\,dt.$$

Next, we prove the following claim, which provides tail bounds for modified Gaussians:

Claim C.1.7. *We have the following results:*

1. *For any $x > 0$ and $d = 1$, $\int_x^\infty t^d e^{-t^2/2}\,dt = e^{-x^2/2}$.*
2. *For any $x > 0$ and odd integer $d > 1$, $\int_x^\infty t^d e^{-t^2/2}\,dt \ge (d-1)(d-3)\cdots 2\cdot e^{-x^2/2}$.*
3. *For any $x > 0$ and even integer $d > 1$, $\int_x^\infty t^d e^{-t^2/2}\,dt \ge (d-1)(d-3)\cdots 3\cdot 1\cdot x e^{-x^2/2}$.*
4. *For any $x > 0$ and integer $d \ge 1$, $\int_x^\infty t^d e^{-t^2/2}\,dt \ge x^{d-1}e^{-x^2/2}$.*

Proof. Part (1) is simple calculation.

If d is odd, say $d = 2a+1$, then by repeated use of integration by parts,

$$\int_x^\infty t^d e^{-t^2/2}\,dt = \sum_{j=0}^{a-1}\left(\prod_{k=1}^{j}(d-(2k-1))\right)x^{d-(2j+1)}e^{-x^2/2} + (d-1)(d-3)\cdots 2\int_x^\infty te^{-t^2/2}\,dt \tag{C.2}$$

$$\begin{aligned}&\ge (d-1)(d-3)\cdots 2\int_x^\infty te^{-t^2/2}\,dt\\ &= (d-1)(d-3)\cdots 2\cdot e^{-x^2/2},\end{aligned}$$

which establishes part (2).

On the other hand, if d is even, say $d = 2a$, then we have

$$\int_x^\infty t^d e^{-t^2/2}\,dt = \sum_{j=0}^{a-1}\left(\prod_{k=1}^{j}(d-(2k-1))\right)x^{d-(2j+1)}e^{-x^2/2} + (d-1)(d-3)\cdots 3\int_x^\infty e^{-t^2/2}\,dt \tag{C.3}$$

$$\ge (d-1)(d-3)\cdots 3\cdot 1\cdot xe^{-x^2/2},$$

which establishes part (3) of the claim.

Finally, note that (C.2) and (C.3) are both bounded from below by $x^{d-1}e^{-x^2/2}$ (since this is the first term of the summation in both expressions), which establishes part (4). □

We also need the following property about Gaussian random variables.

Claim C.1.8. *Let $t \geq 13$, and $a_1, a_2, \ldots, a_t$ be sampled independently according to the Gaussian distribution given by probability density function $\frac{1}{\sqrt{2\pi}} e^{-x^2/2}$. Let $a^* = \max_{1 \leq j \leq t} |a_j|$. Then,*

$$\Pr\left[\frac{1}{\sqrt{2\pi}} e^{-a^{*2}/2} \leq \frac{2\sqrt{2\ln t}}{t}\right] \geq 1 - e^{-1} \geq \frac{1}{2}.$$

Proof. Choose q_1 such that

$$\int_{q_1}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2}\, dx = \frac{1}{t}. \tag{C.4}$$

Note that by Claim C.1.6, we have

$$\frac{1}{\sqrt{2\pi}} \int_{\sqrt{2\ln t}}^{\infty} e^{-x^2/2}\, dx \leq \frac{1}{\sqrt{2\pi}\, t \sqrt{2\ln t}} \leq \frac{1}{t}.$$

Thus, $q_1 \leq \sqrt{2\ln t}$.

Also, since $\frac{1}{t} \leq \frac{1}{13}$, we have that $q_1 \geq \sqrt{2}$. Thus, by another application of Claim C.1.6,

$$\frac{1}{t} = \frac{1}{\sqrt{2\pi}} \int_{q_1}^{\infty} e^{-x^2/2}\, dx \geq \left(\frac{1}{q_1} - \frac{1}{q_1^3}\right) \frac{1}{\sqrt{2\pi}} e^{-q_1^2/2} \geq \frac{1}{2q_1} \cdot \frac{1}{\sqrt{2\pi}} e^{-q_1^2/2},$$

and so,

$$\frac{1}{\sqrt{2\pi}} e^{-q_1^2/2} \leq \frac{2q_1}{t} \leq \frac{2\sqrt{2\ln t}}{t}.$$

Therefore,

$$\begin{aligned}
\Pr\left[\frac{1}{\sqrt{2\pi}} e^{-a^{*2}/2} \leq \frac{2\sqrt{2\ln t}}{t}\right] &\geq \Pr\left[\frac{1}{\sqrt{2\pi}} e^{-a^{*2}/2} \leq \frac{1}{\sqrt{2\pi}} e^{-q_1^2/2}\right] \\
&= \Pr\left[a^* \geq q_1\right] \\
&= 1 - \left(1 - \frac{1}{t}\right)^t \\
&\geq 1 - \frac{1}{e} \geq \frac{1}{2},
\end{aligned}$$

as desired. □

C.2 Tight Upper Bound on the Gaussian Kernel Leverage Scores

It is easy to verify that if we shift all points by the same constant vector, the leverage function stays the same (the reason is that $\mathbf{K}$ is shift invariant, while the shift corresponds to a phase shift in $\mathbf{z}(\boldsymbol{\eta})$ and a reverse phase shift in $\mathbf{z}(\boldsymbol{\eta})^*$). This ensures that without loss of generality we can assume $\mathbf{x}_1, \ldots, \mathbf{x}_n \in [-R, R]^d$.

Recall from Lemma 4.7.1 that

$$\tau_\lambda(\boldsymbol{\eta}) = \min_{y \in L_2(\mu)} \lambda^{-1} \left\| \boldsymbol{\Phi} y - \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta}) \right\|_2^2 + \left\| y \right\|_{L_2(\mu)}^2. \tag{C.5}$$

To upper bound $\tau_\lambda(\boldsymbol{\eta})$ for any $\boldsymbol{\eta} \in \mathbb{R}^d$, we exhibit a test function, $y_{\boldsymbol{\eta}}(\cdot)$, and compute the argument in (C.5). As discussed in Section 4.7.2, $y_{\boldsymbol{\eta}}(\cdot)$ will be a 'softened spike' given by:

Definition C.2.1 (Softened spike function)**.** For any $\boldsymbol{\eta} \in \mathbb{R}^d$, and any $u > 0$ define $y_{\boldsymbol{\eta},u} : \mathbb{R}^d \to \mathbb{R}$ as follows:

$$y_{\boldsymbol{\eta},u}(\mathbf{t}) = \frac{\sqrt{p(\boldsymbol{\eta})}}{p(\mathbf{t})} \cdot e^{-u^2\|\mathbf{t}-\boldsymbol{\eta}\|_2^2/4} \cdot v^d \operatorname{sinc}\big(v(\mathbf{t}-\boldsymbol{\eta})\big) \tag{C.6}$$

where $v = 2\left(R + u\sqrt{\ln n_\lambda}\right)$.

The reweighted function $g_{\boldsymbol{\eta},u}(\mathbf{t}) = p(\mathbf{t}) \cdot y_{\boldsymbol{\eta},u}(\mathbf{t})$ is just a d-dimensional Gaussian with standard deviation $\Theta(1/u)$ multiplied by a sinc function with width $\tilde{O}\left(\frac{1}{u+R}\right)$, both centered at $\boldsymbol{\eta}$. Taking the Fourier transform of this function yields a Gaussian with standard deviation $\Theta(u)$ convolved with a box of width $\tilde{O}(u) + R$. The box is wide enough to cover nearly all the mass of the Gaussian when centered between $[-R, R]^d$, and so the Fourier transform is nearly identically 1 on the range $[-R, R]^d$. Shifting by $\boldsymbol{\eta}$, means that it is very close to a pure cosine wave with frequency $\boldsymbol{\eta}$ on this range, and hence makes the first term of (C.5) small.

C.2.1 Bounding $\lambda^{-1} \left\| \boldsymbol{\Phi} y_{\boldsymbol{\eta},u} - \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta}) \right\|_2^2$

Lemma C.2.1 (Test function Fourier transform bound)**.** *For any integer n, every parameter $0 < \lambda \leq \frac{n}{2}$ and every $u > 0$ and any $\boldsymbol{\eta} \in \mathbb{R}^d$, and any kernel density function $p(\boldsymbol{\eta})$ and $d \leq 10 n_\lambda$ if $\mathbf{x}_j \in [-R, +R]^d$ for all $j \in [n]$, then:*

$$\lambda^{-1} \left\| \boldsymbol{\Phi} y_{\boldsymbol{\eta},u} - \sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta}) \right\|_2^2 = \frac{1}{\lambda} \sum_{j=1}^{n} \left| \hat{g}_{\boldsymbol{\eta},u}(\mathbf{x}_j) - \sqrt{p(\boldsymbol{\eta})} \cdot \mathbf{z}(\boldsymbol{\eta})_j \right|^2 \leq p(\boldsymbol{\eta}),$$

where $g_{\boldsymbol{\eta},u}(\mathbf{t}) \stackrel{\text{def}}{=} p(\mathbf{t}) y_{\boldsymbol{\eta},u}(\mathbf{t})$.

Proof. By $g_{\boldsymbol{\eta},u}(\mathbf{t}) = p(\mathbf{t}) y_{\boldsymbol{\eta},u}(\mathbf{t}) = \sqrt{p(\boldsymbol{\eta})} e^{-u^2\|\mathbf{t}-\boldsymbol{\eta}\|_2^2/4} \cdot v^d \operatorname{sinc}\big(v(\mathbf{t}-\boldsymbol{\eta})\big)$, we have:

$$\begin{aligned}
\hat{g}_{\boldsymbol{\eta},u}(\mathbf{x}_j) &= \sqrt{p(\boldsymbol{\eta})} \int_{\mathbb{R}^d} e^{-2\pi i \mathbf{t}^\top \mathbf{x}_j} e^{-u^2\|\mathbf{t}-\boldsymbol{\eta}\|_2^2/4} \cdot v^d \operatorname{sinc}\big(v(\mathbf{t}-\boldsymbol{\eta})\big) d\mathbf{t} \\
&= \sqrt{p(\boldsymbol{\eta})} e^{-2\pi i \mathbf{x}_j^T \boldsymbol{\eta}} \cdot \mathscr{F}\left[e^{-u^2\|\mathbf{t}\|_2^2/4} \cdot v^d \operatorname{sinc}(v\mathbf{t}) \right](\mathbf{x}_j) \\
&= \sqrt{p(\boldsymbol{\eta})} \cdot \mathbf{z}(\boldsymbol{\eta})_j \cdot h(\mathbf{x}_j),
\end{aligned} \tag{C.7}$$

where $h(\mathbf{x}) = \left(\frac{2\sqrt{\pi}}{u}\right)^d e^{-4\pi^2\|\mathbf{x}\|_2^2/u^2} * \operatorname{rect}_v(\mathbf{x})$ by the convolution theorem (Claim C.1.1), $\mathscr{F}\left[e^{-u^2\|\mathbf{t}\|_2^2/4} \right] =$

$\left(\frac{2\sqrt{\pi}}{u}\right)^d e^{-4\pi^2\|\mathbf{x}\|_2^2/u^2}$, and $\mathscr{F}\left[v^d \operatorname{sinc}(v\mathbf{t})\right] = \operatorname{rect}_v(\mathbf{x})$.

For every $\mathbf{x} \in [-R,R]^d$, by Claim C.1.6 and the fact that $v = 2R + 2u\sqrt{\ln n_\lambda}$, we have:

$$\begin{aligned} h(\mathbf{x}) &= \int_{\mathbf{y}-\mathbf{x}\in[-v/2,+v/2]^d} \left(\frac{2\sqrt{\pi}}{u}\right)^d e^{-4\pi^2\|\mathbf{y}\|_2^2/u^2}\, d\mathbf{y} \\ &\geq \left(1 - 2\int_{v/2-R}^{\infty} \frac{2\sqrt{\pi}}{u} e^{-4\pi^2 y_1^2/u^2}\, dy_1\right)^d, \end{aligned}$$

where y_1 is a scalar variable. Hence by Claim C.1.6 we have the following:

$$\begin{aligned} h(\mathbf{x}) &\geq 1 - 2d\int_{v/2-R}^{\infty} \frac{2\sqrt{\pi}}{u} e^{-4\pi^2 y_1^2/u^2}\, dy_1 \\ &\geq 1 - \frac{d}{2\pi^{3/2}} \cdot \frac{u}{v/2-R} e^{-4\pi^2(v/2-R)^2/u^2} \\ &\geq 1 - \frac{1}{n_\lambda} \qquad (\text{since } d \leq 10 n_\lambda). \end{aligned}$$

Additionally, because $e^{-4\pi^2\|\mathbf{x}\|_2^2/u^2}$ is a positive function, $h(\mathbf{x}) \leq \int_{\mathbb{R}^d} \left(\frac{2\sqrt{\pi}}{u}\right)^d e^{-4\pi^2\|\mathbf{x}\|_2^2/u^2} d\mathbf{x} = 1$ for all $\mathbf{x}$. Plugging into (C.7) gives

$$\begin{aligned} \left|\hat{g}_{\boldsymbol{\eta},u}(\mathbf{x}_j) - \sqrt{p(\boldsymbol{\eta})}\cdot \mathbf{z}(\boldsymbol{\eta})_j\right|^2 &= p(\boldsymbol{\eta})\left|h(\mathbf{x}_j) - 1\right|^2 \\ &\leq \frac{p(\boldsymbol{\eta})}{n_\lambda^2} \leq \frac{p(\boldsymbol{\eta})}{n_\lambda}, \end{aligned}$$

and so,

$$\frac{1}{\lambda}\sum_{j=1}^{n} \left|\hat{g}(\mathbf{x}_j) - \sqrt{p(\boldsymbol{\eta})}\cdot \mathbf{z}(\boldsymbol{\eta})_j\right|^2 \leq \frac{n}{\lambda}\cdot\frac{p(\boldsymbol{\eta})}{n_\lambda} = p(\boldsymbol{\eta}),$$

proving the lemma. □

C.2.2 Bounding $\left\|y_{\boldsymbol{\eta},u}\right\|_{L_2(\mu)}^2$

Having established Lemma C.2.1, showing that the weighted Fourier transform of $y_{\boldsymbol{\eta},u}$ is close to $\sqrt{p(\boldsymbol{\eta})}\mathbf{z}(\boldsymbol{\eta})$, bounding the leverage function reduces to bounding the norm of the test function. To that effect, we show the following:

Lemma C.2.2 (Test Function ℓ_2 Norm Bound)**.** *For any integer n, any parameter $0 < \lambda \leq \frac{n}{2}$, every $\boldsymbol{\eta} \in \mathbb{R}^d$ with $\|\boldsymbol{\eta}\|_\infty \leq 10\sqrt{\ln n_\lambda}$, and every $200\ln n_\lambda \leq u \leq 10 n_\lambda$, if $y_{\boldsymbol{\eta},u}(\mathbf{t})$ is defined as in* (C.6)*, as per Definition C.2.1, then we have*

$$\left\|y_{\boldsymbol{\eta},u}\right\|_{L_2(d\mu)}^2 \leq \left(6.2R + 6.2u\sqrt{\ln n_\lambda}\right)^d. \tag{C.8}$$

We first prove the following claim:

Claim C.2.1. *For every $0 < \lambda \le \frac{n}{2}$, every $c > 0$, every $\boldsymbol{\eta} \in \mathbb{R}^d$ with $\|\boldsymbol{\eta}\|_\infty \le 10\sqrt{\ln n_\lambda}$, every $\mathbf{t} \in \mathbb{R}^d$ such that $\|\mathbf{t}-\boldsymbol{\eta}\|_\infty \le \frac{c\sqrt{\ln n_\lambda}}{\sigma}$, and every $\sigma \ge 10c\ln n_\lambda$, we have $e^{\frac{\|\mathbf{t}\|_2^2}{2}-\frac{\|\boldsymbol{\eta}\|_2^2}{2}} \le 3^d$.*

Proof. Let $\boldsymbol{\Delta} = \mathbf{t}-\boldsymbol{\eta}$. Then, note that $\|\boldsymbol{\Delta}\|_\infty \le \frac{c\sqrt{\ln n_\lambda}}{\sigma}$, and so,

$$\begin{aligned} e^{\frac{\|\mathbf{t}\|_2^2}{2}-\frac{\|\boldsymbol{\eta}\|_2^2}{2}} &= e^{\boldsymbol{\Delta}^\top\boldsymbol{\eta}+\frac{\|\boldsymbol{\Delta}\|_2^2}{2}} \\ &\le e^{d\cdot\|\boldsymbol{\Delta}\|_\infty\cdot\|\boldsymbol{\eta}\|_\infty}\cdot e^{d\cdot\|\boldsymbol{\Delta}\|_\infty^2} \\ &\le e^d\cdot e^{d\left(c\sqrt{\log n_\lambda}/\sigma\right)^2} \le 3^d, \qquad (\text{since } \sigma \ge 10c\ln n_\lambda \text{ and } n_\lambda \ge 2). \end{aligned}$$

□

Now, we are ready to prove Lemma C.2.2.

Proof of Lemma C.2.2. Recall that $p(\mathbf{t}) = \frac{1}{(\sqrt{2\pi})^d}e^{-\|\mathbf{t}\|_2^2/2}$. We calculate:

$$\int_{\mathbb{R}^d}\left|y_{\boldsymbol{\eta},u}(\mathbf{t})\right|^2 d\mu(\mathbf{t}) = p(\boldsymbol{\eta})\int_{\mathbb{R}^d}\left(\sqrt{2\pi}\right)^d e^{\|\mathbf{t}\|_2^2/2}\cdot e^{-u^2\|\mathbf{t}-\boldsymbol{\eta}\|_2^2/2}\cdot v^{2d}\operatorname{sinc}\left(v(\mathbf{t}-\boldsymbol{\eta})\right)^2 d\mathbf{t}$$

Hence, it is enough to upper bound the following integral:

$$\begin{aligned} &\int_{\mathbb{R}^d} e^{\|\mathbf{t}\|_2^2/2}\cdot e^{-u^2\|\mathbf{t}-\boldsymbol{\eta}\|_2^2/2}\cdot\operatorname{sinc}\left(v(\mathbf{t}-\boldsymbol{\eta})\right)^2 d\mathbf{t} \\ &\quad= \prod_{l=1}^d\int_{\mathbb{R}} e^{|t_l|^2/2}\cdot e^{-u^2|t_l-\eta_l|^2/2}\cdot\operatorname{sinc}\left(v(t_l-\eta_l)\right)^2 dt_l \end{aligned} \tag{C.9}$$

We proceed by upper bounding the one dimensional integral along some fixed coordinate l:

$$\begin{aligned} &\int_{\mathbb{R}} e^{|t_l|^2/2}\cdot e^{-u^2|t_l-\eta_l|^2/2}\cdot\operatorname{sinc}\left(v(t_l-\eta_l)\right)^2 dt_l \\ &\quad= \int_{|t_l-\eta_l|\le\frac{20\sqrt{\ln n_\lambda}}{u}} e^{|t_l|^2/2}\cdot e^{-u^2|t_l-\eta_l|^2/2}\cdot\operatorname{sinc}\left(v(t_l-\eta_l)\right)^2 dt_l \\ &\quad+ \int_{|t_l-\eta_l|\ge\frac{20\sqrt{\ln n_\lambda}}{u}} e^{|t_l|^2/2}\cdot e^{-u^2|t_l-\eta_l|^2/2}\cdot\operatorname{sinc}\left(v(t_l-\eta_l)\right)^2 dt_l \end{aligned} \tag{C.10}$$

For the integral over the region $|t_l - \eta_l| \geq \frac{20\sqrt{\ln n_\lambda}}{u}$ we have:

$$\begin{aligned}
&\int_{|t_l-\eta_l|\geq \frac{20\sqrt{\ln n_\lambda}}{u}} e^{|t_l|^2/2} \cdot e^{-u^2|t_l-\eta_l|^2/2} \cdot \operatorname{sinc}\big(v(t_l-\eta_l)\big)^2 \, dt_l \\
&\quad \leq \left(v\frac{20\sqrt{\ln n_\lambda}}{u}\right)^{-2} \int_{|t_l-\eta_l|\geq \frac{20\sqrt{\ln n_\lambda}}{u}} e^{t_l^2/2} \cdot e^{-u^2(t_l-\eta_l)^2/2} \, dt_l \\
&\quad \leq \frac{n_\lambda}{v} \int_{|t_l-\eta_l|\geq \frac{20\sqrt{\ln n_\lambda}}{u}} e^{t_l^2/2} \cdot e^{-u^2(t_l-\eta_l)^2/2} \, dt_l. \qquad \text{(C.11)}
\end{aligned}$$

The first inequality is because by definition of sinc, for all $|t_l - \eta_l| \geq \frac{20\sqrt{\ln n_\lambda}}{u}$:

$$\operatorname{sinc}\big(v(t_l-\eta_l)\big)^2 = \left|\frac{\sin(\pi v(t_l-\eta_l))}{\pi v(t_l-\eta_l)}\right|^2 \leq \left(v\frac{20\sqrt{\log n_\lambda}}{u}\right)^{-2}$$

The last inequality in (C.11) due to the fact that:

$$\begin{aligned}
\left(v\frac{20\sqrt{\ln n_\lambda}}{u}\right)^{-2} &= \frac{1}{v} \cdot \frac{u^2}{400 \cdot v \ln n_\lambda} \\
&\leq \frac{1}{v} \cdot \frac{u}{800 \cdot \ln^{1.5} n_\lambda} \quad \text{(since } v \geq 2u\sqrt{\ln n_\lambda}\text{, see Definition C.2.1)} \\
&\leq \frac{n_\lambda}{v} \qquad\qquad \text{(since } u \leq 10 n_\lambda\text{)},
\end{aligned}$$

Now note that, using the inequality $t_l^2 \leq 2(t_l-\eta_l)^2 + 2\eta_l^2$, for all $|t_l - \eta_l| \geq \frac{20\sqrt{\ln n_\lambda}}{u}$:

$$\begin{aligned}
t_l^2 &\leq 2(t_l-\eta_l)^2 + 2\eta_l^2 \\
&\leq 2(t_l-\eta_l)^2 + 200 \log n_\lambda \qquad \text{(by the assumption } \|\boldsymbol{\eta}\|_\infty \leq 10\sqrt{\ln n_\lambda}\text{)} \\
&\leq 2(t_l-\eta_l)^2 + u^2(t_l-\eta_l)^2/2 \quad \text{(since } |t_l-\eta_l| \geq \frac{20\sqrt{\ln n_\lambda}}{u}\text{)} \\
&\leq \frac{2}{3}u^2(t_l-\eta_l)^2
\end{aligned}$$

where the last inequality follows from $u \geq 200 \log n_\lambda \geq \sqrt{12}$ (because $n_\lambda \geq 2$). Hence,

$$\begin{aligned}
\frac{n_\lambda}{v} \int_{|t_l-\eta_l|\geq \frac{20\sqrt{\ln n_\lambda}}{u}} e^{t_l^2/2} \cdot e^{-(t_l-\eta_l)^2u^2/2} dt_l &\leq \frac{n_\lambda}{v} \int_{|t_l-\eta_l|\geq \frac{20\sqrt{\ln n_\lambda}}{u}} e^{-(t_l-\eta_l)^2u^2/6} dt_l \\
&= \frac{n_\lambda}{v} \int_{|t'|\geq \frac{20\sqrt{\ln n_\lambda}}{u}} e^{-(t')^2u^2/6} dt' \\
&\leq \frac{1}{10v} \qquad \text{(C.12)}
\end{aligned}$$

The last inequality follows from Claim C.1.6 along with the assumption $n_\lambda \geq 2$.

Now, we bound the first integral on the right hand side of (C.10):

$$\begin{aligned}\int_{|t-\eta_l|\le\frac{20\sqrt{\ln n_\lambda}}{u}} e^{\frac{|t_l|^2}{2}} e^{-\frac{u^2|t_l-\eta_l|^2}{2}} \cdot \operatorname{sinc}\big(v(t_l-\eta_l)\big)^2 dt_l &\le \int_{|t-\eta_l|\le\frac{20\sqrt{\ln n_\lambda}}{u}} e^{\frac{|t_l|^2}{2}} \cdot \operatorname{sinc}\big(v(t_l-\eta_l)\big)^2 dt_l \\ &\le 3e^{\frac{|\eta_l|^2}{2}} \int_{\mathbb{R}} \operatorname{sinc}\big(v(t_l-\eta_l)\big)^2 dt_l \\ &= \frac{3e^{\frac{|\eta_l|^2}{2}}}{v}, \end{aligned} \tag{C.13}$$

where the second inequality follows from Claim C.2.1 with $c = 20$ and $\sigma = u$ because by assumption $u \ge 200 \ln n_\lambda$.

Now by incorporating (C.12) and (C.13) into (C.10), we have

$$\begin{aligned}&\int_{\mathbb{R}} e^{|t_l|^2/2} \cdot e^{-u^2|t_l-\eta_l|^2/2} \cdot \operatorname{sinc}\big(v(t_l-\eta_l)\big)^2 dt_l \\ &\qquad \le \frac{3e^{\frac{|\eta_l|^2}{2}}}{v} + \frac{1}{10v} = \frac{(3.1)e^{\frac{|\eta_l|^2}{2}}}{v}.\end{aligned}$$

If we plug the above inequality into (C.9), we get the following:

$$\int_{\mathbb{R}^d} \left|y_{\boldsymbol{\eta},u}(\mathbf{t})\right|^2 d\mu(\mathbf{t}) \le \left(\sqrt{2\pi}\right)^d p(\boldsymbol{\eta}) \cdot v^{2d} \cdot \frac{(3.1)^d e^{\frac{\|\boldsymbol{\eta}\|_2^2}{2}}}{v^d} \le (3.1v)^d. \tag{C.14}$$

□

Proof of Theorem 4.7.1. By the assumptions of the theorem n is an integer, parameter $0 < \lambda \le n/2$, and $R > 0$, and all $\mathbf{x}_1, ..., \mathbf{x}_n \in [-R, R]^d$ and $p(\boldsymbol{\eta}) = \frac{1}{\sqrt{2\pi}} e^{-\frac{\|\boldsymbol{\eta}\|_2^2}{2}}$, therefore Lemmas C.2.1, and C.2.2 go through. Hence the theorem follows immediately by setting $u = 200 \ln n_\lambda$ and then plugging Lemmas C.2.1 and C.2.2 into (C.5). □

C.3 A Lower Bound on the Gaussian Kernel Leverage Scores

With the choice of a Gaussian kernel with $\sigma = (2\pi)^{-1}$ we have $p(\boldsymbol{\eta}) = (2\pi)^{-d/2} \exp(-\|\boldsymbol{\eta}\|_2^2/2)$. Recall from Lemma 4.7.2 that

$$\tau_\lambda(\boldsymbol{\eta}) = \max_{\boldsymbol{\alpha}\in\mathbb{C}^n} \frac{p(\boldsymbol{\eta}) \cdot |\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta})|^2}{\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\|_{L_2(\mu)}^2 + \lambda \|\boldsymbol{\alpha}\|_2^2}. \tag{C.15}$$

In particular, this gives us a method of bounding the leverage function from below, namely, by exhibiting some $\boldsymbol{\alpha}$ and computing the argument of (C.15).

This section is organized as follows. In Section C.3.1, we construct our candidate set of data points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ along with the vector $\boldsymbol{\alpha}$. In particular, $\boldsymbol{\alpha}$ will be chosen to be a vector of

samples of a function $f_{\boldsymbol{\Delta},b,v}$ at each of the data points. Section C.3.2 then describes basic Fourier properties of the function $f_{\boldsymbol{\Delta},b,v}$ and $\boldsymbol{\alpha}$ that we will require later. The next sections then bound each relevant quantity that appears in (C.15) for our specific choice of $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ and $\boldsymbol{\alpha}$. In particular, Section C.3.3 shows a lower bound on $\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta})$, while Section C.3.4 shows an upper bound on $\|\boldsymbol{\alpha}\|_2^2$ and Section C.3.5 shows an upper bound on $\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\|_{L_2(\mu)}^2$.

C.3.1 Construction of data point set and the vector of coefficients $\boldsymbol{\alpha}$

In this section, we construct a set of data points as well as an $\boldsymbol{\alpha}$. As discussed in Section 4.7, we choose the data points to lie on an evenly spaced grid inside $[-R, R]^d$. Moreover, because of the duality of Lemmas 4.7.2 and 4.7.1, we choose $\boldsymbol{\alpha}$ to be related to the test function $y_{\boldsymbol{\eta}}$ in the leverage score upper bound provided in Section C.2. In particular, $\boldsymbol{\alpha}$ is formed by taking samples of a modified version of $\boldsymbol{\Phi} y_{\boldsymbol{\eta}}$ (i.e., a weighted Fourier transform of $y_{\boldsymbol{\eta}}$) on the data points. In particular, the function we sample is $f_{\boldsymbol{\Delta},b,v}$, which we now formally define. We then proceed to proving some useful properties before formally defining $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ and $\boldsymbol{\alpha}$.

Definition C.3.1. For parameters $\boldsymbol{\Delta} \in \mathbb{R}^d$, $b > 0$ and $v > 0$, let the function $f_{\boldsymbol{\Delta},b,v} : \mathbb{R}^d \to \mathbb{R}$ be defined as follows:

$$\begin{aligned} f_{\boldsymbol{\Delta},b,v}(\mathbf{a}) &= 2\cos(2\pi \boldsymbol{\Delta}^\top \mathbf{a}) \left(\frac{1}{\left(\sqrt{2\pi} b\right)^d} e^{-\|\cdot\|_2^2/2b^2} * \mathrm{rect}_v \right)(\boldsymbol{a}) \\ &= 2\cos(2\pi \boldsymbol{\Delta}^\top \mathbf{a}) \int_{a_1 - v/2}^{a_1 + v/2} \int_{a_2 - v/2}^{a_2 + v/2} \cdots \int_{a_d - v/2}^{a_d + v/2} \frac{1}{\left(\sqrt{2\pi} b\right)^d} e^{-\|\mathbf{t}\|_2^2/2b^2} \, dt_d \cdots dt_2 \, dt_1, \end{aligned}$$

where $\mathbf{a} = (a_1, a_2, \ldots, a_d)$ and $\mathbf{t} = (t_1, t_2, \ldots, t_d)$.

Lemma C.3.1. *For any $\boldsymbol{\Delta} \in \mathbb{R}^d$, $v > 0$, and $b > 0$, if we define the function $f_{\boldsymbol{\Delta},b,v}$ as in Definition C.3.1, then*

$$\mathscr{F}\left[f_{\boldsymbol{\Delta},b,v}\right](\boldsymbol{\xi}) = e^{-2\pi^2 b^2 \|\boldsymbol{\xi} - \boldsymbol{\Delta}\|_2^2} \cdot v^d \operatorname{sinc}(v(\boldsymbol{\xi} - \boldsymbol{\Delta})) + e^{-2\pi^2 b^2 \|\boldsymbol{\xi} + \boldsymbol{\Delta}\|_2^2} \cdot v^d \operatorname{sinc}(v(\boldsymbol{\xi} + \boldsymbol{\Delta})).$$

Proof. Note that $\mathscr{F}\left[\frac{1}{(\sqrt{2\pi} b)^d} e^{-\|\cdot\|_2^2/2b^2}\right](\boldsymbol{\xi}) = e^{-2\pi^2 b^2 \|\boldsymbol{\xi}\|_2^2}$. Therefore, by the convolution theorem (see Claim C.1.1),

$$\mathscr{F}\left[\frac{1}{(\sqrt{2\pi} b)^d} e^{-\|\cdot\|_2^2/2b^2} * \mathrm{rect}_v\right](\boldsymbol{\xi}) = e^{-2\pi^2 b^2 \|\boldsymbol{\xi}\|_2^2} \cdot v^d \operatorname{sinc}(v(\boldsymbol{\xi})).$$

Now by the duality of phase shift in time domain and frequency shift in the Fourier domain,

$$\begin{aligned} \mathscr{F}[f_{\boldsymbol{\Delta},b,v}](\boldsymbol{\xi}) &= \mathscr{F}\left[\left(e^{2\pi i \boldsymbol{\Delta}^\top \cdot} + e^{-2\pi i \boldsymbol{\Delta}^\top \cdot}\right)\left(\frac{1}{(\sqrt{2\pi} b)^d} e^{-\|\cdot\|_2^2/2b^2} * \mathrm{rect}_v\right)\right](\boldsymbol{\xi}) \\ &= \mathscr{F}\left[\frac{1}{(\sqrt{2\pi} b)^d} e^{-\|\cdot\|_2^2/2b^2} * \mathrm{rect}_v\right](\boldsymbol{\xi} - \boldsymbol{\Delta}) + \mathscr{F}\left[\frac{1}{(\sqrt{2\pi} b)^d} e^{-\|\cdot\|_2^2/2b^2} * \mathrm{rect}_v\right](\boldsymbol{\xi} + \boldsymbol{\Delta}) \\ &= e^{-2\pi^2 b^2 \|\boldsymbol{\xi} - \boldsymbol{\Delta}\|_2^2} \cdot v^d \operatorname{sinc}(v(\boldsymbol{\xi} - \boldsymbol{\Delta})) + e^{-2\pi^2 b^2 \|\boldsymbol{\xi} + \boldsymbol{\Delta}\|_2^2} \cdot v^d \operatorname{sinc}(v(\boldsymbol{\xi} + \boldsymbol{\Delta})). \end{aligned}$$

□

Definition C.3.2 (Construction of data points and $\boldsymbol{\alpha}$)**.** We let $n = m^d$ for an odd integer $m > 0$. Then, we define a set of n data points $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n \in \mathbb{R}^d$ as follows: We index the points by a d-tuple $\mathbf{j} = (j_1, j_2, \ldots, j_d) \in \{1, 2, \ldots, m\}^d$ for convenience. In particular, we rename $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ as $\mathbf{x}^{\mathbf{j}}$, over $\boldsymbol{j} \in \{1, 2, \ldots, m\}^d$, where $\mathbf{x}^{\boldsymbol{j}}$ is defined as

$$\mathbf{x}^{\boldsymbol{j}} = \left(\left(j_1 - \frac{m+1}{2}\right) \cdot \frac{2R}{m}, \left(j_2 - \frac{m+1}{2}\right) \cdot \frac{2R}{m}, \ldots, \left(j_d - \frac{m+1}{2}\right) \cdot \frac{2R}{m}\right)^\top.$$

Thus, the data points are on a grid of width $\frac{2R}{m}$ extending from $-R$ to R in all d dimensions. For convenience, we let $c_j = \left(j - \frac{m+1}{2}\right) \cdot \frac{2R}{m}$. Therefore, we simply have $\mathbf{x}^{\boldsymbol{j}} = (c_{j_1}, c_{j_2}, \ldots, c_{j_d})$.

Given a point $\boldsymbol{\eta} \in \mathbb{R}^d$ at which we wish to bound the ridge leverage function, we define the vector $\boldsymbol{\alpha} \in \mathbb{C}^n$ to be the vector of evaluations of $f_{\boldsymbol{\eta},b,v}$ at points $\mathbf{x}^{\mathbf{j}}$, for some choice of parameters b and v that we set later. More specifically, we define $\boldsymbol{\alpha} = \{\alpha_{\boldsymbol{j}}\}_{\boldsymbol{j} \in [m]^d}$ by,

$$\begin{aligned}\boldsymbol{\alpha}_{\mathbf{j}} &= f_{\boldsymbol{\eta},b,v}\left(\mathbf{x}^{\boldsymbol{j}}\right) \\ &= 2\cos\left(2\pi\boldsymbol{\eta}^\top \mathbf{x}^{\boldsymbol{j}}\right) \int_{x_1 - \frac{v}{2}}^{x_1 + \frac{v}{2}} \cdots \int_{x_d - \frac{v}{2}}^{x_d + \frac{v}{2}} \frac{1}{(\sqrt{2\pi}b)^d} e^{-\|\mathbf{t}\|_2^2/2b^2}\, d\,t_d \cdots d\,t_1. \end{aligned} \tag{C.16}$$

C.3.2 Basic properties of $f_{\boldsymbol{\Delta},b,v}$ and $\boldsymbol{\alpha}$

By the Nyquist-Shannon sampling theorem, we have the following lemma.

Lemma C.3.2. *For any parameters $\boldsymbol{\Delta} \in \mathbb{R}^d$, $v > 0$, $b > 0$, and any $w > 0$, if $f_{\boldsymbol{\eta},b,v}$ is the function as in Definition C.3.1, then:*

$$\begin{aligned}\mathscr{F}\left[f_{\boldsymbol{\Delta},b,v}(\cdot) \cdot \sum_{\mathbf{j} \in \mathbb{Z}^d} \delta(\cdot - w\mathbf{j})\right](\boldsymbol{\xi}) = w^{-d} v^d \sum_{\mathbf{j} \in \mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi} - \boldsymbol{\Delta} - w^{-1}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}\left(v(\boldsymbol{\xi} - \boldsymbol{\Delta} - \mathbf{j}/w)\right) \\ + w^{-d} v^d \sum_{\mathbf{j} \in \mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi} + \boldsymbol{\Delta} - w^{-1}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}\left(v(\boldsymbol{\xi} + \boldsymbol{\Delta} - \mathbf{j}/w)\right).\end{aligned}$$

Proof. By Claim C.1.5, we have

$$\mathscr{F}\left(f_{\boldsymbol{\Delta},b,v}(\cdot) \sum_{\mathbf{j} \in \mathbb{Z}^d} \delta_d(\cdot - w\mathbf{j})\right)(\boldsymbol{\xi}) = \sum_{\mathbf{j} \in \mathbb{Z}^d} w^{-d} \mathscr{F}\left[f_{\boldsymbol{\Delta},b,v}\right](\boldsymbol{\xi} - \mathbf{j}/w). \tag{C.17}$$

Thus, by Lemma C.3.1, we find that (C.17) can be written as,

$$\sum_{\mathbf{j}\in\mathbb{Z}^d} w^{-d}\mathscr{F}\left[f_{\boldsymbol{\Delta},b,v}\right](\boldsymbol{\xi}-\mathbf{j}/w) = w^{-d}\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2\|\boldsymbol{\xi}-\boldsymbol{\Delta}-w^{-1}\mathbf{j}\|^2}\cdot v^d\operatorname{sinc}\big(v(\boldsymbol{\xi}-\boldsymbol{\Delta}-\mathbf{j}/w)\big) + w^{-d}\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2\|\boldsymbol{\xi}+\boldsymbol{\Delta}-w^{-1}\mathbf{j}\|_2^2}\cdot v^d\operatorname{sinc}\big(v(\boldsymbol{\xi}+\boldsymbol{\Delta}-\mathbf{j}/w)\big),$$

which completes the proof. □

Lemma C.3.3. *For every odd integer $m \geq 3$, positive integer $d \leq 18 n_\lambda \ln^{3/2} n_\lambda$, where $n = m^d$, every $0 < \lambda \leq n/2$, $\boldsymbol{\eta} \in \mathbb{R}^d$, $0 < v \leq R$, and every $0 < b \leq \frac{R}{6\sqrt{\ln n_\lambda}}$, if $f_{\boldsymbol{\eta},b,v}$ is the function as in Definition C.3.1, then for every $\boldsymbol{\xi} \in \mathbb{R}^d$,*

$$\left|\mathscr{F}\left[\sum_{\substack{\mathbf{j}\in\mathbb{Z}\\ \|\mathbf{j}\|_\infty > \frac{m}{2}}} f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right)\cdot\delta\left(\cdot-\frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi})\right| \leq \sqrt{\lambda n}.$$

Proof. By definition of $f_{\boldsymbol{\eta},b,v}$, we have the following for all $\mathbf{a} = (a_1, a_2, \ldots, a_d)$:

$$\left|f_{\boldsymbol{\eta},b,v}(\mathbf{a})\right| \leq \int_{a_1-\frac{v}{2}}^{a_1+\frac{v}{2}}\int_{a_2-\frac{v}{2}}^{a_2+\frac{v}{2}}\cdots\int_{a_d-\frac{v}{2}}^{a_d+\frac{v}{2}} \frac{2}{(\sqrt{2\pi}b)^d} e^{-\|\mathbf{t}\|_2^2/2b^2}\, dt_d\cdots dt_2\, dt_1. \tag{C.18}$$

Note that if $\mathbf{j}\in\mathbb{R}^d$ satisfies $|j_k| > \frac{m}{2}$ for some $k \in \{1,2,\ldots,d\}$, then (C.18) implies,

$$\begin{aligned}
\left|f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right)\right| &\leq 2\prod_{i=1}^{d}\int_{\frac{2R}{m}j_i-\frac{v}{2}}^{\frac{2R}{m}j_i+\frac{v}{2}} \frac{1}{\sqrt{2\pi}b} e^{-t_i^2/2b^2}\, dt_i\\
&\leq \left(\frac{2}{\sqrt{2\pi}b}\int_{\frac{R}{m}|j_k|}^{\infty} e^{-t_k^2/2b^2}\, dt_k\right)\prod_{\substack{1\leq i\leq d\\ i\neq k}}\int_{\frac{2R}{m}j_i-\frac{v}{2}}^{\frac{2R}{m}j_i+\frac{v}{2}} \frac{1}{\sqrt{2\pi}b} e^{-t_i^2/2b^2}\, dt_i\\
&\leq \frac{2}{\sqrt{2\pi}}\cdot\frac{mb}{R|j_k|}\cdot e^{-\frac{R^2|j_k|^2}{2m^2b^2}}\prod_{\substack{1\leq i\leq d\\ i\neq k}}\int_{\frac{2R}{m}j_i-\frac{v}{2}}^{\frac{2R}{m}j_i+\frac{v}{2}} \frac{1}{\sqrt{2\pi}b} e^{-t_i^2/2b^2}\, dt_i\\
&\leq \frac{2b}{R}\cdot e^{-\frac{R^2|j_k|^2}{2m^2b^2}}\prod_{\substack{1\leq i\leq d\\ i\neq k}}\int_{\frac{2R}{m}j_i-\frac{v}{2}}^{\frac{2R}{m}j_i+\frac{v}{2}} \frac{1}{\sqrt{2\pi}b} e^{-t_i^2/2b^2}\, dt_i,
\end{aligned}$$

where we have used the fact that $\frac{2R}{m}|j_k| - |\frac{v}{2}| \geq \frac{2R}{m}|j_k| - \frac{R}{2} \geq \frac{R}{m}|j_k|$, along with Claim C.1.6.

Therefore,

$$\left|\mathscr{F}\left[\sum_{\substack{\mathbf{j}\in\mathbb{Z}\\ \|\mathbf{j}\|_\infty>\frac{m}{2}}} f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right)\cdot\delta\left(\cdot-\frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi})\right| \le \sum_{\substack{\mathbf{j}\in\mathbb{Z}\\ \|\mathbf{j}\|_\infty>\frac{m}{2}}}\left|f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right)\right|$$
$$\le \sum_{k=1}^{d}\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d\\ |j_k|>\frac{m}{2}}}\left|f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right)\right|$$
$$\le \sum_{k=1}^{d}\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d\\ |j_k|>\frac{m}{2}}}\frac{2b}{R}e^{-\frac{R^2|j_k|^2}{2m^2b^2}}\prod_{\substack{1\le i\le d\\ i\neq k}}\int_{\frac{2R}{m}j_i-\frac{v}{2}}^{\frac{2R}{m}j_i+\frac{v}{2}}\frac{e^{-t_i^2/2b^2}}{\sqrt{2\pi}b}\,dt_i$$

We bound:

$$\sum_{k=1}^{d}\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d\\ |j_k|>\frac{m}{2}}}\frac{2b}{R}e^{-\frac{R^2|j_k|^2}{2m^2b^2}}\prod_{\substack{1\le i\le d\\ i\neq k}}\int_{\frac{2R}{m}j_i-\frac{v}{2}}^{\frac{2R}{m}j_i+\frac{v}{2}}\frac{1}{\sqrt{2\pi}b}e^{-t^2/2b^2}\,dt$$
$$\le \frac{2b}{R}\sum_{k=1}^{d}\sum_{|j_k|>\frac{m}{2}}e^{-\frac{R^2|j_k|^2}{2m^2b^2}}\cdot\prod_{\substack{1\le i\le d\\ i\neq k}}\left(\sum_{j_i=-\infty}^{\infty}\int_{\frac{2R}{m}j_i-\frac{v}{2}}^{\frac{2R}{m}j_i+\frac{v}{2}}\frac{1}{\sqrt{2\pi}b}e^{-t_i^2/2b^2}\,dt_i\right)$$
$$\le \frac{2b}{R}\sum_{k=1}^{d}\sum_{|j_k|>\frac{m}{2}}e^{-\frac{R^2|j_k|^2}{2m^2b^2}}\cdot\prod_{\substack{1\le i\le d\\ i\neq k}}\left(\left\lceil\frac{vm}{2R}\right\rceil\int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}b}e^{-t^2/2b^2}\,dt\right)$$

where the last inequality is due to the fact that each point in $\mathbb{R}$ gets integrated at most $\lceil\frac{vm}{2R}\rceil$ times in the infinite sum. Again using Claim C.1.6:

$$\left|\mathscr{F}\left[\sum_{\substack{\mathbf{j}\in\mathbb{Z}\\ \|\mathbf{j}\|_\infty>\frac{m}{2}}} f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right)\cdot\delta\left(\cdot-\frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi})\right| \le \frac{2b}{R}\left(\frac{m+1}{2}\right)^{d-1}\sum_{k=1}^{d}\left(\sum_{|j_k|>\frac{m}{2}}e^{-\frac{R^2|j_k|^2}{2m^2b^2}}\right)$$
$$\le \frac{4b}{R}\left(\frac{m+1}{2}\right)^{d-1}\sum_{k=1}^{d}\int_{\frac{m-1}{2}}^{\infty}e^{-\frac{R^2t^2}{2m^2b^2}}\,dt$$
$$= \frac{4bd}{R}\cdot\left(\frac{m+1}{2}\right)^{d-1}\int_{\frac{m-1}{2}}^{\infty}e^{-\frac{R^2t^2}{2m^2b^2}}\,dt$$
$$\le \frac{4bd}{R}\cdot\left(\frac{m+1}{2}\right)^{d-1}\cdot\frac{m^2b^2/R^2}{\left(\frac{m-1}{2}\right)}e^{-\frac{R^2\left(\frac{m-1}{2}\right)^2}{2m^2b^2}}$$
$$\le 12dn\left(\frac{b}{R}\right)^3 e^{-\frac{R^2}{18b^2}} \le \lambda \le \sqrt{\lambda n},$$

since $m\ge 3$, $n=m^d$, $R\ge 6b\sqrt{\ln n_\lambda}$, $d\le 18n_\lambda\ln^{3/2}n_\lambda$, and $\lambda\le n/2$. □

Lemma C.3.4. *Consider the preconditions of Lemma C.3.3. If* $\boldsymbol{\alpha}$ *is defined as in* (C.16) *of*

Definition C.3.2, then:

$$\left|\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\xi}) - \left(\frac{mv}{2R}\right)^d \sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right) + e^{-2\pi^2 b^2 \|\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right| \le \sqrt{\lambda n}.$$

Proof. By definition of $\boldsymbol{\alpha}$ and $\mathbf{z}$,

$$\begin{aligned}
\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\xi}) &= \sum_{1\le j_1, j_2, \ldots, j_d \le m} \alpha_{\mathbf{j}} e^{-2\pi i \boldsymbol{\xi}^\top \mathbf{x}^{\mathbf{j}}} = \sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \|\mathbf{j}\|_\infty \le \frac{m}{2}}} f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right) \cdot e^{-2\pi i \left(\frac{2R}{m}\right)\mathbf{j}^\top \boldsymbol{\xi}} \\
&= \mathscr{F}\left[\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \|\mathbf{j}\|_\infty \le \frac{m}{2}}} f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right) \cdot \delta\left(\cdot - \frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi}) \\
&= \mathscr{F}\left[\sum_{\mathbf{j}\in\mathbb{Z}^d} f_{\boldsymbol{\eta},b,v}(\cdot) \cdot \delta\left(\cdot - \frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi}) - \mathscr{F}\left[\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \|\mathbf{j}\|_\infty > \frac{m}{2}}} f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right) \cdot \delta\left(\cdot - \frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi}). \qquad \text{(C.19)}
\end{aligned}$$

By Lemma C.3.2 (applied with $w = 2R/m$), the first term in (C.19) can be written as:

$$\begin{aligned}
&\mathscr{F}\left[\sum_{\mathbf{j}\in\mathbb{Z}^d} f_{\boldsymbol{\eta},b,v}(\cdot) \cdot \delta\left(\cdot - \frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi}) \\
&= \left(\frac{mv}{2R}\right)^d \sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right) + e^{-2\pi^2 b^2 \|\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right). \qquad \text{(C.20)}
\end{aligned}$$

Now, by assumption, that preconditions of Lemma C.3.3 hold, therefore, the second term in (C.19) can be bounded, by invoking Lemma C.3.3, as,

$$\left|\mathscr{F}\left[\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \|\mathbf{j}\|_\infty > \frac{m}{2}}} f_{\boldsymbol{\eta},b,v}\left(\frac{2R}{m}\mathbf{j}\right) \cdot \delta\left(\cdot - \frac{2R}{m}\mathbf{j}\right)\right](\boldsymbol{\xi})\right| \le \sqrt{\lambda n}. \qquad \text{(C.21)}$$

Thus, the desired result follows by combining (C.19), (C.20), and (C.21). □

C.3.3 Bounding $\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta})$

Lemma C.3.5. *For every odd integer $m \ge 8 \ln n_\lambda$, positive integer $d \le 8 n_\lambda$, where $n = m^d$, every parameter $0 < \lambda \le \left(\frac{v}{2R}\right)^{2d} \cdot \frac{n}{64}$, every $\boldsymbol{\eta} \in \mathbb{R}^d$ satisfying $\|\boldsymbol{\eta}\|_\infty \le \frac{n^{1/d}}{4R}$, if $\boldsymbol{\alpha}$ is defined as in* (C.16) *of Definition C.3.2 with parameters $0 < v \le R$ and $b = \frac{R}{6\sqrt{\ln n_\lambda}}$, then:*

$$\left|\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta})\right| \ge \frac{n}{4}\left(\frac{v}{2R}\right)^d.$$

Proof. Since $\lambda \le n/256 \le n/2$, $m \ge 8\ln n_\lambda \ge 3$, $d \le 8n_\lambda$, and $b = \frac{R}{6\sqrt{\ln n_\lambda}}$, Lemma C.3.4 implies,

$$\left|\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta}) - \left(\frac{mv}{2R}\right)^d \sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc} v\left(\frac{m}{2R}\mathbf{j}\right) + e^{-2\pi^2 b^2 \|2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc} v\left(2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\right)\right| \le \sqrt{\lambda n}.$$

Therefore, by the fact that $|\operatorname{sinc}(\cdot)| \le 1$ and $\operatorname{sinc}(\cdot) \ge -\frac{1}{4}$, we have

$$\begin{aligned}
\left|\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta})\right| &\ge \left(\frac{mv}{2R}\right)^d \left|\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc} v\left(\frac{m}{2R}\mathbf{j}\right) + e^{-2\pi^2 b^2 \|2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc} v\left(2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\right)\right| - \sqrt{\lambda n} \\
&\ge \left(\frac{mv}{2R}\right)^d \left(1 + e^{-2\pi^2 b^2 \|2\boldsymbol{\eta}\|_2^2} \cdot \operatorname{sinc} v(2\boldsymbol{\eta}) - \sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \mathbf{j}\neq\mathbf{0}}} e^{-2\pi^2 b^2 \|\frac{m}{2R}\mathbf{j}\|_2^2} + e^{-2\pi^2 b^2 \|2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\|_2^2}\right) - \sqrt{\lambda n} \\
&\ge \frac{3}{4}\left(\frac{mv}{2R}\right)^d - \left(\frac{mv}{2R}\right)^d \sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \mathbf{j}\neq\mathbf{0}}} e^{-2\pi^2 b^2 \|\frac{m}{2R}\mathbf{j}\|_2^2} + e^{-2\pi^2 b^2 \|2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\|_2^2} - \sqrt{\lambda n}. \qquad \text{(C.22)}
\end{aligned}$$

Now we show that $\sum_{\mathbf{j}\in\mathbb{Z}^d, \mathbf{j}\neq\mathbf{0}} \left(e^{-2\pi^2 b^2 \|\frac{m}{2R}\mathbf{j}\|_2^2} + e^{-2\pi^2 b^2 \|2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\|_2^2}\right)$ is small. Note that the assumption $b = \frac{R}{6\sqrt{\ln n_\lambda}}$, implies $e^{-2\pi^2 b^2 \|-\frac{m}{2R}\mathbf{j}\|_2^2} \le e^{-\frac{1}{8}\cdot\frac{m^2}{\log n_\lambda}\|\mathbf{j}\|_2^2} \le e^{-m\|\mathbf{j}\|_1}$, since $m \ge 8\ln n_\lambda$. Thus,

$$\begin{aligned}
\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \mathbf{j}\neq\mathbf{0}}} e^{2\pi^2 b^2 \|\frac{m}{2R}\mathbf{j}\|_2^2} &\le \sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \mathbf{j}\neq\mathbf{0}}} e^{-m\|\mathbf{j}\|_1} \\
&= \left(\sum_{j_1=-\infty}^{\infty} e^{-m|j_1|}\right)\left(\sum_{j_2=-\infty}^{\infty} e^{-m|j_2|}\right)\cdots\left(\sum_{j_d=-\infty}^{\infty} e^{-m|j_d|}\right) - 1 \\
&= \left(1 + \frac{2e^{-m}}{1-e^{-m}}\right)^d - 1 \\
&\le e^{3de^{-m}} - 1 \le 1/16, \qquad \text{(C.23)}
\end{aligned}$$

where the last inequality follows because by the assumption $m \ge 8\ln n_\lambda$, we have $3de^{-m} \le \frac{3d}{n_\lambda^8} \le \frac{1}{64}$, since $d \le 8n_\lambda$ and $n_\lambda \ge 256$. Moreover, recall that $\left\|\boldsymbol{\eta}\right\|_\infty \le \frac{m}{4R}$, and so, $\left\|2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\right\|_2^2 \ge \|\frac{m}{4R}\mathbf{j}\|_2^2$. Thus, in a similar fashion,

$$\begin{aligned}
\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \mathbf{j}\neq\mathbf{0}}} e^{-2\pi^2 b^2 \|2\boldsymbol{\eta} - \frac{m}{2R}\mathbf{j}\|_2^2} &\le \sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \mathbf{j}\neq\mathbf{0}}} e^{-2\pi^2 b^2 \|\frac{m}{4R}\mathbf{j}\|_2^2} \\
&\le \sum_{\substack{\mathbf{j}\in\mathbb{Z}^d \\ \mathbf{j}\neq\mathbf{0}}} e^{-m\|\mathbf{j}\|_1/4} \\
&\le \left(1 + \frac{2e^{-m/4}}{1-e^{-m/4}}\right)^d - 1 \\
&\le e^{3de^{-m/4}} - 1 \le 1/4. \qquad \text{(C.24)}
\end{aligned}$$

Thus, combining (C.22), (C.23), and (C.24), we have

$$\begin{aligned}\left|\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta})\right| &\geq \left(\frac{mv}{2R}\right)^d \left(\frac{3}{4} - \frac{1}{16} - \frac{1}{4}\right) - \sqrt{\lambda n} \\ &\geq \frac{n}{4}\left(\frac{v}{2R}\right)^d,\end{aligned}$$

where the final inequality follows since $\sqrt{\lambda n} \leq \sqrt{\frac{n}{64}\left(\frac{v}{2R}\right)^{2d} \cdot n} = \frac{n}{8}\left(\frac{v}{2R}\right)^d$.

□

C.3.4 Bounding $\|\boldsymbol{\alpha}\|_2^2$

Lemma C.3.6. *For every odd integer $m \geq 3$, $n = m^d$, every $\boldsymbol{\eta} \in \mathbb{R}^d$, and every $b, v > 0$, if $\boldsymbol{\alpha}$ is defined as in* (C.16) *of Definition C.3.2, then we have*

$$\|\boldsymbol{\alpha}\|_2^2 \leq 4n.$$

Proof. Let $w = 2R/m$. Then, letting $\mathbf{j} = (j_1, j_2, \ldots, j_d)$, we observe that

$$\begin{aligned}\|\boldsymbol{\alpha}\|_2^2 &= \sum_{\mathbf{j} \in \{1,2,\ldots,m\}^d} \boldsymbol{\alpha}_{\mathbf{j}}^2 \\ &= \sum_{\substack{\mathbf{j} \in \mathbb{Z}^d \\ \|\mathbf{j}\|_\infty \leq \frac{m-1}{2}}} \left(2\cos(2\pi w \boldsymbol{\eta}^\top \mathbf{j})\left(\frac{1}{\left(\sqrt{2\pi}b\right)^d} e^{-\|\cdot\|_2^2/2b^2} * \mathrm{rect}_v\right)(w\boldsymbol{j})\right)^2 \\ &\leq \sum_{\substack{\mathbf{j} \in \mathbb{Z}^d \\ \|\mathbf{j}\|_\infty \leq \frac{m-1}{2}}} 4 \\ &= 4m^d = 4n,\end{aligned}$$

as desired. □

C.3.5 Bounding $\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\|_{L_2(d\mu)}^2$

Note that all the results so far hold for any kernel $p(\boldsymbol{\eta})$ and are independent of the kernel function. Now, we upper bound $\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\|_{L_2(d\mu)}$. This quantity depends on the particular choice of kernel, which we assume to be Gaussian.

Lemma C.3.7. *For every odd integer $m \geq 8\ln n_\lambda$, positive integer $d \leq 8n_\lambda$, where $n = m^d$, every parameter $0 < \lambda \leq \left(\frac{1}{2}\right)^{2d} \cdot \frac{n}{64}$, every $\boldsymbol{\eta}$ satisfying $\|\boldsymbol{\eta}\|_\infty \leq 10\sqrt{\ln n_\lambda}$, and any $60\ln^{3/2} n_\lambda \leq R \leq \frac{m}{80\sqrt{\ln n_\lambda}}$, if $\boldsymbol{\alpha}$ is defined as in* (C.16) *of Definition C.3.2 with parameters $b = \frac{R}{6\sqrt{\ln n_\lambda}}$ and $v = R$,*

then for the Gaussian kernel with pdf $p(\boldsymbol{\xi}) = \frac{1}{(\sqrt{2\pi})^d} e^{-\|\boldsymbol{\xi}\|_2^2/2}$*, we have:*

$$\left\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\right\|_{L_2(\mu)}^2 \leq 16n^2 \left(\frac{3}{4R}\right)^d \cdot p(\boldsymbol{\eta}) + 4\lambda n. \tag{C.25}$$

Proof. Recall that we set $v = R$ and $b = \frac{R}{6\sqrt{\ln n_\lambda}}$. Thus, since $\lambda \leq \left(\frac{1}{2}\right)^{2d} \cdot \frac{n}{64}$, and $d \leq 8n_\lambda$, Lemma C.3.4 implies that

$$\begin{aligned}
|\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\xi})|^2 &\leq \left(\left|\left(\frac{m}{2}\right)^d \sum_{\mathbf{j}\in\mathbb{Z}^d} \left(e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right.\right.\right. \\
&\qquad \left.\left.\left. + e^{-2\pi^2 b^2 \|\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right| + \sqrt{\lambda n}\right)^2 \\
&\leq 2\left(\frac{m}{2}\right)^{2d} \left(\sum_{\mathbf{j}\in\mathbb{Z}^d} \left(e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right.\right. \\
&\qquad \left.\left. + e^{-2\pi^2 b^2 \|\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 + 2(\sqrt{\lambda n})^2.
\end{aligned}$$

By the definition of the $L_2(\mu)$ norm, $\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\|_{L_2(\mu)}^2 = \int_{\mathbb{R}^d} |\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\xi})|^2 p(\boldsymbol{\xi})\, d\boldsymbol{\xi}$, we have

$$\begin{aligned}
\left\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\right\|_{L_2(\mu)}^2 &\leq \int_{\mathbb{R}^d} 2\left(\frac{m}{2}\right)^{2d} \left(\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right. \\
&\qquad \left. + e^{-2\pi^2 b^2 \|\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}+\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 p(\boldsymbol{\xi})\, d\boldsymbol{\xi} + \int_{\mathbb{R}^d} 2\lambda n p(\boldsymbol{\xi})\, d\boldsymbol{\xi} \\
&= 8\left(\frac{m}{2}\right)^{2d} \int_{\mathbb{R}^d} \left(\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 p(\boldsymbol{\xi})\, d\boldsymbol{\xi} + 2\lambda n,
\end{aligned} \tag{C.26}$$

where the last equality holds because the kernel pdf $p(\boldsymbol{\xi})$ is symmetric in our case, and the sum is over all $\mathbf{j} \in \mathbb{Z}^d$. The integral in (C.26) can be split into two integrals as follows:

$$\begin{aligned}
&\int_{\mathbb{R}^d} \left(\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 p(\boldsymbol{\xi})\, d\boldsymbol{\xi} \\
&\quad = \int_{\|\boldsymbol{\xi}\|_\infty \leq 10\sqrt{\ln n_\lambda}} \left(\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 p(\boldsymbol{\xi})\, d\boldsymbol{\xi} \\
&\quad + \int_{\|\boldsymbol{\xi}\|_\infty \geq 10\sqrt{\ln n_\lambda}} \left(\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 p(\boldsymbol{\xi})\, d\boldsymbol{\xi}.
\end{aligned} \tag{C.27}$$

First, we consider the integral over $\|\boldsymbol{\xi}\|_\infty \leq 10\sqrt{\ln n_\lambda}$. By the assumption of lemma, $\|\boldsymbol{\eta}\|_\infty \leq 10\sqrt{\ln n_\lambda}$, and hence, $\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_\infty \leq 20\sqrt{\ln n_\lambda}$. This implies that $\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_\infty \leq \frac{1}{2}(\frac{m}{2R})$, since we assume that $R \leq \frac{m}{80\sqrt{\ln n_\lambda}}$. Therefore, for any $\mathbf{j} \neq (0,0,\ldots,0)$, there exists some k such that $j_k \neq 0$,

and so,

$$\begin{aligned}
\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d\\ \mathbf{j}\neq\mathbf{0}}} e^{-2\pi^2 b^2\left\|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right\|_2^2} &\le \sum_{k=1}^{d}\sum_{\substack{\mathbf{j}\in\mathbb{Z}^d\\ j_k\neq 0}} e^{-2\pi^2 b^2\left\|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right\|_2^2}\\
&= \sum_{k=1}^{d}\left[\left(\sum_{|j_k|\ge 1} e^{-2\pi^2 b^2\left(\xi_k-\eta_k-\frac{m}{2R}j_k\right)^2}\right)\prod_{\substack{1\le i\le d\\ i\neq k}}\sum_{j_i=-\infty}^{\infty} e^{-2\pi^2 b^2\left(\xi_i-\eta_i-\frac{m}{2R}j_i\right)^2}\right]\\
&\le \sum_{k=1}^{d}\left[\left(\sum_{|j_k|\ge 1} e^{-\frac{\pi^2 b^2 m^2}{8R^2}(2|j_k|-1)^2}\right)\prod_{\substack{1\le i\le d\\ i\neq k}}\left(1+\sum_{|j_i|\ge 1} e^{-\frac{\pi^2 b^2 m^2}{8R^2}(2|j_i|-1)^2}\right)\right]\\
&\le \sum_{k=1}^{d}\left[\left(2\sum_{j_k=1}^{\infty} e^{-mj_k/4}\right)\prod_{\substack{1\le i\le d\\ i\neq k}}\left(1+2\sum_{j_i=1}^{\infty} e^{-mj_i/4}\right)\right]\\
&\le d\left(3e^{-m/4}\right)\left(1+3e^{-m/4}\right)^{d-1}\\
&\le \frac{4d}{n_\lambda^2},
\end{aligned} \tag{C.28}$$

where we have used the assumptions $b=\frac{R}{6\sqrt{\ln n_\lambda}}$ and $m\ge 8\ln n_\lambda$, as well as $n_\lambda\ge 256$ and $d\le 8n_\lambda$.

Now, using (C.28), we see that the first integral in (C.27) can be bounded as follows:

$$\begin{aligned}
&\int_{\|\boldsymbol{\xi}\|_\infty\le 10\sqrt{\ln n_\lambda}} p(\boldsymbol{\xi})\left(\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2\left\|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right\|_2^2}\cdot \operatorname{sinc} v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 d\boldsymbol{\xi}\\
&\quad\le 2\int_{\|\boldsymbol{\xi}\|_\infty\le 10\sqrt{\ln n_\lambda}} p(\boldsymbol{\xi})\left(e^{-2\pi^2 b^2\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_2^2}\cdot \operatorname{sinc} v(\boldsymbol{\xi}-\boldsymbol{\eta})^2\right)^2 d\boldsymbol{\xi}\\
&\qquad + 2\int_{\|\boldsymbol{\xi}\|_\infty\le 10\sqrt{\ln n_\lambda}} p(\boldsymbol{\xi})\left(\sum_{\mathbf{j}\neq\mathbf{0}} e^{-2\pi^2 b^2\left\|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right\|_2^2}\cdot \operatorname{sinc} v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 d\boldsymbol{\xi}\\
&\quad\le 2\int_{\|\boldsymbol{\xi}\|_\infty\le 10\sqrt{\ln n_\lambda}} \frac{1}{(\sqrt{2\pi})^d} e^{-\|\boldsymbol{\xi}\|_2^2/2}\left(e^{-2\pi^2 b^2\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_2^2}\operatorname{sinc}\left(v(\boldsymbol{\xi}-\boldsymbol{\eta})\right)^2+\frac{4d}{n_\lambda^2}\right)d\boldsymbol{\xi}\\
&\quad= 2\int_{\|\boldsymbol{\xi}\|_\infty\le 10\sqrt{\ln n_\lambda}} \frac{1}{(\sqrt{2\pi})^d} e^{-\|\boldsymbol{\xi}\|_2^2/2} e^{-2\pi^2 b^2\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_2^2}\cdot\operatorname{sinc}\left(v(\boldsymbol{\xi}-\boldsymbol{\eta})\right)^2 d\boldsymbol{\xi}+\frac{8d}{n_\lambda^2}.
\end{aligned} \tag{C.29}$$

Next, by applying Claim C.2.1 (with $c=1$ and $\sigma=b$), we have $e^{-\|\boldsymbol{\xi}\|_2^2/2}\le 3^d e^{-\|\boldsymbol{\eta}\|_2^2/2}$ for $\left\|\boldsymbol{\xi}-\boldsymbol{\eta}\right\|_\infty\le\frac{\sqrt{\ln n_\lambda}}{b}$ (since $\|\boldsymbol{\xi}\|_\infty\le 10\sqrt{\ln n_\lambda}$ and $b=\frac{R}{6\sqrt{\ln n_\lambda}}\ge 10\ln n_\lambda$). Hence,

$$\int_{\substack{\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_\infty \le \frac{\sqrt{\ln n_\lambda}}{b} \\ \|\boldsymbol{\xi}\|_\infty \le 10\sqrt{\ln n_\lambda}}} \frac{1}{(\sqrt{2\pi})^d} e^{-\|\boldsymbol{\xi}\|_2^2/2} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}\|_2^2} \cdot \operatorname{sinc}\left(v(\boldsymbol{\xi}-\boldsymbol{\eta})\right)^2 d\boldsymbol{\xi}$$

$$\le 3^d e^{-\|\boldsymbol{\eta}\|_2^2/2} \int_{\mathbb{R}^d} \frac{1}{(\sqrt{2\pi})^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}\|_2^2} \cdot \operatorname{sinc}\left(v(\boldsymbol{\xi}-\boldsymbol{\eta})\right)^2 d\boldsymbol{\xi}$$

$$\le \frac{3^d}{(\sqrt{2\pi})^d} e^{-\|\boldsymbol{\eta}\|_2^2/2} \int_{\mathbb{R}^d} \operatorname{sinc}\left(v(\boldsymbol{\xi}-\boldsymbol{\eta})\right)^2 d\boldsymbol{\xi}$$

$$= \left(\frac{3}{v}\right)^d p(\boldsymbol{\eta}) \tag{C.30}$$

Note that the last line follows from the fact that $v^d \cdot \operatorname{sinc} v(\cdot)$ is the Fourier transform of rect_v, and so, by the convolution theorem (Claim C.1.1), $\int_{\mathbb{R}^d} (v^d \cdot \operatorname{sinc} v\mathbf{t})^2 \, d\mathbf{t} = (\operatorname{rect}_v * \operatorname{rect}_v)(0) = v^d$.

Additionally,

$$\int_{\substack{\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_\infty \ge \frac{\sqrt{\ln n_\lambda}}{b} \\ \|\boldsymbol{\xi}\|_\infty \le 10\sqrt{\ln n_\lambda}}} \frac{1}{(\sqrt{2\pi})^d} e^{-\|\boldsymbol{\xi}\|_2^2/2} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}\|_2^2} \cdot \operatorname{sinc}\left(v(\boldsymbol{\xi}-\boldsymbol{\eta})\right)^2 d\boldsymbol{\xi} \le n_\lambda^{-10} \int_{\mathbb{R}^d} \frac{1}{(\sqrt{2\pi})^d} e^{-\|\boldsymbol{\xi}\|_2^2/2} d\boldsymbol{\xi}$$

$$= n_\lambda^{-10}, \tag{C.31}$$

since $\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_2 \ge \|\boldsymbol{\xi}-\boldsymbol{\eta}\|_\infty$. Thus, (C.29), (C.30), and (C.31) imply that

$$\int_{\|\boldsymbol{\xi}\|_\infty \le 10\sqrt{\ln n_\lambda}} \left(\sum_{\mathbf{j}\in\mathbb{R}^d} e^{-2\pi^2 b^2 \|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \cdot \operatorname{sinc} v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right) \right)^2 p(\boldsymbol{\xi}) \, d\boldsymbol{\xi}$$

$$\le 2\left(\frac{3}{v}\right)^d p(\boldsymbol{\eta}) + 2n_\lambda^{-10} + \frac{8d}{n_\lambda^2}. \tag{C.32}$$

Next, we bound the second integral in (C.27). We first show that the quantity in parentheses is upper bounded by a constant for all $\boldsymbol{\xi}$ in the appropriate range, and then use this bound to upper bound the integral itself. Consider $\boldsymbol{\xi} \in \mathbb{R}^d$ satisfying $\|\boldsymbol{\xi}\|_\infty \ge 10\sqrt{\ln n_\lambda}$. Let t_i, for every $i = 1, \dots d$, be an integer such that $|\xi_i - \eta_i - t_i m/2R| \le m/4R$. Note that the following upper

bound holds:

$$\begin{aligned}
\left|\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2\|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2}\cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right| &\le \sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2\|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2} \\
&\le \prod_{i=1}^d \sum_{j_i=-\infty}^{\infty} e^{-2\pi^2 b^2\left(\xi_i-\eta_i-\frac{m}{2R}j_i\right)^2} \\
&\le \prod_{i=1}^d \left(1+\sum_{j_i\neq t_i} e^{-2\pi^2 b^2\left(\xi_i-\eta_i-\frac{m}{2R}j_i\right)^2}\right) \\
&\le \prod_{i=1}^d \left(1+2\sum_{j_i=1}^{\infty} e^{-\frac{\pi^2 b^2 m^2}{8R^2}(2|j_i|-1)^2}\right) \\
&\le \prod_{i=1}^d \left(1+\sum_{|j_i|\ge 1} e^{-mj_i/4}\right) \\
&\le \prod_{i=1}^d \left(1+3e^{-m/4}\right) \\
&\le e^{\frac{3d}{n_\lambda^2}} \le 3,
\end{aligned}$$

since $d \le 8n_\lambda$, $m \ge 8\ln n_\lambda$ and $n_\lambda \ge 256$. Thus, we can bound the second integral in (C.27) as follows:

$$\begin{aligned}
&\int_{\|\boldsymbol{\xi}\|_\infty \ge 10\sqrt{\ln n_\lambda}} \left(\sum_{\mathbf{j}\in\mathbb{Z}^d} e^{-2\pi^2 b^2\|\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\|_2^2}\cdot \operatorname{sinc}^v\left(\boldsymbol{\xi}-\boldsymbol{\eta}-\frac{m}{2R}\mathbf{j}\right)\right)^2 p(\boldsymbol{\xi})\,d\boldsymbol{\xi} \\
&\qquad \le 9\int_{\|\boldsymbol{\xi}\|_\infty \ge 10\sqrt{\ln n_\lambda}} p(\boldsymbol{\xi})\,d\boldsymbol{\xi} \\
&\qquad \le 9\sum_{k=1}^d \left(2\int_{10\sqrt{\ln n_\lambda}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\xi_k^2/2}\,d\xi_k\right)\prod_{i\neq k}\int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}} e^{-\xi_i^2/2}\,d\xi_i \\
&\qquad \le \frac{18d}{\sqrt{2\pi}}\cdot\frac{n_\lambda^{-50}}{10\sqrt{\log n_\lambda}} \le n_\lambda^{-40},
\end{aligned} \tag{C.33}$$

by Claim C.1.6.

Combining (C.26), (C.27), (C.32), and (C.33) implies that

$$\begin{aligned}
\left\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\right\|_{L_2(\mu)}^2 &\le 8\left(\frac{m}{2}\right)^{2d}\left(2\left(\frac{3}{v}\right)^d p(\boldsymbol{\eta}) + 2n_\lambda^{-10} + \frac{8d}{n_\lambda^2} + n_\lambda^{-40}\right) + 2\lambda n \\
&\le 8\left(\frac{m}{2}\right)^{2d}\left(2\left(\frac{3}{v}\right)^d p(\boldsymbol{\eta}) + n_\lambda^{-1}\right) + 2\lambda n \\
&\le 16n^2\left(\frac{3}{4R}\right)^d \cdot p(\boldsymbol{\eta}) + 4\lambda n,
\end{aligned}$$

as desired. In the above, the second inequality follows from $2n_\lambda^{-10} + \frac{8d}{n_\lambda^2} + n_\lambda^{-40} \le n_\lambda^{-1}$, because

$n_\lambda \geq 256$, and $d \leq 8n_\lambda$. □

Proof of Theorem 4.7.2. Note that we can choose data points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ and the vector $\boldsymbol{\alpha}$ according to the construction in Definition C.3.2 with $v = R$ and $b = \frac{R}{6\sqrt{\log n_\lambda}}$. Thus, Lemmas C.3.5, C.3.6, and C.3.7, as well as (C.15), imply that

$$\begin{aligned}
\tau_\lambda(\boldsymbol{\eta}) &\geq \frac{p(\boldsymbol{\eta}) \cdot |\boldsymbol{\alpha}^* \mathbf{z}(\boldsymbol{\eta})|^2}{\|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\|^2_{L_2(d\mu)} + \lambda \|\boldsymbol{\alpha}\|_2^2} \\
&\geq \frac{p(\boldsymbol{\eta}) \cdot \left(\frac{n}{4}\left(\frac{1}{2}\right)^d\right)^2}{16n^2\left(\frac{3}{4R}\right)^d p(\boldsymbol{\eta}) + 4\lambda n + \lambda(4n)} \\
&\geq \frac{1}{128}\left(\frac{R}{3}\right)^d \cdot \frac{p(\boldsymbol{\eta})}{2p(\boldsymbol{\eta}) + (4R/3)^d n_\lambda^{-1}},
\end{aligned}$$

as desired. □

C.4 Proof of Corollary 4.7.1

In the proof of corollary 4.7.1 we often need to compute the volume of a d-dimensional ball hence we state it as a claim.

Claim C.4.1. *For any integer $d \geq 1$ and any $R > 0$ the following holds:*

$$\int_{\substack{\boldsymbol{\eta} \in \mathbb{R}^d \\ \|\boldsymbol{\eta}\|_2 \leq R}} 1 d\boldsymbol{\eta} = \frac{(\sqrt{\pi}R)^d}{\Gamma(d/2+1)}$$

where Γ is the Gamma function.

First claim of the corollary (upper bound on statistical dimension): Let $t = 10\sqrt{\ln n_\lambda}$. We have:

$$s_\lambda = \int_{\mathbb{R}^d} \tau(\boldsymbol{\eta}) d\boldsymbol{\eta} = \int_{\substack{\boldsymbol{\eta} \in \mathbb{R}^d \\ \|\boldsymbol{\eta}\|_2 \leq t}} \tau(\boldsymbol{\eta}) d\boldsymbol{\eta} + \int_{\substack{\boldsymbol{\eta} \in \mathbb{R}^d \\ \|\boldsymbol{\eta}\|_2 > t}} \tau(\boldsymbol{\eta}) d\boldsymbol{\eta}$$

By the naive bound in Proposition 1 we have:

$$\begin{aligned}\int_{\substack{\boldsymbol{\eta}\in\mathbb{R}^d\\ \|\boldsymbol{\eta}\|_2>t}} \tau(\boldsymbol{\eta})d\boldsymbol{\eta} \le n_\lambda \int_{\substack{\boldsymbol{\eta}\in\mathbb{R}^d\\ \|\boldsymbol{\eta}\|_2>t}} e^{-\frac{\|\boldsymbol{\eta}\|_2^2}{2}} d\boldsymbol{\eta} &= n_\lambda \Big(\prod_{i=1}^{d-1}\int_{\theta_i\in[0,2\pi]} d\theta_i\Big)\int_{[t,\infty]} r^{d-1}e^{-r^2/2}dr \\ &= (\sqrt{2\pi})^{d-1} n_\lambda \int_{[t,\infty]} r^{d-1}e^{-r^2/2}dr \\ &\le (\sqrt{2\pi})^{d-1} n_\lambda \int_{[t,\infty]} e^{-r^2/4}dr \\ &\le (\sqrt{2\pi})^{d-1} n_\lambda \cdot \left(\frac{2e^{-t^2/4}}{t}\right) \le (\sqrt{2\pi})^d n_\lambda^{-25} \qquad \text{(C.34)}\end{aligned}$$

where the first equality follows by converting from cartesian coordinates to polar coordinates. The second inequality uses the fact that if $d \le \frac{t^2}{4\log t}$ then for all r with $r \ge t$ we have $r^{d-1}e^{-r^2/2} \le e^{-r^2/4}$ which holds true by the assumption of the lemma. To see this note that for r with $r \ge t$, $r^{d-1} = e^{(d-1)\log r} \le e^{\frac{t^2}{4\log t}\log r} \le e^{\frac{r^2}{4\log r}\log r} = e^{r^2/4}$ and therefore, $r^{d-1}e^{-r^2/2} \le e^{-r^2/4}$. The third inequality in (C.34) follows from Claim C.1.6

Further, by the refined bound of Theorem 4.7.1, for any $\boldsymbol{\eta}$ with $\|\boldsymbol{\eta}\|_\infty \le t$ and hence $\|\boldsymbol{\eta}\|_2 \le t$ we have

$$\begin{aligned}\int_{\substack{\boldsymbol{\eta}\in\mathbb{R}^d\\ \|\boldsymbol{\eta}\|_2\le t}} \tau(\boldsymbol{\eta})d\boldsymbol{\eta} &\le \int_{\substack{\boldsymbol{\eta}\in\mathbb{R}^d\\ \|\boldsymbol{\eta}\|_2\le t}} \left(\left(6.2R + 1240\ln^{1.5} n_\lambda\right)^d + 1\right) d\boldsymbol{\eta} \\ &\le (2t)^d/\Gamma(d/2+1)\cdot\left(\left(6.2R + 1240\ln^{1.5} n_\lambda\right)^d + 1\right) \\ &= \left(20\sqrt{\ln n_\lambda}\right)^d \left(\left(6.2R + 1240\ln^{1.5} n_\lambda\right)^d + 1\right)\Big/\Gamma(d/2+1). \qquad \text{(C.35)}\end{aligned}$$

The second inequality follows from Claim C.4.1. Combining (C.34) and (C.35) gives the lemma.

Second claim of the corollary: We use the same construction of points as in Theorem 4.7.2. Note that for all $\|\boldsymbol{\eta}\|_2 \le \sqrt{2\ln\frac{n_\lambda}{(4R/3)^d}}$ we have $p(\boldsymbol{\eta}) \ge \frac{1}{3}(4R/3)^d n_\lambda^{-1}$, hence we have:

$$2p(\boldsymbol{\eta}) + (4R/3)^d n_\lambda^{-1} \le 5p(\eta)$$

Hence, by Theorem 4.7.2, we have:

$$\tau(\eta) \ge \frac{1}{640}\left(\frac{R}{3}\right)^d.$$

Therefore,

$$\begin{aligned} s_\lambda(\mathbf{K}) &= \int_{\mathbb{R}^d} \tau(\boldsymbol{\eta})\, d\boldsymbol{\eta} \\ &\geq \int_{\|\boldsymbol{\eta}\|_2 \leq \sqrt{2\ln \frac{n_\lambda}{(4R/3)^d}}} \frac{1}{640}\left(\frac{R}{18}\right)^d d\boldsymbol{\eta} \\ &= \Omega\left(\left(\frac{\sqrt{\pi}R}{3}\sqrt{\log\frac{n_\lambda}{(4R/3)^d}}\right)^d \Big/ \Gamma(d/2+1)\right) \end{aligned} \tag{C.36}$$

The first inequality above is because τ is a non-negative function everywhere. The final line above is due to Claim C.4.1.

C.5 Proof of Theorem 4.5.1

We now show our lower bound on the number of samples required for achieving spectral approximation using classical random Fourier features. This bound is closely related to the leverage score lower bound of Theorem 4.7.2 and the leverage score characterization given by the maximization problem in Lemma 4.7.2.

Our goal is to show that if we take s samples $\boldsymbol{\eta}_1, \boldsymbol{\eta}_2, \ldots, \boldsymbol{\eta}_s$ from the distribution defined by p, for s too small, then there is an $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in \mathbb{R}^n$ such that with constant probability,

$$\boldsymbol{\alpha}^\top(\mathbf{K} + \lambda \mathbf{I}_n)\boldsymbol{\alpha} < \frac{2}{3}\boldsymbol{\alpha}^\top(\mathbf{Z}\mathbf{Z}^* + \lambda \mathbf{I}_n)\boldsymbol{\alpha}. \tag{C.37}$$

Informally, a frequency $\boldsymbol{\eta}$ with high ridge leverage score implies by Lemma 4.7.2 the existence of an $\boldsymbol{\alpha}$ which is concentrated at $\boldsymbol{\eta}$ (i.e. $|\mathbf{z}(\boldsymbol{\eta})^*\boldsymbol{\alpha}|^2$ is large compared to $\|\boldsymbol{\Phi}^*\boldsymbol{\alpha}\|^2_{L_2(\mu)} + \lambda\|\boldsymbol{\alpha}\|_2^2$.) If $\boldsymbol{\eta}$ is not sampled with high enough probability then $\boldsymbol{\alpha}^\top(\mathbf{K} + \lambda \mathbf{I}_n)\boldsymbol{\alpha}$ will not be well approximated. Formally, by (4.3):

$$\begin{aligned} \boldsymbol{\alpha}^T \mathbf{K} \boldsymbol{\alpha} &= \sum_{j,k} \alpha_j \alpha_k \cdot k(\mathbf{x}_j, \mathbf{x}_k) \\ &= \sum_{j,k} \int_{\mathbb{R}^d} e^{-2\pi i \boldsymbol{\eta}^T(\mathbf{x}_j - \mathbf{x}_k)} \alpha_j \alpha_k p(\boldsymbol{\eta})\, d\boldsymbol{\eta} \\ &= \int_{\mathbb{R}^d} \left| \sum_{j=1}^n \alpha_j e^{2\pi i \boldsymbol{\eta}^T \mathbf{x}_j} \right|^2 p(\boldsymbol{\eta})\, d\boldsymbol{\eta}. \end{aligned}$$

Also, by the definition of $\mathbf{Z}$ and φ (see Section 4.2.2), we have

$$\begin{aligned} \boldsymbol{\alpha}^T \mathbf{Z}\mathbf{Z}^* \boldsymbol{\alpha} &= \left\| \sum_{j=1}^n \alpha_j \varphi(\mathbf{x}_j) \right\|_2^2 \\ &= \frac{1}{s} \sum_{k=1}^s \left| \sum_{j=1}^n \alpha_j e^{2\pi i \boldsymbol{\eta}_k^T \mathbf{x}_j} \right|^2, \end{aligned}$$

where $\boldsymbol{\eta}_1, \boldsymbol{\eta}_2, \ldots, \boldsymbol{\eta}_s$ are the s samples from the distribution given by p. Hence, (C.37) is equivalent to

$$\int_{\mathbb{R}^d} p(\boldsymbol{\eta}) \left| \sum_{j=1}^{n} \alpha_j e^{2\pi i \boldsymbol{\eta}^T \mathbf{x}_j} \right|^2 d\boldsymbol{\eta} + \frac{1}{3}\lambda \|\boldsymbol{\alpha}\|_2^2 < \frac{2}{3} \cdot \frac{1}{s} \sum_{k=1}^{s} \left| \sum_{j=1}^{n} \alpha_j e^{2\pi i \boldsymbol{\eta}_k \mathbf{x}_j} \right|^2. \tag{C.38}$$

We again use the same construction of n data points $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$, according to the construction in Definition C.3.2 with $d = 1$. Moreover, we define η^* to be

$$\eta^* = \arg\max_{\eta \in \{\eta_1, \eta_2, \ldots, \eta_s\}} |\eta|.$$

We also let $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ be given by

$$\alpha_j = f_{\eta^*, b, R}(x_j),$$

where $b = \frac{R}{6\sqrt{\ln(n/\lambda)}}$. We show that this choice of data points and vector $\boldsymbol{\alpha}$ satisfies (C.38) with large constant probability.

Lemma C.5.1. *Under the preconditions of Theorem 4.5.1, with probability 0.99 over the samples we have* $|\eta^*| \leq 10\sqrt{\ln n_\lambda}$.

Proof. Let γ be a random variable with density $p(\gamma) = (2\pi)^{-1/2} e^{-\gamma^2/2}$. The limits on n and λ alongside Claim C.1.6 imply that $\Pr\left[|\gamma| \geq 10\sqrt{\ln n_\lambda}\right] < n_\lambda^{-1}/100$. Now, consider the s different random variables $\eta_1, \ldots, \eta_s$. Each of these random variables are distributed identically as γ, so by union-bound the probability that the maximum value is bigger than $10\sqrt{\ln n_\lambda}$ is bounded by $s n_\lambda^{-1}/100$. Since $s \leq n_\lambda$, the probability that the maximum value is bigger than $10\sqrt{\ln n_\lambda}$ is bounded by $1/100$, hence, the lemma follows. □

First, we upper bound the first term on the left side of (C.38). Note that by Lemmas C.5.1 and C.3.7, with probability at least 0.99 over the samples $\eta_1, \eta_2, \ldots, \eta_s$, we have

$$\begin{aligned} \int_{\mathbb{R}^d} p(\eta) \left| \sum_{j=1}^{n} \alpha_j e^{2\pi i \eta^\top \mathbf{x}_j} \right|^2 d\eta &\equiv \|\boldsymbol{\Phi}^* \boldsymbol{\alpha}\|_{L_2(\mu)}^2 \\ &\leq 16 n^2 \left(\frac{3}{4R}\right)^d \cdot p(\eta^*) + 4\lambda n, \end{aligned}$$

where we have let $\eta = \eta^*$. Now, in order to estimate $p(\eta^*)$, note that by Claim C.1.8, we have that with probability at least $1 - e^{-1}$ over the samples $\eta_1, \eta_2, \ldots, \eta_s$, $p(\eta^*) \leq \frac{2\sqrt{2\ln s}}{s}$.

Thus, with probability at least $1 - e^{-1} - 1/100 \geq 1/2$, we have

$$\int_{\mathbb{R}^d} p(\boldsymbol{\eta}) \left| \sum_{j=1}^{n} \alpha_j e^{2\pi i \boldsymbol{\eta}^\top \mathbf{x}_j} \right|^2 d\boldsymbol{\eta} \leq 32 n^2 \cdot \frac{3}{4R} \cdot \frac{\sqrt{2\ln s}}{s} + 4\lambda n. \tag{C.39}$$

Next, we bound the right side of (C.38) from below. Note that by $b = \frac{R}{6\sqrt{\ln(n/\lambda)}}$ and with the choice of $\nu = R$ and $\eta = \eta^*$, Lemma C.3.5 holds true. Therefore, by Lemma C.3.5, we have

$$\begin{aligned}\frac{1}{s}\sum_{k=1}^{s}\left|\sum_{j=1}^{n}\alpha_j e^{2\pi i\eta_k^\top x_j}\right|^2 &\geq \frac{1}{s}\left|\sum_{j=1}^{n}\alpha_j e^{2\pi i\eta^*\cdot x_j}\right|^2 = \frac{1}{s}\left|\boldsymbol{\alpha}^*\mathbf{z}(\eta^*)\right|^2 \\ &\geq \frac{1}{s}\left(\frac{n}{4\cdot 2^d}\right)^2 = \frac{n^2}{64s}. \end{aligned} \tag{C.40}$$

We also require the following estimate of $\|\boldsymbol{\alpha}\|_2^2$, which is provided by Lemma C.3.6:

$$\|\boldsymbol{\alpha}\|_2^2 \leq 4n. \tag{C.41}$$

We also need the bound:

$$32n^2\cdot\frac{3}{4R}\cdot\frac{\sqrt{2\ln s}}{s} \leq \frac{n^2}{98s} \tag{C.42}$$

which holds by the assumptions $R \geq 600\ln^{3/2} n_\lambda$, $s \leq \frac{n_\lambda}{2^{15}}$ and $\lambda \leq \frac{n}{256}$.

Finally, by combining (C.39), (C.40), (C.41), and (C.42) we have that with probability at least $1/2$,

$$\begin{aligned}\int_{\mathbb{R}^d} p(\eta)\left|\sum_{j=1}^{n}\alpha_j e^{2\pi i\eta^\top x_j}\right|^2 d\eta + \frac{1}{3}\lambda\|\boldsymbol{\alpha}\|_2^2 &\leq 32n^2\cdot\frac{3}{4R}\cdot\frac{\sqrt{2\ln s}}{s} + 4\lambda n + \frac{4}{3}\lambda n \\ &\leq \frac{n^2}{98s} + \frac{16\lambda n s}{3s} \\ &\leq \frac{n^2}{98s} + \frac{n^2}{3\cdot 2^{11}s} \\ &< \frac{n^2}{96s} \\ &\leq \frac{2}{3}\cdot\frac{1}{s}\sum_{k=1}^{s}\left|\sum_{j=1}^{n}\alpha_j e^{2\pi i\eta_k^\top x_j}\right|^2. \end{aligned}$$

This completes the proof.

D Near-optimal Sketching of Tensors

D.1 JL Moment Properties of the Tensoring of Sketches

The JL moment property readily implies the following moment bound for the inner product of vectors:

Lemma D.1.1 (Two vector JL Moment Property)**.** *For any $x, y \in \mathbb{R}^d$, if S has the (ε, δ, t)-JL Moment Property, then*

$$\left\| (Sx)^\top (Sy) - x^\top y \right\|_{L^t} \leq \varepsilon \delta^{1/t} \|x\|_2 \|y\|_2. \tag{D.1}$$

Proof. We can assume by linearity of the norms that $\|x\|_2 = \|y\|_2 = 1$. We then use that $x^\top y = (\|x+y\|_2^2 - \|x-y\|_2^2)/4$. Plugging this into the left hand side of (D.1) gives

$$\begin{aligned}
\left\| (Sx)^\top (Sy) - x^\top y \right\|_{L^t} &= \left\| \|Sx+Sy\|_2^2 - \|x+y\|_2^2 - \|Sx-Sy\|_2^2 + \|x-y\|_2^2 \right\|_{L^t} / 4 \\
&\leq \left(\left\| \|S(x+y)\|_2^2 - \|x+y\|_2^2 \right\|_{L^t} + \left\| \|S(x-y)\|_2^2 - \|x-y\|_2^2 \right\|_{L^t} \right) / 4 \\
&\leq \varepsilon \delta^{1/t} \left(\|x+y\|_2^2 + \|x-y\|_2^2 \right) / 4 \qquad \text{(JL moment property)} \\
&= \varepsilon \delta^{1/t} \left(\|x\|_2^2 + \|y\|_2^2 \right) / 2 \\
&= \varepsilon \delta^{1/t}.
\end{aligned}$$

□

The next lemma shows that the direct sum of matrices inherit the JL moment property.

Lemma D.1.2. *Let $t \in \mathbb{N}$ and $\alpha \geq 0$. If $P \in \mathbb{R}^{m_1 \times d_1}$ and $Q \in \mathbb{R}^{m_2 \times d_2}$ are two random matrices (not necessarily independent), such that,*

$$\begin{aligned}
\left\| \|Px\|_2^2 - \|x\|_2^2 \right\|_{L^t} \leq \alpha \|x\|_2^2 \quad \text{and} \quad \mathbb{E}\left[\|Px\|_2^2 \right] = \|x\|_2^2, \\
\left\| \|Qy\|_2^2 - \|y\|_2^2 \right\|_{L^t} \leq \alpha \|y\|_2^2 \quad \text{and} \quad \mathbb{E}\left[\|Qy\|_2^2 \right] = \|y\|_2^2,
\end{aligned}$$

for any vectors $x \in \mathbb{R}^{d_1}$ and $y \in \mathbb{R}^{d_2}$, then for any vector $z \in \mathbb{R}^{d_1+d_2}$,

$$\left\| \|(P \oplus Q)z\|_2^2 - \|z\|_2^2 \right\|_{L^t} \leq \alpha \|z\|_2^2 \quad \textit{and} \quad \mathbb{E}\left[\|(P \oplus Q)z\|_2^2\right] = \|z\|_2^2.$$

Proof. Let $z \in \mathbb{R}^{d_1+d_2}$ and choose $x \in \mathbb{R}^{d_1}$ and $y \in \mathbb{R}^{d_2}$, such that, $z = x \oplus y$. By triangle inequality,

$$\begin{aligned}
\left\| \|(P \oplus Q)z\|_2^2 - \|z\|_2^2 \right\|_{L^t} &= \left\| \|Px\|_2^2 + \|Qy\|_2^2 - \|x\|_2^2 - \|y\|_2^2 \right\|_{L^t} \\
&\leq \left\| \|Px\|_2^2 - \|x\|_2^2 \right\|_{L^t} + \left\| \|Qy\|_2^2 - \|y\|_2^2 \right\|_{L^t} \\
&\leq \alpha \|x\|_2^2 + \alpha \|y\|_2^2 \\
&= \alpha \|z\|_2^2 .
\end{aligned}$$

Moreover, $\mathbb{E}\left[\|(P \oplus Q)z\|_2^2\right] = \mathbb{E}\left[\|Px\|_2^2\right] + \mathbb{E}\left[\|Qy\|_2^2\right] = \|x\|_2^2 + \|y\|_2^2 = \|z\|_2^2$. □

An easy consequence of this lemma is that for any matrix, S, with the (ε, δ, t)-JL Moment Property, $I_k \times S$ has the (ε, δ, t)-JL Moment Property. This follows simply from $I_k \times S = \underbrace{S \oplus S \oplus \ldots \oplus S}_{k \text{ times}}$. Similarly, $S \times I_k$ has the (ε, δ, t)-JL Moment Property, since $S \times I_k$ is just a reordering of the rows of $I_k \times S$, which trivially does not affect the JL Moment Property. The same arguments show that if S has the Strong (ε, δ)-JL Moment Property then $I_k \times S$ and $S \times I_k$ has the Strong (ε, δ)-JL Moment Property.

Proof of Lemma 5.4.1

We will prove a correspondence between Π^p and Π^q. Let $E_1 \in \mathbb{R}^{d \times n}$ be a matrix whose first row is equal to one and is zero elsewhere. By Definition 5.3.3 for any matrix $A \in \mathbb{R}^{d^p \times n}$, $\Pi^p A = \Pi^q \left(A \otimes E_1^{\otimes(q-p)} \right)$. A simple calculation shows that for any matrices $A, B \in \mathbb{R}^{d^p \times n}$ then

$$\left(A \otimes E_1^{\otimes(q-p)} \right)^\top \left(B \otimes E_1^{\otimes(q-p)} \right) = A^\top B \circ \left(E_1^{\otimes(q-p)} \right)^\top \cdot E_1^{\otimes(q-p)} = A^\top B,$$

where $\circ$ denotes the Hadamard product, and the last equality follows since $\left(E_1^{\otimes(q-p)} \right)^\top \cdot E_1^{\otimes(q-p)}$ is all ones matrix. Therefore, $\left\| A \otimes E_1^{\otimes(q-p)} \right\|_F = \|A\|_F$ and $s_\lambda \left(A \otimes E_1^{\otimes(q-p)} \right) = s_\lambda(A)$.

Now assume that Π^q is an $(\varepsilon, \delta, \mu, n)$-Oblivious Subspace Embedding, and let $A \in \mathbb{R}^{d^p \times n}$ and $\lambda \geq 0$ be such that $s_\lambda(A) \leq \mu$. Define $A' = A \otimes E_1^{\otimes(q-p)}$, then

$$\begin{aligned}
&\Pr\left[(1-\varepsilon)(A^\top A + \lambda I_n) \preceq (\Pi^p A)^\top \Pi^p A + \lambda I_n \preceq (1+\varepsilon)(A^\top A + \lambda I_n) \right] \\
&\quad = \Pr\left[(1-\varepsilon)(A'^\top A' + \lambda I_n) \preceq (\Pi^q A')^\top \Pi^q A' + \lambda I_n \preceq (1+\varepsilon)(A'^\top A' + \lambda I_n) \right] \\
&\quad \geq 1 - \delta ,
\end{aligned}$$

where we have used that $s_\lambda(A'^\top A') = s_\lambda(A^\top A) \leq \mu$. This shows that Π^p is an $(\varepsilon, \delta, \mu, n)$-Oblivious Subspace Embedding.

Assume that Π^q has (ε,δ)-Approximate Matrix Multiplication Property, and let $C, D \in \mathbb{R}^{d^p \times n}$. Define $C' = C \otimes E_1^{\otimes(q-p)}$ and $D' = D \otimes E_1^{\otimes(q-p)}$, then

$$\begin{aligned}\Pr\left[\left\|(\Pi^p C)^\top \Pi^p D - C^\top D\right\|_F \geq \varepsilon \|C\|_F \|D\|_F\right] &= \Pr\left[\left\|(\Pi^q C')^\top \Pi^q D' - C'^\top D'\right\|_F \geq \varepsilon \left\|C'\right\|_F \left\|D'\right\|_F\right] \\ &\leq \delta\,,\end{aligned}$$

where we have used that $\left\|C'\right\|_F = \|C\|_F$, $\left\|D'\right\|_F = \|D\|_F$, and $C'^\top D' = C^\top D$. This show that Π^p has (ε,δ)-Approximate Matrix Multiplication Property.

Proof of Lemma 5.4.2

Approximate Matrix Multiplication Let $C, D \in \mathbb{R}^{d\times n}$. We will prove that

$$\left\|\left\|(MC)^\top MD - C^\top D\right\|_F\right\|_{L^t} \leq \varepsilon\delta^{1/t}\|C\|_F\|D\|_F\,. \tag{D.2}$$

Then Markov's inequality will give us the result. Using the triangle inequality together with Lemma D.1.1 we get that:

$$\begin{aligned}\left\|\left\|(MC)^\top MD - C^\top D\right\|_F\right\|_{L^t} &= \left\|\left\|(MC)^\top MD - C^\top D\right\|_F^2\right\|_{L^{t/2}}^{1/2} \\ &= \left\|\sum_{i,j\in[n]}\left((MC_i)^\top MD_j - C_i^\top D_j\right)^2\right\|_{L^{t/2}}^{1/2} \\ &\leq \sqrt{\sum_{i,j\in[n]}\left\|(MC_i)^\top MD_j - C_i^\top D_j\right\|_{L^t}^2} \\ &\leq \sqrt{\sum_{i,j\in[n]}\varepsilon^2\delta^{2/t}\|C_i\|_2^2\left\|D_j\right\|_2^2} \\ &= \varepsilon\delta^{1/t}\|C\|_F\|D\|_F\,.\end{aligned}$$

Using Markov's inequality we now get that

$$\Pr\left[\left\|(MC)^\top MD - C^\top D\right\|_F \geq \varepsilon\|C\|_F\|D\|_F\right] \leq \frac{\left\|\left\|(MC)^\top MD - C^\top D\right\|_F\right\|_{L^t}^t}{\varepsilon^t\|C\|_F^t\|D\|_F^t} \leq \delta\,.$$

Oblivious Subspace Embedding. We will prove that for any $\lambda \geq 0$ and any matrix $A \in \mathbb{R}^{d\times n}$,

$$(1-\varepsilon)(A^\top A + \lambda I_n) \preceq (MA)^\top MA + \lambda I_n \preceq (1+\varepsilon)(A^\top A + \lambda I_n)\,, \tag{D.3}$$

holds with probability at least $1 - \left(\frac{s_\lambda(A^\top A)}{\mu}\right)^t\delta$, which will imply our result. First consider $\lambda > 0$. Then $A^\top A + \lambda I_n$ is positive definite. Thus, by left and right multiplying (D.3) by $(A^\top A+\lambda I_n)^{-1/2}$, we see that (D.3) is equivalent to

$$(1-\varepsilon)I_n \preceq \left(MA(A^\top A+\lambda I_n)^{-1/2}\right)^\top MA(A^\top A+\lambda I_n)^{-1/2} + \lambda(A^\top A+\lambda I_n)^{-1} \preceq (1+\varepsilon)I_n\,.$$

which, in turn, is implied by the following:

$$\left\| \left(MA(A^\top A+\lambda I_n)^{-1/2}\right)^\top MA(A^\top A+\lambda I_n)^{-1/2} + \lambda(A^\top A+\lambda I_n)^{-1} - I_n \right\|_{\text{op}} \le \varepsilon\,.$$

Note that $(A^\top A+\lambda I_n)^{-1/2}A^\top A(A^\top A+\lambda I_n)^{-1/2} = I_n - \lambda(A^\top A+\lambda I_n)^{-1}$. Letting $Z = A(A^\top A+\lambda I_n)^{-1/2}$, we note that it suffices to establish $\left\|(MZ)^\top MZ - Z^\top Z\right\|_{\text{op}} \le \varepsilon$. Using (D.2) together with Markov's inequality we find that

$$\Pr\left[\left\|(MZ)^\top MZ - Z^\top Z\right\|_{\text{op}} \ge \varepsilon\right] \le \Pr[\left\|(MZ)^\top MZ - Z^\top Z\right\|_F \ge \varepsilon] \le \left(\frac{\|Z\|_F^2}{\mu}\right)^t \delta = \left(\frac{s_\lambda(A^\top A)}{\mu}\right)^t \delta\,,$$

where the last equality follows since $\|Z\|_F^2 = \operatorname{tr}\left(A^\top A(A^\top A+\lambda I_n)^{-1}\right) = s_\lambda(A^\top A)$.
To prove the result for $\lambda = 0$ we will use Fatou's lemma.

$$\begin{aligned}
&\Pr\left[\left((1-\varepsilon)A^\top A \preceq (MA)^\top MA \preceq (1+\varepsilon)A^\top A\right)^C\right] \\
&\le \liminf_{\lambda\to 0^+} \Pr\left[\left((1-\varepsilon)(A^\top A+\lambda I_n) \preceq (MA)^\top MA+\lambda I_n \preceq (1+\varepsilon)(A^\top A+\lambda I_n)\right)^C\right] \\
&\le \liminf_{\lambda\to 0^+} \frac{s_\lambda(A^\top A)}{\mu}\delta \\
&= \frac{s_0(A^\top A)}{\mu}\delta,
\end{aligned}$$

where the last equality follows from continuity of $\lambda \mapsto s_\lambda(A^\top A)$.

D.2 Spectral Concentration of the Tensoring of Sketches

We start this section by presenting the definitions and tools from the literature that we use for proving concentration properties of random matrices.

Lemma D.2.1 (Matrix Bernstein Inequality (Tropp et al., 2015, Theorem 6.1.1))**.** *Consider a finite sequence Z_i of independent, random matrices with dimensions $d_1 \times d_2$. Assume that each random matrix satisfies $\mathbb{E}[Z_i] = 0$ and $\|Z_i\|_{op} \le B$ almost surely. Define $\sigma^2 = \max\left\{\|\sum_i \mathbb{E}[Z_i Z_i^*]\|_{op}, \|\sum_i \mathbb{E}[Z_i^* Z_i]\|_{op}\right\}$. Then for every $t > 0$,*

$$\mathbb{P}\left[\left\|\sum_i Z_i\right\|_{op} \ge t\right] \le (d_1+d_2)\cdot\exp\left(\frac{-t^2/2}{\sigma^2 + Bt/3}\right).$$

To analyze the performance of TensorSRHT we need the following claim which shows that with high probability individual entries of the Hadamard transform of a vector with random signs on its entries do not "overshoot the mean energy" by much.

Claim D.2.1. *Let D_1, D_2 be two independent $d \times d$ diagonal matrices, each with i.i.d. Rademacher*

diagonal entries. Also, let H be a $d \times d$ Hadamard matrix. Then, for every $x \in \mathbb{R}^{d^2}$,

$$\Pr_{D_1,D_2}\left[\|[(HD_1)\times(HD_2)]\cdot x\|_\infty \le 4\log_2(d/\delta)\cdot\|x\|_2\right] \ge 1-\delta.$$

Proof. By Claim 5.2.1 we can write that,

$$(HD_1)\times(HD_2) = (H\times H)(D_1\times D_2),$$

where $H\times H$ is indeed a Hadamard matrix of size $d^2\times d^2$ which we denote by H'. The goal is to prove

$$\Pr_{D_1,D_2}\left[\|H'(D_1\times D_2)\cdot x\|_\infty \le 4\log_2(d/\delta)\cdot\|x\|_2\right] \ge 1-\delta.$$

By generalized Khintchine's inequality as per (Ahle et al., 2020, Lemma 4.9), we have that for every $t \ge 1$ and every $j \in [d^2]$ the j^{th} element of $H'(D_1\times D_2)x$ has a bounded t^{th} moment as follows,

$$\left\|\left[H'(D_1\times D_2)x\right]_j\right\|_{L^t} \le t\cdot\|x\|_2.$$

Hence by applying Markov's inequality to the t^{th} moment of $\left|\left[H'(D_1\times D_2)x\right]_j\right|$ for $t=\log_2(d/\delta)$ we find that,

$$\Pr\left[\left|\left[H'(D_1\times D_2)x\right]_j\right| \ge 4\log_2(d/\delta)\cdot\|x\|_2\right] \le \delta/d^2.$$

The claim follows by a union bound over all entries $j\in[d^2]$.

□

D.2.1 Spectral property of Identity×TensorSRHT

In this section we prove Lemma 5.5.3 as follows.

Fix a matrix $U\in\mathbb{R}^{kd\times n}$ with $\|U\|_F^2\le\mu_F$ and $\|U\|_{op}^2\le\mu_2$. Partition U by rows into $d\times n$-sized submatrices $U_1,U_2,\ldots,U_k$ such that $U^\top=\begin{bmatrix}U_1^\top & U_2^\top & \cdots & U_k^\top\end{bmatrix}$. We easily find that,

$$U^\top(I_k\times S)^\top(I_k\times S)U = U_1^\top S^\top SU_1+\cdots U_k^\top S^\top SU_k.$$

The proof first considers the simpler case of a TensorSRHT sketch of rank 1 and then applies the matrix Bernstein inequality as per Lemma B.1.1 in Appendix B.1. Let R denote a rank one TensorSRHT sketch which is a $1\times d$ matrix defined in Definition 5.3.7 by setting $m=1$ as follows,

$$R \overset{\text{def}}{=} P\cdot(HD_1\times HD_2),$$

where $P\in\{0,1\}^{1\times d}$ has one non-zero element at a uniformly random position in $[d]$. It is easy to verify that $S^\top S\in\mathbb{R}^{d\times d}$ is the average of m independent copies of $R^\top R$, i.e., $S^\top S=$

$\frac{1}{m}\sum_{i\in[m]} R_i^\top R_i$, for i.i.d. $R_1, R_2, \ldots, R_m \sim R$. Therefore,

$$U^\top (I_k \times S)^\top (I_k \times S) U = \frac{1}{m} \sum_{i\in[m]} U^\top (I_k \times R_i)^\top (I_k \times R_i) U.$$

Hence, in order to use Lemma B.1.1, we need to bound the maximum operator norm of $U^\top (I_k \times R)^\top (I_k \times R) U$ as well as the operator norm of its second moment. We proceed to upper bound the operator norm of $U^\top (I_k \times R)^\top (I_k \times R) U$. First, define the set

$$\mathscr{E} := \left\{ (D_1, D_2) : \left\| (HD_1 \times HD_2) U_j^i \right\|_\infty^2 \le 16 \log_2^2 \left(\frac{nd\mu_F k}{\epsilon\delta} \right) \cdot \left\| U_j^i \right\|_2^2 \text{ for all } j \in [k] \text{ and all } i \in [n] \right\},$$

where U_j^i is the i^{th} column of U_j. By Claim D.2.1, for every $i \in [n]$ and $j \in [k]$,

$$\Pr_{D_1, D_2} \left[\left\| (HD_1 \times HD_2) U_j^i \right\|_\infty^2 \le 16 \log^2 \left(\frac{nd\mu_F k}{\epsilon\delta} \right) \left\| U_j^i \right\|_2^2 \right] \ge 1 - \frac{\epsilon\delta}{nk\mu_F d}.$$

Thus, by a union bound over all $i \in [n]$ and $j \in [k]$, $\mathscr{E}$ occurs with probability at least $1 - \frac{\epsilon\delta}{\mu_F d}$,

$$\Pr\left[(D_1, D_2) \in \mathscr{E} \right] \ge 1 - \frac{\epsilon\delta}{\mu_F d},$$

where the probability is over the random choice of D_1, D_2. From now on, we fix $(D_1, D_2) \in \mathscr{E}$ and proceed having conditioned on this event.

Upper bounding $\left\| U^\top (I_k \times R)^\top (I_k \times R) U \right\|_{\mathbf{op}}$. Using the fact that $(D_1, D_2) \in \mathscr{E}$, we have,

$$\begin{aligned} L \stackrel{\text{def}}{=} \left\| U^\top (I_k \times R)^\top (I_k \times R) U \right\|_{\text{op}} &= \| U_1^\top R^\top R U_1 + \cdots U_k^\top R^\top R U_k \|_{op} \\ &\le \| RU_1 \|_2^2 + \cdots + \| RU_k \|_2^2 \\ &\le 16 \log_2^2 \left(\frac{nd\mu_F k}{\epsilon\delta} \right) \cdot \| U \|_F^2 \\ &\le 16 \mu_F \cdot \log_2^2 \left(\frac{nd\mu_F k}{\epsilon\delta} \right). \end{aligned}$$

Upper bounding $\left\| \mathbb{E}_P \left[\left(U^\top (I_k \times R)^\top (I_k \times R) U \right)^2 \right] \right\|_{\mathbf{op}}$. Using the fact that $(D_1, D_2) \in \mathscr{E}$, we find that,

$$\begin{aligned} \mathbb{E}_P \left[\left(U^\top (I_k \times R)^\top (I_k \times R) U \right)^2 \right] &\le \mathbb{E}_P \left[\left\| U^\top (I_k \times R)^\top (I_k \times R) U \right\|_{\text{op}} \cdot \left(U^\top (I_k \times R)^\top (I_k \times R) U \right) \right] \\ &= L \cdot \mathbb{E}_P \left[U^\top (I_k \times R)^\top (I_k \times R) U \right]. \end{aligned}$$

Now it suffices to upper bound $\left\| \mathbb{E}_P \left[U^\top (I_k \times R)^\top (I_k \times R) U \right] \right\|_{\text{op}}$. For every $x \in \mathbb{R}^d$ with $\|x\|_2 = 1$,

we have,

$$\begin{aligned} x^\top \mathbb{E}_P\left[U^\top (I_k \times R)^\top (I_k \times R) U\right] x &= \mathbb{E}_P\left[\sum_{j\in[k]} \left(RU_j x\right)^2\right] \\ &= \mathbb{E}_P\left[\left(P(HD_1 \times HD_2)U_j x\right)^2\right] \\ &= \sum_{j\in[k]} \left\|U_j x\right\|_2^2 \\ &= \|Ux\|_2^2 \le \mu_2, \end{aligned}$$

where we used $\mathbb{E}_P\left[\left(P(HD_1 \times HD_2)U_j x\right)^2\right] = \frac{1}{d}\|(HD_1 \times HD_2)U_j x\|^2 = \|U_j x\|_2^2$ for all x.

Since the matrix $\left(U^\top (I_k \times R)^\top (I_k \times R) U\right)^2$ is positive semi-definite for any D_1, D_2, and P, it follows that

$$M \overset{\text{def}}{=} \left\|\mathbb{E}_P\left[\left(U^\top (I_k \times R)^\top (I_k \times R) U\right)^2\right]\right\|_{op} \le L \cdot \mu_2.$$

Combining one-dimensional TensorSRHT transforms. Recall that $(D_1, D_2) \in \mathscr{E}$ with probability at least $1 - \frac{\epsilon\delta}{d\mu_F}$, therefore we have the following conditional expectation,

$$\mathbb{E}\left[\, U^\top (I_k \times R)^\top (I_k \times R) U \middle| (D_1, D_2) \in \mathscr{E}\right] \le \frac{\mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U\right]}{\Pr[(D_1, D_2) \in \mathscr{E}]} \le \frac{U^\top U}{1 - \epsilon\delta / d\mu_F}.$$

Furthermore, by Cauchy-Schwarz we have,

$$\begin{aligned} &\mathbb{E}\left[\, U^\top (I_k \times R)^\top (I_k \times R) U \middle| (D_1, D_2) \in \mathbb{E}\right] \\ &\quad \succeq \mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U\right] - \mathbb{E}\left[\, U^\top (I_k \times R)^\top (I_k \times R) U \middle| (D_1, D_2) \notin \mathscr{E}\right] \cdot \Pr\left[(D_1, D_2) \notin \mathscr{E}\right] \\ &\quad \succeq U^\top U - d\|U\|_F^2 \Pr\left[(D_1, D_2) \notin \mathscr{E}\right] \cdot I_n \\ &\quad \succeq U^\top U - \frac{\epsilon}{2} \cdot I_n. \end{aligned}$$

These two bounds together imply that $\left\|\mathbb{E}\left[\, U^\top (I_k \times R)^\top (I_k \times R) U \middle| (D_1, D_2) \in \mathscr{E}\right] - U^\top U\right\|_{\text{op}} \le \frac{\epsilon}{2}$.

To conclude, we recall that the Gram matrix, $S^\top S \in \mathbb{R}^{d\times d}$, is the average of m independent copies of $R^\top R$, i.e., $S^\top S = \frac{1}{m}\sum_{i\in[m]} R_i^\top R_i$, for i.i.d. $R_1, R_2, \ldots, R_m \sim R$, and therefore,

$$(I_k \times S)^\top (I_k \times S) = \frac{1}{m} \sum_{i\in[m]} (I_k \times R_i)^\top (I_k \times R_i).$$

Now note that the random variables $R_i^\top R_i$ are independent conditioned on $(D_1, D_2) \in \mathscr{E}$. Hence, using the upper bounds $L \le 16\mu_F \cdot \log_2^2\left(\frac{nd\mu_F k}{\epsilon\delta}\right)$ and $M \le L \cdot \mu_2$, which hold when

$(D_1, D_2) \in \mathscr{E}$, we have the following by Lemma B.1.1,

$$\begin{aligned}
&\Pr\left[\left\|U^\top (I\times S)^\top (I\times S)U - U^\top U\right\|_{\mathrm{op}} \geq \epsilon\right] \\
&\quad \leq \Pr\left[\left\|U^\top (I\times S)^\top (I\times S)U - \mathbb{E}\left[U^\top (I\times R)^\top (I\times R)U \middle| (D_1,D_2)\in\mathscr{E}\right]\right\|_{\mathrm{op}} \geq \frac{\epsilon}{2}\middle| (D_1,D_2)\in\mathscr{E}\right] \\
&\qquad + \Pr\left[(D_1,D_2)\notin\mathscr{E}\right] \\
&\quad \leq 8n\cdot\exp\left(-\frac{m\epsilon^2/2}{M+2\epsilon L/3}\right) + \delta/2 \\
&\quad \leq \delta,
\end{aligned}$$

where the last inequality follows by setting $m = \Omega\left(\log\frac{n}{\delta}\log^2(\frac{ndk}{\epsilon\delta})\cdot\frac{\mu_F\mu_2}{\epsilon^2}\right)$. This shows that $I_k \times S$ satisfies the $(\mu_F, \mu_2, \epsilon, \delta, n)$-spectral property.

D.2.2 Spectral property of Identity×OSNAP

In this section we prove Lemma 5.5.4 as follows.

Fix a matrix $U \in \mathbb{R}^{kd\times n}$ with $\|U\|_F^2 \leq \mu_F$ and $\|U\|_{\mathrm{op}}^2 \leq \mu_2$. Partition U by rows into $d\times n$-sized sub-matrices $U_1, U_2, \ldots, U_k$ such that $U^T = \begin{bmatrix} U_1^\top & U_2^\top & \cdots & U_k^\top \end{bmatrix}$. We can easily see that

$$U^\top (I_k\times S)^\top (I_k\times S)U = U_1^\top S^\top S U_1 + \cdots U_k^\top S^\top S U_k.$$

The proof first considers the simpler case of a rank 1 OSNAP sketch and then applies matrix Bernstein inequality. Let R denote a rank one OSNAP, which is a $1\times d$ matrix defined as,

$$R_i \stackrel{\mathrm{def}}{=} \sqrt{\frac{m}{s}}\cdot\delta_i\sigma_i, \tag{D.4}$$

where σ_i for all $i\in[d]$ are i.i.d. Rademacher random variables and δ_i are i.i.d. Bernoulli random variables with $\mathbb{E}[\delta_i] = \frac{s}{m}$. In order to use Lemma B.1.1, we need to bound the maximum operator norm of $U^\top (I_k\times R)^\top (I_k\times R)U$ as well as the operator norm of its second moment. We proceed to upper bound the operator norm of $U^\top (I_k\times R)^\top (I_k\times R)U$. First, define the set

$$\mathscr{E} := \left\{R : (RU_j)^\top RU_j \preceq C\left(\frac{m}{s}\ln^2\left(\frac{ndk\mu_F}{\epsilon\delta}\right)\cdot U_j^\top U_j + \ln\left(\frac{ndk\mu_F}{\epsilon\delta}\right)\|U_j\|_F^2\cdot I_n\right) \text{ for all } j\in[k]\right\},$$

where $C > 0$ is a large enough constant. We show that $\Pr[R\in\mathscr{E}] \geq 1 - \frac{\epsilon\delta}{dm\mu_F}$, where the probability is over the random choices of $\{\sigma_i\}_{i\in[d]}$ and $\{\delta_i\}_{i\in[d]}$. To show this we first need to prove the following claim,

Claim D.2.2. *For every matrix $Z\in\mathbb{R}^{d\times n}$, if we let R be defined as in* (D.4)*, then,*

$$\Pr\left[Z^\top R^\top RZ \preceq C\left(\frac{m}{s}\cdot\ln^2(n/\delta)Z^\top Z + \ln(n/\delta)\|Z\|_F^2 I_n\right)\right] \geq 1-\delta.$$

Proof. The proof is by Matrix Bernstein inequality as per Lemma D.2.1. For any matrix Z let

$A = Z(Z^\top Z + \mu I_n)^{-1/2}$, where $\mu = \frac{s}{m}\frac{1}{\ln(n/\delta)}\|Z\|_F^2$. We can write $RA = \sqrt{\frac{m}{s}}\sum_{i\in[d]}\delta_i\sigma_i A_i$, where A_i is the i^{th} row of A. Note that $\mathbb{E}[\delta_i\sigma_i A_i] = 0$ and $\|\delta_i\sigma_i A_i\|_2 \le \|A_i\|_2 \le \|A\|_{\text{op}}$. Also note that

$$\sum_{i\in[d]}\mathbb{E}\left[(\delta_i\sigma_i A_i)(\delta_i\sigma_i A_i)^*\right] = \sum_{i\in[d]}\frac{s}{m}\|A_i\|_2^2 = \frac{s}{m}\|A\|_F^2,$$

and,

$$\sum_{i\in[d]}\mathbb{E}\left[(\delta_i\sigma_i A_i)^*(\delta_i\sigma_i A_i)\right] = \sum_{i\in[d]}\frac{s}{m}A_i^* A_i = \frac{s}{m}A^\top A.$$

Therefore,

$$\max\left\{\left\|\sum_{i\in[d]}\mathbb{E}\left[(\delta_i\sigma_i A_i)(\delta_i\sigma_i A_i)^*\right]\right\|_{\text{op}}, \left\|\sum_{i\in[d]}\mathbb{E}\left[(\delta_i\sigma_i A_i)^*(\delta_i\sigma_i A_i)\right]\right\|_{\text{op}}\right\} \le \frac{s}{m}\|A\|_F^2.$$

By Lemma D.2.1,

$$\Pr\left[\left\|\sum_{i\in[d]}\delta_i\sigma_i A_i\right\|_{\text{op}} \ge t\right] \le (n+1)\cdot\exp\left(\frac{-t^2/2}{\frac{s}{m}\|A\|_F^2 + \|A\|_{\text{op}}t/3}\right).$$

Hence if $t = \frac{C'}{2}\cdot\left(\sqrt{\frac{s}{m}\ln(n/\delta)}\|A\|_F + \ln(n/\delta)\|A\|_{\text{op}}\right)$, then $\Pr\left[\left\|\sum_{i\in[d]}\delta_i\sigma_i A_i\right\|_{\text{op}} \ge t\right] \le \delta$. By plugging $\|RA\|_{\text{op}}^2 = \frac{m}{s}\cdot\|\sum_{i\in[d]}\delta_i\sigma_i A_i\|_{\text{op}}^2$ into the above we get the following,

$$\Pr\left[\|RA\|_{\text{op}}^2 \le \frac{C'^2}{2}\left(\frac{m}{s}\cdot\ln^2(n/\delta)\|A\|_{\text{op}}^2 + \ln(n/\delta)\|A\|_F^2\right)\right] \ge 1-\delta.$$

Now note that for the choice of $A = Z(Z^\top Z+\mu I_n)^{-1/2}$, we have $\|A\|_{\text{op}}^2 \le \frac{\|Z^\top Z\|_{op}}{\|Z^\top Z\|_{op}^2+\mu} \le 1$ and also $\|A\|_F^2 = \sum_i \frac{\lambda_i(Z^\top Z)}{\lambda_i(Z^\top Z)+\mu} \le \frac{\sum_i\lambda_i(Z^\top Z)}{\mu} = \frac{m}{s}\ln(n/\delta)$. By plugging these into the above we find that,

$$\Pr\left[\left\|RZ(Z^\top Z+\mu I_n)^{-1/2}\right\|_{\text{op}}^2 \le C'^2\cdot\frac{m}{s}\cdot\ln^2(n/\delta)\right] \ge 1-\delta.$$

Hence,

$$(Z^\top Z+\mu I_n)^{-1/2}Z^\top R^\top RZ(Z^\top Z+\mu I_n)^{-1/2} \preceq C\frac{m}{s}\cdot\ln^2(n/\delta)I_n,$$

with probability $1-\delta$, where $C = C'^2$. Composing both sides of the above on the left and right with the positive definite matrix $(Z^\top Z+\mu I_n)^{1/2}$ gives (recall that $\mu = \frac{s}{m}\cdot\frac{\|Z\|_F^2}{\ln(n/\delta)}$),

$$Z^\top R^\top RZ \preceq C\left(\frac{m}{s}\cdot\ln^2(n/\delta)Z^\top Z + \ln(n/\delta)\|Z\|_F^2 I_n\right).$$

□

By applying Claim D.2.2 with failure probability $\frac{\epsilon\delta}{dk\mu_F}$ on each of U_j's and then applying union bound, we find that $\Pr[R\in\mathscr{E}] \ge 1-\frac{\epsilon\delta}{dm\mu_F}$. From now on, we condition on $R\in\mathscr{E}$ and proceed.

Upper bounding $\left\|U^\top (I_k \times R)^\top (I_k \times R) U\right\|_{\mathbf{op}}$. From the fact that we have conditioned on $R \in \mathbb{E}$, note that,

$$\begin{aligned} L \overset{\text{def}}{=} \left\|U^\top (I_k \times R)^\top (I_k \times R) U\right\|_{\text{op}} &= \left\|U_1^\top R^\top R U_1 + \cdots U_k^\top R^\top R U_k\right\|_{\text{op}} \\ &\leq \left\| \sum_{i \in [k]} C\left(\frac{m}{s} \ln^2\left(\frac{ndk\mu_F}{\epsilon\delta}\right) \cdot U_j^\top U_j + \ln\left(\frac{ndk\mu_F}{\epsilon\delta}\right) \|U_j\|_F^2 \cdot I_n\right)\right\|_{\text{op}} \\ &= \left\| C\left(\frac{m}{s} \ln^2\left(\frac{ndk\mu_F}{\epsilon\delta}\right) \cdot U^\top U + \ln\left(\frac{ndk\mu_F}{\epsilon\delta}\right) \|U\|_F^2 \cdot I_n\right)\right\|_{\text{op}} \\ &\leq C\left(\frac{m}{s} \ln^2\left(\frac{ndk\mu_F}{\epsilon\delta}\right) \cdot \mu_2 + \ln\left(\frac{ndk\mu_F}{\epsilon\delta}\right) \cdot \mu_F\right). \end{aligned}$$

Upper bounding $\left\|\mathbb{E}\left[\left(U^\top (I_k \times R)^\top (I_k \times R) U\right)^2\right]\right\|_{\mathbf{op}}$. Using the condition $R \in \mathscr{E}$, it follows that

$$\begin{aligned} &\mathbb{E}\left[\left(U^\top (I_k \times R)^\top (I_k \times R) U\right)^2 \middle| R \in \mathscr{E}\right] \\ &\quad \preceq \mathbb{E}\left[\left\|U^\top (I_k \times R)^\top (I_k \times R) U\right\|_{\text{op}} \cdot \left(U^\top (I_k \times R)^\top (I_k \times R) U\right) \middle| R \in \mathscr{E}\right] \\ &\quad \preceq L \cdot \mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U \middle| R \in \mathscr{E}\right] \\ &\quad \preceq \frac{L}{\Pr[R \in \mathscr{E}]} \cdot U^\top U \end{aligned}$$

where the last line follows from the fact that the random variable $U^\top (I_k \times R)^\top (I_k \times R) U$ is positive semidefinite and the conditional expectation can be upper bounded by its unconditional expectation as follows,

$$\mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U \middle| R \in \mathscr{E}\right] \preceq \frac{\mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U\right]}{\Pr[R \in \mathscr{E}]} = \frac{U^\top U}{\Pr[R \in \mathscr{E}]}.$$

Therefore, we can bound the above operator norm as follows,

$$M \overset{\text{def}}{=} \left\|\mathbb{E}\left[\left(U^\top (I_k \times R)^\top (I_k \times R) U\right)^2\right]\right\|_{\text{op}} \leq 2L \cdot \left\|U^\top U\right\|_{\text{op}} \leq 2L \cdot \mu_2.$$

Combining one-dimensional OSNAP transforms. To conclude, we note that the Gram matrix, $S^\top S \in \mathbb{R}^{d \times d}$, is the average of m independent copies of $R^\top R$ with R defined as in (D.4) – i.e., $S^\top S = \frac{1}{m} \sum_{i \in [m]} R_i^\top R_i$ for i.i.d. $R_1, R_2, \ldots, R_m \sim R$, and therefore,

$$(I_k \times S)^\top (I_k \times S) = \frac{1}{m} \sum_{i \in [m]} (I_k \times R_i)^\top (I_k \times R_i).$$

Note that by union bound, $R_i \in \mathscr{E}$ simultaneously for all $i \in [m]$ with probability at least $1 - \frac{\epsilon\delta}{d\mu_F}$. Now note that the random variables $R_i^\top R_i$ are independent conditioned on $R_i \in \mathscr{E}$ for all

$i \in [m]$. Furthermore, we can bound the following conditional expectation,

$$\begin{aligned}
&\mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U \middle| R \in \mathscr{E} \right] \\
&\quad \succeq \mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U\right] - \mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U \middle| R \notin \mathscr{E} \right] \cdot \Pr[R \notin \mathscr{E}] \\
&\quad \succeq U^\top U - d\|U\|_F^2 \Pr[R \notin \mathscr{E}] \cdot I_n \\
&\quad \succeq U^\top U - d\|U\|_F^2 \cdot \frac{\epsilon}{2} \cdot I_n.
\end{aligned}$$

Moreover, we have shown earlier that,

$$\mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U \middle| R \in \mathscr{E} \right] \preceq \frac{\mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U\right]}{\Pr[R \in \mathscr{E}]} \preceq \frac{U^\top U}{1 - \frac{\epsilon\delta}{d\mu_F}}.$$

These two bounds together imply that,

$$\left\| \mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U \middle| R \in \mathbb{E} \right] - U^\top U \right\|_{op} \leq \epsilon/2.$$

Now, using the upper bounds $L \leq C\left(\frac{m}{s}\ln^2\left(\frac{ndk\mu_F}{\epsilon\delta}\right)\cdot\mu_2 + \ln\left(\frac{ndk\mu_F}{\epsilon\delta}\right)\cdot\mu_F\right)$ and $M \leq 2L\mu_2$, which hold when $R \in \mathscr{E}$, we have that by Lemma B.1.1 (see Appendix B.1),

$$\begin{aligned}
&\Pr\left[\left\| U^\top (I_k \times S)^\top (I_k \times S) U - U^\top U \right\|_{\mathrm{op}} \geq \epsilon \right] \\
&\quad \leq \Pr\left[\left\| U^\top (I_k \times S)^\top (I_k \times S) U - \mathbb{E}\left[U^\top (I_k \times R)^\top (I_k \times R) U \middle| R \in \mathscr{E} \right] \right\|_{\mathrm{op}} \geq \frac{\epsilon}{2} \middle| R \in \mathscr{E} \right] + \Pr[R \notin \mathscr{E}] \\
&\quad \leq 8n \cdot \exp\left(-\frac{m\epsilon^2/8}{M + \epsilon L/3} \right) + \delta/2 \leq \delta,
\end{aligned}$$

where the last inequality follows by setting the parameter $s = \Omega\left(\log^2\left(\frac{ndk\mu_F}{\epsilon\delta}\right)\log\frac{nd}{\delta}\cdot\frac{\mu_2^2}{\epsilon^2}\right)$ and $m = \Omega\left(\log\left(\frac{ndk\mu_F}{\epsilon\delta}\right)\log\frac{nd}{\delta}\cdot\frac{\mu_2\mu_F}{\epsilon^2}\right)$. This shows that $I_k \times S$ satisfies the $(\mu_F, \mu_2, \epsilon, \delta, n)$-spectral property.

Bibliography

Abramovich, Y. A., Abramovich, Y. A., and Aliprantis, C. D. (2002). *An invitation to operator theory*, volume 1. American Mathematical Soc.

Achlioptas, D. (2003). Database-friendly random projections: Johnson-lindenstrauss with binary coins. *Journal of computer and System Sciences*, 66(4):671–687.

Ahle, T. D., Kapralov, M., Knudsen, J. B., Pagh, R., Velingker, A., Woodruff, D. P., and Zandieh, A. (2020). Oblivious sketching of high-degree polynomial kernels. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 141–160. SIAM.

Ailon, N. and Chazelle, B. (2006). Approximate nearest neighbors and the fast johnson-lindenstrauss transform. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 557–563.

Akavia, A. (2010). Deterministic sparse fourier approximation via fooling arithmetic progressions. In *COLT*, pages 381–393.

Akavia, A., Goldwasser, S., and Safra, S. (2003). Proving hard-core predicates using list decoding. In *FOCS*, volume 44, pages 146–159.

Alaoui, A. and Mahoney, M. W. (2015). Fast randomized kernel ridge regression with statistical guarantees. In *Advances in Neural Information Processing Systems*, pages 775–783.

Amrollahi, A., Zandieh, A., Kapralov, M., and Krause, A. (2019). Efficiently learning fourier sparse set functions. In *Advances in Neural Information Processing Systems*, pages 15094–15103.

Avron, H., Clarkson, K. L., and Woodruff, D. P. (2017a). Faster kernel ridge regression using sketching and preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1116–1138.

Avron, H., Clarkson, K. L., and Woodruff, D. P. (2017b). Sharper bounds for regularized data fitting. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*.

Avron, H., Kapralov, M., Musco, C., Musco, C., Velingker, A., and Zandieh, A. (2017c). Random fourier features for kernel ridge regression: Approximation bounds and statistical guarantees. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 253–262. JMLR. org.

Avron, H., Kapralov, M., Musco, C., Musco, C., Velingker, A., and Zandieh, A. (2019). A universal sampling method for reconstructing signals with simple fourier transforms. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, pages 1051–1063, New York, NY, USA. Association for Computing Machinery.

Avron, H., Nguyen, H., and Woodruff, D. (2014). Subspace embeddings for the polynomial kernel. In *Advances in neural information processing systems*, pages 2258–2266.

Ba, K. D., Indyk, P., Price, E., and Woodruff, D. P. (2010). Lower bounds for sparse recovery. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 1190–1197. SIAM.

Bach, F. (2013). Sharp analysis of low-rank kernel matrix approximations. In *Conference on Learning Theory*, pages 185–209.

Bach, F. (2017). On the equivalence between kernel quadrature rules and random feature expansions. *The Journal of Machine Learning Research*, 18(1):714–751.

Bach, F. R. (2010). Structured sparsity-inducing norms through submodular functions. In *Advances in Neural Information Processing Systems*, pages 118–126.

Bah, B., Baldassarre, L., and Cevher, V. (2014). Model-based sketching and recovery with expanders. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1529–1543. SIAM.

Baldassarre, L., Bhan, N., Cevher, V., Kyrillidis, A., and Satpathi, S. (2016). Group-sparse model selection: Hardness and relaxations. *IEEE Transactions on Information Theory*, 62(11):6508–6534.

Baraniuk, R. G., Cevher, V., Duarte, M. F., and Hegde, C. (2010a). Model-based compressive sensing. *IEEE Transactions on information theory*, 56(4):1982–2001.

Baraniuk, R. G., Cevher, V., and Wakin, M. B. (2010b). Low-dimensional models for dimensionality reduction and signal recovery: A geometric perspective. *Proceedings of the IEEE*, 98(6):959–971.

Batson, J. D., Spielman, D. A., and Srivastava, N. (2009). Twice-ramanujan sparsifiers. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 255–262.

Borwein, P. and Erdélyi, T. (1995). *Polynomials and polynomial inequalities*, volume 161. Springer Science & Business Media.

Boufounos, P., Cevher, V., Gilbert, A. C., Li, Y., and Strauss, M. J. (2015). What's the frequency, kenneth?: Sublinear fourier sampling off the grid. *Algorithmica*, 73(2):261–288.

Bourgain, J. (2014). An improved estimate in the restricted isometry problem. In *Geometric aspects of functional analysis*, pages 65–70. Springer.

Boutsidis, C., Mahoney, M. W., and Drineas, P. (2009). An improved approximation algorithm for the column subset selection problem. In *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*, pages 968–977. SIAM.

Bresler, Y. and Macovski, A. (1986). Exact maximum likelihood parameter estimation of superimposed exponential signals in noise. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(5):1081–1089.

Candès, E. J., Romberg, J., and Tao, T. (2006a). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509.

Candès, E. J., Romberg, J. K., and Tao, T. (2006b). Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(8):1207–1223.

Candès, E. J. and Tao, T. (2005). Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215.

Candès, E. J. and Tao, T. (2006). Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425.

Caponnetto, A. and De Vito, E. (2007). Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368.

Cevher, V., Indyk, P., Hegde, C., and Baraniuk, R. (2009). Recovery of clustered sparse signals from compressive measurements. In *International conference on Sampling Theory and Applications (SAMPTA)*, number CONF.

Cevher, V., Kapralov, M., Scarlett, J., and Zandieh, A. (2017). An adaptive sublinear-time block sparse fourier transform. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 702–715.

Charikar, M., Chen, K., and Farach-Colton, M. (2002). Finding frequent items in data streams. In *International Colloquium on Automata, Languages, and Programming*, pages 693–703. Springer.

Charikar, M. S. (2002). Similarity estimation techniques from rounding algorithms. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*, pages 380–388.

Chen, X., Kane, D. M., Price, E., and Song, Z. (2016). Fourier-sparse interpolation without a frequency gap. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 741–750. IEEE.

Chen, X. and Price, E. (2019). Active regression via linear-sample sparsification. In *Conference on Learning Theory*, pages 663–695.

Cheraghchi, M., Guruswami, V., and Velingker, A. (2013). Restricted isometry of fourier matrices and list decodability of random linear codes. *SIAM Journal on Computing*, 42(5):1888–1914.

Cheraghchi, M. and Indyk, P. (2017). Nearly optimal deterministic algorithm for sparse walsh-hadamard transform. *ACM Transactions on Algorithms (TALG)*, 13(3):1–36.

Cipra, B. A. (2000). The best of the 20th century: Editors name top 10 algorithms. *SIAM news*, 33(4):1–2.

Clarkson, K. L. and Woodruff, D. P. (2009). Numerical linear algebra in the streaming model. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 205–214.

Clarkson, K. L. and Woodruff, D. P. (2013). Low rank approximation and regression in input sparsity time. In *Proceedings of the forty-fifth annual ACM symposium on Theory of Computing*, pages 81–90.

Clarkson, K. L. and Woodruff, D. P. (2017). Low-rank approximation and regression in input sparsity time. *Journal of the ACM (JACM)*, 63(6):1–45.

Cohen, A., Davenport, M. A., and Leviatan, D. (2013). On the stability and accuracy of least squares approximations. *Foundations of computational mathematics*, 13(5):819–834.

Cohen, M. B. (2016). Nearly tight oblivious subspace embeddings by trace inequalities. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 278–287. SIAM.

Cohen, M. B., Musco, C., and Musco, C. (2017). Input sparsity time low-rank approximation via ridge leverage score sampling. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1758–1777. SIAM.

Cohen, M. B., Nelson, J., and Woodruff, D. P. (2016a). Optimal approximate matrix product in terms of stable rank. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Cohen, M. B., Nelson, J., and Woodruff, D. P. (2016b). Optimal approximate matrix product in terms of stable rank. 55:11.

Cotter, A., Keshet, J., and Srebro, N. (2011). Explicit approximations of the gaussian kernel. *arXiv preprint arXiv:1109.4603*.

Cover, T. M. and Thomas, J. A. (2012). *Elements of information theory*. John Wiley & Sons.

Cutajar, K., Osborne, M., Cunningham, J., and Filippone, M. (2016). Preconditioning kernel matrices. In *International Conference on Machine Learning*, pages 2529–2538.

Dasgupta, A., Kumar, R., and Sarlós, T. (2010). A sparse johnson: Lindenstrauss transform. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 341–350.

de Prony, G. R. (1795). *Essai experimental et analytique sur les lois de la dilatabilite des fluides elastiques et sur celles de la force expansive de la vapeur de l'eau et de la vapeur de l'alcool a differentes temperatures, par R. Prony*. Journal de l'Ecole Polytechnique.

Deng, C. Y. (2011). A generalization of the sherman–morrison–woodbury formula. *Applied Mathematics Letters*, 24(9):1561–1564.

Deshpande, A. and Rademacher, L. (2010). Efficient volume sampling for row/column subset selection. In *2010 ieee 51st annual symposium on foundations of computer science*, pages 329–338. IEEE.

Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306.

Donoho, D. L. and Stark, P. B. (1989). Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 49(3):906–931.

Drineas, P., Kannan, R., and Mahoney, M. W. (2006a). Fast monte carlo algorithms for matrices i: Approximating matrix multiplication. *SIAM Journal on Computing*, 36(1):132–157.

Drineas, P., Magdon-Ismail, M., Mahoney, M. W., and Woodruff, D. P. (2012). Fast approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research*, 13(Dec):3475–3506.

Drineas, P. and Mahoney, M. W. (2016). Randnla: randomized numerical linear algebra. *Communications of the ACM*, 59(6):80–90.

Drineas, P., Mahoney, M. W., and Muthukrishnan, S. (2006b). Sampling algorithms for l 2 regression and applications. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 1127–1136. Society for Industrial and Applied Mathematics.

Drineas, P., Mahoney, M. W., Muthukrishnan, S., and Sarlós, T. (2011). Faster least squares approximation. *Numerische mathematik*, 117(2):219–249.

Eldar, Y. C. (2015). *Sampling theory: Beyond bandlimited systems*. Cambridge University Press.

Eldar, Y. C. and Unser, M. (2006). Nonideal sampling and interpolation from noisy observations in shift-invariant spaces. *IEEE Transactions on Signal Processing*, 54(7):2636–2651.

ESRI, A. (2001). Environmental systems research institute. *California, USA*, 631.

Feller, W. (2008). *An introduction to probability theory and its applications*, volume 2. John Wiley & Sons.

Foucart, S. and Rauhut, H. (2013). An invitation to compressive sensing. In *A mathematical introduction to compressive sensing*, pages 1–39. Springer.

Friston, K. J., Jezzard, P., and Turner, R. (1994). Analysis of functional mri time-series. *Human brain mapping*, 1(2):153–171.

Frostig, R., Musco, C., Musco, C., and Sidford, A. (2016). Principal component projection without principal component analysis. In *International Conference on Machine Learning*, pages 2349–2357.

Fuderer, M. (1989). Ringing artefact reduction by an efficient likelihood improvement method. In *Science and Engineering of Medical Imaging*, volume 1137, pages 84–91. International Society for Optics and Photonics.

Ghasemi, A. and Sousa, E. S. (2008). Spectrum sensing in cognitive radio networks: requirements, challenges and design trade-offs. *IEEE Communications magazine*, 46(4):32–39.

Ghazi, B., Hassanieh, H., Indyk, P., Katabi, D., Price, E., and Shi, L. (2013). Sample-optimal average-case sparse fourier transform in two dimensions. In *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1258–1265. IEEE.

Gilbert, A. C., Guha, S., Indyk, P., Muthukrishnan, S., and Strauss, M. (2002). Near-optimal sparse fourier representations via sampling. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*, pages 152–161.

Gilbert, A. C., Indyk, P., Iwen, M., and Schmidt, L. (2014). Recent developments in the sparse fourier transform: A compressed fourier transform for big data. *IEEE Signal Processing Magazine*, 31(5):91–100.

Gilbert, A. C., Muthukrishnan, S., and Strauss, M. (2005). Improved time bounds for near-optimal sparse fourier representations. In *Wavelets XI*, volume 5914, page 59141A. International Society for Optics and Photonics.

Gilbert, A. C., Strauss, M. J., and Tropp, J. A. (2008). A tutorial on fast fourier sampling. *IEEE Signal processing magazine*, 25(2):57–66.

Goel, S., Kanade, V., Klivans, A., and Thaler, J. (2017). Reliably learning the relu in polynomial time. In *Conference on Learning Theory*, pages 1004–1042.

Goldreich, O. and Levin, L. A. (1989). A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 25–32.

Golovin, D., Solnik, B., Moitra, S., Kochanski, G., Karro, J., and Sculley, D. (2017). Google vizier: A service for black-box optimization. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1487–1495.

Greengard, L. and Rokhlin, V. (1986). A fast algorithm for particle simulations. Technical report, YALE UNIV NEW HAVEN CT DEPT OF COMPUTER SCIENCE.

Hagerup, T., Mehlhorn, K., and Munro, J. I. (1993). Maintaining discrete probability distributions optimally. In *International Colloquium on Automata, Languages, and Programming*, pages 253–264. Springer.

Handcock, M. S. and Stein, M. L. (1993). A bayesian analysis of kriging. *Technometrics*, 35(4):403–410.

Hardy, G. H., Littlewood, J. E., et al. (1914). Some problems of diophantine approximation: part i. the fractional part of nkθ. *Acta mathematica*, 37:155–191.

Hassanieh, H., Adib, F., Katabi, D., and Indyk, P. (2012a). Faster gps via the sparse fourier transform. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 353–364.

Hassanieh, H., Indyk, P., Katabi, D., and Price, E. (2012b). Nearly optimal sparse fourier transform. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 563–578.

Hassanieh, H., Indyk, P., Katabi, D., and Price, E. (2012c). Simple and practical algorithm for sparse fourier transform. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pages 1183–1194. SIAM.

Hassanieh, H., Shi, L., Abari, O., Hamed, E., and Katabi, D. (2014). Ghz-wide sensing and decoding using the sparse fourier transform. In *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*, pages 2256–2264. IEEE.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.

Haviv, I. and Regev, O. (2017). The restricted isometry property of subsampled fourier matrices. In *Geometric aspects of functional analysis*, pages 163–179. Springer.

Hazan, E. E., Klivans, A., and Yuan, Y. (2018). Hyperparameter optimization: A spectral approach. In *6th International Conference on Learning Representations, ICLR 2018*.

Heider, S., Kunis, S., Potts, D., and Veit, M. (2013). A sparse prony fft. In *Proc. 10th International Conference on Sampling Theory and Applications (SAMPTA)*, pages 572–575.

Heisenberg, W. K. (1930). *The physical principles of the quantum theory*. Dover.

Helmberg, G. (2008). *Introduction to spectral theory in Hilbert space*. Courier Dover Publications.

Hunter, J. K. and Nachtergaele, B. (2001). *Applied analysis*. World Scientific Publishing Company.

Indyk, P. and Kapralov, M. (2014). Sample-optimal fourier sampling in any fixed dimension. In *IEEE Symp. Found. Comp. Sci.(FOCS)*, volume 10.

Indyk, P., Kapralov, M., and Price, E. (2014). (nearly) sample-optimal sparse fourier transform. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 480–499. SIAM.

Indyk, P. and Razenshteyn, I. (2013). On model-based rip-1 matrices. In *International Colloquium on Automata, Languages, and Programming*, pages 564–575. Springer.

Iwen, M. A. (2010). Combinatorial sublinear-time fourier algorithms. *Foundations of Computational Mathematics*, 10(3):303–338.

Iwen, M. A. (2013). Improved approximation guarantees for sublinear-time fourier algorithms. *Applied And Computational Harmonic Analysis*, 34(1):57–82.

Johnson, W. B. and Lindenstrauss, J. (1984). Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, 26(189-206):1.

Kane, D. M. and Nelson, J. (2014). Sparser johnson-lindenstrauss transforms. *Journal of the ACM (JACM)*, 61(1):1–23.

Kannan, R., Vempala, S., and Woodruff, D. (2014). Principal component analysis and higher correlations for distributed data. In *Conference on Learning Theory*, pages 1040–1057.

Kapralov, M. (2016). Sparse fourier transform in any constant dimension with nearly-optimal sample complexity in sublinear time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 264–277.

Kapralov, M. (2017). Sample efficient estimation and recovery in sparse fft via isolation on average. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 651–662. Ieee.

Kapralov, M., Nouri, N., Razenshteyn, I., Velingker, A., and Zandieh, A. (2020). Scaling up kernel ridge regression via locality sensitive hashing. In *23rd International Conference on Artificial Intelligence and Statistics*.

Kapralov, M., Velingker, A., and Zandieh, A. (2019). Dimension-independent sparse fourier transform. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2709–2728. SIAM.

Kotelnikov, V. A. (1933). On the carrying capacity of the" either" and wire in telecommunications. In *Material for the First All-Union Conference on Questions of Communication (Russian), Izd. Red. Upr. Svyzai RKKA, Moscow, 1933*.

Kushilevitz, E. and Mansour, Y. (1993). Learning decision trees using the fourier spectrum. *SIAM Journal on Computing*, 22(6):1331–1348.

Landau, H. (1967). Sampling, data transmission, and the nyquist rate. *Proceedings of the IEEE*, 55(10):1701–1706.

Landau, H. J. and Pollak, H. O. (1961). Prolate spheroidal wave functions, fourier analysis and uncertainty—ii. *Bell System Technical Journal*, 40(1):65–84.

Landau, H. J. and Pollak, H. O. (1962). Prolate spheroidal wave functions, fourier analysis and uncertainty—iii: the dimension of the space of essentially time-and band-limited signals. *Bell System Technical Journal*, 41(4):1295–1336.

Lawlor, D., Wang, Y., and Christlieb, A. (2013). Adaptive sub-linear time fourier algorithms. *Advances in Adaptive Data Analysis*, 5(01):1350003.

Le, Q., Sarlós, T., and Smola, A. (2013). Fastfood-approximating kernel expansions in loglinear time. In *Proceedings of the international conference on machine learning*, volume 85.

Lettington, A. H. and Hong, Q. H. (1995). Image restoration using a lorentzian probability model. *Journal of Modern Optics*, 42(7):1367–1376.

Lin, M., Vinod, A. P., and See, C. M. S. (2011). A new flexible filter bank for low complexity spectrum sensing in cognitive radios. *Journal of Signal Processing Systems*, 62(2):205–215.

Lorch, L. (1983). Alternative proof of a sharpened form of bernstein's inequality for legendre polynomials. *Applicable Analysis*, 14(3):237–240.

Lu, Y., Dhillon, P., Foster, D. P., and Ungar, L. (2013). Faster ridge regression via the subsampled randomized hadamard transform. In *Advances in neural information processing systems*, pages 369–377.

Lustig, M., Donoho, D., and Pauly, J. M. (2007). Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 58(6):1182–1195.

Mahoney, M. W. and Drineas, P. (2009). Cur matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697–702.

Mansour, Y. (1995). Randomized interpolation and approximation of sparse polynomials. *SIAM Journal on Computing*, 24(2):357–368.

Merhi, S., Zhang, R., Iwen, M. A., and Christlieb, A. (2019). A new class of fully discrete sparse fourier transforms: Faster stable implementations with guarantees. *Journal of Fourier Analysis and Applications*, 25(3):751–784.

Minsker, S. (2017). On some extensions of bernstein's inequality for self-adjoint operators. *Statistics & Probability Letters*, 127:111–119.

Mishali, M. and Eldar, Y. C. (2009). Blind multiband signal reconstruction: Compressed sensing for analog signals. *IEEE Transactions on signal processing*, 57(3):993–1009.

Mishali, M. and Eldar, Y. C. (2010). From theory to practice: Sub-nyquist sampling of sparse wideband analog signals. *IEEE Journal of selected topics in signal processing*, 4(2):375–391.

Moitra, A. (2015). Super-resolution, extremal functions and the condition number of vandermonde matrices. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 821–830.

Musco, C. and Musco, C. (2017). Recursive sampling for the nystrom method. In *Advances in Neural Information Processing Systems*, pages 3833–3845.

Musco, C. and Woodruff, D. P. (2017). Sublinear time low-rank approximation of positive semidefinite matrices. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 672–683. IEEE.

Nakos, V., Song, Z., and Wang, Z. (2019). (nearly) sample-optimal sparse fourier transform in any dimension; ripless and filterless. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1568–1577. IEEE.

Nelson, J. and Nguyên, H. L. (2013). Osnap: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *2013 ieee 54th annual symposium on foundations of computer science*, pages 117–126. IEEE.

Nguyen, N. H., Drineas, P., and Tran, T. D. (2015). Tensor sparsification via a bound on the spectral norm of random tensors. *Information and Inference: A Journal of the IMA*, 4(3):195–229.

Nyquist, H. (1928). Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644.

Ogawa, H. (1988). An operator pseudo-inversion lemma. *SIAM Journal on Applied Mathematics*, 48(6):1527–1531.

Pagh, R. (2013). Compressed matrix multiplication. *ACM Transactions on Computation Theory (TOCT)*, 5(3):1–17.

Pauwels, E., Bach, F., and Vert, J.-P. (2018). Relating leverage scores and density using regularized christoffel functions. In *Advances in Neural Information Processing Systems*, pages 1663–1672.

Pawar, S. and Ramchandran, K. (2013). Computing a k-sparse n-length discrete fourier transform using at most 4k samples and o (k log k) complexity. In *2013 IEEE International Symposium on Information Theory*, pages 464–468. IEEE.

Pesquet-Popescu, B. and Véhel, J. L. (2002). Stochastic fractal models for image processing. *IEEE Signal Processing Magazine*, 19(5):48–62.

Pettis, B. J. (1938). On integration in vector spaces. *Transactions of the American Mathematical Society*, 44(2):277–304.

Pham, N. and Pagh, R. (2013). Fast and scalable polynomial kernels via explicit feature maps. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 239–247.

Pilanci, M. and Wainwright, M. J. (2015). Randomized sketches of convex programs with sharp guarantees. *IEEE Transactions on Information Theory*, 61(9):5096–5115.

Pisarenko, V. F. (1973). The retrieval of harmonics from a covariance function. *Geophysical Journal International*, 33(3):347–366.

Price, E. and Song, Z. (2015). A robust sparse fourier transform in the continuous setting. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 583–600. IEEE.

Price, E. and Woodruff, D. P. (2011). (1+ eps)-approximate sparse recovery. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 295–304. IEEE.

Pruessmann, K. P., Weiger, M., Scheidegger, M. B., and Boesiger, P. (1999). Sense: sensitivity encoding for fast mri. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 42(5):952–962.

Rahimi, A. and Recht, B. (2008). Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pages 1177–1184.

Rahimi, A. and Recht, B. (2009). Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in neural information processing systems*, pages 1313–1320.

Ramani, S., Van De Ville, D., and Unser, M. (2005). Sampling in practice: Is the best reconstruction space bandlimited? In *IEEE International Conference on Image Processing 2005*, volume 2, pages II–153. IEEE.

Rasmussen, C. E. (2003). Gaussian processes in machine learning. In *Summer School on Machine Learning*, pages 63–71. Springer.

Ripley, B. D. (1991). *Statistical inference for spatial processes*. Cambridge university press.

Ripley, B. D. (2005). *Spatial statistics*, volume 575. John Wiley & Sons.

Robbins, H. (1952). Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535.

Rokhlin, V., Xiao, H., and Yarvin, N. (2001). Prolate spheroidal wavefunctions, quadrature and interpolation. *Inverse problems*, 17(4):805.

Rudelson, M. and Vershynin, R. (2008). On sparse reconstruction from fourier and gaussian measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 61(8):1025–1045.

Rudi, A., Camoriano, R., and Rosasco, L. (2015). Less is more: Nyström computational regularization. In *Advances in Neural Information Processing Systems*, pages 1657–1665.

Rudi, A. and Rosasco, L. (2017). Generalization properties of learning with random features. In *Advances in Neural Information Processing Systems*, pages 3215–3225.

Sarlos, T. (2006). Improved approximation algorithms for large matrices via random projections. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 143–152. IEEE.

Shannon, C. E. (1949). Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21.

Shawe-Taylor, J., Cristianini, N., et al. (2004). *Kernel methods for pattern analysis*. Cambridge university press.

Slepian, D. and Pollak, H. O. (1961). Prolate spheroidal wave functions, fourier analysis and uncertainty—i. *Bell System Technical Journal*, 40(1):43–63.

Spielman, D. A. and Srivastava, N. (2008). Graph sparsification by effective resistances. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 563–568.

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. (2010). Gaussian process optimization in the bandit setting: no regret and experimental design. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 1015–1022.

Stojnic, M., Parvaresh, F., and Hassibi, B. (2009). On the reconstruction of block-sparse signals with an optimal number of measurements. *IEEE Transactions on Signal Processing*, 57(8):3075–3085.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Totik, V. (2000). Asymptotics for christoffel functions for general measures on the real line. *Journal d'Analyse Mathématique*, 81(1):283–303.

Tropp, J. A. (2011). Improved analysis of the subsampled randomized hadamard transform. *Advances in Adaptive Data Analysis*, 3(01n02):115–126.

Tropp, J. A. et al. (2015). An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning*, 8(1-2):1–230.

Valiant, G. (2012). Finding correlations in subquadratic time, with applications to learning parities and juntas. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 11–20. IEEE.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press.

Whitaker, E. (1915). On the functions which are represented by the expansion of interpolating theory. In *Proc. Roy. Soc. Edinburgh*, volume 35, pages 181–194.

Williams, V. V. (2012). Multiplying matrices faster than coppersmith-winograd. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 887–898.

Woodruff, D. P. (2014). Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157.

Worsley, K. J., Marrett, S., Neelin, P., Vandal, A. C., Friston, K. J., and Evans, A. C. (1996). A unified statistical approach for determining significant signals in images of cerebral activation. *Human brain mapping*, 4(1):58–73.

Wouk, A. (1966). A note on square roots of positive operators. *SIAM Review*, 8(1):100–102.

Xiao, H. (2002). Prolate spheroidal wave functions, quadrature, interpolation, and asymptotic formulae. *PhD thesis, Yale University*.

Zhang, T. (2005). Learning bounds for kernel regression using effective data dimensionality. *Neural Computation*, 17(9):2077–2098.

Zhang, Y., Duchi, J., and Wainwright, M. (2013). Divide and conquer kernel ridge regression. In *Conference on learning theory*, pages 592–617.

Amir Zandieh

EPFL IC IIF THL4, INJ 110
Station 14, CH-1015 Lausanne
Phone: +41 (78) 885-5481
Email: amir.zandieh@epfl.ch
OrcID: 0000-0002-1294-9390

Research Interests

- Algorithm Design, Machine Learning, Sublinear Algorithms, Numerical Linear Algebra, Sparse Fourier Transform

Education

- **École Polytechnique Fédérale de Lausanne**, Switzerland (September 2015 – present)
  PhD Candidate in Computer Science – Supervisor: *Michael Kapralov.*
- **Sharif University**, Iran (2010 – 2015)
  BSc in Two Majors: Electrical Engineering and Computer Science – GPA: 19/20.

Research Visits

- **Carnegie Mellon University**, School of Computer Science (September 2019 – January 2020)
  Hosted by *David Woodruff.*

Awards and Honors

- *Swiss National Science Foundation (SNSF) Post-Doctoral Fellowship*, over CHF 75'000 (2020)
- *Ranked 16 in Iran's university entrance exam*, among 300,000 test takers (2010)
- *Multiple scholarships for high GPA from Sharif University* (2010 – 2015)

Publications

- (**ICML 2020**) D. Woodruff, A. Zandieh. *Near Input Sparsity Time Kernel Embeddings via Adaptive Sampling.* In 37th International Conference on Machine Learning
- (**AISTATS 2020**) M. Kapralov, N. Nouri, I. Razenshteyn, A. Velingker, A. Zandieh. *Scaling up Kernel Ridge Regression via Locality Sensitive Hashing.* In 23rd International Conference on Artificial Intelligence and Statistics
- (**SODA 2020**) T. Ahle, M. Kapralov, J. Knudsen, R. Pagh, A. Velingker, D. Woodruff, A. Zandieh. *Oblivious Sketching of High-Degree Polynomial Kernels.* In 31st Annual ACM-SIAM Symposium on Discrete Algorithms
- (**NeurIPS 2019**) A. Amrollahi, A. Zandieh, M. Kapralov, A. Krause. *Efficiently Learning Fourier Sparse Set Functions.* In 33rd Conference on Neural Information Processing Systems, spotlight talk

- (**STOC 2019**) H. Avron, M. Kapralov, C. Musco, C. Musco, A. Velingker, A. Zandieh. *A Universal Sampling Method for Reconstructing Signals with Simple Fourier Transforms.* In 51st Annual ACM Symposium on the Theory of Computing
- (**SODA 2019**) M. Kapralov, A. Velingker, A. Zandieh. *Dimension-independent Sparse Fourier Transform.* In 30th Annual ACM-SIAM Symposium on Discrete Algorithms
- (**ICML 2018**) A. Norouzi-Fard, J. Tarnawski, S. Mitrović, A. Zandieh, A. Mousavifar, O. Svensson. *Beyond* 1/2*-Approximation for Submodular Maximization on Massive Data Streams.* In 35th International Conference on Machine Learning, long talk
- (**ICML 2017**) H. Avron, M. Kapralov, C. Musco, C. Musco, A. Velingker, A. Zandieh. *Random Fourier features for kernel ridge regression: Approximation bounds and statistical guarantees.* In 34th International Conference on Machine Learning, long talk
- (**STOC 2017**) V. Cevher, M. Kapralov, J. Scarlett, A. Zandieh. *An adaptive sublinear-time block sparse Fourier transform.* In 49th Annual ACM Symposium on the Theory of Computing

Teaching Experience

- Teaching Assistant, EPFL:
  Algorithms (Fall 2016, Fall 2017, Fall 2018)
  Sublinear algorithms for big data analysis (Spring 2018, Spring 2019),
  Mathematics for Mise à niveau (Spring 2017)
- Several lab assistance for undergraduate EE courses, Sharif University (2013–2015)

Professional Service

- Reviewer for: *FOCS* (2020), *ICML* (2019, 2020), *SODA* (2019), *NeurIPS* (2019, 2020), *ICALP* (2017, 2018, 2019)
- Journal Reviewer for: *IEEE Transactions on Information Theory*

Invited Talks

- SIAM SEAS symposium on *Fast Algorithms, Sparsity and Approximation* (September 2019)
  University of Tennessee
- Workshop on *Data Summarization* (March 2018)
  Warwick University
- *Winter Seminar Series*, the second (December 2016)
  Sharif University

Programming Skills

- C++, Python, MATLAB, Verilog Hardware Description Language