

Extensions of Peer Prediction Incentive Mechanisms

Présentée le 8 avril 2024

Faculté informatique et communications
Laboratoire d'intelligence artificielle
Programme doctoral en informatique et communications

pour l'obtention du grade de Docteur ès Sciences

par

Adam Julian RICHARDSON

Acceptée sur proposition du jury

Prof. E. Telatar, président du jury
Prof. B. Faltings, directeur de thèse
Dr G. Radanovic, rapporteur
Dr R. Jurca, rapporteur
Prof. K. Aberer, rapporteur

To my parents and my brother. . .

Acknowledgements

This thesis was made possible by the guidance and support of many people, to whom I feel immense gratitude.

First and foremost I must thank my advisor, Prof. Boi Faltings. His unwavering patience allowed me to push through some of my most difficult periods. I must also thank Karin Gétaz for holding the lab together. I thank my colleagues in the Artificial Intelligence Lab who were sources of motivation and inspiration: Prof. Aris Filos-Ratsikas, Dr. Igor Kulev, Dr. Fei Mi, Dr. Aleksei Triastcyn, Dr. Naman Goel, Dr. Panayiotis Danassis, Dr. Diego Antognini, Dr. Debjit Paul, and PhD Candidates Ljubomir Rokvic, Zeki Erden, and Shaobo Cui. I was also given invaluable aid and guidance from other members of EPFL faculty, especially Profs. Rüdiger Urbanke and Emre Telatar. Finally, I thank the doctoral school of EDIC, EPFL, and the government of Switzerland for granting me this opportunity.

There are a number of friends I must thank for being sources of distraction and moral support. There are the friends I made during my time living in Lausanne. I must especially thank Nico Baldy for inviting me to vacation with him in Barcelona, and for being a pillar of the Lausanne MTG community. I also received support from friends at home in the United States, for which I am grateful. I thank Alex Scarpa, George Solter, Nicole Tipple, and Joe Custodio for their friendship.

Finally, without the love and support of my family, I would not have made it this far. My most significant acknowledgment is reserved for them, who have given me all the opportunities in my life. I dedicate this thesis to them.

Abstract

As large, data-driven artificial intelligence models become ubiquitous, guaranteeing high data quality is imperative for constructing models. Crowdsourcing, community sensing, and data filtering have long been the standard approaches to guaranteeing or improving data quality. The underlying theory, mainly incentive mechanism design, is often limited in its scope of applicability. A subset of incentive mechanisms designed to handle unverifiable or inherently subjective data - Peer Prediction mechanisms - is generally only applicable to settings where the data signal comes from a discrete distribution. In this thesis, we expand the scope of applicability of Peer Prediction mechanisms in two parts.

In the first part, we address a constrained extension of Peer Prediction that is applicable to machine learning. A data collecting entity, known as a Center, may not need to learn a joint distribution of (x, y) pairs. It may only need to learn a parameterized model that minimizes a loss function on the joint distribution. We analyze a statistical measure known as *Influence*, which can be interpreted as a form of Peer Prediction. We will show that the Peer Truth Serum (PTS) is a special case of Influence, and that Influence has desirable game-theoretic properties as an incentive mechanism.

We then take the analysis of Influence into the regime of data filtering, which is uniquely challenging compared to crowdsourcing. We use asymptotic analysis to show that, in the limit of infinite samples, the ability to filter training data using Influence is constrained by the degree of corruption in the validation data. However, finite sample analysis reveals that one can exceed the quality of the validation data if conditions are met regarding higher moments of the data models.

In the second part, we move on from this more constrained extension to the most general extension of Peer Prediction: learning arbitrary distributions. Many crowdsourcing problems involve absolutely continuous distributions, such as Gaussian distributions. The standard approach is to discretize the space and apply a discrete Peer Prediction mechanism. This approach has numerous issues: coarse discretizations result in inaccurate approximations of the distribution and loose incentives, while fine discretizations result in volatile payments, which tend to fail in real world applications. We expand the theory of Peer Prediction, rather than seek a better implementation of current theory. We consider two approaches.

In the first approach, one can discretize the space, which we call partitioning into bins, but pick from a set of partitions rather than just one. In this regime, the notion of peer matching

in Peer Prediction is generalized with the concept of *Peer Neighborhoods*. With a reasonable strengthening of the Agent update condition, we obtain a valid extension of the PTS on arbitrary distributions.

The partitioning approach for arbitrary distributions reveals a more precise theory. By changing perspective from partitioning according to the Lebesgue measure on the space of reports to partitioning according to the public probability measure, we obtain a payment function that doesn't rely on discretization. Using this function as the basis for a mechanism, a *Continuous Truth Serum*, reveals solutions to other underlying problems with Peer Prediction, such as the unobserved category problem.

Key words: Game Theory, Incentive Mechanisms, Peer Prediction, Machine Learning, Data Valuation, Data Filtering

Résumé

Alors que les grands modèles d'intelligence artificielle basés sur des données deviennent omniprésents, il est impératif de garantir la qualité des données pour construire les modèles. Le crowdsourcing, la détection communautaire et le filtrage des données sont depuis longtemps des approches standard pour garantir ou améliorer la qualité des données. La théorie sous-jacente, principalement la conception de mécanismes d'incitation, est souvent limitée dans son champ d'application. Un sous-ensemble de mécanismes incitatifs conçus pour traiter des données invérifiables ou intrinsèquement subjectives - les mécanismes Peer Prediction - n'est généralement applicable qu'à des contextes où le signal de données provient d'une distribution discrète. Dans cette thèse, nous élargissons le champ d'application des mécanismes Peer Prediction en deux parties.

Dans la première partie, nous abordons une extension contrainte du Peer Prediction qui est applicable à l'apprentissage automatique. Une entité de collecte de données, connue sous le nom de Centre, peut ne pas avoir besoin d'apprendre une distribution conjointe de paires (x, y) . Elle peut seulement avoir besoin d'apprendre un modèle paramétré qui minimise une fonction de perte sur la distribution conjointe. Nous analysons une mesure statistique connue sous le nom de *Influence*, qui peut être interprétée comme une forme de Peer Prediction. Nous montrerons que le Peer Truth Serum (PTS) est un cas particulier de l'*Influence*, et que l'*Influence* possède des propriétés souhaitables en théorie des jeux en tant que mécanisme d'incitation.

Nous poussons ensuite l'analyse de l'*Influence* dans le régime du filtrage des données, qui est un défi unique par rapport au crowdsourcing. Nous utilisons une analyse asymptotique pour montrer que, dans la limite d'échantillons infinis, la capacité à filtrer les données d'apprentissage à l'aide d'*Influence* est limitée par le degré de corruption des données de validation. Cependant, l'analyse des échantillons finis révèle que l'on peut dépasser la qualité des données de validation si les conditions sont remplies en ce qui concerne les moments supérieurs des modèles de données.

Dans la deuxième partie, nous passons de cette extension plus contrainte à l'extension la plus générale du Peer Prediction : l'apprentissage de distributions arbitraires. De nombreux problèmes de crowdsourcing impliquent des distributions absolument continues, telles que les distributions gaussiennes. L'approche standard consiste à discrétiser l'espace et à appliquer un mécanisme Peer Prediction discret. Cette approche pose de nombreux problèmes : les

discrétisations grossières donnent lieu à des approximations inexactes de la distribution et à des incitations peu rigoureuses, tandis que les discrétisations fines donnent lieu à des paiements volatils, qui ont tendance à échouer dans les applications réelles. Nous développons la théorie du Peer Prediction, plutôt que de chercher une meilleure mise en œuvre de la théorie actuelle. Nous envisageons deux approches.

Dans la première approche, on peut discrétiser l'espace, ce que nous appelons le partitionnement en bacs, mais en choisissant parmi un ensemble de partitions plutôt qu'une seule. Dans ce régime, la notion d'appariement des pairs dans Peer Prediction est généralisée avec le concept de *Peer Neighborhoods*. Avec un renforcement raisonnable de la condition de mise à jour de l'agent, nous obtenons une extension valide du STP sur des distributions arbitraires. L'approche du partitionnement pour les distributions arbitraires révèle une théorie plus précise. En changeant de perspective et en passant d'un partitionnement en fonction de la mesure de Lebesgue sur l'espace des rapports à un partitionnement en fonction de la mesure de probabilité publique, nous obtenons une fonction de paiement qui ne repose pas sur la discrétisation. L'utilisation de cette fonction comme base d'un mécanisme, un *Continuous Truth Serum*, révèle des solutions à d'autres problèmes sous-jacents de la prédiction par les pairs, tels que le problème de la catégorie non observée.

Mots clefs : Théorie des jeux, mécanismes d'incitation, prédiction par les pairs, apprentissage automatique, évaluation des données, filtrage des données

Zusammenfassung

Da große, datengesteuerte Modelle der künstlichen Intelligenz allgegenwärtig werden, ist die Gewährleistung einer hohen Datenqualität für die Konstruktion von Modellen unerlässlich. Crowdsourcing, Community Sensing und Datenfilterung sind seit langem die Standardansätze zur Gewährleistung oder Verbesserung der Datenqualität. Die zugrunde liegende Theorie, vor allem die Gestaltung von Anreizmechanismen, ist in ihrem Anwendungsbereich oft begrenzt. Eine Untergruppe von Anreizmechanismen, die für den Umgang mit nicht verifizierbaren oder inhärent subjektiven Daten entwickelt wurde - Peer Prediction-Mechanismen - ist im Allgemeinen nur in Situationen anwendbar, in denen das Datensignal aus einer diskreten Verteilung stammt. In dieser Arbeit erweitern wir den Anwendungsbereich von Peer Prediction-Mechanismen in zwei Teilen.

Im ersten Teil behandeln wir eine eingeschränkte Erweiterung von Peer Prediction, die auf maschinelles Lernen anwendbar ist. Eine datenerfassende Instanz, ein sogenanntes Zentrum, muss möglicherweise keine gemeinsame Verteilung von (x, y) -Paaren lernen. Es muss lediglich ein parametrisiertes Modell lernen, das eine Verlustfunktion auf die gemeinsame Verteilung minimiert. Wir analysieren ein statistisches Maß, bekannt als *Influence*, das als eine Form von Peer Prediction interpretiert werden kann. Wir werden zeigen, dass Peer Truth Serum (PTS) ein Spezialfall von Influence ist und dass Influence wünschenswerte spieltheoretische Eigenschaften als Anreizmechanismus hat.

Anschließend führen wir die Analyse von Influence in den Bereich der Datenfilterung ein, der im Vergleich zum Crowdsourcing eine einzigartige Herausforderung darstellt. Wir verwenden asymptotische Analysen, um zu zeigen, dass die Fähigkeit, Trainingsdaten mit Influence zu filtern, im Grenzfall unendlicher Stichproben durch den Grad der Korruption in den Validierungsdaten eingeschränkt wird. Die Analyse endlicher Stichproben zeigt jedoch, dass man die Qualität der Validierungsdaten übertreffen kann, wenn die Bedingungen hinsichtlich höherer Momente der Datenmodelle erfüllt sind.

Im zweiten Teil gehen wir von dieser eher eingeschränkten Erweiterung zur allgemeinsten Erweiterung von Peer Prediction über: dem Lernen beliebiger Verteilungen. Viele Crowdsourcing-Probleme beinhalten absolut kontinuierliche Verteilungen, wie z. B. Gauß-Verteilungen. Der Standardansatz besteht darin, den Raum zu diskretisieren und einen diskreten Peer Prediction-Mechanismus anzuwenden. Dieser Ansatz ist mit zahlreichen Problemen behaftet: Grobe Diskretisierungen führen zu ungenauen Annäherungen an die Verteilung und lockeren An-

reizen, während feine Diskretisierungen zu volatilen Zahlungen führen, die in realen Anwendungen eher versagen. Wir erweitern die Theorie von Peer Prediction, anstatt eine bessere Implementierung der aktuellen Theorie zu suchen. Wir betrachten zwei Ansätze.

Beim ersten Ansatz kann man den Raum diskretisieren, was wir als Partitionierung in Bins bezeichnen, aber aus einer Reihe von Partitionen statt nur einer auswählen. In diesem Regime wird der Begriff des Peer Matching in Peer Prediction mit dem Konzept von *Peer Neighborhoods* verallgemeinert. Mit einer angemessenen Verstärkung der Agentenaktualisierungsbedingung erhalten wir eine gültige Erweiterung des PTS auf beliebige Verteilungen.

Der Partitionierungsansatz für beliebige Verteilungen offenbart eine präzisere Theorie. Indem wir die Perspektive von der Partitionierung nach dem Lebesgue-Maß auf dem Raum der Berichte zur Partitionierung nach dem öffentlichen Wahrscheinlichkeitsmaß ändern, erhalten wir eine Zahlungsfunktion, die nicht auf Diskretisierung angewiesen ist. Die Verwendung dieser Funktion als Grundlage für einen Mechanismus, einen *Continuous Truth Serum*, offenbart Lösungen für andere zugrundeliegende Probleme mit Peer Prediction, wie das Problem der unbeobachteten Kategorie.

Stichwörter: Spieltheorie, Anreizmechanismen, Vorhersage durch Gleichgestellte, maschinelles Lernen, Datenbewertung, Datenfilterung

Contents

Acknowledgements	i
Abstract (English/Français/Deutsch)	ii
List of figures	xi
List of tables	xiv
1 Introduction	1
1.1 Background	1
1.1.1 Game Theory and Mechanism Design	2
1.1.2 Peer Prediction	5
1.2 Contributions	6
1.2.1 Problem Statement	6
1.2.2 Influence Mechanisms	9
1.2.3 Influence Filtering	10
1.2.4 Peer Neighborhoods	11
1.2.5 Continuous Truth Serum	12
1.3 Related Work	13
1.3.1 Peer Prediction	13
1.3.2 Distributed Learning Metrics	16
1.3.3 Peer Consistency Generalizations	16
2 Influence Mechanisms	18
2.1 Introduction	18
2.1.1 Model	19
2.1.2 Influence	21
2.2 Influence-based Incentives	21
2.2.1 The Mechanism	21
2.2.2 Dominant Strategy Incentive-Compatibility	23
2.3 Incentives for the Center	27
2.3.1 Budgeting	27
2.3.2 Improved Equilibria	29
2.4 Relation with Peer Consistency	30

2.5	Practical Considerations	33
2.5.1	Influence Approximation	33
2.5.2	Sequential Data Gathering	35
2.5.3	M-Loss and M-Gain	35
2.6	Summary	39
3	Influence Filtering	40
3.1	Introduction	40
3.1.1	Our Approach	42
3.1.2	Model	43
3.1.3	Shapley value	44
3.2	Influence Analysis	45
3.2.1	Infinite Sample Analysis	46
3.2.2	Finite Sample Analysis	47
3.3	Filtering Schemes	51
3.3.1	Threshold Influence Filtering	52
3.3.2	Iterative Minimal Influence Filtering	53
3.3.3	Uniform Probabilistic Filtering	54
3.4	Empirical Analysis	55
3.4.1	Infinite Sample Regime	56
3.4.2	Finite Sample Regime	57
3.4.3	Incentives	58
3.4.4	Filtering	58
3.5	Summary	58
4	Peer Neighborhoods	60
4.1	Introduction	60
4.1.1	Approach	60
4.1.2	Model	62
4.2	Peer Neighborhood Mechanisms	63
4.2.1	Peer Consistency	63
4.2.2	Partition Spaces	64
4.2.3	The Mechanism Extension	66
4.2.4	Incentive-Compatibility	67
4.3	Analysis of Update Processes	68
4.3.1	Update Convergence	69
4.3.2	Satisfying the Update Conditions	72
4.3.3	Bin Edge Conditions	73
4.4	Simulations	77
4.4.1	Report Perturbation	78
4.4.2	Payment Stability	80
4.4.3	Distributions	82

5	Continuous Truth Serum	87
5.1	Introduction	87
5.1.1	Improving Peer Neighborhoods	87
5.1.2	Approach	88
5.1.3	Model	92
5.2	A Continuous Truth Serum	94
5.2.1	The Mechanism	94
5.2.2	Replicating the Peer Truth Serum	95
5.3	Incentive-Compatibility	98
5.3.1	The Ratio Measure	98
5.3.2	Report Optimization	101
5.3.3	Sufficient Maximizing Conditions	104
5.3.4	Additional Properties	106
5.4	Simulations	108
5.4.1	Report Perturbation	109
5.4.2	Tent Function Dependence	110
5.4.3	Fixed Discretization Payments	111
5.4.4	Distributions	112
6	Conclusion	116
6.1	Influence	116
6.1.1	Future Work	118
6.2	Peer Neighborhoods	119
6.2.1	Future Work	120
	Bibliography	122
	Curriculum Vitae	127

List of Figures

1.1	The Ex-Ante Peer Prediction Game Setting for an Agent.	7
1.2	The Ex-Interim Peer Prediction Game Setting for an Agent.	8
2.1	Empirically observed decrease of Influence on a typical regression model as more and more data is collected. Each batch corresponds to 100 data points. Both the exact Influence and the 2nd order approximation are shown.	27
2.2	The exact influence is shown to become computationally prohibitive for logistic regression with only a moderate number of data points, while the computation time for the approximate influence increases relatively slowly.	34
2.3	M-Loss is trained on all points in current batch, with Influence computed by removing a point. M-Gain is trained on all prior batches, with Influence computed by adding a point from current batch.	37
2.4	Ratio between Sum of Influences and Change in Loss with respect to batch size.	37
3.1	Mean Influences over number of data points. Growth rate of Influences matches $O(\frac{1}{N^2})$	41
3.2	Average over all regression datasets with LS corruption. Y values are the Q values that set the average Influence of accurate and corrupted points equal. Error bars are one standard deviation.	51
3.3	Heat map over all datasets with LS corruption. Coloration represents the difference between average Influence of accurate and corrupted data with $q = p \pm \epsilon$ for $\epsilon \in \{0.05, 0.1, 0.2\}$. More blue means more simulations with accurate data achieving higher Influence, more red means the opposite.	52
3.4	Crime dataset with AGN corruption. Noise mean 0. Noise variance ranges from 0 to 20. p and q are fixed at 0.75.	55
3.5	Normalized difference in average Influence aggregated over all regression datasets. p and q fixed at 0.75. Error bars are one standard deviation.	55
3.6	(a) Crime dataset with AGN corruption. Noise variance is 0. Noise mean ranges from 0 to 20. p and q fixed at 0.25. (b) Crime dataset with AGN corruption. Noise variance is 0. Noise mean ranges from 0 to 20. p and q fixed at 0.5. (c) Crime dataset with AGN corruption. Noise variance is 0. Noise mean ranges from 0 to 20. p and q fixed at 0.75.	55

3.7 Filter performance metrics averaged across all combinations of datasets with LS, XuYu, and XgmmYu corruption. q value is fixed at 0.8. (a) Change in p value. Error bars are $\frac{1}{2}$ standard deviation. A higher value is better. (b) Relative change in real loss, real loss being the loss measured only against the target distribution. Error bars are $\frac{1}{5}$ standard deviation. A lower value is better. 57

4.1 An example of a partition family on \mathbb{R}^2 with θ representing translations of the bins: $\beta_0(i)$ transforms into $\beta_\theta(i)$. Partition families are used to construct Peer Neighborhoods. 64

4.2 Expected payments for reports perturbed from the observation, computed over an Agent’s posterior. Error bars are one standard deviation. In the 2D figures, red lines show the location of the maximum expected payment. 78

4.3 Expected payments for reports perturbed from the observation, computed over truthful Peer reports. Error bars are one standard deviation. In the 2D figures, red lines show the location of the maximum expected payment. 79

4.4 Smaller bins produce a larger variance in payments. Error bars are one standard deviation squared. 80

4.5 True, Public, Kernel, and Posterior distributions for 1D Empirical distribution, Empirical update perturbation simulations. 82

4.6 True, Public, Kernel, and Posterior distributions for 2D Empirical distribution, Empirical update perturbation simulations. 83

4.7 True, Public, Kernel, and Posterior distributions for 1D GMM distribution, Pyramid update perturbation simulations. 84

4.8 True, Public, Kernel, and Posterior distributions for 2D GMM distribution, Pyramid update perturbation simulations. 84

4.9 True, Public, Kernel, and Posterior distributions for Empirical distribution, Empirical update bin size simulations. The Kernel and Posterior distributions are taken with the largest bin size. 85

4.10 True, Public, Kernel, and Posterior distributions for GMM distribution, Pyramid update bin size simulations. The Kernel and Posterior distributions are taken with the largest bin size. 86

5.1 Blue and light-blue represent bins with fixed probability measure $\frac{1}{n}$ in R , with the Gaussian density function $f_R(x)$ shown in orange. As the blue bins rotate around the circle according to the parameter θ transforming into the light-blue bins, they deform to maintain the $\frac{1}{n}$ probabilities. 89

5.2 Blue and light-blue represent bins with fixed probability measure in R . In the real domain, these categories transform into each other according to F_R , but in the quantile domain they transform with offsets. Taking the expectation over a uniform distribution of these offsets, which is equivalent to taking the expectation over R in the real domain, produces a payment taking the form of this *tent function* as in Equation 5.1. 91

5.3	For a categorical distribution, mapping the report to q in the middle of the left and right limits allows the tent function to be contained entirely inside the step interval with small enough b . The tent function integrates to 1, so integrating over $\mathcal{Q}_R^{-1}(r)$ lets the mechanism pick up the length of the step interval, reproducing the Peer Truth Serum.	96
5.4	Expected payments over deviation from truthful. Green plot taken over 100 fixed peer reports.	109
5.5	Expected payments over tent width b . Green plot taken over 100 fixed peer reports.	110
5.6	Expected payments over bin size for a fixed discretization. Plots averaged over 1000 observations from true distribution. This mechanism is only truthful up to the resolution of the bins.	111
5.7	Expected payments over deviation from truthful with fixed discretization payment. Green plot taken over 100 fixed peer reports.	112
5.8	True, Public, Kernel, and Posterior distributions for Report Perturbations	113
5.9	True, Public, Kernel, and Posterior distributions for Tent Function Dependence.	114
5.10	True and Public distributions for Fixed Discretization Payments.	114
5.11	True, Public, Kernel, and Posterior distributions for Report Perturbations with Fixed Discretization Payments.	115



List of Tables

3.1 Shapley value vs. Influence	45
---	----

1 Introduction

1.1 Background

Understanding the world around us requires gathering data and analyzing that data to recognize patterns, make forecasts, and come to conclusions about pressing questions. In 3800 BC, the Babylonian Empire took the first known census, in which they counted livestock and quantities of butter, honey, milk, wool, and vegetables. Presumably the Babylonian government did so to answer a number of questions: What is the economic state of the empire? How should certain resources be allocated? How should the tax code be optimized? Today, the quantity and specificity of the data that we analyze is on an unimaginably different scale. To handle synthesizing all this data, we've come up with tools such as statistics and machine learning. Data aggregation has proven itself useful. The aggregation of countless individuals making decisions, largely in their own self-interests, forms what Adam Smith referred to as the "invisible hand" of the free market economy. In James Surowiecki's "The Wisdom of Crowds," this idea is examined in more granular detail, showing that aggregate decision making is more accurate than most individual decision making (Surowiecki, 2005). The book opens with an anecdote about a crowd of people guessing the weight of an ox. The mean of all the guesses turns out to be more accurate than the majority of guesses.

But self-interested entities can act in ways detrimental to the goal of data gathering and analysis. In ancient Babylon, people would likely wish to hide their true quantity of livestock, butter, honey, and other goods in order to lower their tax burdens. Today, people may wish to hide data for similar reasons, for issues of privacy, or simply because acquiring and reporting data may take effort they don't wish to expend. The content of a website might not be worth the effort spent solving an onerous CAPTCHA. In many modern applications, especially machine learning, the problem of acquiring correct and useful data is critical. Machine learning models can be highly sensitive to the presence of inaccurate data.

1.1.1 Game Theory and Mechanism Design

The notion of analyzing the behavior of self-interested entities has been formalized in the field of Game Theory. And the application - asking the right questions to elicit the right data - is addressed in the sub-field *Incentive Mechanism design*. This thesis attempts to address some problems on the cutting edge of Incentive Mechanism design theory. In order to describe these problems, we must first cover some of the basic formalism. Game theory studies how rational self-interested entities, often called players or *Agents*, interact within the context of a game. A game is a mathematical construct which distributes *utility* to the Agents for playing *strategies* within a predetermined set of rules. Utility is an abstracted quantifiable reward, which can take the form of money, access, or any other object or service desired by an Agent. We provide the formal definition of a game according to Von Neumann and Morgenstern, 2007:

Definition 1.1.1 (Game). A *game* is a triplet $(\mathcal{A}, \mathcal{S}, u)$ where \mathcal{A} is a finite set of Agents $\{A_i\}_{i \in [1, n]}$, \mathcal{S} is a space of *strategies*, and $u: \mathcal{S}^n \rightarrow \mathbb{R}^n$ is a payoff function from a vector of strategies, one for each Agent, called a *strategy profile*, to a vector of utilities, one for each Agent, called an *outcome*.

Agent "rationality" is an assumption that the Agents will play strategies within a game which maximize their utilities. Game theory has proven itself to be a powerful tool in understanding large scale human systems, where many of the irrational idiosyncrasies of individual human behavior are averaged out. It has been applied successfully to fields including economics, voting systems, auctions, and prediction markets. The definition of a game is broad, as it places no restrictions on the complexity of the game's internal structure. It is often relevant to consider how an outcome might be computed from a strategy profile by setting limitations on this structure. Typically there is some causal structure involving decision points for the Agents, which we call game nodes, where Agents take actions according to their strategies. When an Agent makes a decision at a game node, the Agent can use any information that may be available at that particular game node. For example, an Agent might get to observe the sequences of actions made by the other Agents at preceding game nodes. We call the information used to determine an action at a game node the *information set*. In this context, a strategy is realized as a mapping from a game node and information set to an action in a predetermined action space. A strategy is called *pure* if it deterministically maps a game node and information set to a single action. A strategy is called *mixed* if it produces random variables over the space of possible actions. Under mixed strategies, the payoff function u would be computed by taking expectations over the random variables produced by the strategies. In this work, we generally assume that Agents can adopt mixed strategies unless specified otherwise.

The payoff function is defined to be deterministic given the strategy profile. In a round of poker, if we interpret each player's winnings as the payoff, then the game definition requires that the card ordering is determined. Alternatively, we may want to consider a round of poker

where the card ordering is unknown. This would result in randomness in the sequence of game nodes, subsequently leading to randomness in the actual winnings of the players. In this case, the payoff function is computed by taking an expectation over the actual winnings. Such games are often referred to as *Bayesian* games, because Agents have incomplete information about the game structure.

The game theorist is often concerned with identifying properties of games called equilibria. The mechanism designer, on the other hand, is less concerned with analyzing games than with constructing them. We say a mechanism designer wishes to find a utility function such that a particular *strategy profile* satisfies a particular *solution concept* under the assumption of Agent rationality. In other words, the highest payoff strategies should satisfy certain conditions:

Definition 1.1.2 (Solution Concept). A *solution concept* is a predicate on a strategy profile in a game that is consistent with rationality under certain conditions.

The best examples of such solution concepts are *equilibrium concepts*:

Definition 1.1.3 (Dominant Strategy Equilibrium). A *Dominant Strategy Equilibrium* (DSE) is a solution concept which is satisfied by a strategy profile if and only if, for each strategy in the strategy profile, that strategy has the highest payoff for that Agent regardless of the strategies of the other Agents.

Definition 1.1.4 (Bayes-Nash Equilibrium). A *Bayes-Nash Equilibrium* is a solution concept which is satisfied by a strategy profile if and only if, for each strategy in the strategy profile, that strategy is the highest payoff given all the other strategies in the profile.

Such equilibria can also be *strict* if the strategy for each Agent is uniquely highest payoff under the assumed conditions. There are many other solution concepts considered in the field of game theory, such as *Subgame-Perfect Nash Equilibria*, an important concept in economics, especially concerning Stackelberg leadership models (Von Stackelberg, 2010). As we will see, the DSE and BNE solution concepts take on special significance in mechanism design.

The mechanism designer, often referred to as the principle or the *Center*, must have some motive for constructing a game. The motive of the Center must depend on properties of the Agents, otherwise there is no point in involving the Agents. Furthermore, it must depend on hidden properties of the Agents, otherwise the Center doesn't need to construct a game, it can just perform a computation directly on the known properties. So, in a mechanism design problem, each Agent has access to its own private information as part of its information set for at least some game nodes. This private information might be intrinsic to the Agent, such as private medical data. It might be derived from nature, for example, the Agent might be performing some measurement which the Center is not capable of performing itself. We then say that the private information determines the Agent's *type*. The *type* is a formal construct typically denoted by θ in some space of types Θ . We say a set of Agents participating in a game constructed by the Center has a *type profile* $\hat{\theta}$.

The motive of the Center is to produce some outcome which depends on the type profile, formally defined as the *social choice function* f , which maps a type profile to a game payoff outcome. A straightforward example of a social choice function is one which maximizes the social utility of the outcome - the sum of utilities over all the Agents - in a resource allocation game. Suppose each Agent has private individualized utilities assigned to each resource in a set of finite resources. These private utilities constitute each Agent's type. The Center wishes to allocate the resources in order to maximize the social utility. The social choice function would then take the Agents' types and assign each resource to the Agent who attains the highest individual utility for that resource. We say the reason to construct a game is to *implement* the social choice function:

Definition 1.1.5 (Implementing a Social Choice Function). A game with payoff function u implements a social choice function f over a solution concept if and only if, given a type profile $\hat{\theta}$, for any strategy profile \hat{s} which satisfies the solution concept, $u(\hat{s}) = f(\hat{\theta})$.

In our example, the Center might construct a game which asks the Agents to reveal their private utilities. But Agents receive higher utilities by acquiring more resources, so they may want to lie about their private utilities to acquire more resources. The Center then wants to construct the game such that it would be irrational to lie about private utilities under some assumed solution concept. The social choice function need not be concerned with the utilities distributed to the Agents per se, it might maximize some value for the Center, such as profit. Finally, we can state the formal goal of mechanism design:

Definition 1.1.6 (Problems of Mechanism Design). A *problem of mechanism design* involves a Center which chooses a solution concept and a social choice function f . The Center then constructs a game with payoff function u , the *mechanism*, where the Agents take actions which involve reporting a type to the Center. The problem for the Center is to construct u so that the game *implements* f over the solution concept.

The task of the Center can be quite difficult. It's conceivable that there are payoff functions that implement the social choice function without revealing the Agents' types. The Center might have to explore the space of all possible games in order to find such a payoff function. The Center's job would be significantly easier if the strategy profiles which satisfy the solution concept actually do reveal the Agents' true types:

Definition 1.1.7 (Direct Mechanisms). A mechanism is *direct* over a solution concept if and only if the strategy profile in which each Agent reports its true type satisfies the solution concept.

We now introduce a concept that will be the focus of much of this work:

Definition 1.1.8 (Incentive-Compatibility). A mechanism is *incentive-compatible* if the mechanism directly implements a social choice function over either the DSE solution concept or the BNE solution concept. Respectively, we say that a mechanism is either *Dominant-Strategy Incentive-Compatible* (DSIC) or *Bayesian-Nash Incentive-Compatible* (BNIC).

A powerful result known as the *revelation principle* undergirds the field of incentive-compatible mechanism design. Since its discovery, it has been expanded upon and stated in the most general terms. We state it formally:

Theorem 1.1.9. *If there exists a mechanism which implements a social choice function over the DSE or BNE solution concepts, then there exists a DSIC or BNIC mechanism, respectively, which implements the social choice function.*

The revelation principle for DSE mechanisms was discovered by Gibbard, 1973. The principle for BNE mechanisms was later discovered by P. Dasgupta et al., 1979, Holmström, 1978, and Myerson, 1979. The consequence of this principle is that, if the DSE or BNE solution concepts are desirable, the Center only needs to concern itself with discovering direct mechanisms, since if an indirect mechanism exists, a direct one must also exist. We often refer to such mechanisms as *truthful*, although it means the same thing as incentive-compatible.

1.1.2 Peer Prediction

It is intuitive that when the Center has some method to perform baseline evaluations of the Agent reports, such as a set of known correct reports for comparison, this can make it much easier to construct incentive-compatible mechanisms. But there are many reasons why a Center may not have access to such a baseline. The data may be held privately by Agents, such as personal medical data. The data might require Agents taking active roles in completing some tasks with unknown solutions, Or the data might be inherently subjective, such as product ratings. In such a setting, the Center can only compare Agent reports to the reports of other Agents, known as Peer reports. Mechanisms that operate in this setting are called *Peer Prediction* mechanisms. A common feature of a Peer Prediction setting is that the Agents must incur some *cost of effort* to observe or report their own type. This could be because the data is sensitive and they wish to keep it private, such as personal medical data. Or acquiring the data might require the Agent to expend time and mental effort, such as for completing tasks. Either way, this cost of effort must be adequately compensated for by the mechanism. Suppose the cost of effort is less than \$1. A naive Center might offer \$1 for a data report in order to overcome this cost of effort, but a rational, strategic Agent could then report some random or fixed data which has no relation to the correct data, eschewing the cost of effort but receiving the \$1 reward. Such a strategy is referred to as a *heuristic strategy*. Sometimes the Agents or the Center are assumed to have some prior knowledge about the game setting, such as a prior guess about the distribution of Agent types. These are generally referred to as *beliefs*.

Generally, a Peer Prediction setting and mechanism are designed to cover a set of considerations for real world applicability. We list these considerations, similar to those found in Faltings, n.d.:

- **Prior Details:** How much knowledge do the Agents or Center require about the setting and what are the structures of beliefs? No prior assumptions is known as *detail-free* and

is most general.

- Dimensionality: Are Agents observing a univariate signal (single-task) or multivariate (multi-task) signal? Univariate signals are more general.
- Solution Concept: Is the mechanism DSIC or BNIC? If BNIC, is the truthful equilibrium highest payoff? Are there equilibria where Agents reveal no information about types to the Center (*uninformed equilibria*)? DSIC is the most desirable solution concept.
- Finite Agents: Is the mechanism incentive-compatible with a finite set of Agents?
- Minimalism: How much information do Agents need to report? If Agents only report a type, the mechanism is minimal and most practical in this respect.
- Generality: Can the mechanism be applied to arbitrary signal distributions?

1.2 Contributions

1.2.1 Problem Statement

The problem this thesis seeks to address follows a track of progress on Peer Prediction mechanisms. We consider a class of minimal mechanisms which are in general univariate and are incentive-compatible with finite agents. The prior details that we consider range in their generality, but they tend to follow the track of *Peer Consistency mechanisms*. Peer Consistency mechanisms are a sub-class of Peer Prediction mechanisms with these same considerations. This choice of considerations makes it difficult for a mechanism to perform any comparison between reports other than matching. For this reason, most Peer Consistency mechanisms operate only on finite discrete distributions. The prior details for most Peer Consistency mechanisms involve a shared prior belief between the Center and the Agents about the distribution of Peer reports, but allow for heterogeneous belief updates among the Agents. Peer Consistency mechanisms generally describe a payment function that admits a particular class of belief updates. The conditions for being in this class are known as *update conditions*.

The primary goal of the thesis is to construct and analyze mechanisms with these considerations, but extend the applicability to a broader set of distributions than simply discrete. The ultimate goal is to extend Peer Consistency mechanisms to arbitrary distributions, and describe the update conditions which are admitted. Because of the generality of the setting for Peer Consistency, it is quite difficult for such a mechanism to be DSIC, so we focus on BNIC mechanisms with some allowance for DSIC in special circumstances.

Problem Setting

Throughout this thesis, we assume a standard crowdsourcing setting for a Peer Prediction mechanism. There is a *Center* that wishes to implement a social choice function. In this work,

we do not explicitly state the social choice function, we assume that the Center wishes to accurately model a distribution of signals received by a set of *Agents*. The Center establishes the game in which Agents will strategize and submit reports, subsequently receiving a payoff according to the mechanism set by the Center. In the first part of this thesis, we consider that the Center is modeling a mapping between variables in the signal, corresponding to a standard supervised machine learning setting. In the second part, we consider the more general problem of a Center learning the distribution directly by collecting independent samples. We assume that the Center is crowdsourcing because it is either impossible or prohibitively expensive for it to sample the signal distribution directly. The Agents, however, can sample the signal distribution by expending *effort*, which manifests as a negative utility. These negative utilities can be heterogeneous with respect to the Agents, and may depend on the particular observed signal. In order for the Center's incentive mechanism to be incentive-compatible, it must overcome this cost of effort.

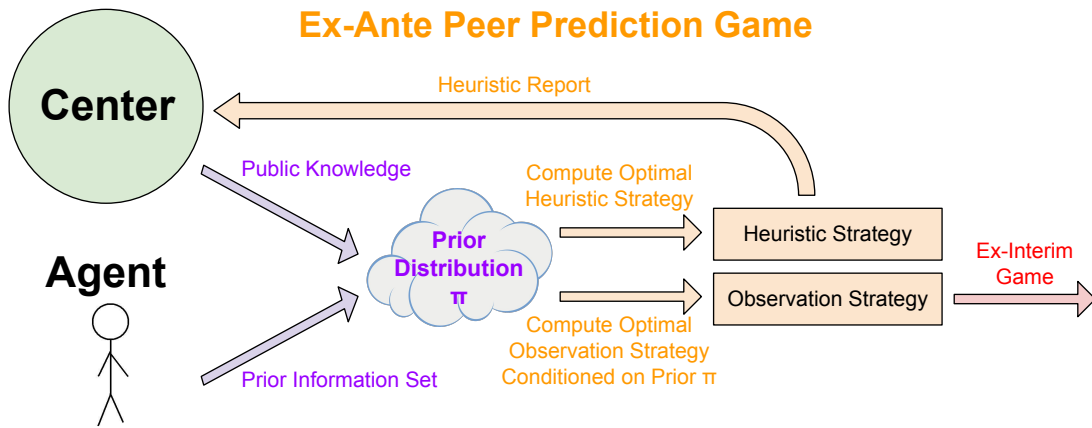


Figure 1.1: The Ex-Ante Peer Prediction Game Setting for an Agent.

Prior to observing a signal, we say an Agent plays the *ex-ante* game, as shown in Figure 1.1:

Definition 1.2.1 (Ex-ante Game). The *ex-ante* game for Peer Prediction is a game in which an Agent with a prior information set, which it uses to construct a *prior belief* about the signal distribution, chooses either to play an *uninformed heuristic strategy*, or to play an *observation strategy*, receiving an expected payoff conditioned on the prior belief.

Definition 1.2.2 (Uninformed Heuristic Strategy). An *uninformed heuristic strategy* is one in which the Agent chooses a report purely on his prior belief, i.e. without making an observation by sampling the signal distribution.

Definition 1.2.3 (Observation Strategy). An *observation strategy* is one in which the Agent expends effort to sample the signal distribution, then makes an report based on his updated information set.

In the *ex-ante* game, the Agent doesn't know what effort it will expend until it makes the observation, so it computes the expected effort based on its prior belief. It also computes

the expected reward for the observation strategy as the expectation over the rewards for the optimal strategy on a new information set, conditioned on the prior belief as the distribution of samples which get added to the information set.

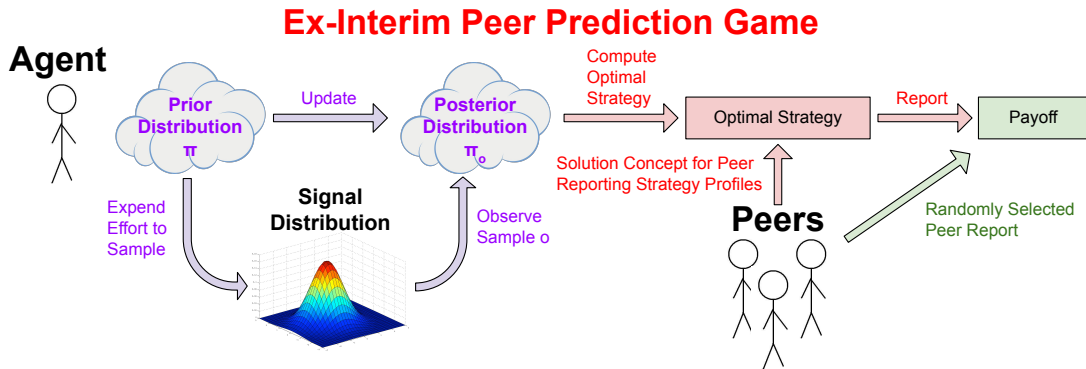


Figure 1.2: The Ex-Interim Peer Prediction Game Setting for an Agent.

Assuming the Agent does not already know the signal distribution and consequently must expend effort to sample it, the first requirement for a Peer Prediction mechanism is that an ex-ante observation strategy has a higher expected payoff than any ex-ante heuristic strategy, even taking into account the negative utility from the expended effort. A truthful strategy is an example of an observation strategy, but an Agent could also apply some data processing after the observation in order to come up with a higher payoff, non-truthful report. After playing the ex-ante game to determine if the Agent will make an observation, we say the Agent then plays the *ex-interim* game where it decides what signal to report to the Center, as shown in Figure 1.2:

Definition 1.2.4 (Ex-Interim Game). The *ex-interim* game for Peer Prediction is a game in which an Agent with an updated information set chooses to report the point with the highest expected reward, conditioned on his updated information set and some solution concept which determines the possible strategies of the Peers.

If the Agent plays an ex-ante heuristic strategy, the ex-interim game is the same as the ex-ante game because the Agent has not updated its information set. If the Agent plays an ex-ante observation strategy, the information set now contains the observation, and the effort has been expended. We often refer to the information set in the ex-interim game as the *posterior beliefs*. With this structure, a mechanism will be incentive-compatible if truthfulness is a highest expected payoff ex-interim observation strategy, either as a Dominant Strategy or conditioned on the Peers playing a truthful strategy, and it is higher expected payoff than any ex-ante heuristic strategy. This decoupling of the different strategy types makes it easier to prove incentive-compatibility, and we make use of it throughout the thesis.

One of the most straightforward models of prior and posterior beliefs is for the Agents to estimate the distribution of signals. When an Agent plays an observation strategy, it performs

a *belief update* from its prior to its posterior, conditioned on the observed signal. The problem can then be posed purely according to priors and belief updates. We will further simplify this by considering *uninformed* Agents:

Definition 1.2.5 (Uninformed). An *uninformed* Agent is one who constructs his information set exclusively from information made publicly available by the Center.

If all Agents are uninformed, they will construct homogeneous prior beliefs, so the problem of incentive-compatibility comes down to the belief updates. It is clear that totally unconstrained belief updates will make the problem impossible, as there will be no information to exploit from Agents' posteriors. To constrain the problem, the mechanism designer will propose, along with the mechanism, a set of belief update conditions:

Definition 1.2.6 (Belief Update). A *belief update* is a predicate on a posterior belief, conditioned on a prior belief and a given observation.

The mechanism designer will then prove incentive-compatibility with respect to a set of belief update conditions. Much of the preceding work on Peer Consistency first presents the belief update conditions, justifies their "reasonableness" for practical settings, then defines the mechanism and proves its incentive-compatibility. In some cases, it is also possible to prove the uniqueness of the particular incentive-compatible mechanism with respect to the update conditions, as is the case for the Peer Truth Serum with respect to the Self-Predicting update condition (Faltings et al., 2014). In much of this thesis, we will work backwards. The structure of analysis we use for assessing Peer Prediction mechanisms is generally as follows:

- Present a Mechanism.
- Compute the expected payoff for an Agent conditioned on its prior and posterior beliefs.
- Analyze the expected payoff to determine what conditions produce truthfulness as a maximum payoff strategy.
- Assess the reasonableness of the update conditions in practical contexts.

1.2.2 Influence Mechanisms

Many modern crowdsourcing problems are related to creating supervised learning models. In these settings, the Center is not concerned with learning the underlying data distribution, which is some joint distribution of *inputs* and *labels* (x, y) . The Center is concerned with learning a mapping between the inputs and labels that minimizes some loss function over the distribution, where the loss function is chosen a priori by the Center in order to achieve some predetermined goal. This imposes additional structure for the incentive mechanism to exploit. A new input and label sample will be used by the Center to adjust the model to minimize the loss function. This adjustment will affect the loss function evaluated on all the

remaining samples. In this way, the Center is automatically provided a relation between an Agent report and a Peer report to use for a Peer Consistency mechanism.

A natural approach to understanding this relation comes from statistics and is known as marginal *Influence* (Cook and Weisberg, 1980). Influence quantifies the "value" of an input-label sample in helping a model predict other input-label samples, but this value is contingent on the other samples used for model optimization, which we call training samples. The Influence of a training sample on a target sample is the difference in loss on the target sample produced by the optimal model with and without the training sample. A practical use for Influence is explaining unexpected behavior of large learning models. Cook and Weisberg, 1982 and Koh and Liang, 2017 demonstrate how Influence can be used to debug "black-box" models by quantitatively identifying which training samples are most responsible for specific predictions. As Influence is defined, it requires retraining the model for every training point. Koh and Liang, 2017 also demonstrate that Influence can be approximated efficiently, and that the approximation has meaningful explanatory power even when the optimal model is not learned.

In studying Influence for Peer Prediction, we are only concerned with approximations from a practical implementation standpoint. We first propose an Influence-based mechanism, then demonstrate update conditions for which the mechanism is incentive-compatible. We show that under some strict assumptions and mechanism implementations, the mechanism can be DSIC, but in general it is BNIC. We theoretically demonstrate that it is a generalization of other Peer Consistency mechanisms by showing that the PTS mechanism is a special case of Influence.

We then move on to some practical considerations for the Center. We consider a regime in which the Center itself is a player in the game and receives utility based on the loss of the model. Under some very realistic assumptions, such as bounded utility, the Center can tune the mechanism to optimize its own utility with respect to Agent effort levels within a bounded budget. We also show that the approximation suggested in Koh and Liang, 2017 is theoretically insufficient for incentive mechanisms, and propose an alternative approximation using higher order terms in the Taylor expansion of the loss function. While the higher order approximation is more computationally expensive than the linear approximation, it is still incomparably more efficient than retraining large models. In addition, we show how batch processing of samples can mitigate the computational expense while maintaining reasonable budget approximations.

1.2.3 Influence Filtering

The efficacy of Influence depends on having a clean set of validation samples, where cleanliness means that the validation samples are independently drawn from the underlying distribution of interest. In incentive mechanism design, this is either accomplished by assumption, which yields a DSIC mechanism, or it's a consequence of a truthful Bayes-Nash Equilibrium

when validation samples are taken from Agent reports. In real world applications, there are many circumstances that can damage the Bayes-Nash Equilibrium. Some Agents may have an interest in the outcome of the model, they may have an interest in maintaining a degree of privacy by adding noise to the reports, or they may simply misunderstand the incentives. For this reason it is important to consider the robustness of the truthful incentives. We refer to this as the *filtering* problem. Even when an incentive mechanism is properly implemented, the data the Center elicits may not all come from the distribution of interest. We call the data from this distribution *accurate*, and other data is *corrupted*. So the goal of the filtering problem is to remove as much corrupted data as possible while keeping as much accurate data as possible.

We use two methods of analysis to understand the incentives in a regime when there is corrupted data. The first method examines the infinite sample regime, where the model converges to the optimal model. We show that, given any proportion of corruption ($1 - q$) in the validation sample set, there is a mixed DSE of reporting truthfully with probability $p = q$ and reporting from the corrupt samples with probability $(1 - p)$. Furthermore, these DSE are stable in the sense that any Agent who believes $p < q$ has a DSE of truthful reporting and, symmetrically, any Agent who believes $p > q$ has a DSE of corrupt reporting. In both cases, the DSE would drive the training sample set towards the $p = q$ mixed DSE. We then apply a second analytical method for finite samples by assuming that the optimal models are Gaussian distributed over the distributions of finite training sample sets. We call these the model posteriors. We see then that the optimal model with finite samples has some expected deviation from the optimal model for infinite samples. The presence of an accurate or corrupt sample in the training set then has an effect on this deviation, which corresponds to an expected Influence value. We obtain an inequality, which depends on the variances of the model posteriors, that determines when an accurate sample has higher expected Influence than a corrupt sample. This can modify the $p = q$ equilibria. The infinite sample analysis unfortunately suggests that there are mixed BNE which are potentially higher payoff than the truthful BNE, but the finite sample analysis demonstrates that this requires a degree of coordination on the part of the Agents. Using this analysis, we then suggest a probabilistic Influence-based filtering algorithm and compare it to more natural deterministic Influence-based algorithms. We show that it achieves similar performance to the best deterministic Influence-based filtering at much lower computational cost.

1.2.4 Peer Neighborhoods

We move away from Influence to examine Peer Prediction generalizations with fewer prior assumptions about the structure of the problem. In fact, we assume no prior assumptions about the Center's goal in collecting data, aside from learning the distribution. Prior work on Peer Consistency presented Agent belief update conditions, justifying their real world applicability, then constructed truthful mechanisms around those update conditions. But the update conditions, such as self-dominating or self-predicting, are not well defined for arbitrary distributions and do not have an obvious natural extension. Our work proposes a

framework for extending Peer Consistency mechanisms, called Peer Neighborhoods, which subsequently induce an extension of the update conditions. Peer Neighborhood mechanisms are constructed by generating *partitions* of the sample space, and then considering a distribution over the partitions. By constructing this partition family following a few basic rules, we show that the Peer Neighborhood extension is BNIC.

The claim of incentive-compatibility for Peer Neighborhood extensions is with respect to the extended update conditions. We address the reasonableness of these conditions by concretely extending the Peer Truth Serum and analyzing the extension of the Self-Predicting update condition (Faltings et al., 2014). Intuitively, an update condition which references the probability of an observed sample cannot function when the distribution is a continuous random variable, since single samples have probability zero. Therefore, any update condition operating on arbitrary distributions must consider neighborhoods around the observed sample. The partitions in the Peer Neighborhood framework naturally construct these neighborhoods, allowing for a well-defined update condition with reasonable constraints. Although such update conditions cannot be justified by a simple application of Bayes' Rule, as is the case for the Self-Predicting condition, we show that the update condition corresponding to the Peer Truth Serum Neighborhood extension admits a broad class of updates. We propose a method for computing a set of such updates, and provide simulations demonstrating the strength and stability of the incentives.

1.2.5 Continuous Truth Serum

In the final section of this thesis, we show that there is a natural choice for how to construct the partitions for Peer Neighborhood extensions. When constructing a family of partitions and picking a distribution over that family, a notion of distance emerges, corresponding to the probability that two points are in the same bin of a partition. This distance can be constructed somewhat arbitrarily depending on the choice of partition family and distribution. But a natural choice exists, the shared prior already suggests a notion of distance between two points: the prior probability between the points. We show that this is a special case of the Peer Neighborhood extension where the partition family is the set of all partitions of connected bins with equal probability. Analyzing this extension becomes much simpler when considering the partitions mapped into the domain of the cumulative distribution function (CDF) of the shared prior. In this domain, the bins become uniform width and the distribution over the partitions is uniform. We find that this partition structure yields a payment which takes the form of a smooth, symmetric function, which we call the *tent function*.

We analyze this new mechanism, which we call the *Continuous Truth Serum*, to discover the update conditions which make the mechanism BNIC. We find that it is difficult to state necessary and sufficient update conditions in a simpler form than the trivial statement: "The expected payment with respect to the Agent's posterior is maximized at the observed sample." But we show that there are reasonable sufficient conditions that admit a broad class

of updates, which correspond to a symmetric subset of updates admitted by the more general Peer Neighborhood framework. The fact that these conditions are sufficient but not necessary means that the true class of updates is even broader. We justify the reasonableness of the update conditions as expressions of a necessary locality structure for any update condition operating on arbitrary distributions. Finally, we again provide simulations demonstrating the strength and stability of the incentives.

1.3 Related Work

1.3.1 Peer Prediction

Most of the work in this thesis adds to the field of Incentive Mechanism design, and more specifically the sub-field of Peer Prediction mechanisms, which operate in the absence of any baseline metrics the Center can use to evaluate the Agents' reports. Some of the first Peer Prediction mechanisms came about from addressing a tangential problem: reputation systems, as in Jurca and Faltings, 2003. The setting involves Agents playing an iterated prisoner's dilemma game where sequences of actions determine an Agent's reputation. They present a mechanism which pays a constant when an Agent's report matches with a randomly selected Peer's report. This simple mechanism became known as *Output Agreement*, and has been studied extensively. Von Ahn and Dabbish, 2004 demonstrate how the Output Agreement can be effectively utilized for an image labelling game.

The term "Peer Prediction" would not be coined until the seminal work of N. Miller et al., 2005. Using the mechanism design considerations listed in the introduction, N. Miller et al., 2005 propose a minimal mechanism that is strictly BNIC with prior details being that there is a shared, static prior belief about the reports of Peers that is known to the Center. It can operate on univariate signals with finite Agents, and can even accommodate continuous signals. The mechanism concept draws from previous work in economics, showing that *proper scoring rules* can be used to construct incentive-compatible mechanisms (Gneiting and Raftery, 2007). Other work has shown proper scoring rules to be useful for evaluating prediction markets (Hanson, 2007). The primary limitations of this mechanism are that the truthful BNE is not necessarily highest payoff, and the prior detail assumptions are very strong and therefore don't correspond to many real world settings.

There was clear room for improvement, and the history of Peer Prediction mechanism design quickly branched into multiple tracks. Much work focused on the track of single-task mechanisms while trying to reduce the prior details towards detail-free, and working to improve on the solution concept. The Disagreement Mechanism eliminates the need for the shared prior to be known to the Center, but it is still static (Kong and Schoenebeck, 2016). The solution concept is also refined such that the truthful BNE is at least as high payoff as a set of undesirable "relabeling strategies". The Shadow Peer Prediction mechanism improves on this by allowing for private priors and satisfies a highest payoff BNE solution concept (Witkowski and Parkes,

2012). However, this more general mechanism is achieved at the cost of minimalism - Agents are required to report before and after observing a signal.

Non-Minimal Mechanisms

One track in Peer Prediction development explores the power of non-minimalism demonstrated by the Shadow Peer Prediction mechanism. Along this line, d'Aspremont and Gérard-Varet, 1979 and McAfee and Reny, 1992 describe mechanisms to elicit private, correlated information in an economics setting. d'Aspremont and Gérard-Varet, 1979 mostly explore how incentive mechanisms affect budgetary concerns for the Center, but in the process they discover that by having Agents report their beliefs about the Peer reports, the Center can constrain the budget for the incentive mechanism. McAfee and Reny, 1992 survey auction mechanisms and show that bargaining mechanisms can generally handle the problem of private information. The bargaining process is an effective example of non-minimalism.

The work of d'Aspremont and Gérard-Varet, 1979 was expanded on by Prelec, 2004, who proposed the Bayesian Truth Serum (BTS). The BTS is a detail-free, strictly highest payoff BNIC mechanism which requires Agents to report both a type and a belief about the distribution of Peer signals. The primary downsides, aside from non-minimalism, are that the signals are assumed to be binary, and the mechanism is only incentive-compatible in the limit of infinite Agents. Radanovic and Faltings, 2013 extend the BTS to non-binary signals, and later Radanovic and Faltings, 2014 extend this further to continuous signals, but still require infinite Agents. Witkowski and Parkes, 2012 propose a similar mechanism which is incentive-compatible for non-binary discrete signals with finite Agents. In general, we refer to mechanisms which require both type reports and belief reports as *BTS-like*. They have also been analyzed extensively in practice and shown to be effective in real world settings (Frank et al., 2017; Loughran et al., 2014; S. R. Miller et al., 2014; Weaver and Prelec, 2013).

Multi-task Mechanisms

Another track in Peer Prediction explores multi-task mechanisms rather than non-minimal mechanisms. A. Dasgupta and Ghosh, 2013 construct a mechanism with a highest payoff truthful BNE for binary signals from multiple tasks. The mechanism is not detail-free, but it assumes a broad setting in which Agents can exert up to some maximum effort to make a noisy observation, with the noise scaled inversely with the effort. This mechanism is later expanded upon by Shnayder et al., 2016 to form the Correlated Agreement (CA) mechanism, which operates on multiple tasks with discrete signals, rather than binary. Furthermore, they propose a detail-free version of Correlated Agreement, but it is only incentive-compatible in the limit of infinite tasks. Kong and Schoenebeck, 2019 show that the CA mechanism has a truthful DSE in the infinite task limit. They do so by analyzing the mechanism from an information-theoretic perspective. They introducing the concept of information-theoretic mechanisms, which function broadly by examining the mutual information between an

Agent's report and Peer reports, and improve on the CA mechanism with the Determinant based Mutual Information mechanism, which achieves this truthful DSE with finite tasks.

The common thread between the BTS-like and multi-task mechanisms is that Agents provide reports beyond a single signal. Although this provides a lot of power for constructing incentive-compatible mechanisms in very general settings, there is concern about the practicality of these mechanisms. This is especially the case for BTS-like variants which require a report containing a description of an Agent's belief about the distribution of Peer reports. This makes it difficult to extend BTS-like mechanisms to arbitrary non-discrete distributions because belief reports require the distribution to be finitely parameterizable.

Peer Consistency

Another track, which is the track this thesis follows, explores minimal, single-task mechanisms with Agent belief settings which may not be detail-free, but apply broadly to real world applications. They generally work by providing tuned payments when an Agent report matches with a Peer report. Such mechanisms are commonly referred to as *Peer Consistency* (Huang and Fu, 2013). Most generally, a Peer Consistency mechanism describes a mechanism in which Agent types are correlated in some way, and the quality of the reports can be measured by how they follow this correlation. The canonical example is simply the Output Agreement mechanism from earlier, but this idea is extended to the Peer Truth Serum (PTS) mechanism (Faltings et al., 2014). The key insight is to notice that Agent beliefs do not necessarily need to be known to the Center, as they are in the seminal work of N. Miller et al., 2005. Rather, only some limited structure needs to be known. This is typically characterized by *belief update conditions*. In a Peer Consistency setting, the Agents may share some prior belief known to the Center, but after observing signals, the Agents will update to *posterior beliefs*. These updates can be heterogeneous, but they must follow certain conditions. The strictness of these conditions determine the relative practicality of the mechanism.

The Output Agreement can be described as the incentive-compatible Peer Consistency mechanism corresponding to the *self-dominating* update condition, which states that the observed signal must have the maximal probability in the posterior belief. Waggoner and Chen, 2014 similarly show how Output Agreement can be characterized as eliciting "common knowledge", rather than as an incentive-compatible mechanism in certain settings. The self-dominating update condition is too strong for most practical applications, because an Agent may not consider an rare signal to be common simply because they observed it once. The Peer Truth Serum is the incentive-compatible Peer Consistency mechanism corresponding to the *self-predicting* update condition, which states that the observed signal should have the greatest increase in log-likelihood from the prior to the posterior (Faltings et al., 2017). This condition admits a much broader class of updates than self-dominating, and satisfies Bayes' Rule.

1.3.2 Distributed Learning Metrics

In the second and third chapters of this thesis we consider the use of Influence as an evaluation metric for the purpose of incentive mechanisms and data filtering (Cook and Weisberg, 1980). We choose Influence because of its applicability to crowdsourcing data for a Center looking to construct a machine learning model, especially a federated learning model, which is trained in a distributed manner among independent Agents. Soltani et al., 2023 survey evaluation metrics for this purpose. The proposals include evaluations based on the amount of data contributed by an Agent (Feng et al., 2019). This work proposes a mechanism for negotiating data contributions based on effort levels, but assumes truthfulness. Zhang et al., 2021 propose a centralized evaluation framework when the Center has access to validation data, while Che et al., 2022 propose a similar evaluation framework in a decentralized setting. Both proposals, however, require an iterated reputation-based mechanism.

Many evaluation metric proposals touch on concepts closely related to Influence. A number of examples consider examining group Influences and clustering Agents accordingly (Chai et al., 2020; Lai et al., 2021; Wu and Wang, 2022). Other proposals consider the *Shapley Value* of data contributions. The Shapley Value is a concept from game theory which considers the utilities in cooperative games of Agents in all possible coalitions (Shapley et al., 1953). But this concept requires defining a utility function. In the examples of Jia et al., 2019 and Ghorbani and Zou, 2019 this utility function is the loss of the model. Under this formulation the Shapley Value actually becomes the average Influence over all orderings of the data. We show that it is not necessary to undergo the computational expense of computing Shapley Values for the purpose of incentive mechanisms. Another proposal considers evaluating contributions based on the change in model parameters (Zhao et al., 2022). This is also closely related to Influence, as it is an element of the Taylor expansion-based approximation method laid out in Koh and Liang, 2017. We focus on Influence because of its close relationship to a number of these concepts, and we show that it is also closely related to previous work in Peer Consistency by reducing it to the Peer Truth Serum mechanism under specific constraints.

There are other metrics for data cleaning in contexts other than federated learning, such as in Rahm, Do, et al., 2000, Dasu and Loh, 2012, or more recently in Ilyas and Chu, 2019. We see that most examples of scoring functions are highly context-dependent, such as Language Model Cross-Entropy in Mansour et al., 2011, DFIRE in Soto et al., 2008, or noise scoring in Luengo et al., 2018. We further note that the analysis of such scoring functions all assume that the evaluation of the score itself is reliable. We relax this assumption in our work.

1.3.3 Peer Consistency Generalizations

In the fourth and fifth chapters of this theses we consider more general extensions of Peer Consistency mechanisms to accommodate settings where the underlying signals can have arbitrary distributions. A primary disadvantage of Peer Consistency mechanisms is the ability to accommodate non-discrete distributions, because they generally rely on the notion of

an exact match between two independent reports. If the underlying signal is a continuous random variable, the probability of an exact match is 0. The attempts at generalization tend to rely on additional prior details about Agent belief structures. The Logarithmic Peer Truth Serum achieves the goal of extending the PTS to arbitrary distributions, with truthful reporting as the highest payoff BNE (Radanovic and Faltings, 2015a, 2015b). But it assumes that Agents can be grouped based on some latent variable which represents a locality structure, with the relationship between the latent variables and the underlying distribution known to the Center. Agents from different groups are assumed to observe independent samples conditioned on the latent variables. Goel and Faltings, 2020 build on this work with the Personalized Peer Truth Serum, which assumes that Agents have latent attributes which are Gaussian clustered, rather than simply categorized in independent groups. There have also been generalizations based on mutual information, as proposed in Kong and Schoenebeck, 2019, but they require that the underlying distribution be drawn from a known, finitely parameterizable distribution family. In addition, they are only truthful in the infinite Agent limit.

We focus on extensions with minimal additional prior details. Instead, we consider more general frameworks for extending Peer Consistency mechanisms, and work backwards to identify the belief update conditions that satisfy incentive-compatibility.

2 Influence Mechanisms

2.1 Introduction

The success of machine learning depends to a large extent on the availability of high quality data. For many applications, data has to be elicited from independent and sometimes self-interested data providers. A good example is federated learning, where a single *Center* (e.g. a large company) collects data from a set of *Agents* to jointly learn a model (Konečný et al., 2016). Other examples of such settings can be found in *crowdsourcing*. We consider a setting in which a Center wishes to construct a predictive model with the prediction accuracy measured according to some non-negative loss function, where lower loss means a more accurate model. The Center does not possess sufficient data with which to learn this model via supervised learning, so it must collect data by crowdsourcing. In federated learning, a set of Agents jointly learns a predictive model without revealing their data to each other. The Center communicates with the Agents and distributes the federated learning model to all of them. Agents can contribute actual data or changes that improve the current model based on the data, which may be more compact (Yang et al., 2019). If we consider this as a setting for applying a Peer Prediction mechanism for crowdsourcing, the current model is similar to a shared public prior. We will consider the slightly more canonical setting for Peer Consistency where there is a shared histogram of prior samples among the Center and Agents, and this shared set of prior samples determines a prior model. Agents will also directly contribute data samples rather than model updates.

There is clearly an incentive to free-ride: an Agent can benefit from the joint model or the mechanism rewards without contributing any novel or useful data, for example by fabricating data that fits the current model, or using random noise. If the rewards for data contributions do not have the correct incentives, the Agent can also benefit from supplying meaningless data. We call such strategies that are not based on the truthful data *heuristic* strategies. An Agent may also wrongly report its data, for example by obfuscating it to achieve differential privacy Dwork, 2008. There is no way for the Center to tell a priori if data has been manipulated, and given that it can strongly degrade the model, it is important to protect the process against it. Even

worse, a malicious Agent could intentionally insert wrong data and poison the model; we do not consider malicious behavior and assume that Agents have no interest in manipulating the model.

Free-riding can be avoided by incentives that compensate for the effort of a contributing Agent. An incentive scheme will distribute rewards to Agents in return for data samples. An instance of the game in this setting involves two game nodes in which Agents take actions:

- *observation node*: Agents decide if they will make the necessary effort to obtain truthful data, rather than use a *heuristic* strategy to make up data with no effort,
- *reporting node*: Agents decide what data sample to report to the Center.

A truthful strategy involves expending the necessary effort to obtain truthful data at the observation node, then truthfully reporting that data at the reporting node. We observe that both properties can be satisfied if contributions are rewarded according to their *Influence* Cook and Weisberg, 1980 on the model.

A similar question to the one in this section was considered by Cai et al., 2015, where the authors design strategy-proof mechanisms for eliciting data and achieving a desired trade-off between the accuracy of the model and the payments issued. The guarantees provided, while desirable, require the adoption of certain strong assumptions. The authors assume that each Agent chooses an *effort level*, and the variance of the accuracy of their reports is a strictly decreasing convex function of that effort. Furthermore, this function needs to be known to the Center. We will see that our only requirement is that the cost of effort is bounded by a known quantity. Furthermore, our strategy space is more expressive in the sense that, as in real-life scenarios, data providers can choose which data to provide and not just which effort level to exert.

2.1.1 Model

We say that there is a distribution Φ , called the *target distribution*, which produces independent random variables $z = (x, y)$, where x and y are referred to as *inputs* and *labels* respectively. The Agents can independently sample Φ by exerting effort, and an Agent A_i has his own personal effort function $e_i(z)$ which may depend on the sample z which gets observed. The effort might come from the Agent having to solve a problem to acquire data, or the data might be of a personal nature so the Agent is reluctant to access or utilize it. We adopt a simple but general effort model, in which the Agent makes a binary choice either to exert effort to acquire data, or exert no effort and report based on some *heuristic*. When the Agent decides to expend effort, there is some expected effort $\mathbf{e}_i = \mathbb{E}_{\Phi}[e_i(z)]$ over the distribution of observations, and \mathbf{e}_i is known to the Agent a priori.

For machine learning the Center is not actually concerned with learning the joint distribution Φ , rather, it is concerned with learning a mapping between x and y that minimizes some *loss*

function. To make the problem tractable, the Center restricts the mapping to a parameterizable model family:

Definition 2.1.1 (Model Family). Let f be a *model family* parameterized by θ such that $f_\theta(x) = \hat{y}$ where \hat{y} is the estimate of the representative value of $\Phi_{y|x}$, otherwise known as the *prediction*.

The Center performs some optimization to pick the model from the family which minimizes a loss function. For the Influence, we are not concerned with the particulars of the optimization procedure, we are only concerned with the outcome:

Definition 2.1.2 (True Risk and Optimizer). Let $L(y, \hat{y})$ be a non-negative loss function. Let Φ be the target distribution of the random variable $z = (x, y)$, with Ω the fundamental set. The *true risk* is given by $\mathbf{L}_\theta(\Phi) = \int_\Omega L(y, f_\theta(x)) d\Phi(z)$. We will often write $L(y, f_\theta(x))$ as $L_\theta(z)$. Then the *true optimizer* is given by $\hat{\theta}(\Phi) = \operatorname{argmin}_\theta \mathbf{L}_\theta(\Phi)$.

Since Φ is unknown, the optimization is performed over a set of samples called the training set:

Definition 2.1.3 (Empirical Risk and Optimizer). Let $Z = \{z_i\}_{i \in [1, n]}$ be a set of n input-label pairs. The *empirical risk* is given by $\mathbf{L}_\theta(Z) = \frac{1}{n} \sum_{i=1}^n L_\theta(z_i)$. Then the *empirical risk optimizer* is given by $\hat{\theta}(Z) = \operatorname{argmin}_\theta \mathbf{L}_\theta(Z)$.

Initialization

We assume the Center possesses at least a small amount of prior knowledge about the distribution Φ . Regardless of the prior knowledge, the Center will construct a *prior sample set* Z_R assumed to be sampled i.i.d from a *prior belief* R about the distribution Φ . The Center might already be in possession of some samples, in which case Z_R is automatically determined. Alternatively, the Center might have information about the distribution Φ . For example, the Center might know bounds on the values of $z \sim \Phi$. In this case the Center could construct R as the maximal entropy distribution with this property, in other words R would be uniform over the range of values. As another example, the Center might only know the mean and covariance of $z \sim \Phi$. Again, the Center would construct R as the maximal entropy distribution, which would be Gaussian with the known mean and covariance. After constructing R , the Center would sample R to generate Z_R , with the number of samples representing the degree of confidence the Center has in the prior R . If the Center has effectively no confidence, it would only sample R a minimal number of times in order to determine an initial model θ . For example, if the model family is a one-dimensional linear regression, there needs to be at least two samples to determine a model.

2.1.2 Influence

In order to compute Influence, the Center will use two sets of samples Z_T and Z_V as the training set and validation set respectively. We write Z_T^{-i} as the set Z_T with z_i removed. The Influence is a pair-wise score which quantifies the effect of one training sample on the model family's loss with respect to a validation sample:

Definition 2.1.4 (Influence). Given a model family f parameterized by θ to minimize a loss function L , and given $z_i \in Z_T$ and $z_j \in Z_V$, the Influence of z_i on z_j is given by:

$$\mathcal{I}(z_i, Z_T, z_j) = L_{\hat{\theta}(Z_T^{-i})}(z_j) - L_{\hat{\theta}(Z_T)}(z_j) \quad (2.1)$$

We often consider the average Influence over the validation set, written as:

$$\mathcal{I}(z_i, Z_T, Z_V) = \mathbf{L}_{\hat{\theta}(Z_T^{-i})}(Z_V) - \mathbf{L}_{\hat{\theta}(Z_T)}(Z_V) \quad (2.2)$$

We will often omit Z_T or Z_V from the argument when they are clear from context. We can also write $\mathcal{I}(z_i, Z_T, \psi)$ as the expected Influence over validation points sampled from a distribution ψ .

Clearly, Influence is a useful measure from the point of view of the Center, since it rewards contributions that make the model converge as fast as possible. We will see that a mechanism which rewards Agents according to Influence, under some basic assumptions, has the following incentives:

- Truthful Dominant Strategy Equilibrium: when the validation set is known to be sampled from Φ .
- Truthful Bayes-Nash Equilibrium: when the validation set is taken randomly from Agent reports.

2.2 Influence-based Incentives

2.2.1 The Mechanism

In a single round of data collection, the Center shares the model family f , the loss function L , and the prior sample set Z_R with the Agents. It is assumed that both the Center and the Agents are capable of computing $\hat{\theta}(Z)$ for any finite set of samples Z . The Center then collects a set of samples Z_A from the Agents during a *data collection period*. The Center may or may not be in possession of a private *trusted* validation set Z_V with samples drawn i.i.d. from Φ . If there is not a trusted validation set, the Center can construct the validation set by randomly picking some proportion $(1 - s)$ of the samples Z_A . With the validation set, the Center can randomly pick a validation sample and compute the Influence of a training sample on that validation sample. Alternatively, the Center could compute the average Influence on the validation set,

which gives an equivalent incentive, since the Agent computes its score as an expectation over the distribution of validation samples. We will use this formulation.

There are also two possible ways to construct the training set Z_T . When computing the Influence for a particular Agent with a particular report z , the training set could be constructed as $Z_T = Z_R \cup z$. We call this a *single-trained* mechanism. Alternatively, the training set could be constructed as $Z_T = Z_R \cup Z_A$, which we call a *mixed-trained* mechanism.

Given a training set Z_T and a validation set Z_V , the reward given to an Agent for reporting a sample $z_i \in Z_T$ is given by:

$$\tau(z_i, Z_T, Z_V) = F(Z_V) + c\mathcal{J}(z_i, Z_T, Z_V)$$

where $F(Z_V) = \mathbb{E}_{Z_V \in Z_V} [f(Z_V)]$ and $c > 0$ is a constant. In the case when Z_V is constructed with a proportion $(1 - s)$ of the Agent reports, those samples receive a reward of 0, so the remaining samples must be scaled:

$$\tau^*(z_i, Z_A) = \begin{cases} \frac{\tau(z_i, Z_T, Z_V)}{s} & z_i \notin Z_V \\ 0 & z_i \in Z_V \end{cases}$$

Since $z_i \notin Z_V$ with probability s , the expected reward over this random choice is $s \frac{\tau(z_i, Z_T, Z_V)}{s} + (1 - s) * 0 = \tau(z_i, Z_T, Z_V)$. So the expected reward is the same as in the case when there is a trusted validation set.

We will first show that if a truthful dominant strategy exists in the case of a trusted validation set, then there must be a truthful Bayes-Nash Equilibrium in the case of a validation set taken from Agent reports:

Proposition 2.2.1. *If an Agent has a truthful dominant strategy when rewarded according to τ with a trusted Z_V , then there exists a truthful BNE when rewarded according to τ^* with Z_V taken from a random proportion $(1 - s)$ of the Agent reports.*

Proof. If an Agent has a truthful dominant strategy when rewarding according to τ with a trusted Z_V , then if he observes sample o from Φ , $\forall Z_T^{-i}$, reporting $z_i = o$ yields the highest expected reward for the Agent according to τ . If Z_V is taken from a proportion $(1 - s)$ of Agent reports, and the other Agents report truthfully, then with probability $(1 - s)$, z_i will be in Z_V and the Agent will receive a reward of 0, and with probability s , z_i will not be in Z_V , so Z_V will consist of truthful samples, making it equivalent in distribution to the trusted Z_V . The expected payment of the Agent is $\tau(z_i, Z_T, Z_V)$, which has highest payoff for $z_i = o$. Therefore, if other Agents report truthfully, it is highest expected payoff for the Agent to report truthfully, so there is a truthful BNE. \square

We can now focus our attention on the case of a trusted validation set and search for a Dominant Strategy Equilibrium.

2.2.2 Dominant Strategy Incentive-Compatibility

Agent Beliefs

In order for an Agent to compute a report to optimize the reward, it must have some belief about the distribution of the validation set Z_V . Since the validation set is private and trusted, this is equivalent to having a belief about the distribution Φ . We assume that, prior to the data collection period, Agents are *uninformed* about the distribution Φ :

Definition 2.2.2 (Uninformed as Representative). An Agent is *uninformed* if he adopts Z_R as *representative* of the distribution Φ .

What does it mean for a sample set Z_R to be representative of a distribution? The Agent must ultimately construct a distribution of Z_V to compute expected Influences. The Agent can use a number of potential distribution modeling techniques to construct a prior π from the prior samples Z_R . Most simply, π could be the empirical measure of Z_R : $\pi(a) = \frac{1}{|Z_R|} \sum_{z \in Z_R} \mathbb{1}_{z \in a}$ where a is some event. Alternatively, π could be constructed as a Gaussian mixture model of Z_R . However, the prior π must be constructed in a way that the optimizer of π is the same as the optimizer of Z_R :

Definition 2.2.3 (Representative). A set of samples Z is *representative* of a distribution π with respect to a model family f if and only if $\hat{\theta}(\pi) = \hat{\theta}(Z)$.

We can also consider going in the reverse direction. Given some distribution π , we can consider the equivalence class of representative samples of size n :

Definition 2.2.4 (Representation Class). Given a distribution π and a positive integer n , $\mathbf{Z}_{\pi,n}$ is the class of sets of samples Z such that $|Z| = n$ and Z is representative of π .

After constructing a prior π , the Agent might expend effort and observe a sample o of Φ , after which they update their prior to a posterior π_o . Behind the prior π is the representative sample set Z_π with n samples. Most naturally, the Agent would simply add o to the sample set to produce $Z_{\pi_o} = Z_\pi \cup o$, then use the same process of mapping a sample set to a distribution to produce π_o such that Z_{π_o} is a representative:

Definition 2.2.5 (Empirically Consistent). An Agent's belief update is *empirically consistent* if and only if, for a prior π constructed from a prior sample set Z_π with size n , $Z_\pi \cup o \in \mathbf{Z}_{\pi_o, n+1}$ for the posterior π_o .

Single-Trained Mechanism

An uninformed, empirically consistent Agent A_i will adopt the prior sample set Z_R as representative of a prior belief R , the Agent's estimate of Φ . After making an observation o , the Agent will update his belief to a posterior belief R_o such that $Z_R \cup o$ is representative of R_o . In

order for truthfulness to be a dominant strategy, it must be highest payoff regardless of the strategy profile of the Peers. In this case, the Peer reports have no effect on the payment to the Agent. The Agent computes the expected reward for reporting r :

$$\begin{aligned}\tau(r, Z_R \cup r, R_o) &= F(R_o) + c\mathcal{J}(r, Z_R \cup r, R_o) \\ &= F(R_o) + c(\mathbf{L}_{\hat{\theta}(Z_R)}(R_o) - \mathbf{L}_{\hat{\theta}(Z_R \cup r)}(R_o))\end{aligned}$$

Maximizing this over r is equivalent to minimizing $\mathbf{L}_{\hat{\theta}(Z_R \cup r)}(R_o)$ over r . Since $Z_R \cup o$ is representative of R_o , $\hat{\theta}(R_o) = \hat{\theta}(Z_R \cup o)$, so $r = o$ is a minimizer of $\mathbf{L}_{\hat{\theta}(Z_R \cup r)}(R_o)$:

Proposition 2.2.6. *An uninformed, empirically consistent Agent who observes o has a dominant strategy of reporting $r = o$ when $Z_T = Z_R \cup r$.*

Proof. An uninformed, empirically consistent Agent who observes o constructs the distribution R_o as an estimate of Φ such that $\hat{\theta}(Z_R \cup o) = \hat{\theta}(R_o)$:

$$\begin{aligned}\hat{\theta}(R_o) &= \operatorname{argmin}_{\theta} \mathbf{L}_{\theta}(R_o) \\ \Rightarrow \hat{\theta}(Z_R \cup o) &= \operatorname{argmin}_{\theta} \mathbf{L}_{\theta}(R_o) \\ \Rightarrow \forall z \neq o: \quad \mathbf{L}_{\hat{\theta}(Z_R \cup o)}(R_o) &\leq \mathbf{L}_{\hat{\theta}(Z_R \cup z)}(R_o) \\ \Rightarrow \mathbf{L}_{\hat{\theta}(Z_R)}(R_o) - \mathbf{L}_{\hat{\theta}(Z_R \cup o)}(R_o) &\geq \mathbf{L}_{\hat{\theta}(Z_R)}(R_o) - \mathbf{L}_{\hat{\theta}(Z_R \cup z)}(R_o) \\ \Rightarrow \tau(o, Z_R \cup o, R_o) &\geq \tau(z, Z_R \cup z, R_o)\end{aligned}$$

□

Mixed-Trained Mechanism

Now suppose that the mechanism constructs $Z_T = Z_R \cup Z_A$. Let us right Z_A^{-r} as the set of Agent reports excluding the particular Agent with report r , in other words Z_A^{-r} are the Peer reports. Once again, the Agent computes the expected reward for reporting r :

$$\begin{aligned}\tau(r, Z_R \cup Z_A, R_o) &= F(R_o) + c\mathcal{J}(r, Z_R \cup Z_A, R_o) \\ &= F(R_o) + c(\mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r})}(R_o) - \mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r} \cup r)}(R_o))\end{aligned}$$

In order for $r = o$ to be a maximizer, the following inequality must hold:

$$\begin{aligned}\forall z \neq o: \quad \mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r})}(R_o) - \mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r} \cup o)}(R_o) &\geq \mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r})}(R_o) - \mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r} \cup z)}(R_o) \\ \Rightarrow \mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r} \cup o)}(R_o) &\leq \mathbf{L}_{\hat{\theta}(Z_R \cup Z_A^{-r} \cup z)}(R_o)\end{aligned}$$

This is clearly no longer satisfied for any choice of Z_A^{-r} , so there can be no Dominant Strategy Equilibrium. However, there may still be an optimal strategy when the other Agents are truthful, in other words, a truthful Bayes-Nash Equilibrium. Under a truthful BNE, the Agent believes

that the samples in Z_A^{-r} are distributed according to R_o . The Agent also believes that R_o is the true distribution and that $\hat{\theta}(Z_R \cup o)$ is the true optimizer of this distribution. When adding a random variable to the sample set, the Agent computes the optimal model to be the model which optimizes the expected loss over the distribution of the random variable:

Lemma 2.2.7. *Let z be a random variable distributed according to a distribution π and let Z be a set of samples which are representative of π . Then $\hat{\theta}(Z \cup z) = \hat{\theta}(Z)$.*

Proof. From Definition 2.1.3, $\hat{\theta}(Z) = \operatorname{argmin}_{\theta} \sum_{z' \in Z} L_{\theta}(z') = \operatorname{argmin}_{\theta} \mathbb{E}_{z \sim \pi} [\sum_{z' \in Z} L_{\theta}(z')]$. Also, $\hat{\theta}(\pi) = \operatorname{argmin}_{\theta} \mathbb{E}_{z \sim \pi} [L_{\theta}(z)]$. Since Z is representative of π , from Definition 2.2.3 $\hat{\theta}(Z) = \hat{\theta}(\pi)$. Therefore $\hat{\theta}(Z) = \operatorname{argmin}_{\theta} \mathbb{E}_{z \sim \pi} [L_{\theta}(z)] + \mathbb{E}_{z \sim \pi} [\sum_{z' \in Z} L_{\theta}(z')] = \mathbb{E}_{z \sim \pi} [L_{\theta}(z) + \sum_{z' \in Z} L_{\theta}(z')] = \hat{\theta}(Z \cup z)$. \square

Corollary 2.2.8. *Let Z^* be a finite set of random variables i.i.d. according to a distribution π and let Z be a set of samples which are representative of π . Then $\hat{\theta}(Z \cup Z^*) = \hat{\theta}(Z)$.*

Corollary 2.2.8 follows directly from Lemma 2.2.7 by induction.

We can now prove the existence of a truthful optimal strategy.

Proposition 2.2.9. *An uninformed, empirically consistent Agent who observes o has a highest payoff strategy of reporting $r = o$ when $Z_T = Z_R \cup Z_A$ and the Peers report truthfully.*

Proof. Since the Peers are truthful, the Agent believes that Z_A^{-r} is composed of i.i.d. samples distributed according to R_o . From the assumption that the Agent is uninformed and empirically consistent, $Z_R \cup o$ is representative of R_o . Then from Corollary 2.2.8, $\hat{\theta}(Z_R \cup Z_A^{-r} \cup o) = \hat{\theta}(Z_R \cup o) = \hat{\theta}(R_o)$. The argument then follows directly along the lines of Proposition 2.2.6. \square

Eliminating Heuristic Strategies

We have shown when there are dominant truthful strategies after observing a sample o , but observing the sample requires effort. The Agent can play a heuristic strategy without expending effort and making an observation. We will first show that a heuristic strategy will have an expected payoff of $F(R)$ when the Agent is uninformed and believes the model family is *risk-monotonic* with respect to Φ :

Definition 2.2.10 (Risk-monotonic). A model family f_{θ} minimizing a loss function L is *risk-monotonic* with respect to a distribution Φ if $\forall n > n_{\min}$ and $z, z_i \sim \Phi$, $\mathbb{E}[L_{\hat{\theta}(\{z_i\}_{i \in [1, n-1]})}(z)] \geq \mathbb{E}[L_{\hat{\theta}(\{z_i\}_{i \in [1, n]})}(z)]$. The model family is *strictly risk-monotonic* if this inequality is strict.

Risk-monotonicity simply states that when sampling from a distribution, the empirical optimizer on the samples is expected to improve its predictions on the distribution. While Loog

et al., 2019 show that not all model families are risk-monotonic, their counter-examples are adversarially constructed. As Agents have no prior knowledge about Φ , we consider it reasonable to make this an assumption.

An uninformed Agent does not know anything about the relationship between R and Φ . In general, a report r is a random variable based on some reporting strategy. If the random variable r is not distributed according to R , and the Agent believes it will be higher scoring than a report distributed according to R , then the Agent must believe that the distribution of r is closer to Φ than R is, otherwise risk-monotonicity is violated. This would mean that the distribution of r represents a better prior estimate of Φ than R , violating the assumption that the Agent is uninformed. So we have the relation:

$$F(R) + c\mathbb{E}_{z \sim r}[\mathcal{J}(z, Z_T, R)] \leq F(R) + c\mathbb{E}_{z \sim R}[\mathcal{J}(z, Z_T, R)]$$

Prior to making an observation, if $Z_T = Z_R \cup z$, or $Z_T = Z_R \cup Z_A$ and the Peers are truthful, then $\mathbb{E}_{z \sim R}[\mathcal{J}(z, Z_T, Z_V)] = \mathbf{L}_{\hat{\theta}(R)}(R) - \mathbf{L}_{\hat{\theta}(R)}(R) = 0$. So the payment for any uninformed heuristic strategy is at most $F(R)$. We compare the expected payoff of a truthful strategy to that of a heuristic strategy, noting that a truthful optimal strategy requires that the truthful report be highest payoff:

$$\begin{aligned} \mathbb{E}_{z_i \sim \Phi}[\tau(z_i, Z_T, Z_V) - e_i(z)] &= F(Z_V) + c\mathbb{E}_{z_i \sim \Phi}[\mathcal{J}(z_i, Z_T, Z_V)] - \mathbf{e}_i > F(Z_V) \\ &\Rightarrow c\mathbb{E}_{z_i \sim \Phi}[\mathcal{J}(z_i, Z_T, Z_V)] - \mathbf{e}_i > 0 \end{aligned}$$

As long as $\mathbb{E}_{z_i \sim \Phi}[\mathcal{J}(z_i, Z_T, Z_V)] > 0$, the Center can choose $c > \frac{\mathbf{e}_i}{\mathbb{E}_{z_i \sim \Phi}[\mathcal{J}(z_i, Z_T, Z_V)]}$ to satisfy the inequality. The inequality $\mathbb{E}_{z_i \sim \Phi}[\mathcal{J}(z_i, Z_T, Z_V)] > 0$ follows directly from the risk monotonicity assumption.

Incentive-Compatibility

We can finally make a statement about the existence of truthful dominant strategies:

Theorem 2.2.11. *Given $c > \frac{\mathbf{e}_i}{\mathbb{E}_{z \sim \Phi}[\mathcal{J}(z, Z_T, Z_V)]}$ in τ for an uninformed, empirically consistent Agent, the Agent has a highest payoff strategy of reporting $r = o$ when the model family is risk-monotonic and either: 1) $Z_T = Z_R \cup r$, or 2) $Z_T = Z_R \cup Z_A$ and the Peers are truthful.*

Proof. We have $c > \frac{\mathbf{e}_i}{\mathbb{E}_{z \sim \Phi}[\mathcal{J}(z, Z_A, Z_V)]} \Rightarrow \mathbb{E}_{z_i \sim \Phi}[\tau(z_i, Z_A, Z_V) - e_i(z)] > 0$, so uninformed heuristic strategies are expected to be lower payoff than truthful reporting prior to making an observation. Truthful reporting is highest payoff after making an observation o following Propositions 2.2.6 and 2.2.9. \square

Corollary 2.2.12. *There exists a Bayes-Nash Equilibrium of truthful reporting under payoff function τ^* given the same assumptions as in Theorem 2.2.11, but with Z_V composed of randomly selected reports.*

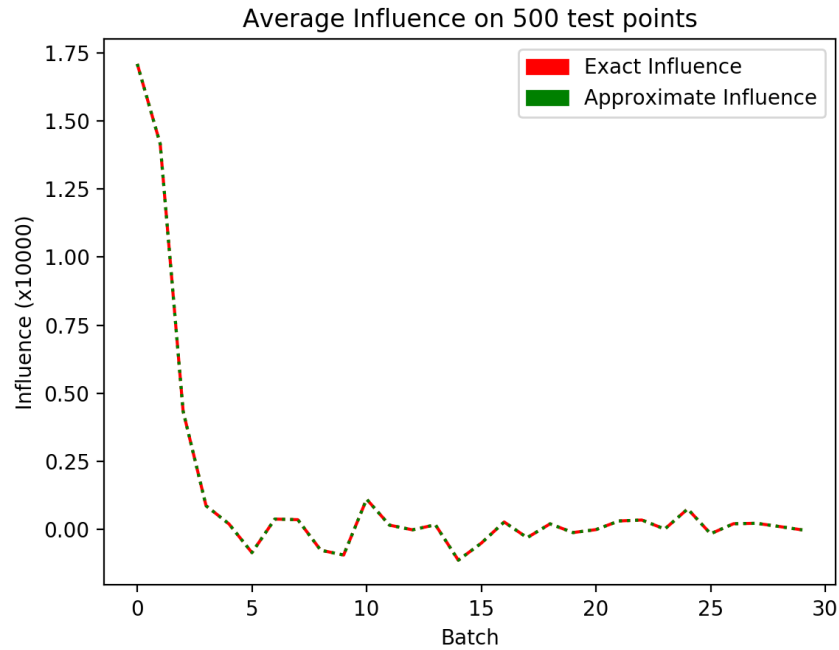


Figure 2.1: Empirically observed decrease of Influence on a typical regression model as more and more data is collected. Each batch corresponds to 100 data points. Both the exact Influence and the 2nd order approximation are shown.

Proof. This follows directly from Theorem 2.2.11 and Proposition 2.2.1. □

2.3 Incentives for the Center

2.3.1 Budgeting

In general, the share of an additional data point in a model based on $n - 1$ earlier data points is $1/n$. Many loss functions, such as the mean-squared error or the cross entropy error, decrease as $1/n$ with the number of samples. The Influence is proportional to the derivative of the loss function and thus decreases as $1/n^2$. Figure 2.1 shows an example of the actual decrease of Influence on a regression model. We can observe two phases: an initial phase, where additional data is necessary to make the model converge, and a converged phase where the expected Influence is close to zero. This is because the model is never a perfect fit to the data, but always leaves some remaining variance. Once this variance is reached, additional data will not help to reduce it, and no further incentives should be given to provide such data.

Using Influence as an incentive has the property that the expected reward is either close to 0, or it decreases as $1/n^2$. Therefore, it is always best for Agents to report data as early as possible. This is a valuable property for real world application. If an Agent is incentivized to wait to

submit data, the Center might never be able to learn a good model.

More importantly, Influence mechanisms allow for bounded budgeting by the Center. The Center must have a reason to be building a model. Suppose the Center has a *value function* $V(\mathbf{L})$ which returns the Center's utility for a model with true risk \mathbf{L} on Φ . We assume that the value increases as the risk of the model decreases, so V is monotonically decreasing. Let us assume that, prior to the crowdsourcing endeavor, the Center possesses a baseline model with risk \mathbf{L}_0 attaining an initial value $V(\mathbf{L}_0) = 0$ without loss of generality. There is also some minimum risk \mathbf{L}_{\min} which is achieved by the true optimizer, and achieves a maximum value $V_{\max} = V(\mathbf{L}_{\min})$.

Let us examine the values a Center might expect to achieve through crowdsourcing via an Influence mechanism. For simplicity we will set $F(Z_V) = 0$ in the payment function τ . For a particular choice of c , when the Center elicits n reports, it expects to pay $c \int_1^n \frac{\mathcal{J}_0}{n^2} = c \mathcal{J}_0 \frac{n-1}{n}$ where \mathcal{J}_0 is the initial Influence value. The risk attained by the model is expected to be $\mathbf{L}_{\min} + \frac{\mathbf{L}_0 - \mathbf{L}_{\min}}{n}$, so the value is $V(\mathbf{L}_{\min} + \frac{\mathbf{L}_0 - \mathbf{L}_{\min}}{n})$. A particularly risk averse Center might want an assurance that it never loses value during the data collection process. Perhaps the Center may stop before exhausting the budget, hoping to have attained some value. In other words, the Center requires that $c \mathcal{J}_0 \frac{n-1}{n} < V(\mathbf{L}_{\min} + \frac{\mathbf{L}_0 - \mathbf{L}_{\min}}{n})$. This yields a simple relation for choosing c : $c < \frac{nV(\mathbf{L}_{\min} + \frac{\mathbf{L}_0 - \mathbf{L}_{\min}}{n})}{(n-1)\mathcal{J}_0}$. We see that c is a function of n , so this can be adjusted over multiple data collection periods.

If the Center is not risk averse in this way. Suppose the Center has some a priori budget B and it wishes to exhaust this budget to attain as much value as possible. The Center must attain at least a value B , so it can compute the expected number of points it needs by finding $n_{\min} = \min_n : V(\mathbf{L}_{\min} + \frac{\mathbf{L}_0 - \mathbf{L}_{\min}}{n}) > B$. The Center can achieve this by choosing a constant $c < \frac{n_{\min} B}{(n_{\min} - 1)\mathcal{J}_0}$.

We see, however, that the budget constraints also constrain the Agents. An Agent is only incentivized to make an observation and report truthfully if the Agent's expected effort level satisfies $c > \frac{\mathbf{e}}{\mathbb{E}_{z \sim \Phi}[\mathcal{J}(z, Z_A, Z_V)]} \sim \frac{n^2 \mathbf{e}}{\mathcal{J}_0}$. So now we see our constraints for viable mechanisms satisfying both budget and effort constraints for the case of dynamic and fixed c :

$$\frac{n^2 \mathbf{e}}{\mathcal{J}_0} < c < \frac{nV(\mathbf{L}_{\min} + \frac{\mathbf{L}_0 - \mathbf{L}_{\min}}{n})}{(n-1)\mathcal{J}_0}$$

$$\frac{n_{\min}^2 \mathbf{e}}{\mathcal{J}_0} < c < \frac{n_{\min} B}{(n_{\min} - 1)\mathcal{J}_0}$$

We examine when it is not possible to satisfy these constraints:

$$\mathbf{e} \geq \frac{V(\mathbf{L}_{\min} + \frac{\mathbf{L}_0 - \mathbf{L}_{\min}}{n})}{n(n-1)}$$

$$\mathbf{e} \geq \frac{B}{n_{\min}(n_{\min} - 1)}$$

The Center can use these expressions to determine not only the budgeting for a crowdsourcing project, but it can determine a priori if the project is practically viable if it has a priori knowledge about Agent effort levels.

2.3.2 Improved Equilibria

A Bayes-Nash Equilibrium is a weaker solution concept than a Dominant Strategy Equilibrium, so it is interesting to ask if the Center can make truthful behavior the dominant strategy even when test data has to be obtained from Agents. Clearly, if all test data is supplied by Agents, this is not possible: consider the example where all but one Agent A_i submit test data according to a synthetic model θ' , but only Agent A_i observes true data according Φ , which has a different true optimizer θ^* . Then it will be better for Agent i to report incorrectly according to model θ' , so truthful behavior cannot be a Dominant Strategy.

However, it turns out that if only a fraction of the test data is supplied by untrusted Agents, we can place a bound on this fraction so that truthful behavior is still a highest payoff strategy. To obtain such a result, we need to exclude consideration of the cost of effort and focus on the reporting strategy only, since we do not know what is the relative cost of obtaining true vs. heuristic data.

Let Φ_1 be the distribution of truthful reports and Φ_2 be the distribution of heuristic reports. We assume they describe an input-output relationship such that $\Phi(x, y) = q(x)p(y|x)$, and $q_1(x) = q_2(x)$. This assumption merely asserts that the data we are collecting is drawn from the same distribution of inputs regardless of the distribution of the output. Distributions Φ_1 and Φ_2 determine, in expectation, true optimizer models θ_1 and θ_2 respectively. Let us now define $\mathbf{L}_{i,j}$ as the expected risk of model θ_i evaluated on distribution Φ_j . Given some fixed training data set with points drawn from a mixture of Φ_i and Φ_j , let $\mathcal{I}_{i,j}$ be the expected Influence of a data point sampled from distribution Φ_i on a test point from distribution Φ_j . Using the standard mean-squared-error loss function, we have that $\mathbf{L}_{i,j} = \mathbf{L}_{j,j} + r$ where $r = \mathbb{E}_{x \sim q} [(f_{\theta_i}(x) - f_{\theta_j}(x))^2]$. We can then state the following:

Theorem 2.3.1. *As long as the test data contains at most a fraction*

$$p < \frac{\mathcal{I}_{2,2}/\mathbf{L}_{2,2}}{\mathcal{I}_{1,1}/\mathbf{L}_{1,1} + \mathcal{I}_{2,2}/\mathbf{L}_{2,2}} + \frac{\mathcal{I}_{1,1} - \mathcal{I}_{2,2}}{r(\mathcal{I}_{1,1}/\mathbf{L}_{1,1} + \mathcal{I}_{2,2}/\mathbf{L}_{2,2})}$$

of non-truthful reports, truthful reporting remains the highest payoff strategy for Agents that do not choose to opt out.

Proof. Now suppose we sample n_1 points from Φ_1 and n_2 points from Φ_2 to form our training set $\{z\}$, and call the resulting distribution Φ_c . Note that as $\mathbf{L}_{1,2} - \mathbf{L}_{1,1} = r$, and Influence is proportional to the empirical risk, the Influence of a sample following θ_1 but tested on a

sample from Φ_2 is decreased as follows:

$$\mathcal{I}_{1,2} = \mathcal{I}_{1,1}(1 - r/\mathbf{L}_{1,1})$$

and so the expected Influence when evaluating on the mixture (n_1, n_2) is

$$\begin{aligned}\mathcal{I}_{1,c} &= \mathcal{I}_{1,1}\left(1 - r/\mathbf{L}_{1,1} \frac{x_2}{n}\right) = \mathcal{I}_{1,1}(1 - pr/\mathbf{L}_{1,1}) \\ \mathcal{I}_{2,c} &= \mathcal{I}_{2,2}\left(1 - r/\mathbf{L}_{2,2} \frac{x_1}{n}\right) = \mathcal{I}_{2,2}(1 - r(1-p)/\mathbf{L}_{2,2})\end{aligned}$$

To ensure that reporting samples from Φ_1 carry a higher expected reward, we want to satisfy:

$$\begin{aligned}\mathcal{I}_{1,c} &> \mathcal{I}_{2,c} \\ \mathcal{I}_{1,1} - \mathcal{I}_{2,2}(1 - r/\mathbf{L}_{2,2}) &> pr(\mathcal{I}_{1,1}/\mathbf{L}_{1,1} + \mathcal{I}_{2,2}/\mathbf{L}_{2,2}) \\ p &< \frac{\mathcal{I}_{1,1} - \mathcal{I}_{2,2}(1 - r/\mathbf{L}_{2,2})}{r(\mathcal{I}_{1,1}/\mathbf{L}_{1,1} + \mathcal{I}_{2,2}/\mathbf{L}_{2,2})} \\ &= \frac{\mathcal{I}_{2,2}/\mathbf{L}_{2,2}}{\mathcal{I}_{1,1}/\mathbf{L}_{1,1} + \mathcal{I}_{2,2}/\mathbf{L}_{2,2}} + \frac{\mathcal{I}_{1,1} - \mathcal{I}_{2,2}}{r(\mathcal{I}_{1,1}/\mathbf{L}_{1,1} + \mathcal{I}_{2,2}/\mathbf{L}_{2,2})}\end{aligned}$$

□

If $\mathcal{I}_{2,2}/\mathbf{L}_{2,2} = \mathcal{I}_{1,1}/\mathbf{L}_{1,1}$, the first term is $= 1/2$. The second term is a correction: if $\mathcal{I}_{1,1} > \mathcal{I}_{2,2}$, more non-truthful reports are tolerated as the Influence when improving the first model is stronger, otherwise it is the other way around.

A Center could use this result to decide how much test data to obtain from Agents. As the underlying phenomenon could evolve over time, it is advantageous for the Center to include some contributed data in its test set so that such evolution can be tracked. To evaluate the bound, the Center could compare the statistics of scores obtained with trusted test data with those obtained using contributed test data, and thus estimate the parameters $\mathcal{I}_{i,j}$, as well as the empirical risks of models fitted to the trusted and contributed data to estimate the parameters $\mathbf{L}_{i,j}$. The Center could thus obtain a stronger guarantee on the quality of the test data.

2.4 Relation with Peer Consistency

We have shown that, under certain assumptions, an Influence-based mechanism is incentive-compatible. It is also clear that Influence mechanisms operate on arbitrary distributions, with the caveat that the model family and loss function are not arbitrary. But in the case of the distribution Φ being finite and discrete, the setting is the same as the classical crowdsourcing setting for Peer Consistency. We show that with the correct model family and loss function,

the Influence mechanism actually replicates the rewards structure of the Peer Truth Serum (Faltings et al., 2014). We write the PTS reward function here for reference:

Definition 2.4.1 (Peer Truth Serum Reward). The *PTS reward function* is given by

$$\tau_{\text{PTS}}(r, rr, R) = f(rr) + \frac{c \mathbb{1}_{r=rr}}{R(r)} \quad (2.3)$$

where r is the Agent's report, rr is a randomly chosen Peer report, and R is the current estimate of the distribution Φ by the Center.

In this setting, the Center constructs a model of the distribution Φ as a histogram of the observations reported by the Agents. Suppose there are k discrete elements x_j of Φ . In the machine learning setting, Agents observe input-label pairs. In this setting, the current estimate R of ϕ is made

Given a dataset Z with n elements, let n_j be the number of samples equal to x_j so that $\sum_{j=1}^k n_j = n$. Let the empirical optimizer $\hat{\theta}(Z) = \langle \frac{n_1}{n}, \frac{n_2}{n}, \dots, \frac{n_k}{n} \rangle$ and the model simply returns the value of the index: $f_{\theta}(z) = \theta_j$ for $x_j = z$. At the start of a data collection period, the Center possess a current empirical optimizer model $\hat{\theta}(Z)$ constructed by a set of samples Z of size n . If the Center observes a new sample x_j , the model is updated to $\hat{\theta}(Z \cup x_j)$ such that:

$$\begin{aligned} \hat{\theta}_j(Z \cup x_j) &= \delta + (1 - \delta)\hat{\theta}_j(Z) \\ \hat{\theta}_i(Z \cup x_j) &= (1 - \delta)\hat{\theta}_i(Z) \text{ for } i \neq j \end{aligned}$$

with $\delta = \frac{1}{n+1}$.

The Center's model is minimizing over the *logarithmic scoring rule* (LSR), which is given by:

$$L_{\theta}^{\text{LSR}}(z) = -\log(f_{\theta}(z)) \quad (2.4)$$

We briefly show that the $f_{\hat{\theta}(Z)}(x_j) = \hat{\theta}_j(Z)$ is the empirical optimizer of L^{LSR} . The empirical risk is given by $-\frac{1}{n} \sum_{i=1}^n \log(f_{\theta}(z_i)) = -\sum_{j=1}^k \frac{n_j}{n} \log(\theta_j)$. We wish to minimize this subject to the constraint $\sum_{j=1}^k \theta_j = 1$. We do so via Lagrange multiplier method: $\frac{\partial}{\partial \theta_j} (-\sum_{j=1}^k \frac{n_j}{n} \log(\theta_j) - \lambda(\sum_{j=1}^k \theta_j - 1)) = -\frac{n_j}{n\theta_j} - \lambda$. Setting this to 0 yields $\theta_j = -\frac{1}{\lambda} \frac{n_j}{n}$ for all j , so $\sum_{j=1}^k \theta_j = -\frac{1}{\lambda} \sum_{j=1}^k \frac{n_j}{n} = -\frac{1}{\lambda}$. So λ must be -1 , yielding $\theta_j = \frac{n_j}{n}$. The second derivative of the loss is $-\frac{\partial}{\partial \theta_j} \frac{n_j}{n} \theta_j^{-1} = \frac{n_j}{n\theta_j^2} = \frac{n_j}{n} > 0$, confirming that this solution is a minimizer.

We could compute the Influence of a new data point on the loss function directly. However, it is instructive to consider an approximation using the Taylor expansion of the loss function. We note that the derivative of $\hat{\theta}(Z \cup x_j)$ with respect to the parameter δ is as follows:

$$\frac{\partial \hat{\theta}(Z \cup x_j)}{\partial \delta} = \mathbf{1}_j - \hat{\theta}(Z)$$

where $\mathbf{1}_{i=j} = \langle 0, 0, \dots, 1, \dots, 0 \rangle$ with 1 only at index j . The derivative of the log scoring rule is:

$$\frac{\partial L_{\hat{\theta}(Z)}^{\text{LSR}}(x_j)}{\partial \hat{\theta}_j(Z)} = -\frac{\mathbf{1}_j}{\hat{\theta}(Z)}$$

and so the Taylor expansion of the logarithmic scoring rule applied to the assumed distribution is as follows. Since we would like random reporting according to the prior distribution $\hat{\theta}(Z)$ to be equal to 0, we make this the starting point of the expansion. We can then write the payment for a sample x_j on a validation sample x_l as:

$$\begin{aligned} & L_{\hat{\theta}(Z)}^{\text{LSR}}(x_l) - L_{\hat{\theta}(Z \cup x_j)}^{\text{LSR}}(x_l) \\ & \approx -\delta \sum_{i=1}^k \frac{\partial L_{\hat{\theta}(Z)}^{\text{LSR}}(x_l)}{\partial \hat{\theta}_i(Z \cup x_j)} \frac{d\hat{\theta}_i(Z \cup x_j)}{d\delta} \\ & = \delta \frac{\mathbf{1}_l}{\hat{\theta}(Z)} (\mathbf{1}_j - \hat{\theta}(Z)) \\ & = \delta \left(\frac{\mathbf{1}_{j=l}}{\hat{\theta}(Z)} - 1 \right) \end{aligned}$$

We note that the Influence of an Agent depends on whether its reported value matches the value of the validation sample, however, the Influence is scaled by the inverse of the prior probability of this value, so that unlikely values carry a higher Influence. From there, we can derive the following payment function:

$$\tau_{\text{PTS}}^{\mathcal{J}}(x_j, Z_V, \hat{\theta}(Z)) = \delta \left(\frac{1}{|Z_V|} \sum_{x_i \in Z_V} \frac{\mathbf{1}_{j=i}}{\hat{\theta}_j(Z)} - 1 \right)$$

where δ is a scaling factor, for example $\delta = \frac{1}{n+1}$. If the values don't match, the Agent has to pay δ (which could be a participation fee charged up front). We note that when δ is a constant, for example $\delta = 1$, this scheme exactly matches the Peer Truth Serum.

We also note that the general condition for having a strictly truthful dominant strategy with an Influence-based mechanism is :

$$\begin{aligned} \forall z \neq o: & \mathbf{L}_{\hat{\theta}(R_{\alpha,o})}(R_o) < \mathbf{L}_{\hat{\theta}(R_{\alpha,z})}(R_o) \\ \Rightarrow & \mathbf{L}_{\hat{\theta}(R)}(R_o) - \mathbf{L}_{\hat{\theta}(R_{\alpha,o})}(R_o) > \mathbf{L}_{\hat{\theta}(R)}(R_o) - \mathbf{L}_{\hat{\theta}(R_{\alpha,z})}(R_o) \\ \Rightarrow & \log(f_{\hat{\theta}_o}(o)) - \log(f_{\hat{\theta}_R}(o)) > \log(f_{\hat{\theta}_o}(z)) - \log(f_{\hat{\theta}_R}(z)) \\ & \Rightarrow \frac{f_{\hat{\theta}_o}(o)}{f_{\hat{\theta}_R}(o)} > \frac{f_{\hat{\theta}_o}(z)}{f_{\hat{\theta}_R}(z)} \end{aligned}$$

This is the exact form of the self-predicting update condition, which ensures incentive-compatibility of the Peer Truth Serum mechanism.

2.5 Practical Considerations

2.5.1 Influence Approximation

Trying to practically implement an Influence incentive mechanism imposes a host of challenges. The first is the computational cost of computing the Influence for an Agent. Specifically, given a set of n Agent reports Z_A , for every $z_i \in Z_A$, the center must compute $\hat{\theta}(Z_A^{-i})$, which requires retraining the model n times. Koh and Liang, 2017 present an approximation method where each sample in Z_A is assigned a weight of 1. First the change in model parameters $\hat{\theta}$ is approximated by a Taylor expansion around the training loss. Then a Taylor expansion is taken around the the validation loss with respect to the model parameters. We present this approximation formula with the first order terms of the expansions, as shown in Koh and Liang, 2017:

$$\mathcal{J}(z_i, Z_T, Z_V) = -\frac{1}{n} \nabla_{\theta} L_{\hat{\theta}(Z_T)}(Z_V)^{\top} H_{\theta}^{-1} \nabla_{\theta} L_{\hat{\theta}(Z_T)}(z_i)$$

where $H_{\theta} = \frac{1}{n} \sum_{z \in Z_T} \nabla_{\theta}^2 L_{\hat{\theta}(Z_T)}(z)$ is the Hessian. This approximation, however, has the undesirable property that the mean Influence of the training samples is 0 in many cases. From Definition 2.1.3, $\hat{\theta}$ is a solution to $\sum_{z \in Z_T} \nabla_{\theta} L_{\hat{\theta}(Z_T)}(z) = 0$, if it exists. As the Influence mechanism relies on the fact that the expected Influence of a truthful sample is positive for strictly risk-monotonic model families, this approximation is insufficient for providing proper incentives. We therefore include the 2nd order term in the Taylor expansions of both the training risk and the validation risk. Let $\partial\theta_i$ be the change in θ due to increasing the weighting a training point z_i , and let $H_{z_i} = \nabla_{\theta}^2 L_{\hat{\theta}(Z_T)}(z_i)$ be the Hessian computed only on z_i .

$$\partial\theta_i = \frac{1}{n} H_{\theta}^{-1} \nabla_{\theta} L_{\hat{\theta}(Z_T)}(z_i) + \frac{1}{n^2} H_{\theta}^{-1} H_{z_i} H_{\theta}^{-1} \nabla_{\theta} L_{\hat{\theta}(Z_T)}(z_i)$$

Then the second order approximation of the loss on a validation sample when computing the Influence is:

$$\mathcal{J}(z_i, Z_T, Z_V) = (\nabla_{\theta} L_{\hat{\theta}(Z_T)}(Z_V) + \frac{1}{2} H_{Z_V} \cdot \partial\theta_i) \cdot \partial\theta_i$$

Although this is significantly more computationally expensive than the first order approximation, since H_{z_i} must be computed for every z_i , it is still far easier than retraining a deep neural network. Additionally, the stochastic estimation techniques used for improving computation time suggested in Koh and Liang, 2017 can still be applied to computing the term $H_{\theta}^{-1} H_{z_i} H_{\theta}^{-1} \nabla_{\theta} L_{\hat{\theta}(Z_T)}(z_i)$.

Experimental Results

We run simulations to confirm the improved accuracy of the 2nd order approximation method and to demonstrate its computational efficiency. For the case of linear regression, computing

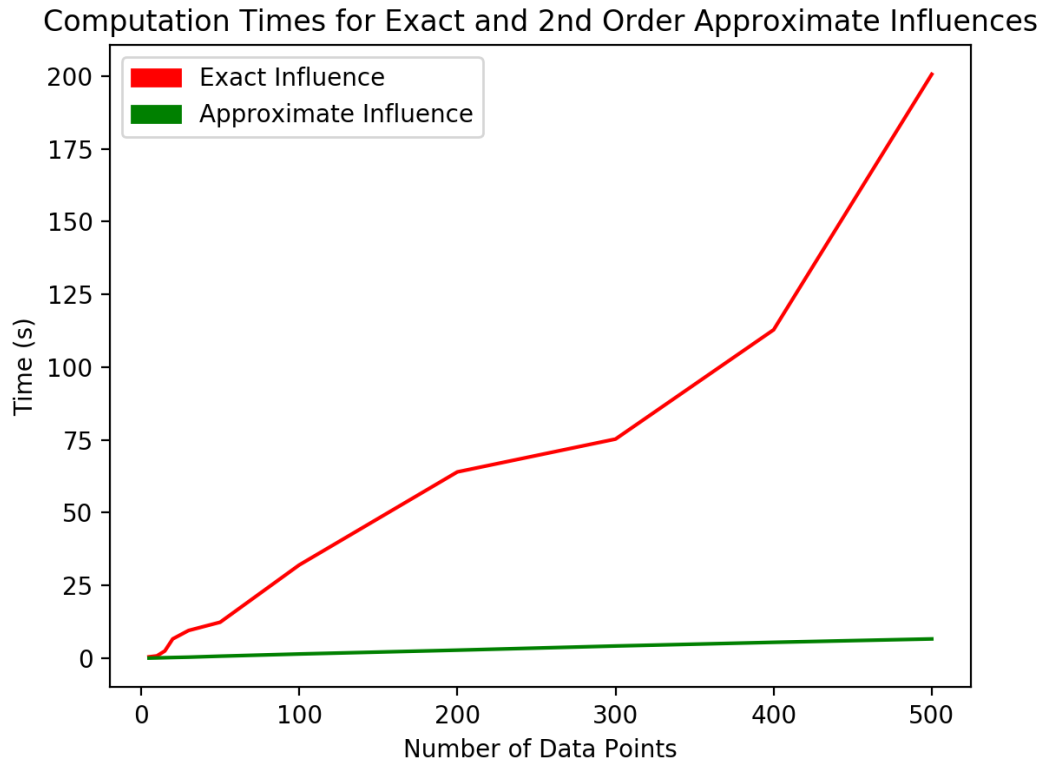


Figure 2.2: The exact influence is shown to become computationally prohibitive for logistic regression with only a moderate number of data points, while the computation time for the approximate influence increases relatively slowly.

the exact Influence for each data point can be computationally feasible, but with a high enough input dimension the approximation will be more computationally efficient. For a model that is learned via a gradient descent method, such as a logistic regressor, it is clear that the Influence approximation will provide significant improvements, as shown in Fig. 2.2. For all of our other simulations, we used the following datasets with a linear regression model:

- *Linear Generated*: We generate linear regression data as follows: pick an angle θ uniformly in $[-\pi/2, \pi/2]$, and a bias term from $N(0, 1)$. Using θ and the bias to determine a linear model, we uniformly sample $x \in [-1, 1]$ and determine ground truth y_{gt} values. We then add a noise variable drawn from $N(0, 1)$ to produce observations y .
- *Red Wine and White Wine* Cortez et al., 2009, *Air Quality* De Vito et al., 2008, *Communities and Crime (Crime)* Redmond and Baveja, 2002, *Parkinsons Telemonitoring (Parkinsons)* Tsanas et al., 2009: All data sets from the UCI database. We removed attributes that were non-predictive, redundant, or had a substantial number of missing values.

Using 1000 points for training and 200 for testing, we evaluated the exact Influence, 1st order approximation and 2nd order approximation for each data point, recording the L1 and L2 norms between the approximations and the exact Influence. We then evaluated the worst case and average improvement factors for the 2nd order approximation over the 1st order approximation (2nd order error / 1st order error).

The worst case improvement for the L1 norm was 0.410, with the average being 0.0789 (lower is better). The worst case improvement for the L2 norm was 0.482, with the average being 0.0821. This means that for both the L1 and the L2 norms, the 2nd order approximation was on average about 12 times as accurate as the 1st order approximation. We also report the means of the L1 and L2 norms between the 2nd order approximation and the exact Influence to demonstrate that it is indeed accurate: $1.16 * 10^{-3}$ for L1 and $2.45 * 10^{-5}$ for L2.

2.5.2 Sequential Data Gathering

In a practical implementation, data arrives sequentially. Prior, we assumed that the Influences would be computed over the entire set of reports once they have been collected. Ideally, the Center could compute the Influence and provide the payment immediately when a data point arrives. This has the advantage of allowing the Center to perform even more accurate budgeting. Suppose we have a sample set $\{z_i\}$ such that i indicates the time of arrival of each sample starting from 1 up to n . Then the sum of Influences is the overall change in risk on a fixed validation set Z_V .

$$\sum_{j=1}^n \mathcal{I}(z_j, Z_R \cup \{z_i\}_{i \leq j}, Z_V) = \sum_{j=1}^n \mathbf{L}_{\hat{\theta}_{Z_R \cup \{z_i\}_{i \leq j}}}(Z_V) - \mathbf{L}_{\hat{\theta}_{Z_R \cup \{z_i\}_{i \leq j-1}}}(Z_V) = \mathbf{L}_{\hat{\theta}_{Z_R}}(Z_V) - \mathbf{L}_{\hat{\theta}_{\{z_i\}}}(Z_V)$$

When the Center constructs the validation set from Agent reports, it may want to add more reports to the validation set over time. In this case, the sum of Influences may not be quite the same as the change in empirical validation risk, but if the reports are truthful, it will be the same in expectation. With the Center having a known value function $V(\mathbf{L})$, it does not need to rely on estimating the change in risk and Influence over time with the $\frac{1}{n}$ and $\frac{1}{n^2}$ models respectively. Setting $b = 0$ and having a fixed c in the reward function τ , the Center computes the budget necessary to achieve a final risk \mathbf{L}_f as $c(\mathbf{L}_0 - \mathbf{L}_f)$ where $\mathbf{L}_0 = \mathbf{L}_{\hat{\theta}_{Z_R}}(Z_V)$, achieving a gain in value of $V(\mathbf{L}_f)$. Therefore the overall utility is $V(\mathbf{L}_f) - c(\mathbf{L}_0 - \mathbf{L}_f)$. In order for the Center to profit, this quantity must be positive, so the Center computes the necessary value of c to achieve a profit with a model of final risk \mathbf{L}_f as $c < \frac{V(\mathbf{L}_f)}{(\mathbf{L}_0 - \mathbf{L}_f)}$.

2.5.3 M-Loss and M-Gain

Computing the Influence for each data point as it arrives can be computationally prohibitive, even using the Influence approximation. The computation time of the approximation, in terms of complexity, is dominated by computing H^{-1} , which must be computed every time

the model is updated. The Center can strike a balance between the two extremes by grouping the data into batches, such that H^{-1} is only computed once per batch. With respect to a single batch, the incentives are the same as the one-batch case, however, we must now consider how batch processing affects incentives with respect to the time of reporting. We observe that the 1st order Influence approximation has absolute error with respect to the exact Influence of $O(\frac{1}{n^2})$, and is 0 in expectation. Therefore, as mentioned previously, the exact Influence is $O(\frac{1}{n^2})$. With this, it is clear that batch processing incentivizes Agents to report as early as possible.

With batch processing, the Center has two choices in how to implement the mechanism. The Center may include the most current batch in updating the model and compute the Influence of each data point as though it were removed, or it could exclude the current batch and compute the Influence of each data point as though it were added to the rest. We call these two methods *M-Loss* and *M-Gain* respectively, as shown in Fig. 2.3. It is clear by construction that for a batch size of one, these two methods are equivalent, and the sum of Influences is equal to the overall change in risk. For the sake of computational efficiency, the Center will want to choose a batch size greater than 1. We note that M-Loss will underestimate the expected Influence in the 1-batch case because the Influence of points that arrive early in the batch won't be computed until the later points arrive. Symmetrically, M-Gain will overestimate.

If the Center has a fixed a priori budget B and wants to compute the reward scaling c , it is necessary to compensate for the underestimation or overestimation of M-Loss or M-Gain respectively. Fortunately, this is quite simple as the Center computes the change in risk over an entire batch. The Influence of a data point within a batch is simply multiplied by the change in risk and divided by the sum of Influences in the batch. With Influences normalized to match the change in risk, the Center can very easily apply the appropriate scaling factor and maintain accurate budgeting.

We wish to characterize the extent to which M-Loss and M-Gain underestimate and overestimate respectively, so the Center can compensate. We restrict ourselves to the case of linear regression, but the analysis can be extended to any model in which the optimal parameters have a closed-form solution.

Let us consider two probability distributions Φ_1 and Φ_2 . As before, we assume they describe an input-output relationship such that $\Phi(x, y) = q(x)p(y|x)$, and $q_1(x) = q_2(x)$. Distributions Φ_1 and Φ_2 determine, in expectation, models θ_1 and θ_2 respectively. Let us now define $\mathbf{L}_{i,j}$ as the expected risk of model θ_i evaluated on distribution Φ_j . Using the standard mean-squared-error loss function, we have that $\mathbf{L}_{i,j} = \mathbf{L}_{j,j} + r$ where $r = \mathbb{E}_{x \sim q}[(f_{\theta_i}(x) - f_{\theta_j}(x))^2]$. Now suppose we sample n_1 points from Φ_1 and n_2 points from Φ_2 to form our training set Z_T . Because the linear regression solution is linear with respect to y , and $q(x)$ is fixed, then Z_T determines in expectation a model θ_c such that $\mathbb{E}_q[f_{\theta_c}] = \frac{n_1 \mathbb{E}_q[f_{\theta_1}] + n_2 \mathbb{E}_q[f_{\theta_2}]}{n_1 + n_2}$. With this, let us consider the practical application where Φ_1 is the initialization distribution and Φ_2 is the distribution of reports from the Agents. Then when we evaluate the model, we are only concerned with the

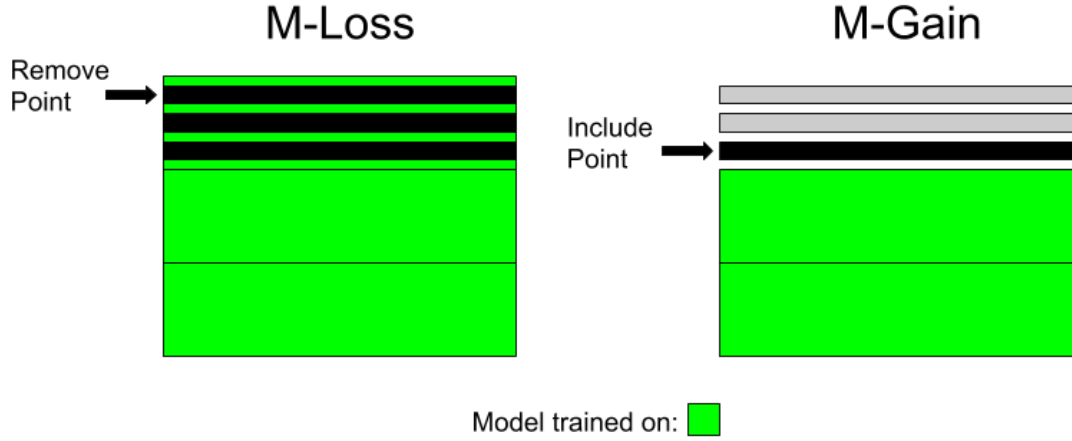


Figure 2.3: M-Loss is trained on all points in current batch, with Influence computed by removing a point. M-Gain is trained on all prior batches, with Influence computed by adding a point from current batch.

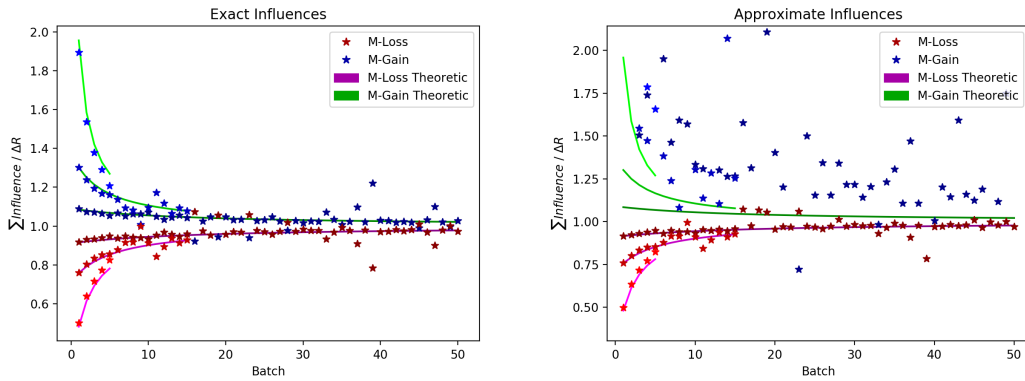


Figure 2.4: Ratio between Sum of Influences and Change in Loss with respect to batch size.

error of the mixed model θ_c evaluated on $\bar{\Phi}_2$:

$$\mathbf{L}_{c,2} = \mathbf{L}_{2,2} + \left(\frac{n_1}{n_1 + n_2} \right)^2 r$$

To simplify, we fix $n_1 = p$ as the number of points used for initialization and we let n_2 vary as x . Then we have our expected empirical risk in terms of x :

$$\mathbf{L}(x) = \frac{p^2 r}{(p + x)^2} + R_{2,2}$$

We can approximate the Influence of a data point arriving after x data points as the negative

of the derivative of the risk:

$$-\frac{\partial \mathbf{L}}{\partial x} = \frac{2p^2 r}{(p+x)^3}$$

Now we consider batch size b . We can compute the expected overall change in loss of some arbitrary batch k , with k index starting at 1.

$$\Delta \mathbf{L}_b(k) = \mathbf{L}((k-1)b) - \mathbf{L}(kb) = \frac{bp^2 r(2p + (k-1)b)^2}{(p + (k-1)b)^2(p + kb)^2}$$

Now we consider the sum of Influences of points in batch k for M-Loss and M-Gain.

$$S_{\text{loss},b}(k) = -b \frac{\partial \mathbf{L}}{\partial x} \Big|_{kb} = \frac{2bp^2 r}{(p + kb)^3}$$

$$S_{\text{gain},b}(k) = -b \frac{\partial \mathbf{L}}{\partial x} \Big|_{(k-1)b} = \frac{2bp^2 r}{(p + (k-1)b)^3},$$

Comparing these to the change in risk, we get the following ratios:

$$D_{\text{loss},b}(k) = \frac{S_{\text{loss},b}(k)}{\Delta \mathbf{L}_b(k)} = \frac{2(p + (k-1)b)^2}{(p + kb)(2p + (2k-1)b)}$$

$$D_{\text{gain},b}(k) = \frac{S_{\text{gain},b}(k)}{\Delta \mathbf{L}_b(k)} = \frac{2(p + kb)^2}{(p + (k-1)b)(2p + (2k-1)b)}$$

By computing these values, the Center can pick an arbitrary batch size and divide the Influence scores by these formulas such that the expected sum of Influences is equal to the overall change in risk, as in the case of batch size 1. We note that these formula have constant growth rate with respect to the number of points $p + kb$ and they asymptotically approach the constant function $D_b(k) = 1$. Therefore, dividing the Influence scores by these formulas will not affect the incentive for early reporting.

We note that this analytic method only applies to linear regression, and that it can be reasonably extended to learners with closed-form solutions for the optimal parameters. However, the Center can approximate this method by using the observed Influences and change in risk across a batch and re-scaling with the ratio of these two empirical values, rather than the a-priori expected ratio.

Experimental Results

We present experimental results to demonstrate the validity of the re-scaling formula in real scenarios. We ran simulations, using the same datasets as in Section 2.5.1, to estimate the effect of the batch size on the ratios D_{loss} and D_{gain} . We ran each simulation with 1500 total training points with a varying batch size. Given a fixed batch size, we ran 10 trials for every dataset and aggregated them to form a more general estimate of S_{loss} , S_{gain} , and ΔR . We then

took the ratios of these aggregates and compared against our theoretical results for D_{loss} and D_{gain} in Fig. 2.4. We ran this same simulation with different numbers of initial points 20, 100, 200, and 500. We have chosen only to show the case with 500 initial points, although the other simulations show the same relationship. Each line represents a different batch size. We have chosen to plot batch sizes 30, 100, and 300 for ease of visualization.

2.6 Summary

First, we present the concept of using Influence as a scoring function for a Peer Prediction incentive mechanism. We first demonstrate that such an incentive mechanism is incentive-compatible when Agents have uninformed prior beliefs, and update their beliefs such that the underlying data is representative. We find that the mechanism is DSIC for a single-trained mechanism and BNIC for a mixed-trained mechanism.

We then show how the mechanism allows for a priori budgeting for the Center. Depending on the Center's beliefs about the expected effort levels of the Agents, the Center can know ahead of time if the crowdsourcing is economically viable. We show that the incentives are also robust to a certain degree of corruption, i.e. if some proportion of Agents do not report truthfully.

The Influence mechanism is shown to be an extension, in some ways a generalization, of the Peer Truth Serum mechanism for discrete valued reports. In this way, Influence is in line with prior work on Peer Consistency.

Finally, we address some practical implementation concerns. We present an approximation method that is theoretically sound for the purpose of the incentive mechanism. We also introduce the possibility of sequential data gathering with batch influence computation to strike a balance between computational complexity and budget accuracy.

3 Influence Filtering

3.1 Introduction

In the previous chapter, we proposed using the classical statistical notion of *Influence* as an incentive mechanism. We constructed such a mechanism and proved that it's incentive-compatible, with a truthful Dominant Strategy Equilibrium solution when the validation set is composed of truthful samples, and a truthful Bayes-Nash Equilibrium solution when the validation set is composed of randomly selected reports. We also addressed some practical considerations for implementing such a mechanism, including an Influence approximation method and a batch processing method. In this chapter, we address a much more difficult practical consideration: what happens when Agents don't act perfectly rationally? Influence relies on trusted validation samples in order to accurately reflect the quality of the training samples on model accuracy with respect to the distribution which generates the trusted validation samples, which we refer to as the *target* distribution. In reality, the validation set may contain corrupted samples, such as mislabeled samples or heuristic reports from irrational Agents. We model this as a probabilistic combination of two distributions: the *target* distribution and the *corrupted* distribution. Samples from the target distribution are referred to as *accurate* while samples from the corrupted distribution are simply referred to as *corrupted*. The corrupted distribution could itself be a combination of many different distributions from different sources, but we absorb them into one distribution because we assume no prior information about how this distribution is composed. We observe the surprising behavior that, even with a small amount of corruption in the validation samples, a single corrupted samples in training will have an extremely high Influence on the corrupted samples in validation, outweighing any small negative Influence it might have on all the accurate samples in validation. This is because of the statistical properties of Influence, which decreases as $O(\frac{1}{n^2})$ with the number of training points N as shown in the previous chapter. With only one corrupted sample, one can think of the accurate samples as being further along in this curve and thus achieving a lower expected Influence.

We have yet to see other work examining the properties of Influence when the validation

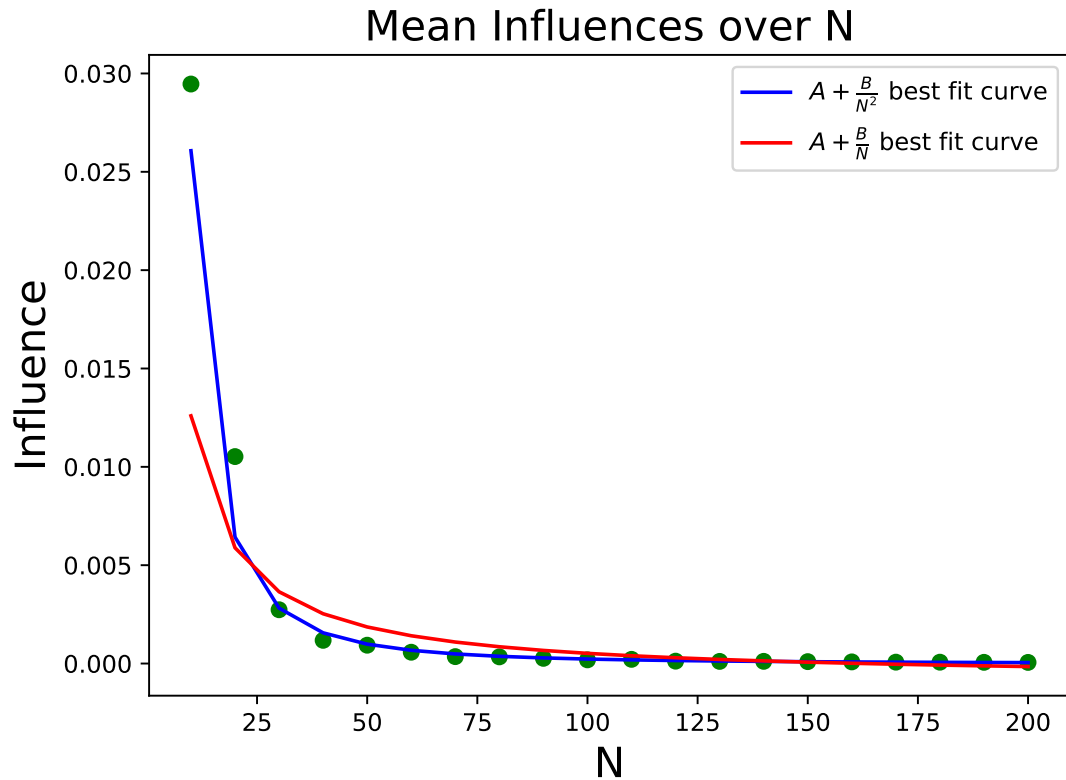


Figure 3.1: Mean Influences over number of data points. Growth rate of Influences matches $O(\frac{1}{N^2})$.

samples are *untrusted*. Intuitively, if the validation samples are drawn from a completely different distribution from the target distribution, a scoring function may not give the Center any relevant information about the training sample in question. We examine two intimately related practical use cases of Influence as a scoring function: incentives and data filtering, which can be applied to many practical settings including federated learning. We assume that the Agents providing samples are self-interested and must be compensated for their reports. The Center does this by constructing a reward function that induces the desired Agent behavior, which is to contribute accurate instances of the target distribution, known as *truthful reporting*. According to the game theoretic principle of Agent rationality, in order for an incentive mechanism to induce the desired behavior, such behavior must maximize Agent utility functions in expectation. By extension, it is necessary and sufficient that the scoring function is maximized in expectation on accurate sample reports. For data filtering, on the other hand, the Center must use the scoring function to establish a partial ordering on the samples, representing their relative values in contributing to model accuracy. So not only do we require that accurate samples have the maximal score in expectation, we require that accurate samples be ordered above corrupted samples.

3.1.1 Our Approach

The Influence, as defined in the previous chapter, quantifies the effect of a single training sample on the parameters of a learning model, but Influence is not an absolute quality metric, rather it is relative. The Influence of a training samples depends on all the other training samples that are present. If one wished to track all the cross-dependencies in the training samples, one would be forced to examine the Influences of all possible coalitions of training samples, which is equivalent to computing the Shapley value of each report. We show that, while the Shapley Value is vastly more computationally intensive, it still suffers from the same theoretical disadvantage that it assumes the presence of trusted validation samples. In the case of regression models, analyzing the Influence allows us to consider decomposing a model into components constructed entirely from either accurate or corrupted samples, since the Influence describes a tractable perturbation of the combination of these model components.

Using this method, we're able to compute conditions under which accurate samples can be partially distinguished from corrupted samples according to their expected Influences. Let us define p to be the proportion of accurate samples in training, the other $(1 - p)$ being proportion being corrupted samples. Similarly, let us define q to be the proportion of accurate samples in validation. We find that, given infinite samples, if and only if $p < q$, the average Influence of an accurate sample will be greater than that of a corrupted sample, and vice versa when $p > q$. In the case of finite samples, these conditions are slightly modified according to the means and variances of the model predictions produced by samples of the target and corrupted distributions. With this modification, even if $p > q$ by some small amount, the accurate samples might still have a greater expected Influence if the corrupted distribution produces models with higher variance than models produced by the target distribution.

These conditions describe a class of *equilibria* when using Influence as an incentive mechanism: there are *Dominant Strategy Equilibria* symmetric about perturbations of every $p = q$ value, which drive the system to converge towards a mixed *Bayes-Nash Equilibrium* at $p = q$. They also naturally lead to the construction of a theoretically sound *Probabilistic Filtering* scheme. The probabilistic filtering requires a choice of CDF $P_Z(z_i)$ for $z_i \in Z$. We will show that a uniform CDF is the most natural choice, which we call *Uniform Probabilistic Filtering* (UPF), with protocol described in Figure 3. The choice of α and β can be tuned as hyper-parameters to adjust the sensitivity of the filter, but the intuitive choices are $\alpha = \min_i \mathcal{I}(z_i, Z)$ and $\beta = \max_i \mathcal{I}(z_i, Z) - \min_i \mathcal{I}(z_i, Z)$. We prove that, in the untrusted validation regime, UPF is expected to improve the true model accuracy, i.e. the accuracy evaluated only on the target distribution, up to a certain limit. We also consider more intuitive deterministic filtering schemes, the first being a naive filtering protocol, *Threshold Influence Filtering* (TIF), which removes all data points with Influence values below a threshold, as described in Figure 1. We refine such a scheme to iteratively remove minimal Influence data points with *Iterative Minimal Influence Filtering* (IMIF); protocol described in Figure 2. We prove that, in a trusted validation regime, IMIF with a threshold of 0 is guaranteed to improve model accuracy. However, using only the analysis we present, one cannot make any guarantees about the filtering

performance of such deterministic filters in the untrusted validation regime.

3.1.2 Model

We consider the same model as in the previous chapter. We repeat definitions for ease of reading:

Definition 3.1.1 (Model Family). Let f be a *model family* parameterized by θ such that $f_\theta(x) = \hat{y}$ where \hat{y} is the estimate of the representative value of $\Phi_{y|x}$, otherwise known as the *prediction*.

Definition 3.1.2 (True Risk and Optimizer). Let $L(y, \hat{y})$ be a non-negative loss function. Let Φ be the true distribution of the random variable $z = (x, y)$, with Ω being the fundamental set. The *true risk* is given by $\mathbf{L}_\theta^* = \int_\Omega L(y, f_\theta(x)) d\Phi(z)$. We will often write $L(y, f_\theta(x))$ as $L_\theta(z)$. Then the *true optimizer* is given by $\theta^* = \operatorname{argmin}_\theta \mathbf{L}_\theta^*$.

The true optimizer may not be unique. It is sometimes possible to not only minimize over the joint distribution Φ , but to minimize over all the conditional distributions $\Phi_{y|x}$.

Definition 3.1.3 (Absolute Optimizer and Realizability). Let $L(y, \hat{y})$ be a non-negative loss function. Let $\Phi_{y|x}$ be the true conditional distribution of y given x , and let Φ_x be the true marginal distribution of x , with Ω_x and Ω_y being the fundamental sets in the x and y coordinates. The *absolute optimizer* is $f^* : \Omega_x \rightarrow \Omega_y$ where $\forall x$ in the support of Φ_x , $f^*(x) = \inf_{\hat{y}} \int_{\Omega_y} L(y, \hat{y}) d\Phi_{y|x}(y)$. We say f^* is *realizable* if $\exists \theta^*$ such that $f_{\theta^*} = f^*$. We say it is *uniquely realizable* if θ^* is unique.

Definition 3.1.4 (Empirical Risk and Optimizer). Let $Z = \{z_i\}_{i \in [1, n]}$ be a set of n input-label pairs. The *empirical risk* is given by $\mathbf{L}_\theta(Z) = \frac{1}{n} \sum_{i=1}^n L_\theta(z_i)$. Then the *empirical risk optimizer* is given by $\hat{\theta}(Z) = \operatorname{argmin}_\theta \mathbf{L}_\theta(Z)$.

Definition 3.1.5 (Influence). Given a model family f parameterized by θ to minimize a loss function L , and given $z_i \in Z_T$ and $z_j \in Z_V$, the Influence of z_i on z_j is:

$$\mathcal{I}(z_i, Z_T, z_j) = L_{\hat{\theta}(Z_T^{-i})}(z_j) - L_{\hat{\theta}(Z_T)}(z_j) \quad (3.1)$$

We often consider the average Influence over the validation set, written as:

$$\mathcal{I}(z_i, Z_T, Z_V) = \mathbf{L}_{\hat{\theta}(Z_T^{-i})}(Z_V) - \mathbf{L}_{\hat{\theta}(Z_T)}(Z_V) \quad (3.2)$$

We will often omit Z_T or Z_V from the argument when they are clear from context.

Corruption Model

In many circumstances, the Center is assumed to have a set of data points that it knows to be accurate samples from Φ , making it relatively easy to perform meaningful evaluation metrics,

like Influence, on samples in the training set. In our setting, we relax this assumption. If the Center is crowdsourcing data or has acquired data from an untrusted source, there might be a subset of samples which are *corrupted*. This corruption might come from a variety of sources, such as Agents injecting noise for privacy reasons, Agents with faulty sampling procedures, or even malicious Agents. The subset of corrupted points might even come from a combination of these sources, but we can fold all the sources together and generally state that the corruption comes from some corrupted distribution $\bar{\Phi}_C$, as opposed to the accurate distribution which we will write as $\bar{\Phi}_A$. We assume that the marginal distributions of x are the same, so that the corruption is in the labels y : $\bar{\Phi}_{C,x} = \bar{\Phi}_{A,x}$.

The Center has collected a training set $Z_T = Z_{T,A} \cup Z_{T,C}$ where $Z_{T,A}$ are sampled i.i.d. from $\bar{\Phi}_A$ and $Z_{T,C}$ are sampled i.i.d. from $\bar{\Phi}_C$. We say the *mixing proportion* is $p = \frac{|Z_{T,A}|}{|Z_T|}$. Similarly, the Center has collected a validation set $Z_V = Z_{V,A} \cup Z_{V,C}$ with mixing proportion $q = \frac{|Z_{V,A}|}{|Z_V|}$. The Center may potentially know these mixing proportions, but it does not know which samples are from which distribution. It wants to perform some *filtering*, i.e. removing points from the training set, in order to improve the "model accuracy". We use the term "model accuracy" loosely, because the Center may be able to lower the true risk of the empirical risk minimizer by removing more accurate samples than corrupted samples, but intuitively this would be a pathological case. As a proxy, we say that the "model accuracy" is improved on average when the mixing proportion p of the training set increases. So then the goal of filtering is to remove proportionately more corrupted than accurate data samples.

3.1.3 Shapley value

We consider the Shapley value of Influences as an alternative to the marginal Influence (Shapley et al., 1953). We provide the definition of the Shapley value:

Definition 3.1.6 (Shapley Value). Let N be the number of players in a game, v the characteristic function of the game (how much utility each coalition generates), Σ the set of all possible orderings of players, and C_σ the coalition formed by players according to the ordering σ . Then the *Shapley value* $\phi_i(v)$ for player i is given by:

$$\phi_i(v) = \frac{1}{N!} \sum_{\sigma \in \Sigma} (v(C_\sigma \cup \{i\}) - v(C_\sigma)),$$

In our case, the quantity $v(C_\sigma \cup \{i\}) - v(C_\sigma)$ represents the Influence of player i in coalition C_σ . So the Shapley value of Influences is the average Influence of a report r among all possible subsets of reports containing r in the training set. The Shapley value has nice intuitive properties in the machine learning context. The value of a particular sample might vary greatly depending on what other samples are present in the training set. The Influence may not capture these group-dependencies in an adequate way.

Although the Shapley value captures the group-dependencies of samples, we are not con-

cerned with how the scoring function affects a particular sample, rather we are concerned with how the scoring function affects filtering performance in expectation. Let Φ_A be the accurate distribution and Φ_C be the corrupted distribution. We ask if the Shapely value affects the expected Influences of the two groups in the training set, i.e. does the Shapely value introduce a bias as compared to the Influence? Symmetry dictates that it cannot introduce a bias if the two distributions are equally represented, but we conduct empirical analysis to see if the Shapely value tends to favor a group when they are unbalanced.

We conduct experiments with 1000 total training points, 750 from group one and 250 from group two. "Artificial Linear" data is generated by picking an slope uniformly at random, then sampling points with mean 0 variance 1 additive Gaussian noise. Details for the other data sets can be found in Section 3.4. Influences are evaluated against 250 validation points from each group and the average is taken. The exact Shapley value is prohibitively expensive to compute, so we approximate it as follows: we iterate over a sampling of coalition sizes and sum over them weighted according to their binomial coefficients. For each coalition size, we iterate over each possible linear combination of groups, weighted according to its binomial coefficient. For each of these, we conduct 5 trials randomly selecting points from each group. We then compare these values to expected Influences at the mean of the binomial distribution.

Table 3.1: Shapley value vs. Influence

Group One			Group Two		
Name	Shapley value	Influence	Name	Shapley value	Influence
Artificial Linear	9.69772e-5	9.46879e-5	Artificial Linear	-2.56188e-4	-2.51391e-4
Red Wine	8.41934e-3	8.38264e-3	Artificial Linear	-2.38488e-2	-2.39453e-2
Air Quality	4.61109e-2	4.67155e-2	Artificial Linear	-1.31635e-1	-1.34375e-1
Parkinsons	2.44179e-1	2.41919e-1	Random Binary	-5.45787e-1	-5.41967e-1
Crime	1.41606e-4	1.41072e-4	Artificial Linear	-1.39161e-4	-1.31191e-4
Artificial Binary	1.73201e-6	1.77475e-6	Random Binary	-4.40148e-6	-4.55086e-6

In Table 3.1 we observe that the Shapley value is indeed very close to the expected Influence at the mean for both groups. The L1 error is on the order of one in a thousand, which could easily be due to the sub-sampling we perform for the Shapley value approximation. In addition, there is no discernible bias. This suggests that, while the Shapley value may have some desirable properties over Influence in terms of individual data valuation, it has no clear advantage when it comes to filtering.

3.2 Influence Analysis

In order for the Center to successfully perform filtering using the Influence score, it must observe some differentiation in the distributions of Influences of accurate and corrupted data samples. We begin by examining the Influences in an infinite sample regime.

3.2.1 Infinite Sample Analysis

Analyzing the distribution of Influences with no assumptions about the loss function is very unconstrained. The loss function can impose arbitrary relationships between points. In addition, we require that the corrupted and accurate distributions are different in terms of optimal modeling. We call this guarantee *discernibility*:

Definition 3.2.1 (Discernibility). Let Φ_C and Φ_A be the corrupt and accurate distributions with a shared marginal distribution Φ_x . Let $L(y, \hat{y})$ be a non-negative loss function, producing absolute optimizers f_A^* and f_C^* for Φ_C and Φ_A respectively. We say Φ_C and Φ_A are *discernible by L* if $\Phi_x(\{x : f_C^*(x) = f_A^*(x)\}) < 1$.

To further simplify the analysis, we assume that the loss function is the mean squared error so the learning model is a least-squares regressor. Then the absolute optimizer is simply the expected values of the conditionals.

Proposition 3.2.2. Let $L(y, \hat{y}) = (y - \hat{y})^2$. Then $f^*(x) = \mathbb{E}_{\Phi_{y|x}}[y]$.

This follows directly from the fact that the expected value of a distribution is the variance minimizer. Finally, we must make an assumption about the degree of stochasticity in the model family. There are potential pathological examples of model families and distributions for which the optimal model can have arbitrary variability in its predictions, even in the limit of infinite samples. This is only the case if the absolute optimal model is not realizable, but this is a stronger assumption than necessary. Instead, we assert that the model family must have *limited stochasticity* with respect to the training distribution:

Definition 3.2.3 (Limited Stochasticity). Given a training distribution Φ_T and a model family f , let $Z_n = \{z_i\}_{i \in [1, n]}$ be a set of n i.i.d. random variables sampled from Φ_T . We call $\Phi_{T, n}$ the distribution of Z_n . The model family has *limited stochasticity* with respect to Φ_T if $\forall \epsilon > 0 : \lim_{n \rightarrow \infty} \Phi_{T, n}(\{Z_n : (\max_x |f_{\hat{\theta}(Z_n)}(x) - f_{\theta^*}(x)|) > \epsilon\}) = 0$.

In other words, as the number of samples goes to infinity, the probability of getting a set of samples which causes a deviation from the optimal predictions goes to 0. With these assumptions, we can then prove the following statement about the expected Influence values of corrupt and accurate data samples:

Theorem 3.2.4. Let $L(y, \hat{y}) = (y - \hat{y})^2$. Let Φ_C and Φ_A be the corrupt and accurate distributions such that they are discernible by L . Let the training set be mixed by p and the testing set be mixed by q , such that the training distribution is Φ_p and the validation distribution is Φ_q . Suppose the model family has limited stochasticity with respect to Φ_p . Then $p < q$ if and only if the expected Influence of a training point $z_A \sim \Phi_A$ is greater than the expected Influence of a training point

$z_C \sim \Phi_C$:

$$\begin{aligned}
& \mathbb{E}_{z_A \sim \Phi_A} [\mathbb{E}_{z_V \sim \Phi_q} [\mathcal{J}(z_A, Z_T, z_V)]] \\
& > \mathbb{E}_{z_C \sim \Phi_C} [\mathbb{E}_{z_V \sim \Phi_q} [\mathcal{J}(z_C, Z_T, z_V)]] \\
& \iff p < q
\end{aligned} \tag{3.3}$$

Proof. We consider the optimal models produced by the accurate and corrupt distributions individually. Let $f_A^*(x) = \mathbb{E}_{y \sim \Psi_{A,(y|x)}} [y]$ and $f_C^*(x) = \mathbb{E}_{y \sim \Psi_{C,(y|x)}} [y]$. Consider the mixed model with mixture proportion p at a x : $f_p^*(x) = p f_A^*(x) + (1-p) f_C^*(x)$.

But f_p^* may not be realizable, so there will be a residual term that we cannot determine a priori. We model this residual term as a random variable δ_x over y , so $f_{\theta^*}(x) = f_p^*(x) + \delta_x$. We first observe that $\mathbb{E}_{\Psi_{T,(y|x)}} [\delta_x] = 0$ since θ^* is a variance minimizer. We then compute the expected risk $\mathbf{L}_{\theta^*}^V$ on the validation set with mixture proportion $q > p$. Let :

$$\begin{aligned}
\mathbf{L}_{\theta^*}^V &= \mathbb{E}_{x \sim \Psi_x} [\mathbb{E}_{y \sim \Psi_{V,(y|x)}} [\mathbb{E}_{\Psi_{T,(y|x)}} [L(y, f_{\theta^*}(x))]]] \\
&= \mathbb{E}_{\Psi_x} [\mathbb{E}_{\Psi_{V,(y|x)}} [L(y, f_p^*(x)) + \mathbb{E}_{\Psi_{T,(y|x)}} [\delta_x^2]]] \\
&= \mathbb{E}_{\Psi_x} [\mathbb{E}_{\Psi_{V,(y|x)}} [L(y, f_p^*(x))] + \mathbb{E}_{x \sim \Psi_x} [\mathbb{E}_{\Psi_{T,(y|x)}} [\delta_x^2]]] \\
&= \mathbb{E}_{\Psi_x} [(q \mathbb{E}_{\Psi_{A,(y|x)}} + (1-q) \mathbb{E}_{\Psi_{C,(y|x)}}) [L(y, f_p^*(x))] + \Delta]
\end{aligned}$$

Differentiating with respect to p , we obtain:

$$\frac{d}{dp} \mathbf{L}_{\theta^*}^V = 2(p-q) \mathbb{E}_{x \sim \Psi_x} [L(y, f_p^*(x))]$$

From the discernibility of Φ_A and Φ_C by L , the expectation is strictly positive, so $\frac{d}{dp} \mathbf{L}_{\theta^*}^V = \alpha(p-q)$ with $\alpha > 0$.

Finally, the limited stochasticity assumption yields $\lim_{|Z_T| \rightarrow \infty} \mathbb{E}_{z_T \sim \Phi_A} [\mathbb{E}_{z_V \sim \Phi_{V,q}} [\mathcal{J}(z_T, Z_T, z_V)]] = -\frac{d}{dp} \mathbf{L}_{\theta^*}^V$, which is positive for $p < q$.

By symmetry, we find that: $\lim_{|Z_T| \rightarrow \infty} \mathbb{E}_{z_T \sim \Phi_C} [\mathbb{E}_{z_V \sim \Phi_{V,q}} [\mathcal{J}(z_T, Z_T, z_V)]] < 0$. \square

An obvious corollary is that the expected Influences are equal when $p = q$.

3.2.2 Finite Sample Analysis

The infinite sample analysis reveals an intuitive, but disheartening result: filtering via the expected Influences cannot perform better than the proportion of corruption in the validation set. We will elaborate on what precisely it means to filter via expected Influences later. The analysis relies on the fact that in the infinite sample regime, the model converges to the expected value, but in reality, with noisy data and a finite number of training points, the

Center will never actually acquire the expected value of the mixed model. It will acquire some perturbation of the expected model. A new sample added to the training set from either the accurate or corrupt distribution will tend to drive this perturbed model towards the limiting expected value model at different rates. Since the expected value model is the risk minimizer, we'd expect the sample which drives the model towards this limiting model faster to be the one with the higher Influence.

Analyzing this effect will require understanding the distributions of the perturbed models, which we call the posterior distributions. We will see that, unlike in the infinite sample analysis, this effect depends on higher moments of these posterior distributions. The analysis becomes prohibitively complicated when considering arbitrary posterior distributions, so we will make the simplifying assumption that the posteriors are Gaussian distributions:

Definition 3.2.5 (Gaussian Posterior). Let $Z_n = \{z_i\}_{i \in [1, n]}$ sampled i.i.d. from some distribution Φ . Denote the distribution of Z_n as Φ_n . Then given some model family f , the empirical optimizer $\hat{\theta}(Z_n)$ is a random variable according to Φ_n , so $f_{\hat{\theta}(Z_n)}(x)$ is also a random variable with some distribution. We say the model family f has *Gaussian posterior* if $\forall x$ in the support of Φ_x , $f_{\hat{\theta}(Z_n)}(x) \sim N(\mu(x), \sigma(x)^2)$ where $N(\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2 .

To demonstrate the effect of the higher moments, in this case simply the variance, we will assume $p = q$ so that the expected Influences would be equal in the limit of infinite samples.

Definition 3.2.6 (Mixture Component Distributions). Suppose $Z_T = \{z_i\}_{i \in [1, n]}$ with mixture proportion p . Let us denote $Z_{T,A}$ as the set of points from Φ_A so the distribution of $Z_{T,A}$ is $\Phi_{A, pn}$. We similarly denote $Z_{T,C}$ distributed according to $\Phi_{C, (1-p)n}$.

Theorem 3.2.7. Let the training set $Z_{T,n,p} = Z_{T,A} \cup Z_{T,C} = \{z_i\}_{i \in [1, pn]} \cup \{z_j\}_{j \in [1, (1-p)n]}$ with $z_i \sim \Phi_A$ and $z_j \sim \Phi_C$. Let the validation set be similarly mixed according to $q = p$. Suppose the model family has Gaussian posterior with respect to both $Z_{T,A}$ and $Z_{T,C}$, with respective distributions $N(\mu_A(x), \sigma_A(x)^2)$ and $N(\mu_C(x), \sigma_C(x)^2)$. Then the expected Influence of a training sample $z_A \sim \Phi_A$ is greater than the expected Influence of a training sample $z_C \sim \Phi_C$ if and only if $\mathbb{E}_{x \sim \Phi_x} [(\sigma_C^2(x) - \sigma_A^2(x)) + (2p - 1)(\mu_C(x) - \mu_A(x))^2] > 0$:

$$\begin{aligned} & \mathbb{E}_{z_A \sim \Phi_A} [\mathbb{E}_{z_V \sim \Phi_V} [\mathcal{I}(z_A, Z_{T,n,p}, z_V)]] \\ & > \mathbb{E}_{z_C \sim \Phi_C} [\mathbb{E}_{z_V \sim \Phi_V} [\mathcal{I}(z_C, Z_{T,n,p}, z_V)]] \\ \iff & \mathbb{E}_{x \sim \Phi_x} [(\sigma_C^2(x) - \sigma_A^2(x)) + (2p - 1)(\mu_C(x) - \mu_A(x))^2] > 0 \end{aligned} \quad (3.4)$$

so that in the limit of infinite validation samples, we can say that each sample is distributed according to $\Phi_q = q\Phi_A + (1 - q)\Phi_C$.

Proof. Note that $f_{\hat{\theta}(Z_{T,A})}(x)$ can be written as $\mu_A(x) + \sigma_A(x)r_A(x)$ and $f_{\hat{\theta}(Z_{T,C})}(x)$ can be written as $\mu_C(x) + \sigma_C(x)r_C(x)$ where r_A and r_C are independent normal random variables $N(0, 1)$. Sup-

pose the center observes some mixed model value $f_{\hat{\theta}(Z_{T,n,p})}(x) = p f_{\hat{\theta}(Z_{T,A})}(x) + (1-p) f_{\hat{\theta}(Z_{T,C})}(x)$. Then we can decompose this value into its constituents from the two partial models. Let $L(x)^2 = p^2 \sigma_A^2(x) + (1-p)^2 \sigma_C^2(x)$. Let $F_A(x)$ be the random variable distributed according to $f_{\hat{\theta}(Z_{T,A})}(x)$ conditioned on $f_{\hat{\theta}(Z_{T,n,p})}(x)$, and let $F_C(x)$ be the random variable distributed according to $f_{\hat{\theta}(Z_{T,C})}(x)$ conditioned on $f_{\hat{\theta}(Z_{T,n,p})}(x)$. For now we omit the argument x for ease of notation:

$$\begin{aligned} F_A &\sim N(\hat{\mu}_A, \hat{\sigma}_A^2) \\ F_C &\sim N(\hat{\mu}_C, \hat{\sigma}_C^2) \end{aligned}$$

where

$$\begin{aligned} \hat{\mu}_A &= \frac{p\sigma_A^2(f_{\hat{\theta}(Z_{T,n,p})} - (1-p)\mu_C)}{L^2} \\ \hat{\sigma}_A^2 &= \frac{(1-p)^2\sigma_A^2\sigma_C^2}{L^2} \\ \hat{\mu}_C &= \frac{p^2\sigma_A^2\mu_C + (1-p)\sigma_C^2 f_{\hat{\theta}(Z_{T,n,p})}}{L^2} \\ \hat{\sigma}_C^2 &= \frac{p^2\sigma_A^2\sigma_C^2}{L^2} \end{aligned}$$

Critically, we observe that the conditional decompositions are still Gaussian random variables. We now compute the expected values. Define $\bar{F}_A = \mathbb{E}[F_A]$ and $\bar{F}_C = \mathbb{E}[F_C]$. We also define $\bar{F}_p = \mathbb{E}[f_{\hat{\theta}(Z_{T,n,p})}] = p\mu_A + (1-p)\mu_C$. Note that $f_{\hat{\theta}(Z_{T,n,p})}$ is a linear combination of r_A and r_C , so we can find a new basis such that $f_{\hat{\theta}(Z_{T,n,p})}$ is aligned with a basis vector.

$$\begin{bmatrix} w_{\parallel} \\ w_{\perp} \end{bmatrix} = \frac{1}{L} \begin{bmatrix} p\sigma_A & (1-p)\sigma_C \\ -(1-p)\sigma_C & p\sigma_A \end{bmatrix} \begin{bmatrix} r_A \\ r_C \end{bmatrix}$$

Inverting, we have:

$$\begin{bmatrix} r_A \\ r_C \end{bmatrix} = \frac{1}{L} \begin{bmatrix} p\sigma_A & -(1-p)\sigma_C \\ (1-p)\sigma_C & p\sigma_A \end{bmatrix} \begin{bmatrix} w_{\parallel} \\ w_{\perp} \end{bmatrix}$$

$f_{\hat{\theta}(Z_{T,n,p})}$ is some deviation from the mean \bar{F}_p , and we know that as we add more points, this model will converge to the mean. So we wish to know which of the two partial models contribute more towards pushing $f_{\hat{\theta}(Z_{T,n,p})}$ towards the mean. Since the loss function is the squared error, we compute the distribution of the squared error from the mean $(f_{\hat{\theta}(Z_{T,A})} - \bar{F}_p)^2$

conditioned on the mixed model $f_{\hat{\theta}(Z_{T,n,p})}$.

$$\begin{aligned} (f_{\hat{\theta}(Z_{T,A})} - \bar{F}_p)^2 &= (\sigma_A r_A - (\bar{F}_p - \mu_A))^2 \\ &= \left(\frac{\sigma_A}{L}(p\sigma_A w_{\parallel} - (1-p)\sigma_C w_{\perp}) - (\bar{F}_p - \mu_A)\right)^2 \\ &= \left(\frac{\sigma_A}{L}(-(1-p)\sigma_C w_{\perp}) - (\bar{F}_p - \mu_A - \frac{\sigma_A}{L}p\sigma_A w_{\parallel})\right)^2 \end{aligned}$$

Define $\mathcal{E}_A = (f_{\hat{\theta}(Z_{T,A})} - \bar{F}_p)^2 | f_{\hat{\theta}(Z_{T,n,p})}$, and define \mathcal{E}_C similarly. Conditioning on $f_{\hat{\theta}(Z_{T,n,p})}$ is equivalent to conditioning on w_{\parallel} , so this yields:

$$\begin{aligned} \mathcal{E}_A &= \frac{((1-p)\sigma_A\sigma_C)^2}{L^2} + (\bar{F}_p - \mu_A - \frac{\sigma_A}{L}p\sigma_A w_{\parallel})^2 \\ &= \frac{((1-p)\sigma_A\sigma_C)^2}{L^2} + (\bar{F}_p - \mu_A - \frac{\sigma_A^2}{L^2}p(f_{\hat{\theta}(Z_{T,n,p})} - \bar{F}_p))^2 \end{aligned}$$

By symmetry, we have:

$$\mathcal{E}_C = \frac{(p\sigma_A\sigma_C)^2}{L^2} + (\bar{F}_p - \mu_C - \frac{\sigma_C^2}{L^2}p(f_{\hat{\theta}(Z_{T,n,p})} - \bar{F}_p))^2$$

We wish to consider what the expected contribution from each distribution will be over the distribution of $f_{\hat{\theta}(Z_{T,n,p})}$:

$$\begin{aligned} \mathbb{E}[\mathcal{E}_A] &= \sigma_A^2 + (1-p)^2(\mu_C - \mu_A)^2 \\ \mathbb{E}[\mathcal{E}_C] &= \sigma_C^2 + p^2(\mu_C - \mu_A)^2 \end{aligned}$$

Finally, we restore the argument x and consider the expectation of these values over the distribution Φ_x :

$$\begin{aligned} \mathbb{E}_{\Phi_x}[\mathbb{E}[\mathcal{E}_A(x)]] &= \mathbb{E}_{\Phi_x}[\sigma_A^2(x) + (1-p)^2(\mu_C(x) - \mu_A(x))^2] \\ \mathbb{E}_{\Phi_x}[\mathbb{E}[\mathcal{E}_C(x)]] &= \mathbb{E}_{\Phi_x}[\sigma_C^2(x) + p^2(\mu_C(x) - \mu_A(x))^2] \end{aligned}$$

Intuitively, the lower the value, the more a random sample from the corresponding distribution is expected to contribute towards moving the current mixed model towards the expected value of the mixed model, leading to a greater expected Influence:

$$\begin{aligned} &\mathbb{E}_{\Phi_x}[\sigma_A^2(x) + (1-p)^2(\mu_C(x) - \mu_A(x))^2] \\ &< \mathbb{E}_{\Phi_x}[\sigma_C^2(x) + p^2(\mu_C(x) - \mu_A(x))^2] \\ &\Rightarrow \mathbb{E}_{\Phi_x}[(\sigma_C^2(x) - \sigma_A^2(x)) + (2p-1)(\mu_C(x) - \mu_A(x))^2] \\ &> 0 \end{aligned}$$

□

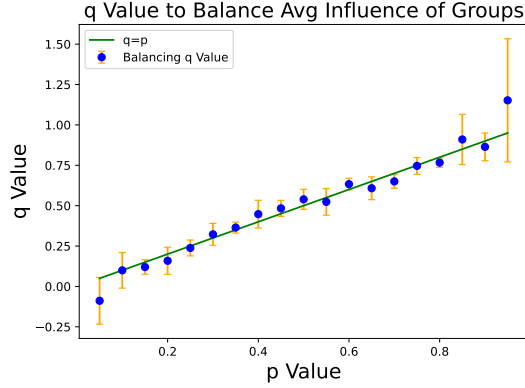


Figure 3.2: Average over all regression datasets with LS corruption. Y values are the Q values that set the average Influence of accurate and corrupted points equal. Error bars are one standard deviation.

We examine the consequences of this inequality in some simplified contexts. First, we consider obfuscation used to achieve privacy, such as in a differential privacy setting, where some agents add noise to their data. In this case, we have $\mu_C - \mu_A = 0$, and $\mathbb{E}_{\mathcal{D}_x} [\sigma_A^2(x)] < \mathbb{E}_{\mathcal{D}_x} [\sigma_C^2(x)]$. This confirms the intuition that an Agent who uses less noise will in expectation have less privacy but contribute more value to the model. On the other hand, if the variances of the distributions are the same, then the more valuable distribution is determined by majority vote according to the $(2p - 1)$ term. This satisfies the intuition that equally accurate partial models are a priori indistinguishable in terms of their contributions to the mixed model. Most importantly, this analysis suggests that in the finite sample regime, filtering might be able to achieve a training set mixing proportion better than $p = q$. It is important to note that the variances themselves depend in some way on the value of p : the higher the p value, the more samples are used to determine $f_{\hat{\theta}(Z_{T,n,p})}$, so the variance of the posterior will be lower.

3.3 Filtering Schemes

In the previous sections we demonstrated that relative expected Influences of accurate and corrupt data samples depend on their relative presence in the training samples as well as how noisy the corresponding models are. We propose a number of filtering schemes which we apply to the untrusted validation data regime. We first examine two filtering schemes which would be natural to consider in a trusted validation data regime, in which lower Influence values reliably indicate lower quality data. In the untrusted validation regime, lower Influence values can only indicate lower quality data probabilistically in relation to the average Influence value. This is also contingent on the variables which determine the relative expected Influences according to the previous analysis. We propose a probabilistic filtering scheme which seeks to take advantage of the differences in expected Influences when these variables are favorable to the Center, i.e. the validation samples are untrusted but at least more trusted than the training

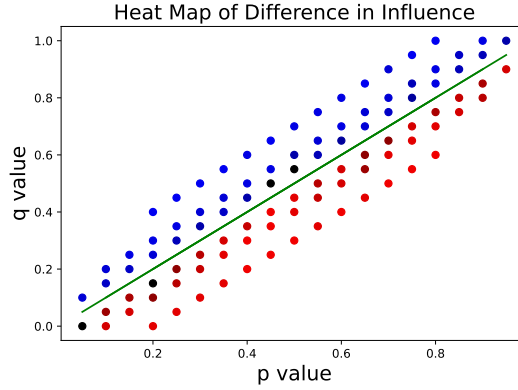


Figure 3.3: Heat map over all datasets with LS corruption. Coloration represents the difference between average Influence of accurate and corrupted data with $q = p \pm \epsilon$ for $\epsilon \in \{0.05, 0.1, 0.2\}$. More blue means more simulations with accurate data achieving higher Influence, more red means the opposite.

Algorithm 1 Threshold Influence Filtering (TIF)

Input: Z, \mathcal{I}, τ
Output: Z_f
 Initialize $Z_f = Z$
for $\forall i$ **do**
 if $\mathcal{I}(z_i, Z) < \tau$ **then**
 Remove z_i from Z_f
 end if
end for

samples.

3.3.1 Threshold Influence Filtering

A common filtering scheme would be to compute some performance score for each data sample, rank them, then eliminate the samples with the worst performance score. We could treat the Influence score the same way, resulting in the naive *Threshold Influence Filtering* (TIF) scheme, as described in Algorithm 1.

With the analysis in the previous sections, we have a better understanding of the structure of expected Influence values in an untrusted validation data regime. Unfortunately, this does not tell us much about the TIF scheme, as the expected Influence score gives very little information about the overall distribution of Influences. The TIF scheme is concerned with which data samples are most likely to have the lowest Influence score. This is related to higher moments in the distribution of Influences, which is difficult to analyze without strong assumptions. Nonetheless, it only requires a single round of Influence computations. Let $g(n)$ be the computational complexity of computing a single Influence score when there are n

Algorithm 2 Iterative Minimal Influence Filtering (IMIF)

Input: Z, \mathcal{I}, τ
Output: Z_f
Initialize $Z_f = Z$
while $\min_{z_i \in Z_f} \mathcal{I}(z_i, Z_f) < \tau$ **do**
 Remove $\arg \min_{z_i \in Z_f} \mathcal{I}(z_i, Z_f)$ from Z_f
end while

Algorithm 3 Uniform Probabilistic Filtering (UPF)

Input: $Z, \mathcal{I}, \alpha \in \mathbb{R}, \beta \in (0, \infty)$
Output: Z_f
Initialize $Z_f = \{\}$
for $\forall i$ **do**
 $p = \frac{\mathcal{I}(z_i, Z) - \alpha}{\beta}$ clipped to $[0, 1]$
 With probability p , add z_i to Z_f
end for

training points, with $O(g(n)) > O(n)$. Then, ignoring the complexity of sorting the Influence scores, the computational complexity of the TIF scheme is $O(n g(n))$. TIF is also the most intuitive way to approach filtering with Influence scores.

3.3.2 Iterative Minimal Influence Filtering

The TIF scheme suffers from a moving target problem. Suppose we run TIF to remove all the data samples with negative Influence in one shot. Instead, if we were to remove them one at a time, the Influences of the remaining samples will change, and some might become positive. So the order of removal matters, and the data samples can have complicated relationships with each other than affect Influence scores. We consider an "optimal" Influence filtering scheme, in the sense that it will filter out data samples such that the resulting empirical risk minimizer has the minimum possible risk on a filtered training set. We simply check the empirical risk for every possible subset of the training set, down to some minimal number of samples necessary for determining a model, and pick the subset which produces the model with the lowest empirical risk. This would be an absurd proposition to attempt in practice, but we observe that for this optimal subset, the set of samples that gets removed has some properties in terms of Influence values. If we were to check the average Influence of each of these samples over every possible ordering of removal, we would find that the average Influence of each point must be negative. Furthermore, if we were to rank the average Influence of every sample in every possible ordering of removal down to the minimum subset size, we would find that these points have the lowest average Influence. These averages can be said to be the Shapley values of Influence for each sample.

Unfortunately, computing Shapley values of Influence is still prohibitively computationally expensive, but outside of some pathological counterexamples, filtering the data sample with

the minimum Influence score one at a time would not be much worse than filtering according to minimum Shapley value, especially for large training sets. We call such a filtering scheme *Iterative Minimal Influence Filtering* (IMIF), and the protocol is shown in Algorithm 2. As long as the threshold for removal is 0, IMIF is guaranteed to improve the model risk. Because of this property, and its relationship to the optimal filtering scheme using Shapley values, we say that this scheme is a *near-optimal* greedy filtering scheme. Despite begin far less computationally complex than computing Shapley values, the IMIF scheme is still significantly more complex than TIF. IMIF can perform up to $\sum_{i=1}^n n g(n)$ operations, which is $O(n^2 g(n))$.

3.3.3 Uniform Probabilistic Filtering

Finally, we address the notion of "filtering according to expected Influence". Since we do not have information about how the Influence values of samples from the accurate and corrupt distributions should be ordered, we cannot address this with deterministic filtering schemes like TIF or IMIF. Instead, we propose a probabilistic filtering scheme. The probabilistic filtering requires a choice of CDF $P_{Z_T}(z_i)$ for $z_i \in Z_T$. We will show that a uniform CDF is the most natural choice. We call the probabilistic filtering scheme with uniform CDF *Uniform Probabilistic Filtering* (UPF), with protocol described in Algorithm 3. The choice of α and β can be tuned as hyper-parameters to adjust the sensitivity of the filter, but the intuitive choices are $\alpha = \min_i \mathcal{J}(z_i, Z)$ and $\beta = \max_i \mathcal{J}(z_i, Z) - \min_i \mathcal{J}(z_i, Z)$. We prove that, in the untrusted data regime, UPF is expected to improve the true model accuracy, i.e. the accuracy evaluated only on the target distribution, up to a certain limit determined by the expected Influence score equilibrium set forth by the infinite and finite sample analyses.

Theorem 3.3.1. *Let Z_T be the training set with n samples and mixed according to p . Let the validation set be mixed according to q with $p < q$. Then the expected probability of filtering out a point $z_a \in Z_T$ drawn from Φ_a is less than the probability of filtering out a point $z_c \in Z_T$ drawn from Φ_C , according to the UPF protocol.*

Proof. From Theorem 3.2.4 we have

$$\begin{aligned} & \mathbb{E}[\mathcal{J}(z_a, Z_T)] > \mathbb{E}[\mathcal{J}(z_c, Z_T)] \\ \Rightarrow & \frac{\mathbb{E}[\mathcal{J}(z_a, Z_T)] - \alpha}{\beta} > \frac{\mathbb{E}[\mathcal{J}(z_c, Z_T)] - \alpha}{\beta} \\ \Rightarrow & \mathbb{E}\left[\frac{\mathcal{J}(z_a, Z_T) - \alpha}{\beta}\right] > \mathbb{E}\left[\frac{\mathcal{J}(z_c, Z_T) - \alpha}{\beta}\right] \end{aligned}$$

□

We note the computational complexity of UPF is $O(n * g(n))$, the same as TIF.

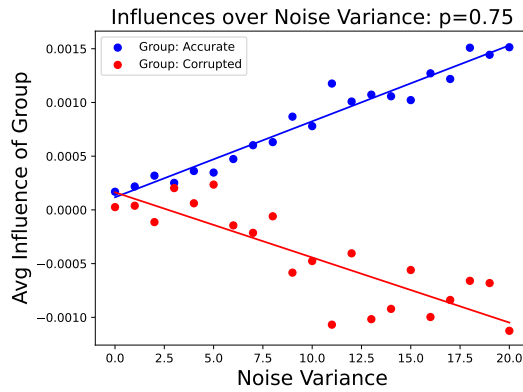


Figure 3.4: Crime dataset with AGN corruption. Noise mean 0. Noise variance ranges from 0 to 20. p and q are fixed at 0.75.

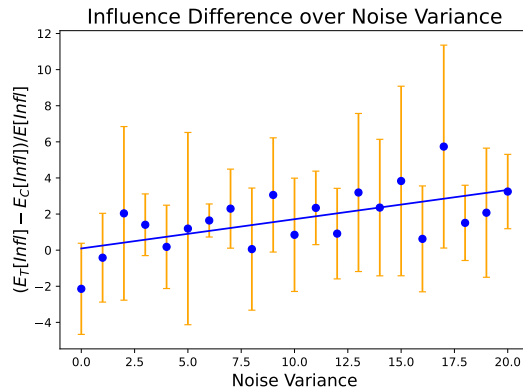


Figure 3.5: Normalized difference in average Influence aggregated over all regression datasets. p and q fixed at 0.75. Error bars are one standard deviation.

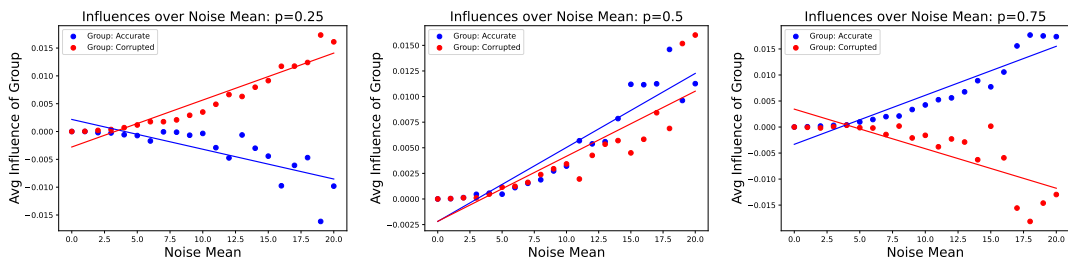


Figure 3.6: (a) Crime dataset with AGN corruption. Noise variance is 0. Noise mean ranges from 0 to 20. p and q fixed at 0.25. (b) Crime dataset with AGN corruption. Noise variance is 0. Noise mean ranges from 0 to 20. p and q fixed at 0.5. (c) Crime dataset with AGN corruption. Noise variance is 0. Noise mean ranges from 0 to 20. p and q fixed at 0.75.

3.4 Empirical Analysis

For our simulations we use a combination of real datasets, many from the UCI Machine Learning Repository (Dua and Graff, 2017) and different forms of data corruption, which we

outline here:

- Regression
 - *Air Quality* (De Vito et al., 2008)
 - *Communities and Crime (Crime)* (Redmond and Baveja, 2002)
 - *Parkinsons Telemonitoring (Parkinsons)* (Tsanas et al., 2009)
 - *Red Wine* (Cortez et al., 2009)
 - *White Wine* (Cortez et al., 2009)
- Classification
 - *Audit Risk (Audit)* (Hooda et al., 2018)
 - *Banknote Authentication (Bank)* (Dua and Graff, 2017)
 - *MNIST* (Deng, 2012)

We provide a representative selection of data when it is not sensible to aggregate all the datasets. For all the datasets, we removed attributes that were non-predictive, redundant, or had a substantial number of missing values.

Data from these sources are treated as accurate. For corruption we use the following methods:

- Label Shuffle (LS): All labels in the dataset are shuffled uniformly prior to sampling. This represents a common form of corruption due to human error.
- Uniform Input, Uniform Label (XuYu): Both input and label values are sampled from a uniform distribution inside the bounding box of the data.
- Gaussian Mixture Input, Uniform Label (XgmmYu): Label values are sampled from a uniform distribution inside the bounding box. The distribution of input values of the data is approximated by a Gaussian Mixture Model. The input values of the data are given equal weight and each Gaussian has the same covariance. We compute the covariance of all the input values. We then compute the average density of the data as $D = \frac{\text{volume of bounding box}}{\# \text{ of data points}}$. Finally, we normalize the covariance as $\frac{\text{Cov} * D}{\text{Tr}(\text{Cov})}$.
- Additive Gaussian Noise (AGN): Gaussian noise is added to the labels. This is only applicable to regression data and is used for simulations related to the finite sample regime.

3.4.1 Infinite Sample Regime

We check if empirical results match the predicted behavior in an infinite sample regime by varying the training dataset mixture proportion p . In all experiments, validation is performed

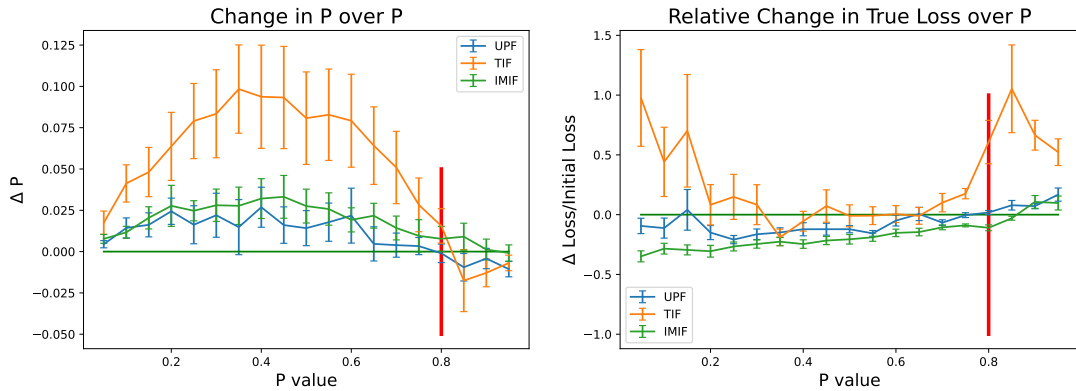


Figure 3.7: Filter performance metrics averaged across all combinations of datasets with LS, XuYu, and XgmmYu corruption. q value is fixed at 0.8. (a) Change in p value. Error bars are $\frac{1}{2}$ standard deviation. A higher value is better. (b) Relative change in real loss, real loss being the loss measured only against the target distribution. Error bars are $\frac{1}{5}$ standard deviation. A lower value is better.

on a large number of both accurate and corrupted data. We can then simulate a validation dataset mixture according to q by taking an appropriately weighted combination of the average Influence on each group. The theory predicts that setting $q = p$ should set the expected Influences of accurate and corrupted training points to be equal. Figure 3.2 shows behavior that is clearly consistent with theory, with low deviation from the predicted behavior. Unfortunately, the simulations with the classification datasets produce more instability and thus don't demonstrate the theory on visual inspection. To show that the theory is consistent across datasets, Figure 3.3 shows that when we are perturbed from the $q = p$ regime, nearly every simulation produces behavior consistent with the theory: when $p < q$, accurate data achieves a higher Influence score, and vice versa when $p > q$.

3.4.2 Finite Sample Regime

We check if empirical results match the predicted behavior in a finite sample regime when $p = q$. We set $(\mu_C - \mu_T)^2 = 0$ by having the corrupted data draw samples from the same dataset as the accurate data, but with mean 0 Gaussian noise added. This is sufficient to demonstrate the theory. Although we do not estimate the variances of the model posteriors directly, we simply note that increasing variance in the noise will result in an increase in the variance of the posterior. Of course we cannot "de-noise" the accurate data, and the underlying distributions are the same, so reversing the inequality is symmetric. It is clear that with finite samples, increasing the variance of the noise will increase the variance of the corresponding model posterior. We also expect that as the inequality becomes more pronounced, the difference in expected Influence should increase. Figure 3.4 clearly reveals this separation for one particular dataset. We include an aggregation of all the datasets in figure 3.5 to demonstrate that they follow the predicted trend, i.e. the difference in expected Influences increases.

Figure 3.6 demonstrates the correctness of Formula 3.4 when the variances are equal. We observe that when the $(2p - 1)$ term is non-zero, the differences in average Influences increase with the difference in means. The direction of this separation is reversed between $p = 0.25$ and $p = 0.75$. Intuitively, there is no way to distinguish the target distribution from the corrupted distribution, the labels are arbitrary. But if we do have some a priori knowledge about whether or not the trusted distribution is the majority, this can help distinguish them. When $p = 0.5$, the majority selection term $(2p - 1)$ is 0, so we expect no separation as is shown by Figure 3.6.

3.4.3 Incentives

Figure 3.3 demonstrates the robustness of the incentives when deviating from $p = q$. Such an incentive mechanism will induce reports that asymptotically achieve $p = q$. Formula 3.4 modifies the location of this asymptote to some perturbation of $p = q$. Figures 3.4 and 3.6 clearly illustrate that this perturbation grows as the inequality becomes more pronounced. Nonetheless, in real world settings this perturbation will be small, as is demonstrated by the consistency of figure 3.3.

3.4.4 Filtering

We evaluate filter performance across many datasets for different p values and a fixed $q = 0.8$. For all filters, s is simply Influence. For UPF, $\alpha = \min_j s(z_j, Z)$ and $\beta = \max_j s(z_j, Z) - \min_j s(z_j, Z)$. For TIF and IMIF, $\tau = 0$. We consider two performance metrics: 1) The change in the proportion p of accurate data in the training dataset, and 2) The change in real loss, i.e. the loss evaluated against only accurate data. We consider real loss over validation loss because it is the desired performance metric of the center. We observe in Figure 3.7 that IMIF and UPF perform similarly for both metrics, granting a small improvement in p , but a significant improvement in true loss (up to a 40% reduction), with IMIF performing slightly better. UPF having comparable performance despite far lower computational complexity than IMIF demonstrates the efficacy of the theory and suggests that probabilistic filtering techniques may be under-explored. The surprising observation that TIF performs significantly better in Figure 3.7 on the Δp metric but worse in Figure 3.7 on true loss can be explained by TIF naively removing points with a very small negative value, which if correctly classified have the same effect on p as removing more influential points, but have little effect on the loss. Filter performance becomes detrimental when $p > q$, as predicted by the theory.

3.5 Summary

We examine the use of Influence for the problem of data filtering, which is related to the problem of incentive mechanism design, but different in that scores like Influence cannot be used to exploit knowledge gaps between Agents' prior and posterior beliefs. Instead, the scores must reflect the actual properties of the data. Because of this difficulty, most data filtering

schemes rely on a trusted validation data set for producing accurate scores and metrics. We examine the setting when both the training data set and the validation data set are untrusted.

By analyzing Influence scores in the infinite sample limit, we obtain a straightforward result that the expected influence scores of accurate data will only be greater than the expected influence scores of corrupted data when the relative presence of accurate data is higher in the validation data set. An alternative form of analysis in a finite sample regime demonstrates that this result can be slightly modified depending on the variances of the model posteriors produced by finite samplings of accurate or corrupt data.

Using this analysis we propose a probabilistic Influence based filtering scheme and compare it to more straightforward deterministic filtering schemes, which would be the natural consideration for a trusted validation data setting. We find that the probabilistic filtering performs comparably to the far more computationally complex "near-optimal" deterministic filter.

4 Peer Neighborhoods

4.1 Introduction

In previous chapters we addressed Influence-based mechanisms for both crowdsourcing and filtering. An important aspect of Influence-based mechanisms is that they operate on a broader set of distributions than classical Peer Prediction mechanisms, which generally operate only on categorical, or finite discrete, distributions because they rely on a notion of report matching, i.e. the Agent and Peer report a sample from the same category. We refer to such mechanisms as *Peer Consistency* mechanisms. Because they are restricted to categorical distributions, they eschew any notion of locality among the categories. This disadvantage presents a clear theoretical roadblock for applying such mechanisms to arbitrary distributions, since continuous random variables are only understandable through measuring local neighborhoods.

In this chapter, we present a novel framework for extending Peer Consistency mechanisms to arbitrary distributions. We call such extensions *Peer Neighborhood* mechanisms. To our knowledge, this is the only work that does so without assuming that the Center possesses a priori knowledge of properties of the Agents or of the underlying distribution. We only assume that Agents are rational and that they follow some reasonable belief update conditions, which we will show admit a broad class of updates. By analyzing an extension of the Peer Truth Serum, we prove that it can admit truthful Bayes-Nash Equilibria on the ex-ante game produced by the mechanism.

4.1.1 Approach

A natural approach to extending Peer Consistency mechanisms to arbitrary distributions is to discretize the space of reports and then apply the discrete mechanism as normal. Suppose the Center is collecting temperature data from a set of Agents with sensors. A simple implementation would be to consider only the whole number of the temperature. This may be sufficient for the Center's needs if it is trying to model the temperature for a purpose that does not

require a high degree of accuracy, like vacation planning, or if the sensors are not very accurate themselves. It is also important to note that Agents are not incentivized to be perfectly truthful, as decimal digits are irrelevant to the payments. The Center can also rely on the fact each whole number temperature range is large enough that a significant number of truthful reports would be in this range. Since Peer Consistency mechanisms pay for an exact match with a Peer, a higher probability of matching means less volatile payments. But what if the Center requires a much higher degree of accuracy in the reports, say up to thousandths of a degree? The true probabilities of matching may be so small that the Center would have to collect enormous amounts of data for the mechanism to reasonably approximate this probability of matching via random Peer reports. This means much more volatile payments, which may be unacceptable to a risk averse Agent. The volatility of the payments does not only affect the Agents; the Center will have a harder time budgeting this crowdsourcing effort. We see that the Center needs to strike a balance between the accuracy of reports and volatility of the payments when implementing a Peer Consistency mechanism in this way. It is possible that circumstances make it impossible to strike this balance.

The Center can improve on this paradigm by increasing the number of mechanisms it uses. Consider again a Center collecting temperature data with a mechanism considering only the whole number of the temperature. But now consider a second discretization with only whole number ranges from $n + \frac{1}{2}$ to $n + \frac{3}{2}$ and the same mechanism applied to this discrete set. For each of the two mechanisms, the range of values in each bin is the same, so one can assume that the true probabilities of matching will be similar. The Center can scale the payments from each mechanism by $\frac{1}{2}$ to give approximately the same expected payoff, but for the Agents there is a different pattern of payoffs. Suppose an Agent reports a value of $n + \frac{3}{4}$. For one mechanism this matches with a range of Peer reports from n to $n + 1$. For the other mechanism this matches with a range from $n + \frac{1}{2}$ to $n + \frac{3}{2}$. We see that the Agent will get the full payment when matching with a Peer report in the intersection: the range $n + \frac{1}{2}$ to $n + 1$, but only half payment for matching in the ranges n to $n + \frac{1}{2}$ and $n + 1$ to $n + \frac{3}{2}$. In this way, the Agent's incentive for reporting is refined to a smaller range without changing the volatility in the payments.

This can be refined even further by adding more mechanisms with different discretizations. The region where the Agent gets a payoff for every mechanism is the intersection of the regions which contain the report across all the mechanisms. By construction, the intersection contains the report, but in order for this combination of mechanisms to be truthful, the intersection should not contain any other point. How can this be achieved? Instead of considering a weighted sum of mechanisms, let's consider a construction with equivalent payoff as randomly selecting a single mechanism from a set of mechanisms, then taking the expectation over this distribution. We can then consider a continuous distribution over a set of mechanisms, which gives the level of refinement necessary to isolate a single point as the intersection.

But how does this construction correspond to the prior details? We have assumed that nearby ranges of values have similar probabilities of matching, but this may not be the case. If an

Agent believes that a small range of values has an extremely high probability, this can outweigh the additional value achieved by matching across all the mechanisms. The applicability of this multi-mechanism construction comes down to the reasonableness of the prior details which guarantee incentive-compatibility. To evaluate this, we consider a prior detail setting in which the Agents possess a *prior belief* about the distribution of truthful Peer reports, then after observing a sample, they update to a *posterior belief*. Then the only considerations for the prior details are the *update conditions* which an Agent must follow when constructing a posterior from a prior and an observation. It is clear that a reasonable Agent should follow some conditions. For example, on a discrete distribution the update should be consistent with Bayes' Rule. For a continuous distribution, it is less clear what constitutes "reasonable" update conditions, but broadly there should be some notion of locality with respect to the observation, i.e. probabilities of events should increase more the closer they are to the observation.

When considering a discrete mechanism to extend in this way, if the mechanism is incentive-compatible with respect to some discrete update condition, we can present a general formulation of sufficient update conditions based on this discrete update condition. To assess the "reasonableness" of these conditions, we analyze a specific instance of this mechanism extension with respect to the Peer Truth Serum discrete mechanism (Faltings et al., 2014), which is incentive-compatible with respect to the *self-predicting* update condition. We demonstrate that the extension of the update condition still admits a broad class of updates, generally following some locality, boundedness, and symmetry constraints. We show how to construct some of these updates, which appear "reasonable". Finally, we will demonstrate empirically that the incentives are clear and stable when Agents use these updates.

4.1.2 Model

In a crowdsourcing setting there is a Center that wishes to learn an arbitrary distribution Φ , which we call the *true distribution*, but the Center can't probe this distribution in a meaningful way. The Center tries to learn the distribution by collecting *reports* from a set of independent, self-interested Agents who can sample Φ to produce an *observation*. Because the Agents are self-interested, they must be *incentivized* to produce reports that help the Center learn Φ . The incentive an Agent experiences is a personal utility function that depends on the Agent's reporting strategy and a set of *beliefs* the Agent has about the setting, such as the distribution Φ and the reporting strategies of other Agents. Agents always act rationally, so they will adopt the reporting strategy which maximizes their expected utility under their current beliefs. The Center's goal is to choose a payment function which dispenses utility to the Agents in exchange for reports, such that Agents will be incentivized to adopt "good" reporting strategies. In our case, we seek *truthful* reporting, meaning Agents report their observations.

In the setting we consider, the Agent does not have a static set of beliefs. We refer to the belief of the Agent about the true distribution before making the observation as π , the *prior belief*. After making an observation o , the prior is updated to π_o , the *posterior belief*. Prior to

the data collection period, the Center also has its own belief about the true distribution, R , which it makes public to the Agents. We refer to R as the *public prior*. It is assumed that these probability measures are on a shared measurable space (Ω, Σ) . When discussing arbitrary distributions, we will assume that $\Omega = \mathbb{R}^d$ for some d , and $\Sigma = \mathcal{B}(\Omega)$, the Borel sets of Ω . At the end of the data collection period, the Center will have received a set of reports $\{r\}$. For each report, it randomly picks a Peer report and performs some comparison between the two reports. This informs the payment for the report. This payment process is also made public to the Agents.

4.2 Peer Neighborhood Mechanisms

4.2.1 Peer Consistency

In the original setting for Peer Consistency mechanisms, Φ is a categorical distribution, and the mechanisms pay an Agent when its report matches with a randomly selected Peer report. We formalize this concept:

Definition 4.2.1 (Peer Consistency). A *Peer Consistency* mechanism is a mechanism which assumes some public prior R with categorical distribution, takes a report r from an Agent and a report rr from a randomly chosen Peer, and pays the Agent $\tau_R(r, rr) = f(rr) + s_R(r) * \mathbb{1}_{r=rr}$ where f depends only on rr and s is a non-negative *scoring function*.

When discussing the incentives of a Peer Consistency mechanism, this is typically in regards to some *update condition*:

Definition 4.2.2 (Update Condition). Given a prior probability measure π , an observation o , and a posterior probability measure π_o , an *update condition* is a Boolean function $S(\pi, \pi_o)$. We call $S^*(\pi, \pi_o)$ the *natural update condition* for some scoring function s_π as $\forall x \neq o : \pi_o(o) * s_\pi(o) > \pi_o(x) * s_\pi(x)$.

A notable example that we will use is the *self-predicting* update condition for the Peer Truth Serum Radanovic et al., 2016. This is the natural update condition and is given by $s_\pi(r) = \frac{1}{\pi(r)}$, so the condition is $\forall x \neq o : \frac{\pi_o(o)}{\pi(o)} > \frac{\pi_o(x)}{\pi(x)}$.

Definition 4.2.3 (Update Process). Given a prior probability measure π and an observation $o \in \Omega$, an *update process* is a function $\mathcal{U}(\pi, o) = \pi_o$. We say an update process satisfies an update condition S if $\forall \omega \in \Omega : S(\pi, \mathcal{U}(\pi, \omega))$ is true.

Definition 4.2.4 (Incentive-Compatibility of Peer Consistency). A Peer Consistency mechanism with public prior R is considered *incentive-compatible* with respect to an update condition S if an Agent, with prior set to R and an update process which satisfies S , believes that for any observation o , their expected payment, over the Peers distributed according to their posterior, is maximized by truthfully reporting o .

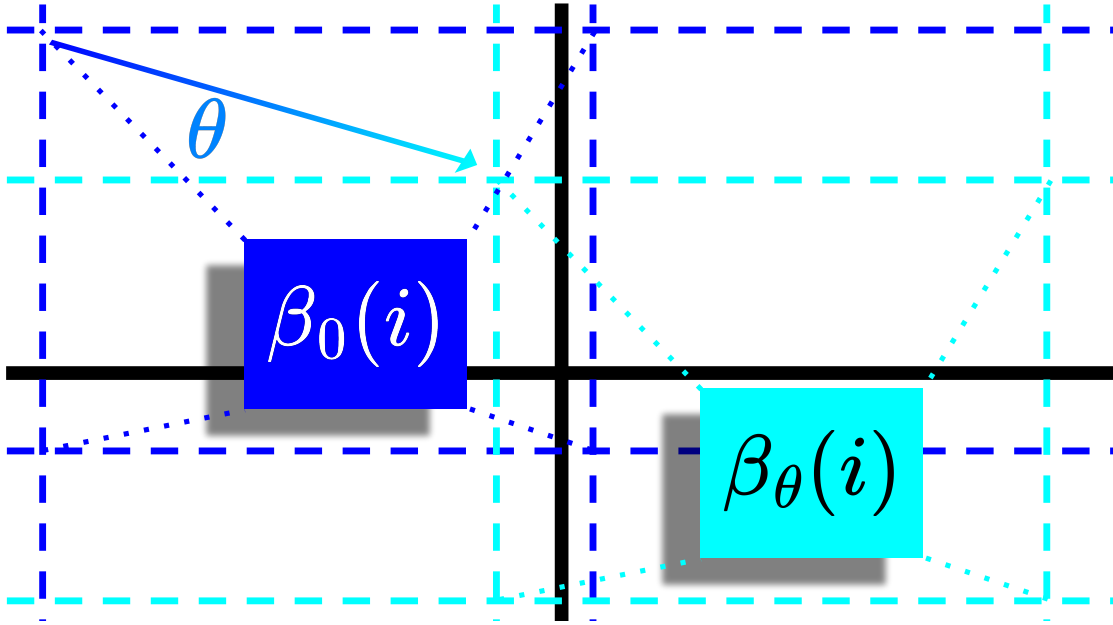


Figure 4.1: An example of a partition family on \mathbb{R}^2 with θ representing translations of the bins: $\beta_0(i)$ transforms into $\beta_\theta(i)$. Partition families are used to construct Peer Neighborhoods.

Proposition 4.2.5. *A Peer Consistency mechanism is incentive-compatible with respect to the natural update condition.*

Proof. Consider an Agent with prior $\pi = R$, observation o , and posterior π_o . Suppose the update satisfies $\forall x \neq o : \pi_o(o) * s_\pi(o) > \pi_o(x) * s_\pi(x)$. Then $\forall x \neq o : \mathbb{E}_{r \sim \pi_o} [s_R(o) * \mathbb{1}_{o=rr}] > \mathbb{E}_{r \sim \pi_o} [s_R(x) * \mathbb{1}_{x=rr}] \Rightarrow \forall x \neq o : \mathbb{E}_{r \sim \pi_o} [\tau_R(o, rr)] > \mathbb{E}_{r \sim \pi_o} [\tau_R(x, rr)]$. \square

4.2.2 Partition Spaces

Peer Neighborhood mechanisms place a layer of abstraction on top of Peer Consistency mechanisms to introduce a notion of locality. They do so by considering a *family of partitions* of the space of reports. A standard approach to applying a Peer Consistency mechanism to a continuous distribution, such as a Gaussian distribution, is to pick some fixed discretization, or partition. Each *bin* of the partition corresponds to a category for the mechanism. A truthful mechanism for some update condition over a categorical distribution would then have a bin-truthful Bayes-Nash Equilibrium over this continuous distribution if the Agent's belief update satisfies the update condition with respect to the bin-categories. An Agent would have an equal incentive to report any value inside the bin containing the truthful report.

By considering a family of partitions rather than a single one, the incentives can be refined. Consider a Gaussian distribution partitioned into integer length bins $(n, n + 1]$. An incentive-compatible Peer Consistency mechanism would then incentivize any report with the correct integer value. If a second partition is introduced with bins $(n + \frac{1}{2}, n + \frac{3}{2}]$, and the Agent satisfies

the update condition for both partitions, the report is now incentivized to be within an interval of length $\frac{1}{2}$, corresponding to the intersection of the truthful bins from each partition. We will see that if the partition family is constructed correctly, this intersection can be refined to contain only truthful report.

Definition 4.2.6 (Partition Family). A *partition family* T is a function which maps a set of parameters $\theta \in \Theta$ to a partition, which is a set, of at most countable size, of measurable bins β that are disjoint and cover Ω : $T(\theta) = \{\beta_\theta(i)\}_{i \in \mathbb{Z}_\theta^*}$ where $\mathbb{Z}_\theta^* \subseteq \mathbb{Z}$ and $\beta_\theta(i) \in \mathcal{B}(\Omega)$ such that $\forall \theta$, $\bigcup_{i \in \mathbb{Z}_\theta^*} \beta_\theta(i) = \Omega$ and $\forall i \neq j$, $\beta_\theta(i) \cap \beta_\theta(j) = \emptyset$

A simple examples of a partition family over \mathbb{R}^2 is shown in Figure 4.1. The Center must have some way of selecting a partition from the family. We have the Center pick the partition randomly according to some distribution, which we call the *partition selection distribution*.

Definition 4.2.7 (Partition Selection Distribution and Partition Space). The *partition selection distribution* is given by a probability measure Ψ over some measurable space (Θ, Σ) , where Θ is the set of parameters for the model family and Σ is some σ -field over Θ . Without loss of generality, let Ψ be supported on Θ . We call the pair (T, Ψ) the *partition space*.

For ease of reading we will often use the *bin selection function* to identify the bin that contains a particular point:

Definition 4.2.8 (Bin Selection Function). The *bin selection function* with respect to a partition family $T(\theta)$ is a function $\mathbb{X}_\theta : \Omega \rightarrow \mathbb{Z}_\theta^*$ such that $\mathbb{X}_\theta(z) = i$ if and only if $z \in \beta_\theta(i)$.

The Bin Selection Function is well-defined as a result of the bins of each partition being disjoint and covering Ω . It is important for the Center's implementation that this function be computable.

If an Agent is to have a strictly truthful incentive, there must be a non-zero probability of any other report failing to match with the truthful report under the partition family. We call this *point-isolating*.

Definition 4.2.9 (Point-isolating). A partition space (T, Ψ) is *point-isolating* over R if: $\omega_1 \neq \omega_2$ in the support of $R \Rightarrow \Psi(\{\theta : \mathbb{X}_\theta(\omega_1) \neq \mathbb{X}_\theta(\omega_2)\}) > 0$.

Reports with a matching probability in R of 0 can result in infinite payments under some Peer Consistency mechanisms. To avoid some degenerate payments, we impose the following condition on the partition family:

Definition 4.2.10 (Bin-supported). A partition space (T, Ψ) is *bin-supported* over R if:

$$\forall \omega \in \Omega : \Psi(\{\theta : R(\beta_\theta(\mathbb{X}_\theta(\omega))) = 0\}) = 0.$$

In simpler terms, for any possible report, the probability of selecting a partition with a 0 R -probability bin containing the report is 0 in Ψ .

Proposition 4.2.11. $\forall R, \exists(T, \Psi)$ such that (T, Ψ) is point-isolating and bin-supported over R

Proof. Suppose $\Omega = \mathbb{R}^d$, define $T^*(\theta) = \{\forall n_j \in \mathbb{Z} : \otimes_{j=1}^d [n_j + \theta_j, n_j + 1 + \theta_j]\}$ and $\Theta = \otimes_{j=1}^d [0, 1)$. Let Ψ be uniform over Θ . For all θ , we contract the partition $T^*(\theta)$ by merging any bin with probability 0 in R with the closest bin with positive probability in R , breaking ties with any deterministic process. We call this new partition family T and claim that (T, Ψ) is point-isolating and bin-supported over R .

By construction, (T, Ψ) is clearly bin-supported over R , as all bins with probability 0 have been merged with bins of positive probability. Suppose two points $\omega_1 \neq \omega_2$ are in the support of R . Then there is a set of ω that separates them in T^* with probability $\epsilon > 0$ in Ψ , but in order for T to not be point isolating, the probability that the bin containing one of the points has 0 probability in R must be ϵ . The set of ω which places one of the points on the closed boundary of a bin is clearly probability 0 in Ψ , and for all other θ separating the two points there exist open sets around the two points that are contained within the bins, and there is a positive probability in Ψ of picking partitions that separate the open sets. Because the points are in the support of R , those open sets have positive probability in R , so the bins both have positive probability, so they can't be merged. \square

4.2.3 The Mechanism Extension

Now, we can introduce the Peer Neighborhood mechanism extension. First we must modify the probability measures:

Definition 4.2.12 (Partitioned Probability Measure). Let π be a probability measure on $(\Omega, \mathcal{B}(\Omega))$. Let $T(\theta)$ be a partition of Ω . Then for $i \in \mathbb{Z}_\theta^*$, let the *partitioned probability measure* $\pi^\theta(i) = \pi(\beta_\theta(i))$.

We can then extend any Peer Consistency mechanism as follows:

Definition 4.2.13 (Peer Neighborhood Mechanism Extension). Given some Peer Consistency mechanism with payment function τ , we define the bin-extension payment function with respect to some partition $T(\theta)$ as $\tau_R^\theta(r, rr) = \tau_{R^\theta}(\mathbb{X}_\theta(r), \mathbb{X}_\theta(rr))$. Then given some partition selection distribution Ψ such that (T, Ψ) is point-isolating and bin-supported over R , the *Peer Neighborhood extension mechanism* pays according to:

$$\tau_R^\Psi(r, rr) = \mathbb{E}_{\theta \sim \Psi}[\tau_R^\theta(r, rr)] \quad (4.1)$$

4.2.4 Incentive-Compatibility

Given some Peer Consistency mechanism with scoring function s , we wish to discover an update condition $S^{(T, \Psi)}$ for which the associated Peer Neighborhood extension mechanism is incentive-compatible. We suggested earlier that the incentivized report region could be refined as long as the Agent is incentivized to be bin-truthful for all the partitions, so the most straightforward condition is that S is satisfied with probability 1 in Ψ .

Definition 4.2.14 (Partition-Invariant Update Condition). Given a prior π , a posterior π_o , a partition space (T, Ψ) , and a Peer Consistency mechanism with scoring function s , the *Partition-Invariant* (PI) update condition $S_{PI}^{(T, \Psi)}$ takes the form:

$$\Psi(\{\theta : S^*(\pi^\theta, \pi_o^\theta)\}) = 1 \quad (4.2)$$

Proposition 4.2.15. Given a Peer Consistency mechanism with payment function τ and scoring function s , and given (T, Ψ) point-isolating over R , the Peer Neighborhood extension mechanism $\tau_R^\Psi(r, rr)$ is incentive-compatible with respect to the update condition $S_{PI}^{(T, \Psi)}$.

Proof. Let $f(rr) = 0$ in τ w.l.g. Suppose $\tau_R^\Psi(r, rr)$ is not incentive-compatible w.r.t $S_{PI}^{(T, \Psi)}$. Then there exists an update process $\pi_o = \mathcal{U}(R, o)$ which satisfies $S_{PI}^{(T, \Psi)}$, but the expected payment $\mathbb{E}_{rr \sim \pi_o}[\tau_R^\Psi(o, rr)] < \mathbb{E}_{rr \sim \pi_o}[\tau_R^\Psi(x, rr)]$ for some $x \neq o$. This implies that

$$\begin{aligned} & \mathbb{E}_{\theta \sim \Psi}[\mathbb{E}_{rr \sim \pi_o}[\tau_R^\theta(o, rr)]] \\ & < \mathbb{E}_{\theta \sim \Psi}[\mathbb{E}_{rr \sim \pi_o}[\tau_R^\theta(x, rr)]] \\ \Rightarrow & \mathbb{E}_{\theta \sim \Psi}[\pi_o^\theta(\mathbb{X}_\theta(o)) * s_{R^\theta}(\mathbb{X}_\theta(o))] \\ & < \mathbb{E}_{\theta \sim \Psi}[\pi_o^\theta(\mathbb{X}_\theta(x)) * s_{R^\theta}(\mathbb{X}_\theta(x))] \end{aligned}$$

The partition space (T, Ψ) being point-isolating implies that $\Psi(\{\theta : \mathbb{X}_\theta(o) \neq \mathbb{X}_\theta(x)\}) > 0$, so there is a set of θ with positive probability in Ψ such that $\Psi(\{\theta : \pi_o^\theta(\mathbb{X}_\theta(o)) * s_{R^\theta}(\mathbb{X}_\theta(o)) < \pi_o^\theta(\mathbb{X}_\theta(x)) * s_{R^\theta}(\mathbb{X}_\theta(x))\}) > 0$. It follows directly that for this set of θ , $S^*(R^\theta, \pi_o^\theta)$ is false, violating our assumption that $\mathcal{U}(R, o)$ satisfies $S_{PI}^{(T, \Psi)}$. \square

While this update condition clearly guarantees incentive-compatibility of the Peer Neighborhood extension mechanism, it is often stronger than necessary, and we will see that in some cases it excludes simple update processes that have a truthful Bayes-Nash Equilibrium. We present a more relaxed update condition:

Definition 4.2.16 (Partition-Expected Update Condition). Given a prior π , a posterior π_o , a partition space (T, Ψ) , and a Peer Consistency mechanism with scoring function s , the *Partition-Expected* (PE) update condition $S_{PE}^{(T, \Psi)}$ takes the form:

$$\begin{aligned} \forall x \neq o : & \mathbb{E}_{\theta \sim \Psi}[\pi_o^\theta(\mathbb{X}_\theta(o)) * s_{\pi^\theta}(\mathbb{X}_\theta(o))] \\ & > \mathbb{E}_{\theta \sim \Psi}[\pi_o^\theta(\mathbb{X}_\theta(x)) * s_{\pi^\theta}(\mathbb{X}_\theta(x))] \end{aligned}$$

Proposition 4.2.17. *Given a Peer Consistency mechanism with payment function τ and scoring function s , and given (T, Ψ) point-isolating over R , the Peer Neighborhood extension mechanism $\tau_R^\Psi(r, rr)$ is incentive-compatible with respect to the update condition $S_{PE}^{(T, \Psi)}$.*

Proof. This proof follows directly from the previous proof, in which we showed that if $\tau_R^\Psi(r, rr)$ is not incentive compatible with respect to $S_{PE}^{(T, \Psi)}$, then

$$\begin{aligned} \exists x \neq o : \mathbb{E}_{\theta \sim \Psi} [\pi_o^\theta(\mathbb{X}_\theta(o)) * s_{R^\theta}(\mathbb{X}_\theta(o))] \\ < \mathbb{E}_{\theta \sim \Psi} [\pi_o^\theta(\mathbb{X}_\theta(x)) * s_{R^\theta}(\mathbb{X}_\theta(x))] \end{aligned}$$

which violates the assumption that the update $\pi_o = \mathcal{U}(R, o)$ satisfies $S_{PE}^{(T, \Psi)}$. \square

Furthermore, we show that the PE condition is a relaxed form of the PI condition, in that any update process which satisfies PI also satisfies PE.

Lemma 4.2.18. *Given a partition space (T, Ψ) that is point-isolating over R , and a Peer Consistency mechanism with scoring function s , any update process $\mathcal{U}(R, o)$ which satisfies the PI extended update condition $S_{PI}^{(T, \Psi)}$ also satisfies the PE extended update condition $S_{PE}^{(T, \Psi)}$.*

Proof. Let $\pi_o = \mathcal{U}(\pi, o)$, $i = \mathbb{X}_\theta(o)$.

$$\begin{aligned} S_{PI}^{(T, \Psi)}(\pi, \pi_o) \\ \Rightarrow \Psi(\{\theta : S^*(\pi^\theta, \pi_o^\theta)\}) = 1 \\ \Rightarrow \Psi(\{\theta : \forall j \neq i : \pi_o^\theta(i) * s_{R^\theta}(i) > \pi_o^\theta(j) * s_{R^\theta}(j)\}) = 1 \end{aligned}$$

Let $j_x = \mathbb{X}_\theta(x)$. Suppose $S_{PE}^{(T, \Psi)}$ is not satisfied, then $\forall x \neq o : \mathbb{E}_{\theta \sim \Psi} [\pi_o^\theta(i) * s_{R^\theta}(i)] \leq \mathbb{E}_{\theta \sim \Psi} [\pi_o^\theta(j_x) * s_{R^\theta}(j_x)]$. Then either $\Psi(\{\theta : \forall x \neq o, j_x = i\}) = 1$, which contradicts the assumption that the partition space (T, Ψ) is point-isolating, or $\Psi(\{\theta : \forall x \neq o : j_x \neq i, \pi_o^\theta(i) * s_{R^\theta}(i) < \pi_o^\theta(j_x) * s_{R^\theta}(j_x)\}) > 0$. \square

4.3 Analysis of Update Processes

We have constructed a framework that extends Peer Consistency mechanisms to arbitrary distributions, but the crux of this extension is the strengthened update condition PE. We will examine what types of update processes satisfy this condition, but we must first address a practical concern which will restrict our update processes, namely whether or not an update process is consistent with convergence of the posterior to the true distribution.

4.3.1 Update Convergence

When an Agent makes an observation and computes a posterior according to some update process, that process should generally bring the Agent's estimate closer to the true distribution. With finite observations, it is always possible that an Agent can observe a very unlikely sequence, leading to a bias in the posterior. But in the limit of infinite observations, the posterior should converge to the true distribution. We then wish to describe update processes which can be performed iteratively to converge to the true distribution.

Definition 4.3.1 (Convergent Update Process Sequence). Consider a sequence of update processes \mathcal{U}_i for all $i \in \mathbb{Z}_+$. The sequence is *convergent* if, when the sequence $\{\mathcal{U}_i\}$ is applied iteratively to a sequence of i.i.d. observations $\{o_i\}$ sampled from the true distribution, the sequence of posteriors converges in distribution to the true distribution.

In order to get a better grasp on such update processes, we will restrict ourselves to a particular type of update process, which we call *additive*:

Definition 4.3.2 (Additive Update). An update process $\pi_o = \mathcal{U}(\pi, o)$ is *additive* if $\pi_o = (1 - \alpha) * \pi + \alpha * K_o$ where K_o is a probability measure which we call the *update kernel*, and $\alpha \in (0, 1)$.

Often we will refer to an update process of this form simply by referring to the update kernel. The Agent picks $(1 - \alpha)$ to represent the Agent's confidence in the accuracy of its prior.

Suppose an Agent were to observe a sequence of samples from the true distribution, and update after observing each sample. Assuming the samples are i.i.d. from the true distribution, it would be unreasonable for the Agent to treat the sequence differently than if they had seen the observations in any other order, so all the update kernels should be given equal weight. Given some additive update with α_1 , the next update must then have $\alpha_2 = \frac{\alpha_1}{1 + \alpha_1}$. A simple choice for α_1 would be $\frac{1}{k}$ for some positive integer k , so $\alpha_2 = \frac{1}{k+1}$. This process of decreasing α like $\frac{1}{n}$ can be applied iteratively, and we refer to this as a *linear additive* update process:

Definition 4.3.3 (Linear Additive Update). An update process $\pi_o = \mathcal{U}_k(\pi, o)$ is *linear additive* if $\pi_o = \frac{k}{k+1}\pi + \frac{1}{k+1}K_o$. The linear additive update sequence is defined for a sequence of n observations $\{o_i\}$: $\pi_{\{o_i\}} = \mathcal{U}_{k+n-1}(\mathcal{U}_{k+n-2}(\dots \mathcal{U}_k(\pi, o_1) \dots, o_{n-1}), o_n) = \frac{k}{k+n}\pi + \frac{1}{k+n} \sum_{i=1}^n K_{o_i}$.

First we show that the convergence of this update process sequence depends only on the structure of the update kernels. We say this update process sequence is *prior agnostic*:

Lemma 4.3.4. *The linear additive update process sequence converges in distribution to the same distribution as the average of the kernels: $\frac{1}{n} \sum_{i=1}^n K_{o_i} \xrightarrow{d} X \iff \frac{k}{k+n}\pi + \frac{1}{k+n} \sum_{i=1}^n K_{o_i} \xrightarrow{d} X$.*

Proof. Assume $\frac{1}{n} \sum_{i=1}^n K_{o_i} \xrightarrow{d} X$, then $\forall x, \lim_{n \rightarrow \infty} |\frac{1}{n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x)| = 0$. Then $\forall x$:

$$\begin{aligned} & \left| \frac{k}{k+n} F_\pi(x) + \frac{1}{k+n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x) \right| \\ & \leq \left| \frac{k}{k+n} F_\pi(x) \right| + \left| \frac{1}{k+n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x) \right| \\ & \leq \left| \frac{k}{k+n} F_\pi(x) \right| + \left| \frac{1}{n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x) \right| \end{aligned}$$

We have that $F_\pi(x)$ is bounded in $[0, 1]$, so $\lim_{n \rightarrow \infty} |\frac{k}{k+n} F_\pi(x)| + |\frac{1}{n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x)| = \lim_{n \rightarrow \infty} |\frac{k}{k+n} F_\pi(x)| \leq \lim_{n \rightarrow \infty} |\frac{k}{k+n}| = 0$. Therefore, $\frac{k}{k+n} \pi + \frac{1}{k+n} \sum_{i=1}^n K_{o_i} \xrightarrow{d} X$.

For the other direction, first we note that $\frac{1}{n} \sum_{i=1}^n F_{K_{o_i}}(x) = \frac{k}{n(k+n)} F_{K_{o_i}}(x) + \frac{1}{k+n} F_{K_{o_i}}(x) \leq \frac{k}{n(k+n)} + \frac{1}{k+n} F_{K_{o_i}}(x)$.

Assume $\forall x, \lim_{n \rightarrow \infty} |\frac{k}{k+n} F_\pi(x) + \frac{1}{k+n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x)| = 0$. Note that $\lim_{n \rightarrow \infty} |\frac{k}{k+n} F_\pi(x)| \leq \lim_{n \rightarrow \infty} |\frac{k}{k+n}| = 0$, so $\lim_{n \rightarrow \infty} |\frac{1}{k+n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x)| = 0$. Then $\forall x$:

$$\begin{aligned} & \left| \frac{1}{n} \sum_{i=1}^n F_{K_{o_i}}(x) - F_X(x) \right| \\ & = \left| \frac{k}{n(k+n)} F_{K_{o_i}}(x) + \frac{1}{k+n} F_{K_{o_i}}(x) - F_X(x) \right| \\ & \leq \left| \frac{k}{n(k+n)} F_{K_{o_i}}(x) \right| + \left| \frac{1}{k+n} F_{K_{o_i}}(x) - F_X(x) \right| \end{aligned}$$

The limit of this expression is 0, so $\frac{1}{n} \sum_{i=1}^n K_{o_i} \xrightarrow{d} X$. \square

We now address the structure of the update kernel K . We will consider two types of kernels, the first is a simple point mass. We call this the *Empirical Update*:

Definition 4.3.5 (Empirical Update). The *Empirical Update* is the additive update where $K_o(A) = \mathbb{1}_{o \in A}$.

Proposition 4.3.6. *The linear additive Empirical Update sequence is convergent.*

Proof. Let $\mathcal{E}_{\{o_i\}} = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{o_i \in A}$, be the *Empirical Measure*. The Empirical Measure converges in distribution to the true distribution, from the Glivenko-Cantelli theorem (Cantelli, 1933; Glivenko, 1933). It follows directly from Lemma 4.3.4 that the linear additive Empirical Update process converges in distribution to the true distribution. \square

The second type of kernel we wish to address is a kernel with a continuous cumulative distribution function (CDF). We call such updates *continuous*:

Definition 4.3.7 (Continuous Additive Update). An additive update is *continuous* if the CDF F_K of the update kernel K is continuous.

We show that a linear additive continuous update process sequence is convergent if the sequence of kernels satisfies a condition on the partial sums of their concentrations around the observed samples:

Theorem 4.3.8. *Let $O_n = \{o_i\}_{i \in [1, n]}$ be a sequence of i.i.d random variables distributed with CDF $F(x)$. Let K_{o_i} be a continuous update kernel. Define $H_n(x) = \frac{1}{n} \sum_{i=1}^n F_{K_{o_i}}(x)$. Consider the random variables $Y_i = |X_i - o_i|$ where X_i is distributed according to K_{o_i} . Define $C_i(\epsilon) = P(Y_i \geq \epsilon)$. If $\forall \epsilon > 0 : \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) = 0$, then $H_n \xrightarrow{d} F$.*

Proof. Let \mathcal{E}_{O_n} be the Empirical Measure. Assuming that $\forall \epsilon > 0, \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) = 0$, we calculate upper and lower bounds on the partial sums of the CDFs of the kernels $H_n(x)$:

$$\begin{aligned}
H_n(x) &\geq \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{o_i < x - \epsilon\}} (1 - C_i(\epsilon)) \\
&\geq \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{o_i < x - \epsilon\}} - \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) \\
&= F_{\mathcal{E}_{O_n}}(x - \epsilon) - \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) \\
&\Rightarrow \liminf_n H_n(x) \geq \liminf_n F_{\mathcal{E}_{O_n}}(x - \epsilon) \\
&\Rightarrow \liminf_n H_n(x) \geq F(x) \text{ at continuity points}
\end{aligned}$$

Symmetrically:

$$\begin{aligned}
H_n(x) &\leq \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{o_i > x + \epsilon\}} C_i(\epsilon) \\
&\leq \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) \\
\Rightarrow 1 - H_n(x) &\geq 1 - \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) \\
&\geq \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{o_i > x + \epsilon\}} - \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) \\
&= 1 - F_{\mathcal{E}_{O_n}}(x + \epsilon) - \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) \\
&\Rightarrow \liminf_n (1 - H_n(x)) \geq \liminf_n (1 - F_{\mathcal{E}_{O_n}}(x + \epsilon)) \\
&\Rightarrow \limsup_n H_n(x) \leq F(x) \text{ at continuity points}
\end{aligned}$$

Therefore $H_n(x) \xrightarrow{d} F(x)$. □

The convergence of the linear additive continuous update process sequence with kernels satisfying this condition follows directly from Lemma 4.3.4.

We present a very simply condition on the kernels which will satisfy Theorem 4.3.8. The kernels merely need to have bounded support, with that bound converging to 0.

Corollary 4.3.9. *Let $\Delta_i = \langle \delta_{i,1}, \delta_{i,2}, \dots, \delta_{i,d} \rangle$ with $\delta_{i,j} > 0$ and $\lim_{i \rightarrow \infty} \delta_{i,j} = 0$. Let $A_i = [o_i - \Delta_i, o_i + \Delta_i]$. Suppose $K_{o_i}(A) = 1$. Then a linear additive continuous update process sequence with these kernels is convergent.*

Proof. Define X_i as the random variables distributed according to K_{o_i} and $Y_i = |X_i - o_i|$. Given any $\epsilon > 0$, from the convergence of Δ_i , we have that $\exists N$ such that $\forall n > N$, $\delta_{n,j} < \epsilon$. So $\forall i > N$, $C_i(\epsilon) = 0$. Therefore $\lim_{n \rightarrow \infty} \sum_{i=1}^n C_i(\epsilon) = \sum_{i=1}^N C_i(\epsilon) \Rightarrow \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n C_i(\epsilon) = 0$. From Theorem 4.3.8, the sequence of averages of the CDFs of the kernels $H_n(x) \xrightarrow{d} F(x)$. It then follows directly from Lemma 4.3.4 that the update process is convergent. \square

4.3.2 Satisfying the Update Conditions

We now analyze update processes which satisfy our extended update conditions PI and PE. Whether or not these conditions are satisfied depends heavily on the choice of scoring function s . We choose to focus our attention to the Peer Neighborhood extension of the PTS, commonly considered the canonical example of a Peer Consistency mechanism. We will then refer to the extension as the *Peer Truth Neighborhood Extension* mechanism.

Definition 4.3.10 (Peer Truth Neighborhood Extension). The *Peer Truth Neighborhood Extension* (PTNE) mechanism is the Peer Neighborhood extension of the Peer Consistency mechanism with scoring function $s_R(r) = \frac{c}{R(r)}$ where c is a positive constant, known as the Peer Truth Serum.

The natural update condition for the PTS is $\forall x \neq o : \frac{\pi_o(o)}{\pi(o)} > \frac{\pi_o(x)}{\pi(x)}$, known as the *self-predicting* update condition.

We first prove that the Empirical Update satisfies PI for the PTNE, and therefore also PE:

Theorem 4.3.11. *Given a partition space (T, Ψ) that is point-isolating and bin-supported over R , the Empirical Update process satisfies $S_{PI}^{(T, \Psi)}$ for the PTNE mechanism.*

Proof. The Empirical Update process yields $\pi_o = (1 - \alpha) * R + \alpha * K_o$. Let $i = \mathbb{X}_\theta(o)$. Then $\forall \theta :$
 $\pi_o^\theta = (1 - \alpha) * R^\theta + \alpha * K_o^\theta$ where $K_o^\theta(j) = \mathbb{1}_{j=i}$. Then $\frac{\pi_o^\theta(j)}{R^\theta(j)} = (1 - \alpha) + \alpha \frac{K_o^\theta(j)}{R^\theta(j)} = \begin{cases} (1 - \alpha) + \frac{\alpha}{R^\theta(j)} & j = i \\ (1 - \alpha) & j \neq i \end{cases}$.

The assumption that (T, Ψ) is bin-supported over R ensures that $R^\theta(j) > 0$. Therefore, $\forall \theta :$
 $\forall j \neq i : \frac{\pi_o^\theta(i)}{R^\theta(i)} > \frac{\pi_o^\theta(j)}{R^\theta(j)}$. \square

The Empirical Update is perfectly reasonable and is used quite frequently in modeling, but an Agent may want to make a continuous update as they may be unsure that their measurement of the true distribution is precise. But with additive continuous updates, whether or not they satisfy our update conditions depends on the structure of the partition space. We choose to focus on partition spaces with a high degree of symmetry, which we call *regular rectangular*:

Definition 4.3.12 (Regular Rectangular Partition Space). A *regular rectangular* partition space (T, Ψ) over a fundamental set $\Omega = \mathbb{R}^d$ is one in which each bin is a rectangular prism with side lengths $\{l_i\}_{i \in [1, d]}$, i.e. $\forall \theta \in \Theta$, $T(\theta) = \{\otimes_{i=1}^d [l_i * (n_i - \frac{1}{2}) + \theta_i, l_i * (n_i + \frac{1}{2}) + \theta_i] \forall n_i \in \mathbb{Z}\}$ and $\Theta = \otimes_{i=1}^d [0, l_i)$, with Ψ uniform over Θ .

This partition space is clearly point-isolating over any R as it is point isolating for all $\omega \in \Omega$. We will assume that R is such that this partition space is bin-supported over R .

4.3.3 Bin Edge Conditions

Let us consider some regular rectangular partition space. Let each bin have dimensions $L = \langle l_1, l_2, \dots, l_d \rangle$. To simplify the notation, we will say the set $[-L, L) = \otimes_{i=1}^d [-l_i, l_i)$. We define the Bin Function $B : \mathbb{R}^d \rightarrow \{0, 1\}$ to be:

$$B(\omega) = \begin{cases} 1 & \omega \in [-\frac{L}{2}, \frac{L}{2}) \\ 0 & \text{otherwise} \end{cases}$$

so the Bin Function is just an indicator for a bin centered at 0.

Assume that an agent with prior and posterior π and π_o respectively has PDFs f_π and f_{π_o} . It's not necessary that such PDFs exist, but we make this assumption for ease of presentation. Let us define the overhead \sim to be the operator such that for a function f , $\tilde{f}(x) = (f \circledast B)(x)$, where \circledast is the convolution operator. Then $\tilde{f}_\pi(x)$ is just the prior probability of a sample landing in a bin centered at x , and same for the posterior $\tilde{f}_{\pi_o}(x)$. We see that these functions can be computed only using the CDFs, but it is useful to define them this way. The quantities we are concerned with regarding the PI and PE conditions for the PTNE mechanism are the ratios $Q(x) = \frac{\tilde{f}_{\pi_o}(x)}{\tilde{f}_\pi(x)}$ and the expected payment for reporting x is simply $\tilde{Q}(x)$. If the update process is additive continuous, then Q and \tilde{Q} are continuous.

The PI condition gives us the following constraint. Let $N = \langle n_1, n_2, \dots, n_d \rangle$ where $n_i \in \mathbb{Z}$ and $N \neq 0$. Then $\forall x \in (o - \frac{L}{2}, o + \frac{L}{2}) : Q(x) > Q(x + N * L)$ where $*$ is element-wise multiplication. Let $Q_o(x) = Q(o + x)$. From the continuity of Q , it follows that for all $i \in [1, d]$ and all $\delta_i \in [-\frac{l_i}{2}, \frac{l_i}{2}]$:

$$Q_o(\delta_1, \dots, -\frac{l_i}{2}, \dots, \delta_d) = Q_o(\delta_1, \dots, \frac{l_i}{2}, \dots, \delta_d) \quad (4.3)$$

We see that these are equalities on every pair of opposing points on the boundary of the bin centered at o .

The PE condition simply constrains o to be the global maximum of \tilde{Q} . As long as the continuous update kernel has mass at o and has sufficiently bounded support, if o is a local maximum of \tilde{Q} , then it will be the global maximum. We'll discuss what sufficiently bounded means later. We write the PE constraint:

$$\nabla_x \tilde{Q}|_{x=o} = 0, \quad \nabla_x^2 \tilde{Q}|_{x=o} < 0.$$

From the continuity of \tilde{Q} we obtain conditions that are much less restrictive than for PI. Let L_{-i} be the vector L with entry l_i removed, and Δ_i be the vector of δ_j s with entry δ_i removed. Then $\forall i \in [1, d]$:

$$\begin{aligned} & \int_{-\frac{L_{-i}}{2}}^{\frac{L_i}{2}} Q_o(\delta_1, \dots, -\frac{l_i}{2}, \dots, \delta_d) \partial \Delta_i \\ &= \int_{-\frac{L_{-i}}{2}}^{\frac{L_i}{2}} Q_o(\delta_1, \dots, \frac{l_i}{2}, \dots, \delta_d) \partial \Delta_i \end{aligned} \quad (4.4)$$

We see that rather than having an equality for every pair of opposing points on the boundary of the bin centered at o , we have a single equality for each opposing boundary surface of the bin. This is equivalent to the constraints for PI in one dimension, since the opposing boundary surfaces are just a single pair of points, but in higher dimensions it is much less constraining.

We also see that a continuous update kernel has "sufficiently bounded support" if it has support within $(o - L, o + L]$. From now on we will refer to such an update kernel as *bin-bounded*.

Failing the Partition-Invariant Update Condition

We will first show that it is impossible in general for a bin-bounded continuous update kernel to satisfy both PI and for the associated linear additive update process sequence to be convergent in dimensions higher than one. We will show the proof for two dimensions, but the same argument applies to higher dimensions.

Lemma 4.3.13. *Given a regular rectangular partition space on \mathbb{R}^2 , let each bin have dimensions $L = \langle l_1, l_2 \rangle$. Let $\Delta = \langle \frac{l_1}{2}, \frac{l_2}{2} \rangle$, and $A = [z - \Delta, z + \Delta]$. There is a prior π such that a continuous update kernel must have bounded probability on A : $K_o(A) < x < 1$ in order to satisfy PI for the PTNE mechanism.*

Proof. We first note that if two or more bins β_1 and β_2 must have equal ratios of posterior to

prior, it must have equal ratios of kernel to prior:

$$\begin{aligned} \frac{\pi_o(\beta_1)}{\pi(\beta_1)} &= \frac{\pi_o(\beta_2)}{\pi(\beta_2)} \\ \Rightarrow (1 - \alpha) + \alpha \frac{K_o(\beta_1)}{\pi(\beta_1)} &= (1 - \alpha) + \alpha \frac{K_o(\beta_2)}{\pi(\beta_2)} \\ \Rightarrow \frac{K_o(\beta_1)}{\pi(\beta_1)} &= \frac{K_o(\beta_2)}{\pi(\beta_2)} \end{aligned}$$

If we place the corner of four bins on the observation point z , let the prior probabilities of the four bins be $A_{r,u}, A_{r,b}, A_{l,u}, A_{l,b}$ corresponding to the upper-right, bottom-right, upper-left, and bottom-left corners. Consider also a bin boundary on o such that the left bin is $\beta_l = [(o_1 - l_1, o_2 - \frac{l_2}{2}), (o_1, o_2 + \frac{l_2}{2})]$ and the right bin is $\beta_r = [(o_1, o_2 - \frac{l_2}{2}), (o_1 + l_1, o_2 + \frac{l_2}{2})]$. Consider a prior such that $\pi(\beta_l) = A_{l,u} + A_{l,b}$ and $\pi(\beta_r) = r(A_{r,u} + A_{r,b})$ where we can construct the prior so r takes on any value in $[0, 1]$.

Now consider the probabilities of the kernel in the four corners of A : $A_{r,u}^*, A_{r,b}^*, A_{l,u}^*, A_{l,b}^*$. PI requires that $(A_{r,u}^*, A_{r,b}^*, A_{l,u}^*, A_{l,b}^*) = \lambda_1(A_{r,u}, A_{r,b}, A_{l,u}, A_{l,b})$. We apply the same PI constraint to the centered left and right bins: $K_o(\beta_l \cap A) = \lambda_2(A_{l,u} + A_{l,b})$ and $K_o(\beta_r \cap A) = \lambda_2 r(A_{r,u} + A_{r,b})$ with $\lambda_2 < \lambda_1$. Suppose that at most $1 - x$ fraction of the kernel probability is outside A . Then we have the following inequalities:

$$\begin{aligned} (1 - x)\lambda_1(A_{l,u} + A_{l,b}) &\geq (\lambda_1 - \lambda_2)(A_{l,u} + A_{l,b}) \geq 0 \\ (1 - x)\lambda_1(A_{r,u} + A_{r,b}) &\geq (\lambda_1 - \lambda_2 r)(A_{r,u} + A_{r,b}) \geq 0 \\ \Rightarrow (1 - x) &\geq 1 - \frac{\lambda_2}{\lambda_1} r > 0 \\ \Rightarrow x &\leq \frac{\lambda_2}{\lambda_1} r < 1 \end{aligned}$$

□

With this we can prove that a continuous update kernel cannot allow for a linear additive update process sequence that is convergent:

Theorem 4.3.14. *Given a regular rectangular partition space on \mathbb{R}^2 , there is a prior π and true distribution Φ such that a continuous update kernel cannot satisfy PI for the PTNE mechanism and admit a linear additive update process sequence that is convergent.*

Proof. From Lemma 4.3.13, there is a prior π such that, in order to satisfy PI, there exists a Δ such that if $A = [z - \Delta, z + \Delta]$, the kernel value $K_o(A)$ is uniformly bounded above. This uniform bound is itself bounded above by $r < 1$, defined in Lemma 4.3.13 as the ratio of the prior probabilities in the two bins to the right of o to the prior probabilities in the two bins to the left of o . In order for the additive update with K_o to satisfy PI, this ratio r must be invariant with respect to the linear update process sequence. Therefore, if the Agent wishes

to update over a sequence of observations O_n with a sequence of kernels $\{K_{o_i}\}$ such that $\pi_i = \frac{k+i-1}{k+i}\pi_{i-1} + \frac{1}{k+i}K_{o_i}$, then all the kernels have the value $K_{o_i}(A)$ uniformly bounded above by r . Then $\lim_{n \rightarrow \infty} \pi_n(A) \leq \limsup_i K_{o_i}(A) \leq r < 1$. If the true distribution Φ is more heavily concentrated inside A , such that $\Phi(A) = r\bar{\Phi} > r$. Then $\lim_{n \rightarrow \infty} |F_{\pi_n}(x) - F_{\bar{\Phi}}(x)|$ is uniformly bounded below by $r\bar{\Phi} - r > 0$. Therefore, this update process sequence cannot satisfy the condition in Theorem 4.3.8, and therefore cannot be convergent. \square

Satisfying the Partition-Expected Update Condition

We will now show that it is always possible to construct a sequence of continuous update kernels that satisfy both PE and are convergent. We will construct these explicitly. First we will restrict our construction so that all the probability of the kernel is within a bounded region $A = [x - \Delta, x + \Delta]$ which contains the observation point o , and where Δ can be arbitrarily small. From Corollary 4.3.9, we find that by allowing the sequence Δ_i to converge to 0, this update process sequence will be convergent. Thus it is sufficient to show that our kernel construction satisfies PE.

Theorem 4.3.15. *Given a regular rectangular partition space on \mathbb{R}^d , for any prior π , there exists a continuous update kernel that satisfies PE for the PTNE mechanism and is arbitrarily bounded around a point o .*

Proof. We are given the following bin boundary conditions for satisfying PE:

$$\int_{-\frac{l_i}{2}}^{\frac{l_i}{2}} Q_o(\delta_1, \dots, -\frac{l_i}{2}, \dots, \delta_d) \partial \Delta_i = \int_{-\frac{l_i}{2}}^{\frac{l_i}{2}} Q_o(\delta_1, \dots, \frac{l_i}{2}, \dots, \delta_d) \partial \Delta_i \quad (4.5)$$

We will construct the kernel K_o^x as having a PDF that is a pyramid with the peak at o and the base at $[x - \Delta, x + \Delta]$ with $\Delta < L$ the dimensions of the bins. We prove that there exists an $x \in [o - \Delta, o + \Delta]$ such that the kernel satisfies PI for the PTNE mechanism. We will demonstrate the construction on \mathbb{R}^2 , but the argument is applicable to all dimensions.

Define $Q_x(\omega) = \frac{\tilde{f}_{K_o^x}(\omega)}{\tilde{f}_R(\omega)}$. We define $S(x)$ as the integrals of $Q_x(\omega)$ over the four edges of the rectangle $[o - \frac{l}{2}, o + \frac{l}{2}]$, with x being the location of the center of the base of the pyramid:

$$\begin{aligned} S_l(x) &= \int_{-\frac{l_2}{2}}^{\frac{l_2}{2}} Q_x(o + \langle -\frac{l_1}{2}, y_2 \rangle) \partial y_2 & S_r(x) &= \int_{-\frac{l_2}{2}}^{\frac{l_2}{2}} Q_x(o + \langle \frac{l_1}{2}, y_2 \rangle) \partial y_2 \\ S_b(x) &= \int_{-\frac{l_1}{2}}^{\frac{l_1}{2}} Q_x(o + \langle y_1, -\frac{l_2}{2} \rangle) \partial y_1 & S_u(x) &= \int_{-\frac{l_1}{2}}^{\frac{l_1}{2}} Q_x(o + \langle y_1, \frac{l_2}{2} \rangle) \partial y_1 \end{aligned}$$

To satisfy PE, according to the bin boundary conditions, we must find an x such that $S_l(x) = S_r(x)$ and $S_b(x) = S_u(x)$. Define $F_h(x) = S_r(x) - S_l(x)$ and $F_v(x) = S_u(x) - S_b(x)$ as the horizontal and vertical residuals. We observe that when $x_1 = o_1 - \delta_1$, $F_h < 0$, and when $x_1 = o_1 + \delta_1$,

$F_h > 0$. Similarly, when $x_2 = o_2 - \delta_2$, $F_v < 0$, and when $x_2 = o_2 + \delta_2$, $F_v > 0$. The two functions $F_h(x)$ and $F_v(x)$ satisfy the assumptions laid out in the Poincaré-Miranda Theorem on the box $[o - \Delta, o + \Delta]$ Miranda, 1940. Therefore, there exists an x in the box such that $F_h(x) = 0$ and $F_v(x) = 0$, thus satisfying the PE condition.

In higher dimensions, the four S functions simply correspond to integrals of Q over the faces of the rectangular hyper-prism. We can define functions F_i corresponding to the residuals in S on opposing faces in the coordinate direction i , and the Poincaré-Miranda Theorem applies as before. \square

The proof further suggest a method for constructing update kernels that satisfy PE. The kernels have PDFs which are hyper-pyramids with a peak at z and a base at $[x - \Delta, x + \Delta]$ for some arbitrary positive $\Delta < L$ where L are the dimensions of the bins. Because $F_i(x)$ is monotonic in x_i and ranges from negative to positive values, the function $G(x) = \sum_{i=1}^d F_i^2(x)$ is convex and the minimizer is at an x which satisfies $\sum_{i=1}^d F_i(x) = 0$. We know such an x exists, therefore applying gradient descent to $G(x)$ is guaranteed to converge to a solution. We write the definition for these update kernels in two dimensions, as they are used in simulations.

Definition 4.3.16 (Pyramid Update Kernels in Two Dimensions). Given a regular rectangular partition space in two dimensions with bin dimensions L and given some $\Delta = \langle \delta_1, \delta_2 \rangle$ with $0 < \Delta < L$, define the pyramid function $\mathcal{P}_{x,o,\Delta}(z)$ as the height at $z \in [x - \Delta, x + \Delta] \subset \mathbb{R}^2$ of a pyramid with maximum height $h = \frac{3}{4\delta_1\delta_2}$ at a location $o \in [x - \Delta, x + \Delta]$ and a base on $[x - \Delta, x + \Delta]$. $\mathcal{P}_{x,o,\Delta}(z) = 0$ for all other $z \in \mathbb{R}^2$. Note that the pyramid has volume 1, so $\int_{\mathbb{R}^2} \mathcal{P}_{x,\Delta}(z) dz = 1$. Given some prior measure R and an observation o , compute x such that it minimizes $G(x) = F_h^2(x) + F_v^2(x)$ with F_h and F_v as defined in the proof of Theorem 4.3.15. From this theorem, a kernel measure $K_{o,\Delta}$ satisfies the PE update condition if its PDF is $\mathcal{P}_{x,o,\Delta}(z)$.

4.4 Simulations

We conduct simulations using the PTNE mechanism to demonstrate the accuracy and stability of the incentives in settings with finite data for constructing models and finite peer reports. We use artificially generated data to form the true and public distributions, which can then be used to analyze expected payments and actual payments from samples. We present two data models: 1) an Empirical distribution constructed by taking finite samples with randomized frequencies, and 2) a continuous distribution constructed as a weighted sum of Gaussian distributions, or a Gaussian Mixture Model (GMM). For the first model, Agents use the Empirical Update, while for the second they update using Pyramid kernels as described in Definition 4.3.16. In all cases, the partition space is regular rectangular.

Payment from Agent's Perspective: Ex-Ante Game

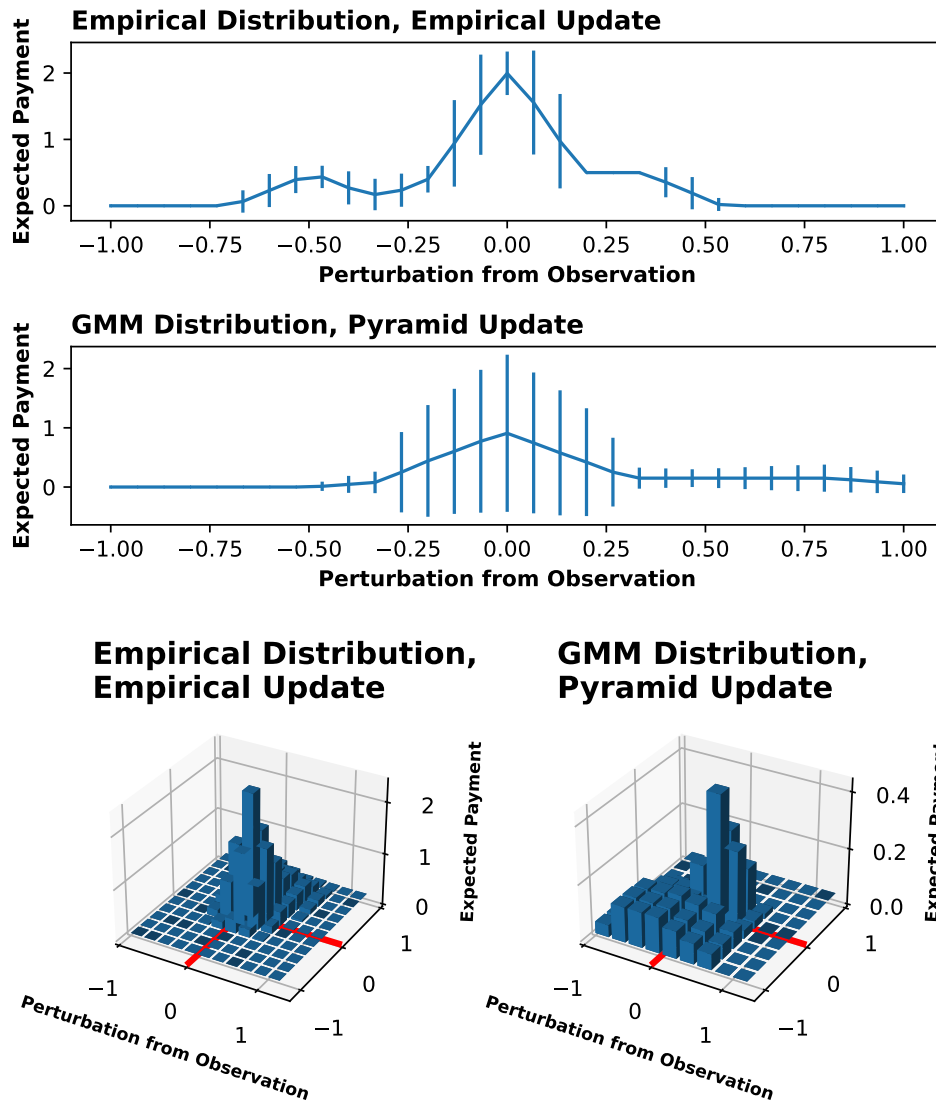


Figure 4.2: Expected payments for reports perturbed from the observation, computed over an Agent's posterior. Error bars are one standard deviation. In the 2D figures, red lines show the location of the maximum expected payment.

4.4.1 Report Perturbation

We simulate the expected payments for an Agent reporting a point that is a perturbation of the observation, meaning the payment for the observation itself is at 0. To generate the distributions, we sample 5 values uniformly in $[0, 1)$ for 1D and $[0, 1) \times [0, 1)$ for 2D. For both the True and Public Distributions, each value is weighted with an independent random variable

Payment from Center's Perspective: Ex-Post Game

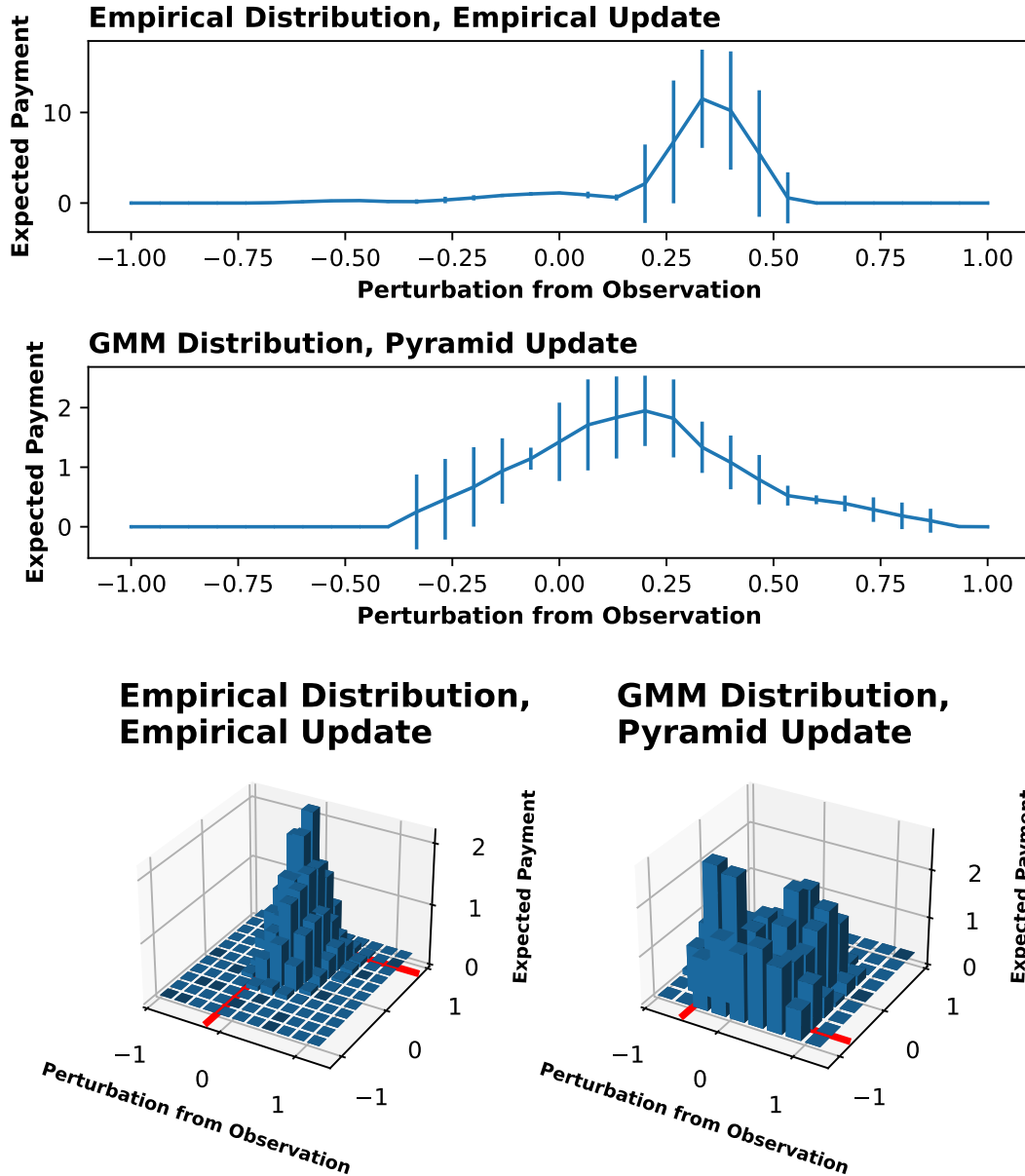


Figure 4.3: Expected payments for reports perturbed from the observation, computed over truthful Peer reports. Error bars are one standard deviation. In the 2D figures, red lines show the location of the maximum expected payment.

in $[0, 1)$, and the weight vector is normalized. The kernel is given a weight of 0.5 in the update. The bin size is 0.2 for 1D and $\sqrt{0.2} \times \sqrt{0.2}$ for 2D. The Partition Selection is just a translation by a random variable sampled uniformly from a bin volume.

Figure 4.2 shows the expected payments computed from the perspective of the Agent over the posterior. The error bars show the standard deviation with respect to the Partition Selection distribution. We observe that the Agent believes their payment will be maximized by truthfully reporting the observation, as expected from the theory. Figure 4.3 shows the same expected payments, but this time computed over a set of truthful Peer reports collected by the Center. The expected payment from the Center's side is not necessarily maximized at the observation point. Since the public distribution is different from the true distribution, the observation made by the Agent might be an over-represented point in the public distribution. If this is the case, the Agent will be underpaid when compared against Peers reporting samples from the true distribution, and some perturbation of the observation might pay better. One can visually inspect the true, public, and kernel distribution figures in Section 4.4.3 to see how the relationships between them produce the skewed figures. This does not matter for the incentives in the ex-ante game that the Agents play, however, as it is an ex-post calculation.

4.4.2 Payment Stability

Means and Variances of Payments over Bin Size

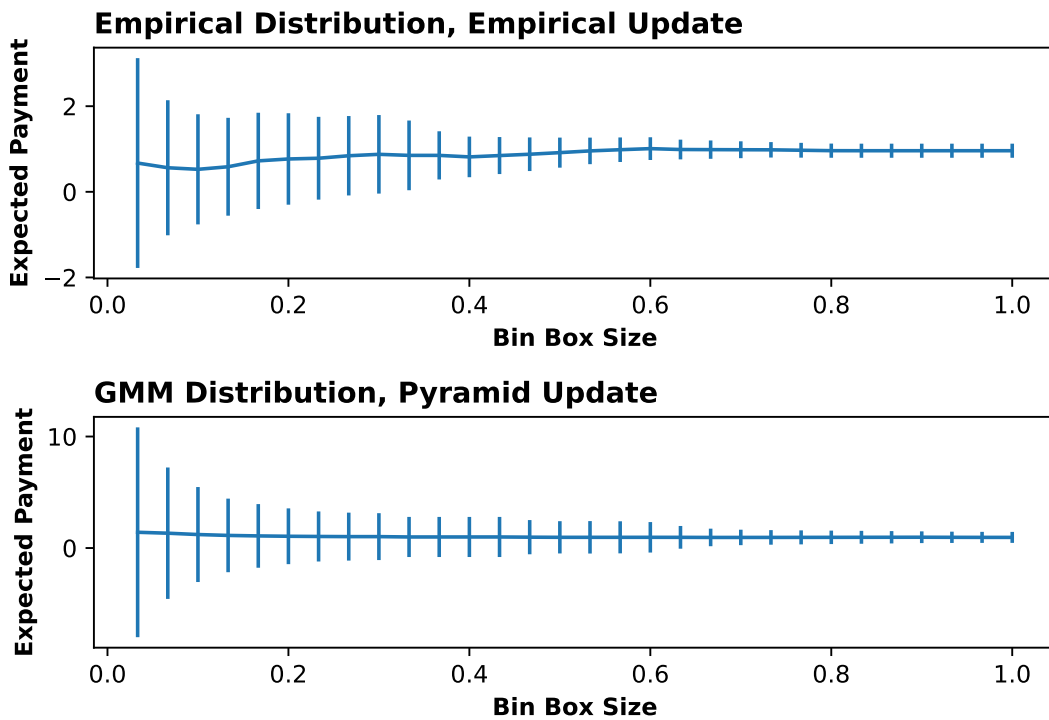


Figure 4.4: Smaller bins produce a larger variance in payments. Error bars are one standard deviation squared.

We simulate the expectation and variance of payments with respect to bin size for the partition. The distributions are all generated the same way as in the previous section, but with the

bin size varying from $\frac{1}{30}$ to 1 in intervals of $\frac{1}{30}$. The distribution figures can be found in Section 4.4.3.

The bin size can affect the expected payment of the Agent in complicated ways when you take into account that bin-bounded kernels must account for the bin size. From the perspective of the Center, however, the bin size should not affect the expected payment. A smaller bin means a lower probability of matching, but a proportionately higher payment when matching. Intuitively, a smaller bin size will lead to higher variance in the payments. We demonstrate this relationship in Figure 4.4. The stability of the payments could be a consideration for designing the mechanism to take into account either Centers or Agents who aren't risk-neutral.

4.4.3 Distributions

Empirical Distribution, Empirical Update Figures 4.5 and 4.6 for Section 4.4.1: Values and weights are treated as weighted delta functions. Expectations are taken over 200 Peer reports, 500 Partition Selection samples for 1D and 400 samples for 2D. Perturbations go from -1 to 1 in intervals of $\frac{1}{30}$ for 1D, and from $(-1, -1)$ to $(1, 1)$ in intervals of $\frac{1}{10} \times \frac{1}{10}$ for 2D.

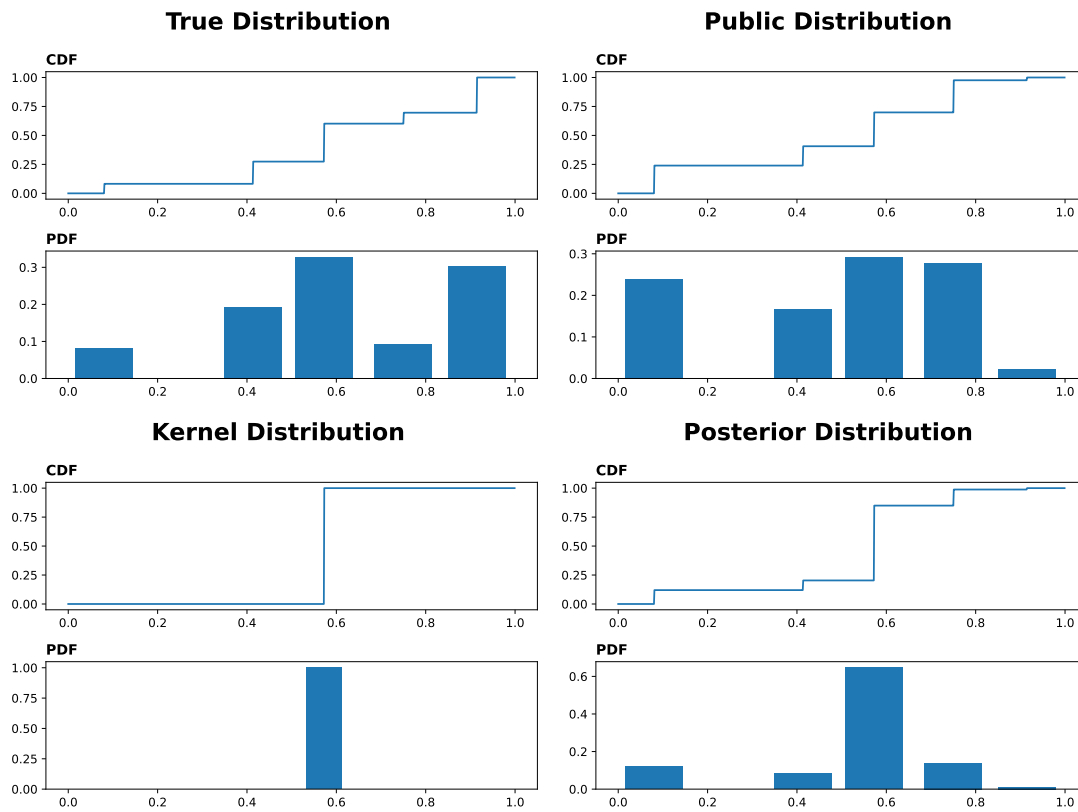


Figure 4.5: True, Public, Kernel, and Posterior distributions for 1D Empirical distribution, Empirical update perturbation simulations.

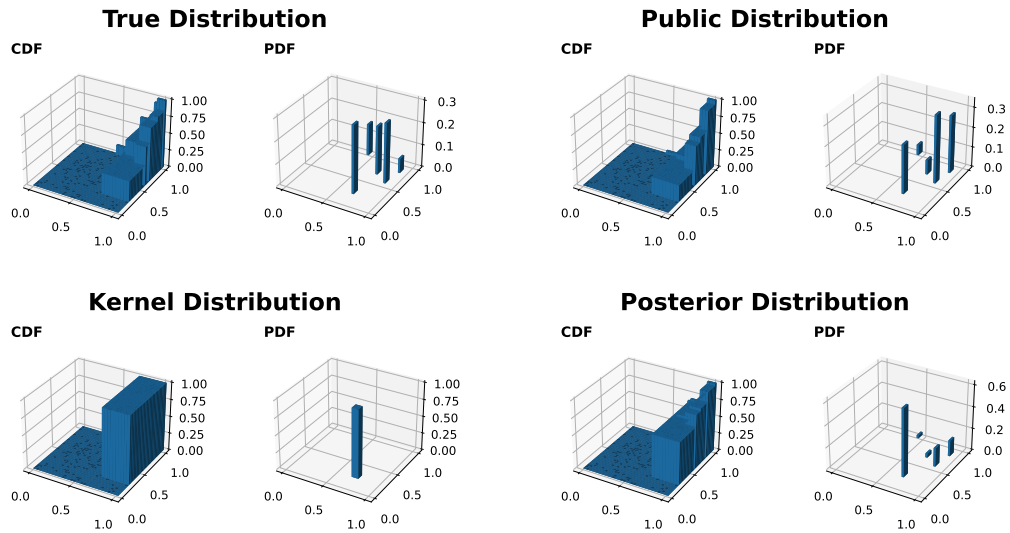


Figure 4.6: True, Public, Kernel, and Posterior distributions for 2D Empirical distribution, Empirical update perturbation simulations.

GMM Distribution, Pyramid Update Figures 4.7 and 4.8 for Section 4.4.1: Values are treated as means of Gaussian distributions. The variance in 1D is taken as $2 * \min_{i \neq j} (|V_i - V_j|)$ where V_i and V_j are from the value list. The covariance in 2D is taken as a diagonal matrix with $2 * \min_{i \neq j} (|V_i - V_j|)$ for each coordinate. The size of the Pyramid kernel base is the one tenth the bin size. Expectations are taken over 200 Peer reports, 200 Partition Selection samples for 1D and 64 samples for 2D. Perturbations go from -1 to 1 in intervals of $\frac{1}{30}$ for 1D, and from $(-1, -1)$ to $(1, 1)$ in intervals of $\frac{1}{8} \times \frac{1}{8}$ for 2D.

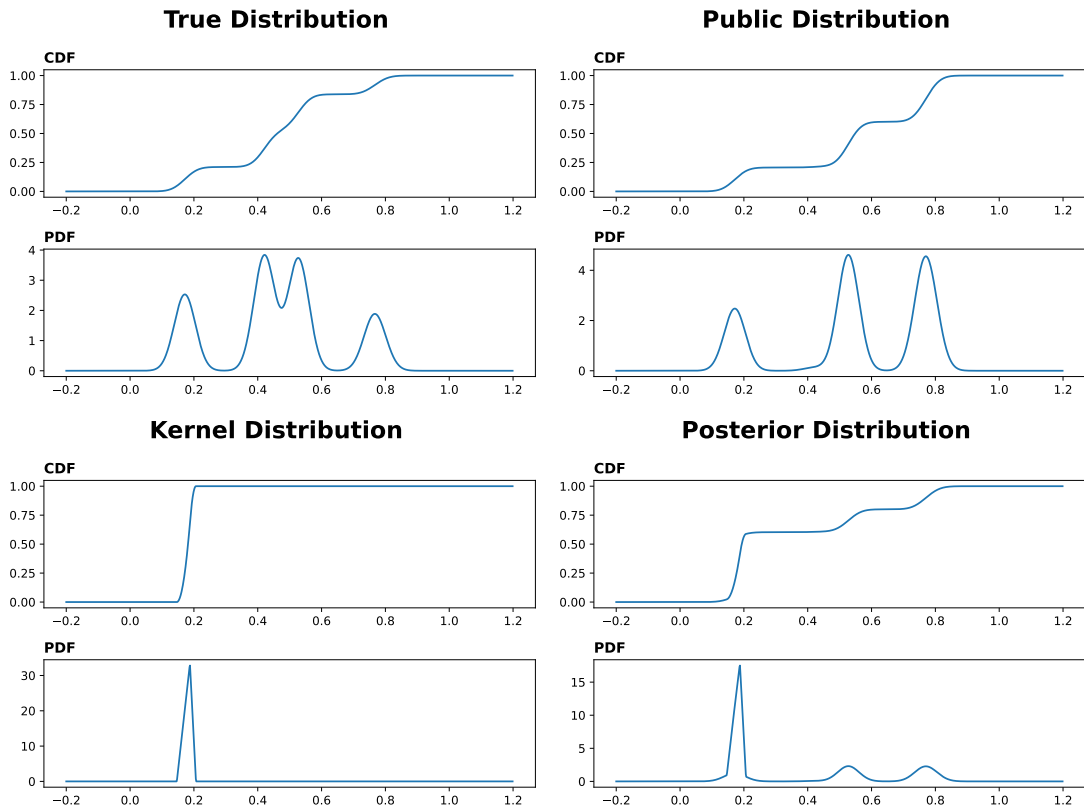


Figure 4.7: True, Public, Kernel, and Posterior distributions for 1D GMM distribution, Pyramid update perturbation simulations.

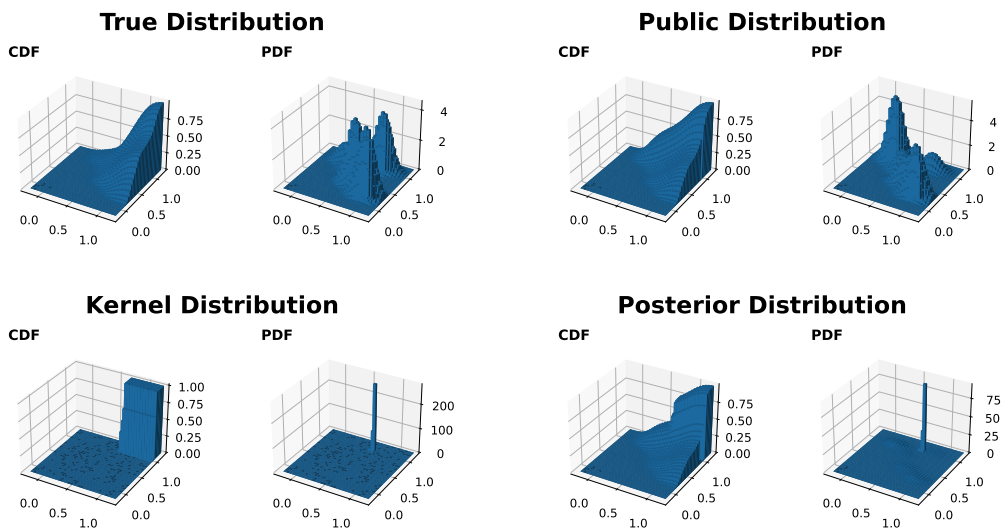


Figure 4.8: True, Public, Kernel, and Posterior distributions for 2D GMM distribution, Pyramid update perturbation simulations.

Empirical Distribution, Empirical Update Figure 4.9 for Section 4.4.2.

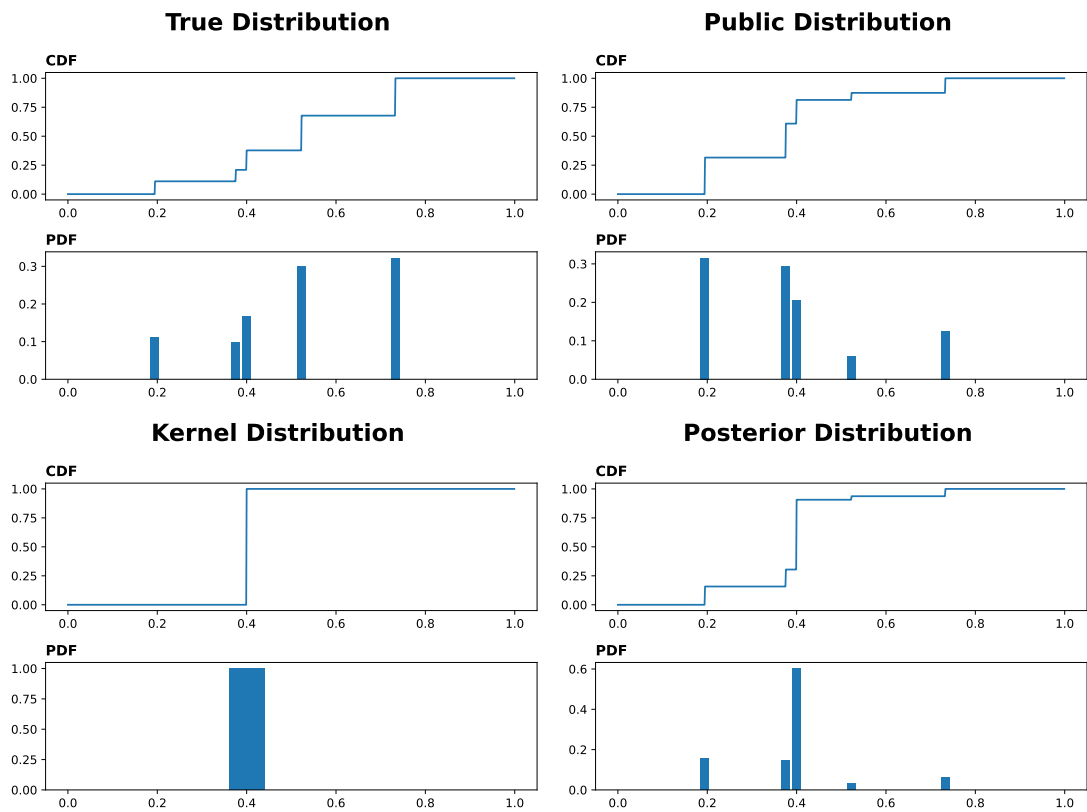


Figure 4.9: True, Public, Kernel, and Posterior distributions for Empirical distribution, Empirical update bin size simulations. The Kernel and Posterior distributions are taken with the largest bin size.

GMM Distribution, Pyramid Update Figure 4.10 for Section 4.4.2.

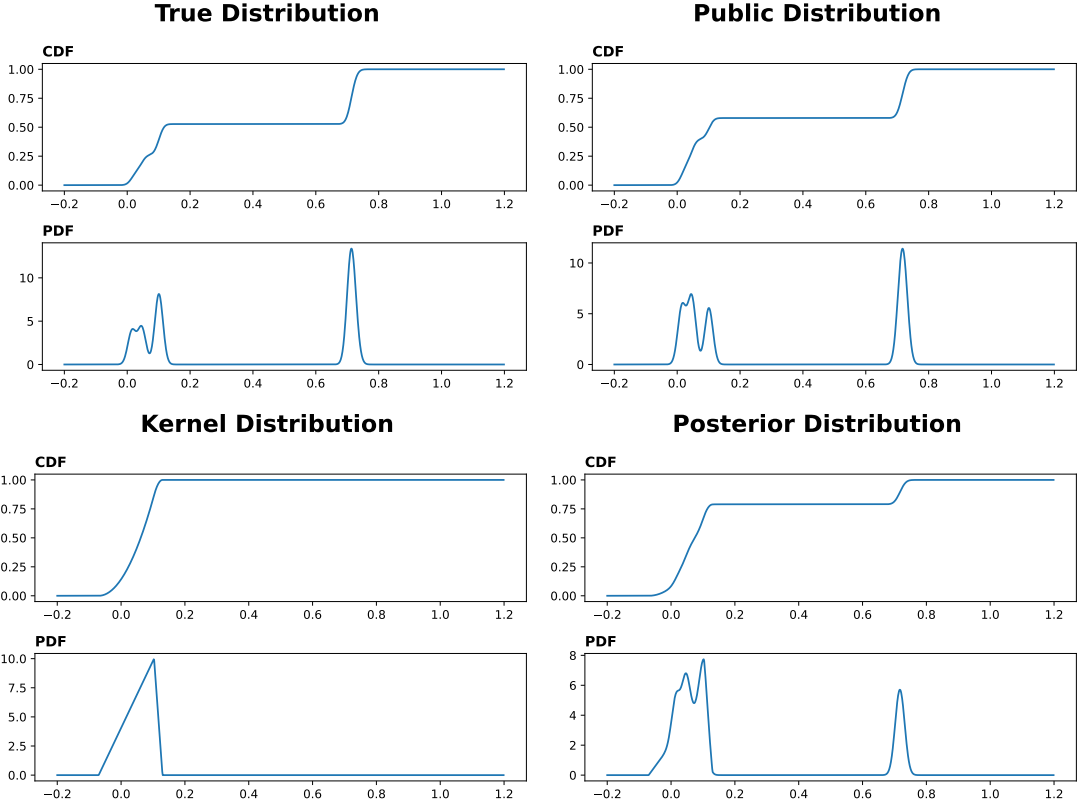


Figure 4.10: True, Public, Kernel, and Posterior distributions for GMM distribution, Pyramid update bin size simulations. The Kernel and Posterior distributions are taken with the largest bin size.

5 Continuous Truth Serum

5.1 Introduction

5.1.1 Improving Peer Neighborhoods

Peer Neighborhoods are a powerful framework for extending Peer Consistency mechanisms. We show how the choice of partition space determines a locality structure around each possible report, and the update condition which induces incentive-compatibility is required to follow this update structure. Paradoxically, this suggests that the framework of Peer Neighborhoods may be *too* powerful: the fact that the locality structures are mostly unconstrained means that Agents must be able to adapt their updates. Even when the partition space is regular rectangular and one-dimensional, we see from the bin edge conditions for the PE update condition in Equation 4.5 that the update kernel must contain the same ratio of probability mass between the left and right sides of the observation as the ratio between the left and right sides in the prior. In essence, the regular rectangular partition space assumes that the fundamental distance unit is the Lebesgue measure in the underlying space of reports, and that the prior introduces some bias with respect to this measure that must be replicated in the update. There is no a priori reason that the Lebesgue measure should be considered the most fundamental measure to consider in the space of reports. Indeed, the Center can construct the partition space to assert exceedingly complicated measures on the space of reports. There is no fundamental restriction in the theory demanding that the bins in a partition even be connected. In this way, the Center can effectively introduce additional dimensionality to the space by connecting distant regions. It's easy to see that this can result in unreasonably complex update conditions, where Agents must construct update kernels that add mass in regions of the space at arbitrary distances from the observed sample, determined solely by the Center and not be any a priori notion of reasonableness. But an a priori reasonable choice does in fact exist: the shared public prior. The prior automatically suggests a locality structure in terms of the probability measure. We will see following this concept leads to significant conceptual improvements in the mechanism.

Another fault of the Peer Neighborhood framework is the problem of unobserved regions. In

classical Peer Consistency, there is an inherent problem of unobserved categories. Suppose there is a Peer Consistency mechanism operating on a categorical distribution. One of the categories has a very small positive probability, but has never been observed, so is unknown to the Center. There is still a positive probability that an Agent observes and reports this category, but the Center has no baseline with which to evaluate this report. Let us consider the example of the Peer Truth Serum, with payment function $\tau_{\text{PTS}}(r, rr, R) = f(rr) + \frac{c\mathbb{1}_{r=rr}}{R(r)}$. Consider a regime in which the True Distribution Φ is categorical with points $\{x_i\}$, all with non-zero probability, but the Center has never observed a category x_a , so it is given 0 probability in R . In such a regime, the PTS breaks down. If an Agent reports x_a and the Peers are truthful, there is a $\Phi(x_a) > 0$ probability that an Agent observes and reports x_a and successfully matches with a Peer report. The Agent is then paid $\frac{1}{R(x_a)} = \frac{1}{0} = \infty$, so the expected payment from the Center is infinite.

The same essential problem arises in Peer Neighborhoods. Suppose Φ has a continuous random variable, but the Center constructs its prior with bounded update kernels, for example the empirical update, so that there is an interval $[0, b')$ such that $R([0, b']) = 0$ but $\Phi([0, b']) > 0$. Now suppose the Center uses a regular rectangular partition space with bin length $b < b'$. This partition space would violate the bin-supported condition, but this condition is only imposed in the previous chapter to artificially prevent this problem. It is not a theoretical necessity. Since Φ is absolutely continuous with respect to the Lebesgue measure, there exists an interval $(x, x + \epsilon) \subset [0, b')$ with $x > 0$ and $\epsilon < \min(b, b' - x)$ such that $\Phi((x, x + \epsilon)) = p > 0$. The partition selection distribution Ψ is Uniform on $[0, b)$, so $\Psi(\{\theta : \exists i : (x, x + \epsilon) \subset \beta_\theta(i) \text{ and } \beta_\theta(i) \subset [0, b')\}) \geq \frac{\min(b - \epsilon, x, b' - x - \epsilon)}{b} > 0$. So there is a positive probability of an Agent observing a sample inside a bin with prior probability 0, yielding an infinite expected payment.

5.1.2 Approach

As suggested, we consider using the prior to determine the locality structure. The natural approach would be to construct a partition with each bin containing the same probability mass in the prior. Let us consider an absolutely-continuous prior R over the real numbers with strictly positive density, say the Normal distribution, with bins containing equal probability $\frac{1}{n}$ for some $n \geq 2$. Because the distribution has continuous CDF $F_R(x) = \frac{1}{2}(1 + \operatorname{erf}(\frac{x}{\sqrt{2}}))$, $\exists x_0$ such that $F_R(x_1) = \frac{1}{n}$. Because the PDF $f_R(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ is positive everywhere, x_0 is unique. So the bin covering $(-\infty, x_1)$ corresponds to the bin of probability $\frac{1}{n}$ in the left tail of the distribution. Fixing the left end point $-\infty$ and reducing x_0 reduces the probability of the bin. Increasing x_0 would increase the probability of the bin, so the left end point would need to be moved as well. This would leave a gap in the tail $(-\infty, x_0)$ with $x_0 < x_1$ such that $R((-\infty, x_0)) < \frac{1}{n}$. Therefore the only partition of this type with connected bins is $\{(-\infty, x_1), [x_1, x_2), \dots, [x_{n-2}, x_{n-1}), [x_{n-1}, \infty)\}$. We find that it is impossible to construct a partition space with connected bins of probability $\frac{1}{n}$ that is point-isolating.

If the distribution is on a circle, on the other hand, it is possible to construct such a partition

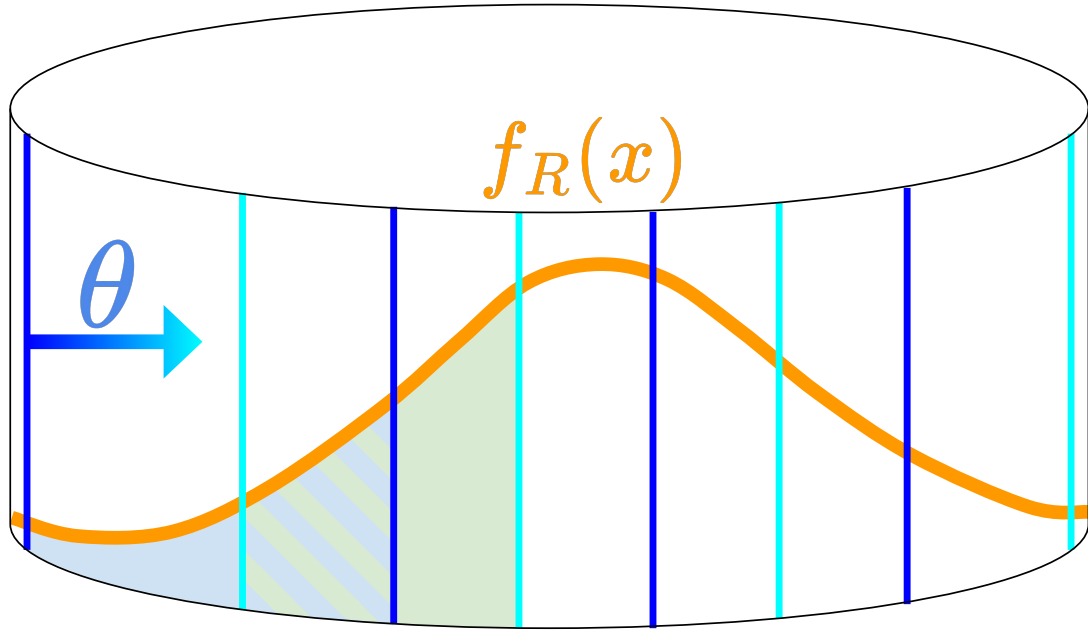


Figure 5.1: Blue and light-blue represent bins with fixed probability measure $\frac{1}{n}$ in R , with the Gaussian density function $f_R(x)$ shown in orange. As the blue bins rotate around the circle according to the parameter θ transforming into the light-blue bins, they deform to maintain the $\frac{1}{n}$ probabilities.

space. Consider the circle to be the real line segment $[0, 1)$ with 1 identified with 0 so arithmetic is taken modulo 1. Let R be the Uniform distribution on $[0, 1)$. If this were trapped on the real line, the only possible bin of probability $\frac{1}{n}$ on the left tail of the distribution would be $[0, \frac{1}{n})$, but now both the left and right end points of the bin can be shifted by a constant: $(\theta \bmod 1, (\theta + \frac{1}{n}) \bmod 1)$. Such bins are unconnected on the real line but connected on the circle. We can then construct a point-isolating partition space:

$$T(\theta) = \{(\theta + \frac{i-1}{n}) \bmod 1, (\theta + \frac{i}{n}) \bmod 1) : i \in [1, n]\}$$

$$\Psi = U(\{0, \frac{1}{n}\})$$

If the distribution is not Uniform, the bin sizes will change as they rotate around the circle. Figure 5.1 shows an example of rotating bins around a circle with a prior probability R with Gaussian density $f_R(x)$.

Let us now compute the payment according to the Peer Neighborhood extension of PTS for a

report r and a Peer report rr :

$$\begin{aligned}\tau_R^\Psi(r, rr) &= \mathbb{E}_{\theta \sim \Psi}[\tau_R^\theta(r, rr)] \\ &= \mathbb{E}_{\theta \sim \Psi}[n \mathbb{1}_{\mathbb{X}_\theta(r) = \mathbb{X}_\theta(rr)}] \\ &= n \Psi(\{\theta : \mathbb{X}_\theta(r) = \mathbb{X}_\theta(rr)\})\end{aligned}$$

If the probability mass between r and rr is greater than $\frac{1}{n}$ on either side, then they will never be in the same bin, so the expected payment is 0. Suppose the probability mass between r and rr is $p = |F_R(r) - F_R(rr)|$ or $1 - |F_R(r) - F_R(rr)| < \frac{1}{n}$, then the probability in Ψ of r and rr being in the same bin is $1 - pn$, so the expected payment is:

$$\max(0, n(1 - n * \min(|F_R(r) - F_R(rr)|, 1 - |F_R(r) - F_R(rr)|)))$$

Let us rewrite this expression in terms of the probability mass contained by the bins $b = \frac{1}{n}$:

$$\max(0, \frac{b - \min(|F_R(r) - F_R(rr)|, 1 - |F_R(r) - F_R(rr)|)}{b^2})$$

We see something curious in this reward function. The choice of b does not appear to have any inherent constraints other than $b \in (0, \frac{1}{2}]$. The strict structure imposed by the requirements on the Peer Neighborhood extension has melted away. We also notice that the distance measurement $\min(|F_R(r) - F_R(rr)|, 1 - |F_R(r) - F_R(rr)|)$ is specific to the case of a distribution on a circle because the bins can wrap around. If we were to try to apply this reward function to a distribution over the real numbers, this reward function would simply be:

$$\max(0, \frac{b - |F_R(r) - F_R(rr)|}{b^2})$$

There does not appear to be an inherent reason why this reward function can't be applied to a distribution over the real numbers. How is this possible? It turns out the Center does not have to partition the entire space prior to receiving reports. When the Center receives a report r and randomly selects a Peer report rr , it only checks for two possibilities: 1) r and rr are in the same bin, or 2) r and rr are not in the same bin. In effect, there are only two relevant bins in this partition, the bin which contains r and the rest of the space.

We say that such a construction chooses a partition space tailored to a specific report r . Consider such a partition space that covers all intervals which contain r and have fixed probability b in R . We simplify the analysis by making a change of variables from r in the real domain to $q = F_R(r)$ in the *quantile domain*. Figure 5.2 shows how the bins in the real domain correspond to fixed-width bins in the quantile domain. Finally, the Center can choose the partition selection distribution as transformations according to F_R in the real domain, which correspond to uniformly random offsets of the bins in the quantile domain. We call the subsequent reward function the *tent function*:

$$T_{b,q}(qq) = \max(0, \frac{b - |q - qq|}{b^2}) \tag{5.1}$$

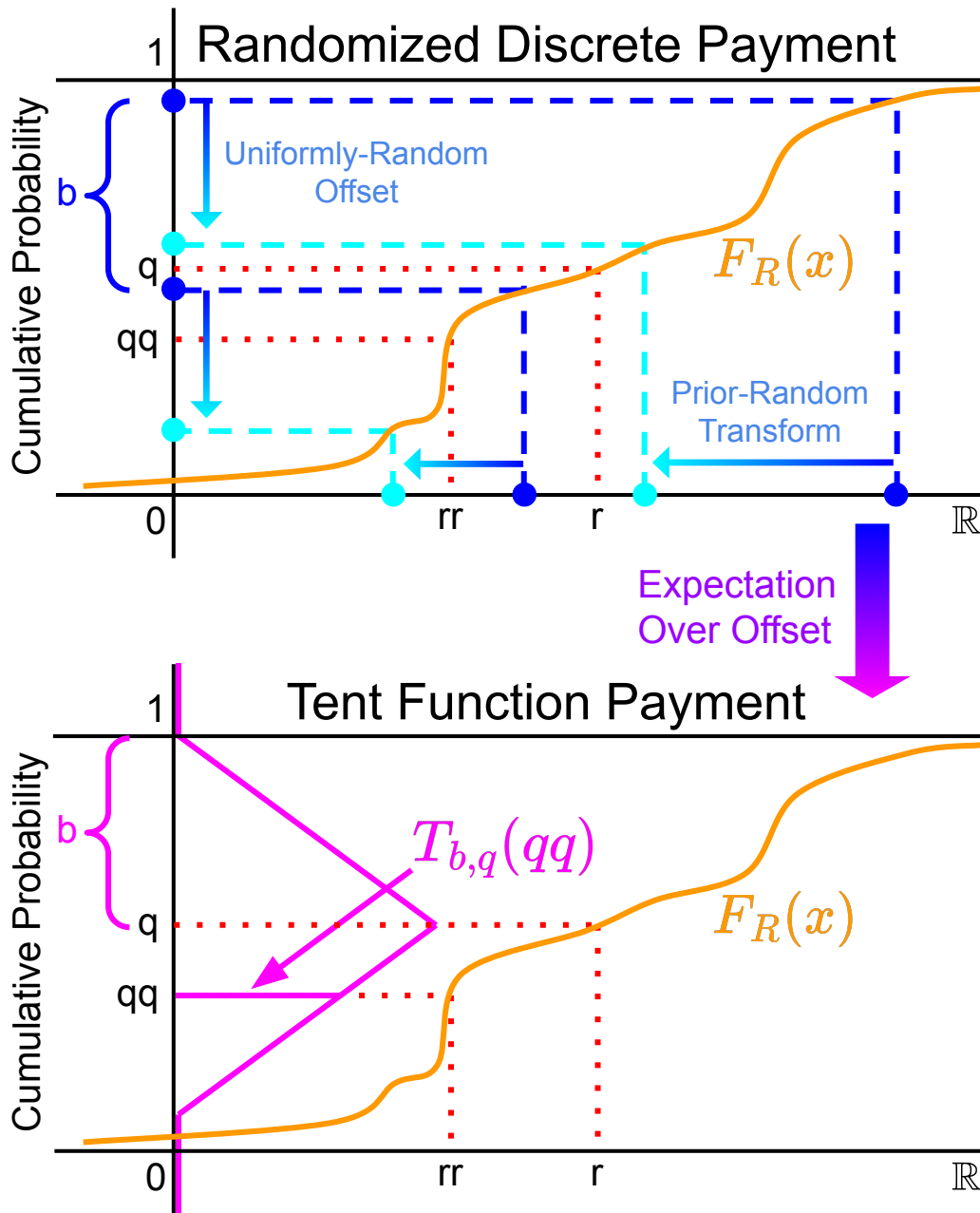


Figure 5.2: Blue and light-blue represent bins with fixed probability measure in R . In the real domain, these categories transform into each other according to F_R , but in the quantile domain they transform with offsets. Taking the expectation over a uniform distribution of these offsets, which is equivalent to taking the expectation over R in the real domain, produces a payment taking the form of this *tent function* as in Equation 5.1.

The tent function determines payments for reports according to the mechanism in this paper, which we call the *Continuous Truth Serum*. We first improve on the definition of the tent function to accommodate delta functions, and we show how this form both replicates the Peer Truth Serum for categorical distributions and addresses the unobserved-category problem. To

show incentive-compatibility we present the expected payment from the perspective of an Agent with a particular posterior belief. We then present an alternative form of this expected payment with a change of variables into the quantile domain. The expected payment takes the form of an integral over a new measure which we call the *ratio measure*, as it is closely related to a Radon-Nikodym derivative. We prove the validity of this measure and the equivalence of the expressions for the expected payment. By analyzing this form of the payment we can more easily establish sufficient update conditions, without being much stricter than necessary, for which the mechanism is provably BNIC. We justify the reasonableness of these update conditions by comparing them to the update conditions for Peer Neighborhoods, and by argument about the necessity of locality constraints for continuous distributions. The conditions admit a broad class of updates, one of which is the same as the pyramid update presented for Peer Neighborhoods, but with slightly different symmetry constraints. We provide simulations using these updates to demonstrate the accuracy and stability of the incentives over deviations from truthfulness and changes in mechanism hyper-parameters.

5.1.3 Model

Consider a setting in which there is some real world phenomenon represented by a *true distribution* which a set of independent, rational, self-interested Agents $\{\mathcal{A}_i\}$ can sample. There is a Center that wishes to learn this distribution with the Agents acting as data providers. There is a single data collection period, at the start of which the Center publicizes its current estimate of the distribution, known as the *public prior* R which is a probability measure on a shared measurable space with the true distribution. Let the measurable space be $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$, and $\mathcal{B}(\mathbb{R})$ are the Borel sets of \mathbb{R} . We will also consider the *quantile measurable space* $([0, 1], \mathcal{B}([0, 1]))$. Agents have individualized distributions $\{\pi_i\}$ on the measurable space $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$, known as *prior beliefs*. Agents choose whether or not to make a single observation o , sampling the true distribution. They then update their prior beliefs to *posterior beliefs* $\{\pi_{i,o}\}$. Each Agent can then report a value from \mathbb{R} of their choice, or make no report, opting out. The Center then pays each Agent for their report according to the incentive mechanism. The Agent believes revealing their observation maximizes their expected payment due to the way the Agent updates their prior to form their posterior. We call this *incentive-compatibility*. For proving incentive-compatibility, we will assume that Agents are *knowledge-less* at the start of the data collection period, so all Agent prior beliefs are set to R .

Notation and Definitions

In order to present the mechanism, we must first present some notation and definitions that will be used throughout the paper.

Let $F_R(x) = R((-\infty, x])$ be the CDF of R . We write $F_R^{-1}(y) = \inf\{x \in \mathbb{R} : y \leq F_R(x)\}$ as the standard quantile function. Let $F(x-) = \lim_{z \uparrow x} F(z)$ and $F_R(x+) = \lim_{z \downarrow x} F(z)$ be the left and right limits of any monotonic function $F(x)$ respectively.

Definition 5.1.1 (Set Inverse Functional CDF). Given a measure R , the *set inverse functional CDF* $\mathbf{F}_R^{-1} : \mathcal{B}([0, 1]) \rightarrow \mathcal{B}(\mathbb{R})$ is given by:

$$\mathbf{F}_R^{-1}(A) = \bigcup_{y \in A} [F_R^{-1}(y-), F_R^{-1}(y+)] \quad (5.2)$$

We briefly prove that \mathbf{F}_R^{-1} is Borel-measurable, in other words it maps Borel sets in $[0, 1]$ to Borel sets in \mathbb{R} :

Lemma 5.1.2. *There are at most countable $y \in [0, 1]$ such that $F^{-1}(y+) > F^{-1}(y-)$.*

Proof. Since $F_R^{-1}(y)$ is monotonic, $y_1 \neq y_2 \Rightarrow (F_R^{-1}(y_1-), F_R^{-1}(y_1+)) \cap (F_R^{-1}(y_2-), F_R^{-1}(y_2+)) = \emptyset$. \square

Proposition 5.1.3. $A \in \mathcal{B}([0, 1]) \Rightarrow \mathbf{F}_R^{-1}(A) \in \mathcal{B}(\mathbb{R})$

Proof. From Lemma 5.1.2, there are at most countable $y \in A$ such that $\mathbf{F}_R^{-1}(y) \neq \{F_R^{-1}(y)\}$. Let us denote Y to be the set of such y . Then $\mathbf{F}_R^{-1}(A) = F_R^{-1}(A) \cup (\bigcup_{y \in Y} [F_R^{-1}(y-), F_R^{-1}(y+)])$. From the monotonicity of F_R^{-1} , $A \in \mathcal{B}([0, 1]) \Rightarrow F_R^{-1}(A) \in \mathcal{B}(\mathbb{R})$. The second term is a countable union of closed intervals, which is clearly in $\mathcal{B}(\mathbb{R})$. \square

Consider uniform distributions $U([a_1, a_2])$ over intervals $[a_1, a_2] \subseteq [0, 1]$, $a_1 \leq a_2$ defined by the CDF:

$$F_{U([a_1, a_2])}(x) = \begin{cases} 0 & x < a_1 \\ \frac{x-a_1}{a_2-a_1} & a_1 \leq x < a_2 \\ 1 & x \geq a_2 \end{cases}$$

Let $\hat{\mathbf{U}}$ be the set of all random variables with such distributions. We define the randomized inverse quantile map $\mathcal{Q}_R^{-1} : \Omega \rightarrow \hat{\mathbf{U}}$:

Definition 5.1.4 (Randomized Inverse Quantile Map). The *randomized inverse quantile map* of x over R is given by $\mathcal{Q}_R^{-1}(x) = y \sim U([F_R(x-), F_R(x+)])$.

If F_R is continuous at x , \mathcal{Q}_R^{-1} maps it to a random variable that is deterministic, i.e. distributed according to a point mass. If F_R has a step at x , corresponding to a point mass in R , \mathcal{Q}_R^{-1} maps x to a uniform random variable over the interval of the step.

The theory in the paper uses distances between values in the quantile space. In the most general theory, distributions can be defined on circles, which can be considered a subset of \mathbb{R} modulo 1 over addition, and the distribution is invariant to translations. On a circle, the distance between 2 points q_1 and q_2 in the quantile space would be $\min(|q_1 - q_2|, 1 - |q_1 - q_2|)$

rather than simply $|q_1 - q_2|$ to capture the cyclic nature of the CDF. We do not present the theory for the case of distributions on circles, but we note when small modifications can be made to produce such a theory.

5.2 A Continuous Truth Serum

5.2.1 The Mechanism

The *Continuous Truth Serum* is a novel incentive mechanism that operates directly on arbitrary real distributions of one variable. The theory is extendable to joint distributions, but we focus on one dimension to demonstrate the principle. The core of the mechanism is the *tent function*:

Definition 5.2.1 (Tent Function). Choose $b > 0$ and consider $q_1 \in [0, 1]$. Then the *tent function* with width b centered at q_1 , $T_{b,q_1} : [0, 1] \rightarrow [0, \frac{1}{b}]$, is

$$T_{b,q_1}(q_2) = \max(0, \frac{b - |q_1 - q_2|}{b^2})$$

where $|q_1 - q_2|$ is specific to a non-cyclic distribution.

If the report r is within b probability of $\pm\infty$, then $q_1 = F_R(r)$ is within b of 0 or 1, so some potential payments will be cut off. An Agent may then be incentivized to move their report away from the boundary. Consider a Peer report rr as a random variable distributed according to R . Then by mapping this random variable according to \mathcal{Q}_R^{-1} , we obtain a random variable qq that is distributed uniformly in $[0, 1]$. If the public prior R is the true distribution, Agents should receive a fixed expected payment for all reports. This is related to a property known as *arbitrage-free* which we elaborate on in Section 5.3.4. In order for the expected payments to be fixed, one must effectively simulate a uniform distribution of qq across the domain of the tent function. This amounts to integrating the tent function outside the boundaries, which can be equivalently achieved by integrating the tent function over $[0, 1]$:

$$1 - \int_0^1 T_{b,q_1}(y) dy$$

If we consider mapping a report r to a probability q and a peer report rr to a probability qq , then the tent function represents a point-wise peer distance score. In order to obtain a single score from the tent function while using \mathcal{Q}_R^{-1} , we take the expectation over the random variables produced by \mathcal{Q}_R^{-1} . There is a choice whether to take the inner expectation, obtaining the mid-point of a step-discontinuity, or to take the outer expectation over the tent function. We present the formulae for all four options $S_{b,r}^{*,*}$, where the *'s indicate inner or

outer expectations for the report and peer report respectively. Let:

$$q_1 = \mathbb{E}_{y_1 \sim \mathcal{Q}_R^{-1}(x_1)}[y_1] = \frac{F_R(x_1-) + F_R(x_1+)}{2}$$

$$q_2 = \mathbb{E}_{y_2 \sim \mathcal{Q}_R^{-1}(x_2)}[y_2] = \frac{F_R(x_2-) + F_R(x_2+)}{2}$$

Then $S_{b,x_1}^{*,*} : \Omega \rightarrow [0, \frac{1}{b}]$ is given by:

$$S_{b,x_1}^{i,i}(x_2) = T_{b,q_1}(q_2) - \int_0^1 T_{b,q_1}(y) dy + 1$$

$$S_{b,x_1}^{i,o}(x_2) = \mathbb{E}_{\mathcal{Q}_R^{-1}(x_2)}[T_{b,q_1}(y_2)] - \int_0^1 T_{b,q_1}(y) dy + 1$$

$$S_{b,x_1}^{o,i}(x_2) = \mathbb{E}_{\mathcal{Q}_R^{-1}(x_1)}[T_{b,y_1}(q_2) - \int_0^1 T_{b,y_1}(y) dy] + 1$$

$$S_{b,x_1}^{o,o}(x_2) = \mathbb{E}_{\mathcal{Q}_R^{-1}(x_1)}[\mathbb{E}_{\mathcal{Q}_R^{-1}(x_2)}[T_{b,y_1}(y_2)] - \int_0^1 T_{b,y_1}(y) dy] + 1$$

We now introduce the Continuous Truth Serum:

Definition 5.2.2 (Continuous Truth Serum). Consider a particularized Agent report r and a set of Peer reports $\{rr\}$ submitted to a Center. Let R be the public prior. The *Continuous Truth Serum* is a category of payment functions for which the Center picks a Peer report rr uniformly at random from $\{rr\}$, then pays the following to the Agent:

$$\tau_b(r, rr, R) = f(rr) + c * S_{b,r}^{*,*}(rr) \tag{5.3}$$

where $f(rr)$ can depend only on rr and $c > 0$.

In most calculations we will set $f(rr) = 0$ and $c = 1$ without loss of generality, and simply refer to the payment function by $S_{b,r}^{*,*}(rr)$.

The mechanism can be understood very simply for a distribution R in which F_R is continuous. The Center simply evaluates the distance between the report r and the peer report rr , but the distance metric is given by R rather than the Euclidean metric on \mathbb{R} . Suppose R is the standard normal distribution with CDF $F_R(x) = \frac{1}{2}(1 + \text{erf}(\frac{x}{\sqrt{2}}))$, so the distance metric is $d(r, rr) = |\text{erf}(\frac{r}{\sqrt{2}}) - \text{erf}(\frac{rr}{\sqrt{2}})|$. But rather than paying inversely proportional to the distance metric, the payment linearly decreases with the distance up to a resolution according to b , but we will see how this resolution is not the same as the resolution for a discretization, because an infinitely small b is not a requirement for strictly truthful incentives.

5.2.2 Replicating the Peer Truth Serum

We examine the regime of the Peer Truth Serum to show that the Continuous Truth Serum is a valid extension. We assume the true distribution is categorical, and the Center has identified

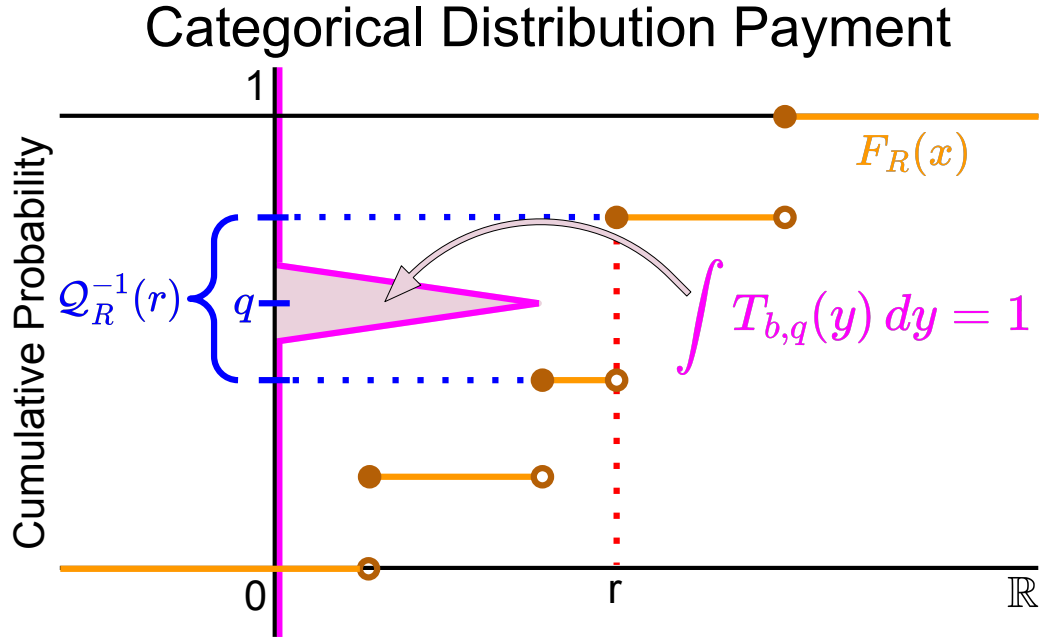


Figure 5.3: For a categorical distribution, mapping the report to q in the middle of the left and right limits allows the tent function to be contained entirely inside the step interval with small enough b . The tent function integrates to 1, so integrating over $\mathcal{Q}_R^{-1}(r)$ lets the mechanism pick up the length of the step interval, reproducing the Peer Truth Serum.

all the categories in the support. We restrict the embedding of the fundamental set to within \mathbb{R} , but we will see that higher dimensional embeddings can be useful. Suppose there are n categories with points $\{x_i\}_{i \in [1, n]}$ in increasing order in the embedding, and $R(x_i) > 0$ for all i . This yields a CDF $F_R(x)$ which is a sum of step functions. We calculate the payments according to the CTS for all options of $S_{b,r}^{*,*}$, first only for reports not on the boundary, meaning $r = x_i$ for $i \in (1, n)$:

$$S_{b,r}^{i,i}(rr) = \begin{cases} \frac{1}{b} & r = rr \\ 0 & \text{otherwise} \end{cases}$$

$$S_{b,r}^{i,o}(rr) = \begin{cases} \frac{1}{R(rr)} & r = rr \\ 0 & \text{otherwise} \end{cases}$$

$$S_{b,r}^{o,i}(rr) = \begin{cases} \frac{1}{R(rr)} & r = rr \\ 0 & \text{otherwise} \end{cases}$$

$$S_{b,r}^{o,o}(rr) = \begin{cases} \frac{1}{R(rr)} - \frac{b}{3R(rr)^2} & r = rr \\ \frac{b}{6R(r)R(rr)} & r = x_i, rr = x_{i \pm 1} \\ 0 & \text{otherwise} \end{cases}$$

We note that these calculations depend on the choice of b . For $S^{i,i}$ and $S^{i,o}$, the tent function is placed in the center of the step. In order for the tent function to isolate each category, it is necessary that $b \leq \frac{\min_i R(x_i)}{2}$. But for $S^{i,i}$, as long as the category is isolated, the information contained in F_R is destroyed; the length of the step is lost. This choice cannot function as a valid extension.

When the outer expectation is taken with respect to the mapping of the report r , the support of the tent function can extend outside the boundaries on the first and last categories in the embedding. We write the equations for $S_{b,r}^{o,i}(rr)$ and $S_{b,r}^{o,o}(rr)$ when $r = x_1$, with the case of $r = x_n$ being symmetric:

$$S_{b,r}^{o,i}(rr) = \begin{cases} \frac{1}{R(r)} + \frac{b}{6R(r)} & r = rr \\ \frac{b}{6R(r)} & \text{otherwise} \end{cases}$$

$$S_{b,r}^{o,o}(rr) = \begin{cases} \frac{1}{R(rr)} - \frac{b}{3R(rr)^2} + \frac{b}{6R(r)} & r = rr \\ \frac{b}{6R(r)R(rr)} + \frac{b}{6R(r)} & r = x_1, rr = x_2 \\ \frac{b}{6R(r)} & \text{otherwise} \end{cases}$$

We see that taking the outer expectation over the report results in error terms from the boundaries and from neighboring categories. While these error terms vanish as b goes to 0, correctly reproducing the PTS, $S_{b,r}^{i,o}(rr)$ reproduces the PTS exactly with only the upper bound $b \leq \frac{\min_i R(x_i)}{2}$. Consider an Agent report $r = x_i$ and Peer report $rr = x_j$. Figure 5.3 shows how $S_{b,r}^{i,o}(rr = r)$ produces a payment by placing the peak of the tent function in the center of the step, and with b small enough, the support of tent function is contained in that interval. Since $\mathcal{Q}_R^{-1}(r)$ is uniform over the length of the step, it is weighted by $\frac{1}{R(r)}$, which exactly reproduces the payment function for the Peer Truth Serum Radanovic et al., 2016.

Addressing Unobserved Categories

We have already seen how both the Peer Truth Serum and the Peer Neighborhood extension generally do not address the problem of unobserved categories or regions, leading to degenerate expected payments. We now show how the Continuous Truth Serum addresses this problem. We examine how the CTS handles this in the case of a categorical distribution, but the principle applies generally to any component of R that is orthogonal to Φ . Let Φ be a categorical distribution embedded in \mathbb{R} with n categories $\{x_i\}$ in increasing order, all with positive probability. Suppose R has positive probability on all categories except x_1 and x_a for some $a \in (2, n)$. The Center chooses $b \leq \frac{\min_{\{x_i: R(x_i) > 0\}} R(x_i)}{2}$. Then $\mathcal{Q}_R^{-1}(x_a)$ is a random variable distributed according to a point mass at $q_a = F_R(x_{a-1}) = F_R(x_{a-1}+) = F_R(x_{a+1}-)$, and $\mathcal{Q}_R^{-1}(x_1)$ is a random variable distributed according to a point mass at 0. So the payment for a match would simply be evaluating the tent function at its peak in both cases, or $\frac{1}{b}$. If Agents report truthfully, matching is probability $\Phi(x_a)$ and $\Phi(x_1)$ respectively, so the contributions towards the expected payment are $\frac{\Phi(x_a)}{b}$ and $\frac{\Phi(x_a)}{b}$ respectively. For the case of x_a , if the

peer is x_{a-1} or x_{a+1} , then $\mathcal{Q}_R^{-1}(x_{a-1}) \sim U((F_R(x_{a-1}-), F_R(x_{a-1}+))) = U((F_R(x_{a-1}-), q_a))$ and $\mathcal{Q}_R^{-1}(x_{a+1}) \sim U((F_R(x_{a+1}-), F_R(x_{a+1}+))) = U((q_a, F_R(x_{a+1}+)))$ respectively. The tent function placed at q_a overlaps with both these intervals in $[q_a - b, q_a]$ and $(q_a, q_a + b]$, yielding payments $\frac{1}{2R(x_{a-1})}$ and $\frac{1}{2R(x_{a+1})}$ respectively. Once again, to contribute to the expected payment we must multiply by the probability of observing such a peer, so the overall expected payment is $\frac{\Phi(x_{a-1})}{2R(x_{a-1})} + \frac{\Phi(x_a)}{b} + \frac{\Phi(x_{a+1})}{2R(x_{a+1})}$. For the case of x_1 , the same argument applies for the contribution from x_2 , but there is no neighbor x_0 . Instead, we obtain the contribution from integrating the tent function outside $[0, 1]$. Since the peak of the tent function is at 0, and the tent function is symmetric, that contribution is $\frac{1}{2}$, so the overall expected payment is $\frac{1}{2} + \frac{\Phi(x_1)}{b} + \frac{\Phi(x_2)}{2R(x_2)}$. We see that the Continuous Truth Serum avoids infinite expected payments when there are unobserved categories by allowing partial matches with neighboring reports. Embedding the categories in higher dimensions allows for arbitrary neighborhood structures.

5.3 Incentive-Compatibility

To show incentive-compatibility, we analyze the expected payments from the perspective of an Agent. An Agent will receive a payment given some Peer report based on the public prior R , but it believes that the Peers will be distributed according to its posterior distribution $P = \pi_o$ for some observation o and a prior $\pi = R$. We write the expected payment for the Agent as:

$$\mathcal{A}_b(r) = \mathbb{E}_{x \sim P}[S_{b,r}^{i,o}(x)] \quad (5.4)$$

5.3.1 The Ratio Measure

We wish to change variables of integration from the real domain to the quantile domain. We start by proving that changing variables between the real domain and the quantile domain keeps the mapped points inside the boundaries set by the left and right limits of the CDF and the inverse CDF. It may not be apparent, but this will be extremely useful for proofs throughout this section.

Proposition 5.3.1. $y \in [F_R(x-), F_R(x+)] \iff x \in [F_R^{-1}(y-), F_R^{-1}(y+)]$

Proof. We first prove the forward implication:

$$\begin{aligned} y &\in [F_R(x-), F_R(x+)] \\ \Rightarrow y &\leq F_R(x+) = F_R(x) \\ \Rightarrow x &\geq F_R^{-1}(y) = F_R^{-1}(y-) \end{aligned}$$

$\forall y' > y, x' < x : F_R(x') \leq F_R(x-) < y' \Rightarrow F_R^{-1}(y') \geq x'$. Since this is true for all $x' < x$, then $F_R^{-1}(y') \geq x$. Since this is true for all $y' > y$, it is true for the limit as $y' \rightarrow y$, so $F_R^{-1}(y+) \geq x$. Therefore $F_R^{-1}(y-) \leq x \leq F_R^{-1}(y+)$.

We now prove the backwards implication similarly:

$$\begin{aligned} x &\in [F_R^{-1}(y-), F_R^{-1}(y+)] \\ \Rightarrow x &\geq F_R^{-1}(y-) = F_R^{-1}(y) \\ \Rightarrow y &\leq F_R(x) = F_R(x+) \end{aligned}$$

$$\forall y' > y, x' < x : F_R^{-1}(y') > x' \Rightarrow F_R(x') \leq y'.$$

Since this is true for all $y' > y$, $F_R(x') \leq y$.

Since this is true for all $x' < x$, it is true for the limit as $x' \rightarrow x$, so $F_R(x-) \leq y$.

Therefore $F_R(x-) \leq y \leq F_R(x+)$. □

We now present the *ratio measure*, which represents the measure which the tent function is integrated against to produce the expected payment according to an Agent's posterior. Let $J_R = \{x \in \mathbb{R} : R(\{x\}) > 0\}$ be the set of point masses in R , or step discontinuities in F_R . Let $I_R(x) = \{y \in [0, 1] : x \in \mathbf{F}_R^{-1}(\{y\})\}$ such that it is $\{F_R(x)\}$ everywhere except at step discontinuities, where it is the set of points in the half-open interval $(F_R(x-), F_R(x+)]$. Then we define the *ratio measure* $\mu_{\frac{P}{R}} : \mathcal{B}([0, 1]) \rightarrow [0, 1]$ as follows:

Definition 5.3.2 (Ratio Measure). Consider two probability measures P and R , then the probability measure $\mu_{\frac{P}{R}}$, which we call the *ratio measure* with respect to P and R , is given by:

$$\mu_{\frac{P}{R}}(A) = P(\mathbf{F}_R^{-1}(A) \setminus J_R) + \sum_{x \in J_R} \frac{P(\{x\})}{R(\{x\})} \mathcal{L}(A \cap I_R(x))$$

where \mathcal{L} is the Lebesgue measure.

This is well defined because $A \in \mathcal{B}([0, 1]) \Rightarrow \mathbf{F}_R^{-1}(A) \in \mathcal{B}(\mathbb{R})$, and J_R is at most a countable set, so $\mathbf{F}_R^{-1}(A) \setminus J_R \in \mathcal{B}(\mathbb{R})$. We prove that this measure is a probability measure over the quantile measurable space.

Lemma 5.3.3. $\mu_{\frac{P}{R}}$ is a probability measure over the measurable space $([0, 1], \mathcal{B}([0, 1]))$.

Proof. $\mu_{\frac{P}{R}}$ is clearly non-negative, since R , P , and \mathcal{L} are all non-negative measures. To show that $\mu_{\frac{P}{R}}$ satisfies countable additivity, since P is a measure, it is sufficient to show that $A \cap A' = \emptyset \Rightarrow \mathbf{F}_R^{-1}(A) \setminus J_R \cap \mathbf{F}_R^{-1}(A') \setminus J_R = \emptyset$.

Let $x \in \mathbf{F}_R^{-1}(A) \setminus J_R$. Then from Proposition 5.3.1, $\exists y \in A : x \in [F_R^{-1}(y-), F_R^{-1}(y+)]$ and $y \in [F_R(x-), F_R(x+)]$. Since $x \notin J_R$, $y = F_R(x)$, so $F_R(x) \in A$. Finally, $\mu_{\frac{P}{R}}([0, 1]) = P((-\infty, \infty)) = 1$. □

While the ratio measure is similar to a Radon–Nikodym derivative, step discontinuities in F_R need to be handled separately because the tent function may cover only a portion of a step discontinuity, and the relative weight of that portion must be included.

We can get a cleaner representation of $\mathcal{A}_b(r)$ by considering an extension of $\mu_{\frac{P}{R}}$. After reporting r and computing $q = \mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(r)}[y]$, the Agent takes an expectation over peers drawn according

to P mapped into the quantile domain with \mathcal{Q}_R^{-1} , then computes the average score under the tent function in the interval $[0, 1]$. Outside the boundaries, the Agent takes the expectation over an extended distribution. If the distribution of P is on \mathbb{R} , the extension would be Lebesgue. If the distribution of P is on a circle, the extension would be a periodic repetition of the inverse CDF of P . We only show the analysis for the case of the distribution of P on a \mathbb{R} , but the same analysis applies to the case of the circle with minor adjustments:

Definition 5.3.4 (Extended Ratio Measure). For all $A \in \mathcal{B}(\mathbb{R})$, the *extended ratio measure* $\hat{\mu}_R^P : \mathcal{B}(\mathbb{R}) \rightarrow [0, 1]$ is given by $\hat{\mu}_R^P(A) = \mu_R^P(A \cap [0, 1]) + \mathcal{L}(A \setminus [0, 1])$ where \mathcal{L} is the Lebesgue measure.

Note that the extended ratio measure is not a probability measure, but it is still sigma-finite and non-negative.

By using the extended ratio measure, we no longer need to include the extra terms in $S_{b,r}^{i,o}(x)$, so our final simplified expression for the expected payment becomes:

$$\mathcal{A}_b^*(q) = \int_{\mathbb{R}} T_{b,q}(y) d\hat{\mu}_R^P(y) \quad (5.5)$$

This is a finite integral because $T_{b,q}(y)$ is bounded and supported on finite interval, and $\hat{\mu}_R^P$ of this interval is finite.

Theorem 5.3.5. $\mathcal{A}_b^*(q) = \mathcal{A}_b(r)$ where $q = \mathbb{E}_{y^* \sim \mathcal{Q}_R^{-1}(r)}[y^*]$.

Proof. It is clear that these are identical with respect to the extra terms, since they are just integrating the tent function outside the boundaries. Then we are only concerned with the interval $[0, 1]$ and the probability measure μ_R^P .

Consider some function $f : [0, 1] \rightarrow \mathbb{R}$. Define $\tilde{f} : \mathbb{R} \rightarrow \mathbb{R}$ such that $\tilde{f}(x) = \mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(x)}[f(y)]$. We will show the conditions by which $\mathbb{E}_{x \sim P}[\tilde{f}(x)] = \mathbb{E}_{y \sim \mu_R^P}[f(y)]$. This is true by the construction of μ_R^P if $f = \mathbb{1}_{\{y < d\}}$ or $f = \mathbb{1}_{\{y \leq d\}}$ for some $d \in [0, 1]$. Then, by linearity of expectation, this is true for weighted sums of interval functions $f = \sum_i a_i \mathbb{1}_{\alpha_i}$ where $\alpha_i = (a, b)$ with $a, b \in [0, 1]$, $b > a$, and the interval can be closed on either side or both.

Consider two functions f_1 and f_2 . Then $\tilde{f}_1(x) - \tilde{f}_2(x) = \mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(x)}[f_1(y) - f_2(y)]$. If $f_1 - f_2$ is uniformly bounded, $\tilde{f}_1 - \tilde{f}_2$ has the same uniform bound: $\forall y \in [0, 1] |f_1(y) - f_2(y)| < \epsilon \Rightarrow |\tilde{f}_1(y) - \tilde{f}_2(y)| < \epsilon$.

Given a continuous f_1 , it can be approximated with an arbitrary uniform bound by f_2 as a sum of interval functions. By shrinking the lengths of the intervals and applying the appropriate weights, one can achieve an arbitrarily low uniform error bound, so f_2 converges to f_1 point-wise. Therefore, the corresponding functions \tilde{f}_2 converges to \tilde{f}_1 point-wise. Both expectations

are continuous in this limit, and if we let f_1 be the tent function $T_{b,q}(y)$, which is continuous, then:

$$\begin{aligned} & \mathbb{E}_{x \sim P} [\mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(x)} [T_{b,s}(y)]] \\ &= \mathbb{E}_{y \sim \mu_R^P} [T_{b,q}(y)] \\ &= \int_0^1 T_{b,q}(y) d\mu_R^P(y) \end{aligned}$$

□

5.3.2 Report Optimization

The Agent wishes to know for which q the expression $A_b^*(q)$ is maximized in order to choose a report which will map to q and achieve the highest expected payment. We start by searching for critical points of A_b^* . Only the left and right derivatives exist in general. Let $\mathcal{I}_q(y) = \begin{cases} -1 & y < q \\ 1 & y \geq q \end{cases}$. Then $T_{b,q}(y) = \max(0, \frac{b-(y-q)\mathcal{I}_q(y)}{b^2})$. We evaluate the left and right derivatives of $T_{b,q}(y)$ with respect to q and obtain the following:

$$\begin{aligned} \frac{d}{dq^-} T_{b,q}(y) &= \begin{cases} -\frac{1}{b^2} & q-b \leq y < q \\ \frac{1}{b^2} & q \leq y < q+b \\ 0 & \text{otherwise} \end{cases} \\ \frac{d}{dq^+} T_{b,q}(y) &= \begin{cases} -\frac{1}{b^2} & q-b < y \leq q \\ \frac{1}{b^2} & q < y \leq q+b \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

The left and right derivatives of $A_b^*(q)$ are given by:

$$\begin{aligned} \frac{d}{dq^-} A_b^*(q) &= \hat{\mu}_R^P((q, q+b)) - \hat{\mu}_R^P((q-b, q)) + \hat{\mu}_R^P(\{q\}) - \hat{\mu}_R^P(\{q-b\}) \\ \frac{d}{dq^+} A_b^*(q) &= \hat{\mu}_R^P((q, q+b)) - \hat{\mu}_R^P((q-b, q)) + \hat{\mu}_R^P(\{q+b\}) - \hat{\mu}_R^P(\{q\}) \end{aligned}$$

In order for q to be at a local maximum, it is necessary that $\frac{d}{dq^-} A_b^*(q) \geq 0$ and $\frac{d}{dq^+} A_b^*(q) \leq 0$. From this we obtain the necessary condition:

$$\hat{\mu}_R^P(\{q-b\}) - \hat{\mu}_R^P(\{q\}) \leq \hat{\mu}_R^P((q, q+b)) - \hat{\mu}_R^P((q-b, q)) \leq \hat{\mu}_R^P(\{q\}) - \hat{\mu}_R^P(\{q+b\})$$

For this to be satisfied, it is further necessary that:

$$\hat{\mu}_{\frac{P}{R}}(\{q-b\}) + \hat{\mu}_{\frac{P}{R}}(\{q+b\}) \leq 2\hat{\mu}_{\frac{P}{R}}(\{q\})$$

We will trivially satisfy this condition by requiring that an update only add mass in the region $(q-b, q+b)$ in the quantile domain, so $\hat{\mu}_{\frac{P}{R}}(\{q-b\}) = \hat{\mu}_{\frac{P}{R}}(\{q+b\}) = 0$. We then obtain the simplified necessary condition:

$$|\hat{\mu}_{\frac{P}{R}}((q, q+b)) - \hat{\mu}_{\frac{P}{R}}((q-b, q))| \leq \hat{\mu}_{\frac{P}{R}}(\{q\})$$

We call this the *balancing condition*:

Definition 5.3.6 (b-Probability Balanced Update). An update from a prior R to a posterior P is *b-probability balanced* around q if $|\hat{\mu}_{\frac{P}{R}}((q, q+b)) - \hat{\mu}_{\frac{P}{R}}((q-b, q))| \leq \hat{\mu}_{\frac{P}{R}}(\{q\})$. We say that it is *strictly b-probability balanced* if $|\hat{\mu}_{\frac{P}{R}}((q, q+b)) - \hat{\mu}_{\frac{P}{R}}((q-b, q))| < \hat{\mu}_{\frac{P}{R}}(\{q\})$ for $\hat{\mu}_{\frac{P}{R}}(\{q\}) > 0$.

In the case where $|\hat{\mu}_{\frac{P}{R}}((q, q+b)) - \hat{\mu}_{\frac{P}{R}}((q-b, q))| = \hat{\mu}_{\frac{P}{R}}(\{q\})$, it is undetermined if q is a local maximum, so we have to check that the second handed derivatives are ≤ 0 . For these we choose to require strict inequalities. Without this requirement, one can continue this analysis indefinitely and still fail to acquire necessary and sufficient conditions, since there is no guarantee that $A_b^*(q)$ is analytic. We compute the handed second derivatives as the left derivative of the left derivative and the right derivative of the right derivative:

$$\begin{aligned} & \frac{d}{dq^-} (F_{\mu_{\frac{P}{R}}}((q-b)^-) + F_{\mu_{\frac{P}{R}}}((q+b)^-) - 2F_{\mu_{\frac{P}{R}}}(q-)) \\ & \frac{d}{dq^+} (F_{\mu_{\frac{P}{R}}}((q-b)^+) + F_{\mu_{\frac{P}{R}}}((q+b)^+) - 2F_{\mu_{\frac{P}{R}}}(q+)) \end{aligned}$$

Proposition 5.3.7. *If an update is strictly b-probability balanced update around q , and the right and left second derivatives at q exist and are negative, q is the location of a local maximum in A_b^* .*

We omit a proof for this proposition, as it is trivial from its construction.

We will add an additional condition to simplify the requirements for global maximization:

Definition 5.3.8 (b-Probability Bounded Update). An update from a prior R to a posterior P is *b-probability bounded* around q if the following is true. Let $P = \alpha R + (1 - \alpha)K$ for some $\alpha \in (0, 1)$ and some probability measure K , which we call the *update kernel*, then $\exists q'$ where $q \in (q' - \frac{b}{2}, q' + \frac{b}{2})$, such that $\forall x$ in the support of K , $\mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(x)}[y] \in (q' - \frac{b}{2}, q' + \frac{b}{2})$.

Proposition 5.3.9. *If an update is b-probability bounded, then for any point x such that $\mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(x)}[y] \in (q' - \frac{b}{2}, q' + \frac{b}{2})$, if x is a local maximum then x is a global maximum, and the set of all such x forms an interval.*

In order to prove this, we first prove the following Lemma:

Lemma 5.3.10. $A_b^*(q)$ is absolutely continuous.

Proof.

$$\begin{aligned} |A_b^*(q_1) - A_b^*(q_2)| &= \int_{\mathbb{R}} |T_{b,q_1}(y) - T_{b,q_2}(y)| d\hat{\mu}_{\frac{P}{R}}(y) \\ &\leq \int_{q_1-b}^{q_2+b} \frac{1}{b^2} |q_1 - q_2| d\hat{\mu}_{\frac{P}{R}}(y) \\ &\leq \frac{2}{b^2} |q_1 - q_2| \end{aligned}$$

for $q_1 < q_2$ w.l.g. □

We now prove Proposition 5.3.9:

Proof. Outside $(q' - \frac{b}{2}, q' + \frac{b}{2})$, $\mu_{\frac{P}{Q}} = \alpha * \mathcal{L}$, and $\mu_{\frac{P}{Q}}((q' - \frac{b}{2}, q' + \frac{b}{2})) > \alpha * b$. So $\frac{d}{dq+} A_b^*(q) \geq 0$ for $q \leq q' - \frac{b}{2}$ and $\frac{d}{dq+} A_b^*(q) \leq 0$ for $q \geq q' + \frac{b}{2}$. $A_b^*(q)$ is monotonically non-decreasing inside $(q' - \frac{b}{2}, q' + \frac{b}{2})$. From Lemma 5.3.10, $A_b^*(q) = \int_{-\infty}^q \frac{d}{dy+} A_b^*(y) dy$. So there exists some interval inside $(q' - \frac{b}{2}, q' + \frac{b}{2})$ where $A_b^*(q)$ is both locally and globally maximized. From monotonicity of F_R , this corresponds to some interval in the real domain. □

We present the following corollary:

Corollary 5.3.11. At all continuity points of F_R and with $b < \min_{x:R(\{x\})>0} \frac{R(\{x\})}{2}$, $\frac{d}{dq} F_{\mu_{\frac{P}{R}}}(q-b) = \frac{d}{dq} F_{\mu_{\frac{P}{R}}}(q+b) = 0$.

Then the only requirement on the second derivative for continuity points is that $\frac{d}{dq-} F_{\mu_{\frac{P}{R}}}(q-) > 0$ and $\frac{d}{dq+} F_{\mu_{\frac{P}{R}}}(q+) > 0$. If strict b-probability balance is observed, this case is only relevant when $\hat{\mu}_{\frac{P}{R}}(\{q\}) = 0$, so as long as $\mu_{\frac{P}{Q}}$ is "well-behaved" at that point, then a probability density $f_{\mu_{\frac{P}{R}}}(q)$ exists. We write this as a condition:

Definition 5.3.12 (Concentrated Update). An update from a prior R to a posterior P is *concentrated* around r if the following is true. Let $q = \mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(r)}[y]$. Then either $f_{\mu_{\frac{P}{R}}}(q)$ exists and is positive, or $\mu_{\frac{P}{R}}(\{q\}) > 0$.

We will see that these three conditions ensure truthfulness.

Optimizing in the Real Domain

We've identified how to optimize the payment in the quantile domain, but the Agent is not necessarily able to pick any point q in the quantile domain, it can only pick a report r in the

real domain, which then gets mapped to a point in the quantile domain. At any point mass in R , the report gets mapped to the mid-point of the step in F_R . We wish to identify when such a mid-point is a local maximum in the real domain. In this case, it is sufficient to show that the midpoint achieves a higher expected payment than the endpoints.

Lemma 5.3.13. *Let $q = \mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(r)}[y]$, $q_+ = F_R(r+)$, and $q_- = F_R(r-)$. If $b \leq \frac{q_+ - q_-}{2}$ and the update from R to P is b -probability bounded around q , then $A_b^*(q) > A_b^*(q_+)$ and $A_b^*(q) > A_b^*(q_-)$.*

Proof. In the interval $[q_-, q_+]$, $\hat{\mu}_R^P(A) = d * \mathcal{L}(A)$ where d is some positive constant. Then the expected payment at those endpoints is:

$$\begin{aligned} \mathcal{A}_b^*(q_-) &= \int_{q_- - b}^{q_-} T_{b,q}(y) d\hat{\mu}_R^P(y) + \frac{d}{2} \\ \mathcal{A}_b^*(q_+) &= \int_{q_+}^{q_+ + b} T_{b,q}(y) d\hat{\mu}_R^P(y) + \frac{d}{2} \end{aligned}$$

and the expected payment at the midpoint is simply d . The midpoint achieves a higher expected payment if, when updating from R to P , additional probability mass is only placed in a region which maps inside (q_-, q_+) . This is trivially satisfied by the b -probability bounded condition if $b \leq \frac{q_+ - q_-}{2}$. \square

5.3.3 Sufficient Maximizing Conditions

We have constructed a set of sufficient conditions that guarantee that an observation point r is the unique global maximizer of $A_b(r)$.

Theorem 5.3.14. *Suppose R contains finite point masses and $b < \min_{x: R(\{x\}) > 0} \frac{R(\{x\})}{2}$. Then r is at a unique global maximum of A_b if the update from R to P is concentrated around r , and for $q = \mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(r)}[y]$, the update is b -probability balanced and bounded around q .*

Proof. First suppose r is at a point mass in R . Then with the restriction on b , the b -probability bounded condition ensures that the update kernel K is a point mass at r . The update simply adds weight to the interval $(q - b, q + b)$ in the quantile domain, so this is naturally consistent with the b -probability balanced condition. There may be an interval inside $(q - b, q + b)$ where A_b^* is locally maximized, but Lemma 5.3.13 ensures that r is a unique local maximum in A_b . Proposition 5.3.9 then ensures that the locally maximizing q s are global maximizers of A_b^* , so r is a unique global maximizer of A_b .

Now suppose r is at a point of continuity in F_R . Proposition 5.3.7 guarantees that q is a local maximum. Then Proposition 5.3.9 guarantees that q is a global maximum, but there may be an interval around q that is equally globally maximizing. But this implies that the K measure of this interval, excluding end points, is 0. Since this interval contains q , this violates the concentration condition unless q is the only value in the interval. Because the update is

concentrated and r is a point of continuity in F_R , Corollary 5.3.11 implies that q is a unique global maximizer of A_b^* . Therefore, r is a unique global maximizer of A_b . \square

In practical terms, the b-probability bounded condition enforces a notion of locality in the update, but the locality is determined according to the R measure rather than the Euclidean metric on the report space. It is difficult to envision an incentive mechanism for arbitrary distributions that does not introduce a similar locality requirement on updates. The b-probability balanced condition requires that the update be unbiased in the sense that the update kernel should place equal mass on either side of the observed point. Finally, the concentration condition requires that the kernel not place 0 probability mass around the observed point. It would be unreasonable for an Agent to not follow this condition, as it would suggest that the neighborhood of the observed point is relevant, but not the observed point itself. The class of update kernels permitted by these conditions is extremely broad. Any update kernel with support in the bound that places mass around the observation and balances the mass on either side is permitted. These are also merely sufficient conditions, so the overall class of incentive-compatible updates is larger.

Admitted Updates

We cannot present an exhaustive list of admitted updates, but we can compare the conditions to those found in the previous chapter. Let us consider the case of one dimension. The first observation is that for additive update kernels, the conditions from both the previous chapter and this one require boundedness of the update kernel around the observed point, with the bounds related to the sizes of the bins which contain the point. We also find that both have a sort of balancing condition for the amount of probability the kernel can have on either side of the point. In the previous chapter we found that, for a regular rectangular partition space, the Partition-Expected update conditions can be described by conditions on the edges of the bins which surround the observed point as shown in Equation 4.5. These bin edge conditions in one dimension simply state that the ratio of the left kernel probability to the right kernel probability must be the same as the ratio of those regions in the prior. In other words, the fixed bin shape forces the Agent to maintain any probabilistic bias it has in the prior. If the left and right bins had equal probability instead of equal width, this ratio would be 1, and the symmetry condition would be that the kernel must place equal probability on either side of the observation point. This is identical to the symmetry condition we just derived.

The Continuous Truth Serum mechanism admits a similar class of updates as the Peer Neighborhood extension of the Peer Truth Serum. We discovered that it is always possible to construct a valid update using pyramid kernels, or triangular in one dimension. The base of the triangle is arranged so it satisfies the symmetry condition given by Equation 4.5. Now, the triangle must simply be Isosceles with the peak at the observed point:

Proposition 5.3.15. *Given a prior R with at most finite point masses, tent function width $b < \min_{x:R(\{x\})>0} \frac{R(\{x\})}{2}$, and observation o , consider $\delta_l : F_R(\delta_l) > F_R(o) - \frac{b}{2}$ and $\delta_r : F_R(\delta_r) <$*

$F_R(o) + \frac{b}{2}$. Let $\delta = \min(\delta_l, \delta_r) > 0$. An update to posterior $P = \alpha R + (1 - \alpha)K_o$, such that kernel K_o has probability density function supported in $[o - \delta, o + \delta]$: $f_{K_o}(x) = \min(\frac{x-o+\delta}{\delta^2}, \frac{o-x+\delta}{\delta^2})$, satisfies the conditions in Equation 5.3.14 so that o is a global maximizer of A_b .

Proof. By the definition of δ , the kernel K_o is b-probability bounded. Since K_o has positive probability density at o , either $f_{\mu_{\frac{P}{R}}}(F_R(o))$ exists and is positive because R has positive density at o , or $\mu_{\frac{P}{R}}(\{q\}) > 0$ because R has density 0 or the density does not exist at o . Therefore, P is concentrated around o . Finally, the density of K_o is symmetric about o , so the update is b-probability balanced. \square

5.3.4 Additional Properties

We briefly address other important properties of the mechanism for practical viability.

Arbitrage-Free

Definition 5.3.16 (Arbitrage-Free). A mechanism is *Arbitrage-Free* if an Agent with posterior equal to R believes that, if the peer reports are truthful, they will receive the same expected reward for any report.

We denote the strategy outlined in this definition as the *arbitrage strategy*, which involves the Agent failing to make an observation. We have occasionally touched on this concept when discussing how to handle the tent function extending outside the boundaries of $[0, 1]$. We see that with our construction, the Continuous Truth Serum is Arbitrage-Free. To prove this we present a number of statements which build towards the final proof:

Lemma 5.3.17. *The CDF of the inverse map of a probability y is lower bounded by y : $\forall y \in [0, 1]$, $F_R(F_R^{-1}(y)) \geq y$*

Proof.

$$\begin{aligned} x &> F_R^{-1}(y) \\ \Rightarrow F_R(x) &\geq y \\ \Rightarrow F_R(F_R^{-1}(y)+) &\geq y \\ \Rightarrow F_R(F_R^{-1}(y)) &\geq y \end{aligned}$$

\square

Lemma 5.3.18. *If the CDF of the inverse map of a probability y is strictly greater than y , then y must be within the range of a jump discontinuity of the CDF: $F_R(F_R^{-1}(y)) > y \Rightarrow F_R^{-1}(y) \in J_R$*

Proof. Let $x = F_R^{-1}(y)$. From Proposition 5.3.1 we have $y \in [F_R(x-), F_R(x+)]$. We also have $y < F_R(x) = F_R(x+)$, so $F_R(x-) < F_R(x+)$. Therefore $x \in J_R$. \square

Definition 5.3.19 (CDF Range). Define $L_R = \{0, 1\} \cup F_R((-\infty, \infty))$ to be the range of F_R .

Lemma 5.3.20. *If the inverse map of a probability y is in the set inverse functional of a set A and is not at a jump discontinuity, then y must be in A and in L_R : $F_R^{-1}(y) \in \mathbf{F}_R^{-1}(A) \setminus J_R \Rightarrow y \in A \cap L_R$*

Proof. Choose any y' such that $x = F_R^{-1}(y') \in \mathbf{F}_R^{-1}(A) \setminus J_R$. From the definition of \mathbf{F}_R^{-1} and Proposition 5.3.1, $\exists y \in A$ such that $x \in [F_R^{-1}(y-), F_R^{-1}(y+)]$ and $y \in [F_R(x-), F_R(x+)]$.

Since $x \notin J_R$, $y = F_R(x)$, and from Lemma 5.3.17 and Lemma 5.3.18, we have $F_R(F_R^{-1}(y')) = y'$. Therefore $y = y'$ and $y' \in A$.

Since $y' = F(x)$, $y' \in L_R$, so $y = y' \in A \cap L_R$. \square

Definition 5.3.21 (CDF Step Intervals). For $x \in J_R$, define $I_R^o(x) = (F_R(x-), F_R(x+))$ and $I_R^c(x) = [F_R(x-), F_R(x+)]$.

Lemma 5.3.22. *If a probability value y is in a set A with the closed CDF step intervals removed, then the inverse map of y is in the set inverse functional of A with the jump discontinuities removed: $y \in A \setminus \bigcup_{x \in J_R} I_R^c(x) \Rightarrow F_R^{-1}(y) \in \mathbf{F}_R^{-1}(A) \setminus J_R$*

Proof. Let $x = F_R^{-1}(y)$, from Proposition 5.3.1, $y \in [F_R(x-), F_R(x+)]$, so $x \notin J_R$. Also $y \in A \Rightarrow x \in \mathbf{F}_R^{-1}(A)$, so $x = F_R^{-1}(y) \in \mathbf{F}_R^{-1}(A) \setminus J_R$. \square

Lemma 5.3.23. *The range of F_R does not contain the open CDF step intervals: $L_R \subset [0, 1] \setminus \bigcup_{x \in J_R} I_R^o(x)$*

Proof. $y \in L_R \Rightarrow \exists x \in \mathbb{R} : y = F_R(x) = F_R(x+)$.

$x' > x \Rightarrow F_R(x) \leq F_R(x'-)$, so if $x' \in J_R$, $F_R(x) \notin I_R^o(x)$.

$x' < x \Rightarrow F_R(x'+) \leq F_R(x-) \leq F_R(x)$, so if $x' \in J_R$, $F_R(x) \notin I_R^o(x)$.

Therefore, $y = F_R(x) \notin \bigcup_{x \in J_R} I_R^o(x)$. \square

Proposition 5.3.24. *The R measure of the set inverse functional of a set A with jump discontinuities removed, is the Lebesgue measure of the set A with the closed CDF step intervals removed:*

$$R(\mathbf{F}_R^{-1}(A) \setminus J_R) = \mathcal{L}(A \setminus \bigcup_{x \in J_R} I_R^o(x)) = \mathcal{L}(A \setminus \bigcup_{x \in J_R} I_R^c(x))$$

Proof. $R(\mathbf{F}_R^{-1}(A) \setminus J_R) = \mathcal{L}(\{y \in [0, 1] : F_R^{-1}(y) \in \mathbf{F}_R^{-1}(A) \setminus J_R\})$.

From Lemma 5.3.20, $\mathcal{L}(\{y \in [0, 1] : F_R^{-1}(y) \in \mathbf{F}_R^{-1}(A) \setminus J_R\}) \leq \mathcal{L}(A \cap L_R)$.

Then from Lemma 5.3.23, $\mathcal{L}(A \cap L_R) \leq \mathcal{L}(A \setminus \bigcup_{x \in J_R} I_R^o(x))$.

From Lemma 5.3.22, $\mathcal{L}(\{y \in [0, 1] : F_R^{-1}(y) \in \mathbf{F}_R^{-1}(A) \setminus J_R\}) \geq \mathcal{L}(A \setminus \bigcup_{x \in J_R} I_R^c(x))$.

$\bigcup_{x \in J_R} I_R^c(x) \setminus \bigcup_{x \in J_R} I_R^o(x)$ is at most a countable number of points in $[0, 1]$, so $\mathcal{L}(A \setminus \bigcup_{x \in J_R} I_R^c(x)) = \mathcal{L}(A \setminus \bigcup_{x \in J_R} I_R^o(x)) = R(\mathbf{F}_R^{-1}(A) \setminus J_R)$ \square

We can now prove that $\mu_{\mathbb{R}}^R$ is the Lebesgue measure.

Proposition 5.3.25. *The ratio measure of R and R , $\mu_{\frac{R}{R}}$, is the Lebesgue measure over $[0, 1]$. $\forall A \in \mathcal{B}([0, 1]) : \mu_{\frac{R}{R}}(A) = \mathcal{L}(A)$.*

Proof. $\mu_{\frac{R}{R}}(A) = R(\mathbf{F}_R^{-1}(A) \setminus J_R) + \sum_{x \in J_R} \frac{R(\{x\})}{R(\{x\})} \mathcal{L}(A \cap I_R(x)) = R(\mathbf{F}_R^{-1}(A) \setminus J_R) + \mathcal{L}(\bigcup_{x \in J_R} A \cap I_R^c(x))$.
From Proposition 5.3.24, $\mu_{\frac{R}{R}}(A) = \mathcal{L}(A \setminus \bigcup_{x \in J_R} I_R^c(x)) + \mathcal{L}(\bigcup_{x \in J_R} A \cap I_R^c(x)) = \mathcal{L}(A)$. \square

The extension $\hat{\mu}_{\frac{R}{R}}$ simply places the Lebesgue measure outside $[0, 1]$, so $\hat{\mu}_{\frac{R}{R}} = \mathcal{L}$ over \mathbb{R} . The expected payment is then computed by simply integrating the tent function over \mathbb{R} , which is always 1. Knowing that the mechanism is Arbitrage-Free, the Center can eliminate this "lazy strategy" from the set of viable strategies by subtracting 1 from the payment.

Overcoming Cost of Effort

The mechanism being Arbitrage-Free allows it to address *cost of effort*. In many circumstances, an Agent might experience some negative utility for playing a truthful strategy. Suppose that for some Agent A_i , the cost of effort for that Agent is e_i . Suppose the payment function takes the form $c * (S_{b,r}^{i,o}(rr) - 1)$ so the arbitrage strategy has 0 expected payment. An Agent playing a truthful strategy and obeying the update conditions will clearly compute $\mathbb{E}_{rr \sim P}[S_{b,r}^{i,o}(rr)] > 1$, since there is a net increase in probability under the tent. In other words, the Agent expects to be paid $\delta_i(o) > 0$ for observing and truthfully reporting a sample o . A priori, the Agent can calculate the expected utility for observing a sample and truthfully reporting it as $\Delta_i = \mathbb{E}_{o \sim R}[\delta_i(o)] > 0$. We see that the Agent will have a positive incentive to play this truthful strategy as long as $\Delta_i > e_i$. The Center can scale the payments with c to overcome some maximum cost of effort, but this is an implementation decision for the Center based on numerous factors, such as budget and the distribution of costs of efforts for the Agents.

5.4 Simulations

We conduct simulations to demonstrate the accuracy and stability of the incentives. In all simulations, the true distribution is generated as follows. We take 5 values $\{v_i\}$ in increasing order uniformly at random in $[0, 1]$, then sample those values uniformly at random 20 times. Those samples are then used to produce a Gaussian mixture model. Each Gaussian is centered on each of the samples and given equal weight. They all share the same variance which is $(\frac{v_5 - v_1}{2 * (5-1)})^2$. To produce the public distribution R , this distribution is sampled 10 times; let $\{s_i\}$ be the set of samples in increasing order. The samples are then used to produce a Gaussian mixture model in the same way as before, but with the variance set to $(\frac{s_{10} - s_1}{2 * (10-1)})^2$. For the Fixed Discretization Payments, the public distribution instead uses truncated Gaussian distributions with the bounds at 4 standard deviations from the mean. Agents update to their posteriors by taking $0.9 * R + 0.1 * K$ where K is the kernel distribution, which is the symmetric triangle. Computing K involves taking the CDF inverse map of $[q - \frac{b}{2}, q + \frac{b}{2}]$ where $q = \mathbb{E}_{y \sim \mathcal{Q}_R^{-1}(o)}[y]$ for an observation o . Let us call this interval $[o - l_b, o + r_b]$. We define the

b -radius as $d_b = \min(l_b, r_b)$. The Agent update kernel K then has a PDF which is a symmetric triangle in $[o - \gamma * d_b, o + \gamma * d_b]$ where $\gamma \in (0, 1)$. This update kernel trivially satisfies the sufficient update conditions. All Figures 5.4-5.6 show expected payments taken over both the true and posterior distributions for a fixed report, sampled from the true distribution. A constant 1 is subtracted to eliminate arbitrage. Error bars are $\frac{1}{5}$ standard deviations. The underlying distributions used for each simulation can be viewed in Section 5.4.4.

5.4.1 Report Perturbation

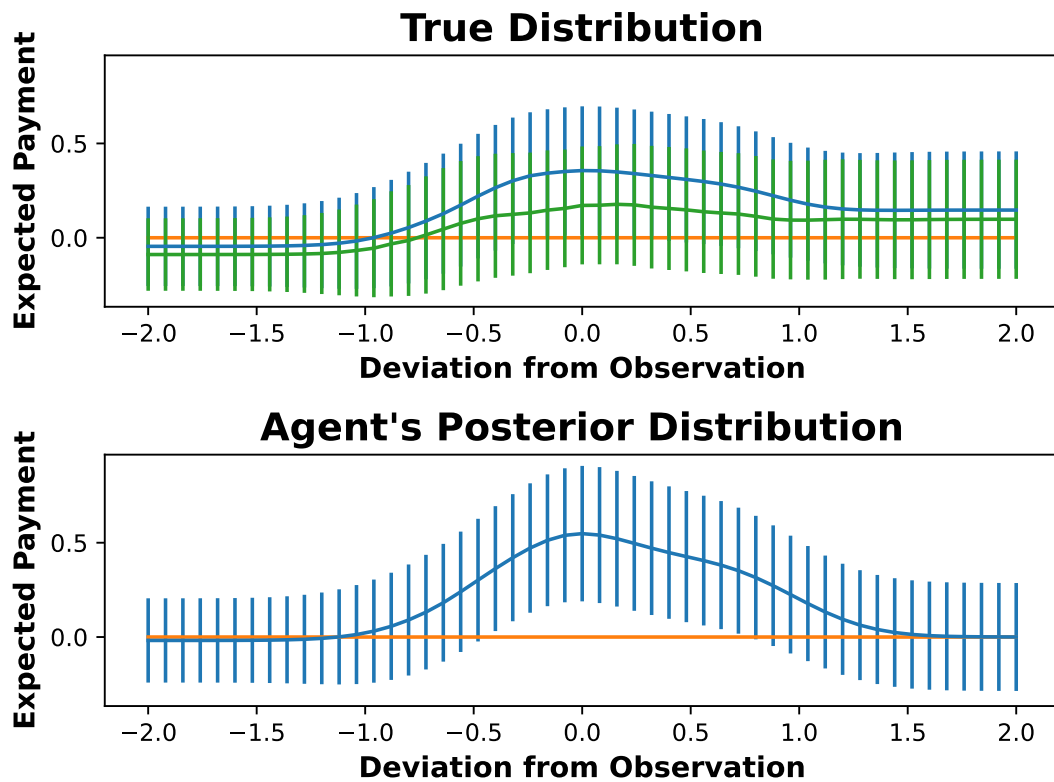


Figure 5.4: Expected payments over deviation from truthful. Green plot taken over 100 fixed peer reports.

We simulate the payments for an Agent reporting a point that is a perturbation of the observation, putting the observation at $x = 0$. Figure 5.8 shows the distributions which produced the results. Figure 5.4 shows the expected payments computed over the true distribution and the Agent's posterior distribution. The x-axis is scaled by d_b , so a deviation of $2 * d_b$ represents a point where, for a uniform distribution, the tent function on one side would be completely disjoint from the tent function centered at o . We observe that the Agent believes their payment will be maximized by truthfully reporting the observation, as expected from the theory. There is no incentive to deviate from the truthful report. While the error bars only represent one-fifth of a standard deviation, we see that the inherent variance is at a similar scale to the

payments. Since the variance of the payments scales inversely with the number of peers, only a small number of peers are required to make the variance scale significantly smaller than the payment scale. While the true payments also appear maximized at the observed point, the green plot shows this is not the case in general, especially with finite peer reports.

5.4.2 Tent Function Dependence

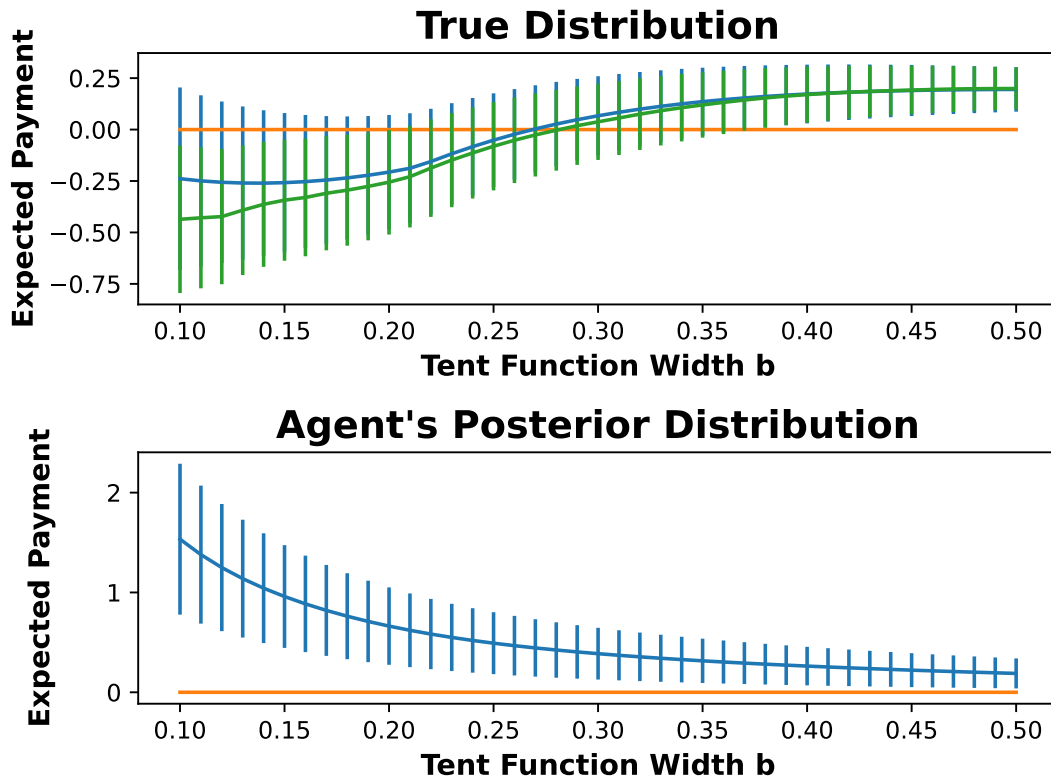


Figure 5.5: Expected payments over tent width b . Green plot taken over 100 fixed peer reports.

We simulate the payments with respect to the width of the tent function: b . Figure 5.8 shows the distributions which produced the results, with the Kernel and Posterior distributions only being shown for the largest value of b . Figure 5.5 shows the expected payments computed over the true distribution and the Agent's posterior distribution. We observe that the payments, computed over all distributions, decrease in variance as b is increased, which is to be expected because the payment function becomes flatter. We note that, although the expected payments over the posterior asymptotically decrease towards 0, the expected payments over the true distribution start below 0. This will not always be the case, but it illustrates an example when a region around the report is over-represented in the public distribution. Intuitively this would result in negative expected payments, since reporting a point in an over-represented region drives the public distribution even further away from the true distribution. We see that a tighter tent function results in higher expected payments because the tent function is

constructed to integrate to 1. For a Center designing a mechanism to overcome cost of effort, this must be taken into account when scaling the payments.

5.4.3 Fixed Discretization Payments

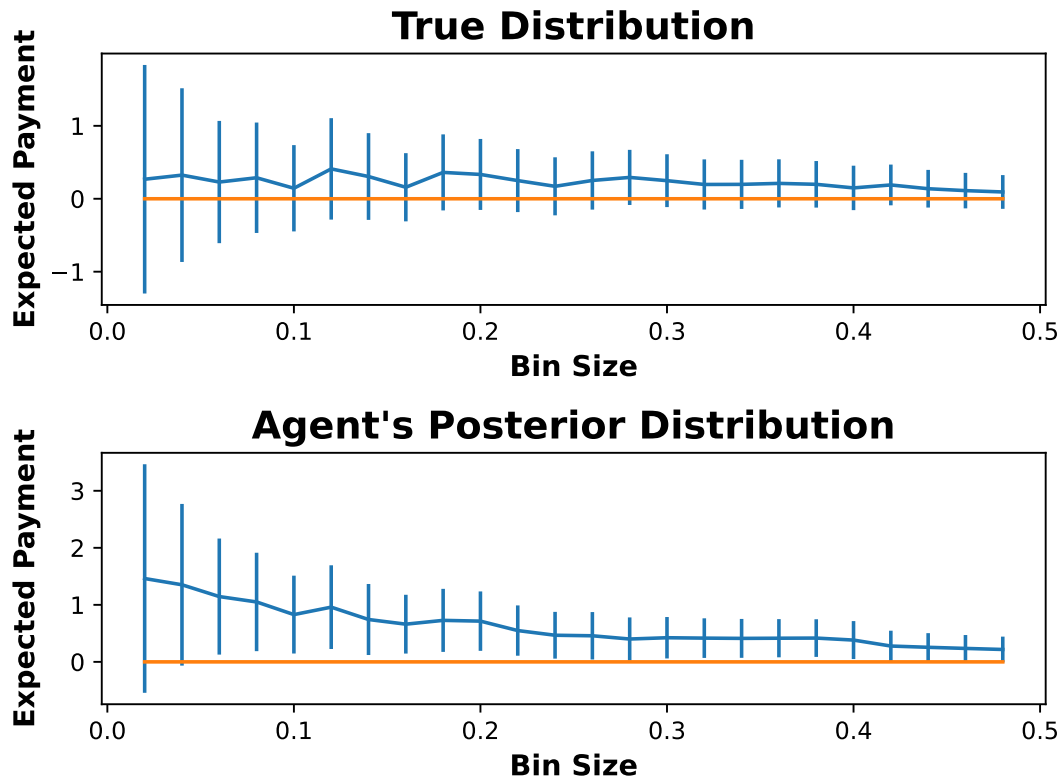


Figure 5.6: Expected payments over bin size for a fixed discretization. Plots averaged over 1000 observations from true distribution. This mechanism is only truthful up to the resolution of the bins.

We compare the payments according to the tent function to payments made using the Peer Truth Serum on fixed discretizations of the report space. Figure 5.10 shows the distributions which produced the results. Figure 5.6 shows an analogous plot to Figure 5.5, with expected payments computed for different widths of each "bin" in the discretization, normalized over the bounds of the distribution. We see that the tent function payments obtain a similar scale of variance compared to fixed discretizations, without suffering from the problem that the reports are only reliably truthful up to a certain resolution. Rather than determining the resolution of a "truthful" report, the width of the tent function only determines the precision of the update required for truthfulness. We also observe the payments when the report is a perturbation from the true report. Figure 5.7 shows an analogous plot to Figure 5.4, with a fixed bin size of 0.02. We see that the reward from the perspective of the Agent is maximized in the truthful bin, but there is a range of deviation that still produces this maximal reward.

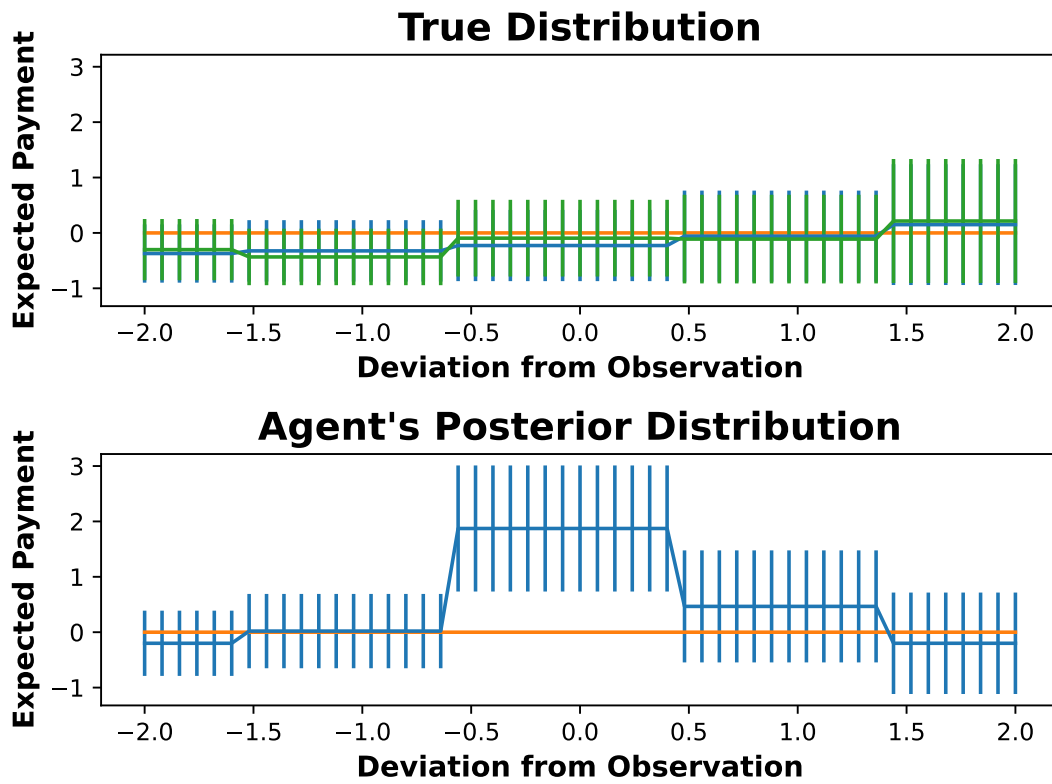


Figure 5.7: Expected payments over deviation from truthful with fixed discretization payment. Green plot taken over 100 fixed peer reports.

5.4.4 Distributions

Report Perturbations Figure 5.8 shows the distributions used for simulations in Section 5.4.1.

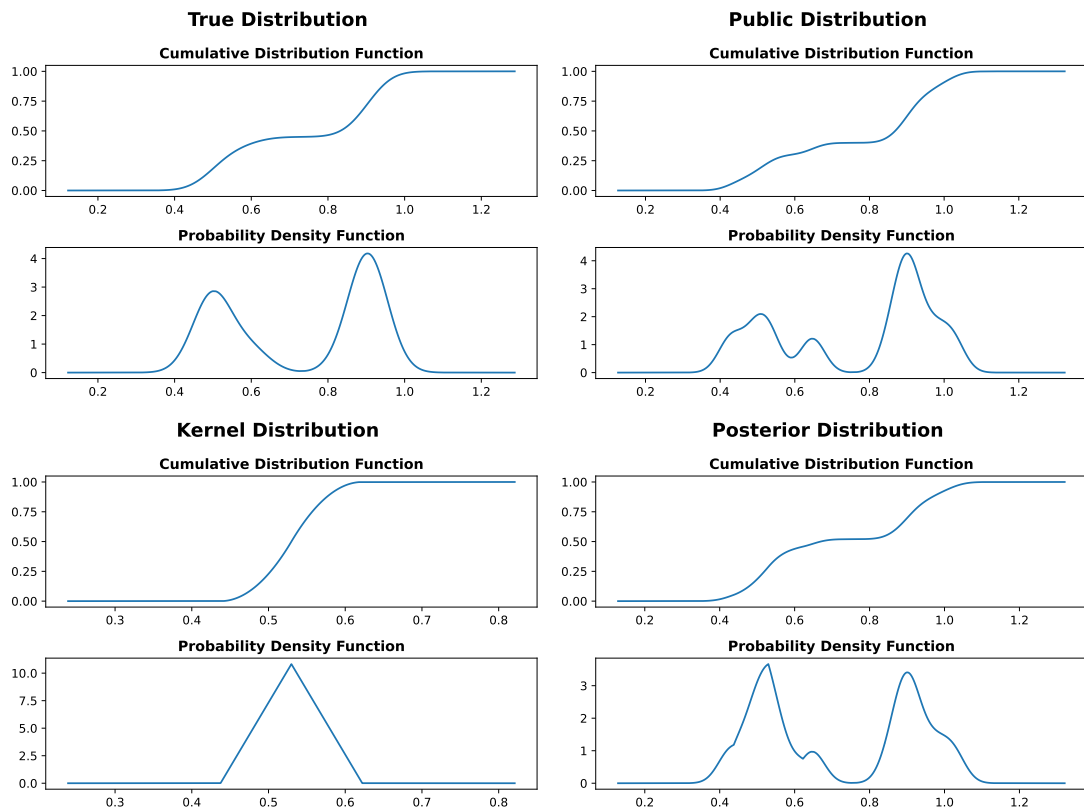


Figure 5.8: True, Public, Kernel, and Posterior distributions for Report Perturbations

Tent Function Dependence Figure 5.9 shows the distributions used for simulations in Section 5.4.2.

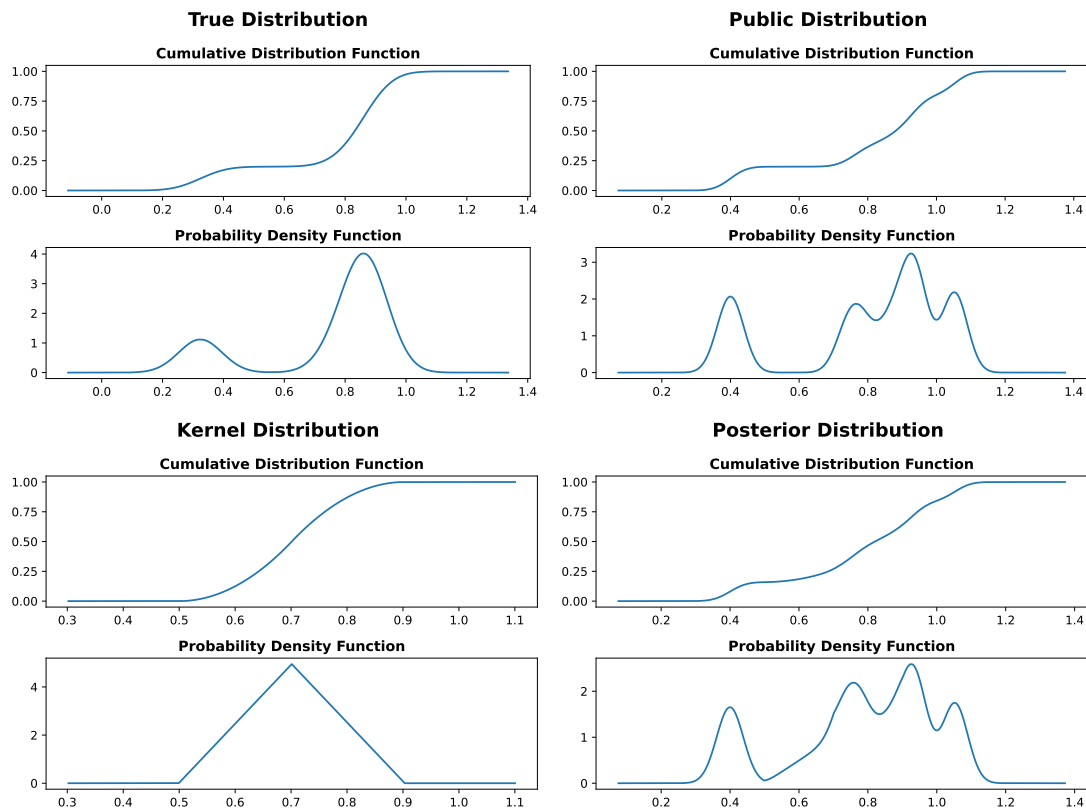


Figure 5.9: True, Public, Kernel, and Posterior distributions for Tent Function Dependence.

Fixed Discretization Payments Figures 5.10 and 5.11 shows the distributions used for simulations in Section 5.4.3. The Kernel and Posterior distributions are excluded for simulation for Figure 5.6 because the plots are averaged over many observations.

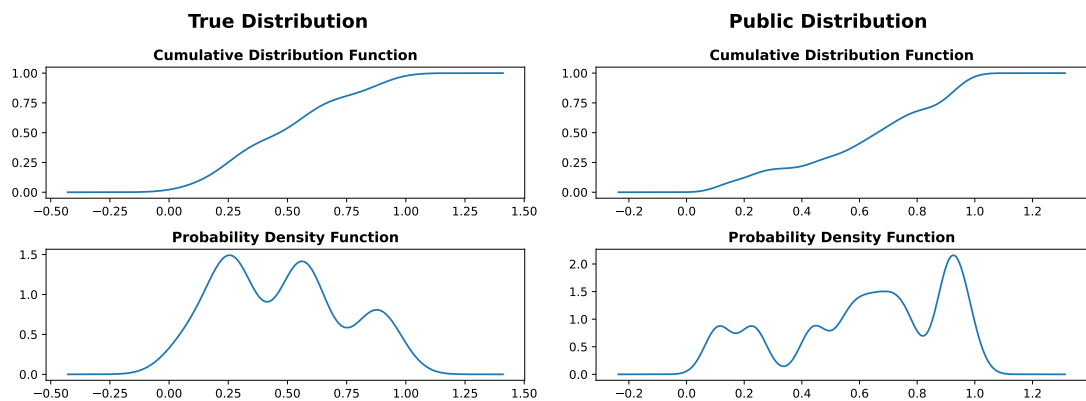


Figure 5.10: True and Public distributions for Fixed Discretization Payments.

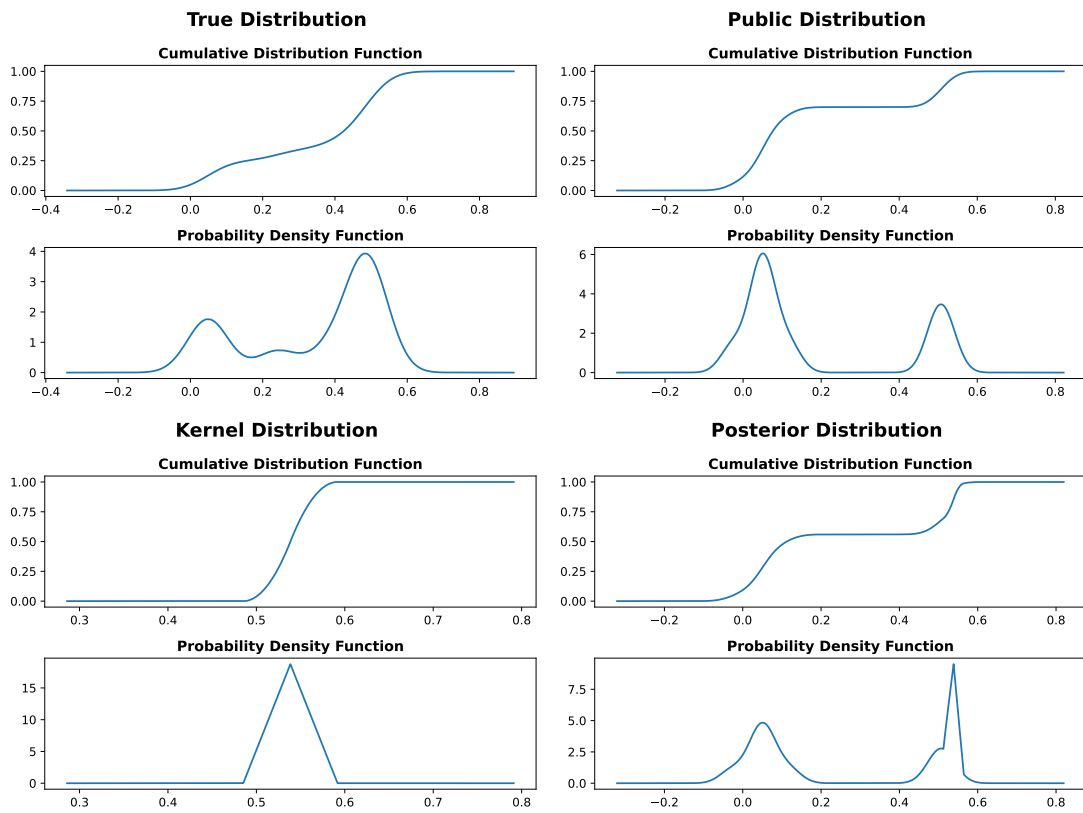


Figure 5.11: True, Public, Kernel, and Posterior distributions for Report Perturbations with Fixed Discretization Payments.

6 Conclusion

This thesis explores two methods of novel incentive mechanism design, specifically the design of Peer Prediction mechanisms. Peer Prediction mechanisms operate in the absence of any baseline metrics a Center can use to evaluate Agent reports, but they often struggle to accommodate arbitrary signal distributions. Most of the work in this direction has focused on achieving mechanisms on more general distributions by making stricter assumptions about prior details, increasing the the number and information embedded in Agent reports, or increasing the number of independent signal distributions.

6.1 Influence

Chapter two of this work follow along these lines by effectively applying stricter prior details. The additional restriction is that the Center is not trying to learn the signal distribution, but to learn a lower dimensional mapping between the variables representing the "inputs" and "labels" of the distribution, a classic supervised machine learning problem. We propose using the statistical measure known as Influence as the basis for an incentive mechanism, and we prove the mechanism's incentive-compatibility under certain prior detail assumptions about Agent belief updates. Specifically, we prove that a truthful Dominant Strategy Equilibrium exists when the validation set is composed of data accurately sampled from the signal distribution. It then follows that a truthful Bayes-Nash Equilibrium exists even in the absence of this validation set, when the validation samples need to be taken from Agent reports.

In addition to proving incentive-compatibility of the Influence-based mechanism, we cover a number of practical considerations for the implementation of such a mechanism. First we show that, in the case of least squares regression models, the truthful Dominant Strategy Equilibrium is maintained even if a fraction of the validation set is taken from Agent reports, and we analytically prove a bound on this fraction. Second, we address practical concerns about the computational expense of the mechanism. The computation of Influence for a large set of Agent reports can be prohibitively computationally expensive for large models, since each computation involves retraining the model. We demonstrate a theoretically sound

approximation method, extending the approximation in Koh and Liang, 2017, using Taylor expansions of the loss function. Third, we address economic considerations for the Center. One application for an Influence-based incentive mechanism is federated learning, but the economics of a federated learning system can be complex (Yu et al., 2020). It is important that the Center have some budgeting guarantees that are related to the quality of the data and the subsequent model. We show that the Center is capable of computing, a priori, an expected budget for building a model, taking into account the utility a Center can receive from constructing a model with a particular quality. This budgeting takes advantage of some statistical properties of Influence and model loss with respect to the number of data points in general. The budget can be computed exactly if Influences are computed sequentially with each report, but this can introduce unacceptable computational expense. We demonstrate how the Center can strike a balance between computation and budgeting accuracy with a batch processing method, and show analytically how the Center can apply an appropriate correction factor to undo the distortion from the batch processing on the expected budget.

Chapter three takes a brief aside from incentive mechanism design to consider other aspects of Influence. We consider the problem of data filtering, which is related to incentive mechanism design but has more stringent requirements. While a problem of incentive mechanism design can be solved by identifying truthful reports when all Agents report truthfully, thus demonstrating the existence of a truthful Bayes-Nash Equilibrium, a problem of filtering must be able to identify truthful reports in the presence of non-truthful, or corrupted, reports. This is similar to identifying Dominant Strategy Equilibria, but involves a more granular level of detail. Even if truthful data is not perfectly identifiable in certain contexts, if it can be differentiated from corrupted data, there is the potential to be able to improve the quality of the training set through Influence-based filtering.

We mainly attempt to differentiate truthful data from corrupted data by analyzing their expected Influence scores. We first show that, in the limit of infinite samples, Influence has the intuitive property that it gives higher expected scores to under-represented data in the training set when compared to the validation set. If the validation set is cleaner than the training set, in the sense that it has a higher proportion of truthful data, the expected Influence of truthful data will be higher than the expected Influence of corrupted data. But the reverse is also true. Unfortunately, this suggests that one cannot gain any "free lunch" from expected Influence: filtering according to the expected Influence would cause the quality of the training set to converge towards the quality of the validation set.

However, using another set of analysis, we show that under certain circumstances, perhaps a "free snack" can be obtained. The analysis assumes that, given finite training samples, optimal models form Gaussian distributions at input values. We see the "model posteriors" are Gaussian. Under this assumption, we show that the expected Influence scores of truthful or corrupted data depend on higher moments of the posteriors of the models produced by each distribution. This is important for demonstrating the efficacy of Influence-based filtering in a setting where Agents might add unbiased noise in an attempt to obfuscate their data for

privacy reasons. In the limit of infinite samples, this unbiased noise would not allow you to distinguish between noisy and clean samples, because they produce the same optimal model. But the finite sample analysis shows that these are distinguishable with the Influence score based on the fact that noisy samples would produce model posteriors with higher variance. Another consequence of this analysis is that, if the model posterior produced by the truthful and corrupted data are similar in variance, which might be the case of the corrupted data is produced by a set of Agents colluding to disrupt the model with a fake model, the group with the higher expected Influence score is determined by majority vote. This means that filtering according to expected Influence can be slightly robust to malicious collusion. In general, the finite analysis shows that under certain circumstances, the quality of the training set will not converge exactly to the quality of the validation set, rather, it will converge to a quality that is a slight perturbation of the validation quality. In this way, filtering the training set according to expected Influence can sometimes outperform the baseline validation quality.

We conduct simulations of Influence scores with different models and datasets in both the infinite sample and finite sample regimes, demonstrating that the theory and underlying assumptions are robust. We use the theory to propose a novel probabilistic filtering scheme and compare it to more intuitive deterministic filtering schemes, including a greedy but computationally expensive filtering schemes that can be considered "near-optimal". We find that the probabilistic filtering scheme obtains similar performance to the near-optimal filter in a trusted validation set regime, despite having far lower computational complexity. which obtains similar performance to a near-optimal filter in a trusted validation set regime, despite having far lower computational complexity. We conduct simulations to empirically verify the theoretical results for the infinite sample regime, finite sample regime, and the filtering schemes, which we conclude are robust.

6.1.1 Future Work

Much of the analysis on Influence-based mechanisms is specific to certain models or loss functions. The analysis on the robustness of the Dominant Strategy Equilibrium to some Agent data in the validation set is based on the linearity of least squares regression models. This analysis could be performed on a number of different salient loss functions with closed form optimizers or optimizers which mix linearly. Optimizers that don't satisfy the necessary constraints can be examined empirically. As shown empirically in Koh and Liang, 2017, sometimes the smooth Influence approximation can even be accurate for models with a high degree of stochasticity in the optimal model parameters. When assumptions are shown in this way to not always be critical to the applicability of a theory, this suggests that there might be a deeper theory.

There are many additional settings to consider for practical application of an Influence-based mechanism. For example, in federated learning often the data samples are held privately by the Agents and they only report model updates. Is it possible for a mechanism to compute the

Influences on the private data without revealing the private data? For limited cases, we have shown how to compute an Influence approximation privately (Richardson et al., 2020), but the general question remains open.

For the application of Influence to filtering, we focus our attention on being able to differentiate accurate data from corrupted data by the expected Influences of points. One could try to differentiate the data by examining higher moments of the distributions of Influences. It might even be possible in some cases to analytically solve for the distributions directly in the limit of infinite samples.

Alternatively, there are some statistical properties of Influence that can be exploited for filtering. For example, if one has an estimate of the proportions of accurate and corrupted data in the data set, one might be able to say something about the relative orderings of accurate and corrupted data based on the Influence score. For example, if there is a large proportion of accurate data in both training and validation, the small number of corrupted samples are expected to achieve a very high Influence score measured against the small number of corrupted samples in validation. It may be possible to identify these points by examining the Influences of every pair of training and validation points.

6.2 Peer Neighborhoods

In chapter four we present a novel framework, which we call Peer Neighborhoods, for extending existing Peer Prediction mechanisms so that they may accommodate arbitrary distributions. The extension involves the Center choosing a set of partitions with specific properties which can be used to discretize the space of reports. Existing Peer Prediction mechanisms that can only be applied to discrete distributions can then be utilized over this set of partitions. Rather than follow the previous paradigm of assuming some additional structure in the prior details, allowing the mechanism to elicit more information than just the Agents' types, or leveraging the power of multiple tasks, we work backwards from the mechanism to discover the belief structures that satisfy incentive-compatibility. These belief structures are encoded in a specific form called belief update conditions.

We present a belief update condition, the Partition Expected extension, that satisfies incentive-compatibility for Peer Neighborhood extensions, and analyze a specific instance of this condition for the Peer Neighborhood extension of the Peer Truth Serum. We show that the condition still admits a broad class of update processes, and present an example of such an update using what we call pyramid kernels. The proof that these pyramid kernels satisfy the condition further suggest a method for computing the kernels. Finally, to address practical implementation concerns, we conduct simulations to demonstrate the strength of the incentives with respect to perturbations from truthfulness, and the stability of payments with respect to the bin size of the partitions chosen by the Center. In some examples we use the pyramid kernel update and demonstrate that it satisfies the condition by showing that the highest expected payment occurs at the truthful report.

In chapter five we expand on the theory in the previous chapter. We explain how the Peer Neighborhoods framework is actually too unconstrained, and that there is a natural choice for how to specifically construct the set of partitions based on a shared prior belief. Analyzing this concept, we find that the extended mechanism can be described better in a functional form, rather than relying on the partition set. With some modifications, we present the functional form extension of the Peer Truth Serum, which we call the Continuous Truth Serum. We show how this mechanism solves another theoretical problem not addressed generally by the Peer Neighborhood framework: the problem of orthogonal components in the prior. In a Peer Prediction setting with a discrete signal distribution, this would correspond to the problem of an unobserved, unknown category in the distribution. This results in degenerate payments for many classical Peer Prediction mechanisms, including the Peer Truth Serum, but is handled sensibly by the Continuous Truth Serum.

As in the previous chapter, we analyze the conditions under which the Continuous Truth Serum is Bayesian-Nash Incentive-Compatible. While the most general sufficient and necessary condition is difficult to break down into an easily comprehensible set of rules for an Agent, we show that there is a reasonable set of sufficient update conditions which merely enforce notions of locality and symmetry in continuous distributions. Although this analysis is specific to this extension of the Peer Truth Serum, the concepts involved in producing this functional form of a Peer Neighborhood framework can be applied to many existing mechanisms. We conduct simulations along the same lines as in the previous chapter, again demonstrating the strength and stability of the incentives. The Continuous Truth Serum presents the broadest theory of discrete Peer Prediction mechanism extension to date, and it is practically implementable.

6.2.1 Future Work

We present Peer Neighborhoods as a framework for extending Peer Prediction mechanisms which operate on discrete distributions. The example we work with is the extension of the Peer Truth Serum, but there are other extensions worth analyzing in detail. One such mechanism is the Correlated Agreement mechanism, which pays according to a matrix representing correlations between signals across multiple tasks. The setting and mechanism are far more complicated than those of the Peer Truth Serum, which is a minimal single-task mechanism, but the Peer Neighborhoods framework would still apply. The critical analysis comes in examining the prior details about Agent beliefs that would satisfy incentive-compatibility for an extension of Correlated Agreement. There is also a detail-free version of Correlated Agreement. The extension would require some details to be added, and it would be worth exploring whether or not these details have a valid real world interpretation.

The work in chapter five also shows that there is some incompleteness in chapter four regarding extensions of update conditions. We present the "Partition-Expected" update extension as the natural extension for the Peer Neighborhoods framework, and demonstrate the exis-

tence of a broad class of possible updates under this condition. It is intuitive because the Peer Neighborhood extension is an expectation over a set of discrete mechanisms, so the update condition should be an expectation over discrete conditions. However, the analysis in chapter five shows that this natural extension is sufficient, but not necessary. In particular, the Partition-Expected update extension of the self-predicting condition for the Peer Truth Serum enforces boundedness of additive update kernels. For the Continuous Truth Serum, the specific extension of the Peer Truth Serum with equal probability partition bins, we find boundedness to be a useful element of sufficient conditions to characterize the class of admissible updates, but it is not a necessary condition. It would be worth exploring why the Partition-Expected update extension appears to be stronger than necessary, and is there a universal way to extend update conditions in a way that is necessary and sufficient.

Bibliography

- Cai, Y., Daskalakis, C., & Papadimitriou, C. (2015). Optimum statistical estimation with strategic data sources. *Conference on Learning Theory*, 280–296.
- Cantelli, F. P. (1933). Sulla determinazione empirica delle leggi di probabilita. *Giorn. Ist. Ital. Attuari*, 4(421-424).
- Chai, Z., Ali, A., Zawad, S., Truex, S., Anwar, A., Baracaldo, N., Zhou, Y., Ludwig, H., Yan, F., & Cheng, Y. (2020). Tifl: a tier-based federated learning system. *Proceedings of the 29th international symposium on high-performance parallel and distributed computing*, 125–136.
- Che, C., Li, X., Chen, C., He, X., & Zheng, Z. (2022). A decentralized federated learning framework via committee mechanism with convergence guarantee. *IEEE Transactions on Parallel and Distributed Systems*, 33(12), 4783–4800.
- Cook, R. D., & Weisberg, S. (1980). Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics*, 22(4), 495–508.
- Cook, R. D., & Weisberg, S. (1982). Residuals and influence in regression.
- Cortez, P., Cerdeira, A., Almeida, F., Matos, T., & Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4), 547–553. <https://archive.ics.uci.edu/ml/datasets/Wine+Quality>
- Dasgupta, A., & Ghosh, A. (2013). Crowdsourced judgement elicitation with endogenous proficiency. *Proceedings of the 22nd international conference on World Wide Web*, 319–330.
- Dasgupta, P., Hammond, P., & Maskin, E. (1979). The implementation of social choice rules: some general results on incentive compatibility. *The Review of Economic Studies*, 46(2), 185–216.
- d'Aspremont, C., & Gérard-Varet, L.-A. (1979). Incentives and incomplete information. *Journal of Public economics*, 11(1), 25–45.
- Dasu, T., & Loh, J. M. (2012). Statistical distortion: consequences of data cleaning. *arXiv preprint arXiv:1208.1932*.
- De Vito, S., Massera, E., Piga, M., Martinotto, L., & Di Francia, G. (2008). On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. *Sensors and Actuators B: Chemical*, 129(2), 750–757.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6), 141–142.

- Dua, D., & Graff, C. (2017). UCI machine learning repository. <http://archive.ics.uci.edu/ml>
- Dwork, C. (2008). Differential privacy: a survey of results. *International conference on theory and applications of models of computation*, 1–19.
- Faltings, B. (n.d.). Game-theoretic mechanisms for eliciting accurate information.
- Faltings, B., Jurca, R., Pu, P., & Tran, B. D. (2014). Incentives to counter bias in human computation. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 2, 59–66.
- Faltings, B., Jurca, R., & Radanovic, G. (2017). Peer truth serum: incentives for crowdsourcing measurements and opinions. *arXiv preprint arXiv:1704.05269*.
- Feng, S., Niyato, D., Wang, P., Kim, D. I., & Liang, Y.-C. (2019). Joint service pricing and cooperative relay communication for federated learning. *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (Green-Com) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 815–820.
- Frank, M. R., Cebrian, M., Pickard, G., & Rahwan, I. (2017). Validating bayesian truth serum in large-scale online human experiments. *PloS one*, 12(5), e0177385.
- Ghorbani, A., & Zou, J. (2019). Data shapley: equitable valuation of data for machine learning. *International conference on machine learning*, 2242–2251.
- Gibbard, A. (1973). Manipulation of voting schemes: a general result. *Econometrica: journal of the Econometric Society*, 587–601.
- Glivenko, V. (1933). Sulla determinazione empirica delle leggi di probabilita. *Gion. Ist. Ital. Attauri.*, 4, 92–99.
- Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477), 359–378.
- Goel, N., & Faltings, B. (2020). Personalized peer truth serum for eliciting multi-attribute personal data. *Uncertainty in Artificial Intelligence*, 18–27.
- Hanson, R. (2007). Logarithmic markets coring rules for modular combinatorial information aggregation. *The Journal of Prediction Markets*, 1(1), 3–15.
- Holmström, B. R. (1978). *On incentives and control in organizations*. Stanford University.
- Hooda, N., Bawa, S., & Rana, P. S. (2018). Fraudulent firm classification: a case study of an external audit. *Applied Artificial Intelligence*, 32(1), 48–64.
- Huang, S.-W., & Fu, W.-T. (2013). Enhancing reliability using peer consistency evaluation in human computation. *Proceedings of the 2013 conference on Computer supported cooperative work*, 639–648.
- Ilyas, I. F., & Chu, X. (2019). *Data cleaning*. Morgan & Claypool.
- Jia, R., Dao, D., Wang, B., Hubis, F. A., Hynes, N., Gürel, N. M., Li, B., Zhang, C., Song, D., & Spanos, C. J. (2019). Towards efficient data valuation based on the shapley value. *The 22nd International Conference on Artificial Intelligence and Statistics*, 1167–1176.
- Jurca, R., & Faltings, B. (2003). An incentive compatible reputation mechanism. *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 1026–1027.

- Koh, P. W., & Liang, P. (2017). Understanding black-box predictions via influence functions. *International conference on machine learning*, 1885–1894.
- Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., & Bacon, D. (2016). Federated learning: strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.
- Kong, Y., & Schoenebeck, G. (2016). Equilibrium selection in information elicitation without verification via information monotonicity. *arXiv preprint arXiv:1603.07751*.
- Kong, Y., & Schoenebeck, G. (2019). An information theoretic framework for designing information elicitation mechanisms that reward truth-telling. *ACM Transactions on Economics and Computation (TEAC)*, 7(1), 1–33.
- Lai, F., Zhu, X., Madhyastha, H. V., & Chowdhury, M. (2021). Oort: efficient federated learning via guided participant selection. *15th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 21)*, 19–35.
- Loog, M., Viering, T., & Mey, A. (2019). Minimizers of the empirical risk and risk monotonicity. *Advances in Neural Information Processing Systems*, 32.
- Loughran, T. A., Paternoster, R., & Thomas, K. J. (2014). Incentivizing responses to self-report questions in perceptual deterrence studies: an investigation of the validity of deterrence theory using bayesian truth serum. *Journal of Quantitative Criminology*, 30, 677–707.
- Luengo, J., Shim, S.-O., Alshomrani, S., Altalhi, A., & Herrera, F. (2018). Cnc-nos: class noise cleaning by ensemble filtering and noise scoring. *Knowledge-Based Systems*, 140, 27–49.
- Mansour, S., Wuebker, J., & Ney, H. (2011). Combining translation and language model scoring for domain-specific data filtering. *Proceedings of the 8th International Workshop on Spoken Language Translation: Papers*.
- McAfee, R. P., & Reny, P. J. (1992). Correlated information and mechanism design. *Econometrica: Journal of the Econometric Society*, 395–421.
- Miller, N., Resnick, P., & Zeckhauser, R. (2005). Eliciting informative feedback: the peer-prediction method. *Management Science*, 51(9), 1359–1373.
- Miller, S. R., Bailey, B. P., & Kirlik, A. (2014). Exploring the utility of bayesian truth serum for assessing design knowledge. *Human-Computer Interaction*, 29(5-6), 487–515.
- Miranda, C. (1940). *Un'osservazione su un teorema di brouwer*. Consiglio Nazionale delle Ricerche.
- Myerson, R. B. (1979). Incentive compatibility and the bargaining problem. *Econometrica: journal of the Econometric Society*, 61–73.
- Prelec, D. (2004). A bayesian truth serum for subjective data. *science*, 306(5695), 462–466.
- Radanovic, G., & Faltings, B. (2013). A robust bayesian truth serum for non-binary signals. *Proceedings of the AAAI Conference on Artificial Intelligence*, 27(1), 833–839.
- Radanovic, G., & Faltings, B. (2014). Incentives for truthful information elicitation of continuous signals. *Proceedings of the AAAI Conference on Artificial Intelligence*, 28.

- Radanovic, G., & Faltings, B. (2015a). Incentive schemes for participatory sensing. *Proceedings of the 14th international conference on autonomous agents and multiagent systems (AAMAS'15)*, (CONF), 1081–1089.
- Radanovic, G., & Faltings, B. (2015b). Incentivizing truthful responses with the logarithmic peer truth serum. *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, 1349–1354.
- Radanovic, G., Faltings, B., & Jurca, R. (2016). Incentives for effort in crowdsourcing using the peer truth serum. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 7(4), 1–28.
- Rahm, E., Do, H. H., et al. (2000). Data cleaning: problems and current approaches. *IEEE Data Eng. Bull.*, 23(4), 3–13.
- Redmond, M., & Baveja, A. (2002). A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research*, 141(3), 660–678. <https://archive.ics.uci.edu/ml/datasets/Communities+and+Crime>
- Richardson, A., Filos-Ratsikas, A., Rokvic, L., & Faltings, B. (2020). Privately computing influence in regression models. *AAAI 2020 Workshop on Privacy-Preserving Artificial Intelligence*, 188.
- Shapley, L. S., et al. (1953). A value for n-person games.
- Shnayder, V., Agarwal, A., Frongillo, R., & Parkes, D. C. (2016). Informed truthfulness in multi-task peer prediction. *Proceedings of the 2016 ACM Conference on Economics and Computation*, 179–196.
- Soltani, B., Zhou, Y., Haghghi, V., & Lui, J. (2023). A survey of federated evaluation in federated learning. *arXiv preprint arXiv:2305.08070*.
- Soto, C. S., Fasnacht, M., Zhu, J., Forrest, L., & Honig, B. (2008). Loop modeling: sampling, filtering, and scoring. *Proteins: Structure, Function, and Bioinformatics*, 70(3), 834–843.
- Surowiecki, J. (2005). *The wisdom of crowds*.
- Tsanas, A., Little, M. A., McSharry, P. E., & Ramig, L. O. (2009). Accurate telemonitoring of parkinson's disease progression by noninvasive speech tests. *IEEE transactions on Biomedical Engineering*, 57(4), 884–893. <https://archive.ics.uci.edu/ml/datasets/Parkinsons+Telemonitoring>
- Von Ahn, L., & Dabbish, L. (2004). Labeling images with a computer game. *Proceedings of the SIGCHI conference on Human factors in computing systems*, 319–326.
- Von Neumann, J., & Morgenstern, O. (2007). *Theory of games and economic behavior (60th anniversary commemorative edition)*. Princeton university press.
- Von Stackelberg, H. (2010). *Market structure and equilibrium*. Springer Science & Business Media.
- Waggoner, B., & Chen, Y. (2014). Output agreement mechanisms and common knowledge. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 2, 220–226.
- Weaver, R., & Prelec, D. (2013). Creating truth-telling incentives with the bayesian truth serum. *Journal of Marketing Research*, 50(3), 289–302.

- Witkowski, J., & Parkes, D. (2012). A robust bayesian truth serum for small populations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 26(1), 1492–1498.
- Witkowski, J., & Parkes, D. C. (2012). Peer prediction without a common prior. *Proceedings of the 13th ACM Conference on Electronic Commerce*, 964–981.
- Wu, H., & Wang, P. (2022). Node selection toward faster convergence for federated learning on non-iid data. *IEEE Transactions on Network Science and Engineering*, 9(5), 3099–3111.
- Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2), 1–19.
- Yu, H., Liu, Z., Liu, Y., Chen, T., Cong, M., Weng, X., Niyato, D., & Yang, Q. (2020). A sustainable incentive scheme for federated learning. *IEEE Intelligent Systems*, 35(4), 58–69.
- Zhang, J., Wu, Y., & Pan, R. (2021). Incentive mechanism for horizontal federated learning based on reputation and reverse auction. *Proceedings of the Web Conference 2021*, 947–956.
- Zhao, J., Chang, X., Feng, Y., Liu, C. H., & Liu, N. (2022). Participant selection for federated learning with heterogeneous data in intelligent transport system. *IEEE transactions on intelligent transportation systems*, 24(1), 1106–1115.

Adam Richardson

+1 973 699 1682 adamjri@gmail.com Google Scholar adamjri

Summary

Experienced researcher in artificial intelligence, machine learning, multi-agent systems, algorithmic game theory, and incentive mechanism design. Successfully applying probabilistic modeling to real world problems involving large numbers of independent, self-interested entities for the purpose of facilitating coordination. Methods have demonstrated success in contexts such as crowdsourcing, data sharing, and data filtering. Expertise in rapidly prototyping and iterating solutions in Python, as well as implementing scientific computing solutions in C++.

Education





- BA Columbia University**, Math, Computer Science Sept. 2013 to May 2017
- GPA: 3.4/4.0
 - **Coursework:** Data Structures and Algorithms, Machine Learning, Complexity Theory, Microprocessor Architecture, Probability Theory, Topology, Complex and Real Analysis
- PhD École Polytechnique Fédérale de Lausanne**, Artificial Intelligence Lab under Prof. Boi Faltings Sept. 2018 to Dec. 2023
- Thesis: Extensions of Peer Prediction Incentive Mechanisms
 - **Coursework:** Multi-Agent Systems, Probability Theory, Geometric Computing

Experience

- Columbia University Robotics Lab**, Intern under Prof. Paul Allen NY, USA
- Implemented novel algorithm for merging dense surface point clouds with sparse volumetric voxel grids to produce mixed-resolution surface meshes of 3D objects. June 2016 to Aug. 2016
3 months
 - Implemented a novel mesh evaluation metric based on topological features for the purpose of estimating robotic grasp planning similarities.
 - Published a paper on Shape Completion Enabled Robotic Grasping in IROS 2016.
- Qualcomm**, Intern under Dr. Michael Luby CA, USA
- Developed a low level car driving simulator for the purpose of testing autonomous driving algorithms. June. 2017 to Aug. 2017
3 months
 - Implemented and tested numerous Kalman filter variants, synthesizing car sensory inputs for the purpose of localization.
- École Polytechnique Fédérale de Lausanne**, Research Intern under Prof. Rüdiger Urbanke Lausanne, Switzerland
- Investigated different theories of model complexity in neural network models to address over-fitting. Sept. 2018 to May 2019
8 months
 - Applied a novel form of model complexity to the theory of model dropout, demonstrating a negative link between complexity and dropout.

Publications

- Shape Completion Enabled Robotic Grasping** Jan. 2017
- Jacob Varley, Chad DeChant, **Adam Richardson**, Joaquín Ruales, Paul Allen
 Proceedings: IROS 2017 [↗](#)

Rewarding High-Quality Data via Influence Functions	2019
<i>Adam Richardson</i> , Aris Filos-Ratsikas, Boi Faltings 10.48550/arXiv.1908.11598 	
Budget-Bounded Incentives for Federated Learning	2020
<i>Adam Richardson</i> , Aris Filos-Ratsikas, Boi Faltings Book Chapter: "Federated Learning. Lecture Notes in Computer Science", Springer, Cham 	
Budget-Bounded Incentives for Federated Learning	2020
<i>Adam Richardson</i> , Aris Filos-Ratsikas, Ljubomir Rokvic, Boi Faltings AAAI 2020 Workshop on Privacy-Preserving Artificial Intelligence 	
Peer Neighborhood Mechanisms: A Framework for Mechanism Generalization	2024
<i>Adam Richardson</i> , Boi Faltings Proceedings: AAAI 2024 	

Additional Experience

Instructor (2019 - 2023): Teaching Assistant for eight full-credit Computer Science courses.

Technologies

Languages: Python, C++, C, Java, JavaScript, C#, SQL

Frameworks: Keras, Tensorflow, PyTorch, React, Node.js

Software: Visual Studio, Eclipse, MySQL Server and Workbench, Insomnia