EPFL

Thèse n° 9317

Kernel Methods and Similarity Learning Applied in Computational Chemistry

Présentée le 16 février 2022

Faculté des sciences de base
Laboratoire de design moléculaire computationnel
Programme doctoral en chimie et génie chimique

pour l'obtention du grade de Docteur ès Sciences

par

Raimon FABREGAT I DE AGUILAR-AMAT

Acceptée sur proposition du jury

Prof. J. Vanicek, président du jury
Prof. A.-C. Corminboeuf, directrice de thèse
Prof. Ph. Marquetand, rapporteur
Dr L. Roch, rapporteur
Dr F. Aquilante, rapporteur

École polytechnique fédérale de Lausanne

2022

The task of philosophy is not to provide answers, but to show how the way we perceive a problem can be itself part of a problem.

— Slavoj Zizek

Acknowledgements

First of all I want to thank my supervisor Prof. Clémence Corminboeuf for her support, guidance and infinite patience. Her enthusiasm for research and scientific curiosity have been a constant source of inspiration to explore new ideas. I am truly grateful to her for having given me the opportunity to be part of LCMD for the last 4 years, an unrepeatable experience that I truly enjoyed and that I will always remember with great affection.

I want to express my special gratitude to Dr. Alberto Fabrizio and Dr. Benjamin Meyer, who have been an enormous source of help and encouragement. I want to thank also Simone Gallarati and Veronika Jurásková for our hiking trips, which helped to keep the morale high through the worst parts of 2020. I am very grateful to my partner and dearest friend Eugenia Oshurko, for her infinite love and care, and to my parents and my sister for always being there. Finally, I want to thank the rest of the LCMD members that have accompanied me through this journey: Véronique Bujard, Matthew Wodrich, Daniel Jana, Antonio Prlj, Laurent Vannay, Štěpán Růžička, Daniel Hollas, Anya Gryn'ova, Giulia Mangione, Boodsarin Sawatlon, Kun-Han Lin (*a.k.a.* Mr. Han), the Catalan crew (Sergi Vela & Maria Fumanal), Tovarish Ksenia Briling, Terence J. Blaskovits, Shubhajit Das, Ruben Laplaza, Marc Hamilton, Frédéric Célerse and Puck van Gerwen. I wish the best to all of them.

I am grateful to all the collaborators who contributed to the research presented in this thesis. These are Dr. Alberto Fabrizio, Dr. Benjamin Meyer, Dr. Daniel Hollas, Dr. Edgar Engel, Veronika Juraskova, Simone Gallarati, Dr. Ruben Laplaza, Sinjini Bhattacharjee, Dr. Matthew Wodrich, Prof. Michele Ceriotti and Prof. Clémence Corminboeuf.

I want to express my gratitude to Prof. Jiří Vaníček, Dr. Francesco Aquilante, Prof. Philipp Marquetand and Dr. Loïc Roch for having accepted to be on my jury.

The National Centre of Competence in Research (NCCR), the Swiss National Science Foundation (SNSF), the European Research Council (ERC) and the EPFL are acknowledged for financial support.

Lausanne, December 15, 2021

Abstract

Over the last two decades, data-powered machine learning (ML) tools have profoundly transformed numerous scientific fields. In computational chemistry, machine learning applications have permitted faster predictions of chemical properties and provided powerful analytical tools, facilitating the exploration of the chemical space. The original work presented in this thesis leverages the paradigm-shifting influence of ML and focuses on bridging the divide between unsupervised and supervised learning with the overarching objective of improving the predictive power of similarity-based machine learning algorithms such as kernel regression.
Despite their widespread use in chemistry, current implementations of kernel regression suffer from biased definitions of similarity between chemical environments. This problematic originates from the rigidity of current numerical approaches for encoding molecular information, based on expert-crafted representations. Moreover, it is amplified by the incorrect (yet generalized) assumption that increasing the amount of information encoded in molecular representations unequivocally improves the evaluation of molecular similarity. As a result, the performance of kernel models can be sub-optimal, reducing their broad applicability.
To overcome such limitations, we introduce a series of statistical tools and methodologies based on supervised dimensionality reduction and metric learning capable of filtering and adapting the features of common molecular representations. This allows tailoring the notion of "molecular similarity" in order to optimize the prediction of specific chemical targets.
Using examples such as the exploration of the free-energy landscape of oligopeptides or the prediction of subtle properties associated with the outcome of chemical reactions (*i.e.*, enantiomeric excess), we demonstrate how the methods proposed in this thesis unlock the optimal performance of kernel regression and, more generally, of any similarity-based algorithm.
Overall, the work presented in this manuscript is part of a larger, more comprehensive effort aimed at extending the capabilities of computational modeling to increasingly complex chemical situations by exploiting the latest advances in statistical learning.

Keywords: Machine learning, molecular similarity, metric learning, computational chemistry, potential energy surfaces, statistical sampling.

Résumé

Au cours des vingt dernières années, les outils d'apprentissage automatique (ML, pour Machine Learning en anglais) alimentés par les données ont profondément transformé de nombreux domaines scientifiques. En chimie computationnelle, les applications d'apprentissage automatique permettent une prédiction plus rapide des propriétés chimiques quantiques et servent de base à de puissants outils analytiques qui facilitent l'exploration de l'espace chimique. Le travail original présenté dans cette thèse reflète ce changement de paradigme apporté par le ML et se concentre sur la connexion entre l'apprentissage supervisé et non supervisé, avec l'objectif primordial d'améliorer le pouvoir prédictif des algorithmes d'apprentissage automatique basés sur la similarité, tels que la régression kernel.
Malgré leur utilisation répandue en chimie, les implémentations actuelles de la régression kernel souffrent de définitions biaisées de la similitude entre les environnements chimiques. Cette problématique trouve son origine dans la rigidité des approches numériques actuelles pour encoder l'information moléculaire basées sur des représentations élaborées par des experts. De plus, le probleme est amplifié par l'hypothèse incorrecte, mais généralisée, selon laquelle l'augmentation de la quantité d'informations encodées dans les représentations moléculaires améliore sans équivoque l'évaluation de la similarité moléculaire. En conséquence, les performances des modèles kernel peuvent être sous-optimales et leur applicabilité en être considérablement réduite.
Pour surmonter ces limitations, nous introduisons une série d'outils et de méthodologies statistiques basés sur la réduction supervisée de la dimensionnalité et l'apprentissage de métriques capables de filtrer et d'adapter les caractéristiques des représentations moléculaires courantes. Cela permet d'adapter la notion de "similarité moléculaire" afin d'optimiser la prédiction de cibles chimiques spécifiques. A l'aide d'exemples tels que l'exploration de surfaces d'énergie libre des oligopeptides ou de la prédiction de propriétés subtiles associées au résultat de réactions chimiques (*ie*, excès énantiomérique), nous démontrons comment les méthodes proposées dans cette thèse débloquent le performances de la régression kernel et, plus généralement, de tout algorithme basé sur la similarité.
Plus largement, les travaux exposés sont ancrés dans un effort plus vaste et plus complet visant à améliorer les capacités de modélisation informatique dans des contextes chimiques de plus en plus complexes, et ce en exploitant les dernières avancées en matière d'apprentissage automatique.

Mots clefs : apprentissage automatique, similitude moléculaire, apprentissage des métriques, chimie computationnelle, surfaces d'énergie potentielle, échantillonnage statistique.

Contents

List of Figures

1 Introduction

Machine learning (ML), algorithms to perform inference and construct prediction models from raw data, is rapidly becoming a fundamental technique in scientific research. The field of chemistry is especially well-positioned to benefit from these computational techniques, as there is a great deal of interest in extracting relationships and patterns from highly non-intuitive datasets that can subsequently be used to build predictive models. Indeed, the first examples of ML applications to chemical data can be traced back over 50 years. The specific use of supervised learning methods, which aimed to create maps between input and target variables, originates from the 1970s with examples ranging from predicting synthetic routes for organic molecules[16] to protein secondary structure.[17,18] Other notably examples include constructing structure-activity relationships to predict quantities such as mutagenicity[19] and to assist drug design.[20] As a complement to supervised learning, unsupervised learning algorithms, a series of techniques that learn patterns and elucidate structures in unlabeled data[21] (*e.g.*, factor analysis and Principal Component Analysis (PCA)) have been exploited since the 1960s[22] to interpret multivariate data and aid in the experimental design of spectroscopy, chromatography, and chemometrics,[23] as well as to elucidate the behaviour of collective motions in dynamical simulations.[24,25] However, as in many other disciplines, the use of ML methods in chemistry remained rather peripheral until about a decade ago. The scarcity of datasets associated with the difficulties in accessing, storing and sharing data, combined with the unavailability of computational power and the relatively low sophistication or open-access of ML algorithms severely limited the applicability of these methods.

Beginning in the 2000s, the combination of digital storage, personal computers with increasing power, along with the ability to share information through the internet allowed the potential of machine learning and artificial intelligence tools to be fully unlocked. Today, these factors combined with the exponential growth of data allow desktop machines to routinely solve problems that could only be tackled by supercomputers just a few decades ago. The "Big Data" revolution of the last twenty years has been a by-product of these phenomena, affecting all scientific fields to various degrees, sometimes radically. This revolution has, correspondingly, led to a boom in the application of ML methods to a wide variety of fields, which in turn has

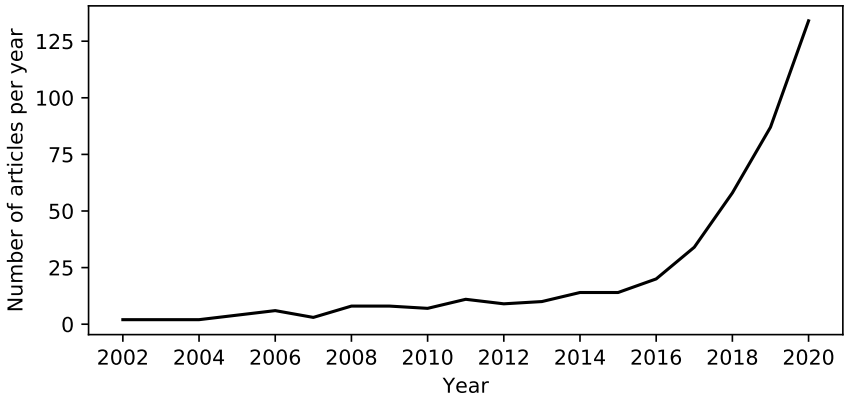


Figure 1.1 – Publications each year from a web of science search with topics of "machine learning" and "computational chemistry".

accelerated the speed at which new ML algorithms are developed. In the fields associated with computational and quantum chemistry, the advances in ML combined with the increasing availability of data (powered by novel technologies such as GPU accelerated quantum chemistry[26] and high throughput experiments[27]) have brought about a profound impact and today can be considered nothing short of a "change of era". Treating molecules as virtual entities with different numeric properties, the framework was perfectly suited to embrace the Big Data revolution. Pioneering works include Behler-Parrinello atomistic potentials built with neural networks[28] and the first use of kernel methods and physics-based molecular representations to construct structure-property relationships (a.k.a. Quantum Machine Learning or QML),[29] which showed unprecedented accuracy for such negligible computational costs. These examples triggered a wave of ML models applied in computational chemistry (see Figure 1.1) that continues to grow exponentially. Indeed, the predictive power of supervised learning techniques has steadily improved over the last few years, allowing the modelling of chemical targets with increasing complexity: from simple scalar properties (*e.g.*, atomization and isomerization energies[29–31]) to vectors and tensorial quantities (*e.g.*, forces,[32–34] multipole moments,[35] polarizabilities[36,37]) and even potential energy surfaces,[28,38–40] excited-state properties,[41] electron densities,[8,42–46] and many-body wavefunctions.[47] Once trained, these models are orders of magnitude faster than traditional first-principle computations, which facilitate the exploration of otherwise unimaginably vast swathes of chemical space,[29,48–54] rapid access to complex chemical properties[8,42–47] and achieving statistically converged trajectories without sacrificing quantum chemical accuracy.[32,55–58] Similarly, unsupervised techniques have increasingly been used to rationalize the conformational space of molecules,[59–62] to aid in the design molecular representations,[63] to create drug discovery maps,[64] to classify molecules in chemical databases according to different properties,[65–67] and even for automatic molecular design.[68–71]

While the dichotomy between supervised/unsupervised methods has been useful to classify

ML algorithms depending on their nature and utility, the latest developments in computer science show that the two classes are often interlinked.[72–74] This is especially true for high-dimensional datasets, for which only a small unknown number of feature variables are relevant for a particular application. The inclusion of data variables that lack useful information for a specific target often degrade the performance of supervised or unsupervised algorithms, sometimes radically. On the one hand, the more descriptive feature variables are selected, the more training instances are necessary to maintain the density of training data in the feature space and to avoid the appearance of spurious relationships between unrelated variables. This aspect is particularly relevant for supervised algorithms as it leads to overfitting and therefore to poor generalization. On the other hand, uninformative features can contaminate the notion of similarity between training instances, which is fundamental to both supervised and unsupervised algorithms (*e.g.*, the outcome of clustering algorithms becomes meaningless if the data are grouped by uninformative features). As our information-gathering capacity increases, so does the dimensionality of the datasets. Constructing maps between variables and transforming data to interpretable low-dimensional representations has thus become part of the same problem. A deeper understanding of this relationship has brought several recent advances in computer science in the form of techniques that combine both supervised and unsupervised methods, such as metric and similarity learning techniques,[72] supervised manifold learning,[75] and supervised neural network-based encoders.[76] In the fields of computational and quantum chemistry, however, most of these techniques are still largely underused, with exceptions in the area of automated molecular design with supervised autoencoders.[70,76] This is crucial for "similarity-based" algorithms such as kernel regression that are heavily used by the quantum chemical community. Their construction, based on computing similarity measures,[21] allows the user to easily inject physical and chemical knowledge to improve the performance of a model. This improvement can be achieved by tailoring the selection of the reference training instances according to an expert-crafted criterion (external information about the characteristics of training data can be used to select the landmark reference training instances, which is key for the adequate performance of kernel models).[77,78] Still, filtering based only on the composition of the training data can be inefficient in situations where the selected training instances are not those that are tailored for the learning of the target property. (Δ-)[55]ML potentials represent a typical illustration of this issue, as the vast majority of chemical environments are well described by a baseline model while the error is concentrated in localized areas. In this thesis, we introduce an approach that leverages supervised dimensional reduction algorithm to eliminate the redundancy and identify the most relevant reference environments used to train kernel-based atomic potentials.

An alternative way to improve the regression task is to rely upon molecular representations built to encode the physical features of the chemical target.[29,53,77,79–82] These strategies improve the generality and performance of kernel-based models, especially when the number of acquirable training instances is limited.[30,31] Yet, the predefined importance of the features in molecular representations somewhat limits the adaptability of similarity-based models, notably when targeting complex properties related to phenomena involving more than a single

molecule.[10,83,84] Experimentally relevant properties that depend on a multitude of stationary points (*e.g.* catalytic properties) are a typical example. Rather than engineering new molecular representations for each target, we will propose a step-by-step strategy based on similarity learning that is adaptable to other applications and which filters the information in existing representations.

The main objective of this thesis is thus to leverage the capabilities of supervised/unsupervised ML algorithms interplay to improve the performance and adaptability of kernel-based models. Specifically, we aim at tailoring the selection of the two key ingredients at the basis of kernel regression (*i.e.*, reference pools and features of the representation) with applications in the field of computational organic chemistry. We propose two sets of tools based on (i) supervised dimensionality reduction to select the most relevant atomic environments and (ii) similarity learning to amplify the most relevant features of the representation. The performance and utility of the approaches are illustrated on two categories of applications involving the accurate and transferable sampling of free energy landscapes and the prediction of one of the most challenging catalytic properties, enantionmeric excess.

The material of the thesis is organized as follows.

An overview of the relevant theoretical background is presented in **Chapter 2**. We first introduce the fundamental elements of machine learning models and present the methodologies that are broadly used by the quantum chemical community. This includes strategies to construct molecular representations and ways to exploit them in supervised and unsupervised learning approaches. We then discuss the concept of similarity between elements in high-dimensional feature spaces and how this similarity measure can be improved. The last section summarizes the theoretical foundations of enhanced sampling methods and alternatives to perform free energy computations, with special focus placed on the Replica Exchange techniques at the center of this work.

Chapter 3 is a preliminary chapter that sets up the fundamental methodologies central to the developments presented in the thesis, including the Modular Replica Exchange Simulator (MORESIM). Kernel-based machine learning potentials are trained to accurately reproduce the free energy landscapes of highly flexible organic molecules. They are efficiently combined with the proposed Hamiltonian-reservoir Replica Exchange (Hres-RE) method, a novel enhanced sampling technique based on the combination of Hamiltonian Replica Exchange[85] and Reservoir Replica exchange[86] that are all integrated in our modular python implementation of replica exchange (*i.e.*, MORESIM). Hres-RE allows for an effective conformational sampling of molecular systems without requiring atomic forces. Specifically, Hres-RE generates "jumps" between free energy basins without crossing energy barriers, preventing the exploration of conformational regions outside the domain of applicability of the trained ML potential. The value of the approach is demonstrated through achieving CCSD(T)/CBS[87,88] free energy landscapes of flexible middle sized (40-50 atoms) organic molecules that would have otherwise been inaccessible.

Despite being robust and accurate, the kernel ridge model used in **Chapter 3** is not transferable to predict potential energy surfaces of molecules other than those represented in the training set. This limitation is addressed in **Chapter 4** with the introduction of a local kernel regression (LKR) approach that combines the scalability and transferability of neutral networks while preserving the benefits of kernel methods. The proposed LKR framework acts on the selection of reference atomic environments in a pool of highly redundant entries. While this filtering task is traditionally tackled by unsupervised learning algorithms based on dissimilarity in the input feature space, the later does not necessarily correlate with the dissimilarity in the latent (*i.e.*, target property) space. We address this issue by combining the LKR framework with Orthogonal Matching Pursuit[89] (OMP), a regression algorithm with supervised sparsity. This results in a supervised dimensionality reduction algorithm that selects the optimal reference atomic environments that optimize the prediction of the target property. The performance of LKR-OMP, trained on thermally sampled dipeptide conformers, is validated on the prediction of the potential energy surface of oligopeptides and compared with that of a state-of-the-art Behler-Parrinello neural network. The LKR-OMP shows equal or even superior performance to the NN model, but also comes with a unique set of analytical tools.

Chapter 5 focuses on the other necessary ingredient for the kernel regression, which are the features of the representation that define molecular similarity. This objective is motivated by the fact that the molecular representations commonly used in QML are overly complete, containing a large portion of irrelevant or redundant information. As a result, measures of molecular dissimilarity are contaminated and biased, which leads to a loss of connection with the dissimilarity in the target property. This effect is exacerbated if the property of interest depends on more than a single molecule. Here, we illustrate the problem by considering the ML prediction of the DFT-computed enantioselective excess of a Lewis base-catalysed propargylation reaction. We then introduce the concept of reaction-based representations and exploit metric learning and supervised feature selection techniques to filter the information contained in the molecular representations. This featurization dramatically improves the performance of the similarity-based machine learning model.

Finally, **Chapter 6** closes the thesis by summarizing the main conclusions and presenting possible future developments.

Several appendices contain additional materials supplementing this thesis:

Appendix A Contains a small summary of the most used supervised and unsupervised learning methods with advice on how and when they should be used.

Appendix B contains a description of the modular python package MORESIM, used to implement the Hres-RE simulations from chapters 3 and 4.

Appendix C contains a description of MolView, a Python script based on the library Dash that allows easily constructed interactive visualization web-apps to explore chemical data.

Appendix D contains the description of a set of Jupyter Notebooks developed for the summer school BDML4Chem with hands-on tutorials on how to practically use ML algorithms in a real world scenario.

Appendix E contains a set of scientific artwork developed with the open-source 3D modelling software Blender for several scientific publications.

2 Theory

This chapter lays out an overview of the theoretical background relevant to the material presented in this thesis.

The first part contains a description of the basic machine learning elements used in this thesis. Aiming at providing a basic introduction as well as source material for further reference, the chapter is structured like a manual and allows following the different steps of the ML pipelines used throughout the works presented. It begins with the different approaches to numerically represent molecular data are reported, with special emphasis on those developed by the quantum chemistry community. A separate subsection is dedicated to the basic elements of supervised and unsupervised statistical learning, including details and practical advice to consider at each step of the ML workflow. Given their prominent role in this thesis, supervised kernel-based regression techniques are then more extensively discussed, with alternative supervised and unsupervised algorithms being described in Appendix A. The final subsection highlights the importance of the metric and the notion of similarity in machine learning as well as its implication in approaches based on both supervised and unsupervised learning.

The second section introduces the fundamentals of canonical simulations and enhanced sampling techniques. Replica Exchange schemes are presented as one convenient and reliable tool for unbiased explorations of conformational landscapes, which are central to the applications presented in this thesis.

2.1 Machine learning from chemical data

Machine Learning (ML) refers to the family of computer algorithms that improve automatically through experience and by the use of data.[90] While the term "machine learning" was coined in 1959 by the pioneer in artificial intelligence Arthur Samuel working at IBM,[91] it was Tom M. Mitchell that later provided a formal definition of machine learning algorithms as: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with

experience E.". [91]

Machine learning algorithms have been traditionally classified into two broad main classes, supervised and unsupervised, based on the task expected from the program. Specifically, the goal of unsupervised learning is to find patterns in unlabelled data, while the goal of supervised learning is to build prediction models of target properties based on input data. Additionally, a third class of algorithms known as reinforced learning deals with the problem of finding a set of actions that maximize some notion of cumulative reward. [92] This last class of methods is outside the scope of this work.

2.1.1 Encoding Molecular information: Molecular representations

Applying ML to solve problems in computational chemistry requires a framework to encode molecular information into numerical data that computers can process, that is, a vectorial molecular representation. [93,94]

Building molecular representations in terms of predetermined numerical descriptors is a long-established practice in chemistry and materials informatics, mainly used in the past to construct (linear) Quantitative Structure to Property Relationships (QSPR). [95–99] Common examples of readily available molecular descriptors include the atomic composition, the electronegativity of the constituent atoms, and electronic structure properties such as the HOMO-LUMO gap of a molecule. The main disadvantage of this kind of fingerprints is that they usually require prior knowledge of the problem and their efficiency is generally case-specific. [100] For this reason, simple chemoinformatics descriptors are more often used to rationalize the behavior of specific classes of compounds, focusing on macroscopic properties such as solubility, pharmacological activity, or toxicity. [101–103]

In contrast to QSPR, Quantum Machine Learning (QML) has the objective to develop universal, physics-based representations that encode all the necessary information to characterize any chemical system. The common starting point of this type of representation is the electronic Schrödinger equation, as it governs the electronic wavefunction and thus all the quantum chemical properties. In this context, the large majority of QML representations encode the nuclear positions ($\mathbf{R_I}$) and charges (Z_I), since (assuming charge neutrality) these properties are sufficient information to fix the electronic Hamiltonian for any chemical system. [93,94,104–108] Besides fixing the Hamiltonian, nuclear position and charges are also readily available quantities to inject into the machine learning workflow.

While nuclear charges are simple scalar quantities that are rather straightforward to include in a numerical representation, this is not the case for nuclear positions, which are generally expressed in a set of Cartesian coordinates. For the algorithm to be able to learn, it is imperative for the representation to be covariant with the target property upon any arbitrary coordinate transformation. The majority of quantum chemical properties are scalars (*e.g.* electronic, conformational, atomization energies), which are invariant under the most fundamental

symmetries of physics: rigid rotation and translation of the molecules and permutation of their atoms. Therefore, most of the effort in the development of efficient QML representations is devoted to transform and encode the input nuclear coordinates in such a way to preserve all spatial symmetries.[93]

The first well-established example of physics-based representation is the Coulomb matrix (CM),[29] which is a molecular descriptor built to mimic the electrostatic interaction between nuclei. As these interactions depend on the distances between atom pairs and not on their absolute positions, rotational and translation invariance is guaranteed by construction. Since the Coulomb matrix represents each molecule as a whole, indivisible entity (*i.e.*, one molecule corresponds to one vector), CM is part of a more comprehensive family of descriptors called global representations. In contrast, local representations, often atom-centered, describe molecules as a collection of atoms and their environments. This category is well represented by the seminal example of the Behler-Parrinello symmetry functions,[28] which are fixed size vectors constructed using products of an atom-centered radial and angular basis set.[39,79] Local representations are scalable, as their computational cost increases linearly with the number of atoms, and transferable since a large chemical diversity can be described as a combination of a rather limited number of atom-centered environments.[8,109,110] Coulomb matrix and Behler-Parrinello symmetry functions were the first universally applicable, physical-based representations that promoted fast and accurate predictions of molecular properties. However, they have both stringent limitations. The Coulomb matrix is not invariant upon permutation of atoms in the molecule and electrostatic interactions are not always well correlated with molecular properties. The original Behler-Parrinello symmetry functions result in impracticably large vectors when the database contains more than a few different atom types.

During the last decade, the increased interest of the quantum chemical community in improving physics-based representations led to the development of a rather diverse choice of descriptors, each attempting to successively overcome the limitations of others. Overall, these representations can be classified into two distinct groups, according to the physics that they try to model:

- **Representations mimicking a potential**. A large group of representations relies on sets of one-, two-, three-, and N-body descriptors to mimic physical properties in the same way force fields use bonds, angles and dihedrals to model potential energy surfaces.[111–113] In this sense, ML models based on this type of representation can be seen as modern expression of classical force fields, where, however, the functional form is not defined *a priori* and the target property is not necessarily the ground-state electronic potential energy. A few examples of the most common representations falling in this first category are the already mentioned Coulomb matrices,[29] the bag of bonds,[80] permutation invariant polynomials,[114,115] sine and Ewald sum matrices,[116], many-body expansions[81,94] and many-body tensor representation,[117] descriptors with constant complexity,[118] histograms of distances, angles, or dihedrals (HDAD),[51] Bonds-Angles Machine Learning (BAML),[94] the Spectrum of London and Axilrod-Teller-Muto

(SLATM) [119] and FCHL. [82]

- **Representations mimicking a particle density**. Another family of representations branched from the Behler-Parrinelo (BP) symmetry functions, [28,39,79] use atom-centered functions to represent the density of neighboring nuclei within a pre-established cutoff distance. Examples in this category include the bispectrum, [93] partial radial distribution functions, [120] simple elemental descriptors, [121] Fourier series of atomic radial distribution functions, [122] weighted BP symmetry functions, [123] and the Smooth Overlap of Atomic Positions. [77,93,124]

Independently from the nature of the features used, molecular representations can be further classified based on other criteria, such as whether they represent a molecule globally as a whole entity such as the CM or locally as a collection of environments like representations based on symmetry functions. They can also be classified depending on which symmetries do they contain, whether the representation is vectorial or a tensor of higher order, and depending on their suitability for periodic and non-periodic systems. [125] Recently, they have also been classified and ranked based on how much information do they contain. [126] Using tools from information theory, it is possible to construct a hierarchy of the molecular representations that contain the most information, although it is not clear that this is correlated with their performance (as we will discuss later in this chapter). Even though QML representations can be classified into distinct groups according to their physics, suitability for condensed phase, [125] and even on their information content, [126] navigating over the totality of physics-based descriptors is a highly non-trivial task. Despite all the effort, it is still unclear if an optimal representation, which would lead to efficient learning for all the quantum chemical properties, exists. In principle, an excellent molecular representation would generate a feature space where the target properties are smooth and slowly varying, indicating that learned maps can be generalized to new data. In general, a representation must satisfy four distinct conditions to allow efficient and transferable learning of chemical properties:

- **Completeness** (injectivity) and **uniqueness**. There must be a one-to-one map between the representation of a molecule and its properties. Although in general QML representations are complete and unique for a large spectrum of chemical systems, it has been recently pointed out that many physics-based descriptors are unable to distinguish highly-symmetric (homometric) systems. [127–129]

- **Continuity** and **differentiability**. [93] A representation must be continuous and smooth, as the smoothness of target properties is also a basic underlying assumption of most statistical learning methods. [130,131]

- **Covariance**. The representation must encode the same symmetries as the target property, [36,93,132] including invariance to the basic symmetries of physics: rotation, reflection, translation, and permutation of atoms of the same species.

- **Size independence**. The size of the representation (*i.e.* the number of features) should be independent from the molecular size.[133]

Beyond these four fundamental properties, the more the representations reflect fundamental physical and chemical principles (*e.g.* nearsightedness of electronic matter, multi-scale nature of chemical interactions, similarity of chemical properties in the same periodic table group) the more robust, transferable and data-efficient the models become.[30,132,134] Even in light of the recent introduction of sophisticated deep-learned representations,[74] the role of physics-based descriptors remains paramount in quantum machine learning to the point that, for practical applications, the choice of representation is often more important than the choice of a learning algorithm.[53,135]

Among all QML representations, the Spectrum of London and Axilrod-Teller-Muto potential (SLATM) plays a particularly important role in the work presented in this thesis. For this reason, the mathematical form and the physical motivation of SLATM are detailed in Section 2.1.2.

2.1.2 The SLATM representation

The SLATM representation was introduced to overcome some of the limitations of representations like the Coulomb matrix: SLATM has fixed length independently of the molecular size, it is invariant by construction to atom permutations and include information about one-, two- and three-body terms. In contrast to Coulomb matrix and other more sophisticated representations such as the histograms of distances, angles, and dihedrals (HDAD),[51] and the Bonds-Angles Machine Learning (BAML),[94] SLATM mimics explicitly a specific part of the electronic Hamiltonian, the long-range correlation potential. It has been proposed that incorporating the bond and angular information through a potential leads in general to faster and more efficient learning of quantum chemical properties.[30,119]

The conceptual starting point of SLATM is an expansion in many-body terms of artificially defined, atom-centered nuclear charge densities. This expansion takes the following mathematical form:

- One body contribution: the atomic number Z_I, for each atom I in the molecule.

$$\mathrm{SLATM}^1_I = Z_I \tag{2.1}$$

- Two-body contribution: London potential of the nuclear charge density.

$$\mathrm{SLATM}^2_I = \frac{1}{2}\sum_{J\neq I} Z_J \frac{1}{\sigma\sqrt{2\pi}} e^{-(\mathbf{r}-\mathbf{R}_{IJ})^2} \frac{1}{\mathbf{r}^6} \tag{2.2}$$

  The London potential has a much faster asymptotic decay than the electrostatic interac-

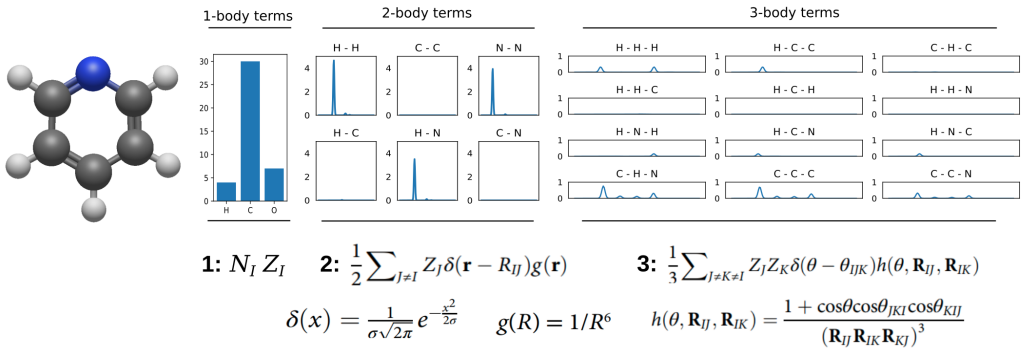


Figure 2.1 – Different (1, 2 and 3-body) terms in the (global) SLATM representation.

tions used in *e.g.* Coulomb matrix and thus is more appropriate to describe the covalent bond regime.[94]

- Three-body: Axilrod-Teller-Muto potential of the nuclear charge density.

$$\text{SLATM}_I^3 = \frac{1}{3}\sum_{J\neq K\neq I} Z_J Z_K \frac{1}{\sigma\sqrt{2\pi}} e^{-(\theta-\theta_{IJK})^2} h(\theta, \mathbf{R}_{IJ}, \mathbf{R}_{IK}), \tag{2.3}$$

where θ is a continuous variable the spans the angle between the vectors $\mathbf{R}_{IJ}$ and $\mathbf{R}_{IK}$. The function $h(\theta, \mathbf{R}_{IJ}, \mathbf{R}_{IK})$ is the core of the three-body contribution and depends on both on pairwise distances and angles as:

$$h(\theta, \mathbf{R}_{IJ}, \mathbf{R}_{IK}) = \frac{1 + cos\theta cos\theta_{JKI} cos\theta_{KIJ}}{(R_{IJ}R_{IK}R_{KJ})^3}. \tag{2.4}$$

The complete form of SLATM results from the concatenation of the one-, two- and three-body terms in a single vector (see Figure 2.1). Explicitly depending on the atom-centered nuclear charge densities, SLATM is originally a local representation (better specified as aSLATM, or atomic-SLATM), but the global version is readily obtained upon summation over all the atoms in a molecule.

There are three user-defined parameters needed to construct the SLATM representation: the cutoff radius r_c for the 2-body term, the width of smearing Gaussian function for both radial and angular terms, and the density of grid to sample the radial and angular terms. Since the London potential has a rapid asymptotic decay, most SLATM applications use a standard cutoff radius of 4.8 Å, for which the potential is usually well-converged. Following the original publication,[119] the width of the Gaussian functions are set to 0.05 Å and 0.05 rad, and the sample grid densities are set to 0.03 Å and 0.03 rad. These values were found to optimize the performance of SLATM across multiple datasets.[119]

Since its first introduction in 2017, the SLATM representation has been extensively used in the quantum machine learning community to predict a wide variety of chemical properties

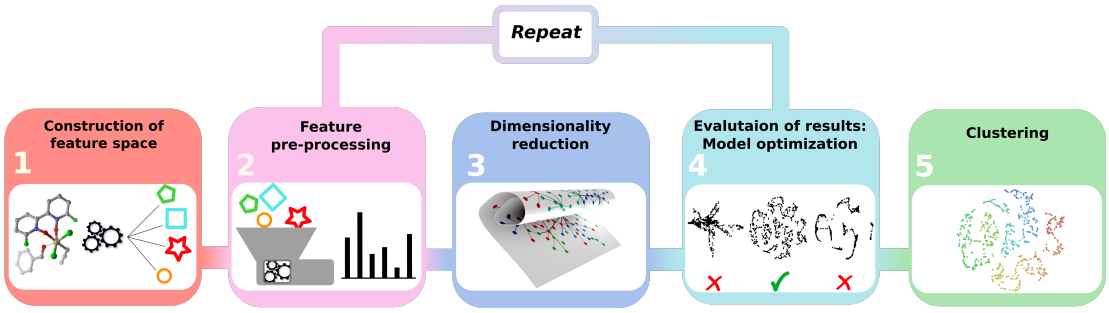


Figure 2.2 – Fundamental steps to analyze data with unsupervsied learning.

ranging from simple atomization energies,[119,136] to HOMO-LUMO energies and gaps, dipole moments, polarizabilities, zero-point vibrational energies, heat capacities, and vibrational frequencies.[82] However, SLATM should not be used to simultaneously represent systems that are not fully described by their atomic coordinates, for example transition metals with different magnetization or identical molecules with different charges. In such cases, alternative representations that are able to encode this information should be used instead.

On account of its robustness and its widespread success, SLATM has been the molecular representation of choice for all the scientific work presented in this thesis.

The previous sections discuss how to encode molecular information into numerical data that computers can process. In the following, we describe in detail the different steps of supervised and unsupervised machine learning pipelines.

2.1.3 Unsupervised Machine Learning Pipeline

Unsupervised learning is a comprehensive term that refers to a broad class of algorithms, which allows inferring the structure and the patterns present in unlabelled data. More precisely, unsupervised learning algorithms can be broadly divided into two classes according to their goal: dimensionality reduction and clustering.

The goal of nonlinear dimensionality reduction techniques is to transform data from a high-dimensional into a low-dimensional space while retaining to the greatest extent the same data structure as in the original dataset. These machine learning models are often employed to visualize and understand the intrinsic structure of the data and to improve the performance of other algorithms (*e.g.* clustering), which break down when applied on high-dimensional data. The whole concept of dimensionality reduction relies upon the hypothesis that some features in the database are strongly correlated to each and thus the intrinsic dimensionality of the data is in reality much smaller than the one fed by the user. If on the contrary, for a given dataset, all the features are independent and equally important, dimensionality reduction techniques will only result in a generalized loss of information.

In contrast to dimensionality reduction, the goal of clustering algorithms is to group data so that elements belonging to the same group are more “similar" to each other than to those in other groups. Each different clustering algorithm is distinguishable in the way it defines “similarity" and a “cluster".

While unsupervised learning techniques are all conceptually different, their application on a database follow in general a common 5-points procedure:

1. **Input feature space construction**

   Data may be fed by the user either as a set of predefined features (molecular size, distances and angles between atoms, type of atomic species, predefined molecular representations, *etc.*) or as a set of general instances, like molecular compounds, whose features have to be defined. The construction of the feature space is quintessential for unsupervised learning. As the data are unlabelled, the algorithm cannot determine *a priori* which features would structure the data in a way that is relevant for the user or any other post-processing task.

2. **Feature pre-processing**

   In general, similarity measures, like the widely used Euclidean distance, attribute higher importance to features that vary the most through the database. If no information on the relative importance of features is available, the most common approach is to normalize each feature so that its values fit a normal distribution centered at zero and with a standard deviation equal to one. This effectively enforces similarity measures to treat all features on equal footing. Periodic features, such as angles, can be replaced by other variables, such as their sine and cosine, which will help non-periodic metrics (*e.g.* Euclidean distances) to encode adequately periodicity.

3. **Choosing the dimensionality reduction algorithm**

   It is always possible to choose among many algorithms for dimensionality reduction, each of them based on different underlying principles to build the reduced feature space. If there is no fundamental reason for choosing a specific algorithm, it is good practice to use several and compare their outcome to the desired result (see appendix A for a summary of common dimensionality reduction algorithms and tips on how to use them).

   Dimensionality reduction algorithms can be subdivided into linear methods, often referred to as matrix factorization or matrix decomposition techniques, and nonlinear methods referred to as manifold learning. The archetype of linear dimensionality reduction is Principal Component Analysis (PCA), which consists in forming linear combinations of the original feature vectors to construct an orthogonal basis onto which the data can be projected. The final dimensionality is chosen by the user and it is determined by the number of basis vectors constructed. Within PCA, these orthogonal

vectors represent the directions of maximum variance in the original high-dimensional space.

Nonlinear dimensionality reduction can be thought of as a generalization of linear frameworks to capture local structures of data.[137] Unlike linear methods, the obtained dimensions bear no pre-defined (physical) meaning, except for indicating a “distance" or a (dis)similarity between two data points. Nevertheless, they are generally much more capable to elucidate the structure and distribution of data in a high-dimensional space. Perhaps the most popular method for manifold learning is t-distributed Stochastic Neighbours Embedding[138] (t-SNE), which relies on the idea that the probability for two points to be neighbors should be conserved upon projection to low-dimensional spaces(see Figure 2.3 for a comparison of PCA and t-SNE applied in a chemical database).

4. **Model optimization**

   Most machine learning algorithms are tuned by a set of user-defined constants, the hyper-parameters, that can significantly alter their outcome and must be optimized. However, in the context of unsupervised learning, the evaluation of the quality of a projection is often subjective and it is rare to have quantitative criteria to guide the optimization of hyper-parameters. Nevertheless, the fitness of an unsupervised learning algorithm can be always evaluated using somewhat heuristic metrics such as the fact that spread data points are generally preferred, or that a smooth transition between clusters, if there are any, is generally preferred over large gaps. Another possibility to assess the quality of the projection consists in verifying that the feature space variables vary smoothly across the projected map. To further optimize the obtained projection, steps 2-4 can be repeated while using different pre-processing techniques and dimensionality reduction algorithms. If features have been designed by the user, it is good practice to include step 1 in the loop.

5. **Clustering**

   Dimensionality reduction is often used in tandem with clustering algorithms to better highlight the similarity between groups of data (see Figure 2.3 d). As for dimensionality reduction, a plethora of clustering algorithms have been developed in the past, each of which can be classified according to their definition of “similarity" and “cluster" (see appendix A for a summary of common clustering algorithms and tips on how to use them). Alternatively, clustering can be applied directly to the input feature space, but the outcome is seldom comparable or better than clustering after dimensionality reduction (see Figure 2.3 c and d). Similar to dimensionality reduction, the quality of clustering depends largely on the subjective judgment of the user. Often the evaluation of clustering results is as difficult as the clustering itself.[139] While there exist some mathematical tools to analyze the quality of a clustering result,[140–142] there is no quantitative approach that can ultimately replace human evaluation.[143]

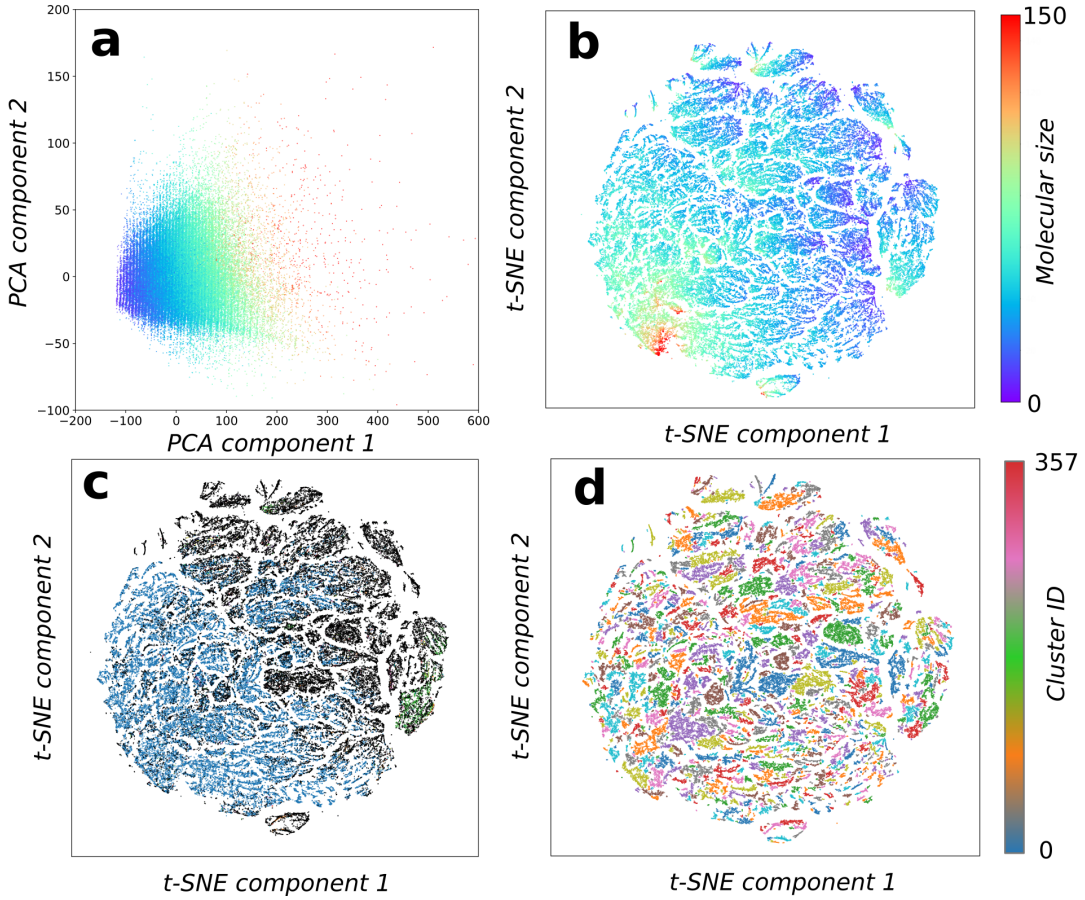


Figure 2.3 – Results of 2D dimensionality reduction using PCA (a) and t-SNE (b) for a subsample of 130.000 compouds of the Cambridge Crystallographic Data Centre (CCDC) database.[1] Each point represents a molecule, and the color code represents its size. SLATM was used as the input representation for both algorithms. As the overall variance of the SLATM representation is dominated by the size of the molecule, the components of PCA basically capture this magnitude and not much more. Alternatively, t-SNE is able to capture the local structure and different cluster in the data, which give a projection much richer in details. On the lower part of the figure the t-SNE projection is color coded using the cluster labels obtained applying DBSCAN[2] on the t-SNE coordinates (d) or directly on the SLATM representation (c). We used DBSCAN as it does not require as input a specific number of clusters (see Appendix A for more information on DBSCAN). This shows why is important to apply dimensionality reduction prior to clustering. DBSCAN applied directly to the input feature space groups most points in a single cluster and considers the rest as noise (black).

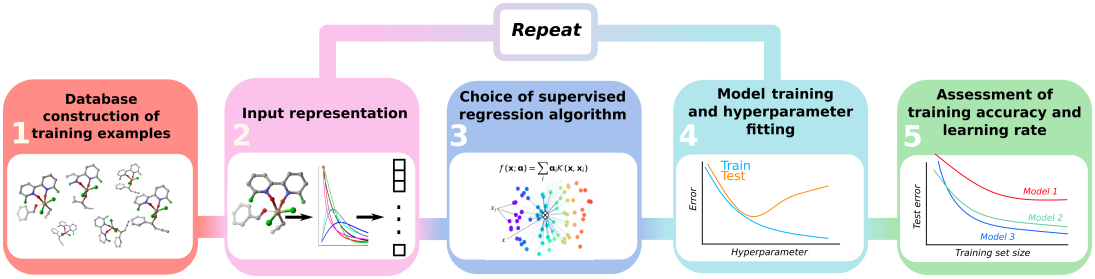


Figure 2.4 – Fundamental steps in building a supervised ML model.

2.1.4 Supervised Machine Learning Pipeline

In contrast to unsupervised learning, discussed in the previous section, the goal of supervised learning algorithms is to find a mathematical relationship between input data and their outputs (typically scalars or labels). Ideally, an ML algorithm is able to generalize the map from the training data and use this mapping to predict the output of "unseen" instances with controlled accuracy. The statistical performance of an algorithm is measured through this generalization error, *i.e.* the error computed on a test set different from the training. Similarly to unsupervised learning, building a supervised model generally consists of 5 steps (see Figure 2.4):

1. **Database construction of training examples**

   The first step in building any ML model is gathering data, which can be the most computationally expensive step. In quantum chemistry, acquiring data typically involves building a pool of molecular compounds and computing the desired target molecular quantity for each compound. While there is a large number of existing quantum chemical datasets, chemistry is so heterogeneous [144,145] that they are generally not relevant outside the framework they were built for, and very often each application requires the construction of a task-specific dataset.

   When constructing a database, it is extremely important to select the samples to avoid redundancy, data imbalance, and to achieve uniform sampling of the ensemble of interest. Intelligent database construction generally involves a step of unsupervised learning to understand the characteristics of the pool of training instance candidates. For instance, a very common tool to perform uniform sampling is Farthest Point Sampling (FPS), a simple greedy algorithm that selects instances from a database so that they are maximally separated from each other.

2. **Input representation**

   Choosing the most adequate molecular representation for the specific learning exercise is not always straightforward, but it is essential for the accuracy and generality of the trained model. If there is no stringent physical reason for which a specific representation should be chosen, steps 2 to 5 in Figure 2.4 should be repeated for different representations and the user should select the best-performing descriptor.

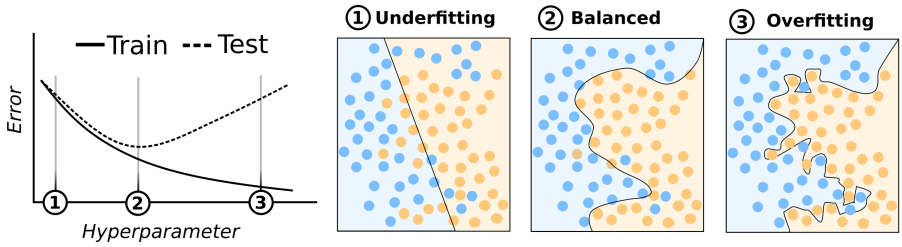


Figure 2.5 – Schematic representation of different bias/variance trade-off scenarios.

3. **Supervised regression algorithm**:

   The most widely used Supervised ML algorithms can be broadly classified into five groups (see Appendix A): linear models, similarity or kernel-based methods, neural networks, decision trees and ensemble methods, each with different strengths and weaknesses. There is no single best algorithm, and their performance can vary from case to case. Given the central role that kernel-based methods, and in particular, Kernel Ridge Regression (KRR) plays in the context of this thesis, we dedicate the following section to introduce this method in more detail.

4. **Model training and hyper-parameter fitting**:

   As any machine learning model, supervised algorithms also depend on a set of hyper-parameters that broadly define their complexity. Ideally, a model would learn the regularities in training data while adequately generalize them to unseen data.[21] Unfortunately, it is typically impossible to do both simultaneously, and in the optimal case there should be an equilibrium between both, the so-called bias-variance trade-off (see Figure 2.5).

5. **Assessment of training accuracy and learning rate**:

   In order to select the optimal model for a learning task, it is crucial to compare several machine learning frameworks trained on the same dataset, but using different representations and algorithms. This comparison is crucial to assess if any of the generated models have been able to exploit all the information in the training data to construct the mapping. The most straightforward approach to compare different models is to plot their learning curves (*i.e.* the generalization error vs. the number of training data instances). However, an important caveat on this practice is that the learning curve only reflects the global performance of a model. For this reason, it is good practice to use additional metrics and performance tests on subgroups of the data in order to understand the real behavior of any ML model.

2.1.5 Similarity-based regression: Kernel methods

The core assumption behind similarity-based supervised learning methods is that similar data instances (*i.e.* compounds in chemistry) are characterized by similar target properties. The prototypical algorithm for this class of models is the K-Nearest-Neighbours (KNN), which predicts new labels or scalar targets by averaging the labels and values of the K (a user-defined number) most similar points in the training data. Due to its simplicity, KNN is a very robust, lightweight, and easy-to-understand model, which performs surprisingly well in many situations.

The second paradigm of similarity-based regression models is represented by Kernel methods. Kernel methods aim at generating a mapping $f : \boldsymbol{x} \rightarrow y$ and they construct this map by evaluating the similarity between training instances $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N\}$ with known targets $\boldsymbol{y} = \{y_1, y_2, ..., y_N\}$. The concept of similarity is intimately related to the concept of distances between data, which require the evaluation of inner products in high-dimensional spaces.[21]

This operation allows constructing complex nonlinear mapping without explicitly applying a transformation on the coordinates of $\boldsymbol{x}$. More specifically, the expression of a kernel mapping has the form of a weighted sum over the different elements $\boldsymbol{x}_i$ in the training data:

$$f(x) = \sum_i \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i) = \boldsymbol{\alpha}^T \boldsymbol{k}(x), \tag{2.5}$$

where $k(\boldsymbol{x}, \boldsymbol{x}_i)$ is the kernel similarity between the two inputs, and is generally bounded between 0 (not similar) and 1 (similar). Most commonly the target quantity is a scalar value, although generalizations of kernel methods exist to generate multi-output models.[146,147]

Probably the most common kernel used in quantum machine learning is a simple Gaussian function of a distance metric, *i.e.* $k(\boldsymbol{x}, \boldsymbol{x}_i) = exp(-\gamma d(\boldsymbol{x}, \boldsymbol{x}_i)$, with d typically being the Euclidean distance. This combination produces smooth and local solutions, but other alternatives such as the dot product between features vectors ($k(x, x_i) = x \cdot x_i$) also exist.

In quantum machine learning, most of the learning tasks are regressions, as most of the targets are continuous numerical values. Among all possible kernel-based algorithms, Kernel Ridge Regression (KRR) is probably the most widely used in quantum chemistry and it is generally the method used throughout this thesis.

In KRR the coefficients α of Equation 2.5 are computed using an ordinary least squares minimization, *i.e.* minimizing a cost function $C(\boldsymbol{\alpha})$ equal the sum of the squared differences between mapped (predicted) $\hat{y}_i = f(\boldsymbol{x}_i)$ and the real y_i in the training data: $C(\boldsymbol{\alpha}) = \sum_j (y_j - f(\boldsymbol{x}_j))^2 = \sum_j (y_j - \boldsymbol{\alpha}^T \boldsymbol{k}(\boldsymbol{x}_j))^2$. As many regression problems, simple kernel regression would suffer from collinearity among the data, especially when extremely similar training instances are present. To overcome this problem, KRR uses a regularization term (a 'Ridge') $\lambda ||\boldsymbol{\alpha}||_2^2 =$

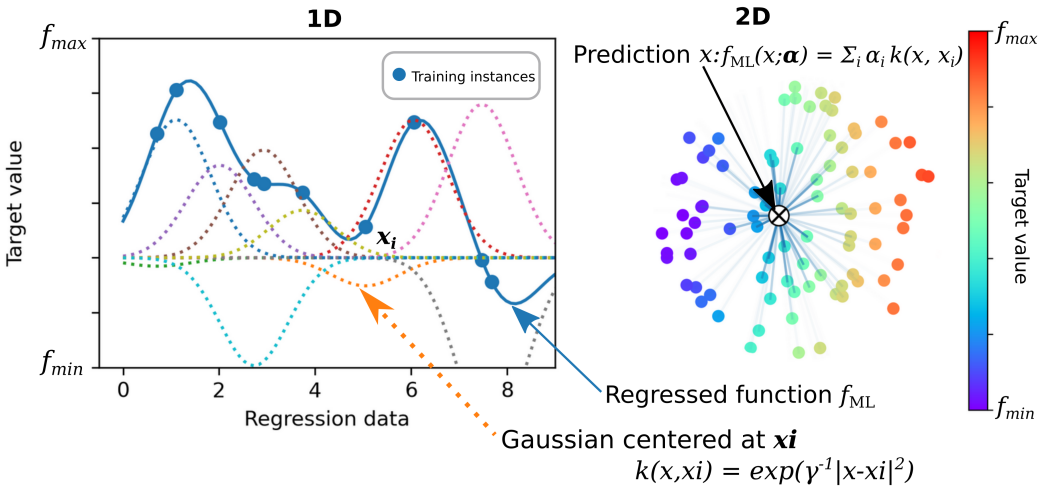


Figure 2.6 – Depiction of a kernel method to predict a scalar quantity using 1D (left) and 2D(left) input data. On the left, the dotted lines are gaussians of the same with centered on the training instances. The scale of the gaussians is determined in the training step of the model with the goal of constructing a smooth curve crossing all the training points. The prediction of a new data entry (the center x in the right, for example) is generated by adding the contributions to nearby points.

$\lambda \sum_i \alpha_i^2$, which adds a penalty to the magnitude of the $\boldsymbol{\alpha}$ coefficients:

$$C(\boldsymbol{\alpha}) = \sum_j (y_j - \boldsymbol{\alpha}^T \boldsymbol{k}(\boldsymbol{x}_j))^2 + \lambda \alpha_j^2. \tag{2.6}$$

Optimal coefficients ($\boldsymbol{\alpha}^*$) are found by finding the roots of the fist derivative if $C(\boldsymbol{\alpha})$, which leads to:

$$\boldsymbol{\alpha}^* = (\mathrm{K} + \lambda \mathrm{I})^{-1} \boldsymbol{y}, \tag{2.7}$$

where $\mathrm{K}_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

2.1.6 Supervised feature selection, similarity measures, and metric learning

The previous sections focused on the steps to build effective supervised and unsupervised machine learning pipelines but omitted one capital aspect of QML: how can we adapt, modify or recast a molecular representation to improve the overall learning. For instance, features coming from different molecular representations could be concatenated to obtain a more complete representation or even to augment existing representations with additional descriptive features.

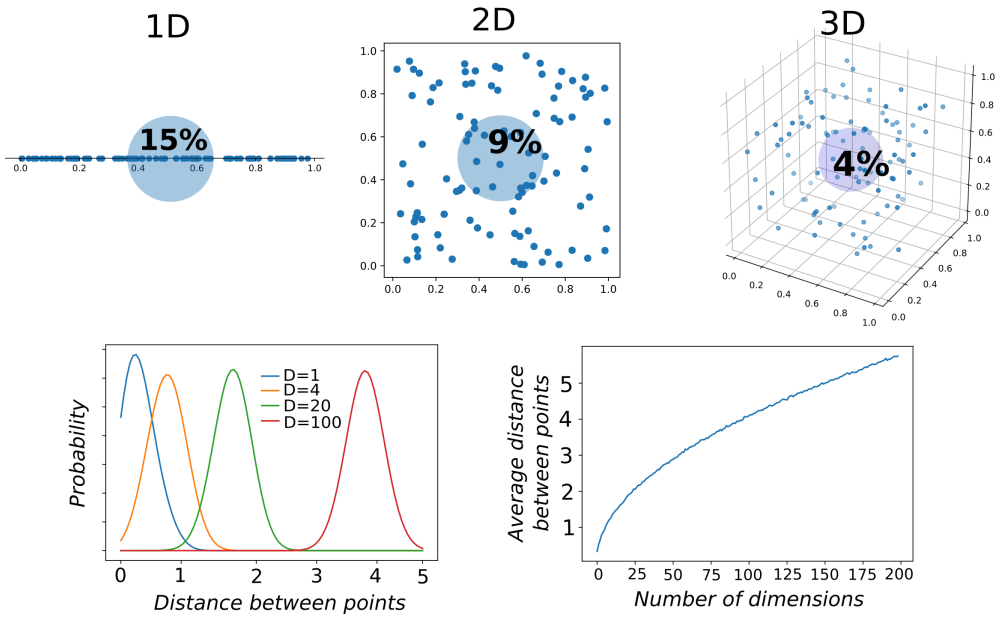


Figure 2.7 – For a given set of data, the higher the dimension of the feature space the more sparsely distributed the data points are. This is exemplified in this figure, where it can be seen how the density of data diminished with the increase of dimension. With increasing dimensionality, a sphere of the same radius contains less and less data points. At the same time, the separation between points increases.

The curse of dimensionality

From the perspective of statistical analysis and ML, the optimal dimensionality of the feature space should be as small as possible, but still large enough to contain all the relevant information for the learning task. The addition of superfluous features is generally detrimental for any statistical model, as they could add significant noise to the data. Moreover, the feature space increases exponentially while adding independent descriptors, and data points distributed in that space become quickly sparse. The sparsification of data in high-dimension leads to a phenomenon called the "curse of dimensionality":[21] the higher the dimensionality of the feature space, the exponentially more examples (*i.e.* data) are needed to sample that space and thus obtain statistically reliable learning, which significantly increases the computational cost of building a statistical model (see Figure 2.7).

Feature selection and extraction

Given the curse of dimensionality, the information contained in the data has to be filtered to determine what is relevant for the learning exercise and what is not. Feature selection techniques address this issue.[148] The underlying assumption of this class of algorithms is that data contains features that are either redundant or irrelevant to the specific learning task, and therefore can be safely eliminated. For practical purposes, however, redundant and irrelevant

are well distinct. Redundancy implies that the information carried by a feature is already encoded into another. Irrelevance means that the specific feature does not carry meaningful information for a specific application. To overcome redundancy is the goal of unsupervised feature selection, while to remove irrelevance is the goal of supervised algorithms. [149]

Unsupervised feature selection methods are based on two basic principles: removing similar features and removing features with low variance. A common way of eliminating similar features is to compute the cross-correlation between all the features and then remove highly correlated ones. Removing low variance features is even more straightforward and can be performed using simple algorithms such as PCA or CUR decomposition. [150]

In contrast, supervised feature selection (or supervised sparsity) can be divided into three main categories, according to the way features are selected. Filter methods use rather cheap statistical tools (*e.g.* correlation coefficients, statistical ranks, mutual information, *etc.*) to filter out irrelevant or redundant features prior to any learning. [151] In contrast, wrapper methods take the performance of the model into account and search greedily among sub-sets of features to identify which combination of descriptors leads to the best overall learning. Finally, embedded methods perform the feature selection directly during the training of the model. Particularly important for this thesis, Orthogonal Matching Pursuit (OMP) [89,152] is a greedy embedded method, which selects progressively the set of features that are the most correlated with the desired target.

Metric Learning

As stated in previous sections, the most common machine learning algorithms used in quantum chemistry are all rooted in the notion of a distance or a similarity between data points, which depends on the features of the representation and the metric employed. Most commonly, pre-defined metrics such as the Euclidean and the Manhattan distances are used. However, metrics like the Euclidean distance ($d_E(\mathbf{a},\mathbf{b}) = \sqrt{\sum_i (a_i - b_i)^2} = ||\mathbf{a}-\mathbf{b}||_2$) are not invariant under monotonic transformations and are sensitive to the scaling of individual coordinates. For instance, the use of the Euclidean metric inflates the importance of the features with higher variance. To counteract this effect, features are often normalized. However, this kind of unsupervised generalized rescaling is not necessarily optimal, as depending on the specific target certain features may be more important than others, and it has been shown that it can deteriorate the learning. [153] To illustrate this point, Figure 2.8 shows an example of *ad-hoc* data, where unsupervised standardization of the individual features worsen the distribution of data in the feature space. The Euclidean metric in the example of Figure 2.8 is simply not adequate, as it treats both features on equal footing, given that they have the same variance. Nonetheless, it appears clearly that feature 1 is more important, as the target function changes faster in that direction than in the direction of feature 2. In a space where feature 2 is shrunk (or alternatively using a metric that gives more importance to feature 1) the target property evolves equally fast in each direction and results in increased prediction accuracy (see Figure 2.8, top-right). However, it is not always straightforward to understand if the

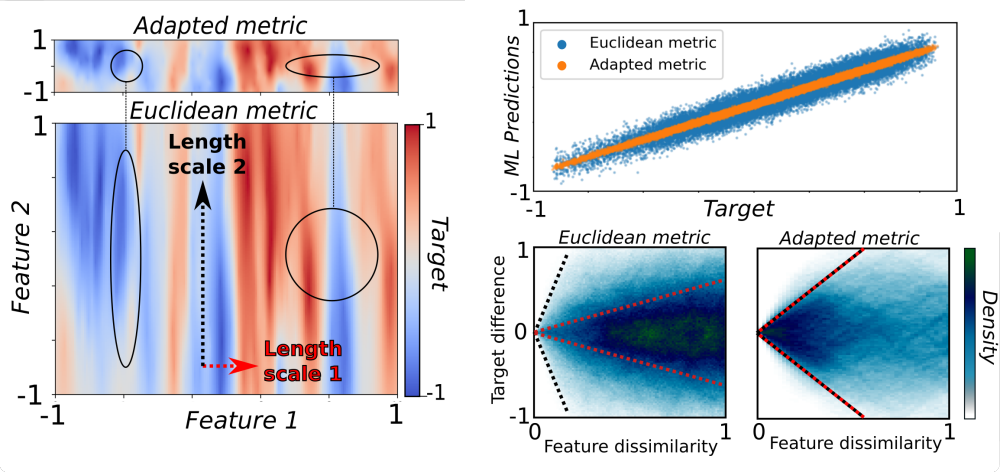


Figure 2.8 – On the left we can see the original and adapted feature spaces with the target function. On the top right we can see the accuracy of test predictions done with 100 training points. The figures on the lower left show the dissimilarity plots for each of the two metrics / feature spaces.

Euclidean metric is adequate for a feature space, especially if the space is highly dimensional. Dissimilarity plots are a tool that can help to visualize the influence of metrics on the learning. These plots correlate the distance between points in the feature space with the difference in target property. If the metric used is adequate, the difference in target property should go to zero as the distance between points goes to zero in the feature space. In this example, the Euclidean metric in the original spaces shows this behavior, although it can be seen that there are two different asymptotic directions, corresponding to the two length scales of the two features (see Figure 2.8, bottom-right). This spurious behavior is corrected using the adapted metric.

The choice of metric is essential for any machine learning task, but it can be extremely difficult to find or design a metric well-suited for the data and learning task of interest.[154] Distance metric learning[153,155] (or metric or similarity learning) aims at automatically predicting task-specific distance metrics from supervised data so that the learned distance metric can then be used to perform additional supervised and unsupervised tasks. As most ML techniques, metric learning algorithms can be classified in linear and non-linear metric learning.

Linear metric learning methods learn a linear transformation $\mathbf{L}$ of the original feature space so that the Euclidean metric can be a good measure of dissimilarity. The Euclidean distance between points $\mathbf{a}$ and $\mathbf{b}$ in the transformed space is $d_E(\mathbf{a}', \mathbf{b}')$:

$$d_E(\mathbf{a}', \mathbf{b}') = ||\mathbf{a}' - \mathbf{b}'||_2 = ||\mathbf{L}\mathbf{a} - \mathbf{L}\mathbf{b}||_2 = ||\mathbf{L}(\mathbf{a} - \mathbf{b})||_2 = d_M(\mathbf{a}, \mathbf{b}), \tag{2.8}$$

where d_M is generally known as the Mahalanobis distance. Linear metric learning methods

optimize the matrix **L** by minimizing the error performed by classification or regression models. The widely used algorithm Large Margin Nearest Neighbor[156] (LMNN) learns a Mahalanobis distance to optimize a K-Nearest-Neighbours (KNN) classification, while the algorithm Metric Learning for Kernel Regression[72] (MLKR) learns a Mahalanobis distance to optimize a kernel regression. Constraints can be added to the optimization procedure to promote certain properties to the learned metric, such as sparsity.[157,158] In that case, the matrix **L** will contain few non-zero elements, effectively behaving as a linear supervised dimensionality reduction method at the same time.

Alternatively, non-linear metric learning methods are a class of techniques that learn variable metrics that change depending on the local environment of points. The first nonlinear metric algorithms were based on learning variable Mahalanobis metrics,[155,159] but more recent methods are based on deep learning (deep metric learning[160]). Most deep metric learning models are inspired by Siamese[161] and Triplet[162] networks. They have been mostly used for image recognition tasks,[160] but also for drug response similarity prediction[163] and sequence-embedding of DNA.[164] Deep metric learning allows better generalization and reliability of deep learning models, as well as better performance for unbalanced classes. However, the higher complexity of non-linear metric learning generally means that they require much more data than linear metric learning methods to avoid overfitting.

Metric learning approaches form a bridge that connects the tasks typically associated with supervised and unsupervised learning algorithms based on similarity. From a technical perspective, metric learning models are supervised, as the data points are labeled. From a task-oriented perspective, filtering and adapting information to define similarity are typical goals of unsupervised learning. Indeed, filtering information, constructing effective representations, and predicting a target property are all related tasks.

2.2 Conformational sampling and Replica Exchange methods

One of the key technical aspects of this thesis is accelerating free energy computations by combining ML-based potentials with enhanced sampling methods. For this reason, the following sections introduce the main concepts and the formalism of conformational sampling and enhanced sampling techniques, with special emphasis on replica exchange simulations.

In statistics, importance sampling is a general framework for estimating the properties of a particular probability distribution. In computational chemistry, conformational sampling refers to the importance sampling of the degrees of freedom of molecular systems to estimate the probability distributions of their conformers. This probability distribution, estimated at thermodynamic equilibrium, is the basis of free energy difference computations.

2.2.1 Canonical Sampling

The conformational sampling presented in this work aims exclusively at computing free energy differences in the canonical, or NVT, ensemble (*i.e.* at a constant number of particles (N), constant volume (V), and constant temperature (T)). The principal thermodynamic variable of the canonical ensemble, determining the probability distribution of states, is the temperature.[165]

In these conditions, the probability of a state S depends on its energy $E(S)$, in the form of the Boltzman distribution:[165]

$$p(S) \propto e^{-\beta E(S)}, \text{where } \beta = \frac{1}{K_b T}. \tag{2.9}$$

Any property average $< A >$ of the distribution can be theoretically obtained by evaluating the probability (*i.e.* the energy) of all the possible states in the space S:

$$< A >= \int_{s \in S} A(s) p(s) ds. \tag{2.10}$$

In practice, the integral in Equation 2.10 cannot be evaluated exactly and even its approximation by strictly random sampling is not efficient as most conformations would have very high energy, and therefore negligible weight in the ensemble average. In importance sampling, the goal is to focus on the conformational space where the weight is significant in the average. More precisely, the goal is to generate a sequence of samples in such a way that the more samples are produced, the more closely their distribution approximates the desired distribution $p(S)$. This is the key idea behind the Metropolis–Hastings Monte Carlo (MC) algorithm.

In MC, samples are produced probabilistically and iteratively in such a way that the distribution of consecutive conformational samples is dependent only on the current state (which

effectively behaves as a Markov chain).[166] At each iteration, an MC algorithm picks a possible move from a set of candidates based on the current state. Then, with some probability, the generated MC step is accepted (in which case the current state is updated) or rejected (in which case the move is reverted and the same initial state is used in the next iteration). The probability of acceptance is determined by comparing probabilities $p(S)$ of the current and candidate samples.

To ensure that the sampling of an MC simulation will converge to the canonical probability distribution, there are two conditions. First, the methods used to generate new candidates have to be ergodic, which means that all the conformational space is accessible through them. Second, they must satisfy the principle of detailed balance. Detailed balance states that at equilibrium, each elementary process is in equilibrium with its reverse process. In practice this means that the probability of being in state A ($p(A)$) and going to state B ($p(A \rightarrow B|A)$) must be the same as being in state B ($p(B)$) and moving to state A ($p(B \rightarrow A|B)$), *i.e.*:

$$p(A)p(A \longrightarrow B|A) = p(B)p(B \longrightarrow A|B). \tag{2.11}$$

By using equation 2.10, this leaves:

$$\frac{p(A \longrightarrow B|A)}{p(B \longrightarrow A|B)} = e^{\beta(E(B)-E(A)}. \tag{2.12}$$

Assuming that both $A \rightarrow B$ and $B \rightarrow A$ exists, then the simplest expression that satisfies this equation is:

$$p(A \longrightarrow B|A) = min(1, e^{\beta(E(B)-E(A)}), \tag{2.13}$$

the last formula being the original Metropolis scheme for the acceptance probability of an MC move in a simulation.

It has been shown that in some circumstances some of the conditions for an acceptable MC move can be relaxed and require “balance", rather than “detailed balance",[167] to achieve the right convergence, but the reader is referred elsewhere[165–167] for an in-depth discussion of the topic.

Alternatively to MC, canonical sampling of atomic conformations can be achieved through dynamical simulations. In the limit of time going to infinit, time averages and ensemble averages are equivalent, and can be generated with Molecular Dynamics (MD) simulations when coupled with a virtual temperature bath (a.k.a. a thermostat[165]).

For the purpose of conformational sampling, the differences between MC and MD reside in their efficiency. For molecular systems, the “interesting region” of a canonical distribution

(with high $p(S)$) is narrowly distributed in the accessible phase space, since most of the possible arrangements of atoms in space do not represent a realistic molecule. As a result, the efficiency of MC methods depends on the design of the moves used to drive the evolution of the sampling. An effective MC sampling must contain "moves" designed to guide the simulation in a large dimensional and convoluted phase space filled with steep potential barriers. Otherwise, a blind search in the form of random displacements is typically very inefficient, resulting in very small probabilities of move acceptance. Alternatively, in MD simulations the dynamic evolution of a system is reproduced and all the structures generated directly fall in the relevant area of the phase space, so every step is accepted by default. Therefore, for molecular conformational sampling, MD is often superior to crude MC. Nevertheless, MD trajectories are bound to diffuse locally, with a time step smaller than the time scale of any of the system's dynamic modes, in order to produce stable and realistic dynamics. In contrast, in MC simulations any rearrangement of the system's coordinates can be used as a candidate move, as long as it satisfies detailed balance. This allows for engineered MC moves designed to generate big changes of the system's coordinates and produce jumps in the phase space. Therefore, while MD has a much faster local diffusion in the phase space, MC can potentially have a much faster global diffusion.

The sampling efficiency is especially relevant for systems with very slow collective motions, for example, related to the transition between basins in rough potential energy landscapes. In such cases, standard MD simulations get trapped in local energy basins, and the time scales associated with barriers-crossing can be many times the time scales that are affordable/feasible to simulate. Alternatively, MC simulations can allow for easy transition between meta-stable states if the MC moves are adequately designed, although they are still ineffective for fast local diffusion. The set of techniques known as "Enhanced" or "Accelerated" sampling addresses this issue. In most cases, enhanced sampling techniques combine MD approaches with fast local diffusion, with MC steps that generate big jumps in phase space. This facilitates the jump over energy barriers and the transitions between the local basins of the PES, allowing to obtain converged canonical sampling with a feasible computational cost. One of the most extensively used methodologies that follows this philosophy is the so-called Replica Exchange simulations.

2.2.2 Replica Exchange methods

Replica Exchange (RE) simulations aim at improving the convergence speed of statistical sampling of the phase space for systems presenting a rough potential energy surface.[85,168,169] A RE simulation consists of a series of independent sampling simulations (replicas) of the same system in different equilibrium conditions, which are allowed to exchange their molecular configurations every so often. The simulation proceeds by alternating between normal thermalized MD, and MC moves that attempt to exchange configurations between neighboring replicas.

Generally, one of the replicas evolves at some target equilibrium conditions, while the rest evolve under modified conditions that facilitate the exploration of the phase space. Replica exchange simulations typically use modifications in the temperature or in terms of the Hamiltonian (such as the atomic masses or the coefficients of the dihedral or Van der Waals terms in molecular force fields[85,170–172]) to facilitate the crossing of energy barriers and thus accelerate the sampling of canonical probability distributions. The MC exchange of structures between two replicas induces big phase space jumps in the replica evolving at the target equilibrium conditions, which allow transitioning between local basins of the PES and critically reducing the convergence time of the canonical sampling. To ensure that the sampling generated in each replica follows the adequate (generally the canonical) distribution, the exchange of structures must be conditioned to satisfy detailed balance, as discussed before in equation 2.11.

In an exchange of conformations between two replicas R_a and R_b in a RE simulation, the probability $p(A)$ (as in eq. 2.11) is equal to the probability of R_a being in conformation s_1 times the probability of R_b being in conformation s_2: $p(A) = p(R_a \text{ in } s_1)p(R_b \text{ in } s_2)$.

In the canonical ensemble the number of particles and the volume remain constant among all the replicas, so that $p(R_a \text{ in } s_0) \propto e^{-\frac{H_a(s_0)}{k_b T}} = e^{-\beta_a H_a(s_0)} = e^{-h_a(s_0)}$ where h_a is the reduced Hamiltonian $h_a = H_a\beta_a$. In this case, the algorithm is referred as Hamiltonian Replica Exchange (H-RE).[85]

Inserting this in the detailed balance equation:

$$\frac{p(A)}{p(B)} = \frac{e^{-\beta_a H_a(s_0)} e^{-\beta_b H_b(s_1)}}{e^{-\beta_a H_a(s_1)} e^{-\beta_b H_b(s_0)}} = \frac{p(B \longrightarrow A|B)}{P(A \longrightarrow B|A)} = R. \tag{2.14}$$

The most trivial choice for $p(A \longrightarrow B|A)$ is therefore:

$$p(A \longrightarrow B|A) = \min(1, R^{-1}) = \min(1, \frac{e^{-\beta_a H_a(s_1)} e^{-\beta_b H_b(s0)}}{e^{-\beta_a H_a(s_0)} e^{-\beta_b H_b(s_1)}}) = \min(1, \frac{e^{-h_a(s_0)} e^{-h_b(s_1)}}{e^{-h_a(s_1)} e^{-h_b(s_0)}}). \tag{2.15}$$

This last equation is the general acceptance rule between two replicas in a canonical replica exchange simulation, which depends on the overlap between the phase space probability density functions of two replicas. This acceptance rule vanishes exponentially when the equilibrium conditions change, so a series of replicas that interpolate the conditions between the target replica and the “acceleration" replica are necessary to keep a sufficiently high acceptance probability of exchange.

Temperature replica exchange

Temperature Replica Exchange (T-RE[169]), also known as Parallel Tempering,[168,173,174] is arguably the most common variation of RE simulations. In T-RE simulations replicas evolve at different temperatures under the same Hamiltonian, and therefore equation 2.15 simply becomes $p(A \longrightarrow B|A) = \min(1, exp(\Delta\beta\Delta H))$. A specific replica evolves in the canonical ensemble at the target temperature of interest, while other replicas evolve at increasingly higher temperatures. This strategy is useful when the sampling efficiency of the target replica is hampered by high potential energy barriers, although it is not effective to accelerate the crossing of entropic barriers. The temperature of the highest replica is chosen to be high enough to be able to easily overcome the existing potential barriers in the system, so that it can freely explore the phase space. Structures at new areas of the phase space discovered by this replica are passed down to other replicas until they reach the lowest replica at the target temperature, allowing it to teleport to other areas of the conformational space without actually having to cross any potential energy barrier.

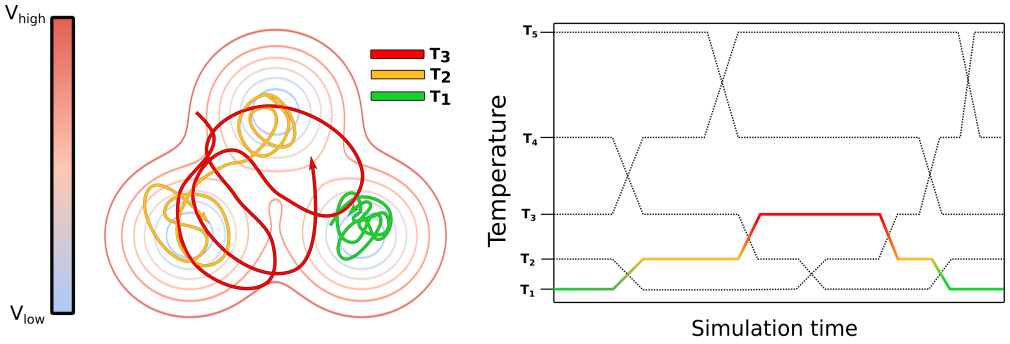


Figure 2.9 – Left: Depiction of different replicas at different temperature navigating a potential energy landscape. Right: Scheme of a temperature exchange simulation.

The optimal number of replicas needed for a T-RE simulation generates an average acceptance probability between 20 and 40%.[175,176] This quantity depends on the overlap of energy distributions between replicas at different temperatures, which depends on the heat capacity. Since the heat capacity is an extensive property, the acceptence probability between two replicas at specific temperatures decreases with the number of particles, which increases the number of replicas necessary to span a given temperature range. The other important parameter (as well as in other RE schemes) is the frequency of attempted exchanges. In principle, in T-RE simulations there are no negative effects in using a high exchange rate frequency, given that the cost of computing the exchange probabilities is very low.[177,178] Notice that a T-RE is equivalent to an H-RE with the Hamiltonian scaled with coefficients β.

Reservoir Replica Exchange

Reservoir Replica Exchange (res-RE) is a technique to accelerate replica exchange simulations by replacing the highest replica with a reservoir of structures that randomly exchanges

structures with the other replicas. The rationale behind the use of a reservoir is to reduce the trajectory correlation time and thus the number of single point computations needed to achieve convergence.[86] Of course, res-RE is only adequate to compute ensemble averages if the reservoir contains samples that follow a known physical distribution, as otherwise, it is not possible to derive a proper exchange probability with the other replicas.

3 Hamiltonian-Reservoir Replica Exchange and Machine Learning Potentials for Computational Organic Chemistry

This chapter is based on the following publication:

R. Fabregat, A. Fabrizio, B. Meyer, D. Hollas, C. Corminboeuf, Hamiltonian-Reservoir Replica Exchange and Machine Learning Potentials for Computational Organic Chemistry, *J. Chem. Theory Comput.* **2020**, 16, 5, 3084–3094.

3.1 Introduction

Machine learning techniques are increasingly used to bypass expensive quantum chemical computations. A typical example are machine learning-based potentials that are exploited to propagate the dynamical evolution of molecular systems on *ab initio* potentials at a fraction of the cost. The seminal work in the field comes from Behler and Parrinello,[28] who trained a generalized Artificial Neural Network (ANN) capable of predicting density functional theory-based energies and atomic forces and demonstrated its capability on bulk silicon[179] and then on carbon[180,181] and sodium.[182] Behler and Marquetand then applied the same approach to n-alkanes[183] and alanine tripeptides.[184] Comparable capabilities were achieved by Csanyi and co-workers, who used kernel-based methods (*i.e.*, the Gaussian approximation potential)[38] to propagate the Density Functional Theory (DFT)-molecular dynamics (MD) of bulk crystal,[77] amorphous carbon[185] and silicon.[186] Kernel ridge methods were also exploited for the "on-the-fly" propagation of the dynamic in the electronic states, circumventing the need for the explicit Time dependent-DFT or the CASSCF computations.[187] Roitberg et al. pushed this approach further and proposed a deep neural network (ANAKIN-ME) with modified Behler-Parrinello symmetry functions to learn the potential of organic molecules approaching the CCSD(T)/CBS accuracy.[40,188,189] Such a high accuracy level was also achieved by Tkatchenko and co-workers using a gradient-domain machine learning (GDML).[33,34,190] Similarly, the SchNet[47,191,192] and PhysNet[193] deep learning architectures were also exploited to predict the

potential energy surface and other quantum chemical properties of molecules and materials.

Overall, machine learning potentials (neural network or kernel-based) achieving post-Hartree-Fock or DFT accuracy were essentially employed for the molecular dynamics of fairly small and rigid systems (*e.g.*, benzene, ethane and malonadehyde, aspirin, uracil, naphthalene, salicylic acid, and toluene)[33,34,42] or alternatively for larger systems with limited chemical diversity (*e.g.*, peptides made of the same amino-acid type).[184] For these reasons, the associated (free) energy landscapes were explored using standard *ab initio* molecular dynamics without the need of making use of accelerated sampling approaches. Describing more flexible organic molecules (*i.e.*, molecules that possess low-frequency (anharmonic) modes and multiple local minima close in energy) with machine learning potentials set additional challenges, which influence both the accuracy of the ML potential and the convergence of the statistical sampling of complex potential energy surfaces.

In 2016, one of us demonstrated[4] the utility of coupling enhanced sampling methods like temperature Replica Exchange Molecular Dynamics[169] (REMD) with the most recent variant of density functional tight binding, *i.e.*, DFTB3[194–196] to map the free-energy landscapes of fluxional organic molecules. This combination allowed to address organic chemistry problems that are not solvable solely relying on static electronic structure computations or standard molecular dynamics, the latter being too short to capture the interconversion between different possible states. A replica exchange simulation overcomes problems associated with running insufficiently long simulations by performing a series of energetically independent simulations (named replicas) of the same system in different equilibrium conditions and allowing them to occasionally exchange their configuration in a way that still ensures a canonical sampling within each replica. Replica exchange is especially appealing when relevant collective variables essential to a metadynamics[197] simulation are not easily identifiable[4] (see Ref.[198] for recent example of metadynamics at the DFTB level). The most common version is Temperature Replica-Exchange (T-RE),[169] an alternative name for REMD, where the replicas differ by their temperature. The additional insights provided by the coupling of REMD and DFTB3 (REMD@DFTB3)[4] were demonstrated on four examples including reaction energy pathways and conformational free energy differences, characteristic of organocatalysts and flexible molecular rotors. While REMD@DFTB3 permits thorough exploration of potential energy surfaces at an affordable computational cost, the accuracy of the electronic structure method was sacrificed to ensure statistical convergence. In fact, the incompatibility associated with obtaining both converged statistical sampling and highly accurate energetics has traditionally prevented the ability of improving the quantum chemical description of moderately sized, yet highly flexible molecules that evolve on complex potential energy surfaces,[199–202] sometimes leading to catastrophic results.[203] In the present work, we achieve high-level *ab initio* accuracy by correcting semi-empirical potentials with a machine learning model based on kernel ridge regression[21] combined with a more general enhanced sampling scheme connecting Hamiltonian[85] (H-RE) and reservoir[86] (res-RE) replica exchange (*i.e.*, resH-RE). With the former, the replicas evolve under a different Hamiltonian instead of a different temperature like in Temperature Replica Exchange, (*i.e.*, T-RE). As for reservoir Replica

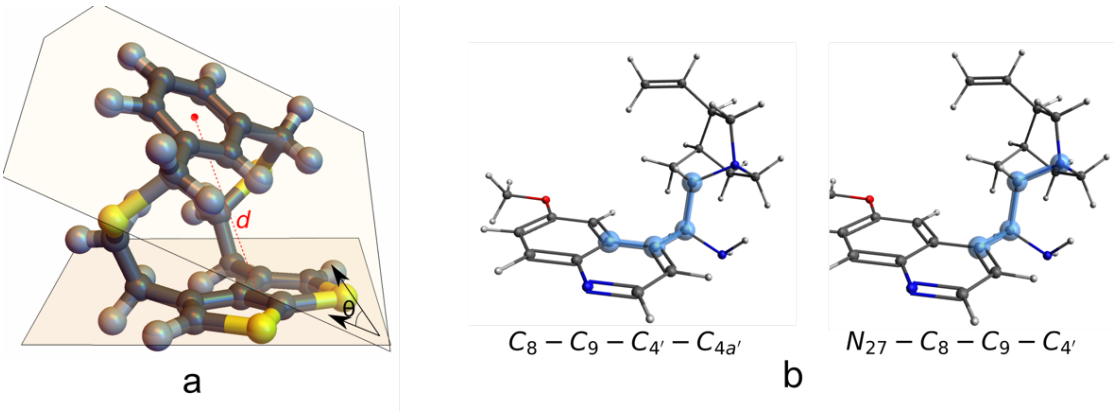


Figure 3.1 – (a) Dithiacyclophane and the collective variables used to characterize its global structure: the distance between the center of masses of each cyclic bulk and the angles between the average planes going through them. (b) The cinchona alkaloid organocatalyst and the two dihedral angles used to characterize its global structure.

Exchange, it was originally developed to improve T-RE by replacing the highest temperature replica with a pool of structures (*i.e.*, reservoir) acting as any other replica but exchanging conformations taken randomly from the pool. The proposed combination of Hamiltonian and reservoir RE dramatically accelerates the exploration of *ab initio* free energy landscapes of archetypes flexible medium-size organic molecules that are dictated by a subtle energetic interplay originating from both enthalpic contributions and conformational entropy. The illustrative systems considered herein are motivated by our previous work[4,204,205] and are (1) the bridged asymmetrically polarized dithiacyclophane, incorporating a thieno[2,3-b][206] and (2) a prototypical cinchona alkaloid organocatalyst.[205,207,208] Specifically, the first molecule is chosen because its relative conformational stability is governed by subtle intramolecular non-covalent interactions that necessitates an accurate *ab initio* treatment, while the large conformational entropy effects can only be accounted for by using accelerated sampling techniques. The free energy landscape of the organocatalyst is a complementary example that depends on individual energy contributions arising from rotational isomerism.

3.2 Methods and Computational Details

3.2.1 Overview

The proposed protocol is schematically illustrated in the workflow given in Figure 3.2, whereas all the details on the quantum chemistry, machine learning models and enhanced sampling approaches are described in the upcoming individual sections. In brief, a low-cost semi-empirical approach is used as a quantum chemical baseline, while the targeted free energies are achieved at an accurate *ab initio* target level with machine learned corrections that learn the difference (Δ) between the baseline and the target (ΔML correction).[55] The semi-empirical level is first used to generate a canonical sampling using T-RE for two purposes. (a) A subset of

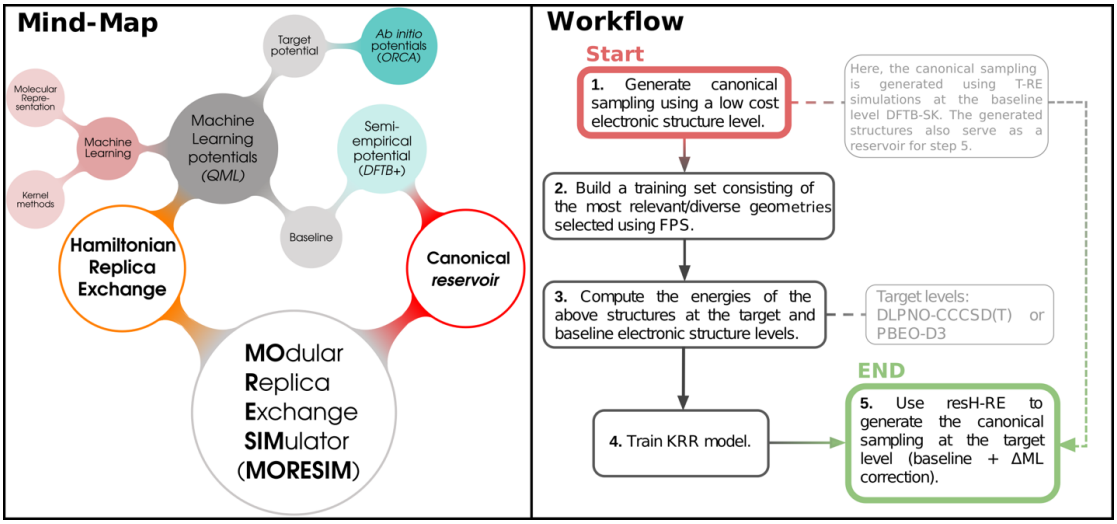


Figure 3.2 – Mind-map and workflow illustrating the proposed methodology.

structures extracted from this T-RE simulation serves to build a training set of energies/structures to train the ΔML model and (b) the pool of structures associated with the T-RE is used as a reservoir (vide infra). The thorough exploration of the free energy landscapes, which is finally performed using a potential corresponding to the semi-empirical level + ΔML correction, combines two variants of replica exchange that are Hamiltonian and reservoir RE (resH-RE). The resH-RE simulations were performed with a modular in-house python implementation of replica exchange uploaded in git-hub[209] in the form of a Python library under the name Modular Replica Exchange Simulator (MORESIM).

3.2.2 Quantum chemical potentials: targets and baseline

DFTB3[194] with the 3OB parameters[195,196] and the Slater-Kirkwood dispersion correction[210] (DFTB-SK) using the DFTB+[211] software is the chosen baseline for the ΔML model and for building the reservoir. The target potential is the domain-based local pair natural orbital coupled-cluster with perturbative triples (DLPNO-CCSD(T)/CBS)[87,88] as implemented in ORCA 4.0.[212] Complete Basis Set (CBS) extrapolations are performed following Neese's scheme starting from Dunning basis sets[213,214] (*i.e.*, cc-pVDZ and cc-pVTZ) computations. PBE0[215]-D3[216]/(6-31G) is also used as target. The TeraChem[217,218] software, which allows GPU acceleration for electronic structure computations, serves to provide a comparison with the direct (exact PBE0 as opposed to ML-based) free energy computations at the PBE0-D3/(6-31G) level. Following the work by Martinez et al.,[218] we utilized an MPI interface between the software for molecular simulations AMBER[219] and TeraChem to perform the GPU accelerated T-RE simulations. At each step of the dynamics, the converged density from the previous step was passed as the initial point for the SCF computation. These PBE0 simulations enable comparison between the explicit *ab initio* free energy landscapes and the faster ML ansatz; a comparison, which is not possible at the DLPNO-CCSD(T) level. Details on the relative

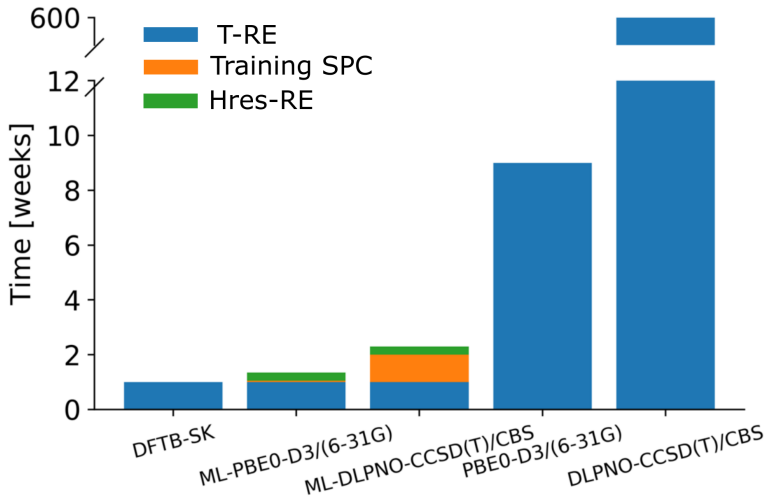


Figure 3.3 – Histogram representing the cost of the computations to generate the dithiacyclophane free energy landscapes. The blue fraction corresponds to the time spent on the T-RE simulations. Orange shows the time spent on the single point computations used to train the ML model. Green is for time spent on the resH-RE simulations. The cost for DLPNO-CCSD(T)/CBS is an estimation.

computational cost is provided in Figure 3.3.

Solvent effects were included implicitly using the SMD58 model (with the dielectric constant of chloroform) at the PBE0-D3/(6-31G) level, also in ORCA 4.0.

3.2.3 Machine Learning Methods

The ML corrections trained to learn the difference between the baseline and target levels are based on Kernel Ridge Regression[21] (KRR) and use the Spectrum of London Axilrod-Teller-Muto (SLATM)[119] molecular representation developed by von Lilienfeld and Huang in the Quantum Machine Learning (QML) package.[220] Among all the tested molecular representations, (*e.g.*, Coulomb Matrix,[29] Bag of bonds[80]), SLATM offered the best accuracy for the class of problems investigated herein. The KRR space was generated with a Gaussian kernel. The training set is built based on the most distinct structures extracted from the DFTB-SK T-RE simulations and correspond to 1500 and 1800 structures and energies for the bridged asymmetrically polarized dithiacyclophane (a), and a prototypical cinchona alkaloid organocatalyst (b) respectively. These sets were divided into a training and a validation set (200 and 300 random structures) were used for validation for each system respectively. Amongst the initial 1500 and 1800 structures/energies, a random subset of 500 were used to optimize the hyper-parameters (*i.e.*, the standard deviation of the Gaussian kernel σ, and the regularization parameter λ) optimized with a Nelder-Mead simplex algorithm.[221] The quality of the trained model is evaluated by the mean absolute errors for the predictions on the test set. Overall, the

final ΔML models offer an accuracy reaching 1 kcal/mol for both the dithiacyclophane and the cinchona alkaloid organocatalyst in comparison to the electronic energy computed at the exact reference level.

3.2.4 Hamiltonian-reservoir Replica Exchange

Two complementary sampling techniques are used for the exploration of the free energy landscapes computed with the ΔML models: Hamiltonian Replica Exchange[85] (H-RE) and reservoir Replica[86] (res-RE). Approaches based on replica exchange typically use parallel simulations with modified parameters (temperature, Hamiltonian, atomic masses, ...) to facilitate the crossing of energy barriers and thus accelerate the sampling of canonical probability distributions.[85,169,173] Over the course of the simulation, the original replica, operating at the target conditions, exchange molecular conformations with the modified replicas as a way to introduce significant jumps in the phase-space. To ensure non-vanishing exchange probabilities, a sufficient number of replicas connecting the original conditions with the other extreme is introduced. In H-RE, the transition between states is accelerated by creating intermediate potentials (*i.e.*, between the baseline and target condition) using a modified Hamiltonian for each of the replicas.[85,171,172] In our implementation, H-RE exploits a reservoir of DFTB-SK structures obtained from previous simulations (vide infra). The replicas evolve at the same temperature (300K) and under a potential $V_\lambda = (1 - \lambda)V_{target} + \lambda V_{low}$ that transition from DFTB-SK (low) to DFTB-SK + ΔML (target) corresponding to post-Hartree-Fock or DFT accuracy. The replica with $\lambda = 0$ evolves with the pure accurate potential $V_0 = V_{target}$ = DFTB-SK + ΔML, while the replica with λ=1 corresponds to the lower-level potential $V_1 = V_{low}$ = DFTB-SK. In practice, the "highest" replica ($\lambda = 1$) is replaced by an available reservoir, generated with the low level potential (*i.e.*, DFTB-SK at 300K), in the spirit of reservoir Replica Exchange (res-RE) (see scheme in Figure 3.4). Here, the canonical DFTB-SK reservoirs (see Figures 3.5a and 3.6a) were taken from previous T-RE simulations (300K) within i-PI.[222,223]

The rationale behind the use of a reservoir is to use the information from the sampling performed at low accuracy to reduce the trajectory correlation time of the sampling at the high level of accuracy, and thus reduce the amount of single point computations needed to achieve convergence.[86] Of course, res-RE is only adequate to compute ensemble averages if the reservoir contains samples that follows a known physical distribution (in our case canonical), as otherwise it is not possible to derive a proper exchange probability with the other replicas that satisfies detailed balance.[165] Coupling the reservoir with H-RE rather than T-RE also allows to accelerate the sampling of the accurate *ab initio* level without increasing the temperature, which leads to several advantages.

First, in comparison to T-RE (achievable only at the semi-empirical level), the reservoir in resH-RE covers the entire conformational space that is accessible to the replica evolving at the target condition, so that exchanging structures with the reservoir effectively generates transitions between free energy basins without the need to actually cross free energy barriers.

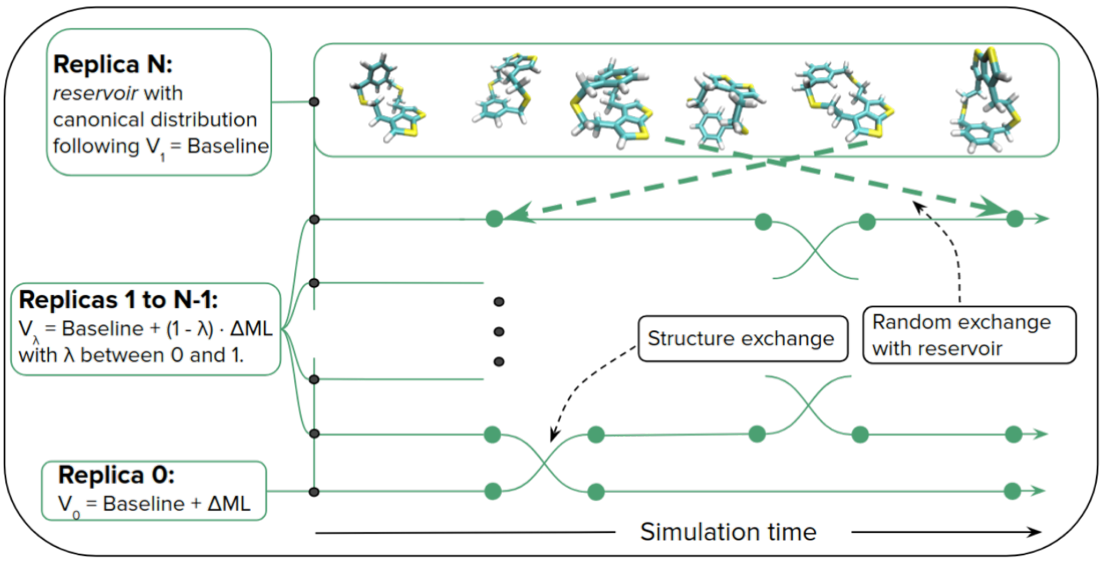


Figure 3.4 – Schematic depiction of resH-RE.

This is convenient because it critically reduces the simulation time needed to obtain converged results, as crossing barriers between free energy basins represent the slowest type of collective motion. In other words, by taking advantage of canonical sampling performed at an affordable semi-empirical level, resH-RE will seamlessly simulate rare events. Second, it prevents the replica trajectories to generate structures out of the domain of applicability of the trained ML potential. Given that the ML model was trained with structures at 300K, its accuracy at higher temperatures can't be guaranteed. Additionally, less replicas (4/6 (H-RE) vs 16/48 (T-RE) (for the two considered molecules) are necessary to achieve optimal exchange probabilities, reducing significantly the computational cost.

Given that swaps between local minima in the free energy landscape (*i.e.*, between basins) and crossings of energy barriers occur through the reservoir, replicas only serve to induce local diffusion in the phase-space. Therefore, the time propagation of each replica can be performed using thermalized Molecular Dynamics, but also with simple Monte Carlo (MC) moves (*e.g.*, random particle moves) that are otherwise largely inefficient for systems with non-linear potential energy surfaces like those investigated herein.[165] In our context, this brings another key advantage, as many of the existing ML-based potentials have not yet been adapted to run molecular dynamics. With kernel-based approaches, the forces can be obtained from deriving the expression of the KRR (and thus of the molecular representation) with respect to the atomic coordinates,[187] but the task is not straightforward for the SLATM representation used here. The alternative, the Gradient Domain ML scheme developed by Müller et al.,[33,34,190] that consists in learning the forces directly is considerably more expensive and only applicable to small molecules. Moreover, only energetic quantities are available for high level *ab initio* potentials like DLPNO-CCSD(T). This work uses resH-RE with MC moves not only because of the unavailability of the forces but also to illustrate that the broad

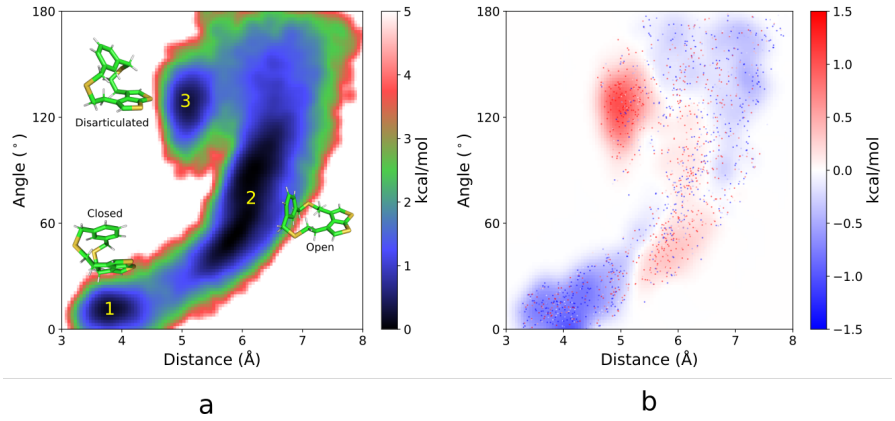


Figure 3.5 – (a) Free energy landscape (DFTB-SK/3OB level) of dithiacyclophane at 300K (T-RE) projected on the 2D space generated by the collective variables visible in Figure 3.1a. (b) Projection of the dataset made of 1500 dithiacyclophane structures extracted with farthest point sampling from the 300K canonical ensemble of 40000 structures and color coded based on the single point energy difference ΔE=((DFTB-SK/3OB) - (DLPNO-CCSD(T)/CBS)). The continuous background is plotted using a gaussian interpolation of the mean energy difference. The smooth histograms were constructed with a Gaussian Kernel Density Estimator (Gaussian KDE) using the SciPy[3] python library.

applicability of the sampling scheme with any of the existing ML potentials.

The convergence of free energy computations was evaluated by analyzing the evolution of the estimated relative free energies between basins. Given that the crossing between basins represent the slowest dynamical mode, the stabilization of the estimated basin free energies represents a good indicator of convergence. Statistical error boundaries on the estimated free energies were evaluated using a block jackknife with a width of one tenth of simulation time.[224]

Note that the adopted approaches are applicable to any molecule but is especially designed for situations when free energy perturbation is not sufficient or suffer from convergence problems. resH-RE is indeed not a simple reweighting scheme; the reservoir in resH-RE is used to accelerate jumps over the conformational space, but the data generated by the replica that samples the canonical distribution of the target potential corresponds to an unbiased sampling of the adequate probability distribution.

3.2.5 Technical details

The 300K free energy landscape of dithiacyclophane was obtained with a resH-RE simulation using 4 replicas (λ=0, 0.33, 0.66, 1) exploiting a reservoir of 40000 structures taken from a previous DFTB-SK canonical distribution of structures obtained with T-RE. A subset with the 1500 most distinct structures were extracted from the reservoir with Farthest Point Sampling[78] (FPS) (Figure 3.5b) and used to train the ΔML corrections. A total of 6 replicas (λ=0, 0.2,

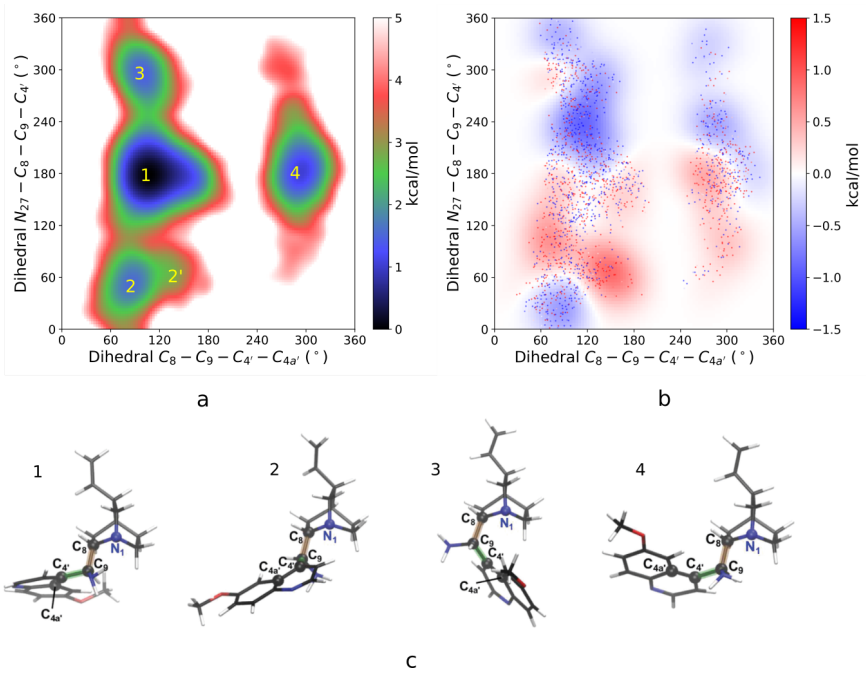


Figure 3.6 – Free energy landscape (DFTB-SK/3OB level) of the cinchona alkaloid organocatalyst at 300K projected on the 2D space generated by the collective variables visible in Figure3.1b. Constructed with canonical structures generated with T-RE simulations with DFTB-SK as potential energy. (b) Projection of the 1800 dataset structures obtained with FPS from a canonical ensemble of 32000 structures at 300K canonical ensemble and color coded based on the single point energy difference ΔE=((DFTB-SK/3OB) - (DLPNO-CCSD(T)/CBS)). (c) Structures representing each of the 4 conformational regions (*i.e.*, basins).

0.6, 0.8, 1) were required for the resH-RE simulations of the cinchona-based asymmetric organocatalyst, and the 1800 most distinct structures (Figure 3.6b) extracted from a reservoir of 32000 structures obtained as discussed above were used for the training. For both resH-RE simulations, it was ensured that the exchange rate between replicas reaches an optimal 30%.[225] Exchange between replica were attempted every 20 MC steps consisting of a Gaussian random displacement of all atoms (in cartesian coordinates) with standard deviation σ=0.03 Å, set to 50% acceptance rate. The MC simulations correspond to a total of 10^6 steps for both systems.

3.3 Results

3.3.1 Dithiacyclophane

The three conformational regions of dithiacyclophane, previously investigated by one of us,[4] and visible in Figure 3.5a, are controlled by very distinct enthalpic and entropic contributions. The π-stacked "closed" conformer is stabilized by long-range correlation effects and only captured at the DFT level upon addition of a London dispersion correction.[204,205,226] In sharp

contrast, the open conformer is highly flexible and driven by entropy and large anharmonic effects. The third "disarticulated" conformer is rigid but less sensitive to London dispersion forces in comparison to the closed region. Our former T-RE DFTB simulations that captured all the conformational entropy effects, highlighted the limitation of the harmonic approximation for describing the relative stability of the most floppy (*i.e.*, open) conformer (see Figure 3.5a).

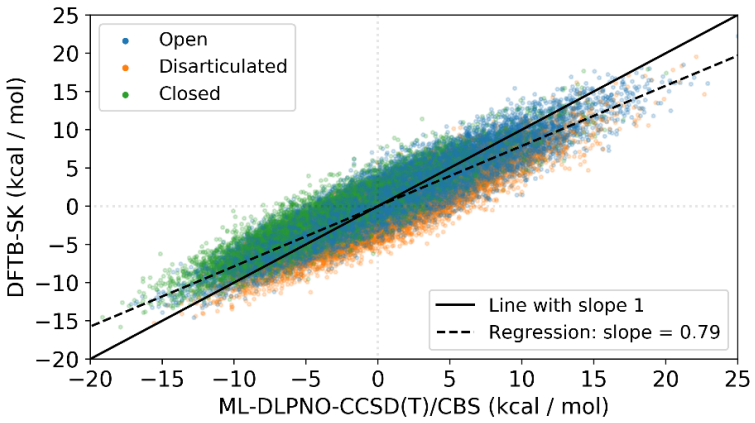


Figure 3.7 – Comparison between the DFTB-SK electronic energy and the ML predictions (*i.e.*, DFTB-SK + ΔML correction) for the 40000 structures in the reservoir.

The harmonic approximation fails to account for the full conformational entropy contributions and the anharmonic nature of the open state (2) (see Ref.[227] for relevant examples of approximations for anharmonic free energies) and to a lesser extent of the closed conformational region (1). The large entropic contributions characterizing the open region (see Figure 3.5a) makes it the lowest-energy conformer at the DFTB-SK level at 300K and temperatures above. Yet, the DFTB relative free energies between the three conformers are very small (within 1 kcal · mol^{-1}). Converging the statistical sampling comes with a quantum chemical cost and the affordable semi-empirical level is not expected to capture all these subtle energy differences accurately. The ML correction to DFTB-SK offers access to converged DLPNO-CCSD(T) free energy profiles at a fraction of the cost (vide supra). Prior to obtaining the full free-energy landscapes with resH-RE, it is interesting to identify the trends emerging from the ΔML correction added to the DFTB-SK energy of the structures in the reservoir (Figure 3.7). The 0.79 regression slope between DFTB-SK and ML-DLPNO-CCSD(T) is indicative of the much flatter potential energy surface of the former or, in other words, an underestimation of the energy differences and barriers across the energy landscape.

The consequence of these energy discrepancies is clear when comparing the full free energy landscapes and relative free energies (upon integration within the free energy basins,[4] Figure 3.8) obtained with resH-RE sampling at different quantum chemical levels. Overall, the shape of the ML-DLPNO-CCSD(T)/CBS and DFTB-SK profiles are very similar but the disarticulated basin is strongly favored by CCSD(T) at the detriment of the close conformer (>2 kcal · mol^{-1} higher). In contrast to the flat DFTB-SK free-energy profile, the ML-DLPNO-CCSD(T) land-

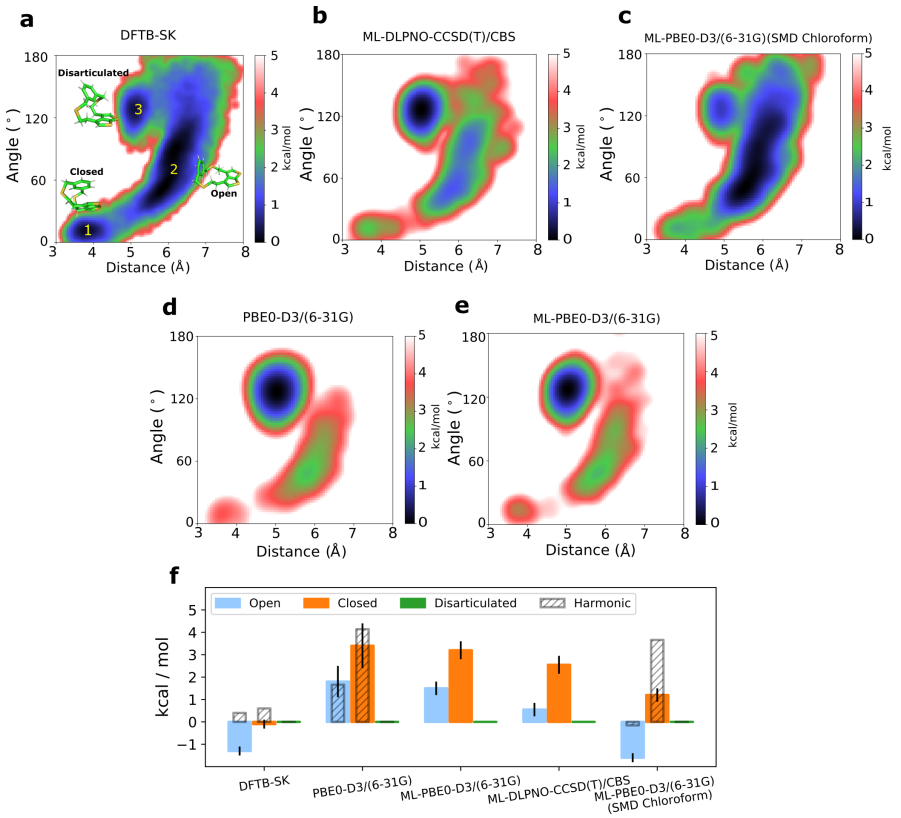


Figure 3.8 – Free energy landscapes at 300K generated with the potential: (a) DFTB-SK (b) ML-[DLPNO-CCSD(T)/CBS] (c) ML-[PBE0-D3/(6-31G)(SMD Chloroform)] (d) PBE0-D3/(6-31G) (e) ML-[PBE0-D3/(6-31G)]. (f) Relative free energies by integration within the local minima.[4] The free energies are all given relative to the Disarticulated. The stripped columns correspond to the static relative free energy using the harmonic approximation (for the solvated system the harmonic free energies were computed with the true potential, and not with the machine learning version). All the free energies maps come from resH-RE expect for the direct PBE0, which uses T-RE, as described in the method section.

scape is clearly uneven highlighting the difficulties of the tight-binding method to reproduce the delicate interaction interplay characterizing the conformational regions of this illustrative system. The PBE0 free energy landscapes are even less flat and favor the disarticulated state even more with the closed structure being around 3 kcal $\cdot$ mol^{-1} higher. Yet, the excellent agreement between PBE0-D3/(6-31G) and ML-PBE0-D3/(6-31G) is a proof-of-principle demonstration that this trend is not an artifact from the ML potentials (see Figure 3.8d and Figure 3.8e).

Note that a direct approach using DLPNO-CCSD(T)/CBS is not achievable, given the intrinsic computational cost of the method. For this specific reason, a relatively small basis set was chosen for the DFT profiles. While some of the deviations between DFTB-SK and the higher level approaches (*i.e.*, much smaller energy differences) are already apparent in the static picture (see the harmonic free energies in Figure 3.8), the deviations between methods are more

pronounced when accounting for thermal fluctuations. With PBE0-D3, the approximated barriers between each conformational regions is over 5 kcal $\cdot$ mol^{-1}, which is significantly higher than the DFTB-SK (<2 kcal $\cdot$ mol^{-1}).

The Monte Carlo approaches used in this work are not easily compatible with the inclusion of explicit solvent but the effect of the environment is of course essential to decipher the true molecular behavior and its associated PES. As a compromise, solvent effects were incorporated implicitly with the SMD continuum model (with the chloroform dielectric constant[206]) at the PBE0 level. The inclusion of these effects severely affect the relative energetic stability and flatten the entire profile looking much more similar to the gas phase DFTB-SK profile. The limitation associated with the continuum model could be overcome by using an additional potential that models the interaction between the solute and explicit solvent within a dynamic simulation, a possibility that we will explore in future work.

3.3.2 Cinchona Alkaloid

The same methodology is applied to a common cinchona-based asymmetric organocatalyst for which we also generated the *ab initio* free energy landscape (Figure 3.9).[4] The 2D conformational map extracted from the 2016 T-RE simulations at the DFTB level revealed four easily accessible conformational regions (1- 4) and one that is much less populated (2'). Unexpected from previous static computations was the dihedral angle (open rather than close) characteristic of the conformational state 3. Other added values from the T-RE simulation were the demonstration of the pronounced entropic nature of 1 at 300 K reversing the relative stability between 1 and 4 in comparison with the static computation and the appearance of region 2'. Yet, the 1 and 4 conformational regions were within 2 kcal $\cdot$ mol^{-1} stressing the importance of improving upon the DFTB level.

With a slope much smaller than 1 (*i.e.*, 0.76), this example confirms the general flatness of the DFTB-SK potential compared to that of DLPNO-CCSD(T). Note that the DFTB3 underestimation of the rotational barriers and of the relative energy differences, which originates from the limited amount of atomic overlap afforded by the use of a minimal basis set, is reminiscent of other examples in the literature.[195,228]

Figure 3.9 compares the full DFTB profile with those obtained with ML-DLPNO-CCSD(T) and ML-PBE0-D3 accounting for the implicit effect of the chloroform environment.[229] The general shape of the ML-DLPNO-CCSD(T)/CBS free energy landscape of the organocatalyst is once again similar to the DFTB-SK one but with significant differences. Specifically, 4 is clearly enthalpically stabilized at the higher level, whereas the flexibility of the conformational region 1 is enhanced (*i.e.*, larger spread over the dihedral angle characteristic of the syn/anti conformation). The meta-stable region 2' is also more populated at this level. Quantitatively, these trends translate into 1 and 4 lying very close in energies (within 0.5 kcal $\cdot$ mol^{-1}) with state 3 being nearly 3 kcal $\cdot$ mol^{-1} higher and disconnected from region 1 (*i.e.*, high barrier separating the two regions). Akin to the dithiacyclophane, the PBE0 gas phase profile is much

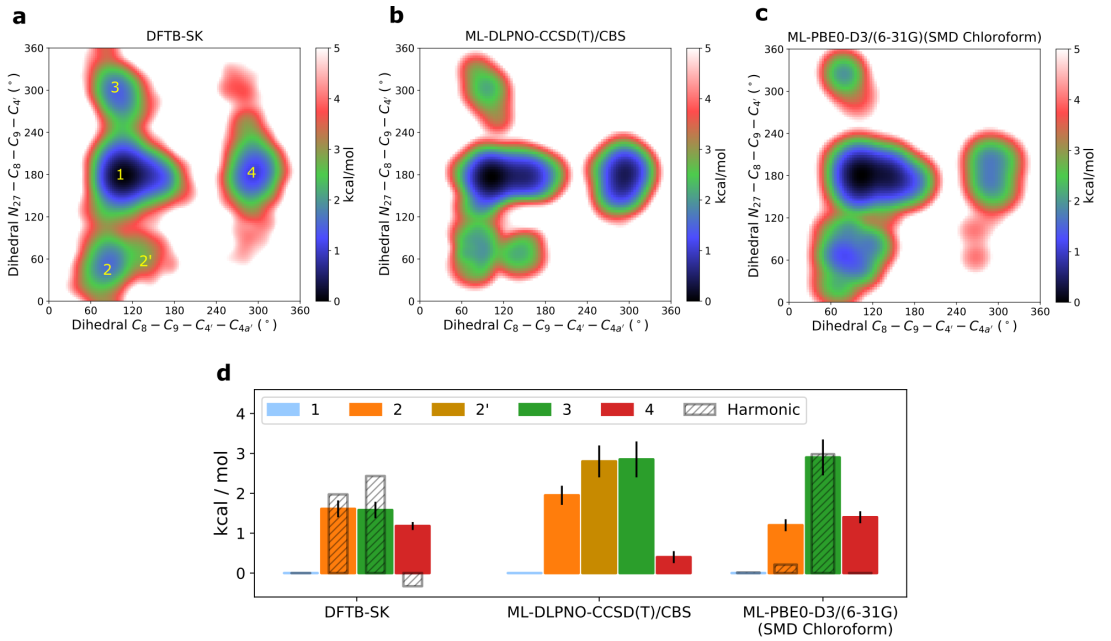


Figure 3.9 – Free energy landscapes at 300K generated with the potential: (a) DFTB-SK b) ML-DLPNO-CCSD(T)/CBS c) ML-[PBE0-D3/(6-31G)(SMD Chloroform)]. (d) Free energies upon integration within the free energy basins. The free energies are all given relative to state 1. The stripped columns are the free energy predictions of the basins using the static free energies using the harmonic correction.

closer to ML-DLPNO-CCSD(T)/CBS than the implicit solvated (in chloroform) profile but the flattening of the free energy landscape of the cinchona derivatives upon implicit solvation is less pronounced than for the dithiacyclophane (see Figure 3.9c). Overall, the effect of the solvent on conformer 3 is negligible but the meta-stable 2' specie disappears, while 2/4 are more/less populated.

These two complementary examples are associated with different energetic driving forces that are the interplays between pronounced intramolecular vdW interactions and conformational entropy in the first case and the individual contributions arising from rotational isomerism in the second.

3.4 Conclusions

In 2016, we highlighted the importance of thorough mapping of the free energy landscapes for solving problems in computational organic chemistry. In this subsequent step, we demonstrate how to exploit a variant of Hamiltonian replica exchange and kernel-based machine learning potentials to achieve a remarkable accuracy/cost ratio and accelerate the accurate predictions of relative free energies, which is one of the most challenging goals in computational quantum chemistry. Overall, our results stress the relevance of improving the entropic

and enthalpic description of flexible organic molecules having complex free energy landscapes dictated by subtle energetic interplays. In particular, based on comparisons between the DFTB-SK baseline and the ML-DLPNO-CCSD(T) target, one concludes that the semi-empirical method generally leads to much flatter free-energy landscapes. Similarly, such systems are poorly described by the picture arising from static free energies, which underestimate the conformational entropy of the most flexible conformational regions. For all these reasons, our original combination of Hamiltonian and reservoir replica exchange and its implementation into a modular environment (the python package MORESIM) represents a powerful solution capable of accelerating enhanced sampling simulations involving any machine learning-based or *ab initio* potential energies. Subsequent objectives will consist of using the same workflow but circumventing the reduced transferability associated with the use of a global molecular machine learning representation by deriving a differentiable kernel approach based on local atomic environment that is also compatible with molecular dynamic simulations.

4 Local Kernel Regression and Neural Network Approaches to the Conformational Landscapes of Oligopeptides

This chapter is based on the following publication:

R. Fabregat, A. Fabrizio, E. Engel, B. Meyer, V. Juraskova, M. Ceriotti, C. Corminboeuf, Local Kernel Regression and Neural Network approaches to the conformational landscapes of oligopeptides, *J. Chem. Theory Comput. Submitted for publication.*

4.1 Introduction

Machine learning (ML) techniques have begun to supplement atomistic simulations by facilitating access to the potential energy surfaces (PES) with an unprecedented accuracy at greatly reduced computational cost.[28,33,230] Behler and Parrinello's seminal work introduced one of the first condensed-phase potentials based on a neural network (NN). Using atom-centered symmetry functions to encode the molecular structures[28,39] and expressing the corresponding potential energy as a sum of the atomic contributions makes the potentials transferable and scalable. In recent years, several NN architectures for atomic based potentials have been proposed, including SchNet[47,128,192,231] and PhysNet,[193,232] which predict energies, forces, and other properties (*e.g.*, dipole moments or chemical potentials) of various chemical systems. Roitberg and coworkers also introduced the ANI-1[188] model, where single-atom atomic environment vectors (AEVs) are used to build deep NN potentials to approach the golden standard of CCSD(T)/CBS for reaction thermochemistry, isomerization and drug-like molecular torsions.[189] Despite their widespread use, NNs have drawbacks: lack of interpretability, the non-deterministic and computationally demanding training, and the large amounts of training data required are some of them.

As an alternative to artificial NNs, kernel-based approaches such as Kernel Ridge Regression (KRR) and Gaussian Process Regression (GPR) overcome some of these limitations.[21] Kernel methods build a map between a target system and its properties by evaluating a similarity measure between the target and a set of known reference points. Gaussian Approximation

Potentials (GAPs)[38,124] pioneered the use of kernels in molecular dynamic simulations and demonstrated that they can achieve results equivalent to NNs. Since then, they have been used to model bulk materials ranging from simple silicon,[77,186,233–235] to ternary $Ge_2Sb_2Te_5$.[236,237] In the wake of GAPs, numerous alternative kernel-based and linear methods have been proposed to predict PESs for atomistic simulations, including support vector machines (SVM)[238] and the Spectral neighbor analysis potentials (SNAPs).[239] More recently, the symmetrized gradient-domain (sGDML) model has proven to yield nearly exact molecular dynamics simulations for small molecules based on coupled-cluster energies and forces.[33,34,190] However, despite the increasing number of kernel-based ML potentials, artificial neural networks remain dominant for driving atomistic simulations.[39,240–247]

When paired with global molecular representations (*e.g.*, Coulomb matrix,[29] bag of bonds (BoB)[80] or the Spectral London Axilrod-Teller-Muto[119] (SLATM)), which encode the key physical information about the structure and composition of molecules as whole indivisible entities, kernel models are often lightweight, making them ideal for predicting molecular properties.[52,53,77,248,249] However, predictions made with these global representations are expected to be accurate only for molecules of similar size and composition with respect to those in the training set. These constraints limit severely the exploration and extrapolation to larger chemical and conformational spaces. Local representations (*e.g.*, FCHL,[31,82] aSLATM,[119] and SOAP[93]), which describe molecules as a collection of atoms within their local environments, provide a greater transferability,[250] but also significantly increase the computational cost of kernel-based methods, as similarity between molecules is then computed as a function of the pairwise similarity between atoms.[125] To restore the data-efficiency typical of kernel-based methods and efficiently exploit the local representations, one can resort to sparse regression techniques. The simplest form amounts to sampling the entire set of atom-centered environments and retaining only the (a priori) most informative environments, assuming that substantial redundancy arises from recurring environments across training structures. The criteria for selection tends to be based on techniques such as Farthest Point Sampling (FPS) or CUR matrix decomposition,[78] that maximize the dissimilarity of the selected environments. While the environments sampled with FPS or CUR based methods represent the most varied set among the training instances, they are not necessarily the best for regressing the property of interest,[251] as the dissimilarity in the representation space does not necessarily correlate with dissimilarity in the target space.[10] (Δ-)[55]ML approaches represent a typical illustration of this issue, as the vast majority of chemical environments are well described by an approximated baseline model while the error is concentrated in localized areas of the feature space. This is particularly true when predicting PESs, where capturing the conformational changes (*e.g.*, a torsion of a single dihedral angle) is as crucial as capturing the dependence on chemical diversity.

In this work, our goal is to address the limitations of traditional unsupervised sparsification techniques and leverage the data-efficiency and transferability of local kernel models, by combining a Local Kernel Regression (LKR) framework with a flexible orthogonal matching pursuit (OMP) algorithm. The efficiency of the resulting model is demonstrated by predicting

the PES of a set of 52'000 conformations of dipeptides comprised of 26 amino acids. In this context, the OMP controls the sparsification process and selects (amongst tens of thousands of atom-centered environments present in the training set) the best possible reference pool for predicting the PESs of any dipeptide. To increase the smoothness of the target energies, the model is baselined with density functional tight-binding (DFTB[195,196]) using a ΔML approach,[55] with the model improving the description of the PES in regions that are traditionally not accurately captured with the semiempirical baseline method (*e.g.*, hydrogen atoms and polarized bonds). To further illustrate the transferability of LKR, we compare its performance with a state-of-the-art Behler-Parrinello type Neural Network, both on the dipeptide set and in an extrapolation test based on the Phe-Gly-Phe tripeptide. The two ML models are then used to drive enhanced sampling simulations to describe the free energy landscape of the tripeptide with DFT accuracy.

4.2 Methods

4.2.1 Machine learning models

The ML potentials presented in this work correct a semi-empirical baseline obtained from density functional tight-binding (DFTB) with the D3(BJ)[252] dispersion correction (shortened DFTB hereafter), and target PBE[253]-dDsC[254–256] (shortened PBE hereafter) for DFT accuracy. For each molecule in the dataset, the property learned within the Δ-ML framework corresponds to the difference between the atomization energy evaluated at DFTB and PBE. For both levels, the atomization energies are computed using a two step procedure. First, the contribution of each atom type to the total energy is evaluated by a multilinear regression (MLR) on the full dataset (dressed-atom energies). Then, the difference between the computed total energy and the sum of the dressed-atom energies yields the atomization energy used herein. The following sections describe the two types of complementary ML architectures exploited in this work.

ML model 1: Sparse Local Kernel Regression

The LKR inputs are the target molecular properties and the atomic representations of the corresponding molecular structures. In this case we used the atomic Spectral London Axilrod-Teller-Muto[119] representation (aSLATM) (see step 1 in Figure 4.1, upper panel) but other local atomic representations could be used. As it is standard procedure for local kernel based atomistic models, LKR uses a selected pool of reference atomic environments taken from the training structures as the regression basis for predicting the target property. The structures available for the training are projected onto the pool of atomic environments using a Gaussian kernel to create the matrix **S**, effectively generating a new vectorial representaion of the molecules (see step 2 in Figure 4.1, upper panel). By assuming a linear relationship between the features of **S** and the global molecular properties, LKR allows to obtain the

regression coefficients for each reference atomic environment without requiring an a priori decomposition of the target property, which is sometimes possible[257] but highly non-trivial for complex PES like the ones discussed here. If the pool of atomic environments is too large, a pre-filtering, which reduces the redundancy of the pool is needed. Here, we use Farthest Point Sampling,[78] which selects the N most distinct environments in terms of their Euclidean distances.

For the final selection of the reference environments, the reduction of the training environments is commonly performed by constructing multiple models including a variable number of the FPS points, which is gradually increased until achieving a satisfying accuracy. It was already hypothesized[258] that some sort of supervision in the sparsification procedure would be desirable. Here, we rely on a supervised sparse regression model called Orthogonal Matching Pursuit (OMP).[89] OMP is a greedy optimization algorithm that finds the best sparse choice of reference environments for a particular application (see step 3 in Figure 4.1, upper panel). The OMP algorithm searches greedily through the whole pool of atom-centered environments and selects at each time the specific environment that reduces the most the prediction error, (*i.e.*, the one with the highest inner product with the targeted property). At each iteration, the contributions from the previously selected environments to the global target property is subtracted and the search continues for the best-match of the residual until convergence. With this procedure, OMP automatically identifies the most suitable, property-specific environment subset (*i.e.* best-matching basis) for the regression of the targeted molecular property in one shot. In the prediction step (see step 4 in Figure 4.1, upper panel), the similarity of each new atomic environment with respect to the reference pool is evaluated by computing a kernel sum with all the selected environments. The reader is referred to Figure 4.1 for a schematic depiction of the workflow.

Overall, LKR-OMP combines the scalability and transferability of NNs, with the faster training and stability of kernel based models. The addition and removal of training data also requires minimal computational effort, as opposed to an NN, for which the procedure requires at best a partial retraining. This would be especially beneficial for active learning approaches,[259] when the training data evolves throughout the process. The counterpart is that the cost of the model scales linearly with the number of reference environments, while the cost of NNs is fixed by the architecture.

ML model 2: Behler-Parrinello neural networks

To benchmark the LKR model against an established NN architecture we further construct a Behler-Parrinello artificial neural network.[28] For each atom, we describe the positions of all neighboring atoms inside a cutoff radius (its "atomic environment") by a set of atom-centered many-body symmetry functions (SF)[260] (see the Computational Details).

To allow for on-the-fly estimation of the uncertainties in the predictions, a committee of four Behler-Parinello neural networks (NN),[28,260] which only differ in the random initialisation of

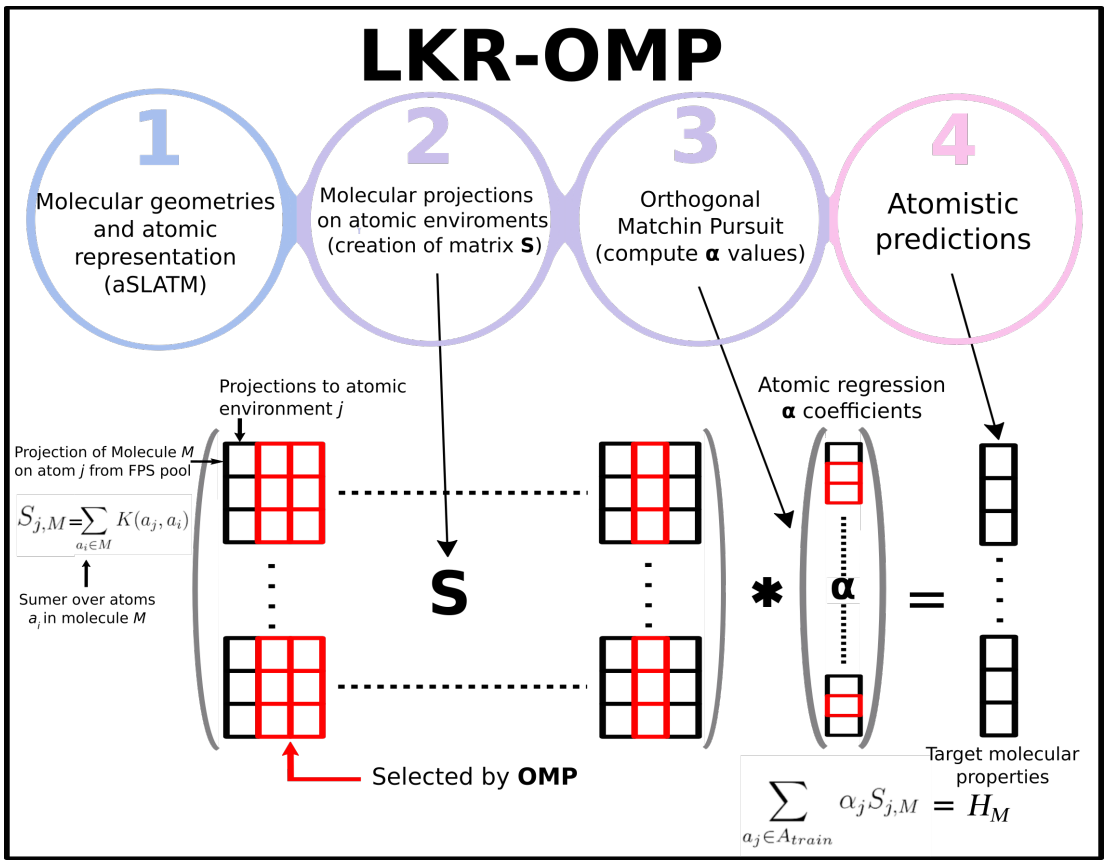


Figure 4.1 – Workflow and schematic depiction of the LKR model.

the NN weights and the internal cross-validation splitting of the training data, was trained to reproduce the differences between the DFTB baseline and the target DFT energies and forces. This permits estimating the uncertainty associated with each committee prediction of the Δ-ML correction following the scheme introduced in reference [Musil 2019].[261] The uncertainty estimates were also used to modulate the application of the NN correction, using the weighted baseline scheme proposed by Imbalzano and coworkers.[262] This procedure minimises the uncertainty in the total potential, and ensures that the it falls back to the baseline whenever the ML correction enters the extrapolative regime, thereby stabilizing the simulation. The total energy is calculated as the sum of the outputs of atomic NNs, and analytic gradients and thus forces are readily available. To train the NN model both energies and forces were used.

Training data

The training set for the construction of the models described in the previous sections was built by selecting configurations from the 300K replica of a DFTB-based temperature replica exchange (T-RE) simulation (with replicas at temperatures between 300K and 1000K) for each amino acid dipeptide. The most distinct 2'000 configurations of each dipeptide were selected

by means of a farthest point sampling algorithm,[77,263] using the the Ramachandran plot[264] coordinates as the independent variables.

For a total of 26 amino acid dipeptides,[265] we obtained a pool of 52'000 conformations. Finally, to include the effects of sidechain-sidechain interactions into the model, the training set was completed with an additional set of 3378 optimized peptide dimers from the BioFragment Database.[266] Single point computations were performed to obtain energy and forces at the target and baseline levels.

4.3 Enhanced sampling methods for the tripeptide

We use the reservoir-Hamiltonian Replica Exchange (resH-RE)[230] technique to sample the canonical ensemble of the selected Phe-Gly-Phe tripeptide at 300 K with the LKR potential. ResH-RE is an enhanced Hamiltonian Replica Exchange[85] scheme, which serves to accelerate the sampling of the configurational space at a high level of theory using a canonical reservoir of structures generated with a less accurate but computationally cheaper potential energy. The replicas essentially help to capture the local diffusion in the phase space, whereas the most dramatic conformational changes, such as swaps between local minima and crossings of energy barriers, occur through coupling with the reservoir. By construction, the resH-RE simulation can be driven by molecular dynamics in the NVT ensemble, but also by simpler Monte Carlo (MC) moves (*i.e.*, random particle moves), which are otherwise largely inefficient for systems characterized by highly non-linear PESs.[165] The possibility of using both molecular dynamics and Monte Carlo moves within resH-RE is especially advantageous given that the atomic forces are not readily available with the LKR model used here, albeit, in principle, obtainable through computing the LKR energy derivatives[a] with respect to the nuclear coordinates.[187]

Considering that the forces are available and actually needed to increase the robustness of the NN potential (*vide infra*), the sampling of the tripeptide in that case was performed using the ATLAS metadynamics framework,[267] which employs a divide-and-conquer strategy to enable efficient biasing when working with many collective variables (CV). In ATLAS, high-dimensional CV space is divided into basins, each of which is described by an automatically-determined, low-dimensional subset of the CVs on which a local, well-tempered metadynamics-like bias is constructed. The local biases are smoothly translated into an effectively high-dimensional bias using indicator functions based on a Gaussian mixture model. Given the high dimensionality of the CV space of the Phe-Gly-Phe tripeptide, attempting convergence with conventional metadynamics would be futile. Meanwhile, the ATLAS framework was specifically designed to work in high dimensions, and it has already been tested on 6D spaces.[267] Alternatively, the sampling of the tripeptide with the NN implementation could have been done using T-RE simulations or other methods based on temperature ac-

[a] To obtain the LKR energy derivatives, it is necessary to derive both the kernel and the underlying molecular representation with respect to the nuclear coordinates. The SLATM representation used in this work has a rather cumbersome mathematical form, whose analytical derivatives are not readily obtained.

celeration.[268,269] However, temperatures past the bond dissociation range are necessary to overcome the increased energetical barriers between basins at DFT and *ab initio* levels of theory, compared to the flatter profile of DFTB.[230]

In this work, space is divided into five basins, identified by applying the PAMM framework[270] to an initial well-tempered metadynamics trajectory using the end-to-end distance of the backbone as the sole CV. Each basin is described and biased based on the two principal axes determined by performing a PCA on the associated distributions of configurations in the six-dimensional CV space. The resultant metadynamics trajectories were unbiased using the ITRE scheme,[271] which makes efficient use of the entire trajectory and that does not require the distribution to be evaluated on a grid, rendering it suitable for high-dimensional CV spaces.

4.4 Computational details

All the baseline computations for the Δ-ML model were performed with DFTB3/3OB[195,196] in combination with the D3BJ[252] dispersion correction (DFTB), as implemented in the DFTB+ software.[211] The target potential was set at PBE[253]-dDsC[254–256] using the def2-TZVP basis set, as implemented in GAMESS-US.[272,273] Canonical sampling of each dipeptide was performed using T-RE simulations using the REMD@DFTB[4] protocol implemented in i-PI.[222] The simulations included 16 replicas with temperatures ranging from 300 K to 1'000 K, equally spaced on a logarithmic scale. A time step of 0.75 fs was used in the dynamics, which ensured the stability and energy conservation of the dynamics, with a Langevin thermostat to control the temperature. The simulations were run for two million steps, which ensured statistical convergence of the results. The final batch of structures was split in two separate sets (respectively 70% (40'000) and 30% (15'378) of the molecules), which were used for training and testing of the models. The resH-RE simulations were run using the MORESIM python package.[230] They included four replicas with a potential linearly evolving from DFTB to DFTB + LKR. This choice resulted in an exchange acceptance probability of 40%. The resH-RE simulations were run for two million steps, which provided converged results. A global random displacement with a Gaussian distribution of standard deviation 0.01 Å was chosen as the Monte Carlo step, which resulted in a 50% acceptance rate.

All metadynamics simulations were performed by coupling the i-PI energy and force engine[223] to the open-source, community-developed PLUMED library[274] version 2.8.0-dev (git: 79bcb8947)[275] to apply a well-tempered bias, and the DFTB+[211] and LAMMPS[276] codes to evaluate the baseline potential and Δ-learned correction, respectively. All simulations employed a time-step of 0.5 fs and a generalized Langevin equation (GLE) thermostat.[277,278]

The Local Kernel Regression implementation (available on github[279]) relies on a Gaussian Kernel[29] and on the aSLATM representation, as provided in the QML-toolkit.[220] The width of the Gaussian Kernel $\sigma = 4.5$ was obtained after a systematic grid search. We used FPS to preselect a first pool of 39'000 local atomic environments. The optimal number of reference

environment selected by OMP can be obtained using a grid search optimization of this parameter (LKR-optimal), although the bigger the number the higher the cost of the model. To achieve a converged statistical sampling (with resH-RE) at a reasonable computational cost, the size of the pool of reference environment is limited to 1'000 (LKR-1000). The python library Sci-Kit Learn [137] was used to perform the OMP regression.

The N2P2 framework [280] was used to train the NN models. Initial many-body symmetry functions (SF), [260] which describe the local, atomic environment of each atom in a configuration and provide the inputs to the NNs, were generated following the protocol of Imbalzano *et al.*, [78] and included G2 functions with $N = 12$ and cutoffs $r_c = 8, 12, 16$ Bohr, and G3 functions with $N = 4$, $r_c = 8$ Bohr, and $\zeta = 1, 2, 4$ and with $N = 2$, $r_c = 12$ Bohr, and $\zeta = 1, 2$. The cut-offs are long enough to describe the environment of the central atom substantially beyond its nearest neighbours in order to address the local differences between DFTB and DFT (long-range discrepancies between DFTB and DFT are also accounted for, albeit in a mean-field manner, through their effect on the local atomic environments). The 512 most informative among them were extracted using the semi-supervised PCovCUR scheme; [281] a modification to the CUR approach, which uses a mixing parameter (here set to 0.5) to smoothly interpolate between a feature-covariance and a linear regression-like loss to identify features that reflect the (structural) variance of the dataset while correlating the the target property. We concatenate their values for a given atomic environment into a feature vector and fed into the "atomic" NN, which in the following consists of two fully-connected, hidden layers with 24 nodes each. This particular architecture has previously proven sufficiently flexible to describe molecular crystals containing up to four chemical species [282,283] and multi-layer perceptron networks with similar depths and widths have seen widespread success for a variety of molecular and condensed matter systems. [284,285]

4.5 Results and Discussion

4.5.1 Performance of the trained machine learning models

The need for correcting DFTB to obtain reliable PESs for each amino acid dipeptide is made evident by Figure 4.2a, showing the histogram of the differences with respect to the target PBE (after removing the multilinear regression contribution). The inaccuracy of DFTB is also illustrated by the regression slopes between the atomization energies at the DFTB level with and without ML corrections (Figure 4.2b) and the PBE atomization energies. For each dipeptide, the slope between uncorrected DFTB and PBE is consistently smaller than unity, implying a systematic overstabilization of the most distorted configurations and an energy understabilization for the most stable ones (see Figure 4.3 for a more detailed analysis of the individual dipeptides). The flatter characteristic of the DFTB PESs has previously been discussed [229,230] and attributed to the limited amount of atomic overlap afforded by its minimal valence basis, which also affects the rotational barriers. [195] As shown in Figures 4.2b and 4.2c, the LKR and NN models correct for the systematic flattening of the PESs (slope ~1,

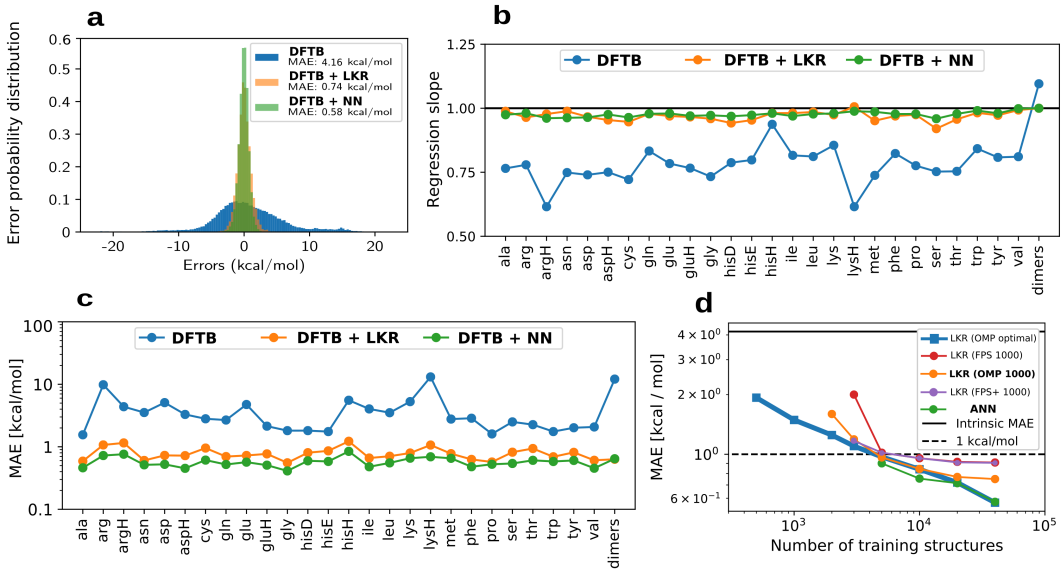


Figure 4.2 – (a) Histogram of errors in test samples of the dipeptide dataset. (b) Regression slopes between 'bonding energies' of DFTB and PBE for each of the training dipeptides and for the dimers. (c) MAE achieved by the models in the test data for each dipeptite and for the peptide dimers. (d) Learning curves, *i.e.*, achieved MAE vs. number of structures used for the training. The different learning curves are: LKR using OMP with the optimized number of atomic environments (blue), LKR exploiting OMP to select the best 1'000 environment (orange), the Behler-Parrinello based NN (green), LKR using FPS to select the most distinct atomic environments, using 200 atoms per atom type (FPS 1000) (red), LKR using FPS to select the most distinct atomic environments but with the same distribution as OMP (FPS+ 1000) (purple).

Figure 4.2b), and also decrease the absolute errors for each dipeptide. As shown by the learning curves (Figure 4.2d), the NN (0.58 kcal/mol, 40'000 training dipeptides) and LKR-OMP(optimal) (0.57 kcal/mol, 40'000 training dipeptides) predictions are equally accurate. The LKR-OMP(1000) model discussed above achieves an accuracy of 0.74 kcal/mol. The relevance of using OMP for the selection of the reference environments instead of simpler algorithms is illustrated by comparing the accuracy of LKR-OMP(1000) and a Ridge Regression based on the same number of environments chosen by FPS. The LKR-OMP(1000) model (referred simply as "LKR" for the rest of the article) is significantly more accurate than the LKR based on FPS model, which additionally highlights the importance of selecting atomic environments tailored for the specific target property.

While the performance of the NN is slightly superior to the LKR in the training step, it must be noted that the latter model is only trained on energy data, whereas the NN uses both energies and forces (*i.e.*, $3 \cdot N_{atoms}$ times more training scalar quantities). However, the mean absolute error for each individual dipeptide is consistently below 1 kcal/mol for both models. The learning rates of both approaches, defined as the error as a function of the number of training structures, are also both very similar and characterized by a decay exponent of -0.2

on a logarithmic scale.

The OMP algorithm provides insightful complementary information, allowing to identify which atomic environment is associated with the largest difficulties in the learning procedure. This feature is unique to OMP, and not available for standard kernel or NN based approaches that do not rely on supervised sparsity methods. In particular, OMP identifies that only a few of the 39'000 atomic environments (as low as 300) are sufficient to reach the accuracy threshold (1 kcal/mol) for the predictions of the dipeptides atomization energies. The OMP selection within LKR-OMP 1000 is 45.1% C, 2.9% H, 18.6% O, 28.9% N and 4.5% S atoms. For the sake of comparison, the atomic composition of the pool of dipeptide training structures is 29.5% C, 53% H, 8.5% O, 9% N, 0.3% S atoms. Evidently, the optimal reference atomic environments selected by OMP do not follow the same atomic distribution as in the overall pool of structures. OMP does not only find an adequate percentage of atom types, but it picks also the most tailored atomic environments for the target property. In contrast, the FPS selection with the same enforced atomic distribution as OMP (FPS+ 1000) is not sufficient to achieve a MAE as low as OMP (Figure 4.2d). In fact, on average 3 times more atomic environments are needed for FPS+ to match OMP. This is further demonstrated by the 2D t-SNE (t-Stochastic Neighbour Embedding[138]) projection (Figure 4.4) of the training atomic environments (constructed using the aSLATM representation as input data for the t-SNE).

The first two rows of t-SNE maps are color-coded based on the average "atomic Kernel Representation Score" (aKRS), *i.e.*, the average value of the kernel similarity between the training atomic environments and the selected reference ($\langle aKRS \rangle_j = \frac{1}{N_a} \sum_{i \in \text{ref.}} K(a_j, a_i)$, where j represents the index of an environment in the training data and i runs over the N_a selected reference environments of each atom type). The score is computed for the reference environments selected by OMP and FPS+. This score, bound between zero and one, shows how well an atomic environment is represented by the selected reference environments. The most striking differences between OMP and FPS+ is in the selection of the oxygen and hydrogen atomic environments, whereas carbon, nitrogen and sulfur are treated very similarly. In other words, the assumption behind the usage of FPS (the larger the variability in the reference environment, the higher the accuracy), is correct for carbon, nitrogen and sulfur, but not for hydrogen and oxygen. The oxygen maps are formed by one large smooth cluster, which represents the amide-bond oxygen atoms [O(a)], and two smaller regions regrouping the carboxylate [O(b)] and hydroxyl [O(c)] oxygen atoms respectively. In comparison with FPS, OMP is placing more emphasis on the amide oxygens and the carboxylate groups but much less on the oxygen in the hydroxyl groups. For the hydrogen atoms, the large number of isolated clusters in the t-SNE is indicative of a large variability in the hydrogen environments, which could intuitively suggest that a high number of hydrogen reference atoms are necessary to get an accurate model. Yet, OMP only selects 2.9% of them. This result reinforces that the choice of tailored environments is the key to achieving a more robust regression model. Interestingly, OMP favors carbon-bonded hydrogen atoms lying in the central cluster, rather than polar hydrogens (*e.g.*, in a O-H bond). Since the model is constructed to capture the variations of the potential energy as a function of the molecule structural changes, the selection of more carbon-bonded

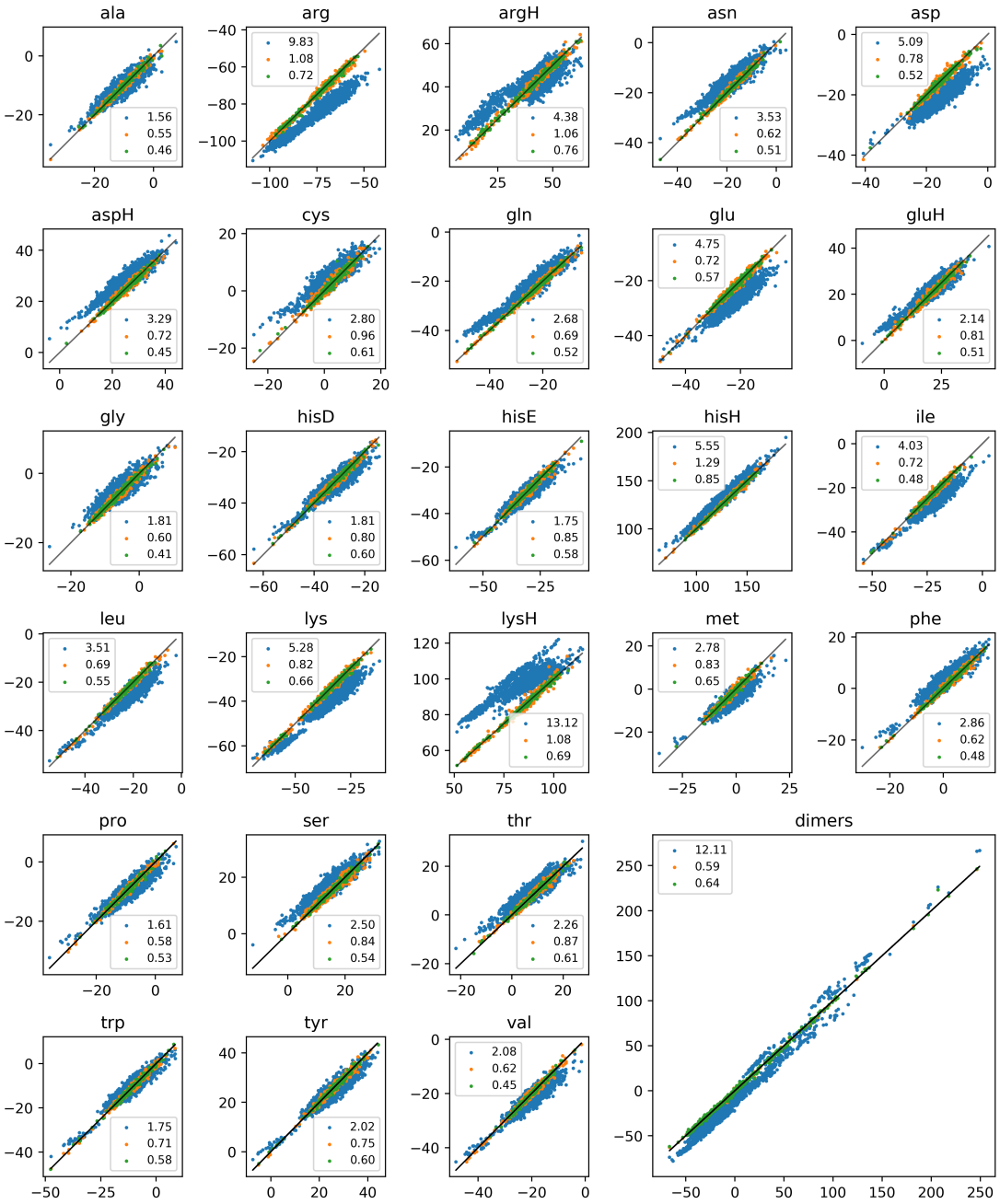


Figure 4.3 – Energy predictions on the test set (y axis) v.s. target PBE (x axis). In blue is DFTB-D3BJ without ML correction, in orange DFTB-D3BJ + LKR and in green DFTB-D3BJ + NN. The number in the legend is the MAE between the predicted energies and the real values.

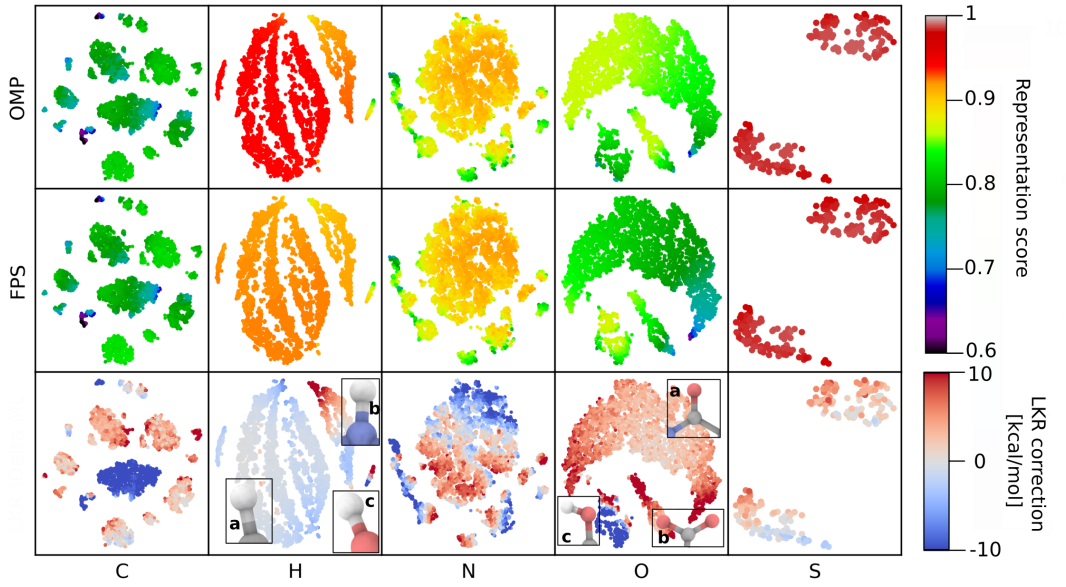


Figure 4.4 – t-SNE maps constructed with the aSLATM representation as input for each atom type. Each point represent an atomic environment in the training data. The color code in the first two rows shows how well represented the training environments are by the reference environments chosen by OMP and FPS+. As representation score we use the average "atomic Kernel Representation Score" (aKRS), the average value of the kernel similarity between each of the training atomic environments and the selected reference environments of the same atom type. The color code in the last row shows the LKR correction on each of the training atomic environments.

hydrogens than any other type has to be attributed to the higher conformational variability of the environments surrounding a C-H bond.

Another useful analysis of how the model behaves involves comparing the choice of atomic environments by OMP with the magnitude of the ML correction in terms of atomic-contributions (last row of Figure 4.4). While one might expect a direct relationship between the atomic selection and the magnitude of the ML atomic error, this intuition is actually incorrect. In fact, a large DFTB error for a given atom type does not necessarily imply that the learning process would be improved by including more atom-environments of the same type. This is especially true if the electronic nature of the DFTB error is uniform across all the conformation available in the training set. This lack of correlation is evident while looking at the bottom panels of Figure 4.4. The DFTB errors are the largest for the hydroxyl functional groups [H(c) and O(c) in the figure], while only a small portion of carbonyl or amide oxygen atoms are characterized by a similar errors of opposite sign. This trend is not reflected in the optimal OMP selection of reference atomic environments. Similarly, the most problematic carbon atoms (in terms of ML errors) are the oxygen-bonded carbons, which include the amide functions (the center cluster), as well as the carbons of the terminal guanidino group of arginine ($HNC(NH_2)_2$). However, OMP does not place special attention to these environments when selecting the

best reference carbons. Nitrogen behaves similarly to carbon. The central cluster is the most well described, which is representative of C-NH-C nitrogens (mainly present in the amide bonds), while the outer clusters, including terminal amines ($-NH_2$), the proline rings and guanidino groups, are less sampled. In contrast to other atoms, the ML correction for nitrogen has similar magnitude in all the clusters. An interactive application to visualize and explore this data is available at https://atomic-environments-dipeptides.herokuapp.com, built with the Molecular Explorer Software.[286]

4.6 Extrapolation

The local nature of the two ML potentials can, in principle, be used to make predictions for any system containing no chemical species other than C, H, O, N, and S, although high accuracy is expected only for local environments similar to those present in the training set, *i.e.*, in peptide chains or oligopeptides. Here we demonstrate the transferability of the two models by exploring the potential and free energy landscapes of the Phe-Gly-Phe tripeptide. The Phe-Gly-Phe tripeptide (in neutral form) is an appealing target to test the transferability of the ML models as it is one of the most suitable chemical systems to model non-covalent interactions in proteins.[287] Additionally, this tripeptide is not an adequate target by existing force fields, which are typically parametrized either for the capped peptides or for charged forms. The Phe-Gly-Phe tripeptide is in the gas phase without capping, and contains a combination of neutral NH2 and COOH groups that are not stable in solution. As a result, many Force Fields (AMOEBA, AMBER) do not accept it as input, or alternatively they generate unstable dynamics (GAFF).

To assess the quality of the extrapolated energies, we compile two datasets made of 1'000 Phe-Gly-Phe structures subdivided into 900/100 subsets illustrative of the conformational landscape explored at 300 K and 0 K respectively at both the baseline and target levels. The first set corresponds to a random selection of 1'000 structures taken from the converged T-RE (300 K) ensembles computed at the DFTB level. Out of these 1'000 structures, 100 are optimized at the same DFTB level (*i.e.*, 0 K static optimization). The second set is a random selection of 1'000 structures taken from the 300 K sampling at the DFTB + LKR-1000 level (see the next section) out of which 100 of them are optimized with PBE.

The most striking difference when comparing the error distributions of DFTB and the ML corrected versions (respectively the blue and orange/green histograms, Figure 4.5a) is the transition from a bimodal Gaussian distribution to a simple Normal distribution centered at zero. The two peaks correspond to the DFTB energies of conformers generated using DFTB as underlying potential [DFTB//DFTB, overstabilized] and to the DFTB energies of conformers generated using a different potential [DFTB//PBE, understabilized. What is perhaps more significant is the fact that the ML corrections not only remove the systematic error (*i.e.*, they center the distribution in zero), but also treat the two sets of structures on an equal footing. The transition from a bimodal to a single Gaussian distribution upon application of the

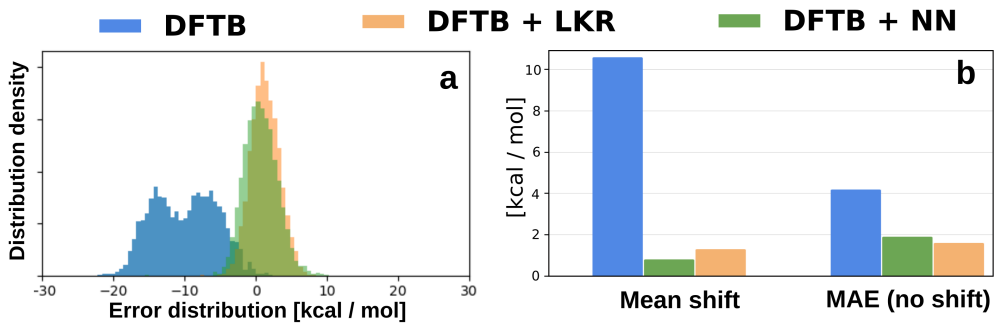


Figure 4.5 – **a)** Histogram of prediction errors made on the tripeptide test set. **b)** Bar plots with the mean shifts of the error distributions and their MAE after being centered.

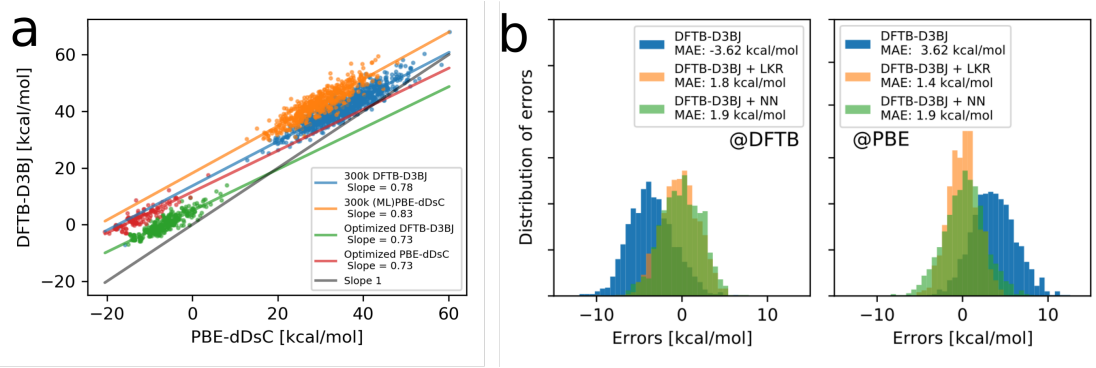


Figure 4.6 – (a) Regression slopes between atomization energies of DFTB-D3BJ and PBE for different test sets of the tripeptide. (b) Histograms of shifted errors (systematic deviations in the atomization energy have been removed) made on the tripeptide test set. The data is divided according to the potential that was used to generate them: (left) DFTB, (right) PBE.

ML-corrections reveals that the DFTB sampled conformational space (*i.e.*, the set of visited structures) would be energetically disjoint from the reference sampled space at PBE if we had to drive a dynamics using a DFTB potential. The ML-corrections allows concluding that this separation is spurious and that the DFTB structures from the PBE perspective [PBE//DFTB] are not peculiar. Interestingly, the slope between DFTB and PBE energies for all the tripeptide test structures combined is 0.96, which would suggest that systematic flattening of the PES by DFTB is not observed in this case. However, the correlation between DFT and DFTB breaks down when considering the 300 K and 0 K conformations separately (in a clear example of the Simpson's paradox[288]), where the typical behavior of DFTB is recovered (slopes: 0.78 at 300 K and 0.73 at 0 K, see again Figure 4.6). Finite temperature effects offset the energies of the 300 K ensemble with respect to the 0 K, so that joint distribution seem to correlate better with the DFT values.

The ML corrections (Figure 4.5 orange and green data) overcome all the issues present in the uncorrected DFTB potential. First, the mean bonding energy shift is reduced from 10.6 kcal/mol to 1.3 kcal/mol by the LKR and to 0.8 kcal/mol by the NN model (see Figure 4.5b). This error

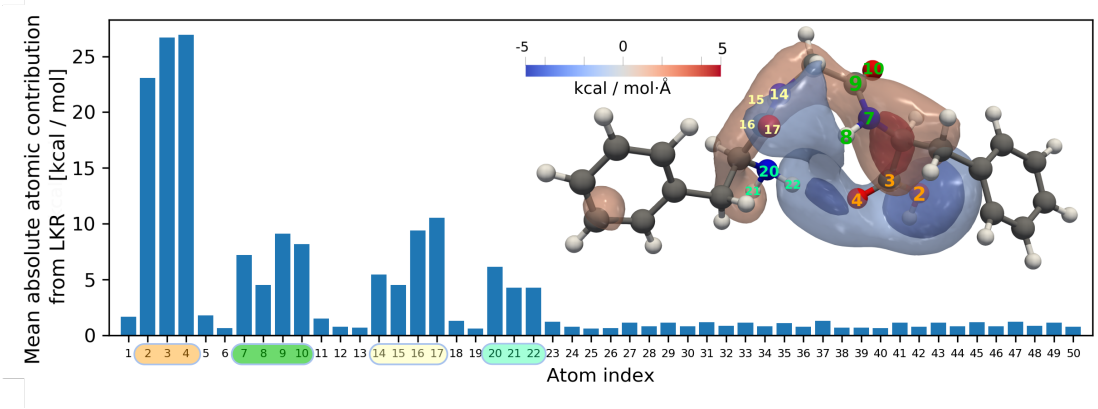


Figure 4.7 – Histogram with the mean absolute atomic contribution to the LKR corrections for the tripeptide for the 2'000 test structures. The figure includes a particular conformation of the tripeptide with isosurfaces of a scalar field representing the localization of the ML correction. The scalar field was generated with the LKR atomic corrections to the energy for that structure, convoluted with the atomic positions and a Gaussian filter of width 1 Å. The isosurfaces correspond to the isovalues -5, -2, +2 and +5.

does not influence the conformational sampling of the molecule, as a constant shift in energy does not alter the relative probability of the conformers. Nevertheless, a decreased error is beneficial when comparing the electronic energies of different molecules. Most importantly, the average absolute deviation from the mean is reduced from 4.2 kcal/mol to 1.6 kcal/mol by the LKR model and to 1.9 kcal/mol by the NN (see Figure 4.5b). All the errors of the LKR model are below 8 kcal/mol, while the NN predictions on the tripeptide present two outliers of -15 and +26 kcal/mol. Additionally, the regression slope between the predictions and the target energies is also corrected to 0.99 for all the sets. These results are crucial since the standard deviation and the regression slope are the most important quantities for conformational sampling. Even a slight deviation from 1 in the regression slope causes significant changes in the resulting free energy surfaces. In particular, the observed regression slope between DFTB and PBE at 300K (0.75) is roughly equivalent to perform sampling with a temperature 1.33 times higher (*e.g.*, 400 K instead of 300 K). At the same time, outliers can lead to unstable dynamics and alter the results of sampling simulations.

Overall, while the NN model performs slightly better on the dipeptide test structures, the LKR provides a more robust extrapolation (lower MAE, less outliers) for the Phe-Gly-Phe tripeptide. It must be noted that the superior stability of LKR is not a consequence of exclusively using energetical data for the training. On the contrary, an equivalent NN trained only with energies shows much poorer transferability and scalability capabilities.

As shown in the previous section, the atomic decomposition of the ML correction naturally provides a measure of the error localization in the molecule. To visualize the error for the tripeptide, we constructed a scalar field using atomic centered Gaussian functions scaled such as to match the LKR atomic predictions (see Figure 4.7). Using this procedure, it is possible to

construct a real-space map highlighting the regions of the tripeptide where the DFTB potential deviates from the PBE reference. An example of these critical regions is identifiable between the oxygen and hydrogen atom forming an intramolecular hydrogen bond (*e.g.*, between atoms 4 and 22 in Figure 4.7). In Figure 4.4, we have shown that the hydrogens bound to an oxygen or a nitrogen are the most difficult to describe at the DFTB level. Figure 4.7 shows the understabilization of the hydrogen bond between the NH_2 and the CO by DFTB, which is corrected by our models. However, this particular example does not imply that all hydrogen bonds are poorly described and in a systematic manner. For example, equivalent figures show that the OCO–H bond in the dipeptide of aspartate is actually overstabilized by DFTB, while the CO–HN in the protonated histidine is understabilized. These inconsistencies have been shown to arise at the DFTB level due to a poor description of short-range electrostatic and polarization interactions arising from the use of a minimal valence basis.[289] While several empirical corrections to DFTB and more generally to semi-empirical methods have been proposed,[289–293] the use of the D3H5 correction (the last of such corrections DFTB-D3H5[294]) does not change the performance of DFTB on the dipeptide set significantly.

Furthermore, the analysis reported in Figure 4.7 shows that the description of the hydrogen-bond interactions are not the only limitation of DFTB. More generally, the highest absolute ML corrections appears whenever the bond between two atoms is polarized, such as in the region of the terminal carboxylic acid (atoms 2, 3 and 4 in Figure 4.7) and the amide moiety of the peptide bond (atoms 7, 8, 9 and 10 in Figure 4.7). In contrast to existing corrections, which are not meant to improve the description of these polarized bonds, the ML models guarantee by construction an equally accurate description for all the regions.

4.7 Free energy surface of tripeptides

Having assessed the robustness of the ML models by evaluating the accuracy of the energy predictions on Phe-Gly-Phe tripeptide conformations and by providing comparisons with uncorrected DFTB, this section goes further and applies the ML corrections to sample the free-energy landscape of the tripeptide in gas phase. As described in the Computational Details section, for the LKR model we use the resH-RE approach for a 300 K canonical sampling of the tripeptide generated with DFTB as a *reservoir* to accelerate the DFTB+LKR sampling without the need for high temperatures or bias potentials. We used T-RE for the exploration of the tripeptide at the DFTB level as it is an unbiased sampling method, which is preferable for high dimensional systems when adequate CV for metadynamics simulations are unknown. Additionally, the resH-RE simulations require a reservoir following a canonical distribution, which is not directly obtainable from biased methods such as metadynamics. The resH-RE approach is especially convenient as the ML model was trained on structural data generated at 300 K and could thus become unstable at high temperatures. Figure 4.8 shows the set of characteristic collective variables (CVs) chosen to analyze the free-energy landscape. The set of CVs includes all the Ramachandran dihedral angles as well as the distance between the benzene rings at each end of the chain. We generated 2D free energy surfaces using all the

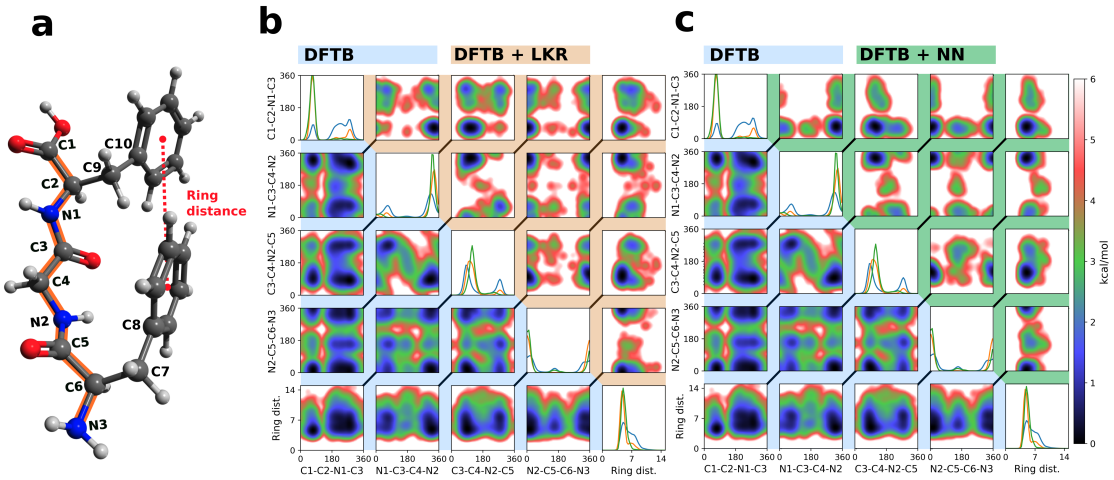


Figure 4.8 – (a) Tripeptide Phe-Gly-Phe with highlighted atoms used for the collective variables in the analysis of the sampling simulations. (b) & (c) Grids with 2D free energy landscapes for each pair of the selected collective variables. The lower diagonals contains results from T-RE simulations using DFTB. The upper diagonals contains the results of the resH-RE simulations using DFTB + LKR (b) and DFTB + NN (c). In the diagonal are the probability distributions of each collective variable for DFTB(blue), DFTB + LKR(orange), and DFTB + NN(green).

pairs of CVs from the samplings obtained using DFTB (lower diagonal of Figure 4.8 b/c) and DFTB+LKR (upper diagonal of Figure 4.8b) in order to provide a complete view of the results. The C4-N2-C5-C6 and C2-N1-C3-C4 dihedral angles were excluded from the plot because their values remain constant throughout the sampling.

To provide a complementary view, we further sample the same tripeptide free energy surfaces using the committee of NN models (upper diagonal of Figure 4.8c). We exploit the availability of forces to perform well-tempered metadynamics simulations, and make use of the ability to assess the uncertainties in the predicted corrections to smoothly fall back onto the DFTB baseline when the NN predictions become uncertain. This suppresses instabilities in the dynamics due to unphysical NN corrections in areas of the PES, which are underrepresented in the training data. Using the analysis of the DFTB-based sampling as guidance, the metadynamics are biased in the six-dimensional CV space spanned by the four peptide bond dihedrals (see Figure 4.8), with the additional dihedrals N1-C2-C9-C10 and C5-C6-C7-C8 to account for the ring distance (for further information see the Computational Details section).

The comparison between the DFTB T-RE results (lower triangular portions of Figure 4.8) and the results of the ML potentials (upper triangular portions of Figure 4.8) shows the effects of correcting the flat PES on the final free energy landscape. In addition to increasing the free energy barriers, translated in very low populations in basin transition areas, the ML corrections dramatically affect the relative stability of the different basins, altering the qualitative dynamic behavior of the tripeptide at 300 K. These effects can be equally observed in both the sampling based on LKR and NN. The results obtained show good agreement. The single

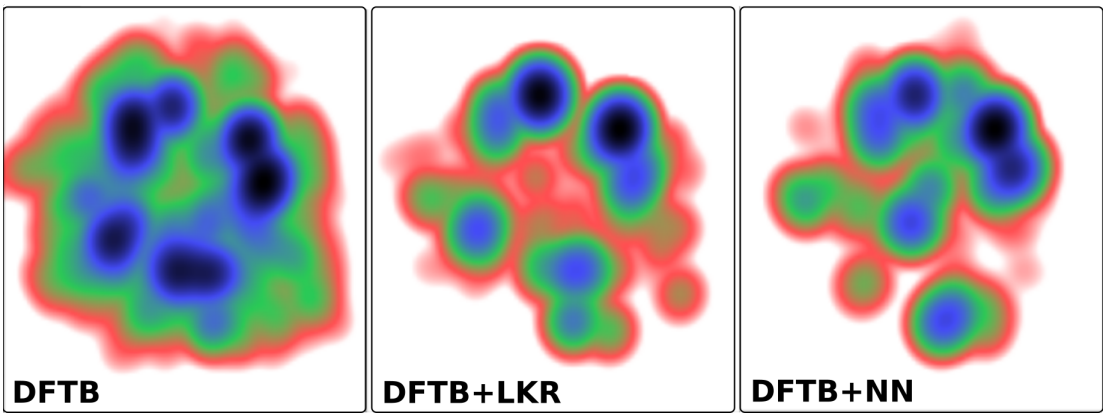


Figure 4.9 – Sketchmap computed with DFTB (left) DFTB + LKR (middle) and DFTB + NN (right) sampling at 300 K using the selected CVs from Figure 4.8.

CV populations are nearly identical, and the lowest free energy minima are unequivocally determined. However, some disagreement in the free energy surfaces obtained by sampling using the two ML frameworks can be observed for the higher-energy portions of the free energy surfaces. Given the highly non-trivial nature of this exercise, it is not easy to pinpoint the source of the discrepancy. The entanglement between uncertainties arising from (i) finite statistics and (ii) possible discrepancies of the ML models difficults the analysis of their relative weight. Nevertheless, it is clear that both ML-corrected frameworks predict a much sharper variation of the free energy compared with DFTB that instead predicts a very smooth landscape as a function of the dihedral angles. This qualitative difference is also clearly visible in a 2D Sketchmap [60,61] projection (Figure 4.9), which indicates that the more diffuse structural distribution at DFTB is a direct consequence of the flatness of the associated PES.

As a final note, it is important to stress that the generation of converged statistics using the target potential (PBE) would have been computationally unfeasible. Alternatively, a comparison with experimental results would requires the addition of solvent effects, which is outside the scope of this work.

4.8 Conclusion

In this work we introduced LKR-OMP, a local kernel regression model which exploits the supervised sparsity algorithm OMP, and compared its performance along with that of a Behler-Parrinello neural network. LKR-OMP benefits from the straightforward training of kernel methods, combining it with the scalability and transferability of models based on neural networks. We juxtapose the two approaches by applying them to the challenging task of learning the PES of oligopeptides at the PBE-dDsC level, using the semiempirical DFTB-D3(BJ) potential as a baseline and training on a combination of dipeptide structures and dimers of small organic fragments.

To achieve comparable computational cost between sparse kernel regression and NNs, it is essential to select carefully the most representative environments. We show, both by comparing the final model accuracy and by combining the representation score with a 2D projection of the local atomic environments, that selection methods relying exclusively structural information, such as FPS or CUR, are not always optimal, and that substantial improvements can be achieved with the supervised strategy adopted in the LKR-OMP scheme.

Using only energies for training, the LRK-OMP model achieves an accuracy and transferability compared to that of the NN-based model, that also uses forces to optimize its parameters. Thanks to the atom-centered construction of the ML correction, we can reveal the origin of the DFTB-D3(BJ) error relative to DFT, interpret in terms of chemical and atomic patterns and demonstrate the relevance of relying upon a correction based on non-linear regression techniques.

As a final demonstration of the possibilities brought about by the use of ML corrections of the PES, we use them in combination with an enhanced sampling approach to explore the conformational energy landscape of the tripeptide Phe-Gly-Phe at an effective PBE-dDsC level. We use two different sampling strategies: resH-RE for LKR-OMP, which at present does not provide easy access to energy derivatives, and ATLAS metadynamics for the NN potential, that instead does. The free energy landscapes obtained with the two frameworks are consistent with each other, and show striking differences compared to the uncorrected baseline potential. This provides another example of the exaggerated smoothness of the DFTB potentials and highlights the dire need to make the accuracy of higher electronic structure levels accessible to the size and time scale that are necessary for free energy computations. In this respect, the fact that ML corrections have now become a mature, trustworthy approach to achieve this goal – with entirely different frameworks achieving comparable accuracy and efficiency is very encouraging. The LKR-OMP model, in particular, offers a good compromise in terms of data-intensiveness, computational cost, generality and accuracy, in addition to providing unique analytical insight into the model performance.

5 Reaction-based machine learning representations for predicting the enantioselectivity of organocatalysts

This chapter is based on the following publication:

S. Gallarati, R. Fabregat, R. Laplaza, S. Bhattacharjee, M. D. Wodrich, C. Corminboeuf, Reaction-based machine learning representations for predicting the enantioselectivity of organocatalysts, *Chem. Sci.* **2021**, 12, 687.

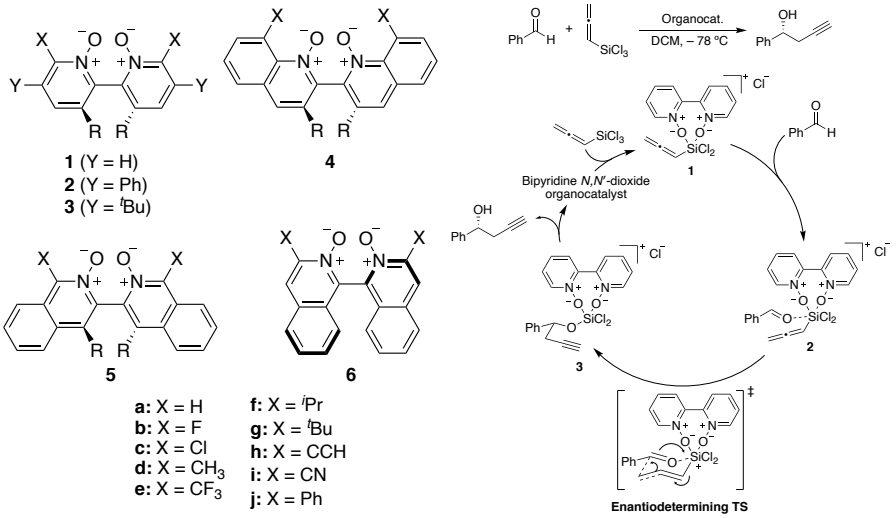
5.1 Introduction

Society's growing need for pharmaceuticals, agricultural chemicals, and materials requires a continuous push in the development of asymmetric catalytic methods.[295,296] In particular, enantioselective organocatalysis has emerged as a powerful strategy for the stereocontrolled assembly of structurally diverse molecules[297–299] with constant effort placed in making chemical transformations more selective, efficient, or generally applicable.[300] Although the computational design of highly selective catalysts has long been viewed as a "Holy Grail" in chemistry,[301,302] it is generally still more efficient to experimentally screen a range of potential organocatalysts for a given reaction than to assess their performance in silico.[303] That is because e.e. (enantiomeric excess) values, estimated as the ratio between the competitive reaction rates leading to the two enantiomeric products,[304] are relatively computationally expensive and challenging to predict accurately with standard electronic structure computations. The energy difference between the transition states (TSs) leading to the major and minor enantiomers can be quite small (< 2 kcal mol^{-1}) and multiple diastereomeric transition states, stemming from the large conformational space of flexible organocatalysts, can yield the same enantiomer.[301,305] As the relation between rate constants and computed selectivity is exponential, minor errors in computed energies can lead to major errors in stereochemical outcome prediction. These factors pose a monumental challenge for traditional quantum mechanical (QM) methods, in terms of both accuracy and cost,[306,307] especially if many conformers and substrate-catalyst combinations have to be computed. While the intrinsic error of the quantum chemical level is often addressed in comprehensive benchmark studies,[304,308–311] automated toolkits,[312,313] such as AARON[314] and CatVS,[315] have been developed

to streamline the tedious and error-prone task of optimising hundreds of thermodynamically accessible stereocontrolling transition states. Starting from user-defined libraries, multiple conformations and configurations of TS structures are located and optimised. Although such accelerated prediction of selectivity is enticing for the prospect of computational catalyst design,[303] the applicability of QM-based tools such as AARON remains limited either by the cost of quantum mechanical computations, which quickly becomes prohibitive, or by the inherent difficulty of locating all transition state structures. On the other hand, tools using QM-derived molecular mechanics force fields (Q2MM), like CatVS, require the development of an MM force field for each new reaction type considered, a major limitation to their widespread application.[305]

An alternative approach pioneered by Norrby[316] and Pradhan[317] and popularised by Sigman and co-workers is to fit experimental reaction outcomes to physical organic molecular descriptors.[318–320] The difference in free energies at the stereocontrolling transition states can be expressed as a polynomial function of global or local steric and electronic parameters, such as Sterimol values, natural bond orbital charges, IR frequencies, HOMO/LUMO energies, and polarisabilities.[321–327] In principle, the resulting statistical model allows for extrapolation to out-of-sample examples,[328,329] however, like all QSSR-type methods,[330] such multivariate linear regression is not easily transferable and most suitable only for closely related analogues of the training set, given that a set of appropriate molecular descriptors must be redefined for every new regression.[315]

Nonlinear regression models (*e.g.*, artificial neural networks, random forest, Gaussian processes, support vector machines)[331] have demonstrated the potential to overcome some of the previous limits in catalyst screening and constitute an alternative to multilinear regressions with parameters derived from chemical knowledge and mechanistic hypotheses (e.g., Hammett constants, Tolman cone angles, percent buried volume, vibrational frequencies, pKa values).[9,52,332–336] Recently, the organic synthetic community has exploited these artificial intelligence-based approaches for predicting $\Delta\Delta G^{\ddagger}$, e.e., the activation energy, the product distribution, or the yield of (asymmetric) catalytic reactions. These models rely on the identification of a large set of system-specific molecular descriptors (e.g., physical organic descriptors like Charton or Sterimol values, NBO charges, NMR chemical shifts, bond distances and angles, HOMO-LUMO gaps, local electro/nucleophilicity, or RDKit descriptors[337]) used as the input from which an algorithm can "learn" while being "supervised" by the reaction outcome (output, *i.e.* $\Delta\Delta G\ddagger$, e.e., or yield).[338–353] While the reaction outcome is often obtained from experiment (*i.e.*, phenomenological models), alternatives based on computed data are highly valuable as well.[354–358] Indeed, so-called quantum (or atomistic) ML models, which map a three-dimensional molecular structure (called molecular representations, *e.g.* CM,[29] SLATM,[119] SOAP[125]) to a representative target computed quantum chemically, constitute an appealing complementary strategy owing to its broad applicability and dependence on the laws of physics.[53,119,359] While these approaches provide a favourable combination of efficiency, scalability, accuracy, and transferability for predicting energetic and more complex molecular properties,[53] identifying enantioselective organocatalysts requires precise predic-



Scheme 5.1 – Library of axially chiral bipyridine N,N'-dioxide organocatalysts. R = H or Me (left). Adapted from Doney and coworkers.[361] Catalytic cycle for the propargylation of benzaldehyde with allenyltrichlorosilane, showing the rate-limiting and stereocontrolling transition state (right). Adapted from Rooks and coworkers.[362]

tions of the relative energy barriers for the stereocontrolling transition states, a target currently beyond their accuracy. Recently, SOAP features of isolated reactants were used to train a machine learning classifier and predict transition state barriers of regioselective arene C-H functionalization. In this work, a large number of molecular fingerprints were combined with the SOAP features to improve the regression, and the resulting model was outperformed in out-of-sample predictions by a random forest model using chemical descriptors with physical organic basis (PhysOrg).[360]

Here, we provide a stepwise route to improve such QML approaches to reach sufficient accuracy for subtle properties such as those associated with asymmetric catalysis (*i.e.*, e.e.). This objective is achieved by rationally designing a reaction-based representation (vide infra) that is a more faithful fingerprint of the enantiodetermining TS energy. The performance of the approach is demonstrated through accurately predicting the DFT-computed enantiomeric excess of Lewis base-catalysed propargylation reactions directly from the structure of the catalytic cycle intermediates. Unlike other ML models trained on (absolute) experimental e.e.'s,[328,329] our model is able to predict the absolute configuration of the excess product, because it is trained on the activation energy of the enantiodetermining step for each pair of enantiomers (pro-(R) and pro-(S) intermediates) independently.

5.2 Methods

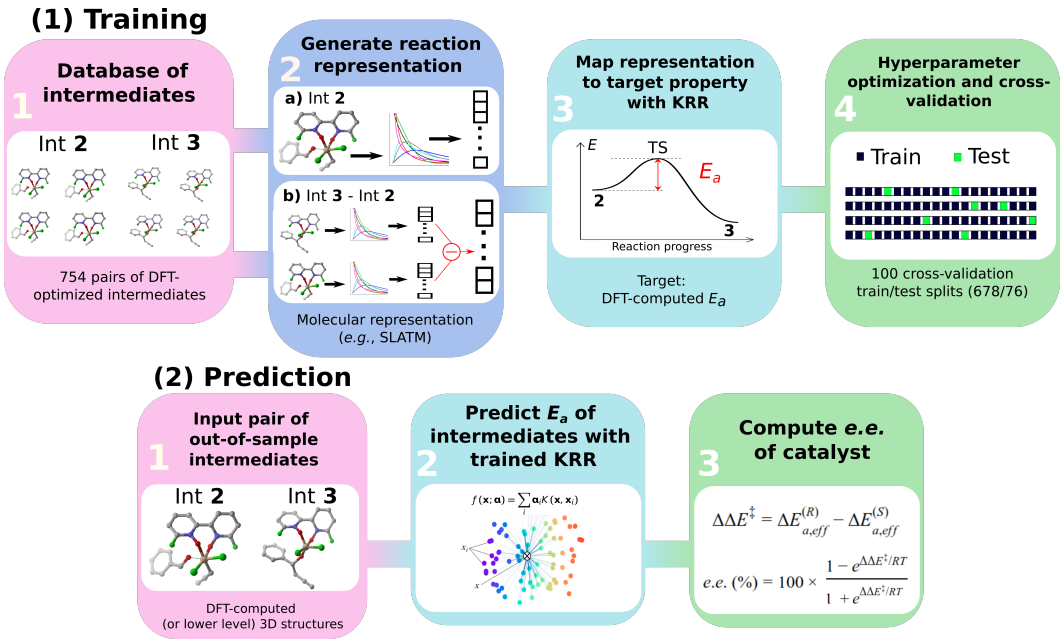
5.2.1 Reaction and Organocatalysts Database

Asymmetric allylations[363–366] and propargylations[367] of aromatic aldehydes are key C-C bond forming transformations, providing access to optically enriched homoallylic and homopropargylic alcohols, respectively, which serve as valuable building blocks for the synthesis of complex chiral molecules.[368] Catalysts that are selective for allylations are generally not highly stereoselective for propargylations, which has led to a dearth of stereoselective propargylation catalysts.[362,369–372] Tools to screen dozens of allylation catalysts to find promising candidates for propargylation reactions are therefore highly valuable.[303] To this end, Wheeler and coworkers have investigated[364] Lewis base organocatalysts (Scheme 5.1)[361] and used the computational toolkit AARON20 to build a database of 760 stereocontrolling transition states to predict their enantioselectivity in the propargylation of benzaldehyde (Scheme 5.1).[308,361,373] Large databases of kinetic data for asymmetric catalysis generated in silico are scarce.[354] Therefore, this library constitutes an ideal training and validation set for the development of an atomistic ML model with reaction-based representations capable of predicting the e.e. of organocatalysts readily from the structures of intermediates. Note that the workflow presented below would improve the ML performance independently of the size of the training data. The target of the ML model is the DFT-computed relative forward activation energy (E_a, *i.e.*, the energy difference between the TS and the preceding intermediate) associated with each of the 10 (R)- or (S)-ligand arrangements of the enantiodetermining TS in Scheme 5.1 for the 76 catalysts in Scheme 5.1 (11 catalysts of type 1, 16 of type 2, 15 of type 3, 11 of type 4, 13 of type 5, and 10 catalysts of type 6), yielding a total of 754 E_a values (the intermediates 1f-S-bp2-2, 3e-R-bp1-3, 3e-S-bp1-3, and 3j-S-bp2-3 could not be converged, therefore the corresponding enantiomeric TS structures were removed from the original database of 760 TSs). e.e. values are computed from E_a (vide infra), thus accurate predictions of E_a lead to accurate e.e. predictions.

5.2.2 General ML Workflow

The general workflow exploited and improved herein relies on a physics-based ML model for the prediction of the e.e. of the asymmetric catalytic reactions, as illustrated in Scheme 5.1 and described hereafter. It comprises two parts: part (1) is a training procedure that relies on the following steps:

1. Database construction: a library of 3D geometries and energies of catalytic cycle intermediates is curated. Here, the structures of 754 pairs of intermediates 2 and 3 are optimised with DFT (see the next section) and used to train the ML model. As shown in our previous work,[52] accurate geometries are not necessarily needed as inputs for atomistic ML models; thus, rough-coordinate estimates (*e.g.*, obtained directly from SMILES strings) or low-cost DFTB structures could potentially be used to generate



Scheme 5.2 – Graphical overview of the workflow used to build an atomistic ML model for e.e. prediction.

suitable molecular representations.

2. Generation of molecular representations: information intrinsically contained within the 3D structure of each intermediate is transformed into a suitable molecular representation. Here we build different variants based on the Spectral London and Axilrod-Teller-Muto (SLATM) [119] representation. SLATM is composed of two- and three-body potentials, which are derived from the atomic coordinates, and contain most of the relevant information to predict molecular properties. [55,125,129,187,374–377]

3. Training of the model: input representations are mapped onto the corresponding target values (E_a, computed at the DFT level, see the next section) using Kernel Ridge Regression (KRR) [21] with a Gaussian kernel. Note that even if target values based on DFT are used here to train the ML model, the strategy proposed hereafter is expected to perform equally well on experimental or more accurate quantum chemical data.

4. Hyperparameter optimisation and cross-validation: the full dataset is split randomly 100 times into 90/10 training/test sets (678/76 datapoints) to optimise the KRR hyperparameters and obtain the learning curves.

In part (2), the trained ML model is used to predict the activation energy of out-of-sample organocatalysts. The model requires as input the 3D structures of 2 and 3 and delivers the corresponding E_a value. Using the energy of 2 as reference, the relative energies of the enantiodetermining (R)- and (S)-TSs can be calculated, and the e.e. of the catalyst under investigation computed (vide infra).

5.3 Computational Details

5.3.1 Quantum Chemistry

Catalytic cycle intermediates 2 and 3 were optimised at the B97-D/TZV(2p,2d) level of theory,[378–380] accounting for solvent effects (dichloromethane, ϵ = 8.93) using the polarizable continuum model (PCM)[381–383] with Gaussian16[384] in analogy with the study by Wheeler and co-workers.[361] Density fitting techniques were used throughout. The structures of 1508 intermediates were obtained via intrinsic reaction coordinate calculations (IRC)[385] from the TS database curated by Wheeler et al.[385] 754 target E_a values (11 catalysts of type 1, 16 type 2, 15 of type 3, 11 of type 4, 13 of type 5, and 10 of type 6, each in 5 distinct pro-(R) and pro-(S) ligand arrangements) were computed (relative to the lowest-lying intermediate 2 ligand arrangement) at the same level, which was shown to provide the best compromise between accurate predictions of low-lying TS energies and stereoselectivities for allylation and propargylation reactions.[308] The e.e. values were not predicted from Gibbs free energy barriers, but rather from relative energy barriers (*i.e.*, electronic energies plus solvation free energies), since they have been found to be more reliable than those based on either relative enthalpies or free energy barriers for this reaction.[308] The symbol E_a was therefore used to indicate the energy (electronic plus solvation) difference between the TS and the preceding intermediate. For each C2-symmetric catalyst (Scheme 5.1),[304] distinct ligand arrangements around a hexacoordinate Si centre are possible (BP1-5, (R)- and (S)-).[362,372,373] Since each of these can lead to thermodynamically accessible reaction pathways, and the stereoselectivity is largely a consequence of which ligand arrangement is low-lying for a particular catalyst, all diastereomeric TSs were considered viable and the e.e. calculated from a Boltzmann weighting of the relative energy barriers.[361] In equations 1-3, $\Delta E_{a,\mathrm{eff}}$ is the relative Boltzmann-weighted activation energy of each (R)- or (S)-species, $\Delta\Delta E^{\ddagger}$ is the difference between the (R)- and (S)-Boltzmann-weighted activation energies, R is the ideal gas constant, and T is the propargylation reaction temperature (195 K).

5.3.2 Machine Learning

The Python package QML[220] was used to construct standard SLATM representations. Feature selection and the construction of the reaction-based representations $\mathrm{SLATM_{DIFF}}$ and $\mathrm{SLATM_{DIFF+}}$ were done using the Python package Scikit-learn.[137] To generate the learning curves and the e.e. predictions, a cross-validation scheme was used with 100 different 90/10 training/test sets (678/76). The KRR hyperparameters (the width of the Gaussian kernel σ and the ridge regularization λ) were optimised for each train/test split, systematically obtaining essentially the same results for each split (see the SI). From the 100 train/test splits, the E_a of each intermediate pair (2 and 3) was predicted approximately 10 times; these test predictions were then averaged to obtain one final prediction. The standard deviation from the ≈10 test predictions were used to generate the error bars. The final average prediction of the E_a value was used to calculate the Boltzmann-weighted $\Delta E_{a,\mathrm{eff}}$ values (eq. 1) and the $\Delta\Delta E^{\ddagger}$ of each (R)-

and (S)-pair (eq. 2), and so the e.e. value of each organocatalyst (eq. 3). The out-of-sample predictions were done with the same $SLATM_{DIFF+}$ models trained in the cross-validation scheme. Additionally, out-of-sample predictions were done re-training the model on the entire dataset, although this did not lead to noticeable improvement. While simpler representations (*e.g.*, CM,[29] BoB[80]) were tested, SLATM performs significantly better.

5.4 Results and Discussion

5.4.1 Molecular representations

The key step of the workflow presented above is generating a molecular representation, which is mapped onto the target value (*i.e.*, the activation energy E_a) and used as a fingerprint of the enantiodetermining TS. Representations can be constructed from single molecules and more recently as "ensemble representations": instead of associating one fixed configuration of atoms to a single-point geometry energetic target value, information from multiple structures can be combined to generate a representation for an ensemble property, such as the free energy of solvation (ΔG_{sol}).[83] This has recently been achieved by calculating the ensemble average of the FCHL19 representations[31,82] of a set of configurational snapshots obtained through MD sampling.[83] Here, we propose an alternative approach that goes beyond standard QML representations (*i.e.*, KRR using one given gas-phase geometry) by describing the chemical transformation occurring during the enantiodetermining step of an asymmetric reaction through the comparison of the representations of the two catalytic cycle intermediates preceding and following the stereocontrolling TS. This allows us to generate a "reaction-based" representation, which can be closely mapped to the activation energy of the enantiodetermining step, as discussed later. We rely on "dissimilarity" plots as a diagnostic tool to determine whether a particular representation can adequately characterize the stereocontrolling step. By dissimilarity plots, we refer to histograms of the Euclidean distance between any two representations vs. the difference in their target property, which in this case is E_a. For a particular representation to be effective, small distances between structures must correspond to small differences between target properties, as the Euclidean distance is used to measure the similarity of two molecular representations. Similar plots have previously been exploited to analyse the behaviour of molecular representations,[74,125] but only parenthetically. Here we highlight their importance as a fundamental analytical tool to understand the performance of molecular representations in kernel methods for asymmetric catalysis and demonstrate their utility for constructing reliable ML models.

Before discussing our proposed representation variants, we report in Figure 5.1a the performance of the standard SLATM representation using the structure of a single intermediate (*e.g.*, 2). Due to the structural similarities between 2 and the enantiodetermining TS (in both, the Si atom has 6 coordination sites occupied, whereas only the coordination number is only 5 or 4 in intermediate 3), intermediate 2 was first chosen to construct the input representation. The learning curve for the prediction of E_a using SLATM (blue) of intermediate 2 (denoted

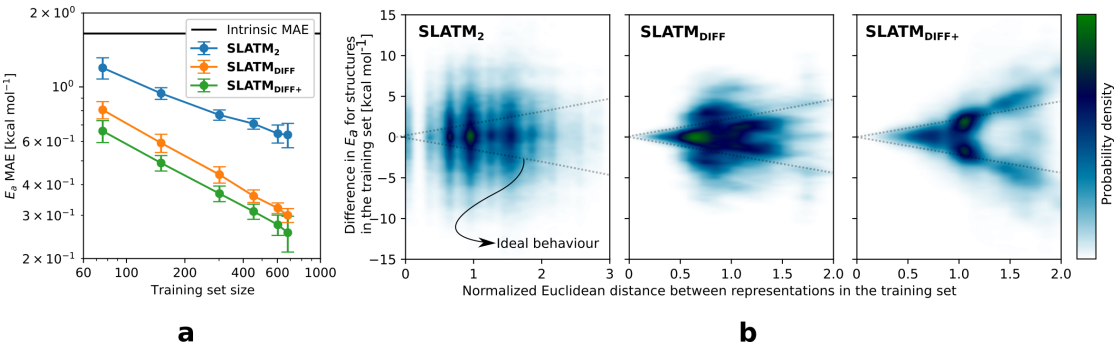


Figure 5.1 – (a) Learning curves with MAE in test sets predictions of E_a for the three approaches discussed. The error bars correspond to the standard deviations and are computed from the results of 100 different random train/test splits. (b) Dissimilarity plots *i.e.*, difference in target values (E_a) vs. Euclidean distance between representations for each pair of points in the dataset (the Euclidean distances have been divided by the average distance between points). When the difference in E_a values tends to zero, the corresponding points should lie in the area delimited by the two dotted straight lines (ideal behaviour).

$SLATM_2$) reaches a Mean Absolute Error (MAE) of 0.54 ± 0.06 kcal mol^{-1} for the prediction of E_a with 90% of the data used for training (*i.e.*, 680 structures). Considering the exponential relationship between relative activation energies and e.e. values, which implies a dramatic propagation of errors, the accuracy of this approach is insufficient. This is further demonstrated in Figure 5.2, which shows the correlation between the predicted and reference $\Delta\Delta E^{\ddagger}$ values (MAE = 0.96 kcal mol^{-1}), and in Figure 5.5, where the e.e. values obtained from $SLATM_2$ are compared to the reference quantities: the large number of red-coloured cells indicates large deviations between ML-predicted and DFT-computed e.e. values. The rather poor mapping between $SLATM_2$ and the E_a of the stereocontrolling step (associated with the key 2 - 3 transition state) is evident from the visual inspection of Figure 5.5, where the large number of red-coloured cells associated with catalysts bearing substituents a, d, e, g, f and j indicates inaccurate predictions of e.e. values, and from the analysis of the corresponding dissimilarity plot in Figure 5.1b (left). In the latter, the large scattering of points lying outside the area delimited by the dotted lines, particularly when the Euclidean distance tends to zero, means that two different structures might be considered equal by the kernel (distance ≈ 0) albeit leading to very different E_a values. Thus, the shape of the dissimilarity plot of $SLATM_2$ deviates considerably from ideal one, indicated by the dotted straight lines.[125] Note that the MAE for E_a increases up to 0.77 ± 0.05 kcal mol^{-1} if starting from the SLATM representation of 3, the intermediate following the enantiodetermining step in the catalytic cycle (Scheme 5.1). The higher accuracy achieved using the representation of 2 vs. 3 could be attributed to the reaction step being exergonic and, according to the Hammond Postulate,[386] the enantiodetermining TS resembling 2 more closely. In any case, neither the structure of 2 or 3 provide sufficiently good fingerprints of E_a on their own.

Unlike other intrinsic molecular properties that depend on the structure of a single molecule,[83] enantioselectivity is determined by electronic and/or steric effects stabilising or destabilising

one enantiomeric TS to a greater or lesser degree than the other. In that sense, it is to be expected that our target accuracy for E_a, well below 1 kcal mol^{-1}, cannot be reached using only one structure that does not adequately describe the stereocontrolling transition state as an input. To improve the model performance, an alternative representation is constructed by comparing the representations of both intermediates. Knowing that neither the structure of 2 or 3 are uniquely related to the corresponding activation energies, we can generate such a "reaction-based" representation that draws information from both structures, subtracting the global SLATM of 2 from 3. This is reminiscent of binary reaction fingerprints (obtained by subtracting the products from reactants in RDKit[337] fingerprints), which reflect changes in molecular features over reaction processes.[358] The resulting representation (denoted $SLATM_{DIFF}$) accounts for the differences between the two intermediates and is thus more sensitive to the structural changes occurring during the enantiodetermining step. By subtracting "reactant" from "product", the global features that do not change during the catalytic cycle step are eliminated from the representation, and the structural changes between intermediates are highlighted. In this way, we obtain a more faithful representation of the reaction step, which corresponds to a more unique fingerprint of E_a. Although the construction of $SLATM_{DIFF}$ requires the SLATM representations of both intermediates (2 and 3), the computational cost associated with its generation is negligible.

As depicted in the dissimilarity plot (Figure 5.1b, middle), the reaction-based representation ($SLATM_{DIFF}$) is significantly better than $SLATM_2$: the difference in E_a values tends to zero as the Euclidean distance between their representations tends to zero. In line with this observation, the learning curve (shown by the orange line in Figure 5.1a) is significantly improved. The MAE of $SLATM_{DIFF}$ is reduced to 0.31 ± 0.2 kcal mol^{-1}, roughly 50% better than $SLATM_2$ and up to 60% better than that of $SLATM_3$ using 90% of the data for training (*i.e.*, 680 structures) in the train/test splits of the cross-validation scheme. Given the rationality of the approach leading to the construction of $SLATM_{DIFF}$, its gain in accuracy is encouraging. As shown in Figure 5.2 and Figure 5.5, the halved MAE leads to a very notable improvement in the prediction of e.e. values. Nevertheless, we note again that very small errors in E_a are amplified when e.e. values are calculated, and therefore even a small accuracy gain can be significant. The high probability density of normalised Euclidean distances between 0.5 and 0.75 seen in Figure 5.1b (middle, $SLATM_{DIFF}$) indicates that the shape adopted by the dissimilarity histogram of $SLATM_{DIFF}$ is not yet ideal, and that further improvement is possible. To achieve higher accuracy, we focus on improving the shape of this dissimilarity plot. Notice that in our ML model, the Euclidean distance is used as a measure of similarity between representations. This means that features with high variance (*i.e.*, that change the most between molecules) dominate the notion of similarity, as they contribute the most to the Euclidean distance between representations. By feature, we mean each of the terms in the molecular representation, which, for SLATM, consist of two- (London dispersion) and three- (Axilrod-Teller-Muto) body potentials computed on groups of atoms closer than a certain cut-off (here, 4.8 Å). The results of these potentials are averaged over their atom-type sets (*e.g.*, all C-C interactions for the two-body terms, all the C-C-C for the three-body terms), which are then concatenated to generate the SLATM vector.

The size of the SLATM representation depends on the existing atom-type sets in the database. Given that our dataset contains the elements C, H, O, N, F, Cl and Si, the total number of features of the SLATM representations is 27827.

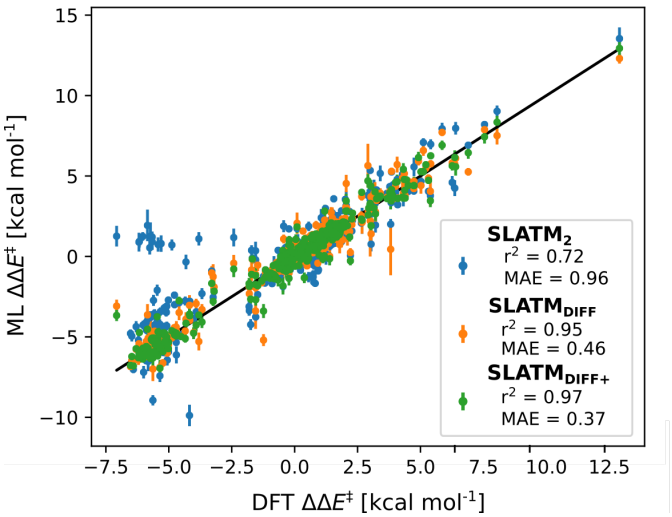


Figure 5.2 – Predictions of $\Delta\Delta E^{\ddagger}$ vs. DFT reference for the three approaches discussed. Mean Absolute Errors (MAE) are reported in kcal mol^{-1}. These predictions are obtained by averaging the predictions obtained from the cross-validation scheme with 100 different random train/test splits. The error bars indicate the standard deviation of ML $\Delta\Delta E^{\ddagger}$, derived from the standard deviations in the E_a prediction of the 100 different random train/test splits.

In $SLATM_{DIFF}$, features with high variance dominate the notion of similarity, measured through the Euclidean distance. However, we are using SLATM to predict a property that is very different from the single-molecule properties for which it was originally designed. Consequently, features with high variance in SLATM are not necessarily the most important fingerprints of E_a. In general, applications that use molecular representations are highly sensitive to the metric used to compute similarities, as common choices like the euclidean distance use a biased notion of similarity that gives more or less importance to features depending on their predefined variances. This is not always a critical problem, because molecular representations are designed with rational principles, but in general the metric used and therefore the idea of similarity should depend on each specific application, as different molecular properties depend on different molecular features. However, so far only unsupervised feature selection and dimensionality reduction methods are commonly used in the computational chemistry community, while their supervised counterparts, much more adequate, are largely unknown.

The inadequacy of the euclidean distance to assess the similarity between standard molecular representations is clearly illustrated in the following example, where we use SLATM to regress the atomization energy of the QM9 database[49,250] containing 134k stable small organic molecules made of up to 9 CONF heavy atoms. If we compute the variances of SLATM for the QM9 database and compare them to the mutual information between features and the

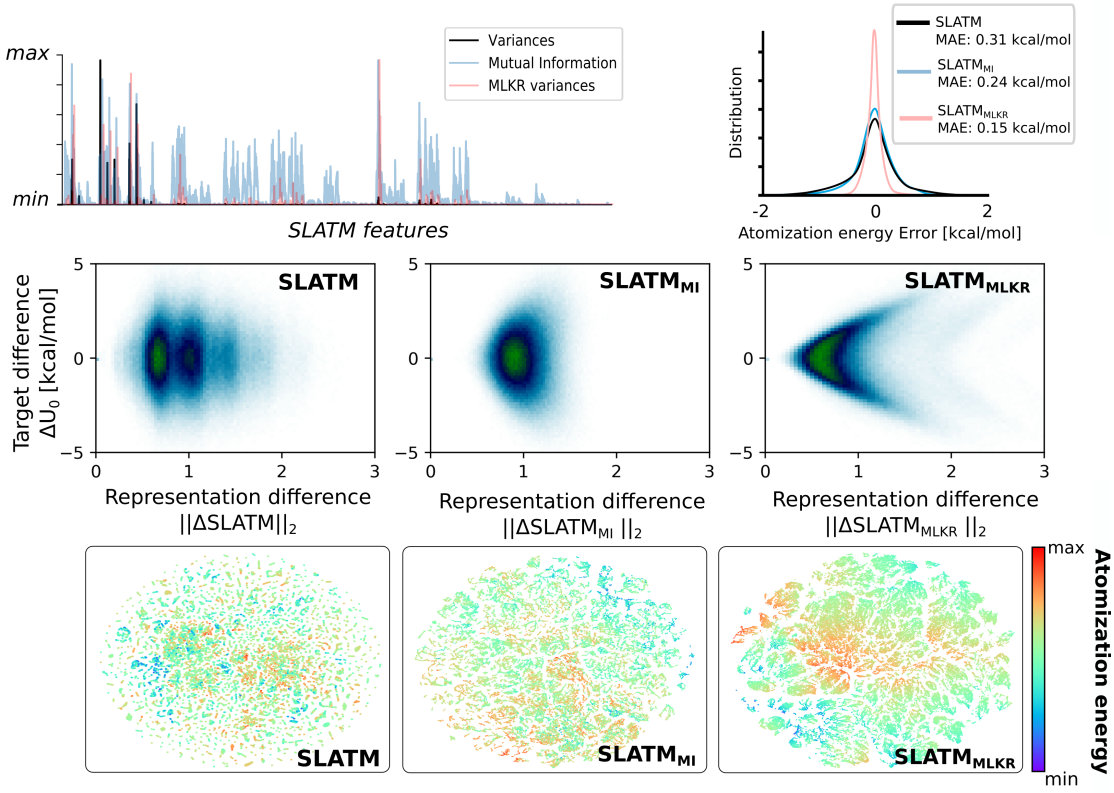


Figure 5.3 – Variance, MI, and obtained MLKR variance of the SLATM features for the QM9 database (top left). Achieved test MAE for each of the representations (top right). Dissimilarity plots for each of the representations (middle). 2D t-SNE projections of the QM9 databse using each of the representations (bottom).

target[387] (the atomization energy) we do not observe a correlation (see Figure 5.3 top left). This indicates that the standard design of SLATM is not optimal even for atomization energies, one of the main targets for which it was designed. Using supervised feature selection, we are able to obtain a naive representation, SLATM_{MI}, that is built using the 1000 features with the highest mutual information. Moving from SLATM to SLATM_{MI} already reduces the test error of atomization energy prediction by 20% of a simple KRR model with 10000 training data points (see Figure 5.3 top right). Subsequent removal of the correlated features from SLATM_{MI}, followed by application of Metric Learning for Kernel Regression (MLKR) leads to a representation, $\text{SLATM}_{\text{MLKR}}$, containing only 516 features that reduces the original error by 50%. The dissimilarity plots in the middle of Figure 5.3 corresponding to each representation clearly show the successive improvement. The obtained feature space is not only useful for improving supervised machine learning models, but also for any task that uses distances and similarities, such as dimensionality reduction methods. As the notion of similarity becomes better adapted to the target at hand, the results of unsupervised learning become more meaningful. This can be clearly observed in the lower plots of Figure 5.3, which show 2D t-SNE projections of the QM9 database for each of the representations (all generated with the

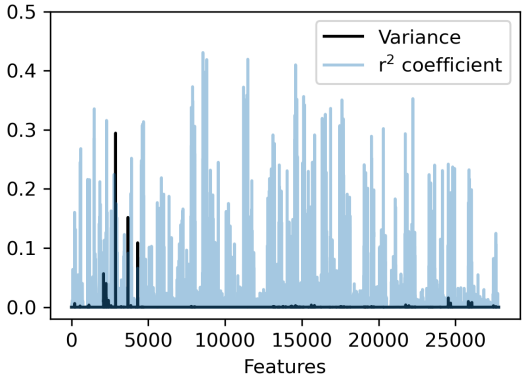


Figure 5.4 – Variance and correlation coefficient with the target value for each of the 27827 features of the $SLATM_{DIFF}$ representation in the dataset.

same t-SNE hyperparameters). Successively improved feature spaces generate 2D projections having better correlation with the atomization energies.

It must be stressed that the results of this example were not achieved by increasing the information encoded in SLATM, but rather by filtering and adapting it to the target at hand. Therefore, categorizing different molecular representations by the amount of information they encode [126] can be misleading, as maximizing the encoded information is not equivalent to maximizing usability. In this sense the pursuit of an ideal, universal and immutable molecular representation can be detrimental, if no additional measures are taken into account. This issue has been largely ignored until now, even after studies have shown how the popular representation SOAP could be optimized for different scenarios. [105] Molecular representations are still envisioned as immutable quantities, that should be equally accurate in any application or situation. This fact has been a source of the current convoluted understanding of molecular representations, where many fundamentally different possibilities exist, yet guidelines for choosing one over another, apart from trial and error specific to each application, remain absent.

In pursuit of the best possible fingerprint for the current application, we follow the previous example and procede to optimize the $SLATM_{DIFF}$ to construct a similarity measure adapted to the activation energy. First, we assign importance scores to each feature and attempt to focus on the most relevant ones. In this case, the linear correlation coefficient (r2) between each feature and the target property is used as an estimate of the importance of the different terms in the representation. The results, presented in Figure 5.4, show that in $SLATM_{DIFF}$ there are only a few high-variance features, while the computed importance scores are spread over many other features that have relatively small variances. Simply put, the variances in the features of the $SLATM_{DIFF}$ representation are not well correlated with their real importance in this application. Based on this observation, an improved representation, labelled $SLATM_{DIFF+}$, is generated by selecting only the Nf most important features of $SLATM_{DIFF}$ (specifically, Nf =

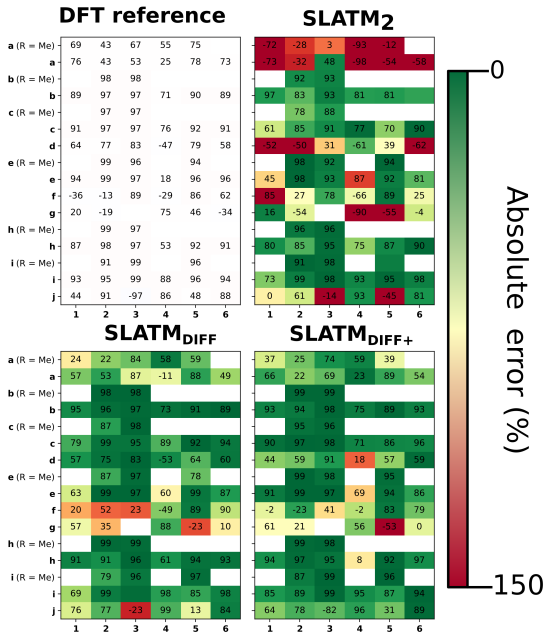


Figure 5.5 – e.e. values obtained from DFT computations (top left) and from the ML predictions of E_a using the three approaches discussed. These predictions are obtained by averaging the predictions obtained from the cross-validation scheme with 100 different random train/test splits. Cells are coloured according to their accuracy with respect to the reference, ranging from dark green (best) to dark red (worst). Positive e.e. values correspond to excess (R)-alcohol formation, negative values to excess (S)-alcohol formation.

500) and discarding the rest. This feature selection was done using only the training data at each train/test split of the cross-validation step, as otherwise it could lead to severe overfitting. Nevertheless, the importance scores were consistent across the cross-validation splits thanks to the robustness of the linear regressions. An improved relationship between representation and target distances (Figure 5.1b, right) is obtained with the $SLATM_{DIFF+}$ representation, in spite of its reduced size. This simple feature selection leads to a noticeable improvement in accuracy, with a cross-validated MAE of 0.25 ± 0.4 kcal mol^{-1} (see the green curve in Figure 5.1a). Using the $SLATM_{DIFF+}$ representation, the resulting cross-validated correlation coefficients for the difference between (R)- and (S)-activation energies ($\Delta\Delta E^{\ddagger}$, Figure 5.2) in the test set are greatly improved (r2 > 0.95). The quality of our fitted model far supersedes previously reported approaches. Good qualitative and even quantitative agreement is achieved between predicted and reference e.e. values computed using the test data splits from the cross-validation runs (Figure 5.5). Since linear correlation constitutes a very limited notion of relevance, other methods, such as nonlinear mutual information criteria,[387] were tested as feature importance estimators, but the resulting models showed similar or even worse performance. Similarly, methods based on metric learning such as MLKR did not lead to any improvement in this case, as the high dimensionality of the problem and little data available led to severe overfitting. Ceriotti et al.[251] suggested the use of principal covariates

regression (PCovR) to solve similar issues. PCovR is a supervised feature selection method that interpolates between principal component analysis (PCA) and linear regression. Herein, because the variance of the features is completely unrelated to the importance scores, the addition of PCA would not offer any advantage. Nevertheless, these findings highlight the importance of adapting molecular representations to the application at hand, while still preserving the overall generality of the approach.

5.4.2 Chemical Insight on Asymmetric Propargylation Catalysts

The ML model is able to reproduce the main trends in e.e. observed across the different catalysts from the 100 different random train/test splits (Figure 5.5, top left table). For example, using $SLATM_{DIFF+}$ (Figure 5.5, bottom right table), which gives the best predictions with respect to the reference data, catalysts built on scaffold 4 (Scheme 5.1) are revealed to be outliers, yielding e.e.'s that are significantly different to those obtained with other scaffolds, for a given substituent a-j. This is due to the different placement of the substituent X on the organocatalysts' scaffold. Excluding 4, the effect of different substituents on the e.e. is qualitatively the same across all scaffolds, with the exception of f (iPr) and j (Ph). The introduction of a phenyl group on the organocatalysts' scaffold leads to highly varied e.e. values, from -97 (S) to 91 (R). This variation, which is due to the presence of favourable π-stacking and CH/π interactions stabilising some (S)-TSs and degrading the enantioselectivity,[361] is nicely captured by $SLATM_{DIFF+}$. Overall, the high enantioselectivity displayed by (most) catalysts in the library can be attributed to the favourable electrostatic interaction between the formyl C-H of benzaldehyde and one of the chlorines bound to Si, which is present in the lowest-lying (R)-ligand arrangement, and absent in the (S)-structures.[361]

In their computational screening with AARON,[361] Wheeler and co-workers identified derivatives of 6 as promising candidates for propargylation reactions. However, these catalysts are difficult to synthesize stereoselectively.[370,388] Recently, Malkov et al. reported the synthesis of a set of terpene-derived atropisomeric bipyridine N,N′-dioxides 7 (Figure 5.6) as easily-separated diastereoisomers.[389] Aromatically-substituted catalysts 7j and 7k were shown to be highly active and selective (e.e. of 96 and 97, respectively); additionally, the TS structures for 7 were computationally shown to be nearly identical to the corresponding substituted forms of 6.[389] Prompted by these results, we decided to test the ML model with $SLATM_{DIFF+}$ to predict the activation energy of the 10 distinct ligand arrangements afforded by 7j and 7k. The out-of-sample results are shown in Figure 5.6. Despite scaffold 7 and substituent k not being in the original training set, excellent correlation between predicted and reference E_a values is obtained (r2 = 0.97). Thus, the enantioselectivity of these out-of-sample catalysts is qualitatively reproduced, despite not achieving exact quantitative agreement between DFT and ML predicted $\Delta\Delta E^{\ddagger}$ values (1.2 and 1.3 for 7j and 7k, respectively, vs. 0.2 and 0.5 kcal mol^{-1}). In summary, we provide a logical route to improve atomistic ML methods for enantioselectivity prediction of asymmetric catalytic reactions, which are limited by both the required accuracy and the small amount of data generally available. Firstly, the intermediates associated with

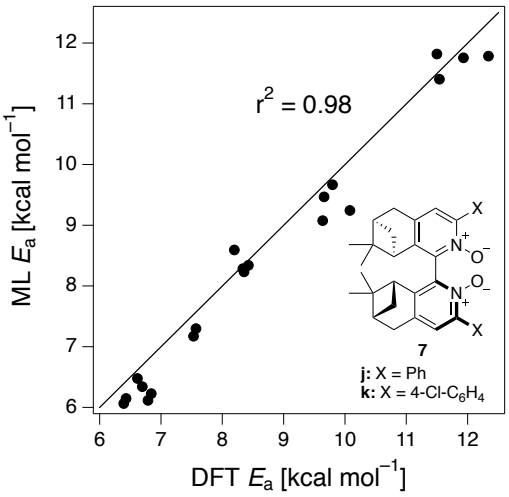


Figure 5.6 – Out-of-sample predictions on terpene-derived atropisomeric organocatalysts 7j and 7k. 10 distinct TSs were computed for each catalyst (BP1-5, (R)- and (S)-). The error bars are the standard deviation of the 100 predictions from each trained model from the cross-validation scheme.

the enantiodetermining step (2 and 3 in Scheme 5.1) must be identified, and their SLATM representations generated. Secondly, using the difference between the two SLATM representations ($SLATM_{DIFF}$) as input, a set of features that map the activation energy accurately can be obtained. Finally, feature engineering can be used to improve $SLATM_{DIFF}$, keeping only the most relevant features that relate to the target property. The results show that the ML workflow presented herein is able to accurately predict enantioselectivity from the molecular structures of catalytic cycle intermediates.

5.5 Conclusions

In this work, we have developed an atomistic machine learning model to predict the DFT-computed e.e. of Lewis base-catalysed propargylation reactions (Scheme 5.1). The use of dissimilarity plots allowed us to rationally develop and progressively improve a reaction-based representation that can be adequately mapped onto the activation energy of the stereocontrolling step. We identified two fundamental limitations of many standard physics-based molecular representations for subtle catalytic properties. First, we have shown that neither the structure of the preceding nor that of the following catalytic cycle intermediate is a fine fingerprint of the energy of the stereocontrolling transition state. This issue can be circumvented by using a reaction-based molecular representation derived from both structures. Finally, we have demonstrated how feature selection can be used to fine-tune this representation.

The resulting model can accurately predict the DFT-computed enantioselectivity of asymmetric propargylations from the structure of catalytic cycle intermediates. Thus, it constitutes

a valuable tool to quickly identify potentially selective propargylation organocatalysts. By design, the model is well-balanced between computational cost, generality and accuracy. It is easy to implement for a wide region of chemical space and seamlessly compatible with experimental (*e.g.*, X-ray structures of stable intermediates) and computational data alike. Our results prove that semi-quantitative predictions of e.e. values in asymmetric catalysis can be achieved by accurately predicting E_a. We conclude that atomistic ML models with adequately tailored molecular representations can be a practical and accurate alternative to both traditional quantum chemical computations of relative rate constants and multivariate linear regression with physical organic molecular descriptors. The stepwise improvement to the model described in this work opens the door to more complex reaction-based and catalytic cycle-based representations. Indeed, ensemble representations, which were recently introduced for properties very sensitive to conformational freedom, such as the free energy of solvation ΔGsol,[83] are a promising path to go beyond the single structure-to-property paradigm and allow for further generalisation, once combined with the approach discussed herein. Such methodologies will be explored in future work for the accurate screening of enantioselective catalysts in asymmetric reactions.

6 General Conclusions and Outlook

6.1 Conclusions

The applications of machine learning in computational chemistry stand at a thrilling stage of development. The increasingly available data, powered by technological innovations such as GPU accelerated quantum chemistry and high throughoutput experiments, represents an extremely fertile playground where to test the innovations in the field of statistical learning. As a result, new data-fuelled prediction and analytical tools are providing unprecedented exploratory power of the chemical space, which is reshaping the methodological paradigms of the field. The work presented in this thesis is part of this evolution and aims at broadening the domain of applicability of machine learning algorithms that rely on similarity measures, among which are some of the most useful supervised and unsupervised approaches. The techniques and chemical situations presented in this work are part of a larger effort to develop novel statistical tools that process the ever-growing amount of chemical data, and belong to a broader family of computational methods to model increasingly complex molecular systems.

In the first part of this thesis we introduced the fundamental concepts and methodologies required for the developments presented in the rest of the thesis. We highlighted the importance of machine learning potentials in quantum chemistry applications, specifically to perform free energy computations with *ab initio* accuracy. This was demonstrated through the computation of free energy landscapes at the CCSD(T)/CBS level of theory of two flexible systems that are dictated by a subtle interplay between enthalpic contributions and conformational entropy. We showed how this could only be achieved thanks to the combination of two elements. On the right hand, kernel-based potentials, which use physics-based molecular representations to achieve extreme data-efficiency. On the left hand, Hres-RE, a novel enhanced sampling algorithm that bypasses the limitations of ML potentials and critically decreases convergence time by recycling the information of reservoirs with canonically distributed conformers. This combination was made only possible through the development of MORESIM, a modular Python package that provides an environment to design and execute hand-crafted replica exchange simulations.

Despite their widespread use to predict molecular properties, kernel methods like the one used in **Chapter 3** suffer from serious drawbacks. On one side, they lack the transferability and scalability of methods based of Neural Networks. On the other side, their accuracy in molecular property prediction is conditioned to the prior existence of molecular representations that are adequate for the application at hand. In this thesis we showed how these limitations originate from the sub-optimal construction of the two key elements that define a kernel method: the set of reference elements and the definition of similarity. In **Chapters 4** and **5** we addressed each of the issues and unveiled that the fundamental source of both problems lies in the presence of redundant and irrelevant information in descriptive data. We tackled this problematic by leveraging the ability of supervised dimensionality reduction and metric learning tools to filter descriptive information and adapt it to specific target properties. We then demonstrated how this greatly improves the capabilities of similarity-based methods for applications in computational organic chemistry.

Specifically, in **Chapter 4** we address the underwhelming performance of kernel approaches to construct transferable ML-based potentials for systems with high chemical diversity. Prior to our work, local kernel models suffered from a poor choice of reference atomic environments due to the inappropriate use of unsupervised learning approaches to filter highly redundant molecular databases. This hindered their performance for systems with high chemical variety and restricted their use to amorphous and crystaline materials with few element species.[235] Our work has unveiled the two reasons that make unsupervised algorithms like FPS sub-optimal for this task. On one hand, they use dissimilarity measures in the input feature space to select the most diverse set of reference environments. This is inadequate as dissimilarity in the representation space is not necessarily correlated with dissimilarity in the target space, given that molecular representations are not adapted to each specific target. On the other hand, sampling the input space uniformly as done by FPS is inefficient when the variability of the target property is not distributed accordingly, but rather localized in certain areas. Δ-ML represents a major example of this case, as the difference between a baseline and the target property is typically concentrated in specific regions of the input space. We tackled this problematic by introducing the LKR-OMP model, which combines a local kernel projection with the sparse regression algorithm OMP. This results in a supervised dimensionality reduction algorithm that allows to select the optimal set of reference environments for the prediction of a specific property. The performance of LKR-OMP, trained on thermally distorted dipeptide conformers, is validated on the prediction of the potential energy surface of oligopeptides and compared with that of a state-of-the-art Behler-Parrinello neural network. The LKR-OMP shows equal or even superior performance to the NN model, but also provides unique analysis tools. The sparse reference environments learned by LKR-OMP, combined with a 2-dimensional manifold learned using unsupervised learning, allows to identify the atomic environments that are most problematic for the training. Moreover, by comparing the results with traditional unsupervised sparsity algorithms like Farthest Point Sampling (FPS), we are able to determine which specific atomic environments are treated differently by OMP and therefore are inadequately described for the application at hand.

In **Chapter 5** we addressed the second and more fundamental problematic of current ML approaches in computational chemistry (present not only in supervised kernel methods but in all unsupervised learning in general): a poor and rigid definition of similarity between atomic and chemical environments. We have demonstrated that the origin of this problem lies in the high quantity of redundant and irrelevant information present in current molecular representations. This pollutes standard similarity metrics like the Euclidean distance and results in artificial similarity scores containing user-biases, which can critically affect the performance of any similarity based algorithm. We highlighted the importance of dissimilarity plots as diagnostic tools to assess this problem and to compare the adequacy of different similarity metrics. We then showed how supervised metric learning and feature selection techniques can be used to filter the information in molecular representations to adapt the idea of molecular similarity to a specific application. These concepts were applied to develop an atomistic machine learning model that predicts the DFT-computed enantiomeric excess of Lewis base-catalysed propargylation reactions. Using dissimilarity plots we rationally developed and progressively improved a reaction-based representation that can be adequately mapped onto the activation energy of the stereocontrolling step. This methodology allowed us to show how semi-quantitative predictions of enantiomeric excess values in asymmetric catalysis can be achieved by accurately predicting activation energies.

The results presented in the previous paragraphs summarize the main conclusions of this work and set the stage for new compelling perspectives, as outlined in the following section.

6.2 Outlook

6.2.1 Metric learning in the chemical space

Rather than engineering new representations, our work presents the possibility to adapt them to different applications by finding new metrics, which offers alternative ways to evaluate their performance. So far, molecular representations have been compared based on how much information they contain, but we have showed that this is not necessarily a reliable measure of efficiency. Instead, molecular representations should be ranked based on how well they can be adapted to different targets through metric learning approaches. In addition, a careful analysis of the similarities and differences between metrics learned for different applications could reveal hidden trends and help to develop improved and more adaptable molecular representations. This leads to an important question left unanswered by this thesis, which is whether learned metrics can be shared or transferred among different applications. If so, this opens the door to transfer-learning for kernel methods, a concept usually reserved for Neural Network models. Transfer-learning refers to storing knowledge gained while solving one problem and applying it to a different but related problem. A clear example would be to learn metrics using approximated target values (*e.g.* DFTB) prior to the learning with the real target property (*e.g.* DFT). This would allow access to much larger datasets to learn metrics, which is specially valuable as the main problem of metric learning approaches is their tendency to

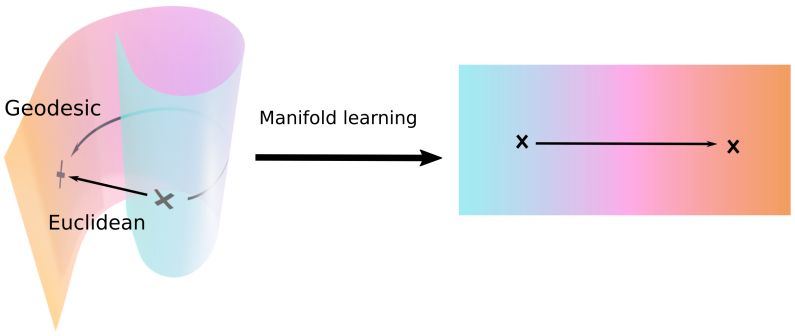


Figure 6.1 – Euclidean and geodesic distance before and after manifold learning.

overfit when data is scarce and the input feature space is large. Nevertheless, novel metric learning methods with sparsity and smoothness regularization [158,390] are continuously being developed and could contribute to alleviate this problem.

6.2.2 Semi-supervised kernel regression

Metric learning allows to linearly transform the space in order to give each feature the right importance and improve the notion of similarity between training elements. However, the obtained similarity is a function of the feature coordinates exclusively, and do not take into account how the data is distributed in space. Including the local structure of data in the comparison between elements could further improve the notion of similarity between elements. This is specially relevant for high-dimensional data, which is often distributed in convoluted manifolds with low effective dimension that can be accessed using manifold learning methods. In such case, the similarity between training elements can be better assessed by using geodesic distances in the manifold rather than global distance metrics like the Euclidean distance (see Figure 6.1). The special advantage of this methodology is that local structures can be captured using unlabelled data, which is generally much more available than the labelled counterpart. This type of approach, where the local structure of unlabelled data is used to aid supervised learning tasks is generally referred as semi-supervised learning (see Figure 6.2).

Semi-supervised learning is specially well-suited for chemical applications, where labelled data is often scarce but unlabelled data is generally available in large quantities. The learning of potential energy surfaces (PES) is a key example. Training ML models to fit a PES usually involves generating thermally distorted conformers using sampling methods with an approximate potential. The target potential is then computed in a small selection of conformers, which serves as training data, while the large majority are discarded. Instead, the whole data could be used to learn local manifold structures where the actual regression will finally take place.

Only few recent works have reported the use of semi-supervised learning along with chemical data, [391,392] although they are without exception used on deep learning frameworks. This is not

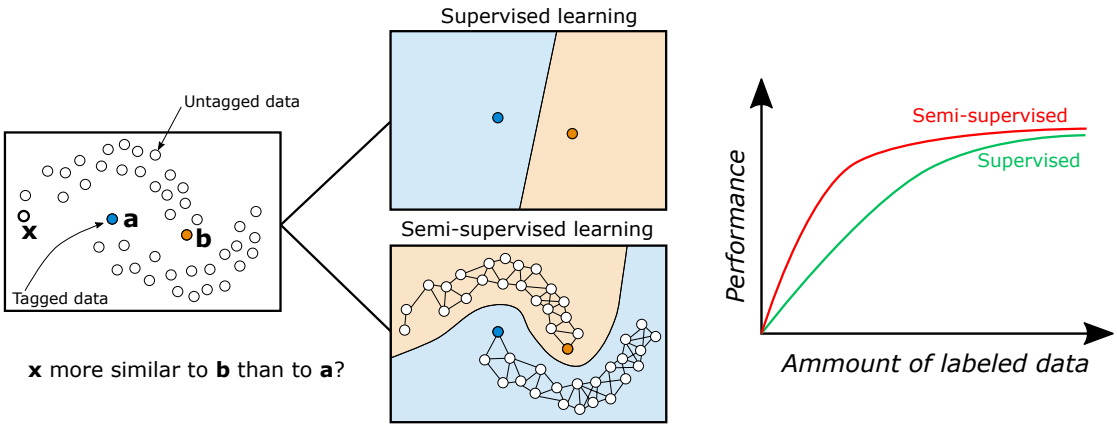


Figure 6.2 – Illustrative depiction of semi-supervised learning for classification. In this example, the distribution of the data in clusters can be used to infer the labels of unlabelled data using label propagation.[5]

a coincidence, as most semi-supervised regression models are based on deep learning. In fact, there are several recent breakthroughs on semi-supervised learning on deep neural networks that have significantly reduced the amounts of labelled data necessary for training in image classification frameworks,[5,393,394] although their applicability in other domains remains to be seen. Additionally, most of the unsupervised regression methods are designed for classification problems,[395] as it is generally easier to understand the relationship between data structure and target if the target variable is categorical rather than continuous. Nevertheless, approaches to include semi-supervised learning to kernel methods do exist.[396–398] A straightforward possibility is to use unsupervised learning dimensionality reduction like t-SNE prior to the kernel regression. We already saw in **Chapter 5** that after using Metric Learning for Kernel Regression (MLKR) the target property showed a very smooth evolution in the manifold learned by t-SNE. When coupled with metric learning, semi-supervised learning is a very promising direction to further boost the applicability of similarity-based regression models in chemical applications.

A Elemental supervised and unsupervised ML algorithms

This appendix contains a summary of the most common used supervised and unsupervised learning algorithms, with some tips on how and when to use them.

A.1 Unsupervised learning algorithms

A.1.1 Dimensionality reduction algorithms

Dimensionality reduction algorithms can be divided in linear methods based on matrix factorization or matrix decomposition techniques, and non-linear methods, often referred as manifold learning.

Linear dimensionality reduction: Matrix decomposition and factorization

Linear dimensionality reduction techniques are based on the approximation of a matrix $\mathbf{M}(n \times d)$ (in a data science setting n would be the number of samples and d the number of dimensions of the representation) as a product of two other matrices $\mathbf{U}(n \times f)$ and $\mathbf{V}(f \times d)$, where f would be the number of dimensions in the new reduced space (i.e. $\mathbf{M} \approx \mathbf{UV}$.

By minimizing the Frobenius norm of the "reconstruction error" ($||\mathbf{M} - \mathbf{UV}||$) we obtain the solution of Principal Component Analysis (PCA). From the solution of PCA we obtain a space of dimension f where the orthogonal basis are a linear combination of the original features, and that represents the directions of maximum variance in the original space of dimension d. Alternatively to the Frobenius norm, different cost functions can be defined to obtain solutions with particular characteristics, such as smoothness or sparsity.[399] For example, by forcing all the elements in $\mathbf{U}$ and $\mathbf{V}$ to be positive, we obtain what is known as Non-Negative Matrix Factorization, which has many uses when negative values are not meaningful, such as in signal decomposition and image reconstruction.[400]

The reconstruction error can be used to determine how many dimensions are necessarily for

the linear method to "explain" most of the variance in the matrix **M**. Typically, the reconstruction error drops very fast with an increasing number of dimensions f, and then stabilizes and decreases at a much slower rate. The number of dimensions where this change of regime occurs is generally used as the optimal value of f.

Non-linear dimensionality reduction: Manifold learning

Manifold Learning can be thought of a generalization of linear frameworks like PCA to be sensitive to non-linear structure in data, and they build a new space by aggregating the results of local approximations. Unlike linear methods, the obtained dimensions bear no clear significance, apart from indicating an idea of distance between points. Different algorithms differ in the way the local structure is defined, and the way the data points are projected.

The first existing methods to perform manifold learning were based on Multi-dimensional Scaling[401,402] (MDS), which finds a low dimensional representation of the original data where the pairwise distances between all points are the same as in the original high-dimensional space. Isomap[403] is a contemporary modification of MDS where only the local distances are computed, and the global distances are inferred from the local ones. In this way, in the low dimensional representation created by Isomap the conserved quantities are the geodesical distances, rather than the absolute euclidean distances themselves. While this allow to better unfold nonlinear manifolds, it makes Isomap very sensitive to court-circuiting (connecting data-points that belong to different parts of the manifold), which can quickly degrade the obtained low dimensional projection. Another variant of MDS common in the computational chemistry community is sketchcmap,[60,61,404] specifically designed to unravel high dimensional free energy landscapes from sampling simulations. It uses a sigmoid function rather than a Gaussian or a hard cut-off to build the network of locally connected points, with the goal of reproducing the distances that lie within a particular length scale that corresponds to the transition pathways between free energy basins. Kernel PCA (KPCA) is another method of the same family that applies PCA on a kernel representation rather than on the data itself. This allows to find a low-dimensional representation in the kernel space such that pairwise kernels are maintained. The PCA components on the kernel space are effectively non-linear components on the original space. When used with a Gaussian kernel, as is most often done, only the local neighbours of each point are captured, so the learned components in the kernel space are the components that maintain neighbours close to each other, which produces similar results as Isomap.

Another group of methods focuses on finding local linear factorizations of the data. Local Linear Embedding[405] (LLE) creates local linear maps that describe each point as a linear combination of its neighbors. Then, it builds a low-dimensional embedding where the local linear relationships are conserved. Effectively this is equivalent of merging a series of local Principal Component Analyses. There exist several modifications of LLE to improve the stability in ceratin conditions (such as MLLE[406] and Hessian Eigenmapping[407] Local Tangent Space Alignment[408] (LTSA)).

Perhaps the most popular family of methods today is based on stochastic neighbor embedding methods. This family of algorithms compute the likelihood that two points in a high-dimensional space are linked, and then builds a low-dimensional embedding attempting to conserve the relationships between pairs of datapoitns. One of the most used dimensionality reduction techniques is t-distributed Stochastic Neighbours Embedding[138] (t-SNE), which belongs to this class. In t-SNE, the links in the original space are described using Gaussian joint probabilities while in the embedded space Student's t-distributions are used. This allows t-SNE to adequately assign small and large pairwise distances to similar and dissimilar datapoints respectively. Consequently, t-SNE has better capabilities to disentangle complex manifolds than other methods that rely only on Gaussian distributions such as KPCA. The long range forces induced by the t-distribution allow t-SNE to deal better with non-uniformly sampled data than methods like Isomap, LLE and variants, which are best suited to unfold a single continuous low dimensional manifold, a case rarely found in heterogenuous datasets.

Unlike linear methods like PCA, where the obtained features are linear combination of the features in the original space, the meaning of the dimensions in non-linear dimensionality reduction is quite obscure, and no clear diagnosis tool exists to determine the "goodness".

A.1.2 Clustering algorithms

Cluster algorithms can be classified according to their inherent notion of cluster, and their performance will depend on how well a the distribution of points in specific dataset fits this definition. Unless it can be shown mathematically that a type of clustering method is more suited for a specific case, the performance of clustering algorithms can only be assessed experimentally. Four of the most common families of clustering algorithms are:

- **Connectivity-based clustering (hierarchical clustering):**

  Connectivity-based clustering methods define a cluster using the maximum cutoff distance needed to connect all the elements in the cluster. Different cutoff distances will generate different clusters, which can be represented hierarchically using a dendogram. Rather than providing a single clustering result, this type of methods generates a hierarchy of clusters that progressively merge with each other with increasing cutoff distances. Hierarchical clustering methods constitute the theoretical foundation of clustering algorithms, but are generally considered obsolete.

- **Centroid-based clustering:**

  In Centroid-based clustering clusters are represented by points in space called "centroids". Each data point belongs to the cluster whose centroid is closest. Given a fixed number of clusters, the algorithm K-means[21] optimizes the centroid locations so that the squared distances between the points in a cluster and its centroid are minimized. Centroid-based algorithms like K-means generate a parition of the feature space in Voronoid cells with linear borders. One of their main drawbacks is that the number

of clusters has to be defined by the user, although there are tools such as the Elbow curve[409,410] that help to infer what is the adequate number. Moreover, they tend to generate clusters of approximately similar size which often leads to incorrectly cut borders if the sizes of the clusters are heterogenous.

- **Distribution-based clustering:**

  In distribution-based clustering clusters can be defined as data points belonging most likely to the same distribution, which mimics the way data sets are often generated: by sampling random elements from a distribution.

  The functional form of the underlying distributions have to be defined a priori. Most often the prior used is a mixture of Gaussian distributions, which yealds the Gaussian Mixture Model, usually solved using the Expectation-Maximization algorithm.[411] As for K-means, Gaussian mixture models require a fixed number of clusters to be given as input. However, unlike K-means, the size of the gaussians can freely vary, which makes this type of model much more adaptable to find clusters of different sizes, although the shape should fit an ellipsoid.

- **Density-based clustering**

  Density-based clustering algorithms define clusters as areas of high density of data points. They approximate the local density of points for each data entry and proceed to find the local minima of the point density, which will represent the centers of clusters.[412] The clusters are constructed by connecting the points with the higher density to points with lower density, which produces a hierarchical network similarly to hierarchical clustering methods. Perhaps the most popular density based clustering methods are DBSCAN[2] and OPTICS.[413] The parameter that defines the output of the model is the lowest density considered. Points where the local density is lower than this threshold are considered noise or outliers, a property that is unique to this family of methods.

  Unlike distribution or centroid based clustering methods, density based algorithms do not require a fixed number of clusters as input parameters, nor do they make assumptions concerning the underlying distribution, which is specially beneficial when dealing with big datasets and a large unspecified number of clusters. However, they require large density changes to detect the borders separating different clusters. Data sets with overlapping Gaussian distributions generate soft cluster borders, which are not always well characterised by Density-based methods. In such cases distribution-based clustering will generally produce better results.

A.2 Supervised ML algorithms

A.2.1 Linear models

A linear model expresses a value y_i as a linear combination of a collection of variables $\{X_1^i, ..., X_n^i\}$. The parameters $\boldsymbol{\alpha}$ and β of the linear combination are obtained through general-

ization from pairs of data ($y_i \in \mathbf{y}, \mathbf{x}_i \in X$) so that $\mathbf{y} \approx \boldsymbol{\alpha} X + \beta$.

Fitting a linear model implies finding an expression $\boldsymbol{\alpha} X + \boldsymbol{\beta}$ that generates a set of values $\mathbf{y}$ as similar as possible as a set of reference or target values $\mathbf{y}*$. This implies optimizing the parameters in the vector $\boldsymbol{\alpha}$ and β, which at the same time implies defining a cost function. The most common approach, ordinary least squares, refers to minimizing the mean squared error of the predictions, i.e. $\text{argmin}_{\boldsymbol{\alpha},\beta} \sum_i (\boldsymbol{\alpha} X + \beta - y_i*)^2$, which is the maximum likelihood estimator under the assumption that errors are normally distributed. Alternative solutions for the regression coefficients (e.g. sparsity, non-negativity...) can be obtained by adding additional or alternative terms in the cost function.[21,414]

Linear models are at the core of many machine learning models, as a large part of non-linear regression and classification models (like kernel methods and neural networks) are based on generating a coordinate system where the target variable can be linearly expressed. A simpler approach to achieve non-linear models using a linear regression consists on generating new features based on non-linear combinations of the input features, for example using polynomial combinations.[21]

A.2.2 Decision Tree

A decision tree is an extremely simple model that creates predictors using consecutive binary splits of feature coordinates (see Figure A.1).

Decision trees are simple to understand and to interpret, and can deal with categorical and continuous scalar features at the same time. As they do not rely on similarity measures or a metric, they are also monotonic transformation invariant, and therefore insensitive to the scale of each feature. However, they have very high variance, as small variations in the data can lead to completely different tree structures. Moreover, they use orthogonal splits to separate the data, which makes them inadequate for regression purposes and for cases when the target property has a complex non-linear expression in the feature space. That makes them relatively inaccurate, and many other predictors perform better with similar data. Nevertheless, decision trees can be used in large numbers to create some of the most reliable machine learning algorithms to date (for some applications).

A.2.3 Ensemble models

Ensemble methods use groups of "weak" learning algorithms, or "learners", to obtain a model superior that what could be achieved with a single "strong" learning model alone.[415] There are broadly two basic approaches to ensemble models, bagging and boosting, which broadly separate algorithm where learners are applied in parallel or sequentially. In either case, the "weak" learners are most often random decision trees.

- **Bagging:** Bagging consists on averaging the classification or regression results of each

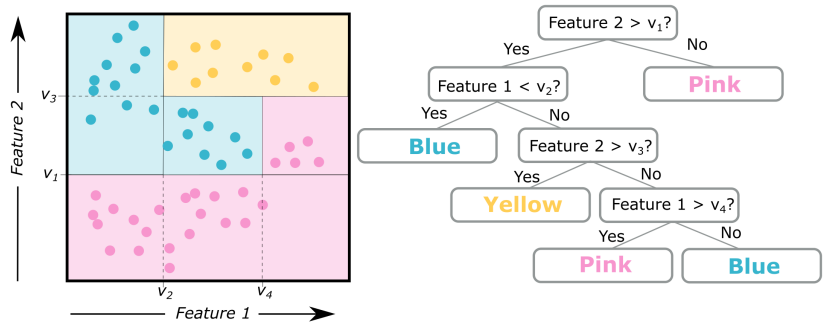


Figure A.1 – Illustration of an example of decision tree used to construct a classification model.

of the individual learners. In order to promote model variance, each learner in the ensemble is trained with a random subset of features and data.

The archetypal example of bagging ensemble models is the random forest algorithm, which combines random decision trees with bagging.[416,417] Random forests perform very well on large datasets with heterogeneous data. They are very robust to noise and to over-fitting, which make them a very popular choice in machine learning competitions. Moreover, they are able to provide a importance score for each feature in a regression/classification based on how often they were used in each decision tree.

- **Boosting:** Rather than averaging the results of multiple learners with equal weight, boosting refers to building an ensemble iteratively by training each new learner putting emphasis on the training instances that previous models mis-classified. The final prediction is based on an average vote, weighted by the individual accuracy of each model. The archetypal boosting ensemble model is AdaBoost,[418] which uses decision trees as the weak learner.

  Alternatively, weak learners can be concatenated by being trained on the remaining errors of the previous learner, which is a different way to give more importance to the difficult instances. This method is called gradient boosting. Some of the most popular algorithms based on gradient boosting are XGBoost[419] and LighGBM,[420] which have won multiple machine learning competitions since they were introduced in 2016.

  Boosting is generally more powerful than bagging, although it is also more sensitive to overfiting.

A.2.4 Neural Networks

The concept of neural networks does not refer to a specific model, but rather to a framework to build models. A specific NN model is defined by the architecture of the NN, which can have wildly different forms and capabilities for both supervised and unsupervised learning approaches.

Artificial Neural Networks (NN) are models formed by small processing units referred as nodes that are loosely inspired by neuronal networks in biological brains. Nodes receive inputs, make a small non-linear transformation, and signal the result to other nodes connected to it by edges, which is reminiscent of the synapse between real neurons. Nodes are arranged in layers, which in most cases can be divided in 3 groups: the input layer, the hidden layers, and the output layer. The input layer connects the input data, such as images and documents, to the hidden layers. The hidden layers transform the data through concatenated non-linear transformations. The output layer gets as input the data transformed by the hidden layers and constructs the final prediction of the neural network. The underlying idea of neural networks is that the concatenation of multiple simple non-linear processes can achieve different tasks like recognizing objects in images or finding relationships in data. To do so, a neural network have to be "trained", which technically means optimizing the parameters that define the non-linear transformation of each of its nodes in order to maximize the performance of the network for a specific task.

A full review of NN models is way outside the scope of this work, and the reader is referred elsewhere for an in depth review of the topic.[421] We broadly describe here the two most fundamental examples of neural network architectures to illustrate how they operate.

Basic neural network architectures

- Fully connected Neural Network: The fully connected NN (sometimes known as multi-layer perceptron) is the most basic NN architecture. It is composed of dense layers where all nodes in a layer are connected to all the nodes of the adjacent layers (see Figure A.2). The number of hidden layers and the number of nodes per layer defines the complexity of the network. Depending on the function used in the output node, they can be trained to perform simple classification or regression techniques. An interactive tool to illustrate the details of fully connected NNs can be found here: playground.tensorflow.org.

- Convolutional neural networks: Convolutional neural networks (CNN) are a type of neural networks that exploit some type of spatial or temporal correlation in data. They are useful when representation features have some sort of connection to other nearby features, for example adjacent pixels in an image (spatial correlation) or adjacent signal points in a sound recording (time correlation). Fully connected NN are inefficient in such cases as they treat each pixel independently. This leads to a huge amount of parameters that prevents them to efficiently deal with real images with millions of

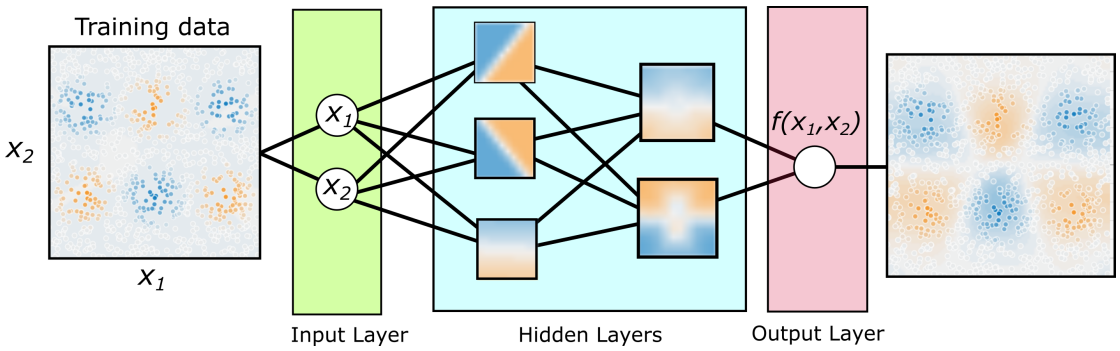


Figure A.2 – Illustration of a standard fully connected NN for regression.

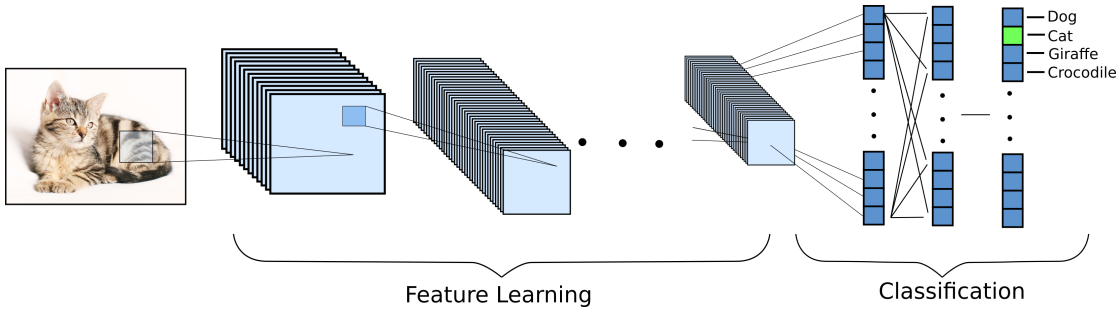


Figure A.3 – Illustration of the standard of a CNN network for image classifiation.

pixels. By exploiting the correlation between adjacent pixels in a common image or photography, convolutional neural networks are able to critically reduce the number of parameters needed to process image data. This is achieved by using consecutive convolutional filters that extract increasingly concrete features. A final fully connected layer is generally used to connect the high-level feature representation with the ultimate prediction (see Figure A.3).

A.2.5 Automatic supervised learning

New approaches are emerging where the whole learning process is being automatized, including data pre-processing, feature selection and extraction and the supervised learning itself. They are still over performed by human users, specially if they have additional external knowledge on the nature and source of the data, but they get better everyday, and they can provide reliable benchmark models.[422] Some of the most used are the open source python packages T-pot,[423] and auto-sklearn,[424,425] and the online "black-box" AutoML from Google (which is not free).

B MORESIM

MORESIM is a python package that allows for easy and modular design of replica exchange simulations. There exist plenty of molecular simulation software that allows to perform replica exchange simulations, but each of them have a rigid set of functionalities that are hard to extend without deep understanding of the source code. At the same time, the choice of underlying potential energy functions used to drive the simulations is often limited for each specific software. These two constraints are big limiting factors for the experimentation and design of replica exchange simulations and for all sampling simulations in general.

MORESIM is aimed at alleviating these issues. Each part of MORESIM is built in a modular way so that they can be combined by the user. This facilitates the design of new replica exchange simulation schemes, and allows to easily add new elements such as new potential energy functions or sampling methods. MORESIM is written completely in the Python programming language, which makes it clear and concise and requires minimal effort to modify.

The structure of MORESIM is based on classes that encompass individual tasks, so that they can be modified without altering other parts of the code. They can be divided in 4 groups, the trajectory class, the energy class, the simulation class and the replica exchange class:

- **Trajectory class:** The Trajectory class is simply a class to create trajectory instances that store the results of simulations, incuding molecular structures (in the form of ASE Atom objects), energies, acceptance probabilities of MC moves, etc...
- **EnergyCalculators:** The EnergyCalculators module contains the Energy class, a parent class that defines the structure that an energy function should have to be adequatly embedded in the framework. An adequate energy class should have the functions energy and force, which take a molecular structure as an input and return the corresponding energy and forces.
- **Simulations:** The Simulations module contains the different simulation classes, MC, MD and *reservoir*. An adequate Simulation class should take as input at least the temperature and an instance of an energy class. An instance of simulation should have a

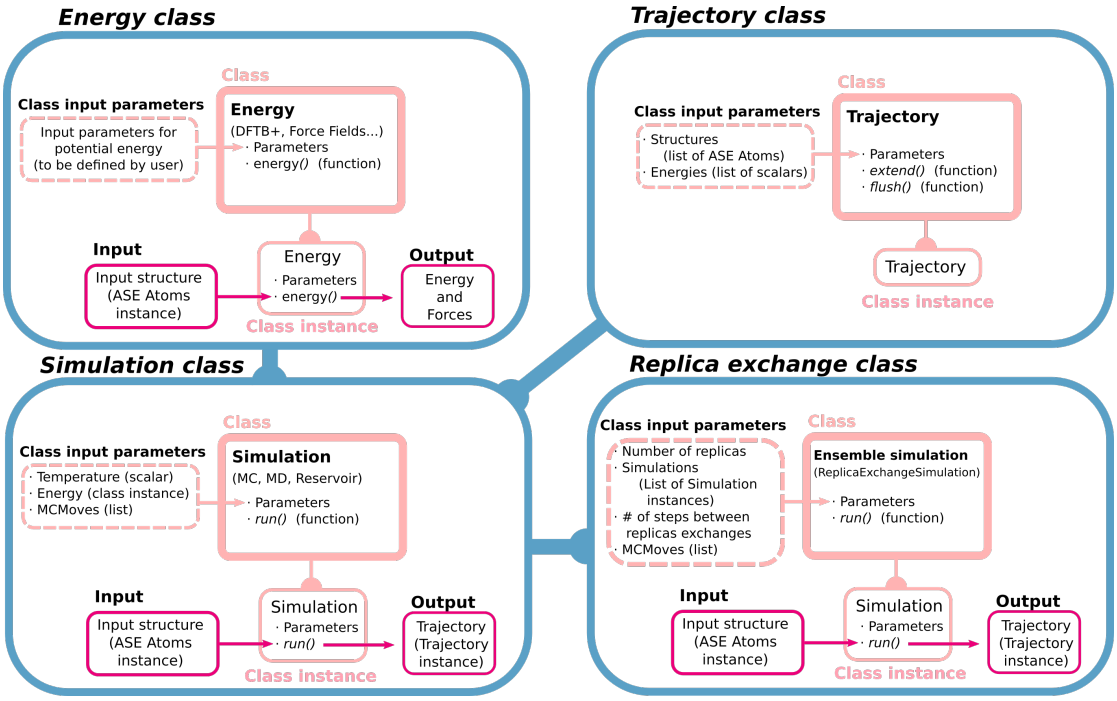


Figure B.1 – Schematic depiction of the different modular elements in MORESIM

method run that would take as input a molecular structure and a number of steps and generate a trajectory of structures.

- **ReplicaExchangeSimulation:** The Simulations module also contains the ReplicaExchangeSimulation class, which works similarly to a Simulation object, but that takes different parameters as input. An instance of a ReplicaExchangeSimulation requires the following parameters: The number of replicas, a list with of Simulation objects with length equal to the number of replicas, a set of initial states for the Simulation objects and the number of simulation steps between each replica exchange. ReplicaExchangeSimulation have the function *run* which takes as input the number of exchanges to run and starts the replica exchange simulation. Each replica is run in parallel and the results are gathered. The structures are then exchanged and the simulations are launched again.

  The ReplicaExchangeSimulation captures different types of replica exchange schemes based on canonical sampling. The baseline is a Hamiltonian-Replica Exchange, which allows for different energy functions and temperatures between replicas. It also allows to use biases in the probability exchanges, which can be used to integrate results from biased simulations like metadynamics.

The different calsses are embedded on a hierarchical order, with the ReplicaExchangeSimulation class being the final wrapper. The ReplicaExchangeSimulation class requires as input a list of instances of other simulation classes, like the MCSimulation class (Monte Carlo). Simulation classes like MCSimulation and MDSimulation require as parameters an instance

of an Energy class. Different simulation classes can require other parameters. For example the MCSimulation class requires as parameter an instance of MCMove class, which defines the possible moves of the MC simulation. Each part of the hierarchy can work individually as long as the lower levels of the hierarchy are defined. For example, a Energy class can work individually to predict energies, and a MCSimulation can run MC simulations individually as long as it has its Energy class defined.

A schematic depiction of each class in the framework can be found in Figure B.1.

C MolView

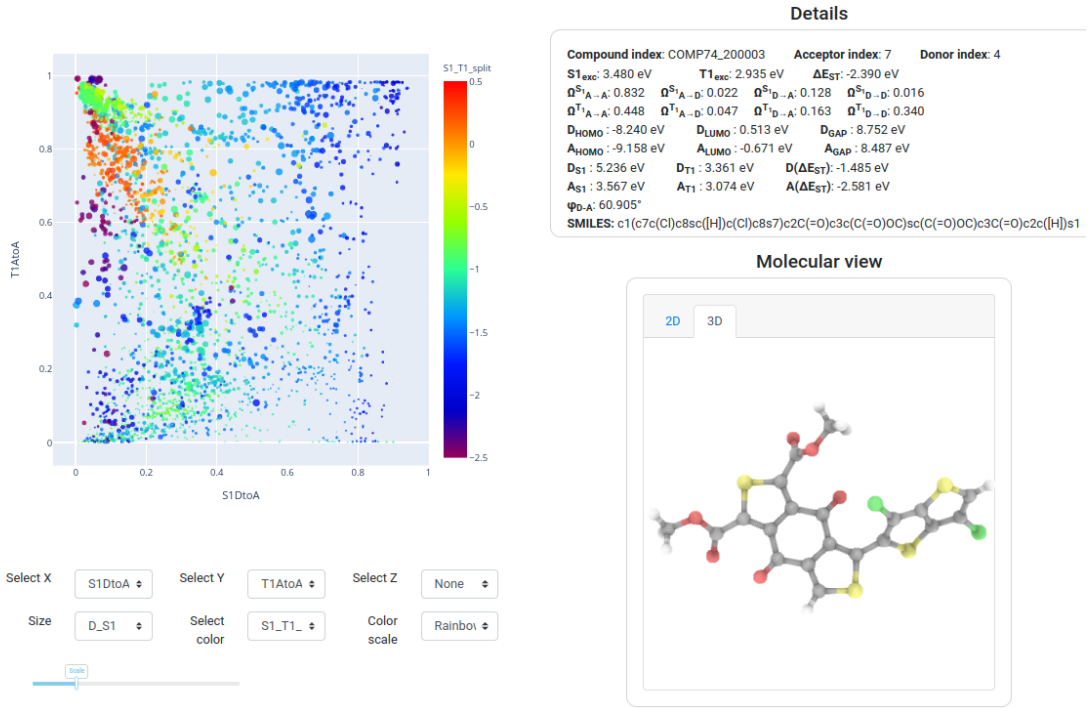


Figure C.1 – MolView example to explore donor-acceptor systems for intramolecular singlet fission (https://www.materialscloud.org/discover/isf#mcloudHeader).[6]

MolView (https://github.com/lcmd-epfl/molecular_data_explorer) is a Python script that allows to build web apps to visualize molecular data straightforwardly. It is based on the Python framework Dash(https://github.com/plotly/dash), a library that allows to build JavaScript apps using Python in a much more concise and clear manner. The basic standalone script MolView takes as input a csv file with data entries and a directory address with molecular structures:

```
python mol_view.py data.csv structures_directory
```

which generates a web app that automatically opens in a web browser (see Figure C.1). Extended functionalities are can be easily added with basic knoweledge of Python. The main two objectives of MolView is accessibility and shareability. Contrary to other software with similar capabilities such as CHEMISCOPE,[426] setting up MolView is straighforward and requires close to none technical skills. At the same time, while it is not comparable to mature software like DataWarrior,[427] it allows to easily deploy the apps on a server and make them available to anyone with internet connection. This is specially convenient using the free services of Heroku (https://www.heroku.com/), which allow to deploy web apps on a server for free. Examples of deployed apps are shown in Figures C.1 and C.2.

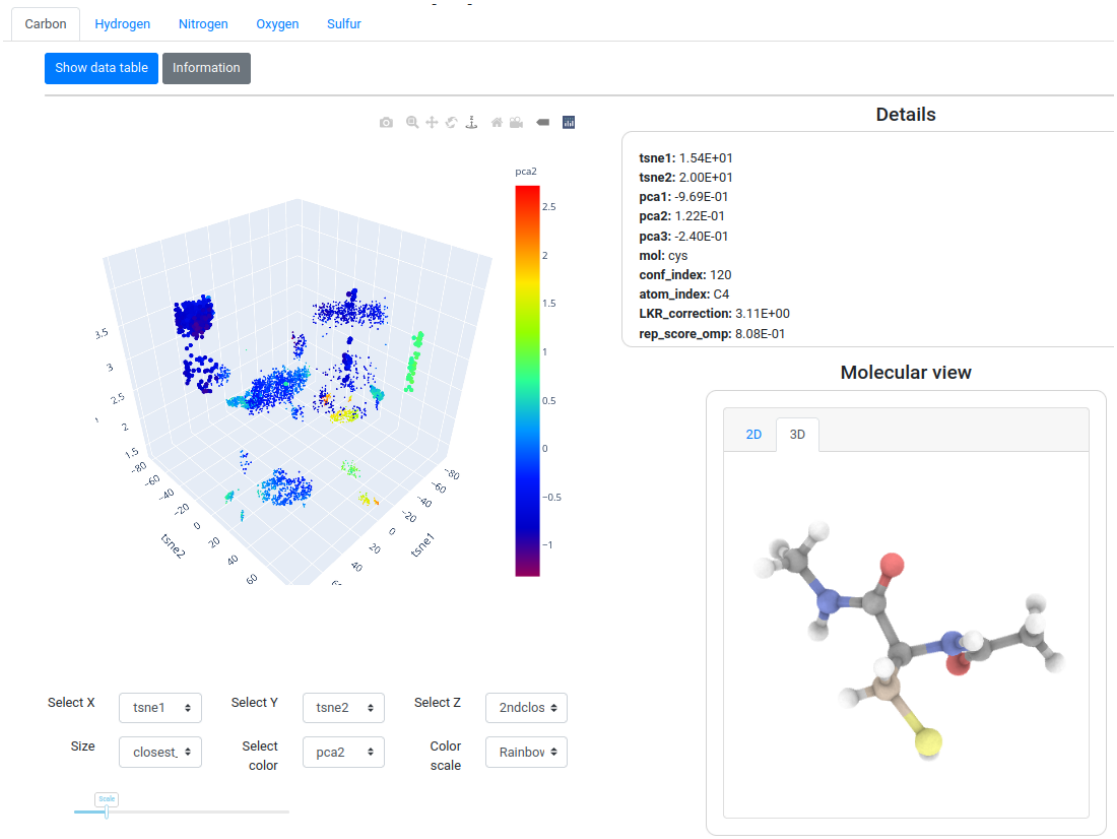


Figure C.2 – MolView example to explore the atomic enviornments of dipeptides https://atomic-environments-dipeptides.herokuapp.com/.[7]

D ML hands on tutorial for Chemists

For the occasion of the summer school Big Data and Machine Learning For Chemistry and with the help of 2 of the coorganizers (Veronika Juraskova and Saurep Chatterjee) I created a practical guide on the usage of many of the elements presented of this thesis in Python, with examples on different datasets. The tutorial is stored on a GitHub repository (https://github.com/lcmd-epfl/BDML4Chem) and freely available. A badge in the repository allows to open it using Deepnote, a cloud service that allows to easily deploy, share and execute virtual environments where code can be executed and visualized.

The tutorial is dived in three sessions, in the form of Jupyter notebooks:

- **1_basic_ML_tutorial.ipynb** : The first notebook contains a step-by-step guide on the different stages of a ML pipeline, including loading data, data visualization, basic statistical tests, clustering and dimensionality reduction and regression using different supervised algorithms. The ML pipeline is applied on the Boston Housing Dataset,[428] a common dataset to showcase ML techniques.

  The notebook also introduces and describes the basic fundamental python libraries related to data science, including Numpy,[429] Pandas,[430,431] Matplotlib,[432] Sci-kit Learn[137] and TensorFlow.[433]

- **2_chemical_example.ipynb** : The second notebook repeats the different stages of the ML pipeline, but this time using a real chemical example.[434]

- **3_atomistic_modelling.ipynb**: The third notebook introduces basic elements of atomistic modeling and molecular representations. A subset of the QM9[49,250] dataset is loaded and visualized using the Atomic Simulation Environment[435] (ASE) python package. Different molecular representations are generated using the QML[220] python package, which are then used to create a model of the atomization energy of the compounds. As an example of unsupervised learning, the atom centered version of SLATM, aSLATM, is used as input for a dimensionality reduction to elucidate the different types of carbon in the dataset.

E Artworks

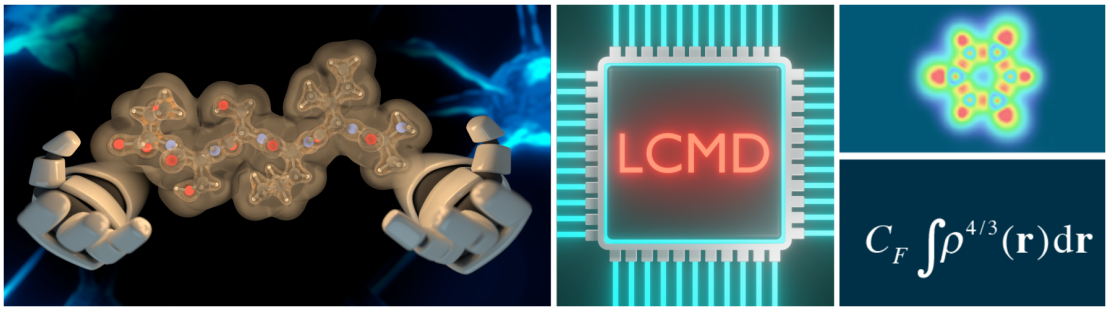


Figure E.1 – Logo of the Laboratory of Computational Molecular Design (LCMD).

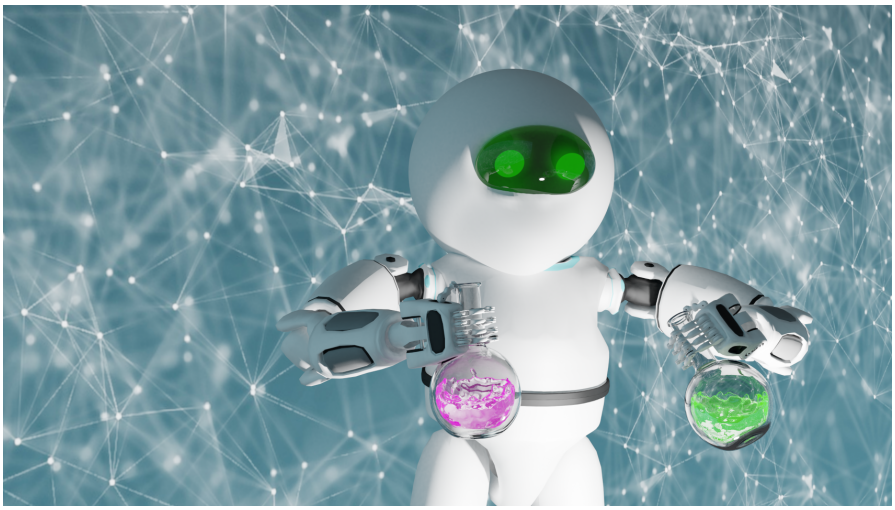


Figure E.2 – Logo of the summer school Big Data and Machine Learning 4 Chemistry 2021 (BDML4Chem)

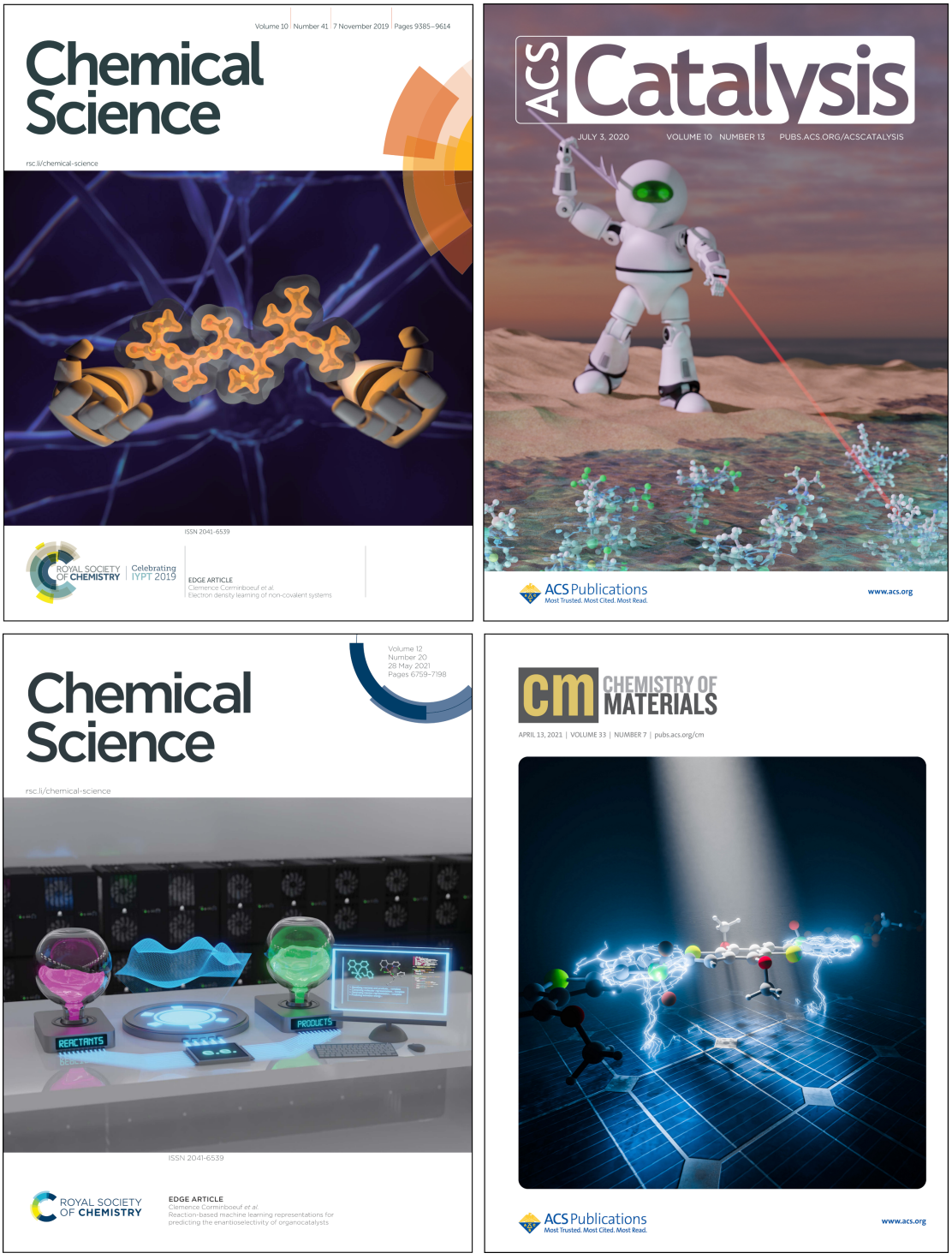


Figure E.3 – Journal covers of LCMD publications.[6,8–10]

Figure E.4 – Journal Table of Contents (ToC) images of LCMD publications.[6,11–14]

Appendix E. Artworks

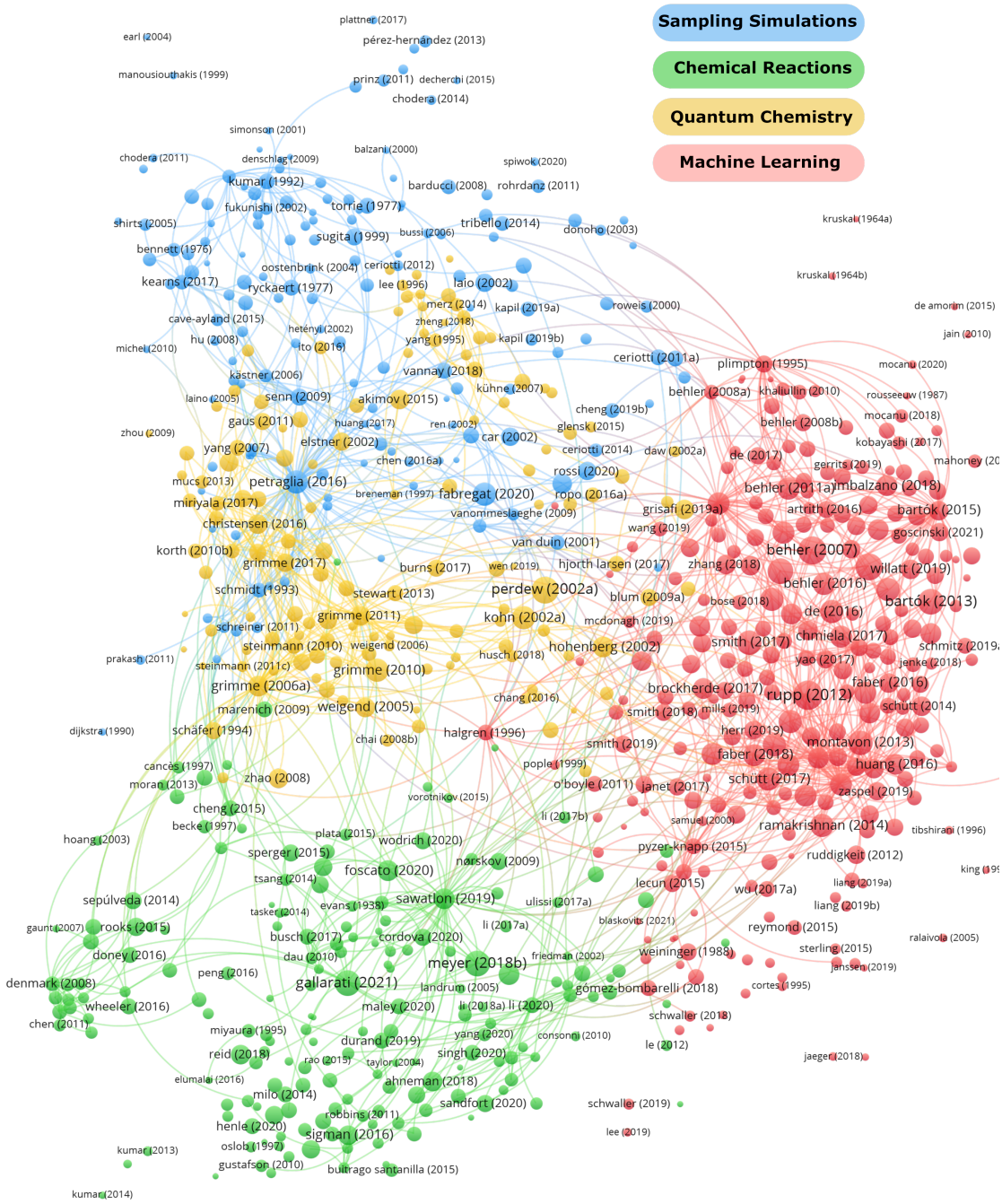


Figure E.5 – Citation network build with the references in this thesis using the software VOSviewer.[15] Nodes represent scientific articles and edges between two nodes indicate that one of them is cited by the other. The size of a node indicates the degree of the node, meaning how many edges it contains, and is a measure of the relevance of the article in the thesis. The articles are clustered in four groups based on the structure of the network, and they loosely represent the four main areas of this thesis: Machine Learning, Sampling Simulations, Quantum Chemistry and Chemical Reaction.

Bibliography

[1] Allen, F. H.; Bellard, S.; Brice, M.; Cartwright, B. A.; Doubleday, A.; Higgs, H.; Hummelink, T.; Hummelink-Peters, B.; Kennard, O.; Motherwell, W., et al. The Cambridge Crystallographic Data Centre: computer-based search, retrieval, analysis and display of information. *Acta Crystallogr. B* **1979**, *35*, 2331–2339.

[2] Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X., et al. A density-based algorithm for discovering clusters in large spatial databases with noise. KDD. 1996; pp 226–231.

[3] Virtanen, P. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272.

[4] Petraglia, R.; Nicolaï, A.; Wodrich, M. D.; Ceriotti, M.; Corminboeuf, C. Beyond static structures: Putting forth REMD as a tool to solve problems in computational organic chemistry. *J. Comput. Chem.* **2015**, *37*, 83–92.

[5] Xie, Q.; Dai, Z.; Hovy, E.; Luong, M.-T.; Le, Q. V. Unsupervised data augmentation for consistency training. *arXiv:1904.12848* **2019**, *pre-print*.

[6] Blaskovits, J. T.; Fumanal, M.; Vela, S.; Fabregat, R.; Corminboeuf, C. Identifying the Trade-off between Intramolecular Singlet Fission Requirements in Donor–Acceptor Copolymers. *Chem. Mater.* **2021**, *33*, 2567–2575.

[7] Fabregat, R.; Fabrizio, A.; Engel, E.; Meyer, B.; Juraskova, V.; Ceriotti, M.; Corminboeuf, C. Local Kernel Regression and Neural Network approaches to the conformational landscape of oligopeptides. **2021**, Submitted for publication.

[8] Fabrizio, A.; Grisafi, A.; Meyer, B.; Ceriotti, M.; Corminboeuf, C. Electron density learning of non-covalent systems. *Chem. Sci.* **2019**, *10*, 9424–9432.

[9] Cordova, M.; Wodrich, M. D.; Meyer, B.; Sawatlon, B.; Corminboeuf, C. Data-Driven Advancement of Homogeneous Nickel Catalyst Activity for Aryl Ether Cleavage. *ACS Catal.* **2020**, *10*, 7021–7031.

[10] Gallarati, S.; Fabregat, R.; Laplaza, R.; Bhattacharjee, S.; Wodrich, M. D.; Corminboeuf, C. Reaction-based machine learning representations for predicting the enantioselectivity of organocatalysts. *Chem. Sci.* **2021**, *12*, 6879–6889.

[11] Fabrizio, A.; Petraglia, R.; Corminboeuf, C. Balancing Density Functional Theory Interaction Energies in Charged Dimers Precursors to Organic Semiconductors. *J. Chem. Theory Comput.* **2020**, *16*, 3530–3542.

[12] Blaskovits, J. T.; Fumanal, M.; Vela, S.; Corminboeuf, C. Designing Singlet Fission Candidates from Donor–Acceptor Copolymers. *Chem. Mater.* **2020**, *32*, 6515–6524.

[13] Wodrich, M. D.; Sawatlon, B.; Busch, M.; Corminboeuf, C. On the generality of molecular volcano plots. *ChemCatChem* **2018**, *10*, 1586–1591.

[14] Blaskovits, J. T.; Lin, K.-H.; Fabregat, R.; Swiderska, I.; Wu, H.; Corminboeuf, C. Is a Single Conformer Sufficient to Describe the Reorganization Energy of Amorphous Organic Transport Materials? *J. Phys. Chem. C* **2021**, *125*, 17355–17362.

[15] Van Eck, N. J.; Waltman, L. Software survey: VOSviewer, a computer program for bibliometric mapping. *Scientometrics* **2010**, *84*, 523–538.

[16] Gelernter, H.; Sanders, A.; Larsen, D.; Agarwal, K.; Boivie, R.; Spritzer, G.; Searleman, J. Empirical explorations of SYNCHEM. *Science* **1977**, *197*, 1041–1049.

[17] Muggleton, S.; King, R. D.; Stenberg, M. J. Protein secondary structure prediction using logic-based machine learning. *Protein Eng. Des. Sel.* **1992**, *5*, 647–657.

[18] King, R. D.; Muggleton, S.; Lewis, R. A.; Sternberg, M. Drug design by machine learning: The use of inductive logic programming to model the structure-activity relationships of trimethoprim analogues binding to dihydrofolate reductase. *Proc. Natl. Acad. Sci. U.S.A.* **1992**, *89*, 11322–11326.

[19] King, R. D.; Muggleton, S. H.; Srinivasan, A.; Sternberg, M. Structure-activity relationships derived by machine learning: The use of atoms and their bond connectivities to predict mutagenicity by inductive logic programming. *Proc. Natl. Acad. Sci. U.S.A.* **1996**, *93*, 438–442.

[20] Martin, Y. C. *Quantitative drug design: a critical introduction*; CRC Press, 2010.

[21] Hastie, T.; Friedman, J.; Tibshirani, R. *The Elements of Statistical Learning*, 2nd ed.; Springer, 2001; New York, USA.

[22] Funke, P. T.; Malinowski, E. R.; Martire, D. E.; Pollara, L. Z. Application of factor analysis to the prediction of activity coefficients of nonelectrolytes. *J. Sep. Sci.* **1966**, *1*, 661–676.

[23] Malinowski, E. R.; Howery, D. G. *Factor analysis in chemistry*; Wiley New York, 1980; Vol. 3.

[24] Ichiye, T.; Karplus, M. Collective motions in proteins: a covariance analysis of atomic fluctuations in molecular dynamics and normal mode simulations. *Proteins Struct. Funct. Bioinf.* **1991**, *11*, 205–217.

[25] Altis, A.; Nguyen, P. H.; Hegger, R.; Stock, G. Dihedral angle principal component analysis of molecular dynamics simulations. *J. Chem. Phys.* **2007**, *126*, 244111.

[26] Stone, J. E.; Hardy, D. J.; Ufimtsev, I. S.; Schulten, K. GPU-accelerated molecular modeling coming of age. *J. Mol. Graphics Model.* **2010**, *29*, 116–125.

[27] Häse, F.; Roch, L. M.; Aspuru-Guzik, A. Next-generation experimentation with self-driving laboratories. *Trends Chem.* **2019**, *1*, 282–291.

[28] Behler, J.; Parrinello, M. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Phys. Rev. Lett.* **2007**, *98*, 146401.

[29] Rupp, M.; Tkatchenko, A.; Müller, K.-R.; von Lilienfeld, O. A. Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. *Phys. Rev. Lett.* **2012**, *108*, 058301.

[30] von Lilienfeld, O. A. Quantum machine learning in chemical compound space. *Angew. Chem. Int. Ed.* **2018**, *57*, 4164–4169.

[31] Christensen, A. S.; Bratholm, L. A.; Faber, F. A.; von Lilienfeld, O. A. FCHL revisited: Faster and more accurate quantum machine learning. *J. Chem. Phys.* **2020**, *152*, 044107.

[32] Li, Z.; Kermode, J. R.; De Vita, A. Molecular dynamics with on-the-fly machine learning of quantum-mechanical forces. *Phys. Rev. Lett.* **2015**, *114*, 096405.

[33] Chmiela, S.; Tkatchenko, A.; Sauceda, H. E.; Poltavsky, I.; Schütt, K. T.; Müller, K.-R. Machine learning of accurate energy-conserving molecular force fields. *Sci. Adv.* **2017**, *3*, e1603015.

[34] Chmiela, S.; Sauceda, H. E.; Müller, K.-R.; Tkatchenko, A. Towards exact molecular dynamics simulations with machine-learned force fields. *Nat. Commun.* **2018**, *9*.

[35] Bereau, T.; Andrienko, D.; von Lilienfeld, O. A. Transferable Atomic Multipole Machine Learning Models for Small Organic Molecules. *J. Chem. Theory Comput.* **2015**, *11*, 3225–3233.

[36] Grisafi, A.; Wilkins, D. M.; Csányi, G.; Ceriotti, M. Symmetry-adapted machine learning for tensorial properties of atomistic systems. *Phys. Rev. Lett.* **2018**, *120*, 036002.

[37] Wilkins, D. M.; Grisafi, A.; Yang, Y.; Lao, K. U.; DiStasio, R. A.; Ceriotti, M. Accurate molecular polarizabilities with coupled cluster theory and machine learning. *Proc. Natl. Acad. Sci. U.S.A.* **2019**, *116*, 3401–3406.

[38] Bartók, A. P.; Payne, M. C.; Kondor, R.; Csányi, G. Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons. *Phys. Rev. Lett.* **2010**, *104*, 136403.

[39] Behler, J. First Principles Neural Network Potentials for Reactive Simulations of Large Molecular and Condensed Systems. *Angew. Chem. Int. Ed.* **2017**, *56*, 12828–12840.

[40] Smith, J. S.; Nebgen, B.; Lubbers, N.; Isayev, O.; Roitberg, A. E. Less is more: Sampling chemical space with active learning. *J. Chem. Phys.* **2018**, *148*, 241733.

[41] Westermayr, J.; Marquetand, P. Machine learning for electronically excited states of molecules. *Chem. Rev.* **2021**, *121*, 9873–9926.

[42] Brockherde, F.; Vogt, L.; Li, L.; Tuckerman, M. E.; Burke, K.; Müller, K.-R. Bypassing the Kohn-Sham equations with machine learning. *Nat. Commun.* **2017**, *8*, 872.

[43] Grisafi, A.; Fabrizio, A.; Meyer, B.; Wilkins, D. M.; Corminboeuf, C.; Ceriotti, M. Transferable Machine-Learning Model of the Electron Density. *ACS Cent. Sci.* **2018**, *5*, 57–64.

[44] Schmidt, E.; Fowler, A. T.; Elliott, J. A.; Bristowe, P. D. Learning models for electron densities with Bayesian regression. *Comput. Mater. Sci.* **2018**, *149*, 250–258.

[45] Alred, J. M.; Bets, K. V.; Xie, Y.; Yakobson, B. I. Machine learning electron density in sulfur crosslinked carbon nanotubes. *Compos. Sci. Technol.* **2018**, *166*, 3–9.

[46] Chandrasekaran, A.; Kamal, D.; Batra, R.; Kim, C.; Chen, L.; Ramprasad, R. Solving the electronic structure problem with machine learning. *Npj. Comput. Mater.* **2019**, *5*, 1–7.

[47] Schütt, K.; Gastegger, M.; Tkatchenko, A.; Müller, K.-R.; Maurer, R. J. Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions. *Nat. Commun.* **2019**, *10*, 1–10.

[48] Blum, V.; Gehrke, R.; Hanke, F.; Havu, P.; Havu, V.; Ren, X.; Reuter, K.; Scheffler, M. Ab initio molecular simulations with numeric atom-centered orbitals. *Comput. Phys. Commun.* **2009**, *180*, 2175–2196.

[49] Ramakrishnan, R.; Dral, P. O.; Rupp, M.; von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Sci. Data* **2014**, *1*.

[50] Faber, F. A.; Lindmaa, A.; von Lilienfeld, O. A.; Armiento, R. Machine Learning Energies of 2 Million Elpasolite (A B C 2 D 6) Crystals. *Phys. Rev. Lett.* **2016**, *117*, 135502.

[51] Faber, F. A.; Hutchison, L.; Huang, B.; Gilmer, J.; Schoenholz, S. S.; Dahl, G. E.; Vinyals, O.; Kearnes, S.; Riley, P. F.; von Lilienfeld, O. A. Prediction Errors of Molecular Machine Learning Models Lower than Hybrid DFT Error. *J. Chem. Theory Comput.* **2017**, *13*, 5255–5264.

[52] Meyer, B.; Sawatlon, B.; Heinen, S.; von Lilienfeld, O. A.; Corminboeuf, C. Machine learning meets volcano plots: Computational discovery of cross-coupling catalysts. *Chem. Sci.* **2018**, *9*, 7069–7077.

[53] von Lilienfeld, O. A.; Müller, K.-R.; Tkatchenko, A. Exploring chemical compound space with quantum-based machine learning. *Nat. Rev. Chem.* **2020**, *4*, 347–358.

[54] Liang, J.; Xu, Y.; Liu, R.; Zhu, X. QM-sym, a symmetrized quantum chemistry database of 135 kilo molecules. *Sci. Data* **2019**, *6*, 1–8.

[55] Ramakrishnan, R.; Dral, P. O.; Rupp, M.; von Lilienfeld, O. A. Big Data Meets Quantum Chemistry Approximations: The Δ-Machine Learning Approach. *J. Chem. Theory Comput.* **2015**, *11*, 2087–2096.

[56] Botu, V.; Ramprasad, R. Adaptive machine learning framework to accelerate ab initio molecular dynamics. *Int. J. Quantum Chem.* **2015**, *115*, 1074–1083.

[57] Krems, R. Bayesian machine learning for quantum molecular dynamics. *Phys. Chem. Chem. Phys.* **2019**, *21*, 13392–13410.

[58] Westermayr, J.; Gastegger, M.; Menger, M. F. S. J.; Mai, S.; González, L.; Marquetand, P. Machine learning enables long time scale molecular photodynamics simulations. *Chem. Sci.* **2019**, *10*, 8100–8107.

[59] Brown, W. M.; Martin, S.; Pollock, S. N.; Coutsias, E. A.; Watson, J.-P. Algorithmic dimensionality reduction for molecular structure analysis. *J. Chem. Phys.* **2008**, *129*, 064118.

[60] Ceriotti, M.; Tribello, G. A.; Parrinello, M. Simplifying the representation of complex free-energy landscapes using sketch-map. *Proc. Natl. Acad. Sci. U.S.A.* **2011**, *108*, 13023–13028.

[61] Tribello, G. A.; Ceriotti, M.; Parrinello, M. Using sketch-map coordinates to analyze and bias molecular dynamics simulations. *Proc. Natl. Acad. Sci. U.S.A.* **2012**, *109*, 5196–5201.

[62] Spiwok, V.; Kriz, P. Time-lagged t-distributed stochastic neighbor embedding (t-SNE) of molecular simulation trajectories. *Front. Mol. Biosci.* **2020**, *7*, 132.

[63] Jaeger, S.; Fulle, S.; Turk, S. Mol2vec: unsupervised machine learning approach with chemical intuition. *J. Chem. Inf. Model.* **2018**, *58*, 27–35.

[64] Janssen, A. P.; Grimm, S. H.; Wijdeven, R. H.; Lenselink, E. B.; Neefjes, J.; van Boeckel, C. A.; van Westen, G. J.; van der Stelt, M. Drug discovery maps, a machine learning model that visualizes and predicts kinome–inhibitor interaction landscapes. *J. Chem. Inf. Model.* **2018**, *59*, 1221–1229.

[65] De, S.; Musil, F.; Ingram, T.; Baldauf, C.; Ceriotti, M. Mapping and classifying molecules from a high-throughput structural database. *J. Cheminformatics* **2017**, *9*, 1–14.

[66] Sawatlon, B.; Wodrich, M. D.; Meyer, B.; Fabrizio, A.; Corminboeuf, C. Data Mining the C- C Cross-Coupling Genome. *ChemCatChem* **2019**, *11*, 4096–4107.

[67] Huang, F.; Li, R.; Wang, G.; Zheng, J.; Tang, Y.; Liu, J.; Yang, Y.; Yao, Y.; Shi, J.; Hong, W. Automatic classification of single-molecule charge transport data with an unsupervised machine-learning algorithm. *Phys. Chem. Chem. Phys.* **2020**, *22*, 1674–1681.

[68] Lim, J.; Ryu, S.; Kim, J. W.; Kim, W. Y. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *J. Cheminformatics* **2018**, *10*, 1–9.

[69] Liu, Q.; Allamanis, M.; Brockschmidt, M.; Gaunt, A. L. Constrained graph variational autoencoders for molecule design. *arXiv:1805.09076* **2018**, *pre-print*.

[70] Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.* **2018**, *4*, 268–276.

[71] Krenn, M.; Häse, F.; Nigam, A.; Friederich, P.; Aspuru-Guzik, A. Self-Referencing Embedded Strings (SELFIES): A 100% robust molecular string representation. *Mach. Learn.: Sci. Technol.* **2020**, *1*, 045024.

[72] Weinberger, K. Q.; Saul, L. K. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.* **2009**, *10(2)*.

[73] Bellet, A.; Habrard, A.; Sebban, M. A survey on metric learning for feature vectors and structured data. *arXiv:1306.6709* **2013**, *pre-print*.

[74] Na, G. S.; Chang, H.; Kim, H. W. Machine-guided representation for accurate graph-based molecular machine learning. *Phys. Chem. Chem. Phys.* **2020**, *22*, 18526–18535.

[75] McInnes, L.; Healy, J.; Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv:1802.03426* **2018**, *pre-print*.

[76] Le, L.; Patterson, A.; White, M. Supervised autoencoders: Improving generalization performance with unsupervised regularizers. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 107–117.

[77] Bartók, A. P.; De, S.; Poelking, C.; Bernstein, N.; Kermode, J. R.; Csányi, G.; Ceriotti, M. Machine learning unifies the modeling of materials and molecules. *Sci. Adv.* **2017**, *3*, e1701816.

[78] Imbalzano, G.; Anelli, A.; Giofré, D.; Klees, S.; Behler, J.; Ceriotti, M. Automatic selection of atomic fingerprints and reference configurations for machine-learning potentials. *J. Chem. Phys.* **2018**, *148*, 241730.

[79] Behler, J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *J. Chem. Phys.* **2011**, *134*, 074106.

[80] Hansen, K.; Biegler, F.; Ramakrishnan, R.; Pronobis, W.; von Lilienfeld, O. A.; Müller, K.-R.; Tkatchenko, A. Machine Learning Predictions of Molecular Properties: Accurate Many-Body Potentials and Nonlocality in Chemical Space. *J. Phys. Chem. Lett.* **2015**, *6*, 2326–2331.

[81] Pronobis, W.; Tkatchenko, A.; Muller, K.-R. Many-body descriptors for predicting molecular properties with machine learning: Analysis of pairwise and three-body interactions in molecules. *J. Chem. Theory Comput.* **2018**, *14*, 2991–3003.

[82] Faber, F. A.; Christensen, A. S.; Huang, B.; von Lilienfeld, O. A. Alchemical and structural distribution based representation for universal quantum machine learning. *J. Chem. Phys.* **2018**, *148*, 241717.

[83] Weinreich, J.; Browning, N. J.; von Lilienfeld, O. A. Machine learning of free energies in chemical compound space using ensemble representations: Reaching experimental uncertainty for solvation. *J. Chem. Phys.* **2021**, *154*, 134113.

[84] Gensch, T.; dos Passos Gomes, G.; Friederich, P.; Peters, E.; Gaudin, T.; Pollice, R.; Jorner, K.; Nigam, A.; Lindner D'Addario, M.; Sigman, M. S.; et al., A Comprehensive Discovery Platform for Organophosphorus Ligands for Catalysis. *ChemRxiv* **2021**, Preprint.

[85] Fukunishi, H.; Watanabe, O.; Takada, S. On the Hamiltonian replica exchange method for efficient sampling of biomolecular systems: Application to protein structure prediction. *J. Chem. Phys.* **2002**, *116*, 9058–9067.

[86] Okur, A.; Roe, D. R.; Cui, G.; Hornak, V.; Simmerling, C. Improving Convergence of Replica-Exchange Simulations through Coupling to a High-Temperature Structure Reservoir. *J. Chem. Theory Comput.* **2007**, *3*, 557–568.

[87] Riplinger, C.; Neese, F. An efficient and near linear scaling pair natural orbital based local coupled cluster method. *J. Chem. Phys.* **2013**, *138*, 034106.

[88] Neese, F.; Valeev, E. F. Revisiting the atomic natural orbital approach for basis sets: Robust systematic basis sets for explicitly correlated and conventional correlated ab initio methods? *J. Chem. Theory Comput.* **2010**, *7*, 33–43.

[89] Pati, Y. C.; Rezaiifar, R.; Krishnaprasad, P. S. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. Proceedings of the 27th Asilomar Conference on Signals, Systems, and Computers. 1993; pp 40–44.

[90] Mitchell, T. M. Does machine learning really work? *AI Mag.* **1997**, *18*, 11–11.

[91] Samuel, A. L. Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.* **1959**, *3*, 210–229.

[92] Sutton, R. S.; Barto, A. G. *Reinforcement learning: An introduction*; MIT press, 2018.

[93] Bartók, A. P.; Kondor, R.; Csányi, G. On representing chemical environments. *Phys. Rev. B* **2013**, *87*, 184115.

[94] Huang, B.; von Lilienfeld, O. A. Communication: Understanding molecular representations in machine learning: The role of uniqueness and target similarity. *J. Chem. Phys.* **2016**, *145*, 161102.

[95] Karelson, M. *Molecular descriptors in QSAR/QSPR*; Wiley-Interscience, 2000.

[96] Todeschini, R.; Consonni, V. *Handbook of molecular descriptors*; John Wiley & Sons, 2008; Vol. 11.

[97] Consonni, V.; Todeschini, R. *Recent advances in QSAR studies*; Springer, 2010; pp 29–102.

[98] Karelson, M.; Lobanov, V. S.; Katritzky, A. R. Quantum-chemical descriptors in QSAR/QSPR studies. *Chem. Rev.* **1996**, *96*, 1027–1044.

[99] David, L.; Thakkar, A.; Mercado, R.; Engkvist, O. Molecular representations in AI-driven drug discovery: a review and practical guide. *J. Cheminformatics* **2020**, *12*, 1–22.

[100] Ghiringhelli, L. M.; Vybiral, J.; Ahmetcik, E.; Ouyang, R.; Levchenko, S. V.; Draxl, C.; Scheffler, M. Learning physical descriptors for materials science by compressed sensing. *New J. Phys.* **2017**, *19*, 023017.

[101] Obrezanova, O.; Csányi, G.; Gola, J. M.; Segall, M. D. Gaussian processes: a method for automatic QSAR modeling of ADME properties. *J. Chem. Inf. Model.* **2007**, *47*, 1847–1857.

[102] Gaulton, A.; Bellis, L. J.; Bento, A. P.; Chambers, J.; Davies, M.; Hersey, A.; Light, Y.; McGlinchey, S.; Michalovich, D.; Al-Lazikani, B., et al. ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Res.* **2012**, *40*, D1100–D1107.

[103] Sterling, T.; Irwin, J. J. ZINC 15–ligand discovery for everyone. *J. Chem. Inf. Model.* **2015**, *55*, 2324–2337.

[104] Christensen, A. S.; Faber, F. A.; von Lilienfeld, O. A. Operators in quantum machine learning: Response properties in chemical space. *J. Chem. Phys.* **2019**, *150*, 064105.

[105] Willatt, M. J.; Musil, F.; Ceriotti, M. Feature optimization for atomistic machine learning yields a data-driven construction of the periodic table of the elements. *Phys. Chem. Chem. Phys.* **2018**, *20*, 29661–29668.

[106] Willatt, M. J.; Musil, F.; Ceriotti, M. Atom-density representations for machine learning. *J. Chem. Phys.* **2019**, *150*, 154110.

[107] Goscinski, A.; Musil, F.; Pozdnyakov, S.; Ceriotti, M. Optimal radial basis for density-based atomic representations. *arXiv:2105.08717* **2021**, *pre-print*.

[108] Langer, M. F.; Goeßmann, A.; Rupp, M. Representations of molecules and materials for interpolation of quantum-mechanical simulations via machine learning. *arXiv:2003.12081* **2020**, *pre-print*.

[109] Kohn, W. Density functional and density matrix method scaling linearly with the number of atoms. *Phys. Rev. Lett.* **1996**, *76*, 3168.

[110] Prodan, E.; Kohn, W. Nearsightedness of electronic matter. *Proc. Natl. Acad. Sci. U.S.A.* **2005**, *102*, 11635–11638.

[111] Mayo, S. L.; Olafson, B. D.; Goddard, W. A. DREIDING: a generic force field for molecular simulations. *J. Phys. Chem.* **1990**, *94*, 8897–8909.

[112] Vanommeslaeghe, K.; Hatcher, E.; Acharya, C.; Kundu, S.; Zhong, S.; Shim, J.; Darian, E.; Guvench, O.; Lopes, P.; Vorobyov, I.; Mackerell, A. D. CHARMM general force field: A force field for drug-like molecules compatible with the CHARMM all-atom additive biological force fields. *J. Comput. Chem.* **2009**, *31*, 671–690.

[113] Halgren, T. A. Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *J. Comput. Chem.* **1996**, *17*, 490–519.

[114] Jiang, B.; Li, J.; Guo, H. Potential energy surfaces from high fidelity fitting of ab initio points: The permutation invariant polynomial-neural network approach. *Int. Rev. Phys. Chem.* **2016**, *35*, 479–506.

[115] Shapeev, A. V. Moment tensor potentials: A class of systematically improvable interatomic potentials. *Multiscale Model. Simul.* **2016**, *14*, 1153–1173.

[116] Faber, F.; Lindmaa, A.; von Lilienfeld, O. A.; Armiento, R. Crystal structure representations for machine learning models of formation energies. *Int. J. Quantum Chem.* **2015**, *115*, 1094–1101.

[117] Huo, H.; Rupp, M. Unified representation for machine learning of molecules and crystals. *arXiv:1704.06439* **2017**, *pre-print*.

[118] Artrith, N.; Urban, A.; Ceder, G. Efficient and accurate machine-learning interpolation of atomic energies in compositions with many species. *Phys. Rev. B* **2017**, *96*, 014112.

[119] Huang, B.; von Lilienfeld, O. A. Quantum machine learning using atom-in-molecule-based fragments selected on the fly. *Nat. Chem.* **2020**, *12*, 945–951.

[120] Schütt, K. T.; Glawe, H.; Brockherde, F.; Sanna, A.; Müller, K. R.; Gross, E. K. U. How to represent crystal structures for machine learning: Towards fast prediction of electronic properties. *Phys. Rev. B* **2014**, *89*, 205118.

[121] Seko, A.; Hayashi, H.; Nakayama, K.; Takahashi, A.; Tanaka, I. Representation of compounds for machine-learning prediction of physical properties. *Phys. Rev. B* **2017**, *95*, 144110.

[122] von Lilienfeld, O. A.; Ramakrishnan, R.; Rupp, M.; Knoll, A. Fourier series of atomic radial distribution functions: A molecular fingerprint for machine learning models of quantum chemical properties. *Int. J. Quantum Chem.* **2015**, *115*, 1084–1093.

[123] Gastegger, M.; Schwiedrzik, L.; Bittermann, M.; Berzsenyi, F.; Marquetand, P. wACSF—Weighted atom-centered symmetry functions as descriptors in machine learning potentials. *J. Chem. Phys.* **2018**, *148*, 241709.

[124] Bartók, A. P.; Csányi, G. Gaussian approximation potentials: A brief tutorial introduction. *Int. J. Quantum Chem* **2015**, *115*, 1051–1057.

[125] De, S.; Bartók, A. P.; Csányi, G.; Ceriotti, M. Comparing molecules and solids across structural and alchemical space. *Phys. Chem. Chem. Phys.* **2016**, *18*, 13754–13769.

[126] Goscinski, A.; Fraux, G.; Imbalzano, G.; Ceriotti, M. The role of feature space in atomistic learning. *Mach. Learn.: Sci. Technol.* **2021**, *2*, 025028.

[127] Parsaeifard, B.; Sankar De, D.; Christensen, A. S.; Faber, F. A.; Kocer, E.; De, S.; Behler, J.; von Lilienfeld, O. A.; Goedecker, S. An assessment of the structural resolution of various fingerprints commonly used in machine learning. *Mach. Learn.: Sci. Technol.* **2021**, *2*, 015018.

[128] Unke, O. T.; Chmiela, S.; Gastegger, M.; Schütt, K. T.; Sauceda, H. E.; Müller, K.-R. SpookyNet: Learning Force Fields with Electronic Degrees of Freedom and Nonlocal Effects. *arXiv:2105.00304* **2021**, *pre-print*.

[129] Pozdnyakov, S.; Willatt, M.; Bartók, A.; Ortner, C.; Csányi, G.; Ceriotti, M. Incompleteness of Atomic Structure Representations. *Phys. Rev. Lett.* **2020**, *125*.

[130] Smola, A. J.; Schölkopf, B.; Müller, K.-R. The connection between regularization operators and support vector kernels. *Neural Netw.* **1998**, *11*, 637–649.

[131] Rudi, A.; Rosasco, L. Generalization Properties of Learning with Random Features. NIPS. 2017; pp 3215–3225.

[132] Glielmo, A.; Sollich, P.; De Vita, A. Accurate interatomic force fields via machine learning with covariant kernels. *Phys. Rev. B* **2017**, *95*, 214302.

[133] Collins, C. R.; Gordon, G. J.; von Lilienfeld, O. A.; Yaron, D. J. Constant size descriptors for accurate machine learning models of molecular properties. *J. Chem. Phys.* **2018**, *148*, 241718.

[134] Musil, F.; Grisafi, A.; Bartók, A. P.; Ortner, C.; Csányi, G.; Ceriotti, M. Physics-inspired structural representations for molecules and materials. *Chem. Rev.* **2021**, *121*, 9759–9815.

[135] Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* **2018**, *9*, 513–530.

[136] Huang, B.; von Lilienfeld, O. A. The "DNA" of chemistry: Scalable quantum machine learning with "amons". *arXiv:1707.04146* **2017**, *pre-print*.

[137] Pedregosa, F. et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

[138] Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9(11)*.

[139] Pfitzner, D.; Leibbrandt, R.; Powers, D. Characterization and evaluation of similarity measures for pairs of clusterings. *Knowl. Inf. Sys.* **2009**, *19*, 361–394.

[140] Rousseeuw, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65.

[141] Sugar, C. A.; James, G. M. Finding the number of clusters in a dataset: An information-theoretic approach. *J. Am. Stat. Assoc.* **2003**, *98*, 750–763.

[142] De Amorim, R. C.; Hennig, C. Recovering the number of clusters in data sets with noise features using feature rescaling factors. *Inf. Sci.* **2015**, *324*, 126–145.

[143] Feldman, R.; Sanger, J., et al. *The text mining handbook: advanced approaches in analyzing unstructured data*; Cambridge university press, 2007.

[144] Kirkpatrick, P.; Ellis, C. Chemical space. *Nature* **2004**, *432*, 823–824.

[145] Reymond, J.-L.; Awale, M. Exploring chemical space for drug discovery using the chemical universe database. *ACS Chem. Neurosci.* **2012**, *3*, 649–657.

[146] Alvarez, M. A.; Rosasco, L.; Lawrence, N. D. Kernels for vector-valued functions: A review. *arXiv:1106.6251* **2011**, *pre-print*.

[147] Borchani, H.; Varando, G.; Bielza, C.; Larranaga, P. A survey on multi-output regression. *Wiley Interdiscip. Rev.: Data Min. Knowl. Discov.* **2015**, *5*, 216–233.

[148] Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.

[149] Kuhn, M.; Johnson, K. *Applied Predictive Modelling*; Springer, 2013; Vol. 26; New York, USA.

[150] Mahoney, M. W.; Drineas, P. CUR matrix decompositions for improved data analysis. *Proc. Natl. Acad. Sci. U.S.A.* **2009**, *106*, 697–702.

[151] Yang, Y.; Pedersen, J. O. A comparative study on feature selection in text categorization. ICML. 1997; p 35, vol. 95, No. 412-420.

[152] Mallat, S. G.; Zhang, Z. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Process.* **1993**, *41*, 3397–3415.

[153] Kulis, B., et al. Metric learning: A survey. *Found. Trends Mach. Learn.* **2012**, *5*, 287–364.

[154] De Vazelhes, W.; Carey, C.; Tang, Y.; Vauquier, N.; Bellet, A. metric-learn: Metric Learning Algorithms in Python. *J. Mach. Learn. Res.* **2020**, *21*, 138–1.

[155] Yang, L.; Jin, R. Distance metric learning: A comprehensive survey. *Michigan State Universiy* **2006**, *2(2)*, 4.

[156] Weinberger, K. Q.; Tesauro, G. Metric learning for kernel regression. Artificial intelligence and statistics. 2007; pp 612–619, PMLR.

[157] Collobert, R.; Weston, J. Sparse metric learning via smooth optimization. 23rd Annual Conference on Advances in Neural Information Processing Systems, Vancouver. 2010; pp 2214–2222.

[158] Huang, R.; Sun, S. Kernel regression with sparse metric learning. *J. Intell. Fuzzy Syst.* **2013**, *24*, 775–787.

[159] Noh, Y.-K.; Sugiyama, M.; Kim, K.-E.; Park, F. C.; Lee, D. D. Generative Local Metric Learning for Kernel Regression. NIPS. 2017; pp 2452–2462.

[160] Kaya, M.; Bilge, H. Ş. Deep metric learning: A survey. *Symmetry* **2019**, *11*, 1066.

[161] Koch, G.; Zemel, R.; Salakhutdinov, R., et al. Siamese neural networks for one-shot image recognition. ICML deep learning workshop. 2015; Vol. 2.

[162] Hoffer, E.; Ailon, N. Deep metric learning using triplet network. International workshop on similarity-based pattern recognition. 2015; pp 84–92.

[163] Jeon, M.; Park, D.; Lee, J.; Jeon, H.; Ko, M.; Kim, S.; Choi, Y.; Tan, A.-C.; Kang, J. ReSimNet: drug response similarity prediction using Siamese neural networks. *Bioinformatics* **2019**, *35*, 5249–5256.

[164] Zheng, W.; Yang, L.; Genco, R. J.; Wactawski-Wende, J.; Buck, M.; Sun, Y. SENSE: Siamese neural network for sequence embedding and alignment-free comparison. *Bioinformatics* **2019**, *35*, 1820–1828.

[165] Frenkel, D.; Smit, B. *Understanding molecular simulation: From algorithms to applications*, 2nd ed.; Elsevier, 2001; London, UK.

[166] Robert, C.; Casella, G. *Monte Carlo statistical methods*; Springer Science & Business Media, 2013; Vol. 2; New York.

[167] Manousiouthakis, V. I.; Deem, M. W. Strict detailed balance is unnecessary in Monte Carlo simulation. *J. Chem. Phys.* **1999**, *110*, 2753–2756.

[168] Hansmann, U. H. Parallel tempering algorithm for conformational studies of biological molecules. *Chem. Phys. Lett.* **1997**, *281*, 140–150.

[169] Sugita, Y.; Okamoto, Y. Replica-exchange molecular dynamics method for protein folding. *Chem. Phys. Lett.* **1999**, *314*, 141–151.

[170] Liu, P.; Kim, B.; Friesner, R. A.; Berne, B. J. Replica exchange with solute tempering: A method for sampling biological systems in explicit water. *Proc. Natl. Acad. Sci. U.S.A.* **2005**, *102*, 13749–13754.

[171] Affentranger, R.; Tavernelli, I.; Di Iorio, E. E. A Novel Hamiltonian Replica Exchange MD Protocol to Enhance Protein Conformational Space Sampling. *J. Chem. Theory Comput.* **2006**, *2*, 217–228.

[172] Wang, L.; Friesner, R. A.; Berne, B. J. Replica Exchange with Solute Scaling: A More Efficient Version of Replica Exchange with Solute Tempering (REST2). *J. Phys. Chem. B* **2011**, *115*, 9431–9438.

[173] Swendsen, R. H.; Wang, J.-S. Replica Monte Carlo simulation of spin-glasses. *Phys. Rev. Lett.* **1986**, *57*, 2607.

[174] Earl, D. J.; Deem, M. W. Parallel tempering: Theory, applications, and new perspectives. *Phys. Chem. Chem. Phys.* **2005**, *7*, 3910–3916.

[175] Denschlag, R.; Lingenheil, M.; Tavan, P. Optimal temperature ladders in replica exchange simulations. *Chem. Phys. Lett.* **2009**, *473*, 193–195.

[176] Patriksson, A.; van der Spoel, D. A temperature predictor for parallel tempering simulations. *Phys. Chem. Chem. Phys.* **2008**, *10*, 2073–2077.

[177] Sindhikara, D. J.; Emerson, D. J.; Roitberg, A. E. Exchange often and properly in replica exchange molecular dynamics. *J. Chem. Theory Comput.* **2010**, *6*, 2804–2808.

[178] Sindhikara, D.; Meng, Y.; Roitberg, A. E. Exchange frequency in replica exchange molecular dynamics. *J. Chem. Phys.* **2008**, *128*, 01B609.

[179] Behler, J.; Martoňák, R.; Donadio, D.; Parrinello, M. Pressure-induced phase transitions in silicon studied by neural network-based metadynamics simulations. *Phys. Status Solidi B* **2008**, *245*, 2618–2629.

[180] Khaliullin, R.; Eshet, H.; Kühne, T.; Behler, J.; Parrinello, M. Graphite-diamond phase coexistence study employing a neural-network mapping of the ab initio potential energy surface. *Phys. Rev. B: Condens. Matter* **2010**, *81*.

[181] Khaliullin, R.; Eshet, H.; Kühne, T.; Behler, J.; Parrinello, M. Nucleation mechanism for the direct graphite-to-diamond phase transition. *Nat. Mater.* **2011**, *10*, 693–697.

[182] Eshet, H.; Khaliullin, R.; Kühne, T.; Behler, J.; Parrinello, M. Ab initio quality neural-network potential for sodium. *Phys. Rev. B: Condens. Matter* **2010**, *81*.

[183] Gastegger, M.; Kauffmann, C.; Behler, J.; Marquetand, P. Comparing the accuracy of high-dimensional neural network potentials and the systematic molecular fragmentation method: A benchmark study for all-trans alkanes. *J. Chem. Phys.* **2016**, *144*.

[184] Gastegger, M.; Behler, J.; Marquetand, P. Machine learning molecular dynamics for the simulation of infrared spectra. *Chem. Sci.* **2017**, *8*, 6924–6935.

[185] Caro, M. A.; Deringer, V. L.; Koskinen, J.; Laurila, T.; Csányi, G. Growth Mechanism and Origin of High sp3 Content in Tetrahedral Amorphous Carbon. *Phys. Rev. Lett.* **2018**, *120*, 166101.

[186] Deringer, V. L.; Bernstein, N.; Bartók, A. P.; Cliffe, M. J.; Kerber, R. N.; Marbella, L. E.; Grey, C. P.; Elliott, S. R.; Csányi, G. Realistic Atomistic Structure of Amorphous Silicon from Machine-Learning-Driven Molecular Dynamics. *J. Phys. Chem. Lett.* **2018**, *9*, 2879–2885.

[187] Hu, D.; Xie, Y.; Li, X.; Li, L.; Lan, Z. Inclusion of Machine Learning Kernel Ridge Regression Potential Energy Surfaces in On-the-Fly Nonadiabatic Molecular Dynamics Simulation. *J. Phys. Chem. Lett.* **2018**, *9*, 2725–2732.

[188] Smith, J. S.; Isayev, O.; Roitberg, A. E. ANI-1: An extensible neural network potential with DFT accuracy at force field computational cost. *Chem. Sci.* **2017**, *8*, 3192–3203.

[189] Smith, J. S.; Nebgen, B. T.; Zubatyuk, R.; Lubbers, N.; Devereux, C.; Barros, K.; Tretiak, S.; Isayev, O.; Roitberg, A. E. Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning. *Nat. Commun.* **2019**, *10*, 2903.

[190] Sauceda, H. E.; Chmiela, S.; Poltavsky, I.; Müller, K.-R.; Tkatchenko, A. Molecular force fields with gradient-domain machine learning: Construction and application to dynamics of small molecules with coupled cluster forces. *J. Chem. Phys.* **2019**, *150*, 114102.

[191] Schütt, K. T.; Arbabzadah, F.; Chmiela, S.; Müller, K. R.; Tkatchenko, A. Quantum-chemical insights from deep tensor neural networks. *Nat. Commun.* **2017**, *8*, 13890.

[192] Schütt, K. T.; Sauceda, H. E.; Kindermans, P.-J.; Tkatchenko, A.; Müller, K.-R. SchNet – A deep learning architecture for molecules and materials. *J. Chem. Phys.* **2018**, *148*, 241722.

[193] Unke, O. T.; Meuwly, M. PhysNet: a neural network for predicting energies, forces, dipole moments, and partial charges. *J. Chem. Theory Comput.* **2019**, *15*, 3678–3693.

[194] Gaus, M.; Cui, Q.; Elstner, M. DFTB3: Extension of the Self-Consistent-Charge Density-Functional Tight-Binding Method (SCC-DFTB). *J. Chem. Theory Comput.* **2011**, *7*, 931–948.

[195] Gaus, M.; Goez, A.; Elstner, M. Parametrization and Benchmark of DFTB3 for Organic Molecules. *J. Chem. Theory Comput.* **2013**, *9*, 338–354.

[196] Gaus, M.; Lu, X.; Elstner, M.; Cui, Q. Parameterization of DFTB3/3OB for Sulfur and Phosphorus for Chemical and Biological Applications. *J. Chem. Theory Comput.* **2014**, *10*, 1518–1537.

[197] Laio, A.; Parrinello, M. Escaping free-energy minima. *Proc. Natl. Acad. Sci. U.S.A.* **2002**, *99*, 12562–12566.

[198] Grimme, S. Exploration of chemical compound, conformer, and reaction space with meta-dynamics simulations based on tight-binding quantum chemical calculations. *J. Chem. Theory Comput.* **2019**, *15*, 2847–2862.

[199] Kaczor, A.; Reva, I. D.; Proniewicz, L. M.; Fausto, R. Importance of Entropy in the Conformational Equilibrium of Phenylalanine: A Matrix-Isolation Infrared Spectroscopy and Density Functional Theory Study. *J. Phys. Chem. A* **2006**, *110*, 2360–2370.

[200] Ess, D. H.; Wheeler, S. E.; Iafe, R. G.; Xu, L.; Çelebi ölçüm, N.; Houk, K. N. Bifurcations on Potential Energy Surfaces of Organic Reactions. *Angew. Chem. Int. Ed.* **2008**, *47*, 7592–7601.

[201] Rehbein, J.; Carpenter, B. K. Do we fully understand what controls chemical selectivity? *Phys. Chem. Chem. Phys.* **2011**, *13*, 20906.

[202] Schreiner, P. R.; Reisenauer, H. P.; Ley, D.; Gerbig, D.; Wu, C.-H.; Allen, W. D. Methylhydroxycarbene: Tunneling Control of a Chemical Reaction. *Science* **2011**, *332*, 1300–1303.

[203] Plata, R. E.; Singleton, D. A. A Case Study of the Mechanism of Alcohol-Mediated Morita Baylis–Hillman Reactions. The Importance of Experimental Observations. *J. Am. Chem. Soc.* **2015**, *137*, 3811–3826.

[204] Brémond, e.; Golubev, N.; Steinmann, S. N.; Corminboeuf, C. How important is self-consistency for the dDsC density dependent dispersion correction? *J. Chem. Phys.* **2014**, *140*, 18A516.

[205] Petraglia, R.; Steinmann, S. N.; Corminboeuf, C. A fast charge-Dependent atom-pairwise dispersion correction for DFTB3. *Int. J. Quantum Chem.* **2015**, *115*, 1265–1272.

[206] Mashraqui, S. H.; Sangvikar, Y. S.; Meetsma, A. Synthesis and structures of thieno[2,3-b]thiophene incorporated [3.3]dithiacyclophanes. Enhanced first hyperpolarizability in an unsymmetrically polarized cyclophane. *Tetrahedron Lett.* **2006**, *47*, 5599–5602.

[207] Dijkstra, G. D.; Kellogg, R. M.; Wynberg, H.; Svendsen, J. S.; Marko, I.; Sharpless, K. B. Conformational study of cinchona alkaloids. A combined NMR, molecular mechanics and x-ray approach. *J. Am. Chem. Soc.* **1989**, *111*, 8069.

[208] Zhou, J.; Wakchaure, V.; Kraft, P.; List, B. Primary-Amine-Catalyzed Enantioselective Intramolecular Aldolizations. *Angew. Chem. Int. Ed.* **2008**, *47*, 7656.

[209] Fabregat, R.; Fabrizio, A.; Corminboeuf, C. Modular Replica Exchange Simulatior. **2020**, http://doi.org/10.5281/zenodo.3630553.

[210] Elstner, M.; Hobza, P.; Frauenheim, T.; Suhai, S.; Kaxiras, E. Hydrogen bonding and stacking interactions of nucleic acid base pairs: A density-functional-theory based treatment. *J. Chem. Phys.* **2001**, *114*, 5149–5155.

[211] Aradi, B.; Hourahine, B.; Frauenheim, T. DFTB+, a Sparse Matrix-Based Implementation of the DFTB Method†. *J. Phys. Chem. A* **2007**, *111*, 5678–5684.

[212] Neese, F. The ORCA program system. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2012**, *2*, 73–78.

[213] Dunning, T. H. Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen. *J. Chem. Phys.* **1989**, *90*, 1007–1023.

[214] Kendall, R. A.; Dunning, T. H.; Harrison, R. J. Electron affinities of the first-row atoms revisited. Systematic basis sets and wave functions. *J. Chem. Phys.* **1992**, *96*, 6796–6806.

[215] Adamo, C.; Barone, V. Toward reliable density functional methods without adjustable parameters: The PBE0 model. *J. Chem. Phys.* **1999**, *110*, 6158–6170.

[216] Grimme, S.; Antony, J.; Ehrlich, S.; Krieg, H. A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu. *J. Chem. Phys.* **2010**, *132*, 154104.

[217] Ufimtsev, I. S.; Martinez, T. J. Quantum Chemistry on Graphical Processing Units. 3. Analytical Energy Gradients, Geometry Optimization, and First Principles Molecular Dynamics. *J. Chem. Theory Comput.* **2009**, *5*, 2619–2628.

[218] Titov, A. V.; Ufimtsev, I. S.; Luehr, N.; Martinez, T. J. Generating Efficient Quantum Chemistry Codes for Novel Architectures. *J. Chem. Theory Comput.* **2012**, *9*, 213–221.

[219] Pearlman, D. A.; Case, D. A.; Caldwell, J. W.; Ross, W. S.; Cheatham, T. E.; DeBolt, S.; Ferguson, D.; Seibel, G.; Kollman, P. AMBER, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structural and energetic properties of molecules. *Comput. Phys. Commun.* **1995**, *91*, 1–41.

[220] Christensen, A.; Faber, F.; Huang, B.; Bratholm, L.; Tkatchenko, A.; Müller, K.; von Lilienfeld, O. A. QML: A Python Toolkit for Quantum Machine Learning. **2017**, http://doi.org/10.5281/zenodo.817332.

[221] Nelder, J. A.; Mead, R. A simplex method for function minimization. *Comput. J.* **1965**, *7*, 308–313.

[222] Ceriotti, M.; More, J.; Manolopoulos, D. E. i-PI: A Python interface for ab initio path integral molecular dynamics simulations. *Comput. Phys. Commun.* **2014**, *185*, 1019–1026.

[223] Kapil, V. et al. i-PI 2.0: A universal force engine for advanced molecular simulations. *Comput. Phys. Commun.* **2018**, *236*, 214.

[224] Kunsch, H. R. The Jackknife and the Bootstrap for General Stationary Observations. *Ann. Statist.* **1989**, *17*, 1217–1241.

[225] Rathore, N.; Chopra, M.; De Pablo, J. Optimal allocation of replicas in parallel tempering simulations. *J. Chem. Phys.* **2005**, *122*.

[226] Vannay, L.; Meyer, B.; Petraglia, R.; Sforazzini, G.; Ceriotti, M.; Corminboeuf, C. Analyzing Fluxional Molecules Using DORI. *J. Chem. Theory Comput.* **2018**, *14*, 2370–2379.

[227] Kapil, V.; Engel, E.; Rossi, M.; Ceriotti, M. Assessment of Approximate Methods for Anharmonic Free Energies. *J. Chem. Theory Comput.* **2019**, *15*, 5845–5857.

[228] Bürgi, T.; Baiker, A. Conformational behavior of cinchonidine in different solvents: a combined NMR and ab initio investigation. *J. Am. Chem. Soc.* **1998**, *120*, 12920–12926.

[229] Huang, M.; Dissanayake, T.; Kuechler, E.; Radak, B. K.; Lee, T.-S.; Giese, T. J.; York, D. M. A multidimensional B-spline correction for accurate modeling sugar puckering in QM/MM simulations. *J. Chem. Theory Comput.* **2017**, *13*, 3975–3984.

[230] Fabregat, R.; Fabrizio, A.; Meyer, B.; Hollas, D.; Corminboeuf, C. Hamiltonian-Reservoir Replica Exchange and Machine Learning Potentials for Computational Organic Chemistry. *J. Chem. Theory Comput.* **2020**, *16*, 3084–3094.

[231] Schutt, K.; Kessel, P.; Gastegger, M.; Nicoli, K.; Tkatchenko, A.; Muller, K.-R. SchNetPack: A deep learning toolbox for atomistic systems. *J. Chem. Theory Comput.* **2018**, *15*, 448–455.

[232] Unke, O. T.; Meuwly, M. A reactive, scalable, and transferable model for molecular energies from a neural network approach based on local information. *J. Chem. Phys.* **2018**, *148*, 241708.

[233] Bartók, A. P.; Kermode, J.; Bernstein, N.; Csányi, G. Machine learning a general-purpose interatomic potential for silicon. *Phys. Rev. X* **2018**, *8*, 041048.

[234] Zhang, C.; Sun, Q. Gaussian approximation potential for studying the thermal conductivity of silicene. *J. Appl. Phys.* **2019**, *126*, 105103.

[235] Deringer, V. L.; Bernstein, N.; Csányi, G.; Mahmoud, C. B.; Ceriotti, M.; Wilson, M.; Drabold, D. A.; Elliott, S. R. Origins of structural and electronic transitions in disordered silicon. *Nature* **2021**, *589*, 59–64.

[236] Mocanu, F. C.; Konstantinou, K.; Lee, T. H.; Bernstein, N.; Deringer, V. L.; Csányi, G.; Elliott, S. R. Modeling the phase-change memory material, Ge2Sb2Te5, with a machine-learned interatomic potential. *J. Phys. Chem. B* **2018**, *122*, 8998–9006.

[237] Mocanu, F.; Konstantinou, K.; Elliott, S. Quench-rate and size-dependent behaviour in glassy Ge2Sb2Te5 models simulated with a machine-learned Gaussian approximation potential. *J. Phys. D: Appl. Phys.* **2020**, *53*, 244002.

[238] Balabin, R. M.; Lomakina, E. I. Support vector machine regression (LS-SVM)—an alternative to artificial neural networks (ANNs) for the analysis of quantum chemistry data? *Phys. Chem. Chem. Phys.* **2011**, *13*, 11710.

[239] Thompson, A.; Swiler, L.; Trott, C.; Foiles, S.; Tucker, G. Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials. *J. Comput. Phys.* **2015**, *285*, 316–330.

[240] Artrith, N.; Urban, A. An implementation of artificial neural-network potentials for atomistic materials simulations: Performance for TiO2. *Comput. Mater. Sci.* **2016**, *114*, 135–150.

[241] Kobayashi, R.; Giofré, D.; Junge, T.; Ceriotti, M.; Curtin, W. A. Neural network potential for Al-Mg-Si alloys. *Phys. Rev. Mat.* **2017**, *1*, 053604.

[242] Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D. R. Protein–ligand scoring with convolutional neural networks. *J. Chem. Inf. Model.* **2017**, *57*, 942–957.

[243] Janet, J. P.; Kulik, H. J. Predicting electronic structure properties of transition metal complexes with neural networks. *Chem. Sci.* **2017**, *8*, 5137–5152.

[244] Hellstrom, M.; Ceriotti, M.; Behler, J. Nuclear quantum effects in sodium hydroxide solutions from neural network molecular dynamics simulations. *J. Phys. Chem. B* **2018**, *122*, 10158–10171.

[245] Zhang, L.; Han, J.; Wang, H.; Car, R.; Weinan, E. Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics. *Phys. Rev. Lett.* **2018**, *120*, 143001.

[246] Noé, F.; Tkatchenko, A.; Müller, K.-R.; Clementi, C. Machine learning for molecular simulation. *Annu. Rev. Phys. Chem.* **2020**, *71*, 361–390.

[247] Rossi, K.; Jurásková, V.; Wischert, R.; Garel, L.; Corminbœuf, C.; Ceriotti, M. Simulating solvation and acidity in complex mixtures with first-principles accuracy: the case of CH3SO3H and H2O2 in phenol. *J. Chem. Theory Comput.* **2020**, *16*, 5139–5149.

[248] Fabrizio, A.; Meyer, B.; Fabregat, R.; Corminboeuf, C. Quantum Chemistry Meets Machine Learning. *CHIMIA* **2019**, *73*, 983–989.

[249] Fabrizio, A.; Briling, K. R.; Girardier, D. D.; Corminboeuf, C. Learning on-top: Regressing the on-top pair density for real-space visualization of electron correlation. *J. Chem. Phys.* **2020**, *153*, 204111.

[250] Ruddigkeit, L.; van Deursen, R.; Blum, L. C.; Reymond, J.-L. Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *J. Chem. Inf. Model.* **2012**, *52*, 2864–2875.

[251] Helfrecht, B. A.; Cersonsky, R. K.; Fraux, G.; Ceriotti, M. Structure-property maps with Kernel principal covariates regression. *Mach. Learn.: Sci. Technol.*. **2020**, *1*, 045021.

[252] Grimme, S.; Ehrlich, S.; Goerigk, L. Effect of the damping function in dispersion corrected density functional theory. *J. Comput. Chem.* **2011**, *32*, 1456–1465.

[253] Perdew, J. P.; Burke, K.; Ernzerhof, M. Generalized Gradient Approximation Made Simple. *Phys. Rev. Lett.* **1996**, *77*, 3865–3868.

[254] Steinmann, S. N.; Corminboeuf, C. A System-Dependent Density-Based Dispersion Correction. *J. Chem. Theory Comput.* **2010**, *6*, 1990–2001.

[255] Steinmann, S. N.; Corminboeuf, C. Comprehensive Benchmarking of a Density-Dependent Dispersion Correction. *J. Chem. Theory Comput.* **2011**, *7*, 3567–3577.

[256] Steinmann, S. N.; Corminboeuf, C. A generalized-gradient approximation exchange hole model for dispersion coefficients. *J. Chem. Phys.* **2011**, *134*, 044117.

[257] Nguyen, T. T.; Székely, E.; Imbalzano, G.; Behler, J.; Csányi, G.; Ceriotti, M.; Götz, A. W.; Paesani, F. Comparison of permutationally invariant polynomials, neural networks, and Gaussian approximation potentials in representing water interactions through many-body expansions. *J. Chem. Phys.* **2018**, *148*, 241725.

[258] Barker, J.; Bulin, J.; Hamaekers, J.; Mathias, S. *LC-GAP: Localized Coulomb Descriptors for the Gaussian Approximation Potential*; Springer Int. Pub.: Cham, 2017; pp 25–42.

[259] Sivaraman, G.; Krishnamoorthy, A. N.; Baur, M.; Holm, C.; Stan, M.; Csányi, G.; Benmore, C.; Vázquez-Mayagoitia, Á. Machine-learned interatomic potentials by active learning: amorphous and liquid hafnium dioxide. *Npj. Comput. Mater.* **2020**, *6*, 1–8.

[260] Behler, J. Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. *Phys. Chem. Chem. Phys.* **2011**, *13*, 17930–55.

[261] Musil, F.; Willatt, M. J.; Langovoy, M. A.; Ceriotti, M. Fast and Accurate Uncertainty Estimation in Chemical Machine Learning. *J. Chem. Theory Comput.* **2019**, *15*, 906.

[262] Imbalzano, G.; Zhuang, Y.; Kapil, V.; Rossi, K.; Engel, E. A.; Grasselli, F.; Ceriotti, M. Uncertainty estimation for molecular dynamics and sampling. *J. Chem. Phys.* **2021**, *154*, 074102.

[263] Eldar, Y.; Lindenbaum, M.; Porat, M.; Zeevi, Y. The farthest point strategy for progressive image sampling. *IEEE Trans. Image Process.* **1997**, *6*, 1305–1315.

[264] Richardson, J. S. *Advances in protein chemistry*; Elsevier, 1981; Vol. 34; pp 167–339.

[265] Ropo, M.; Schneider, M.; Baldauf, C.; Blum, V. First-principles data set of 45,892 isolated and cation-coordinated conformers of 20 proteinogenic amino acids. *Sci. Data* **2016**, *3*, 160009.

[266] Burns, L. A.; Faver, J. C.; Zheng, Z.; Marshall, M. S.; Smith, D. G. A.; Vanommeslaeghe, K.; MacKerell, A. D.; Merz, K. M.; Sherrill, C. D. The BioFragment Database (BFDb): An open-data platform for computational chemistry analysis of noncovalent interactions. *J. Chem. Phys.* **2017**, *147*, 161727.

[267] Giberti, F.; Tribello, G. A.; Ceriotti, M. Global Free-Energy Landscapes as a Smoothly Joined Collection of Local Maps. *J. Chem. Theory Comput.* **2021**, *17*, 3292–3308.

[268] Abrams, J. B.; Tuckerman, M. E. Efficient and direct generation of multidimensional free energy surfaces via adiabatic dynamics without coordinate transformations. *J. Phys. Chem. B* **2008**, *112*, 15742–15757.

[269] Chen, M.; Cuendet, M. A.; Tuckerman, M. E. Heating and flooding: A unified approach for rapid generation of free energy surfaces. *J. Chem. Phys.* **2012**, *137*, 024102.

[270] Gasparotto, P.; Meißner, R. H.; Ceriotti, M. Recognizing Local and Global Structural Motifs at the Atomic Scale. *J. Chem. Theory Comput.* **2018**, *14*, 486–498.

[271] Giberti, F.; Cheng, B.; Tribello, G. A.; Ceriotti, M. Iterative unbiasing of quasi-equilibrium sampling. *J. Chem. Theory Comput.* **2019**, *16*, 100–107.

[272] Schmidt, M. W.; Baldridge, K. K.; Boatz, J. A.; Elbert, S. T.; Gordon, M. S.; Jensen, J. H.; Koseki, S.; Matsunaga, N.; Nguyen, K. A.; Su, S., et al. General atomic and molecular electronic structure system. *J. Comput. Chem.* **1993**, *14*, 1347–1363.

[273] Gordon, M. S.; Schmidt, M. W. *Theory and applications of computational chemistry*; Elsevier, 2005; pp 1167–1189.

[274] The PLUMED consortium, Promoting transparency and reproducibility in enhanced molecular simulations. *Nat. Methods* **2019**, *16*, 670.

[275] Tribello, G. A.; Bonomi, M.; Branduardi, D.; Camilloni, C.; Bussi, G. PLUMED2: New feathers for an old bird. *Comput. Phys. Commun.* **2014**, *185*, 604.

[276] Plimpton, S. Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.* **1995**, *117*, 1–19.

[277] Ceriotti, M.; Bussi, G.; Parrinello, M. Colored-noise thermostats à la carte. *J. Chem. Theory Comput.* **2010**, *6*, 1170–1180.

[278] Ceriotti, M.; Manolopoulos, D. E.; Parrinello, M. Accelerating the convergence of path integral dynamics with a generalized Langevin equation. *J. Chem. Phys.* **2011**, *134*, 084104.

[279] Fabregat, R.; Fabrizio, A.; Corminboeuf, C. Local Kernel Regression. **2021**, https://doi.org/10.5281/zenodo.5172581.

[280] Singraber, A.; Behler, J.; Dellago, C. Library-based LAMMPS implementation of high-dimensional neural network potentials. *J. Chem. Theory Comput.* **2019**, *15*, 1827–1840.

[281] Cersonsky, R. K.; Helfrecht, B.; Engel, E. A.; Kliavinek, S.; Ceriotti, M. Improving sample and feature selection with principal covariates regression. *Mach. Learn.: Sci. Technol.*. **2021**, *2*, 035038.

[282] Engel, E. A.; Kapil, V.; Ceriotti, M. Importance of Nuclear Quantum Effects for NMR Crystallography. *J. Phys. Chem. Lett.* **2021**, *12*, 7701–7707.

[283] Kapil, V.; Engel, E. A. A complete description of thermodynamic stabilities of molecular crystals. *arXiv preprint arXiv:2102.13598* **2021**,

[284] Cheng, B.; Engel, E. A.; Behler, J.; Dellago, C.; Ceriotti, M. Ab initio thermodynamics of liquid and solid water. *Proc. Natl. Acad. Sci. U.S.A.* **2019**, *116*, 1110–1115.

[285] Imbalzano, G.; Ceriotti, M. Modeling the Ga/As binary system across temperatures and compositions from first principles. *Phys. Rev. Mater.* **2021**, *5*, 063804.

[286] Fabregat, R.; Blaskovits, J. T.; Corminboeuf, C. MolView: A python based web app framework to visualize and share chemical data. **2020**, http://doi.org/10.5281/zenodo.4564039.

[287] Valdes, H.; Pluhackova, K.; Hobza, P. Phenylalanyl-Glycyl-Phenylalanine Tripeptide: A Model System for Aromatic-Aromatic Side Chain Interactions in Proteins. *J. Chem. Theory Comput.* **2009**, *5*, 2248–2256.

[288] Blyth, C. R. On Simpson's paradox and the sure-thing principle. *J. Am. Stat. Assoc.* **1972**, *67*, 364–366.

[289] Miriyala, V. M.; Řezáč, J. Description of non-covalent interactions in SCC-DFTB methods. *J. Comput. Chem.* **2017**, *38*, 688–697.

[290] Řezáč, J.; Fanfrlík, J.; Salahub, D.; Hobza, P. Semiempirical Quantum Chemical PM6 Method Augmented by Dispersion and H-Bonding Correction Terms Reliably Describes Various Types of Noncovalent Complexes. *J. Chem. Theory Comput.* **2009**, *5*, 1749–1760.

[291] Korth, M.; Pitoňák, M.; Řezáč, J.; Hobza, P. A Transferable H-Bonding Correction for Semiempirical Quantum-Chemical Methods. *J. Chem. Theory Comput.* **2010**, *6*, 344–352.

[292] Korth, M. Third-Generation Hydrogen-Bonding Corrections for Semiempirical QM Methods and Force Fields. *J. Chem. Theory Comput.* **2010**, *6*, 3808–3816.

[293] Řezáč, J.; Hobza, P. Advanced Corrections of Hydrogen Bonding and Dispersion for Semiempirical Quantum Mechanical Methods. *J. Chem. Theory Comput.* **2012**, *8*, 141–151.

[294] Řezáč, J. Empirical Self-Consistent Correction for the Description of Hydrogen Bonds in DFTB3. *J. Chem. Theory Comput.* **2017**, *13*, 4804–4817.

[295] Taylor, M. S.; Jacobsen, E. N. Asymmetric catalysis in complex target synthesis. *Proc. Natl. Acad. Sci. U.S.A.* **2004**, *101*, 5368–5373.

[296] Gawley, R. E.; Aubé, J. *Principles of asymmetric synthesis*; Elsevier, 2012; pp 63–95.

[297] MacMillan, D. W. C. The advent and development of organocatalysis. *Nature* **2008**, *455*, 304–308.

[298] Dalko, P. I.; Moisan, L. Enantioselective organocatalysis. *Angew. Chem. Int. Ed.* **2001**, *40*, 3726–3748.

[299] Dalko, P. I. *Enantioselective Organocatalysis: Reactions and Experimental Procedures*; John Wiley & Sons, 2007; pp 1–536.

[300] Xiang, S.-H.; Tan, B. Advances in asymmetric organocatalysis over the last 10 years. *Nat. Commun.* **2020**, *11*.

[301] Poree, C.; Schoenebeck, F. A Holy Grail in Chemistry: Computational Catalyst Design: Feasible or Fiction? *Acc. Chem. Res.* **2017**, *50*, 605–608.

[302] Houk, K. N.; Liu, F. Holy grails for computational organic chemistry and biochemistry. *Acc. Chem. Res.* **2017**, *50*, 539–543.

[303] Wheeler, S. E.; Seguin, T. J.; Guan, Y.; Doney, A. C. Noncovalent Interactions in Organocatalysis and the Prospect of Computational Catalyst Design. *Acc. Chem. Res.* **2016**, *49*, 1061–1069.

[304] Peng, Q.; Duarte, F.; Paton, R. S. Computing organic stereoselectivity-from concepts to quantitative calculations and predictions. *Chem. Soc. Rev.* **2016**, *45*, 6093–6107.

[305] Hansen, E.; Rosales, A. R.; Tutkowski, B.; Norrby, P. .; Wiest, O. Prediction of Stereochemistry using Q2MM. *Acc. Chem. Res.* **2016**, *49*, 996–1005.

[306] Hopmann, K. H. Quantum chemical studies of asymmetric reactions: Historical aspects and recent examples. *Int. J. Quantum Chem.* **2015**, *115*, 1232–1249.

[307] Tsang, A. S. .; Sanhueza, I. A.; Schoenebeck, F. Combining Experimental and Computational Studies to Understand and Predict Reactivities of Relevance to Homogeneous Catalysis. *Chem. Eur. J.* **2014**, *20*, 16432–16441.

[308] Sepulveda, D.; Lu, T.; Wheeler, S. E. Performance of DFT methods and origin of stereoselectivity in bipyridine N,N-dioxide catalyzed allylation and propargylation reactions. *Org. Biomol. Chem.* **2014**, *12*, 8346–8353.

[309] Balcells, D.; Clot, E.; Eisenstein, O.; Nova, A.; Perrin, L. Deciphering Selectivity in Organic Reactions: A Multifaceted Problem. *Acc. Chem. Res.* **2016**, *49*, 1070–1078.

[310] Cheong, P. H. .; Legault, C. Y.; Um, J. M.; Çelebi Ölçüm, N.; Houk, K. N. Quantum mechanical investigations of organocatalysis: Mechanisms, reactivities, and selectivities. *Chem. Rev.* **2011**, *111*, 5042–5137.

[311] Sperger, T.; Sanhueza, I. A.; Kalvet, I.; Schoenebeck, F. Computational Studies of Synthetically Relevant Homogeneous Organometallic Catalysis Involving Ni, Pd, Ir, and Rh: An Overview of Commonly Employed DFT Methods and Mechanistic Insights. *Chem. Rev.* **2015**, *115*, 9532–9586.

[312] Foscato, M.; Jensen, V. R. Automated in Silico Design of Homogeneous Catalysts. *ACS Catal.* **2020**, *10*, 2354–2377.

[313] Ingman, V. M.; Schaefer, A. J.; Andreola, L. R.; Wheeler, S. E. QChASM: Quantum chemistry automation and structure manipulation. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2021**, *11*, e1510.

[314] Guan, Y.; Ingman, V. M.; Rooks, B. J.; Wheeler, S. E. AARON: An Automated Reaction Optimizer for New Catalysts. *J. Chem. Theory Comput.* **2018**, *14*, 5249–5261.

[315] Rosales, A. R.; Wahlers, J.; Limé, E.; Meadows, R. E.; Leslie, K. W.; Savin, R.; Bell, F.; Hansen, E.; Helquist, P.; Munday, R. H.; Wiest, O.; Norrby, P. . Rapid virtual screening of enantioselective catalysts using CatVS. *Nat. Catal.* **2019**, *2*, 41–45.

[316] Oslob, J. D.; Åkermark, B.; Helquist, P.; Norrby, P. . Steric influences on the selectivity in palladium-catalyzed allylation. *Organometallics* **1997**, *16*, 3015–3021.

[317] Lipkowitz, K. B.; Pradhan, M. Computational studies of chiral catalysts: A Comparative Molecular Field Analysis of an asymmetric Diels-Alder reaction with catalysts containing bisoxazoline or phosphinooxazoline ligands. *J. Org. Chem.* **2003**, *68*, 4648–4656.

[318] Harper, K. C.; Sigman, M. S. Three-dimensional correlation of steric and electronic free energy relationships guides asymmetric propargylation. *Science* **2011**, *333*, 1875–1878.

[319] Sigman, M. S.; Harper, K. C.; Bess, E. N.; Milo, A. The Development of Multidimensional Analysis Tools for Asymmetric Catalysis and beyond. *Acc. Chem. Res.* **2016**, *49*, 1292–1301.

[320] Reid, J. P.; Sigman, M. S. Comparing quantitative prediction methods for the discovery of small-molecule chiral catalysts. *Nat. Rev. Chem.* **2018**, *2*, 290–305.

[321] Harper, K. C.; Sigman, M. S. Using physical organic parameters to correlate asymmetric catalyst performance. *J. Org. Chem.* **2013**, *78*, 2813–2818.

[322] Santiago, C. B.; Guo, J. .; Sigman, M. S. Predictive and mechanistic multivariate linear regression models for reaction development. *Chem. Sci.* **2018**, *9*, 2398–2412.

[323] Durand, D. J.; Fey, N. Computational Ligand Descriptors for Catalyst Design. *Chem. Rev.* **2019**, *119*, 6561–6594.

[324] Milo, A.; Bess, E. N.; Sigman, M. S. Interrogating selectivity in catalysis using molecular vibrations. *Nature* **2014**, *507*, 210–214.

[325] Denmark, S. E.; Gould, N. D.; Wolf, L. M. A systematic investigation of quaternary ammonium ions as asymmetric phase-transfer catalysts. Application of quantitative structure activity/selectivity relationships. *J. Org. Chem.* **2011**, *76*, 4337–4357.

[326] Milo, A.; Neel, A. J.; Toste, F. D.; Sigman, M. S. A data-intensive approach to mechanistic elucidation applied to chiral anion catalysis. *Science* **2015**, *347*, 737–743.

[327] Bess, E. N.; Bischoff, A. J.; Sigman, M. S.; Jacobsen, E. N. Designer substrate library for quantitative, predictive modeling of reaction performance. *Proc. Natl. Acad. Sci. U.S.A.* **2014**, *111*, 14698–14703.

[328] Werth, J.; Sigman, M. S. Connecting and Analyzing Enantioselective Bifunctional Hydrogen Bond Donor Catalysis Using Data Science Tools. *J. Am. Chem. Soc.* **2020**, *142*, 16382–16391.

[329] Reid, J. P.; Sigman, M. S. Holistic prediction of enantioselectivity in asymmetric catalysis. *Nature* **2019**, *571*, 343–348.

[330] Zahrt, A. F.; Athavale, S. V.; Denmark, S. E. Quantitative Structure-Selectivity Relationships in Enantioselective Catalysis: Past, Present, and Future. *Chem. Rev.* **2020**, *120*, 1620–1689.

[331] Mitchell B.O., J. B. O. Machine learning methods in chemoinformatics. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2014**, *4*, 468–481.

[332] Funes-Ardoiz, I.; Schoenebeck, F. Established and Emerging Computational Tools to Study Homogeneous Catalysis—From Quantum Mechanics to Machine Learning. *Chem* **2020**, *6*, 1904–1913.

[333] Kitchin, J. R. Machine learning in catalysis. *Nat. Catal.* **2018**, *1*, 230–232.

[334] Yang, W.; Fidelis, T. T.; Sun, W. . Machine Learning in Catalysis, from Proposal to Practicing. *ACS Omega* **2020**, *5*, 83–88.

[335] Li, Z.; Wang, S.; Xin, H. Toward artificial intelligence in catalysis. *Nat. Catal.* **2018**, *1*, 641–642.

[336] Wodrich, M. D.; Fabrizio, A.; Meyer, B.; Corminboeuf, C. Data-powered augmented volcano plots for homogeneous catalysis. *Chem. Sci.* **2020**, *11*, 12070–12080.

[337] RDKit: A software suite for cheminformatics, computational chemistry, and predictive modeling. 2013; http://www.rdkit.org.

[338] Zahrt, A. F.; Henle, J. J.; Rose, B. T.; Wang, Y.; Darrow, W. T.; Denmark, S. E. Prediction of higher-selectivity catalysts by computer-driven workflow and machine learning. *Science* **2019**, *363*.

[339] Henle, J. J.; Zahrt, A. F.; Rose, B. T.; Darrow, W. T.; Wang, Y.; Denmark, S. E. Development of a Computer-Guided Workflow for Catalyst Optimization. Descriptor Validation, Subset Selection, and Training Set Analysis. *J. Am. Chem. Soc.* **2020**, *142*, 11578–11592.

[340] Tomberg, A.; Johansson, M. J.; Norrby, P. . A Predictive Tool for Electrophilic Aromatic Substitutions Using Machine Learning. *J. Org. Chem.* **2019**, *84*, 4695–4703.

[341] Singh, S.; Pareek, M.; Changotra, A.; Banerjee, S.; Bhaskararao, B.; Balamurugan, P.; Sunoj, R. B. A unified machine-learning protocol for asymmetric catalysis as a proof of concept demonstration using asymmetric hydrogenation. *Proc. Natl. Acad. Sci. U.S.A.* **2020**, *117*, 1339–1345.

[342] Chen, J.; Jiwu, W.; Mingzong, L.; You, T. Calculation on enantiomeric excess of catalytic asymmetric reactions of diethylzinc addition to aldehydes with topological indices and artificial neural network. *J. Mol. Cat. A Chem.* **2006**, *258*, 191–197.

[343] Amar, Y.; Schweidtmann, A. M.; Deutsch, P.; Cao, L.; Lapkin, A. Machine learning and molecular descriptors enable rational solvent selection in asymmetric catalysis. *Chem. Sci.* **2019**, *10*, 6697–6706.

[344] Banerjee, S.; Sreenithya, A.; Sunoj, R. B. Machine learning for predicting product distributions in catalytic regioselective reactions. *Phys. Chem. Chem. Phys.* **2018**, *20*, 18311–18318.

[345] Beker, W.; Gajewska, E. P.; Badowski, T.; Grzybowski, B. A. Prediction of Major Regio-, Site-, and Diastereoisomers in Diels–Alder Reactions by Using Machine-Learning: The Importance of Physically Meaningful Descriptors. *Angew. Chem. Int. Ed.* **2019**, *58*, 4515–4519.

[346] Maley, S.; Kwon, D. .; Rollins, N.; Stanley, J. C.; Sydora, O. L.; Bischof, S. M.; Ess, D. H. Quantum-mechanical transition-state model combined with machine learning provides catalyst design features for selective Crolefin oligomerization. *Chem. Sci.* **2020**, *11*, 9665–9674.

[347] Dhayalan, V.; Gadekar, S. C.; Alassad, Z.; Milo, A. Unravelling mechanistic features of organocatalysis with in situ modifications at the secondary sphere. *Nat. Chem.* **2019**, *11*, 543–551.

[348] Ahneman, D. T.; Estrada, J. G.; Lin, S.; Dreher, S. D.; Doyle, A. G. Predicting reaction performance in C–N cross-coupling using machine learning. *Science* **2018**, *360*, 186–190.

[349] Nielsen, M. K.; Ahneman, D. T.; Riera, O.; Doyle, A. G. Deoxyfluorination with sulfonyl fluorides: Navigating reaction space with machine learning. *J. Am. Chem. Soc.* **2018**, *140*, 5004–5008.

[350] Jorner, K.; Brinck, T.; Norrby, P. .; Buttar, D. Machine learning meets mechanistic modelling for accurate prediction of experimental activation energies. *Chem. Sci.* **2021**, *12*, 1163–1175.

[351] Sandfort, F.; Strieth-Kalthoff, F.; Kühnemund, M.; Beecks, C.; Glorius, F. A Structure-Based Platform for Predicting Chemical Reactivity. *Chem* **2020**, *6*, 1379–1390.

[352] Granda, J. M.; Donina, L.; Dragone, V.; Long, D. .; Cronin, L. Controlling an organic synthesis robot with machine learning to search for new reactivity. *Nature* **2018**, *559*, 377–381.

[353] Coley, C. W.; Jin, W.; Rogers, L.; Jamison, T. F.; Jaakkola, T. S.; Green, W. H.; Barzilay, R.; Jensen, K. F. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chem. Sci.* **2019**, *10*, 370–377.

[354] Friederich, P.; dos Passos Gomes, G.; De Bin, R.; Aspuru-Guzik, A.; Balcells, D. Machine learning dihydrogen activation in the chemical space surrounding Vaska's complex. *Chem. Sci.* **2020**, *11*, 4584–4601.

[355] Heinen, S.; von Rudorff, G. F.; von Lilienfeld, O. A. Toward the design of chemical reactions: Machine learning barriers of competing mechanisms in reactant space. *J. Chem. Phys.* **2021**, *155*, 064105.

[356] von Rudorff, G. F.; Heinen, S. N.; Bragato, M.; von Lilienfeld, O. A. Thousands of reactants and transition states for competing E2 and S2 reactions. *Mach. Learn.: Sci. Technol.* **2020**, *1*, 045026.

[357] Bragato, M.; von Rudorff, G. F.; von Lilienfeld, O. A. Data enhanced Hammett-equation: reaction barriers in chemical space. *Chem. Sci.* **2020**, *11*, 11859–11868.

[358] Skoraczyñski, G.; DIttwald, P.; Miasojedow, B.; Szymkuc, S.; Gajewska, E.; Grzybowski, B.; Gambin, A. Predicting the outcomes of organic reactions via machine learning: Are current descriptors sufficient? *Sci. Rep.* **2017**, *7(1)*, 1–9.

[359] von Lilienfeld, O. A.; Burke, K. Retrospective on a decade of machine learning for chemical discovery. *Nat. Commun.* **2020**, *11*, 1–4.

[360] Li, X.; Zhang, S. .; Xu, L. .; Hong, X. Predicting Regioselectivity in Radical C-H Functionalization of Heterocycles through Machine Learning. *Angew. Chem. Int. Ed.* **2020**, *59*, 13253–13259.

[361] Doney, A. C.; Rooks, B. J.; Lu, T.; Wheeler, S. E. Design of organocatalysts for asymmetric propargylations through computational screening. *ACS Catal.* **2016**, *6*, 7948–7955.

[362] Rooks, B.; Haas, M.; Sepúlveda, D.; Lu, T.; Wheeler, S. Prospects for the Computational Design of Bipyridine N, N '-Dioxide Catalysts for Asymmetric Propargylation Reactions. *ACS Catal.* **2015**, *5*, 272–280.

[363] Denmark, S. E.; Coe, D. M.; Pratt, N. E.; Griedel, B. D. Asymmetric Allylation of Aldehydes with Chiral Lewis Bases. *J. Org. Chem.* **1994**, *59*, 6161–6163.

[364] Denmark, S. E.; Fu, J. On the mechanism of catalytic, enantioselective allylation of aldehydes with chlorosilanes and chiral Lewis bases [14]. *J. Am. Chem. Soc.* **2000**, *122*, 12021–12022.

[365] Denmark, S. E.; Wynn, T. Lewis base activation of Lewis acids catalytic enantioselective allylation and propargylation of aldehydes. *J. Am. Chem. Soc.* **2001**, *123*, 6199–6200.

[366] Denmark, S. E.; Beutner, G. L. Lewis base catalysis in organic synthesis. *Angew. Chem. Int. Ed.* **2008**, *47*, 1560–1638.

[367] Ding, C. .; Hou, X. . Catalytic asymmetric propargylation. *Chem. Rev.* **2011**, *111*, 1914–1937.

[368] Marshall, J. A. Chiral allylic and allenic metal reagents for organic synthesis. *J. Org. Chem.* **2007**, *72*, 8153–8166.

[369] Nakajima, M.; Saito, M.; Shiro, M.; Hashimoto, S.-I. (S)-3,3'-dimethyl-2,2'-biquinoline N,N'-dioxide as an efficient catalyst for enantioselective addition of allyltrichlorosilanes to aldehydes. *J. Am. Chem. Soc.* **1998**, *120*, 6419–6420.

[370] Nakajima, M.; Saito, M.; Hashimoto, S. Selective synthesis of optically active allenic and homopropargylic alcohols from propargyl chloride. *Tetrahedron Asymmetry* **2002**, *13*, 2449–2452.

[371] Chen, J.; Captain, B.; Takenaka, N. Helical chiral 2,2'-bipyridine N-monoxides as catalysts in the enantioselective propargylation of aldehydes with allenyltrichlorosilane. *Org. Lett.* **2011**, *13*, 1654–1657.

[372] Lu, T.; Porterfield, M.; Wheeler, S. Explaining the disparate stereoselectivities of n-oxide catalyzed allylations and propargylations of aldehydes. *Org. Lett.* **2012**, *14*, 5310–5313.

[373] Lu, T.; Zhu, R.; An, Y.; Wheeler, S. Origin of enantioselectivity in the propargylation of aromatic aldehydes catalyzed by helical N-oxides. *J. Am. Chem. Soc.* **2012**, *134*, 3095–3102.

[374] Vu, K.; Snyder, J.; Li, L.; Rupp, M.; Chen, B.; Khelif, T.; Müller, K.-R.; Burke, K. Understanding kernel ridge regression: Common behaviors from simple functions to density functionals. *Int. J. Quantum Chem.* **2015**, *115*, 1115–1128.

[375] Hansen, K.; Montavon, G.; Biegler, F.; Fazli, S.; Rupp, M.; Scheffler, M.; von Lilienfeld, O. A.; Tkatchenko, A.; Müller, K.-R. Assessment and Validation of Machine Learning Methods for Predicting Molecular Atomization Energies. *J. Chem. Theory Comput.* **2013**, *9*, 3404–3419.

[376] Westermayr, J.; Faber, F.; Christensen, A.; von Lilienfeld, O. A.; Marquetand, P. Neural networks and kernel ridge regression for excited states dynamics of CH 2 NH 2 + : From single-state to multi-state representations and multi-property machine learning models. *Mach. Learn.: Sci. Technol.* **2020**, *1*, 025009.

[377] Nguyen, Q.; De, S.; Lin, J.; Cevher, V. Chemical machine learning with kernels: The impact of loss functions. *Int. J. Quantum Chem.* **2019**, *119(9)*, e25872.

[378] Becke, A. Density-functional thermochemistry. V. Systematic optimization of exchange-correlation functionals. *J. Chem. Phys.* **1997**, *107*, 8554–8560.

[379] Schafer, A.; Huber, C.; Ahlrichs, R. Fully optimized contracted Gaussian basis sets of triple zeta valence quality for atoms Li to Kr. *J. Chem. Phys.* **1994**, *100*, 5829–5835.

[380] Grimme, S. Semiempirical hybrid density functional with perturbative second-order correlation. *J. Chem. Phys.* **2006**, *124*, 034108.

[381] Cancés, E.; Mennucci, B. New applications of integral equations methods for solvation continuum models: Ionic solutions and liquid crystals. *J. Math. Chem.* **1998**, *23*, 309–326.

[382] Cancés, E.; Mennucci, B.; Tomasi, J. A new integral equation formalism for the polarizable continuum model: Theoretical background and applications to Isotropic and anisotropic dielectrics. *J. Chem. Phys.* **1997**, *107*, 3032–3041.

[383] Tomasi, J.; Mennucci, B.; Cammi, R. Quantum mechanical continuum solvation models. *Chem. Rev.* **2005**, *105*, 2999–3094.

[384] Frisch, M.; Trucks, G.; Schlegel, H.; Scuseria, G.; Robb, M.; Cheeseman, J.; Scalmani, G.; Barone, V.; Petersson, G.; Nakatsuji, H., et al. Gaussian 16 rev. *Gaussian Inc.: Wallingford* **2016**, CT, USA.

[385] Fukui, K. The Path of Chemical Reactions - The IRC Approach. *Acc. Chem. Res.* **1981**, *14*, 363–368.

[386] Hammond, G. A Correlation of Reaction Rates. *J. Am. Chem. Soc.* **1955**, *77*, 334–338.

[387] Ross, B. Mutual information between discrete and continuous data sets. *PLoS ONE* **2014**, *9*, e87357.

[388] Malkov, A.; Westwater, M.-M.; Gutnov, A.; Ramírez-López, P.; Friscourt, F.; Kadlčíková, A.; Hodačová, J.; Rankovic, Z.; Kotora, M.; Kočovský, P. New pyridine N-oxides as chiral organocatalysts in the asymmetric allylation of aromatic aldehydes. *Tetrahedron* **2008**, *64*, 11335–11348.

[389] Vaganov, V.; Fukazawa, Y.; Kondratyev, N.; Shipilovskikh, S.; Wheeler, S.; Rubtsov, A.; Malkov, A. Optimization of Catalyst Structure for Asymmetric Propargylation of Aldehydes with Allenyltrichlorosilane. *Adv. Synth. Catal.* **2020**, *362*, 5467–5474.

[390] Suo, Q.; Zhong, W.; Ma, F.; Ye, Y.; Huai, M.; Zhang, A. Multi-task sparse metric learning for monitoring patient similarity progression. 2018 IEEE International Conference on Data Mining (ICDM). 2018; pp 477–486.

[391] Sun, F.-Y.; Hoffmann, J.; Verma, V.; Tang, J. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. *arXiv:1908.01000* **2019**, *pre-print*.

[392] Hao, Z.; Lu, C.; Huang, Z.; Wang, H.; Hu, Z.; Liu, Q.; Chen, E.; Lee, C. ASGN: An active semi-supervised graph neural network for molecular property prediction. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020; pp 731–752.

[393] Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; Raffel, C. Mixmatch: A holistic approach to semi-supervised learning. *arXiv:1905.02249* **2019**, *pre-print*.

[394] Xie, Q.; Luong, M.-T.; Hovy, E.; Le, Q. V. Self-training with noisy student improves imagenet classification. Proceeding IEEE: CVF Conference on Computer Vision and Pattern Recognition. 2020; pp 10687–10698.

[395] Van Engelen, J. E.; Hoos, H. H. A survey on semi-supervised learning. *Mach. Learn.* **2020**, *109*, 373–440.

[396] Wang, M.; Hua, X.-S.; Song, Y.; Dai, L.-R.; Zhang, H.-J. Semi-supervised kernel regression. Sixth International Conference on Data Mining (ICDM'06). 2006; pp 1130–1135.

[397] Kostopoulos, G.; Karlos, S.; Kotsiantis, S.; Ragos, O. Semi-supervised regression: A recent review. *J. Intell. Fuzzy Syst.* **2018**, *35*, 1483–1500.

[398] Jean, N.; Xie, S. M.; Ermon, S. Semi-supervised deep kernel learning: Regression with unlabeled data by minimizing predictive variance. *arXiv:1805.10407* **2018**, *pre-print*.

[399] Fabregat, R.; Pustelnik, N.; Gonçalves, P.; Borgnat, P. Solving NMF with smoothness and sparsity constraints using PALM. *arXiv:1910.14576* **2019**, *pre-print*.

[400] Cichocki, A.; Zdunek, R.; Phan, A. H.; Amari, S.-i. *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*; John Wiley & Sons, 2009.

[401] Kruskal, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* **1964**, *29*, 1–27.

[402] Kruskal, J. B. Nonmetric multidimensional scaling: a numerical method. *Psychometrika* **1964**, *29*, 115–129.

[403] Tenenbaum, J. B. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* **2000**, *290*, 2319–2323.

[404] Ceriotti, M.; Tribello, G. A.; Parrinello, M. Demonstrating the transferability and the descriptive power of sketch-map. *J. Chem. Theory Comput.* **2013**, *9*, 1521–1532.

[405] Roweis, S. T. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* **2000**, *290*, 2323–2326.

[406] Zhang, Z.; Wang, J. MLLE: Modified locally linear embedding using multiple weights. Advances in neural information processing systems. 2007; pp 1593–1600.

[407] Donoho, D. L.; Grimes, C. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proc. Natl. Acad. Sci. U.S.A.* **2003**, *100*, 5591–5596.

[408] Zhang, Z.; Zha, H. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM J. Sci. Comput.* **2004**, *26*, 313–338.

[409] Thorndike, R. L. Who belongs in the family? *Psychometrika* **1953**, *18*, 267–276.

[410] Ketchen, D. J.; Shook, C. L. The application of cluster analysis in strategic management research: an analysis and critique. *Strateg. Manag. J.* **1996**, *17*, 441–458.

[411] Dempster, A. P.; Laird, N. M.; Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc.* **1977**, *39*, 1–22.

[412] Kriegel, H.-P.; Kröger, P.; Sander, J.; Zimek, A. Density-based clustering. *Wiley Interdiscip. Rev. Data Min. Knowl. Discovery* **2011**, *1*, 231–240.

[413] Ankerst, M.; Breunig, M. M.; Kriegel, H.-P.; Sander, J. OPTICS: Ordering points to identify the clustering structure. *ACM Sigmod Rec.* **1999**, *28*, 49–60.

[414] Boyd, S.; Boyd, S. P.; Vandenberghe, L. *Convex optimization*; Cambridge university press, 2004.

[415] Polikar, R. Ensemble based systems in decision making. *IEEE Circuits Syst. Mag.* **2006**, *6*, 21–45.

[416] Ho, T. K. Random decision forests. Proceedings of 3rd international conference on document analysis and recognition. 1995; pp 278–282.

[417] Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32.

[418] Freund, Y.; Schapire, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139.

[419] Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016; pp 785–794.

[420] Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 3146–3154.

[421] Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep learning*; MIT press Cambridge, 2016; Vol. 1.

[422] Balaji, A.; Allen, A. Benchmarking automatic machine learning frameworks. *arXiv:1808.06492* **2018**, *pre-print*.

[423] Olson, R. S.; Moore, J. H. TPOT: A tree-based pipeline optimization tool for automating machine learning. Workshop on automatic machine learning. 2016; pp 66–74.

[424] Feurer, M.; Klein, A.; Eggensperger, K.; Springenberg, J. T.; Blum, M.; Hutter, F. *Automated Machine Learning*; Springer, Cham, 2019; pp 113–134.

[425] Feurer, M.; Eggensperger, K.; Falkner, S.; Lindauer, M.; Hutter, F. Auto-sklearn 2.0: The next generation. *arXiv:2007.04074* **2020**, *pre-print*.

[426] Fraux, G.; Cersonsky, R. K.; Ceriotti, M. Chemiscope: Interactive structure-property explorer for materials and molecules. *J. Open Source Softw.* **2020**, *5*, 2117.

[427] Sander, T.; Freyss, J.; von Korff, M.; Rufener, C. DataWarrior: an open-source program for chemistry aware data visualization and analysis. *J. Chem. Inf. Model.* **2015**, *55*, 460–473.

[428] Harrison Jr, D.; Rubinfeld, D. L. Hedonic housing prices and the demand for clean air. *J. Env. Econ. Manag.* **1978**, *5*, 81–102.

[429] Van Der Walt, S.; Colbert, S. C.; Varoquaux, G. The NumPy array: a structure for efficient numerical computation. *Comput. Sci. Eng.* **2011**, *13*, 22–30.

[430] McKinney, W., et al. Pandas: a foundational Python library for data analysis and statistics. Python High Performance Scientific Computing. 2011; pp 1–9.

[431] Wes McKinney, Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference. 2010; pp 56 – 61.

[432] Hunter, J. D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **2007**, *9*, 90–95.

[433] Abadi, M. et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015; https://www.tensorflow.org.

[434] Palkovits, S. A Primer about Machine Learning in Catalysis–A Tutorial with Code. *ChemCatChem* **2020**, *12*, 3995–4008.

[435] Larsen, A. H. et al. The atomic simulation environment—a Python library for working with atoms. *J. Phys. Condens. Matter* **2017**, *29*, 273002.

Raimon Fabregat

Curriculum Vitae

✉ raimon.fa@gmail.com

Education

2017–2021 **Ph.D. in Computational Chemistry**, EPFL, Lausanne, Switzerland, Director: Prof. C. Corminboeuf.

2016–2017 **M.Sc. in Theoretical Physics/Computer Science**, ENS Lyon, Lyon, France.

2011–2016 **B.Sc. in Theoretical Physics**, Universitat de Barcelona, Barcelona, Spain.

Publications

- Gallarati, S.; Fabregat, R.; Laplaza, R.; Bhatacharjee. S.; Wodrich. M. D.; Corminboeuf, C. Reaction-based machine learning representations for predicting the enantioselectivity of organocatalysts *Chem. Sci.* **2021**, 12, 6879–6889.

- Blaskovits, J. T.; Lin, K. H.; Fabregat, R.; Swiderska, I.; Wu H.; Corminboeuf, C. Is a Single Conformer Sufficient to Describe the Reorganization Energy of Amorphous Organic Transport Materials? *J. Phys. Chem. C* **2021**, 125, 31, 17355–17362.

- Blaskovits, J. T.; Vela, S.; Fumanal, M.; Fabregat, R.; Corminboeuf, C. Identifying the Trade-off between Intramolecular Singlet Fission Requirements in Donor-Acceptor Copolymer *Chem. Mater.* **2021**, 33, 7, 2567–2575.

- Vela, S.; Scheidegger, A.; Fabregat, R.; Corminboeuf, C. Tuning the Thermal Stability and Photoisomerization of Azoheteroarenes through Macrocycle Strain *Chem. - Eur. J.* **2020**, 27, 419.

- Fabregat, R.; Fabrizio, A.; Engels, E. A.; Meyer, B.; Jurasokva, V.; Ceriotti, M.; Corminboeuf, C. Addressing molecular flexibility in oligopeptides withmachine learning, *Submitted for publication.*

- Fabregat, R.; Fabrizio, A.; Meyer, B.; Hollas, D.; Corminboeuf, C. Hamiltonian-reservoir Replica Exchange and Machine Learning Potentials for Computational Organic Chemistry, *J. Chem. Theory Comput.* **2020**, 16, 5, 3084–3094.

- Fabrizio, A.; Meyer, B.; Fabregat, R.; Corminboeuf, C. Quantum Chemistry Meets Machine Learning *CHIMIA* **2019**, 73, 983-989.

- Fabregat, R.; Pustelnik, N.; Gonçalves, P.; Borgnat, P. NMF with smoothness and sparsity constraints using PALM *arXiv:1910.14576*, **2017**.

Teaching Activities

2017-2021 **Teaching assistant**, Advance General Chemistry I by Prof. C. Corminboeuf, EPFL.

2018 **Teaching assistant**, Chemical Thermodynamics by Prof. A. Hagfeldt, EPFL.

Computational Skills

Programming languadges, Python, Bash, Git, FORTRAN, HTML/CSS, Markdown, Latex.

Software, Matlab, Mathematica, Blender, WordPress.

Areas of experise

Statistical analysis, Monte Carlo integration methods, Signal processing, data analysis with Pandas.

Machine learning and pattern decomposition, Linear models, kernel methods, neural networks, matrix factorization, dimensionality reduction, clustering, similarity learning.

Molecular simulations, Molecular dynamics, Monte Carlo simulations, enhanced sampling, free energy computations, response functions.

Dynamical Systems, Nonlinear dynamics, chaotic systems.

Network Science and graph theory, Network properties, centrality measures, topology, community detection, graph signal processing, epidemic processes.

Fundamental theoretical physics, Statistical physics and thermodynamics, quantum physics, classical optics and electromagnetism.

Graphical Design, Image and video editing, vector image editing, 3D modeling.

Languadges

Spanish, Mother tongue.

Catalan, Mother tongue.

English, Fluent.

French, Intermediate.

Awards

2016 **ENS Lyon**, *Ampere Excellence Scholarship for master studies.*