5-2020

Advanced Statistical Methods for Atomic-Level Quantification of Multi-Component Alloys

Adam Spannaus
University of Tennessee, aspannau@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Adam Spannaus entitled "Advanced Statistical Methods for Atomic-Level Quantification of Multi-Component Alloys." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mathematics.

Vasileios Maroulas, Major Professor

We have read this dissertation and recommend its acceptance:

Xiaobing Feng, David Keffer, Kody Law, Tim Schulze

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Advanced Statistical Methods for Atomic-Level Quantification of Multi-Component Alloys

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Adam Spannaus

May 2020

For Julia, Bryan, and Oliver, my favorite random variables.

Acknowledgments

First and foremost, I would like to thank my advisor Professor Vasileios Maroulas for his continuing support, invaluable feedback, and encouragement, Professor David Keffer for showing me how interesting interdisciplinary work can be and for his invaluable discussions about materials science. Professor Kody J.H. Law for his continued encouragement and thoughtful discussions throughout the years and I would also like to express my gratitude to each of my committee members, Professor Xiaobing Feng and Professor Tim Schulze for their invaluable feedback. Your support and feedback have been instrumental in my success. I would like to thank the National Science Foundation (DMS-1821241), the Army Research Office (W911NF-17-1-0313), and the University of Tennessee for support during my graduate studies. Thank you to all of my colleagues at the University of Tennessee, thank you for the insights, instructions, fruitful discussions, and much more.

Lastly, I would like to thank my family for their endless patience over these past years, you have been a source of inspiration and optimism throughout this journey.

Abstract

Advances in materials design have produced novel compounds, such as high-entropy alloys and entropy-stabilized oxides, that exhibit remarkable properties stemming from an amorphous and highly-disordered structure. Due to their high-configurational entropy and nanoscale disorder, such materials are not amenable to traditional techniques that characterize the local atomic-structure. Instead, we invoke techniques such as atom probe tomography that create information-rich datasets containing elemental type and spatial coordinates. The technique used to view these materials at the nanoscale produces a large but sparse dataset, comprised of approximately 10^7 atoms. However, it is corrupted by nontrivial amounts of observational noise.

The advent of such material design techniques necessitate new developments in statistical methodologies and data flows to fully capture the structural variations of these materials at an appropriate scale. A thorough analysis of these atomic-level variations unlock the capability for rapid material discovery. To fully explore and analyze such materials requires developing efficient and thoughtfully designed material descriptors. These descriptors may be continuous or discrete, but must be quantifiable in order to be employed by a statistical learning methodology. Our goal is to develop statistical methods to quantify the atomic structure of these materials. Our strategy is decomposed into three parts: *i.* Classifying the lattice structure of the material, *ii.* Mapping the perturbed observational data onto a model crystal lattice, and *iii.* Finding the optimal matching between observed data and the model lattice, and identifying the elemental type of the atoms in the model lattice.

From these three parts, we could map a noisy and sparse representation of a metallic alloy onto its reference lattice and determine the probabilities of different elemental types that are immediately adjacent, i.e., first neighbors, or are one-level removed and are second neighbors. Having these elemental descriptors of a material, researchers could then develop interaction potentials for molecular dynamics simulations, and make accurate predictions about these novel metallic alloys.

Table of Contents

1 Introduction **1**

1.1 Known Reference . . . 5

1.2 Unknown Reference . . . 5

2 Background and Preliminary Information **7**

2.1 Material Science Background . . . 7

2.1.1 High-Entropy Alloys . . . 7

2.1.2 Atom Probe Tomography . . . 8

2.2 Topological Data Analysis . . . 10

2.2.1 Persistent Homology Background . . . 10

2.3 Markov chain Monte Carlo . . . 13

2.3.1 Hamiltonian Monte Carlo . . . 15

2.3.2 Metropolis Adjusted Langevin Algorithm . . . 17

2.4 Classification . . . 18

2.4.1 Optimal Bayes Classifier . . . 19

2.4.2 Linear Regression . . . 20

2.4.3 Generalized Additive Model . . . 21

2.4.4 Decision Trees . . . 23

2.4.5 AdaBoost Algorithm . . . 26

2.4.6 Forward Stagewise Additive Modeling . . . 27

2.4.7 Optimality of AdaBoost . . . 29

2.5 Variational Bayes . . . 32

2.5.1 Expectation Maximization . . . 32

2.5.2 Variational Expectation Maximization . . . 33

2.5.3 Variational Inference for GMMs . . . 36
2.5.4 Prior distibutions . . . 37
2.5.5 Variational Posterior Distribution . . . 37

3 Known Reference **41**
3.1 Introduction . . . 41
3.1.1 Bayesian Point Set Registration . . . 42
3.2 Bayesian Formulation . . . 43
3.3 Numerical Experiments . . . 45
3.3.1 Sensitivity Analysis . . . 46

4 Unknown Reference **55**
4.1 Materials Fingerprinting . . . 55
4.1.1 Stability . . . 59
4.2 Classification of Materials Data . . . 65
4.2.1 Classification Model . . . 67
4.2.2 Sensitivity Analysis . . . 69
4.3 Materials Fingerprinting . . . 73
4.3.1 Numerical Results . . . 76
4.3.2 Sensitivity Analysis . . . 77
4.4 Variational Atomic Sequencing . . . 78
4.4.1 Statistical Model . . . 79
4.5 Variational Posterior . . . 81
4.5.1 Optimal $q(Z)$ (Variational E-Step) . . . 81
4.5.2 Optimal $q(\omega, \mu, \Lambda, \lambda)$ (Variational M-Step) . . . 82
4.5.3 Convergence . . . 85
4.5.4 Estimating Interaction Potentials . . . 87
4.6 Variational Atomic Sequencing Numerics . . . 88
4.6.1 Atomic Neighborhood Volume . . . 88
4.6.2 Sensitivity Analysis . . . 90
4.6.3 Real APT Data . . . 94

5 Conclusions **98**

5.1 Chapter Summaries 98

5.1.1 Introduction 98

5.1.2 Known Reference 99

5.1.3 Unknown Reference 99

5.2 Conclusions 101

5.3 Future Research 102

Bibliography **104**

Vita **115**

List of Tables

3.1 $\mathcal{E}(\theta)$ Registration Errors 48

3.2 Errors for 125 Completed Registrations 53

4.1 The atomic positions in the APT data is $\mathcal{N}(0, \tau^2)$ distributed with 67% of the atoms missing. We employ the d_p^c classifier, where c has been optimized in each noise level case. The accuracy in the 10-fold cross validation is listed in the third column. 69

4.2 H_0 analysis of varying levels of sparsity and Gaussian noise to inform our choice of radii in the atomic neighborhoods we consider. 88

4.3 Neighbor analysis with varying levels of sparsity and Gaussian noise. 92

4.4 Error Statistics, synthetic APT data with 33% missing and $\mathcal{N}(0, 0.25^2)$ added noise, for 250 neighborhoods. The true lattice parameter is 2.8. 93

4.5 Error Statistics, real APT data. 95

List of Figures

1.1 A slice of an HEA, $Al_{1.3}CoCrCuFeNi$, as seen by an APT experiment, where the crystal structure has been independently corroborated through X-ray diffraction and neutron scattering [1]. Each sphere represents a different atom, and the elemental type is shown by color, as denoted by the key on the right-hand side. The sparsity is evident and can be seen by the white areas, where no atoms are detected by the APT experimental process. We also expect to see a uniform distribution of color outside the orange copper-rich areas, if all atoms are registered by the process. To the eye, no pattern appears to exist in the material, and its crystal structure could be either body-centered cubic or face-centered cubic. This crucial distinction is obscured due to the noise and sparsity introduced through the APT process. For our methods described herein, we use neighborhoods around each atom to determine the local structure. 2

1.2 Examples of atomic neighborhoods where the nearest-neighbor relationships and lattice type of the atomic neighborhood and spacing are obscured due to the noise and sparsity, which increase from left to right. 3

2.1 Example of a 0-simplex, 1-simplex, 2-simplex, and 3-simplex, respectively. 11

2.2 Begin with a point cloud (a). After increasing the radius of the balls around the points, a 1-simplex (line segment) forms in the corresponding Vietoris-Rips complex, (b). Eventually, more 1-simplices are added and a 1-dim hole forms (c). In (d), the persistence diagram tracks all the birth and death times, with respect to the radius for the homological features in each dimension. The corresponding barcode plot is shown in (e). Using the same data, a persistence diagram is created using a sublevel set filtration (f). 12

2.3 Consider two persistence diagrams, one given by the green squares and another by the purple circles. (a) The Wasserstein distance imposes a cost of 0.2 to the extra purple point (the ℓ^∞-distance to the diagonal). (b) The d_p^c distance imposes a penalty c on the point instead. . 14

2.4 Example decision tree for one vector of predictor variables $x = (x, y)^T$, and associated predicted responses $\hat{y} \in \{-1, 1\}$. 24
2.5 If s_1 is the y-coordinate in the tree to the left, then any split will include points from both classes. 24
2.6 Variational inference applied to complete, noiseless data (a) and noisy, sparse data (b), showing how the process is not able to recover the chemical ordering, i.e., the aluminium center has 8 nickel first neighbors and 6 aluminium second neighbors, and vise versa when nickel is the center atom. 40

3.1 Setup for incorrect registration; alternating assignment and ℓ^2 minimization 42
3.2 Example APT data: Left: Hidden truth, Center: Noise added, Right: Missing atoms colored grey. 47
3.3 Error as plotted against various combinations of noise and sparsity. 49
3.4 Histograms of φ parameters, 100000 samples, $\gamma = 0.25$, Observed = 35% 50
3.5 Histograms of θ parameters, 100000 samples, $\gamma = 0.25$, Observed = 35% 51
3.6 Histograms of θ parameters, 100000 samples, $\gamma = 0.25$, Observed = 35% 52
3.7 Autocorrelation and trace plots for φ for our MCMC Bayesian registration method. 54

4.1 An example of face-centered cubic lattices showing the similarities and differences between an ideal, noiseless lattice structure in (a) and the data retrieved from an APT experiment in (c). Atoms in (a) sit precisely at their lattice points and each side of the cube is equal in length. The lattice in (b) shows the distorted lattice structure of an FCC HEA. The atomic positions no longer form a symmetric lattice and the sides are unequal in length. These local distortions are due to different sized atoms sitting at lattice positions and break the symmetry of the idealized FCC lattice in (a). These local lattice distortions make identification of the crystal structure by existing symmetry-based algorithms a challenging problem. In spite of these distortions, the unit cell retains the essential characteristics of an FCC cell: (*i*) number of atoms in the unit cell and (*ii*) atoms on the cube's faces and hollow in the center. We also note the different sized cubes in each one of the cells due to the random distribution of atoms throughout the material. The cell in (c) indicates the sparsity and atomic displacements due to the resolution of APT. Importantly, there are fewer atoms in (c) than in the idealized representation (a). 57

4.2 Example of body-centered cubic, (BCC), (a) and face-centered cubic, (FCC), (b) unit cells without additive noise or sparsity. Notice there is an essential topological difference between the two structures: The body-centered cubic structure has one atom at its center, whereas the face-centered cubic is hollow in its center, but has one atom in the middle of each of its faces. 58

4.3 An example of 8-point arrangements to visualize the proof of Proposition 4.5. (a) A 3-hole configuration vs. (b) a 2-hole configuration. 63

4.4 Sample persistence diagrams of a material from APT data of the alloys $Al_{1.3}CoCrCuFeNi$ and $Al_{0.3}CoCrFeNi$ for the two lattice types considered here: BCC (a) and FCC (b), respectively. Notice the distinguishing 2-dim feature, the blue square, in the diagram derived from an FCC lattice, and the diagram generated from the BCC structure has fewer 0-dim features. 66

4.5 Image of APT data with atomic neighborhoods shown in detail on the left and right. Each pixel represents a different atom, the neighborhood of which is considered. Certain patterns with distinct crystal structures exist, e.g., the orange region is copper-rich (left), but overall no pattern is identified. Putting a single atomic neighborhood under a microscope, the true crystal structure of the material, which could be either BCC (Fig. 4.2a) or FCC (Fig. 4.2b), is not revealed. This distinction is obscured due to experimental noise and sparsity present in the dataset. 67

4.6 Example of persistence diagrams generated by (a) a BCC lattice, and (b) FCC lattice. The data has a noise standard deviation of $\tau = 0.75$ and 67% of the atoms are missing. Note that the BCC diagram has two prominent (far from the diagonal) points representing 1-dim holes and fewer connected components and 1-dim holes than does the FCC diagram. 71

4.7 *Top:* Number of connected components (in this case atoms), $\mathbf{b_0}$, against the number of 1-dim homological features, $\mathbf{b_1}$, of the persistence diagrams. One can see the presence of heteroscedasticity since the variance of $\mathbf{b_1}$ increases as $\mathbf{b_0}$ increases. *Bottom:* Same as in top but using a quadratic transformation of the predictor variable, along with the weighted least squares fit line and 95% prediction intervals provided by Proposition 4.6. 71

4.8 10-fold cross validation accuracy scores for d_p^c (red), Wasserstein (blue), and counting (green) classifiers, plotted against different standard deviations, τ, (see Table 4.1) of the normally distributed noise of the atomic positions. In each instance, the sparsity has been fixed at 67% of the atoms missing, as in a true APT experiment. 72

4.9 Atomic neighborhood from an APT experiment (Figure 4.5); for the alloy $Al_{1.3}$CoCrCuFeNi where the atomic type is illustrated by the color. (a) shows each atom as a point cloud in $\mathbb{R}^3$. As the radius of the sphere centered at each atom increases in (b), a 1-dim hole forms in the atomic structure. Increasing the radii further, in (c) the formation of a 2-dim hole, a void, is evident. Continuing to increase the radii, in (d) the radii have increased such that all atoms form one cluster. The persistence diagram for this structure is shown in (e). In (f) the d_p^c metric computes the distance between two persistence diagrams. Consider two 1-dim persistence diagrams generated by atomic neighborhoods, one shown by the pink triangles, the other by the green triangles. The d_p^c metric measures the distance between the diagrams by first finding the best matching between points. Any unmatched points are then penalized by the regularization term c. 74

4.10 The materials fingerprinting methodology through which the APT data is processed. Individual atomic neighborhoods are extracted from an APT dataset. From these neighborhoods, we create a collection of persistence diagrams, each diagram associated with an atomic neighborhood. We then compute the d_p^c distance between all diagrams in the training set. We create a feature matrix composed of the summary statistics of these distances, which is used as input to the classification algorithm. The algorithm returns the class label for the atomic neighborhood associated to the persistence diagram D_i as either BCC or FCC in its structure, which is subsequently applied to new, unlabeled samples to automatically fingerprint them. 76

4.11 Variational inference applied to synthetic APT data of a binary NiAl alloy. The process is able to recover the chemical ordering, i.e., the aluminium center has only nickel first neighbors and aluminium second neighbors, and vise versa when nickel is the center atom. . 90

4.12 Example of a chemically ordered BCC lattice used in our sensitivity analysis, where color denotes elemental type. The first neighbors of the center atom are exclusively the other type, whereas the second neighbors are of the same species. 91

4.13 Variational inference applied to synthetic APT data of a binary NiAl alloy, *without* preferential ordering. As expected, the process is not able to recover the chemical ordering as in Figure 4.11, and the elemental distribution is random. The total number of atoms, not counting the center, in the aluminium neighborhoods is 1551: 615 aluminium and 936 nickel. Similarly the nickel neighborhoods contained 1564 atoms, 935 were aluminium a and 629 were nickel. 91

4.14 Extracted interaction potentials from our method applied to synthetic data with 33% missing and $\mathcal{N}(0, 0.25^2)$ added noise. 93
4.15 Variational inference applied to synthetic data with 33% missing and $\mathcal{N}(0, 0.25^2)$ added noise. 94
4.16 Neighbor analysis of the BCC phases in the multi-component alloy Al_1.3CoCrCuFeNi[1] considering only the geometry of the two point sets. 96
4.17 H_0 homology of real APT data. 97
4.18 Variational inference applied to real APT data of a Ni_3Al alloy that exhibits preferential ordering. The method is unable to recover the ordering due to the noise and sparsity of the data. 97

List of Algorithms

2.1 Hamiltonian Monte Carlo 16
2.2 Metropolis Adjusted Langevin Algorithm 17
2.3 Find split 25
2.4 AdaBoost.M1 26
2.5 Stagewise Additive Modeling 30
2.6 EM-Algorithm 32
2.7 Variational EM 34
4.1 Materials Fingerprinting 70
4.2 Variational Atomic Sequencing 84

Chapter 1

Introduction

The comprehensive goal of this work is to extract information from a noisy and sparse dataset by engaging statistical and topological methodologies for data analysis, which is a fundamentally interdisciplinary endeavor. The data that we consider is endowed with some geometric structure, and this structure is obscured due to the noise and sparsity present in the data. While the applications presented herein are motivated by materials science, the techniques and methodologies can be applied more broadly.

Atom probe tomography (APT) is an analytical atomic-scale imaging technique with the capability to provide an information-rich descriptor of a material that contains elemental type and geometric coordinates of the detected atoms in $\mathbb{R}^3$. The ability to provide such datasets to material scientists has the potential to revolutionize the materials discovery process. However, there do not presently exist visualization techniques with atomic-scale resolution that have the capability to quantify nearest and second nearest neighbor relationships within a material. The APT process introduces two significant challenges into the data: sparsity and noise. Indeed, the resulting datasets do not preserve neighbor relationships between atoms, i.e., there exists a preference for one elemental type to exclusively have first nearest neighbors of a specific type. As an example of a chemically ordered alloy, consider the binary NiAl system. It is a chemically ordered alloy, in that each nickel atom has only aluminium first neighbors and each aluminium atom has only nickel first neighbors.

Local trends in the atomic ordering present in a material are obscured, but elemental clustering can be seen in the resulting APT dataset. For example see Figure 1.1. The copper rich regions are shown in orange, but the finer details, such as first and second neighbor relationships are obscured by the noise and sparsity introduced through the APT process. Furthermore the lattice type and spacing are not present in the resulting APT dataset, nor are they readily inferred.

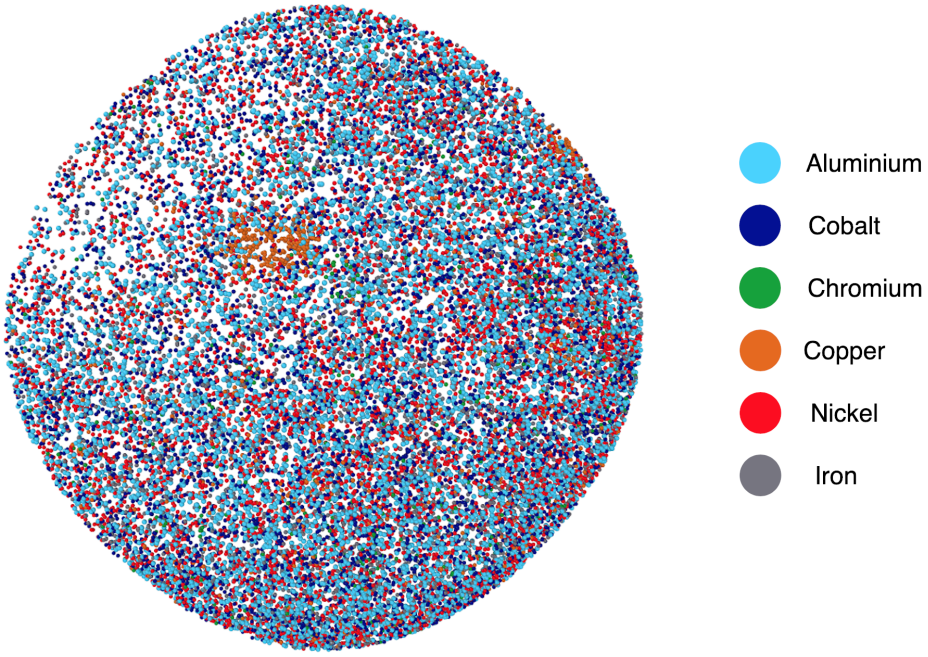
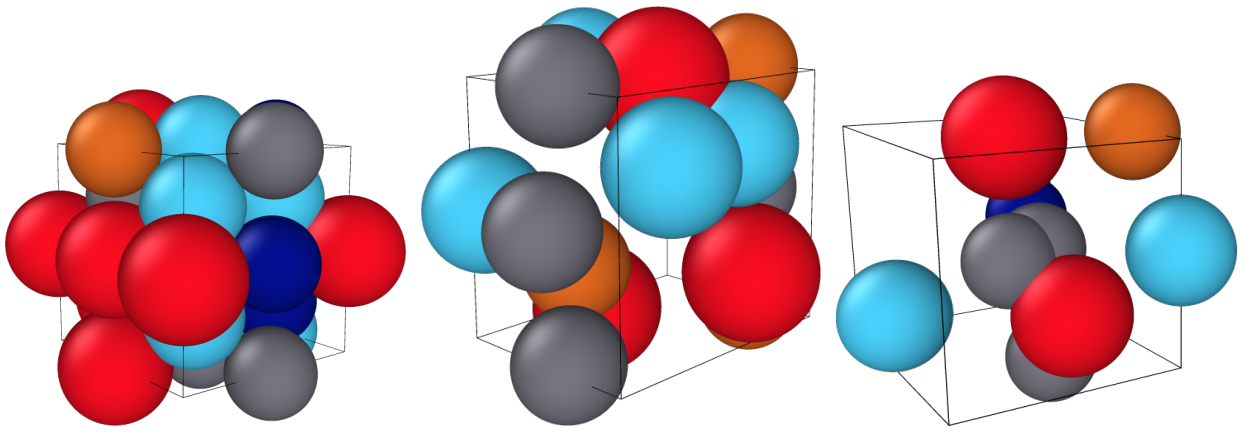


Figure 1.1: A slice of an HEA, $Al_{1.3}CoCrCuFeNi$, as seen by an APT experiment, where the crystal structure has been independently corroborated through X-ray diffraction and neutron scattering [1]. Each sphere represents a different atom, and the elemental type is shown by color, as denoted by the key on the right-hand side. The sparsity is evident and can be seen by the white areas, where no atoms are detected by the APT experimental process. We also expect to see a uniform distribution of color outside the orange copper-rich areas, if all atoms are registered by the process. To the eye, no pattern appears to exist in the material, and its crystal structure could be either body-centered cubic or face-centered cubic. This crucial distinction is obscured due to the noise and sparsity introduced through the APT process. For our methods described herein, we use neighborhoods around each atom to determine the local structure.



(a) (b) (c)

Figure 1.2: Examples of atomic neighborhoods where the nearest-neighbor relationships and lattice type of the atomic neighborhood and spacing are obscured due to the noise and sparsity, which increase from left to right.

As a visual example of the sparsity and noise introduced by the APT process, consider the different representations of the same atomic configuration in Figure 1.2. The lattice type in Fig. 1.2a is a face-centered crystal and the nearest and second-nearest neighbor relationships among the atoms are well-defined, easily seen upon visual examination. In Fig. 1.2b, we have added some Gaussian noise and removed some of the atoms from the configuration. Now these same fundamental relationships between atoms are less clear, and the lattice spacing has been dilated. In Fig. 1.2c, these characteristics become indeterminate, where 2/3rds of the atoms have been removed and those remaining atoms have been significantly perturbed away from their corresponding lattice sites. No longer can we ascertain with any degree of certainty which atoms are first neighbors, nor the type of lattice that the atoms form.

Our goal is to reveal the atomic-scale characteristics of a material from noisy and sparse realizations. Specifically, we are interested in the nearest-neighbor relationships between atoms, the crystal structure, and the lattice spacing (or lattice parameter). The impact to the materials science community of such information is twofold. Primarily, using a quasi-chemical approximation [2], we may construct a first approximation of the interaction potentials governing a multi-component system. While high-quality binary and ternary interaction potentials exist for different atomic configurations [3, 4, 5], estimating the multi-component interaction potentials governing high-entropy alloys (HEAs), a specific type of multi-component alloy, remains an open question in materials science. Secondly, the output from our algorithms is a collection of atomic neighborhoods that have their points mapped onto a crystal lattice. We may compute the distribution of

these configurations by considering both the geometry and atomic composition of the neighborhoods. These rectified configurations define a distribution over the atomic composition and geometry, and each observation is then associated with some probability. We provide these empirical probabilities of the observed states as input for molecular dynamics simulations.

These computational simulations are essential for materials science researchers, as it allows them to make accurate structure-property predictions. Such relationships currently do not exist with regards to HEAs, due in part to the nano-scale disorder engineered into the alloys. In HEAs, multiple elements are mixed in roughly equimolar proportions to create materials of entirely new type possessing unique materials properties. However, discovering these properties *a priori* has remained elusive, due to the inability to perform classical molecular dynamics simulations.

Here we present different statistical models and propose novel algorithmic solutions to infer fundamental properties of a material as seen through the lens of an APT experiment. We are motivated by a desire to uncover the structure of high-entropy alloys, which are amenable to characterization via APT. Our goal is to infer the neighbor relationships, lattice type, and spacing from a noisy and sparse atomic configuration, e.g., the type shown in Fig. 1.2c, and reconstruct the ideal configuration, Fig. 1.2a. The materials we consider are primarily one of two different lattice types, which are *a priori* unknown, but may be inferred through a logistic regression model. To quantify the neighbor relationships and lattice spacing, we model both the observed atoms and an aligned lattice, without labeling the lattice points with an atomic type, as Gaussian Mixture Models, i.e., a convex combination of Gaussian densities. From a mathematical/statistical viewpoint, we pose these inference questions as one of density estimation, where we seek the *maximum a posteriori* estimator, and as a minimization problem, where we seek to minimize the misclassification rate or to minimize the Kullback-Leibler divergence, i.e., the information gain, or relative entropy, between two distributions. We obtain a global minimizer of the divergence, and provide a proof of convergence.

We begin with a discussion of the problem, assuming that we know the true lattice structure of the material, and present a Bayesian formulation of the point-set registration problem. Next, we then work directly with APT data and develop a topologically-informed classification methodology to infer the lattice structure. We further develop the statistical model to infer not only the mapping onto a known lattice, but the chemical identity of the known lattice points as well. It is this latter inference that is of particular interest to the materials science community, as short-range ordering is an open question to materials scientists investigating high-entropy alloys.

1.1 Known Reference

In the case where the reference lattice structure is known, we view the problem of aligning the noisy and spare observations with the reference as a point-set registration problem. This problem arises frequently in computer vision and medical imaging tasks. Typically, one seeks to align two point clouds, called the observed and reference. One then finds the optimal alignment, where the 'optimality' criterion varies widely depending the different settings of this problem, between the reference and observation point clouds.

We take a Bayesian formulation of the problem that will simultaneously find the transformation and correspondence between point sets. Most importantly, it is designed to avoid local basins of attraction and locate a global minimum. Indeed at an additional computational cost, we obtain a distribution of solutions, rather than a point estimate, so that general quantities of interest may be estimated and quantify the uncertainty. In case a single point estimate is required, we define an appropriate optimal one, e.g., the global energy minimizer or probability maximizer. We present our Bayesian methodology and the associated statistical model in Chapter 3 and present numerical results.

1.2 Unknown Reference

We now take the approach of the case where the reference structure is *unknown*, and must be inferred from the APT data. When working with real APT datasets, the true lattice, or reference point set, cannot be determined directly from the data itself, and subsequent analysis must be employed. We create a topologically informed machine learning classification process that is able to classify, with a gig degree of accuracy, the lattice structure of a material from the atom probe tomography dataset.

Having the crystal structure in hand, we may then infer the local atomic structure of each configuration through our variational Bayesian registration process detailed in Section 4.4. Here we relax the rigid transformation assumption of Chapter 3, allowing more freedom of motion to identify the mapping between observation and reference point sets. We further incorporate the chemical identity of the atoms, in order to estimate the interaction potentials governing the multi-component alloys, and provide a nearest-neighbor analysis of the registered point set. From such a neighbor analysis, we may infer the presence of short-range chemical ordering within a material. The presence of short-range ordering and its subsequent analysis provides materials scientists with unprecedented insight into these alloys, paving the way to quantify the structure-property relationships that exist in these novel materials.

Overview of Thesis

Through the techniques and methodologies described above, this document provides a holistic methodology for researchers analyzing sparse and noisy materials data, not strictly limited to HEAs and APT data. In Chapter 2 we give the necessary definitions and ideas for our methods presented herein. Chapter 3 describes our Bayesian point set registration algorithm, and presents numerical results on synthetic materials data. We consider the semi-supervised classification problem in Section 4.1, and provide an accurate classification methodology for inferring the crystal lattice from APT data. Lastly, we consider the point set registration problem from a different viewpoint, by incorporating the elemental type associated with each point in the point sets. This leads us to a variational formulation of the Bayesian registration framework in Section 4.4. We present, to our knowledge, the first proof of convergence in such a setting. We then detail results based on the geometry of the point sets alone, and where we consider the geometry and atomic types on both synthetic and real APT data. These results not only inform materials scientists studying multi-component alloys, but may apprise researchers working on the APT process, as the results of our analysis can be employed as a sensitivity analysis, and guide future improvements of the technique.

Chapter 2

Background and Preliminary Information

In this section we begin by detailing the unique characteristics of high-entropy alloys and the imaging technique used to view these alloys at the nanoscale, which provides the data set for our analysis. We give the necessary background mathematical details for topological data analysis in Section 2.2, Monte Carlo Markov chain methods in Section 2.3, logistic regression and classification in Section 2.4, and variational Bayesian methods in Section 2.5. Our methodologies employ techniques from each of these separate areas in our statistical analysis.

2.1 Material Science Background

2.1.1 High-Entropy Alloys

In recent years, a new class of materials has emerged, called high-entropy alloys. These materials are a type of metallic alloy, first synthesized in the mid 2000's by [6]. As defined in [7], HEAs are composed of at least five atomic elements, each with an atomic concentration between 5% and 35%. These novel alloys have remarkable properties, such as: corrosion resistance [8, 9], increased strength at extreme temperatures, ductility [10, 11, 12], increased levels of elasticity [13], strong fatigue and fracture resistance [10, 14, 15], and enhanced electrical conductivity [16, 17]. HEAs demonstrate a 'cocktail' effect [18], in which the mixing of many components results in a composite effect on materials properties, where the mixing between different elements results in a composite material endowed with properties linked with the individual elements and indirectly correlated with microstructure properties [18]. Although these metals hold great promise for a wide variety of applications, the greatest impediment in tailoring the design of HEAs to specific applications is the inability to accurately predict their atomic structure and chemical ordering. This prevents materials science

researchers from constructing structure-property relationships necessary for targeted materials discovery. Although these structure-property relationships have begin to be explored in disordered materials, such as entropy-stabilized oxides and high-entropy alloys [19, 9], they are now yet well-understood.

Knowledge of the chemical ordering and geometric arrangement of the atoms of any material is essential for developing predictive structure-property relationships. Indeed, the disorder amongst lattice sites present in HEAs [6], hinders the development of structure-property relationships. Considering the number of atomic configurations in a disordered crystal structures found in HEAs [20] the number of possible atomic combinations of even a single unit cell, the smallest collection and ordering of atoms from which an entire material can be built, quickly becomes computationally intractable for existing algorithms [21]. Moreover, this class of metallic alloys lacks a uniform lattice parameter and atomic composition. The high-configurational entropy of HEAs yields a distribution of lattice parameters and cell compositions, as opposed to a single unit cell and lattice constant found in more traditional materials.

For many classes of materials, the lattice structure is either well-known, e.g., sodium chloride (salt) is body-centered cubic, or it can be discovered via X-ray diffraction (XRD) or neutron scattering techniques [1]. These are routine techniques used to determinate the crystal structure of metals, ceramics, and other crystalline materials. They do not yield atomic level elemental distinctions or resolve local lattice distortions on a scale of less than 10Å [1] though, and such information is crucial to researchers working with highly-disordered materials. Furthermore, XRD cannot provide the correlation between atom identity and position. This chemical ordering of atoms is essential to developing predictive relationships between the composition of an HEA and its properties.

2.1.2 Atom Probe Tomography

An important experimental characterization technique used to determine atomic-level structure of materials is atom probe tomography (APT), which has the capability of uncovering nanoscale trends in materials [22, 23]. Although other characterization techniques can yield similar information about the nanoscale structure of a material, such as X-ray diffraction being used to determine crystal structure, APT includes atomic composition in the resulting dataset as well. Indeed, it is the only available technique that yields elemental type and geometric coordinates of atoms present in a material [22, 23]. Such information can be used in determining stoichiometry, i.e., compositional identification and quantification, at the nanoscale.

APT has been successfully applied to the characterization of the HEA, $Al_{1.3}$CoCrCuFeNi [1], and recent advances in the data quality available from a typical experiment have made the atom probe a routine part of a material characterization [23]. The experiment yields a dataset typically comprised of millions of atoms [23],

on the scale of 10^7. Sophisticated reconstruction techniques are employed to generate the coordinates based upon the construction of the experimental apparatus.

The potential of the APT process remains unrealized however, due to data-recovery issues inherent to the technique [24, 23, 25]. It is well known that the reconstructed data is corrupted by some experiment-dependent observational noise, and not all atoms are recovered by the process [26, 27, 24, 23, 25]. Both factors are significant and any subsequent data analysis must work to mitigate these effects. The amount of observational noise is non-trivial, as it can make atoms that are first neighbors in the materials appear as second neighbors in the reconstructed material, and vice-versa [25]. While this permuting of neighborhood relationships is less significant for a global analysis of the material, it makes determining the presence of any chemical ordering or elemental preference between neighboring elements a challenging problem. The difficulty of such a task is only increased when we consider that the percent of the data missing in the reconstructed material can be greater than 60%. Our goal is to correctly infer this chemical ordering from noisy and sparse observations, as are typically retrieved from an APT experiment.

As previously discussed, APT data has two main drawbacks: (i) up to 2/3rds of the data is missing and (ii) the recovered data is corrupted by noise. As noted by [25], the spatial resolution of the APT process is up to 3Å (0.3 nm) in the xy-horizontal plane, which is approximately the length of an atomic unit neighborhood that we consider; see Figure 1.1 for a slice of an HEA as seen by an APT experiment. This experimental noise has a two-fold impact on the data typically retrieved by APT. First, the noise prevents materials science researchers from extracting elemental atomic neighborhood distributions, which are essential for developing interaction potentials for molecular dynamics simulations. Secondly, the experimental noise is significant enough to change the nearest neighbor relationships in a lattice structure [25]. Furthermore, the experimental noise is only one source of distortion to the lattice structure. HEAs exhibit local lattice deformations due to the random distribution of atoms throughout the material and atoms of differing size sitting at adjacent lattice points [20].

The challenge is to uncover the true atomic level structure and chemical ordering amid the noise and missing data, thus giving material scientists an unambiguous description of the atomic structure of these novel alloys. Ultimately, our goal is to infer the correct spatial alignment and chemical ordering of a dataset containing up to 10^7 atoms. We will examine local structure by extracting configurations, on the scale of the alloy's unit cell, and each configuration will be probed by identifying a mapping between the observed points and a reference lattice, i.e., the unperturbed crystal lattice of the material without atomic labels on the lattice points, in a neighborhood around each atom.

In order to infer the correct spatial alignment and chemical identification of each point in the data, we decompose the problem into separate components. In Chapter 3, we view the alignment of the two point sets through the lens of Bayesian inference, in the case where the reference lattice structure is known, and present our Bayesian point set registration methodology. In the true APT data, the lattice is unknown, but can be separated into different classes by creating a topologically informed machine learning classification algorithm described in Section 4.1. Having inferred the correct lattice structure, we may then find the best alignment of the data and infer if chemical ordering exists in the material by again adopting a Bayesian approach, which is detailed in Section 4.4.

2.2 Topological Data Analysis

2.2.1 Persistent Homology Background

This section succinctly explains the construction of persistence diagrams or barcode plots, which are topological summaries of the underlying space; detailed introductions can be found in [28, 29]. The Vietoris-Rips complex provides the necessary computational link between the point cloud, a subset of $\mathbb{R}^d$ under the Euclidean distance, and its persistence diagram or persistence barcode. Instead of considering only clusters of points, the nearby atoms when measured by some distance, homology also incorporates information about the regions enclosed by the points. This approach yields topological features of the data in different homological dimensions. Homology describes connectedness and emptiness present within an object. It allows one to infer global properties of space from local information [30]. In the case of these atomic neighborhoods created by APT experiments, 0-dim homological features are connected components, i.e., the atoms themselves. Analogously, 1-dim homological features are holes, and 2-dim homological features are voids.

Definition 2.1. *A ν-simplex is the convex hull of an affinely independent point set of size $\nu + 1$.*

Definition 2.2. *For a set of points $\mathcal{P}$, an abstract simplicial complex σ is a collection of finite subsets of $\mathcal{P}$ such that for every set A in σ and every nonempty set $B \subset A$, we have that B is in σ. The elements of σ are called abstract simplices and are the combinatorial analogues of the geometric simplices in Definition 2.1.*

Figure 2.1 shows examples of various abstract simplicial complexes of the type typically used in topological data analysis. They may be created by various processes, such as the α-complex, Čech complex, or the Vietoris-Rips complex [31].

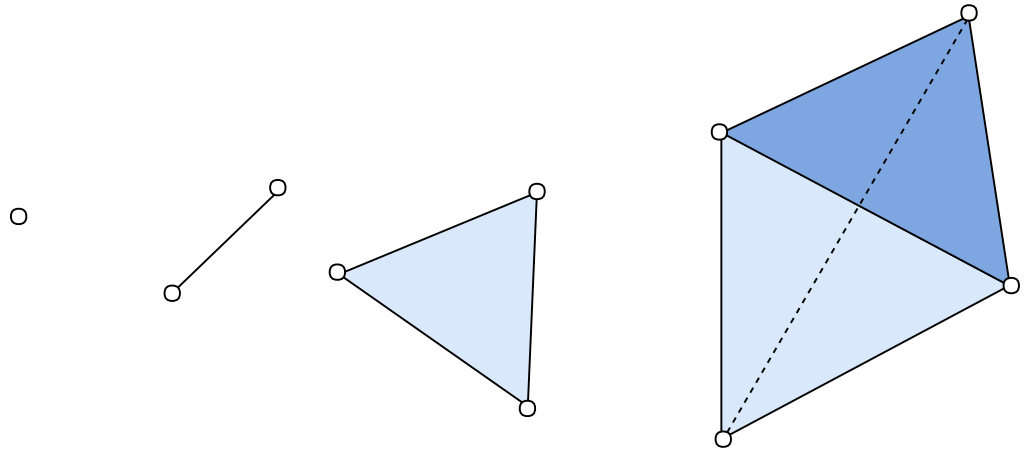


Figure 2.1: Example of a 0-simplex, 1-simplex, 2-simplex, and 3-simplex, respectively.

Definition 2.3. *For a given threshold ϵ, the Vietoris-Rips complex is a simplicial complex formed from a set such that corresponding to each subset of ν points of the set, an ν-simplex is included in the Vietoris-Rips complex each time the subsets have pairwise distances at most ϵ.*

The Vietoris-Rips complex can be visualized by placing a ball of radius $\epsilon/2$ at each point in the set and then adding a ν-simplex at the points corresponding to the intersection of ν balls. See Figure 2.2 for an illustration of how the process works. For the Vietoris-Rips complex corresponding to ϵ, denoted by VR_ϵ, it is clear that $VR_\epsilon \subset VR_{\epsilon'}$ for $\epsilon < \epsilon'$. Thus we need only examine specific ϵ values corresponding to the emergence and disappearance of homological features. These ϵ values are recorded as ordered pairs (b, d) in a persistence diagram, where b denotes the birth of a feature and d its death.

As can be seen in Figure 2.2, a 0-dim homological feature is a connected component of a simplex, a 1-dim homological feature is a hole, such as those created by a loop or the circle S^1, and a 2-dim homological feature describes voids, e.g., the inside of a sphere; see [32] for details. Higher dimensional data analogously yields higher dimensional holes.

Remark 2.1. *Persistence diagrams can also be computed using a pertinent function g from a topological space to $\mathbb{R}$. Such a function can act as an approximation to a point cloud; typical functions used are kernel density estimators as in [33] and the distance to measure function as in [34]. Homological features are born and die within the sublevel sets $g^{-1}(-\infty, t]$ as t increases. These birth and death times create another persistence diagram, see Fig. 2.2f.*

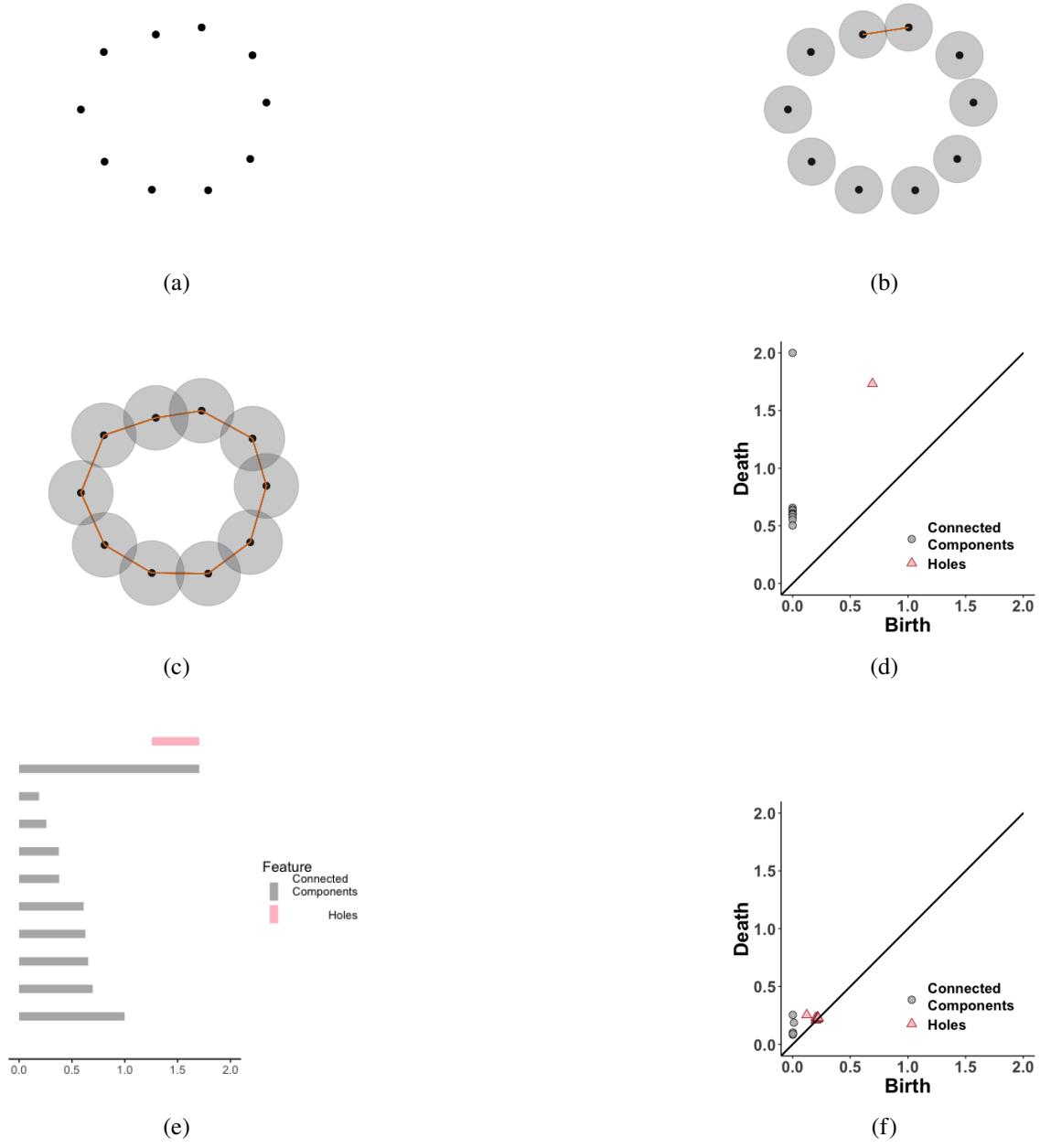


Figure 2.2: Begin with a point cloud (a). After increasing the radius of the balls around the points, a 1-simplex (line segment) forms in the corresponding Vietoris-Rips complex, (b). Eventually, more 1-simplices are added and a 1-dim hole forms (c). In (d), the persistence diagram tracks all the birth and death times, with respect to the radius for the homological features in each dimension. The corresponding barcode plot is shown in (e). Using the same data, a persistence diagram is created using a sublevel set filtration (f).

To calculate the similarity between diagrams for classification problems, a distance on the space of persistence diagrams is needed. A typical distance is the Wasserstein distance.

Definition 2.4. *The p-Wasserstein distance between two persistence diagrams D^1 and D^2 is given by $W_p(D^1, D^2) = \left(\inf_{\eta: D^1 \to D^2} \sum_{x \in D^1} \|x - \eta(x)\|_\infty^p\right)^{\frac{1}{p}}$, where the infimum is taken over all bijections η, and the points of the diagonal are added with infinite multiplicity to each diagram. If $p \to \infty$, then $W_\infty(D^1, D^2) = \inf_{\eta: D^1 \to D^2} \sup_{x \in D^1} \|x - \eta(x)\|_\infty$ is the bottleneck distance between diagrams D^1 and D^2.*

The Wasserstein distance yields the penalty of matched points under the optimal bijection. Points can be matched to the diagonal of each persistence diagram, which is assumed to have infinitely many points with infinite multiplicity; this ensures that a bijection between D^1 and D^2 actually exists, since D^1 and D^2 may not have the same cardinality. In other words, the Wasserstein distance gives no explicit penalty for differences in cardinality between two diagrams. Instead, the Wasserstein distance penalizes unmatched points by using their distance to the diagonal. However, cardinality differences may play a key role in machine learning problems, and to that end, [35] proposed the d_p^c distance given below.

Definition 2.5. *Let D^1 and D^2 be two persistence diagrams with cardinalities n and m respectively such that $n \leq m$ and denoted $D^1 = \{x_1, \ldots, x_n\}$, $D^2 = \{y_1, \ldots, y_m\}$. Let $c > 0$ and $1 \leq p < \infty$ be fixed parameters. The d_p^c distance between two persistence diagrams D^1 and D^2 is*

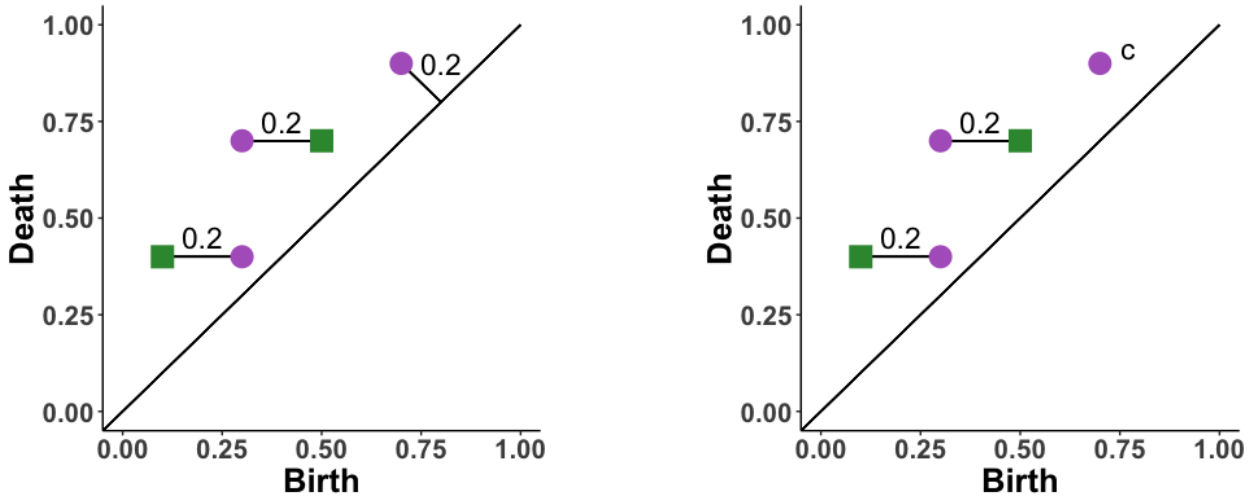
$$d_p^c(D^1, D^2) = \left(\frac{1}{m}\left(\min_{\pi \in \Pi_m} \sum_{\ell=1}^{n} \min(c, \|x_\ell - y_{\pi(\ell)}\|_\infty)^p + c^p |m - n|\right)\right)^{\frac{1}{p}}, \tag{2.2.1}$$

where Π_m is the set of permutations of $(1, \ldots, m)$. If $m < n$, define $d_p^c(D^1, D^2) := d_p^c(D^2, D^1)$.

Remark 2.2. *Note that this distance can be applied to arbitrary point clouds with finite cardinality as well. As shown in [35], a smaller c in Equation (2.2.1) accounts for local geometric differences, while a larger c focuses on global geometry. It is precisely by considering differences in cardinality that the d_p^c distance can distinguish between features of the point cloud that other distances may miss. Also in Equation (2.2.1), if D^1 is fixed and $m \to \infty$, then $d_p^c(D^1, D^2) \to c$.*

2.3 Markov chain Monte Carlo

Markov chain Monte Carlo (MCMC) methods are a natural choice for sampling from distributions which can be evaluated pointwise up to a normalizing constant, such as any posterior arising in Bayesian



(a) (b)

Figure 2.3: Consider two persistence diagrams, one given by the green squares and another by the purple circles. (a) The Wasserstein distance imposes a cost of 0.2 to the extra purple point (the ℓ^∞-distance to the diagonal). (b) The d_p^c distance imposes a penalty c on the point instead.

inference. Furthermore, MCMC comprises the workhorse of Bayesian computation, often appearing as crucial components of more sophisticated sampling algorithms. Formally, an MCMC simulates a distribution μ over a state space Ω by producing an ergodic Markov chain $\{w_k\}_{k\in\mathbb{N}}$ that has μ as its invariant distribution, i.e.

$$\frac{1}{K}\sum_{k=1}^{K} g(w_k) \to \int_\Omega g(w)\mu(\mathrm{d}w) = \mathbb{E}_\mu[g(w)]\,, \tag{2.3.1}$$

with probability 1, for $g \in L^1(\Omega)$.

The Metropolis-Hastings method is a general MCMC method defined by choosing $\theta_0 \in \mathrm{supp}(\pi)$ and iterating the following two steps for $k \geq 0$

(1) Propose: $\theta^* \sim Q(\theta_k, \cdot)$.

(2) Accept/reject: Let $\theta_{k+1} = \theta^*$ with probability

$$\alpha(\theta_k, \theta^*) = \min\left\{1, \frac{\pi(\theta^*)Q(\theta^*, \theta_k)}{\pi(\theta_k)Q(\theta_k, \theta^*)}\right\},$$

and $\theta_{k+1} = \theta_k$ otherwise.

2.3.1 Hamiltonian Monte Carlo

In general, random-walk proposals Q can result in MCMC chains which are slow to explore the state space and susceptible to getting stuck in local basins of attraction. Hamiltonian Monte Carlo (HMC) is designed to improve this shortcoming. HMC is a Metropolis-Hastings method [36, 37] which incorporates gradient information of the log density with a simulation of Hamiltonian dynamics to efficiently explore the state space and accept large moves of the Markov chain. Heuristically, the gradient yields d pieces of information, for a $\mathbb{R}^d$-valued variable and scalar objective function, as compared with one piece of information from the objective function alone. Our description here of the HMC algorithm follows that of [38] and the necessary foundations of Hamiltonian dynamics for the method can be found in [39].

Generally speaking, our objective is to sample from a specific target density

$$\pi(\theta) \propto \exp\{-E(\theta)\} \tag{2.3.2}$$

over θ, where $E(\theta)$ is known as an energy function and is the the negative log of the unnormalized log posterior density, i.e., $E(\theta) = -\log(p(\theta \mid X, Y))$ in the Bayesian registration case.

First, an artificial momentum variable $p \sim \mathcal{N}(0, \Gamma)$, independent of θ, is included into Equation (2.3.2), for a symmetric positive definite mass matrix Γ, that is usually a scalar multiple of the identity matrix. Define a Hamiltonian now by

$$\mathcal{H}(p, \theta) = E(\theta) + \frac{1}{2} p^T \Gamma^{-1} p$$

where $E(\theta)$ is the "potential energy" and $\frac{1}{2} p^T \Gamma^{-1} p$ is the "kinetic energy".

Hamilton's equations of motion for $p, \theta \in \mathbb{R}^d$ are, for $i = 1, \ldots, d$:

$$\begin{aligned} \frac{\mathrm{d}\theta_i}{\mathrm{d}t} &= \frac{\partial \mathcal{H}}{\partial p_i} \\ \frac{\mathrm{d}p_i}{\mathrm{d}t} &= -\frac{\partial \mathcal{H}}{\partial \theta_i} \end{aligned}$$

In practice, the algorithm creates a Markov chain on the joint position-momentum space $\mathbb{R}^{2d}$, by alternating between independently sampling from the marginal Gaussian on momentum p, and numerical integration of Hamiltonian dynamics along an energy contour to update the position. If the initial condition $\theta \sim \pi$ and we were able to perfectly simulate the dynamics, this would give samples from π because the Hamiltonian $\mathcal{H}$ remains constant along trajectories. Due to errors in numerical approximation, the value of

$\mathcal{H}$ will vary. To ensure the samples are indeed drawn from the correct distribution, a Metropolis-Hastings accept/reject step is incorporated into the method.

In particular, after a new momentum is sampled, suppose the chain is in the state (p, θ). Provided the numerical integrator is reversible, the probability of accepting the proposed point (p^*, θ^*) takes the form

$$\alpha((p,\theta),(p^*,\theta^*)) = \min\left\{1, \exp\left\{\mathcal{H}(p,\theta) - \mathcal{H}(p^*,\theta^*)\right\}\right\}. \tag{2.3.3}$$

If (p^*, θ^*) is rejected, the next state remains unchanged from the previous iteration. However, note that a fresh momentum variable is drawn each step, so only θ remains fixed. Indeed the momentum variables can be discarded, as they are only auxiliary variables. To be concrete, the algorithm requires an initial state θ_0, a reversible numerical integrator, integration step-size h, and number of steps L. Note that reversibility of the integrator is crucial such that the proposal integration $Q((p,\theta),(p^*,\theta^*))$ is symmetric and drops out of the acceptance probability in Equation (2.3.3). The parameters h and L are tuning parameters, and are described in detail [38, 37].

The HMC algorithm then proceeds as follows:

Algorithm 2.1 Hamiltonian Monte Carlo

Initialize the algorithm at some $\theta_0 \in \mathbb{R}^d$.
for $k \geq 0$ **do**
  Generate $p_k = \xi$ for $\xi \sim \mathcal{N}(0, \Gamma)$
  function INTEGRATOR(p_k, θ_k, h) **return** (p^*, θ^*)
  end function
  Generate $U \sim \mathcal{U}[0,1]$
  if $U < \min\{1, \exp\{\mathcal{H}(p_k,\theta_k) - \mathcal{H}(p^*,\theta^*)\}\}$ **then**
    $\theta_{k+1} = \theta^*_{k+1}$
  else
    $\theta_{k+1} = \theta_k$.
  end if
  Set $k \leftarrow k+1$.
end for

Under appropriate assumptions [37], this method will provide samples $\theta_k \sim \pi$, such that for bounded $g : \mathbb{R}^d \to \mathbb{R}$

$$\frac{1}{K}\sum_{k=1}^{K} g(\theta_k) \to \int_{\mathbb{R}^d} g(\theta)\mathrm{d}\theta \quad \text{as} \quad K \to \infty.$$

2.3.2 Metropolis Adjusted Langevin Algorithm

Another random walk metropolis algorithm that incorporates gradient information is the the Metropolis Adjusted Langevin algorithm (MALA). This algorithm is derived from a discretization of the diffusion process $\{L_t\}$ that is the solution of the Langevin stochastic differential equation

$$dL_t = \frac{\sigma^2}{2}\nabla \log(\pi(L_t))\mathrm{d}t + \sigma \mathrm{d}B_t \tag{2.3.4}$$

where $\{t \geq 0 : B_t\}$ is a standard d-dimensional Brownian motion, and σ^2 is the variance of π, a probability measure on $\mathbb{R}^d$ with respect to Lebesgue measure [40]. The random walk generated by the solution to Equation (2.3.4) is ergodic to the invariant distribution given by π.

The discretization of the solution to Equation (2.3.4) is

$$\theta^*_{k+1} = \theta_k + \frac{\sigma_d^2}{2}\nabla \log(\pi(\theta_k)) + \frac{\sigma_d}{2}Z_k$$

where σ_d^2 is the step variance, and Z_k is i.i.d. standard normal. We then set $\theta_{k+1} = \theta^*_{k+1}$ with probability

$$\alpha(\theta_k, \theta^*_{k+1}) = \min\left\{\frac{\pi(\theta^*_{k+1})q(\theta^*_{k+1}, \theta_k)}{\pi(\theta_t)q(\theta_k, \theta^*_{k+1})}, 1\right\}$$

and

$$q(x, y) \propto \|y - x - \frac{\sigma_d^2}{2}\nabla \log\{\pi(x)\}\|_2^2$$

Otherwise with probability $1 - \alpha(\theta_k, \theta^*_{k+1})$ we set $\theta_{k+1} = \theta_k$.

The iterative process is then given by:

Algorithm 2.2 Metropolis Adjusted Langevin Algorithm

Initialize the algorithm at some $\theta_0 \in \mathbb{R}^d$.
for all $k \geq 0$ **do**
  Generate $\theta^*_{k+1} = \theta_k + \frac{\sigma_d^2}{2}\nabla \log(\pi(\theta_k)) + \frac{\sigma_d}{2}Z_k$.
  Generate $U \sim \mathcal{U}[0, 1]$
  if $U < \alpha(\theta_k, \theta^*_{k+1})$ **then**
    $\theta_{k+1} = \theta^*_{k+1}$
  else
    $\theta_{k+1} = \theta_k$.
  end if
  Set $k \leftarrow k + 1$.
end for

2.4 Classification

We will begin with a general presentation of the classification problem, proceeding to a discussion of two models: logistic regression and generalized additive models. The latter model leads directly into a discussion of AdaBoost. AdaBoost was initially developed for a binary classification problem [41] and a multi-class extension was given in [42]. While the algorithm and ideas are quite similar in the two cases, there are some essential differences between the two, the objective function differs between the two cases as one example. We will present the theory in binary case, as it is part of our materials fingerprinting procedure, and will give some details about the multi-class extension.

Suppose for the moment we have a labeled set of training data $\mathcal{X} = \{X_1, \ldots, X_N\}$, where the predictor variable $X_i \in \mathbb{R}^d$ is associated with a response y_i, denoting class label, taking values in $Y = \{-1, 1\}$. We make the assumption that the pairs (X_i, y_i) are i.i.d. samples from a joint distribution $\mathcal{D} = \mathcal{X} \times Y$. Our goal is to construct a classification rule, $\hat{C}(x) : \mathcal{X} \to Y$, that correctly assigns class labels to a vector of unlabeled data $\tilde{X}$. That is we seek a rule $\hat{C}$ such that $\hat{C}(\tilde{X}) = \tilde{y}$ for unlabeled data $(\tilde{X}, \tilde{y}) \sim \mathcal{D}$.

We want to construct this rule $\hat{C}(X)$ such that it minimizes the $0/1$-loss given by

$$M(\hat{C}) = \begin{cases} 0 & \text{if } \hat{C}(X) = C(X) \\ 1 & \text{otherwise,} \end{cases} \tag{2.4.1}$$

where $C(X)$ yields the true class label for X. We may more generally write the loss function as

$$M(\hat{C}) = 1 - \mathbb{P}(\hat{C}(X) = y \mid X), \tag{2.4.2}$$

as it is conceivable that datapoints with similar features are from different classes. This is reasonable since we choose the attributes used to differentiate between classes and we do not have access to perfect information. We can only see noisy and sparse representations of an underlying truth. More generally then, a dataum X is not be associated with a specific class, but instead with a vector of class probabilities where the i^{th} entry of the vector denotes the probability of X being in the i^{th} class. Associating the assignment of class labels with a probability introduces an element of uncertainty into the determination of the class label, even with perfect knowledge of $\mathcal{D}$. In the case of perfect information, i.e., $\mathbb{P}(\hat{C}(X) = y \mid X) = 1$, notice that Equation (2.4.2) reduces to Equation (2.4.1).

2.4.1 Optimal Bayes Classifier

A more useful tool allows for us to characterize the best possible outcome of a prediction in the case of less than perfect information. The quantity of interest is then the conditional probability of the dataum X having class label k. From a probabilistic perspective, we are interested in

$$\mathbb{P}(y = k \mid X) = \frac{\mathbb{P}(X \mid y = k)\mathbb{P}(y = k)}{\mathbb{P}(X)}, \tag{2.4.3}$$

which follows from Bayes theorem. Consequently, we may minimize the 0/1-loss of Equation (2.4.2) by maximizing the probability in Equation (2.4.3), and classify data points according to the class that yields the maximum probability. Under certain independence [43] or dependence [44] assumptions on the data, this is the optimal choice of classifier. As is frequently the case when working with real-world data, we cannot make these independence/dependence assumptions on the data, and must devise another classification rule.

For a classification problem, Bayes theorem tells us to compute the quantity of interest, the posterior or conditional probabilities for each class, is all that is necessary to predict the class of any $X \in \mathcal{X}$. From it, we then label X with the most probable class according to the conditional probability. For example, define the quantity $\pi(X) = \mathbb{P}(y = +1 \mid X)$, for any pair $(X, y) \sim \mathcal{D}$, and if we are interested in the case $y = +1$, then the chance of our prediction being incorrect is $1 - \pi(X)$. Similarly, if we want to predict $y = -1$, we are incorrect with probability $\pi(X)$. So to minimize our misclassification error, it is logical to predict using the rule:

$$h_{opt}(X) = \begin{cases} +1 & \text{if } \pi(X) > \frac{1}{2} \\ -1 & \text{if } \pi(X) < \frac{1}{2} \end{cases} \tag{2.4.4}$$

which is known as the optimal Bayes classifier [43, 45, 44]. This classifier may equivalently be written as the Bayes factor

$$h_{opt}(X) = \begin{cases} 1 & \text{if } \frac{\pi(X)}{1-\pi(X)} > 1 \\ -1 & \text{if } \frac{\pi(X)}{1-\pi(X)} < 1, \end{cases} \tag{2.4.5}$$

as the ratio between the two competing hypotheses. The error of the Bayes classifier is the optimal error rate [43, 46, 47], and is given by

$$\text{err}(h_{opt}) = \mathbb{E}[\min\{\pi(X), 1 - \pi(X)\}],$$

where $\text{err}(C) = \mathbb{P}(C(X) \neq y)$ for any $(X, y) \sim \mathcal{D}$.

The goal of any classifier is to minimize the $0/1$ misclassification rate. Recalling that $C(X)$ yields the true class label of X, and $\hat{C}(X) = \hat{y}$ is the class assigned to X by the classifier $\hat{C}$ for any $(X, y) \sim \mathcal{D}$. Then the $0/1$-loss of $\hat{C}$ is

$$\begin{aligned}\mathbb{E}[M(\hat{C})] &= 1 - \mathbb{E}[\mathbb{1}_{\{\hat{y} \neq y\}} \,|\, \mathcal{X} = X] \\ &= 1 - \mathbb{P}(\hat{C}(X) \neq y \,|\, \mathcal{X} = X) \\ &= \mathbb{P}(\hat{C}(X) = y \,|\, \mathcal{X} = X)\end{aligned}$$

where

$$\mathbb{1}_{\{\varepsilon\}} = \begin{cases} 0 & \text{if } \varepsilon \text{ is true} \\ 1 & \text{if } \varepsilon \text{ is false.} \end{cases}$$

Our goal is to construct a classification rule $\hat{C}$ that approximates the optimal error rate of the Bayes classifier

$$C^*(X) = \underset{Y \in \{-1,1\}}{\operatorname{argmax}} \, \mathbb{P}(C(X) = y \,|\, \mathcal{X} = X). \tag{2.4.6}$$

We see that this rate can be achieved precisely when $\hat{C}(X) = y$, for any pair $(X, y) \sim \mathcal{D}$. Additionally, in the case $K = 2$, Equation (2.4.6) agrees with the formulations in Equation (2.4.4) and Equation (2.4.5). The question of how to construct a classification rule yielding the optimal error rate remains. One way to construct such a rule is through the Boosting methodology of [41, 47]. This process of constructing a classification rule asymptotically achieves the optimal error rate [48, 47, 42] we seek. We will first give some discussion about simpler classification models.

2.4.2 Linear Regression

One way we may predict the likelihood in Equation (2.4.3) is by partitioning the input space using linear decision boundaries. These boundaries may by found through linear regression, via least squares, or through maximum likelihood estimation. For the case of least squares, we make predictions by the rule

$$\hat{y} = \hat{\beta}_0 + \sum_{i=1}^{N} X_i \hat{\beta}_i,$$

where the vector of coefficients in the linear model is found via the relation

$$\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y},$$

if in the case that $\mathbf{X}^T\mathbf{X}$ is non-singular, where $\mathbf{X}$ is the $p \times N$ matrix of predictor variables $X \in \mathbb{R}^p$. This general framework applies for the K-class classification problem, in which we may write the response Y as an $N \times K$ indicator matrix, i.e. $\mathbf{Y} = \{0,1\}^{N\times K}$, where one entry in each row is a 1, denoting class label. We will then write the coefficients in matrix form $\hat{\mathbf{B}} \in \mathbb{R}^{(p+1)\times K}$ given by $\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$. For this section, we will present the K-class classification case, as the binary case is an easy special case, as there is only a single linear function.

We may write the fitted linear model for either case as $\hat{f}_k(X) = \hat{\beta}_{k0} + \hat{\beta}_k^T X$ for $k = 1, 2, \ldots, K$. So the decision boundary between any two classes k and j is the hyperplane given by $\{x : (\hat{\beta}_{k0} - \hat{\beta}_{j0}) + (\hat{\beta}_k - \hat{\beta}_j)^T x = 0\}$. For example, in the case where $K = 2$, we are interested in the separating hyperplane where the log-odds

$$\log \frac{\mathbb{P}(C(X) = 1 \mid \mathcal{X} = X)}{\mathbb{P}(C(X) = 2 \mid \mathcal{X} = X)} = \beta_0 + \beta^T X,$$

are precisely 0. By inverting the *logit* transformation, $\log\left(\frac{\pi}{1-\pi}\right)$, for some probability π, we may recover the posterior in Equation (2.4.3)

$$\begin{aligned}
\mathbb{P}(C(X) = 1 \mid \mathcal{X} = X) &= \frac{\exp(\beta_0 + \beta^T X)}{1 + \exp(\beta_0 + \beta^T X)} \\
\mathbb{P}(C(X) = 2 \mid \mathcal{X} = X) &= \frac{1}{1 + \exp(\beta_0 + \beta^T x)}.
\end{aligned}$$

In general for $K > 2$, it is easy to see that

$$\begin{aligned}
\mathbb{P}(C(X) = k \mid \mathcal{X} = X) &= \frac{\exp(\beta_{k0} + \beta_k^T X)}{1 + \sum_{i=1}^{K-1} \exp(\beta_{i0} + \beta_i^T X)}, k = 1, \ldots, K-1, \\
\mathbb{P}(C(X) = K \mid \mathcal{X} = X) &= \frac{1}{1 + \sum_{i=1}^{K-1} \exp(\beta_{i0} + \beta_i^T X)},
\end{aligned}$$

which sum to one.

This type of model is appealing due to the its simplicity and ease of understanding interactions between predictor variables, but it can fail if the data is not linear, or cannot be locally approximated by a linear function. In such cases, we need a more general form of regression, one that lets the data dictate the form of the approximation, as opposed to imposing a linear model.

2.4.3 Generalized Additive Model

A generalized additive model (GAM) relaxes the linear assumptions on the predictor $\mathbf{X}\beta$ with a more general functional form. In the case of a binary response, traditional linear regression relates the mean,

$\pi(X) = \mathbb{P}(y = 1 \mid X)$, to the predictor variables by the logit link function, i.e.,

$$\log\left(\frac{\pi(X)}{1-\pi(X)}\right) = \beta_0 + \sum_{i=1}^{p} X_i\beta_i.$$

In the case of a GAM, this relation takes the form

$$\log\left(\frac{\pi(X)}{1-\pi(X)}\right) = \alpha + \sum_{i=1}^{p} f_i(X_i), \tag{2.4.7}$$

where each of the functions f_i is some non-parametric smooth function and the intercept α takes the place of β_0. Two benefits of using a GAM as opposed to traditional linear logistic regression are:

1. Categorical values are easily incorporated into the model. For example, if one of the components of $x_i \in X$ takes one of K classes (categories), one may incorporate a K-level factor variable into the model. This may be fitted by a histogram type smoother [49].

2. The model allows for interactions between categorical and continuous variables to be easily modeled. For example, if we have reason to believe that X_i has different responses based on a class label, such as survived or not, we can then estimate both functions $f_{i,S}$ and $f_{i,N}$ as opposed to a single function f_i that may not capture the different interactions.

We remark that the functions f_j are estimated in some fashion, and the estimates can reveal the presence or nonlinear responses, or not. Furthermore, not all of the functions f_j need to be nonlinear. In the case that all f_j are linear, then we may take the more traditional course of logistic regression. The point is in the case of GAMs, we allow our data to dictate the functional form of the model.

Recalling our quantity of interest Equation (2.4.7), we will write $F(X) = \sum_{i=1}^{p} f_i(X_i)$ and the logit transformation ensures that all values of $F(x) \in [0, 1]$, thus yielding valid estimates of class probabilities. We may recover these probabilities by inverting to obtain

$$\pi(X) = \mathbb{P}(y = 1 \mid X) = \frac{\exp\{F(X)\}}{1 + \exp\{F(X)\}}.$$

In the classification methodology presented herein, we will consider the case where each of the functions f_i are simple functions, $b(x; \cdot)$, characterized by some pertinent parameter γ and with an associated constant η, hence we write

$$f_i(x) = \eta_i b(x; \gamma_i).$$

The model is then written

$$F_M(x) = \sum_{i=1}^{M} \eta_i b(x; \gamma_i).$$

Some examples of a GAM are:

- A single layer neural network, where $b(x; \gamma) = \sigma(\gamma_0 + \gamma_1^T x)$ and $\sigma(t) = 1/(1 + e^{-t})$ is the sigmoid function, parameterized by a linear combination of the input x.
- Multivariate adaptive regression splines, where γ parameterizes the variables and values at the knots of the spline.
- For tree-based classifiers, γ determines the split points and split variables, i.e., how the input is partitioned into different classes at each level, for the child nodes, and the predicted class at terminal nodes.

GAMs are typically fit by minimizing a loss function, $L(y, f(x))$, averaged over the training data

$$\min_{\{\eta_m, \gamma_m\}_i^M} \sum_{i=1}^{N} L\left(y_i, \sum_{m=1}^{M} \eta_m b(x_i; \gamma_m)\right), \tag{2.4.8}$$

for some coefficients η and functions $f(x; \gamma_m) \in \mathbb{R}$ parameterized by γ_m. Depending on the choice of loss function and/or basis, this can be computationally challenging at best, or intractable at worst.

For our fingerprinting application, we will fit the model in Equation (2.4.7) through a process called Boosting [41, 47]. This methodology creates an ensemble of weak learners, classification rules only slightly better than a random guess, to create an accurate classification rule. Specifically, we will use decision trees as the weak learner in the boosted ensemble.

2.4.4 Decision Trees

We give background information about decision trees, as they form basis for our ensemble classifier in the materials fingerprinting methodology of Section 4.1. For more details about decision trees in regression and classification problems, please see [50, 51, 46].

Decision trees are a method that recursively partitions the input space into rectangles and assigns a constant value to each partition. For each input variable X_i, we can traverse the tree to a specific terminal node according to the decision criteria at each split. So each X_i ending in the same partition is assigned the same class label. We write R_m as the partition formed by the m^{th} node. The algorithm creates two

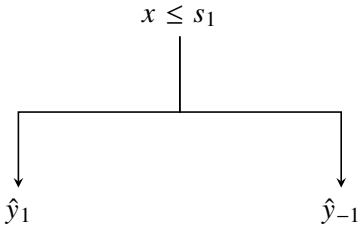


Figure 2.4: Example decision tree for one vector of predictor variables $x = (x, y)^T$, and associated predicted responses $\hat{y} \in \{-1, 1\}$.

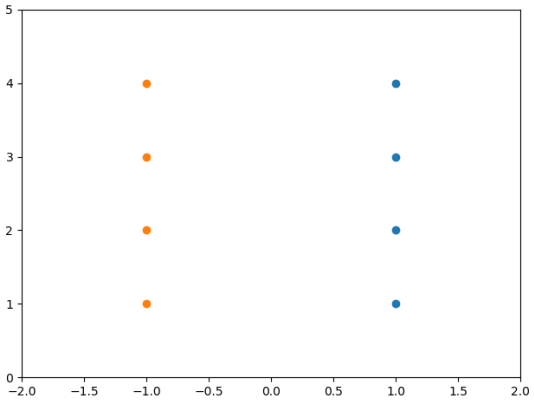


Figure 2.5: If s_1 is the y-coordinate in the tree to the left, then any split will include points from both classes.

child nodes from each node, until the terminal node is reached, by finding a binary split based on one of the predictor variables.

The first issue in constructing a tree-based classifier is how to create binary splits of the data. A split of the dataset is defined by a recursive partitioning of the data $\mathcal{X}$ into smaller disjoint subsets by considering some meaningful quantification of the data. These splits are chosen by a measure of 'purity', i.e., we want the population in each node to be homogeneous with respect to the response variable. For example, consider the datapoints in Figure 2.5 and a one-level tree, seen in Figure 2.4. In this example, the predictor variables are the (x, y)-coordinates of the data points. If we are to use a tree based classifier, we must determine the optimal split value s_1, the input from the predictor variables used to split the dataset into two classes. If s_1 is chosen to be any y-coordinate of the data, then any split will have members from both classes. Alternatively, if s_1 is chosen to compare any x-coordinates, then the child nodes would be homogeneous in their response variable, y_{R_i}, and the impurity measure would be zero for the latter and greater than zero for the former. Consequently, we would choose to split at any of the x-values, to create homogeneous child nodes. While such a simple example rarely occurs in practice, it is an illustrative example of how trees split the dataset.

Generally, the process of splitting a dataset involves iterating over each input variable X_i, checking if each feature of X_i is below or above the split value and assigning it to one of two child nodes. The split value is found through an exhaustive search over all input features as a possible splitting value. We then evaluate the cost of the split using either equation Equation (2.4.9) or equation Equation (2.4.10), and find the split that best separates the data into two child nodes, recalling that the goal is to create homogeneous subsets of the response variable y_i based on a split.

For example, suppose we have 20 observations for a binary classification problem with 10 observations per class. Now suppose our tree created two splits: (15, 5) and (5, 15), while the other split produced (10, 10) and (20, 0). In this case, the misclassification rate for both splits is 25%, but the second split is preferred, as its second node contains only observations of one class. We want to grow splits with the smallest impurity, and avoid the situation where classes are mixed together.

Algorithm 2.3 Find split

Select the best split point for a dataset $\mathbf{X} \in \mathbb{R}^{N \times d}$, N observations in d-dim space.
for $i = 1, \ldots, N$ **do**
  for $j = 1, \ldots, d$ **do**
    $\hat{y}_i =$ `test_split`$(\mathbf{X})$
    gini = `Gini_index`$(\hat{y}_i, y_i)$
  end for
end for

In Algorithm 2.3, the function `test_split` tests a proposed split by iterating over each row, checking if the attribute value is below or above the split value, then and assigning the observation to either $\hat{y}_1$ or $\hat{y}_{-1}$. After creating this proposed split, we compute the impurity measure and track the split variable and value returning the smallest measure of impurity.

Now the splits in the m^{th} node of a tree T are determined through either the Gini index

$$Q_m(T) = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}), \tag{2.4.9}$$

or cross-entropy

$$Q_m(T) = -\sum_{k=1}^{K} \hat{p}_{mk} \log(\hat{p}_{mk}) \tag{2.4.10}$$

for $\hat{p}_{mk} = \frac{1}{N_m} \sum_{X_i \in R_m} \mathbb{1}_{\{y_i = k\}}$, which is the proportion of class k observations in the m^{th} node, in a $k = 1, \ldots, K$-class classification problem. One benefit to using either of these measures of impurity in equation Equation (2.4.9) and equation Equation (2.4.10) as opposed to the 0/1 misclassification rate, given by

$$\frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{\{\hat{y}_i \neq y_i\}}.$$

where $\hat{y}_i$ is the predicted class and y_i denotes the true class of the ith datum, is that the Gini index and cross-entropy are differentiable, and are therefore more amenable to numerical optimization algorithms. Consequently, they are more commonly used in practice.

As noted in [46, 48], trees suffer from high variance. A small perturbation in the data can have a large change on the resulting classification, as the tree could choose a different set of splits. Noting the hierarchical nature of the tree, this small perturbation is passed to all child nodes, resulting in a different end result. Building an ensemble of 'stumps', short trees consisting of one or two levels which are the type used in boosting algorithms, reduces the variance of the final classification rule [46, 47]. We use a collection of classification trees in the AdaBoost classifier in our materials fingerprinting method.

2.4.5 AdaBoost Algorithm

The AdaBoost process is summarized in Algorithm 2.4. The first base classifier g_1 is trained with equal weights on each observation, and is the same process as used for training a single classifier. In all future iterations the weights $w_i^{(m)}$ increase for those observations that are misclassified, and are reduced for those observations that are correctly classified. The end result is that subsequent base classifiers are forced to put greater emphasis on those observations that have been incorrectly classified. The situation is analogous to a classroom environment. Those students that need more help with the material receive more of an instructor's time as compared with those students who are quick learners.

The quantities ϵ_m are weighted measures of the misclassification rates for the base classifiers as measured on the data. As a direct consequence, the α_m terms give greater influence to the more accurate base classifiers in the final majority vote in line 8 of the algorithm.

Algorithm 2.4 AdaBoost.M1

1: Initialize observation weights: $w_i = \frac{1}{N}, 1 \leq i \leq N$
2: **for** $m = 1, \ldots, M$ **do**
3: Fit classifier $g_m(x)$ to training data $\mathcal{X}$ with weights w_i
4: Compute error

$$\epsilon_m = \frac{\sum_{i=1}^{N} w_i \mathbb{1}_{\{y_i \neq g_m(x_i)\}}}{\sum_{i=1}^{N} w_i}.$$

5: Compute $\alpha_m = \log((1-\epsilon_m)/\epsilon_m)$.
6: Update weights

$$w_i \leftarrow w_i \cdot \exp\left\{\alpha_m \mathbb{1}_{\{y_i \neq g_m(x_i)\}}\right\}, \qquad 1 \leq i \leq N.$$

7: **end for**
8: Output $g(x) = \texttt{sign}\left(\sum_{i=1}^{M} \alpha_m g_m(x)\right)$

The AdaBoost algorithm is flexible and can take most any base classifier to build the model. We use an collection of classification trees for our classification problem. While trees are simple and easily interpreted,

they suffer from high variance [48, 46]. Building an ensemble of 'stumps', short trees which are the type used in boosting algorithms, reduces the variance of the final classifier.

AdaBoost is an algorithmic method for solving the optimization problem:

$$\min_{F \in \mathcal{F}} \sum_{i=1}^{N} L(y_i, F(x_i))$$

where

$$L(y, F(x)) = \exp\{-y\, F(x)\} \tag{2.4.11}$$

over all functions $F \in \mathcal{F}$ where $F(x) : \mathcal{X} \to \mathbb{R}$ is a weighted linear combination of some sufficiently smooth functions, i.e., $F_M(x) = \sum_{i=1}^{M} c_m f_m(x)$, and $\mathcal{F}$ is the space of all such functions. In practice, this takes the form

$$L(F) = \frac{1}{N} \sum_{i=1}^{N} \exp\{-y_i F(x_i))\},$$

where

$$F_M(x) = \frac{1}{M} \sum_{i=1}^{M} \alpha_m g_m(x).$$

Here we write $g_m(x)$ as the weak hypothesis (classifier) such that $g_m : \mathcal{X} \to Y$ and $\exists \eta > 0$ such that $M(g) = \frac{1}{2} - \eta$. Writing the function F in such a form leads us naturally to see Boosting through as fitting a generalized additive model [48, 46]. The key to AdaBoost's success is the way it fits an additive model (Section 2.4.3) by Forward Stagewise Additive Modeling [48].

2.4.6 Forward Stagewise Additive Modeling

Forward Stagewise Additive Modeling approximates the solution to a general loss function $L(y, F)$, averaged over the training data, typically of the form Equation (2.4.8) for some coefficients β and basis functions $b(x; \gamma_m)$.

One of the keys to the success of AdaBoost is instead of solving Equation (2.4.8) using sophisticated numerical optimization techniques, AdaBoost constructs the optimal function $F(x)$ by adding one basis function at a time. It seeks

$$\min_{\beta, \gamma} \sum_{i=1}^{N} L(y_i, \beta b(x_i; \gamma)),$$

at each iteration. Specifically, AdaBoost seeks to minimize the exponential loss function Equation (2.4.11). We will first show how to minimize this objective function using forward stagewise additive modeling, then give some discussion of this objective function.

Forward stagewise additive modeling approximates the solution to Equation (2.4.8) by sequentially augmenting the basis function expansion without altering existing functions or their coefficients. The process works by solving for the optimal basis function $b(x;\gamma_m)$ and associated coefficient at the m^{th} iteration to add to the existing expansion of $F_{m-1}(x)$, yielding $F_m(x)$. This process only adds terms to the basis and does not alter previously added terms.

In our implementation of AdaBoost, the individual basis functions are classifiers themselves $g_m \in \{-1, 1\}$. We use tree classifiers, discussed in Section 2.4.4, and as noted previously the parameter γ represents: how the splits are chosen, where to split the input variables, the class of each terminal node and number of terminal nodes of each tree. These quantities are fixed in our implementation, so γ_m can be taken as a constant. Using the exponential loss in Equation (2.4.11), the minimization problem takes the form

$$(\beta_m, g_m) = \underset{\beta, g}{\operatorname{argmin}} \sum_{i=1}^{N} \exp\left\{-y_i(F_{m-1}(x_i) + \beta g(x_i))\right\},$$

for classifier g_m and coefficient β_m added to the basis at the m^{th} step. We may equivalently write this as

$$(\beta_m, g_m) = \underset{\beta, g}{\operatorname{argmin}} \sum_{i=1}^{N} w_i^{(m)} \exp\left\{-y_i \beta g(x_i)\right\}, \tag{2.4.12}$$

where $w_i^{(m)} = \exp\{-y_i\, F_{m-1}(x_i)\}$ by noting the preceding term does not depend on β nor $C(x)$. As the weights depend only on the previous iterations, they can be viewed as an adaptive weight for each observation.

Then alternating between solving for $g_m(x)$ and β_m, the updated approximation to $F^*(x)$ is $F_m(x) = F_{m-1}(x) + \beta_m g_m(x)$, which is Forward Stagewise Additive Modeling, detailed in Algorithm 2.5. To see this, note for any $\beta > 0$ that the optimal classifier g_m^* in Equation (2.4.12) is

$$g_m^* = \underset{g \in \mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{N} w_i^{(m)} \mathbb{1}_{\{y_i \neq g(x_i)\}}, \tag{2.4.13}$$

which is precisely the classifier minimizing the weighted misclassification rate in predicting y. Recalling that $g(x) \in \{-1, 1\}$ and noting that whenever $y_i \neq g(x_i)$, $\texttt{sign}(-y_i g(x_i)) = -1$, and so $\exp\{-y_i \cdot \beta g(x_i)\} = \exp\{\beta\}$.

Now by splitting the sum, we may alternatively express Equation (2.4.12) as

$$\begin{aligned}\underset{\beta,g}{\operatorname{argmin}} \sum_{i=1}^{N} w_i^{(m)} \exp\{-y_i \beta g(x_i)\} &= e^{-\beta} \sum_{y_i = g_m(x_i)} w_i^{(m)} + e^{\beta} \sum_{y_i \neq g_m(x_i)} w_i^{(m)} \\ &= (e^{\beta} - e^{-\beta}) \sum_{i=1}^{N} w_i^{(m)} \mathbb{1}_{\{y_i \neq g(x_i)\}} + e^{-\beta} \sum_{i=1}^{N} w_i^{(m)}.\end{aligned}$$

Substituting the g_m^* from Equation (2.4.13) into Equation (2.4.12) and taking the partial derivative with respect to β, setting it equal to zero, and solving for β we see that

$$\beta_m^* = \frac{1}{2} \log\left(\frac{1 - \epsilon_m}{\epsilon_m}\right), \tag{2.4.14}$$

where we have defined

$$\epsilon_m = \frac{\sum_{i=1}^{N} w_i \mathbb{1}_{\{y_i \neq g_m(x_i)\}}}{\sum_{i=1}^{N} w_i}.$$

Notice that ϵ_m creates a distribution over the misclassified data points.

We may then update the approximation

$$F_m(x) = F_{m-1}(x) + \beta_m g_m(x)$$

and weights

$$w_i^{(m+1)} = w_i^{(m)} \exp\{-\beta_m y_i g_m(x_i)\}.$$

Since $-y_i g_m(x_i) = 2 \cdot \mathbb{1}_{\{y_i \neq g_m(x_i)\}} - 1$ we have that

$$w_i^{(m+1)} = w_i^{(m)} \exp\{\beta_m \mathbb{1}_{\{y_i \neq g_m(x_i)\}}\} \exp\{-\beta_m\}, \tag{2.4.15}$$

defining $\alpha_m = 2\beta_m$, which is the same as line 4 of Algorithm 2.4. As all weights are scaled by the same factor $\exp\{-\beta_m\}$, Equation (2.4.15) is equivalent to line 6 in Algorithm 2.4.

2.4.7 Optimality of AdaBoost

AdaBoost iteratively approximates the optimal Bayes classifier by creating an ensemble of weak classifiers, i.e., classification rules that are only slightly better than random guessing, and combining them to make accurate predictions. The final output from the algorithm is a strong learner, i.e., an accurate classification rule, comprised of a weighted linear combination of weak classifiers. The algorithm approximates the

Algorithm 2.5 Stagewise Additive Modeling

1: Initialize $f_0(x) = 0$.
2: **for** $i = 1, \ldots, M$ **do**
3: Compute

$$(\beta_m, \gamma_m) = \underset{\beta,\gamma}{\operatorname{argmin}} \sum_{i=1}^{N} L(y_i, f_{m-1}(x_i) + \beta b(x_i; \gamma))$$

4: Set $f_m(x) = f_{m-1}(x) + \beta_m b(x; \gamma_m)$
5: **end for**

optimal classifier Equation (2.4.6) by iteratively minimizing the exponential error function over $\mathcal{F}$, the space of functions that can be represented by a linear combination of base classifiers. Hence minimizing

$$L(y, F(x)) = \exp\{-y\, F(x)\}$$

yields a simple update rule and is related to the log-likelihood. We may see this relationship by observing that

$$F^*(x) = \underset{F \in \mathcal{F}}{\operatorname{argmin}}\, \mathbb{E}[\exp\{-yF(x)\} \mid \mathcal{X} = x]$$

where

$$F^*(x) = \frac{1}{2} \log \frac{\mathbb{P}(y = +1 \mid \mathcal{X} = x)}{\mathbb{P}(y = -1 \mid \mathcal{X} = x)}. \tag{2.4.16}$$

We may see this by recalling the connection between probabilities and expectations via indicator functions, Then by setting the partial derivatives of Equation (2.4.11) with respect to y equal to zero and solving for F, we may then find the optimal solution F^*. Furthermore, by considering $\texttt{sign}(F^*)$, we can see AdaBoost achieves the Bayes error rate, g^* in Equation (2.4.6). Hence, in the case of binary classification where $y_i \in \{-1, 1\}$, we are looking for the best approximation of the log-odds ratio.That is

$$\mathbb{P}(y = +1 \mid \mathcal{X} = x) = \frac{\exp\{F^*(x)\}}{\exp\{-F^*(x)\} + \exp\{F^*(x)\}} \tag{2.4.17}$$

$$\mathbb{P}(y = -1 \mid \mathcal{X} = x) = \frac{\exp\{-F^*(x)\}}{\exp\{-F^*(x)\} + \exp\{F^*(x)\}}. \tag{2.4.18}$$

We see that F^* is half the log-odds, i.e., $\log(\pi/(1-\pi))$, for a probability π. Hence, both Equation (2.4.11) and the log-loss, the negative log-likelihood, are minimized by Equation (2.4.16).

We may see this optimality of AdaBoost in another way. We want to minimize the true loss, i.e., the expected loss with respect to $\mathcal{D}$, given by $\mathbb{E}[\exp\{yF(X)\}]$. We may write this expectation as the iterated

expectation

$$\mathbb{E}\left[\mathbb{E}[\exp\{-yF(X)\} \mid X]\right] = \mathbb{E}[\pi(X)\exp\{-F(X)\} + (1-\pi(X))\exp\{F(X)\}], \tag{2.4.19}$$

where $\pi(X) = \mathbb{P}(y = +1 \mid X)$. By setting the partial derivative with respect to $F(X)$ equal to 0, then solving for F^*, we see that

$$\begin{aligned} 0 &= -\pi(X)\exp\{-F(X)\} + (1-\pi(X))\exp\{F(X)\} \\ 2F(X) &= \log\left(\frac{\pi(X)}{1-\pi(x)}\right). \end{aligned}$$

Hence

$$F^*(X) = \frac{1}{2}\log\left(\frac{\pi(X)}{1-\pi(X)}\right). \tag{2.4.20}$$

We note that $F^* \in \mathbb{R} \cup \{\pm\infty\}$, in the case that $\pi(X)$ is 0 or 1.

Remark 2.3. *We remark on the similarity between Equation (2.4.20) and Equation (2.4.14). What we see here is optimal weight update in the classification scheme is precisely the optimal function that minimizes the exponential loss function.*

Now substituting this expression into the RHS of Equation (2.4.19), we may then write

$$\begin{aligned} \mathbb{E}\left[\mathbb{E}[\exp\{-yF(x)\} \mid x]\right] &= \mathbb{E}\left[\pi(x)\exp\left\{-\frac{1}{2}\log\left(\frac{\pi(x)}{1-\pi(x)}\right)\right\} + (1-\pi(x))\exp\left\{\frac{1}{2}\log\left(\frac{\pi}{1-\pi(x)}\right)\right\}\right] \\ &= \mathbb{E}\left[\pi(x)^{\frac{1}{2}}(1-\pi(x))^{\frac{1}{2}} + (1-\pi(x))\left(\frac{\pi(x)}{1-\pi(x)}\right)^{\frac{1}{2}}\right] \\ &= 2\mathbb{E}\left[\sqrt{\pi(x)(1-\pi(x))}\right], \end{aligned}$$

which is the optimal expected exponential loss. Recalling the optimal Bayes classifier Equation (2.4.4), we see that it is precisely $\texttt{sign}(F^*)$. Hence by minimizing the exponential loss over the data we may recover the optimal classifier by considering $\texttt{sign}(F^*)$. What we see is that it is the best predictor over all $\mathbb{R}$-valued functions, not just those that are linear combinations of basis classification functions.

2.5 Variational Bayes

Before delving into the standard machinery of variational inference, we will give some discussion about the Expectation-Maximization (EM) algorithm, including a more modern treatment, and the connection to Bayesian inference.

2.5.1 Expectation Maximization

The EM algorithm was introduced by [52] as a methodology for finding maximum likelihood estimates of likelihood models that can be factored as

$$h(Y \mid \theta) = \int_{\mathcal{Z}} f(Y, Z \mid \theta) \mathrm{d}Z,$$

for observed data Y, parameters θ, and latent variable $Z \in \mathcal{Z}$. The methodology monotonically increases the value of the likelihood, and achieves a maximum likelihood, which may be local or global. We refer the interested reader to the literature [53, 52, 46, 54] for an in-depth discussion of the process. We will outline the process generally, and will discuss the view presented by [55], as it is directly related to our methodology.

A general formulation of the EM-algorithm maximizes the log-likelihood, $\ell(\theta; Y)$, based on observed data Y and parameters θ. We will write the complete data as $S = (Y, Z)$, for missing or latent data Z. Now defining $Q(\tilde{\theta}, \theta) = \mathbb{E}[\ell_0(\tilde{\theta}; S) \mid Z, \theta]$, for the complete data log-likelihood ℓ_0, the EM algorithm proceeds as in Algorithm 2.6

Algorithm 2.6 EM-Algorithm

1: Initial guess for parameters θ_0
2: **for all** $t > 0$ **do**
3: E-Step: Compute

$$Q(\widetilde{\theta}, \theta_t) = \mathbb{E}[\ell_0(\widetilde{\theta}; S) \mid Y, \theta^{(t)}]$$

as a function of $\widetilde{\theta}$.
4: M-Step: Compute

$$\theta^{(t+1)} = \underset{\widetilde{\theta}}{\operatorname{argmax}}\, Q(\widetilde{\theta}, \theta^{(t)})$$

5: Iterate until convergence, i.e., a fixed point of Q, is found.
6: **end for**

The variant of the EM-algorithm presented in [55] maximizes a joint function of the distribution over the latent variables and of the parameters in the model. Hence, it is sometimes referred to a maximization-maximization process. What is different in its approach is that the authors adopt the view that this maximization process is analogous to minimizing the "free energy"' in statistical physics, which in turn

can be seen as minimizing the Kullback-Leibler divergence [56, 57, 55]. In this view, each latent variable is governed by its own variational factor, i.e., $q(Z) = \prod_{i=1}^{N} q_i(z_i)$. Continuing with this interpretation, we may see how the factorization $q(Z, \theta) = q(Z)q(\theta)$ of the variational density may be justified.

The latent variable model under investigation is of the form $z_i \rightarrow y_i \leftarrow \theta$, i.e., the observed datum y_i is dependent on the latent variable z_i and some model parameters θ, where θ represents all of the model's parameters for simplicity of notation. As discussed in Section 2.5.1, in the E-step, we make inferences employing the variational posterior over the latent variables $q(z_i \mid y_i)$ and then compute the sufficient statistics of the model parameters. Through this process of marginalizing over θ we obtain a distribution over θ, as opposed to the *maximum a posteriori* (MAP) estimate from the traditional EM-algorithm, (Algorithm 2.6). This marginalizing over the parameters yields a bound on the evidence, the marginal likelihood, see Equations (2.5.3)–(2.5.5). The process also gives equal standing in the model between latent variables and the model parameters, unlike traditional EM, which yields a point estimate, specifically the MAP estimate [45, 54].

This factorization allows for exchanging a stochastic dependence between θ and Z for a deterministic relationship between pertinent moments of these random variables [56]. Bypassing any interactions between θ and Z yields an analytical approximation of the log likelihood. While this mean-field statistical physics approximation allows us to see the marginal densities of the latent variables, it cannot capture correlations between them.

Remark 2.4. *It is worthwhile to pause and examine these assumptions on the approximating density q to see if they are reasonable, and if we're gaining computational convenience at the cost of over-simplifying our model. The variational Bayes EM relies on the mean-field variational approximation*

$$p(\theta, Z \mid Y) \approx q(\theta)q(Z) = q(\theta) \prod_{i=1}^{N} q(z_i).$$

In our statistical model, Equation (3.2.2), we have assumed that C, which can be recovered by taking an expectation with respect to Z, see Section 2.5.5, is independent of θ, and that the observations are conditionally independent with respect to the correspondence C. The mean-field assumptions agree with or model and do not impose additional constraints on it.

2.5.2 Variational Expectation Maximization

Define

$$\mathcal{F}(\widetilde{Q}, \theta) := \mathbb{E}_{\widetilde{Q}}[\log(p(Y, Z \mid \theta))] + H(\widetilde{Q}), \tag{2.5.1}$$

for $\widetilde{Q} := q(Z \mid Y, \theta)$, and $H(\widetilde{Q})$ is the entropy of $\widetilde{Q}$. In keeping with our previous notation, θ represents the model parameters, Y, the observed data, and Z, the latent variables. Factoring the variational density allows for computation of the expectation with respect to the latent variables in the E-step and the M-step maximizes the function with respect to the model parameters. This view of the EM-algorithm as a variational process was first proposed by [55], and the variational expectation-maximization routine the proceeds as in Algorithm 2.7.

Algorithm 2.7 Variational EM

1: **for all** $t > 0$ **do**
2: E-Step: Compute
$$\widetilde{Q}^{(t)} = \underset{\widetilde{Q}}{\operatorname{argmax}}\, \mathcal{F}(\widetilde{Q}, \theta^{(t-1)})$$
3: M - Step: Compute
$$\Theta^{(t)} = \underset{\theta}{\operatorname{argmax}}\, \mathcal{F}(\widetilde{Q}^{(t)}, \theta)$$
4: **if** $\mathrm{KL}(\widetilde{Q}^{(t)} \parallel p) < \epsilon$ **then** Break
5: **end if**
6: **end for**

Adopting this perspective, the algorithm seeks to minimize the Kullback-Leibler (KL) divergence. To this equivalence, first we relate $\mathcal{F}$ to the KL-divergence. Recall that the Kullback-Leibler divergence between probability distributions $\mathcal{Q}$ and $\mathcal{P}$ with probability density functions q, p respectively is

$$\mathrm{KL}(\mathcal{P} \parallel \mathcal{Q}) = \int_{\Omega} p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x. \tag{2.5.2}$$

Now to see the connection to the KL-divergence, consider the following.

$$\log(p(Y)) = \log\left(\int p(Y, Z, \theta) \mathrm{d}\theta \mathrm{d}Z\right) \tag{2.5.3}$$
$$= \log\left(\int q(Z, \theta) \frac{p(Y, Z, \theta)}{q(Z, \theta)} \mathrm{d}\theta \mathrm{d}Z\right) \tag{2.5.4}$$
$$\geq \int q(Z, \theta) \log\left(\frac{p(Y, Z, \theta)}{q(Z, \theta)}\right) \mathrm{d}\theta \mathrm{d}Z, \tag{2.5.5}$$

where the inequality follows from Jensen's inequality [58]. Substituting the optimal choice of q, i.e., $q^*(Z, \theta) = p(Z, \theta \mid Y)$, changes the inequality in Equation (2.5.5) to equality. This choice does not simplify the problem however, since it requires knowledge of the exact form of the posterior $p(Z, \theta \mid Y)$, which in turn, requires its normalizing constant, the marginal likelihood. Recalling that we require the approximate

density to be separable, $q(Z, \theta) = q(Z)q(\theta)$, we may then write

$$\begin{aligned}
\log(p(Y)) &\geq \int q(Z)q(\theta) \log\left(\frac{p(Y, Z, \theta)}{q(Z)q(\theta)}\right) \mathrm{d}\theta \mathrm{d}Z \\
&= \int q(Z)q(\theta) \log\left(\frac{p(Y, Z \mid \theta)}{q(Z)}\right) + \log\left(\frac{p(\theta)}{q(\theta)}\right) \mathrm{d}Z\mathrm{d}\theta \\
&= \mathcal{F}(q(Z), q(\theta))
\end{aligned}$$

and hence,

$$\begin{aligned}
\log(p(Y)) - \mathcal{F}(q(Z), q(\theta)) &= \int q(Z)q(\theta) \log\left(\frac{q(Z)q(\theta)}{p(Z, \theta \mid Y)}\right) \mathrm{d}Z\mathrm{d}\theta \\
&= \mathrm{KL}(q(Z, \theta) \parallel p(Z, \theta \mid Y)) \\
&\geq 0.
\end{aligned}$$

In Algorithm 2.7, minimizing the KL-divergence is equivalent to maximizing the functional $\mathcal{F}$ in Equation (2.5.1). The methodology yields a unique maximizing distribution $\widetilde{Q}^*$, and that if $\mathcal{F}(\widetilde{Q}, \theta)$ has a local, global resp., at $\widetilde{Q}^*, \theta^*$, then the likelihood, $\log Q(Z \mid \theta)$ is a local, global resp., maximum [55].

Our goal is to maximize the likelihood function, $\mathcal{L}$, with respect to the variational distribution q, which is equivalent to minimizing $\mathrm{KL}(q \parallel p)$. In general however, the KL divergence is intractable, and the ELBO (evidence lower bound) is maximized instead. We may decompose the variational likelihood, ignoring any terms that depend on θ for the moment, as

$$\begin{aligned}
\mathcal{L}(q) &= \int q(Z) \log\left(\frac{p(Y, Z)}{q(Z)}\right) \mathrm{d}Z && (2.5.6) \\
&= \mathbb{E}_q[\log(p(Z \mid Y))] - \mathrm{KL}(q \parallel p) && (2.5.7) \\
&= \int q(Z) \log\left(p(Y, Z)\right) - q(Z) \log(q(Z)) \mathrm{d}Z && (2.5.8) \\
&= \mathbb{E}_q[\log(p(Y, Z))] + H(q), && (2.5.9)
\end{aligned}$$

where $H(q)$ is the entropy of q. What we see here from Equation (2.5.9) is the trade-off between two terms. The expectation forces $q(Z)$ to be large when the the joint distribution $p(Y, Z)$ is large. The entropy term wants the variational distribution $q(Z)$ to be diffuse and spread out over the space. Equation (2.5.9) is known in the literature as the ELBO and is, in general, how convergence is determined for variational Bayesian methods. Also note that $\mathcal{L}(q)$ is the expectation of the complete log-likelihood, which is montonically increased to a maximal value by the expectation-maximization method of [52].

2.5.3 Variational Inference for GMMs

We will illustrate the process of variational inference through a detailed example of a Gaussian mixture model [53, 57]. This model forms the basis for our labeled point-set registration methodology of Section 4.5. A Gaussian mixture model may be written as a linear sum of say N Gaussian densities, each with their own mean, μ_n, and covariance, Λ_n, such that

$$p(x) = \sum_{n=1}^{N} \omega_n \mathcal{N}(x \mid \mu_n, \Lambda_n),$$

with mixing coefficient $\{\omega_n\}_{n=1}^{N}$, such that $0 \leq \omega_n \leq 1$ for all $n \leq N$, and $\sum_{n=1}^{N} \omega_n = 1$.

Let us now define a binary latent variable Z such that $z_n \in \{0, 1\}^N$ and $\sum_n^N z_n = 1$. Clearly, there are N possible states where z_n could equal 1, and the remaining elements are 0. By defining a joint distribution $p(X, Z) = p(X \mid Z)p(Z)$ and considering the marginal distribution over Z with respect to the mixing coefficients of the GMM we see that

$$p(z_n = 1) = \omega_n, \tag{2.5.10}$$

where $0 \leq \omega_n \leq 1$ and $\sum_n^N \omega_n = 1$, in order for the marginal, Equation (2.5.10), to be a valid probability. Thus we may write

$$p(Z) = \prod_{n=1}^{N} \omega_n^{z_n}.$$

Furthermore by the binary construction of Z

$$p(Z \mid z_n = 1) = \mathcal{N}(X \mid \mu_n, \Lambda_n),$$

and hence

$$\mathcal{L}(X \mid Z) = \prod_{n=1}^{N} \mathcal{N}(X \mid \mu_n, \Lambda_n)^{z_n}. \tag{2.5.11}$$

Lastly the joint distribution over X and Z may be factored as $p(X, Z) = p(X \mid Z)p(Z)$, thus yielding the marginal

$$p(X) = \sum_{n=1}^{N} p(X \mid z_n)p(z_n) = \sum_{n=1}^{N} \omega_n \mathcal{N}(X \mid \mu_n, \Lambda_n),$$

which is again a Gaussian mixture.

2.5.4 Prior distibutions

We choose to model the prior over the vector of GMM weights as a Dirichlet distribution, which is a conjugate prior, given by

$$p(\omega) = \mathrm{Dir}(\omega \mid \alpha_0) = C(\alpha_0) \prod_{i=1}^{N} \omega_i^{\alpha_0 - 1}, \tag{2.5.12}$$

where $C(\alpha_0)$ is the normalization constant of the Dirichlet distribution. The parameter α_0 determines the effective number of observations associated with each point in the reference. That is to say, does there exist information that favors associating observations with specific points in the reference. Choosing α_0 small lets the posterior be influenced by the data, as opposed to our prior beliefs.

For the mean and precision of Gaussian component, we use a Gaussian-Wishart prior

$$p(\mu, \Lambda) = p(\mu \mid \Lambda) p(\Lambda) \tag{2.5.13}$$

$$= \prod_{i=1}^{N} \mathcal{N}(\mu_i \mid m_0, (\beta_0 \Lambda_i)^{-1}) \mathcal{W}(\Lambda_i \mid W_0, \nu_0). \tag{2.5.14}$$

Here we write m_0, W_0 as the initial values for the mean and precision, and β_0 and ν_0 are the initial scale and degrees of freedom of the precision matrices respectively. We typically choose $m_0 = 0$ for symmetry.

2.5.5 Variational Posterior Distribution

We are interested in a conditional density given the observations of all random variables: μ, Λ, Z and ω. Thus,

$$p(Y, Z, \omega, \mu, \Lambda) = p(Y \mid Z, \mu, \Lambda) p(Z \mid \omega) p(\omega) p(\mu \mid \Lambda) p(\Lambda), \tag{2.5.15}$$

is our quantity of interest. This distribution is not tractable however, and, following the discussion in the previous section, instead seek a variational distribution $\mathbb{Q}$ such that

$$q(Z, \omega, \mu, \Lambda) = q(Z) q(\omega, \mu, \Lambda)$$

Now writing the optimal distribution as q^* and noting that

$$\log(q^*(Z)) = \mathbb{E}_{\omega,\mu,\Lambda}\left[p(Y, Z, \mu, \Lambda, \omega)\right] \tag{2.5.16}$$

$$= \mathbb{E}_{\omega}\left[\log(p(Z \mid \omega))\right] + \mathbb{E}_{\mu,\Lambda}\left[\log(p(Y \mid Z, \mu, \Lambda))\right] + \zeta \tag{2.5.17}$$

$$= \sum_{m=1}^{M}\sum_{n=1}^{N} z_{mn}\log(\rho_{mn}) + \zeta \tag{2.5.18}$$

by absorbing any term independent of Z into the constant ζ and by substituting the conditional distributions.

Here we have defined

$$\log(\rho_{mn}) = \mathbb{E}[\log(\omega_n)] + \frac{1}{2}\mathbb{E}[\log(|\Lambda_n|)] - \frac{d}{2}\log(2\boldsymbol{\pi}) - \frac{1}{2}\mathbb{E}_{\mu_n,\Lambda_n}[\|Y_m - \mu_n\|^2_{\Lambda_n}]. \tag{2.5.19}$$

Exponentiating both sides of Equation (2.5.18), we see

$$q^*(Z) \propto \prod_{m=1}^{M}\prod_{n=1}^{N} \rho_{mn}^{z_{mn}}.$$

Requiring that q^* be a distribution, i.e., normalized, define

$$c_{mn} = \frac{\rho_{mn}}{\sum_{i=1}^{N} \rho_{mi}},$$

and as $z_{mn} \in \{0, 1\}$ we see that

$$q^*(Z) = \prod_{m=1}^{M}\prod_{n=1}^{N} c_{mn}^{z_{mn}}.$$

So the form of the optimal distribution q^* has the same form as the prior over the correspondence matrix Equation (2.5.11). Furthermore observe that $\mathbb{E}[z_{mn}] = c_{mn}$, and we can see that c_{mn} are the elements of a probabilistic correspondence matrix. By this we mean an assignment matrix that does not make hard, 0 or 1, assignments, but assigns a probability of each observation point being assigned a point in the reference set.

To update the model parameters in the algorithm, we first compute the sufficient statistics of the observed data, given the correspondence. These are given by

$$M_i = \sum_{j=1}^{M} c_{ji} \tag{2.5.20}$$

$$\overline{Y}_i = \frac{1}{M_i} \sum_{j=1}^{M} c_{ji} Y_j \tag{2.5.21}$$

$$S_i = \frac{1}{M_i} \sum_{j=1}^{M} c_{ji} (Y_j - \bar{Y}_i)(Y_j - \bar{Y}_i)^T, \tag{2.5.22}$$

for $1 \leq i \leq N$. By our choice of conjugate prior densities, we find the update equations for each of the model parameters are given by

$$q^*(\mu_i, \Lambda_i) = \mathcal{N}(\mu_i \mid m_i, (\beta_i \Lambda_i)^{-1}) \mathcal{W}(\Lambda_i \mid W_i, \nu_i) \tag{2.5.23}$$

by the update formulas

$$\beta_i = \beta_0 + M_i \tag{2.5.24}$$

$$m_i = \frac{1}{\beta_i}(\beta_0 m_0 + M_i \overline{Y}_i) \tag{2.5.25}$$

$$W_i^{-1} = W_0^{-1} + M_i S_i + \frac{\beta_0 M_i}{\beta_0 + M_i} (\overline{Y}_i - m_0)(\overline{Y}_i - m_0)^T \tag{2.5.26}$$

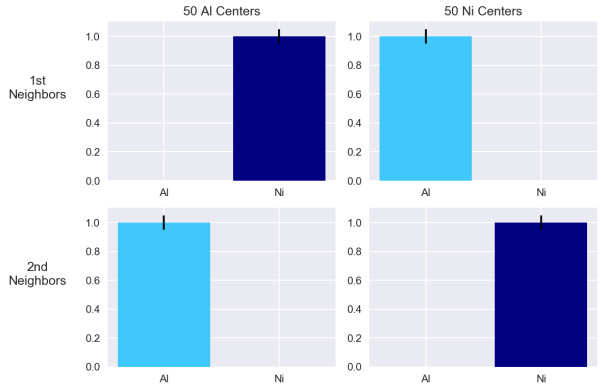
$$\nu_i = \nu_0 + M_i, \tag{2.5.27}$$

where the pertinent parameters for the priors are as defined in Equation (2.5.12) and Equation (2.5.14). Each of these update equations depends on the correspondence matrix C. Recalling Equation (2.5.19), we must compute the normalizing constant for the ρ_{mn}, and the individual terms are given below in Equations (2.5.28)–(2.5.30).

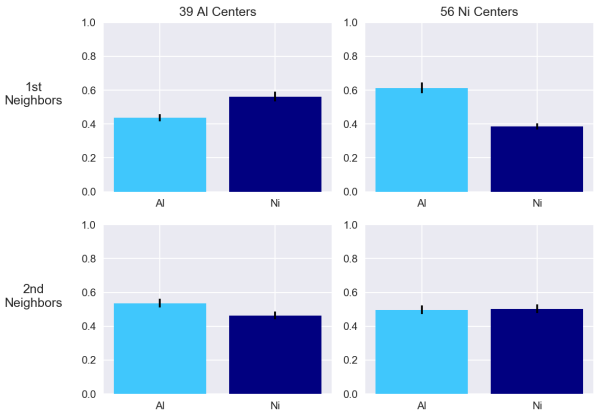
$$\mathbb{E}[\|y_i - \mu_n\|^2_{\Lambda_n}] = \frac{d}{\beta_n} + \nu_n (y_i - \mu_n)^T W_n (y_i - \mu_n) \tag{2.5.28}$$

$$\mathbb{E}[\log |\Lambda_n|] = \sum_{i=1}^{d} \left(\frac{\nu_n + 1 - i}{2} \right) + d \log(2) + \log |W_n| \tag{2.5.29}$$

$$\mathbb{E}[\log(\omega_n)] = \psi(\alpha_n) - \psi(\hat{\alpha}). \tag{2.5.30}$$



(a) Complete and noiseless data, we are able to recover the chemical ordering.



(b) Data with 67% missing and $\mathcal{N}(0,1)$ added to each point, we cannot recover the chemical ordering.

Figure 2.6: Variational inference applied to complete, noiseless data (a) and noisy, sparse data (b), showing how the process is not able to recover the chemical ordering, i.e., the aluminium center has 8 nickel first neighbors and 6 aluminium second neighbors, and vise versa when nickel is the center atom.

Here $\psi(\cdot)$ is the digamma function, $\hat{\alpha} = \sum_{k=1} \alpha_k$, and Equations (2.5.29)–(2.5.30) follow from properties of the Wishart and Dirichlet distributions respectively [53, 59].

This model is insufficient for our purpose here for the APT data, as it lacks any elemental information about the points in the dataset. indeed, applying the process to our data yields a correspondence matrix and transformation between the reference and observation sets, it does not take into account the labels (elemental type) of each point in the observation set. The resulting neighbor analysis is shown in Figure 2.6.

The synthetic data for the numerical experiment in Figure 2.6 was created with interlocking BCC crystals that were exclusively aluminium or nickel in their composition. The result is a chemically ordered data set, where each neighborhood with aluminium at is center only has nickel as first neighbors and aluminium as second neighbors. A similar ordering holds when nickel is at the neighborhood's center. Consequently, we can see that this ordering is well-preserved in Fig. 2.6a, but is lost due to the noise and sparsity in Fig. 2.6b. In Section 4.4, we will show how we can extend this framework to include elemental type and make inferences about short-range chemical ordering in HEAs.

Chapter 3

Known Reference

3.1 Introduction

Here we present our Bayesian formulation of the point set registration problem. This problem is one of the most basic in computer vision tasks that arise from many different applications, e.g., object recognition, medical imaging, and lidar applications [60, 61, 62, 63]. Fundamentally, the problem involves finding the best spatial alignment and point correspondence between two finite point clouds when the point correspondences are not known *a priori*. If the transformation describing the displacement between the tow point clouds is taken to be the rigid transformations, then each of the individual problems is easily solved by itself, and naive methods simply alternate the solution of each individually until convergence. If the transformation is non-rigid, then the problem quickly grows in complexity as the associated transformations are typically non-linear, and consequently, are difficult to model accurately.

One of the most frequently used point set registration algorithms is the iterative closest point method, which alternates between identifying the optimal transformation, i.e., for a given correspondence, it minimizes the mean-squared distance between point sets, and then identifies the closest points between the point sets to define a correspondence between them [60]. If the transformation is rigid, and the point sets are of equal cardinality, then both problems are uniquely solvable. If instead we replace the naive closest point strategy with the assignment problem, so that any two observed points correspond to two different reference points, then again the problem can solved with a linear program [64]. However, when these two solvable problems are combined into one, the resulting problem is non-convex [64, 65]. We may see this by examining the Hessian of the mean-squared error with respect to the transformation parameter. It is a third order tensor, which is not symmetric-positive definite [66], and hence, the objective function is not convex. Thus the point set registration problem no longer admits a unique solution, even for the case of rigid transformations that

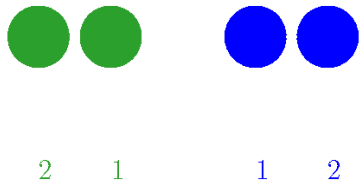


Figure 3.1: Setup for incorrect registration; alternating assignment and ℓ^2 minimization

we consider here. The same strategy has been proposed with more general non-rigid transformations [67], where identification of the optimal transformation is no longer analytically solvable. The method in [68] minimizes an upper bound on their objective function, and is thus also susceptible to getting stuck in a local basin of attraction. We instead take a Bayesian viewpoint and by sampling consistently from the posterior density, may recover the *maximum a posterior* estimator.

3.1.1 Bayesian Point Set Registration

An alloy consists of a large collection of atoms, henceforth "points", which we assume are transformed instances of points on a reference lattice structure, denoted by $X = (X_1, \ldots, X_N)^T, X_i \in \mathbb{R}^d$ for $1 \leq i \leq N$. The tomographic observation of this configuration is missing some percentage of the points and is subject to noise, which is assumed additive and Gaussian. The sample consists of a single point and its M nearest neighbors, where M is of the order 10. If $p \in [0, 1]$ is the percent observed, i.e. $p = 1$ means all points are observed and $p = 0$ means no points are observed, then the reference point set will be comprised of $N = \lceil M/p \rceil$ points. We write the matrix representation of the noisy data point as $Y = (Y_1, \ldots, Y_M)^T, Y_i \in \mathbb{R}^d$, for $1 \leq i \leq M$.

The observed points have labels, but the reference points do not. We seek to register these noisy and sparse point sets, onto the reference point set. The ultimate goal is to identify the ordering of the labels of the points (types of atoms) in a configuration. We will find the best assignment and transformation, in a mean-squared error sense, between the observed point set and the reference point set. Having completed the registration process for all observations in the configuration, we may then construct a three dimensional distribution of labeled points around each reference point, and the distribution of atomic composition is readily obtained.

As a motivating example, we will show how alternating between finding correspondences and minimizing distances can lead to an incorrect registration. Consider now the setup in Figure 3.1. If we correspond closest points first, then all three green points would be assigned to the blue '1'. Then, identifying the single rigid

transformation to minimize the distances between all three green and the blue '1' would yield a local minimum, with no correct assignments. If we consider instead *assignments*, so that no two observation points can correspond to the same reference point, then again it is easy to see two equivalent solutions with the eye. The first is a pure translation, and the second can be obtained for example by one of two equivalent rotations around the mid-point between '1's, by π or $-\pi$. Note that in reality the reference labels are unknown, so both are equivalent for us. Furthermore, this simple example motivates our variational treatment of the registration problem in Chapter 4, where the labels associated with the points greatly impacts the quality of a solution.

Here it is clear what the solutions are, but as the problem grows in scale, the answer is not always so clear. This simple illustration of degenerate (equal energy) multi-modality of the registration objective function arises from physical symmetry of the reference point-set. This is an important consideration for our reference point sets, which arise as a unit cell of a lattice, hence with appropriate symmetry. We will never be able to know the registration beyond these symmetries, but this will nonetheless not be the cause of concern, as symmetric solutions will be considered equivalent. The troublesome multi-modality arises in the presence of noisy and partially observed point sets, where there may be local minima with higher energy than the global minima.

The multi-modality of the combined problem, in addition to the limited information in the noisy and sparse observations, motivates the need for a global probabilistic notion of a solution for this problem. In the following sections we show that the problem lends itself naturally to a flexible Bayesian formulation which circumvents the intrinsic shortcomings of deterministic optimization approaches for non-convex problems.

3.2 Bayesian Formulation

We seek to compute the registration between the observation set and reference set. We are concerned primarily with rigid transformations of the form

$$\mathcal{T}(X;\theta) = XR_\theta + t_\theta, \tag{3.2.1}$$

where $R_\theta \in \mathbb{R}^{d\times d}$ is a rotation and $t_\theta \in \mathbb{R}^d$ is a translation vector.

Write $[\mathbb{T}(X;\theta)]_{ki} = \mathcal{T}_k(X_i)$ for $1 \le i \le N$, $1 \le k \le d$, and where X_i is the i^{th} row of X. Now let $\Gamma \in \mathbb{R}^{d\times M}$ with entries $\Gamma_{ij} \sim N(0,\gamma^2)$, and assume the following statistical model

$$Y = C\mathbb{T}(X;\theta) + \xi, \tag{3.2.2}$$

for ξ, θ, and C independent.

The matrix of correspondences $C \in \{0,1\}^{M\times N}$, is such that $\sum_{k=1}^{N} C_{jk} = 1, 1 \leq j \leq M$, and each observation point corresponds to only one reference point. So if X_i matches Y_j then $C_{ji} = 1$, otherwise, $C_{ji} = 0$. We let C be endowed with a prior, $\pi_0(C_{ji} = 1) = \pi_{ji}$ for $1 \leq i \leq N$ and $1 \leq j \leq M$. Furthermore, assume a prior on the transformation parameter θ given by $\pi_0(\theta)$. The posterior distribution then takes the form

$$\pi(C, \theta \mid X, Y) \propto \mathcal{L}(Y \mid X, C, \theta)\pi_0(C)\pi_0(\theta), \tag{3.2.3}$$

where $\mathcal{L}$ is the likelihood function associated with Equation (3.2.1).

For a given $\tilde{\theta}$, an estimate $\hat{C}$ can be constructed *a posteriori* by letting $\hat{C}_{j,i^*(j)} = 1$ for $j = 1, \ldots, M$ and zero otherwise, where

$$i^*(j) = \underset{1\leq i\leq N}{\operatorname{argmin}} |Y_j - \mathcal{T}(X_i; \tilde{\theta})|^2 . \tag{3.2.4}$$

For example, $\tilde{\theta}$ may be taken as the maximum a posteriori (MAP) estimator or the mean. We note that $\hat{C}$ can be constructed either with a closest point approach, or via assignment to avoid multiple registered points assigned to the same reference.

Lastly, we assume the j^{th} observation only depends on the j^{th} row of the correspondence matrix, and so Y_i, Y_j are conditionally independent with respect to the matrix C for $i \neq j$. This does not exclude the case where multiple observation points are assigned to the same reference point, but as mentioned above such scenario should have zero probability.

To that end, instead of considering the full joint posterior in Equation (3.2.3) we will focus on the marginal of the transformation

$$\pi(\theta \mid X, Y) \propto \mathcal{L}(Y \mid X, \theta)\pi_0(\theta). \tag{3.2.5}$$

Let C_j denote the j^{th} row of C. Since C_j is completely determined by the single index i at which it takes the value 1, the marginal likelihood takes the form

$$\begin{aligned}
\sum_{C} p(Y_j \mid X, \theta, C)\pi_0(C) &= \sum_{i=1}^{N} p(Y_j \mid X, \theta, C_{ji} = 1)\pi_0(C_{ji} = 1) \\
&= \sum_{i=1}^{N} \pi_{ji} p(Y_j \mid X, \theta, C_{ij} = 1) \\
&\propto \pi_{ji} \exp\left\{-\frac{1}{2\gamma^2}|Y_j - \mathcal{T}(X_i; \theta)|^2\right\} .
\end{aligned} \tag{3.2.6}$$

The above marginal together with the conditional independence assumption allows us to construct the likelihood function of the marginal posterior, Equation (3.2.5), as follows

$$\begin{aligned}\mathcal{L}(Y \mid X, \theta) &= \prod_{j=1}^{M} p(Y_j \mid X, \theta) \\ &\propto \prod_{j=1}^{M} \sum_{i=1}^{N} \pi_{ji} \exp\left\{-\frac{1}{2\gamma^2}|Y_j - \mathcal{T}(X_i;\theta)|^2\right\} . \end{aligned} \tag{3.2.7}$$

Thus the posterior in question is

$$\begin{aligned}\pi(\theta \mid X, Y) &\propto \mathcal{L}(Y \mid X, \theta)\pi_0(\theta) \\ &= \prod_{j=1}^{M} \sum_{i=1}^{N} \pi_{ji} \exp\left\{-\frac{1}{2\gamma^2}|Y_j - \mathcal{T}(X_i;\theta)|^2\right\} \pi_0(\theta) . \end{aligned} \tag{3.2.8}$$

At its heart, point set registration is an optimization problem. Consider a prior on θ such that $\pi_0(\theta) \propto \exp(-\lambda R(\theta))$, where $\lambda > 0$. Then we have the following objective function

$$E(\theta) = -\sum_{j=1}^{M} \log \sum_{i=1}^{N} \pi_{ji} \exp\left\{-\frac{1}{2\gamma^2}|Y_j - \mathcal{T}(X_i;\theta)|^2\right\} + \lambda R(\theta) . \tag{3.2.9}$$

The minimizer, θ^*, of the above, Equation (3.2.9) is also the maximizer of a posteriori probability under Equation (3.2.8). This can also be viewed as maximum likelihood estimation regularized by $\lambda R(\theta)$. By sampling consistently from the posterior, we may estimate quantities of interest, such as moments, together with quantified uncertainty. Additionally, we may recover other point estimators, such as local and global modes.

3.3 Numerical Experiments

To illustrate our approach, we consider numerical experiments on synthetic materials datasets, with varying levels of noise and percentage of observed data. We focus our attention to rigid transformations of the form Equation (3.2.1).

For all examples here, the M observation points are simulated as $Y_i \sim N(R_\varphi X_{j(i)} + t, \gamma^2 I_d)$, for a rotation matrix R_φ parameterized by φ, and some t and γ. So, $\theta = (\varphi, t)$. To simulate the unknown correspondence between the reference and observation points, for each $i = 1, \ldots, M$, the corresponding index $i(j) \in [1, \ldots, N]$ is chosen randomly and without replacement. Recall that we define percentage of

observed points here as $p = \frac{M}{N} \in [0,1]$.We tested various percentages of observed data and noise γ on the observation set, then computed the mean square error (MSE), given by Equation (3.3.1), between the reference points and the registered observed points,

$$\mathcal{E}(\theta) = \frac{1}{M} \sum_{j=1}^{M} \min_{X \in \mathbf{X}} |R_{\varphi}^{T}(Y_j - t) - X_{i(j)}|^2 . \tag{3.3.1}$$

3.3.1 Sensitivity Analysis

The datasets from APT experiments are perturbed by additive noise on each of the points. The variance of this additive noise is not known in general, and so in practice it should be taken as a hyper-parameter, endowed with a hyper-prior, and inferred or optimized. It is known that the size of the displacement on the order of several Å (Angstroms), so that provides a good basis for choice of hyper prior. In order to simulate this uncertainty in our experiments, we incorporated additive noise in the form of a truncated Gaussian, to keep all the mass within several Å . The experiments consider a range of variances in order to measure the impact of noise on our registration process.

In our initial experiments with synthetic data, we have chosen percentages of observed data and additive noise similar to what materials scientist experimentalists have reported in their APT datasets. The percent observed of these experimental datasets is approximately 33%. The added noise of these APT datasets is harder to quantify. Empirically, we expect the noise to be Gaussian in form, truncated to be within 1-3 Å. The standard deviation of the added noise is less well-known, so we will work with different values to asses the method's performance. With respect to the size of the cell, a displacement of 3Å is significant. Consider the cell representing the hidden truth in Figure 3.2. The distance between the front left and right corners is on the scale of 3Å. Consequently a standard deviation of 0.5 for the additive noise represents a significant displacement of the atoms.

As a visual example, the images in Figure 3.2 are our synthetic test data used to simulate the noise and missing data from the APT datasets. The leftmost image in Figure 3.2 is the hidden truth we seek to uncover. The middle image is the first with noise added to the atom positions. Lastly, in the right-most image we have 'ghosted' some atoms, by coloring them grey, to give a better visual representation of the missing data. In these representations of HEAs, a color different from grey denotes a distinct type of atom. What we seek is to infer the chemical ordering and atomic structure of the left image, from transformed versions of the right, where $\gamma = 0.5$.

For our initial numerical experiments with simulated APT data, we choose a single reference and observation, and consider two different percentages of observed data, 75% and 45%. For both levels of

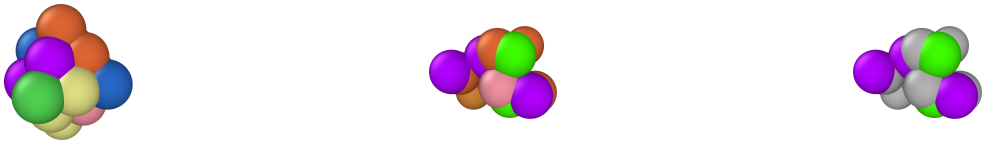


Figure 3.2: Example APT data: Left: Hidden truth, Center: Noise added, Right: Missing atoms colored grey.

observations in the data, we looked at results with three different levels of added noise on the atomic positions: no noise, and Gaussian noise with standard deviation of 0.25 and 0.5. The MSE of the processes are shown in Table 3.1. We initially observe the method is able, within an appreciably small tolerance, find the exact parameter θ in the case of no noise, with both percentages of observed data. In the other cases, as expected, the error scales with the noise. This follows from our model, as we are considering a rigid transformation between the observation and reference, which is a volume preserving transformation. If the exact transformation is used with an infinite number of points, then the RMSE (square root of Equation (3.3.1)) is γ.

Now we make the simplifying assumption that the entire configuration corresponds to the same reference, and each observation in the configuration corresponds to the same transformation applied to the reference, with i.i.d. noise added to it. This enables us to approximate the mean and variance of Equation (3.3.1) over these observation realizations, i.e. we obtain a collection $\{\mathcal{E}^l(\theta^l)\}_{l=1}^L$ of errors, where $\mathcal{E}^l(\theta^l)$ is the MSE corresponding to replacing $\mathbf{Y}^l$ and its estimated registration parameters θ^l into Equation (3.3.1), where L is the total number of completed registrations. The statistics of this collection of values provide robust estimates of the expected error for a single such registration, and the variance we can expect over realizations of the observational noise. In other words

$$\mathbb{E}^L\mathcal{E}(\theta) := \frac{1}{L}\sum_{l=1}^{L}\mathcal{E}^l(\theta^l) \text{ and } \operatorname{Var}^L\mathcal{E}(\theta) := \frac{1}{L}\sum_{l=1}^{L}(\mathcal{E}^l(\theta^l) - \mathbb{E}^L\mathcal{E}(\theta))^2\,. \tag{3.3.2}$$

We have confidence intervals as well, corresponding to a central limit theorem approximation based on these L samples.

In Figs. 3.3a–3.3d we computed the registration for $L = 125$ i.i.d. observation sets corresponding to the same reference, for each combination of noise and percent observed data. We then averaged all 125 registration errors for a fixed noise/percent observed combination, as in Equation (3.3.2), and compared the values. What we observe in Figs. 3.3a–3.3d is the registration error scaling with the noise, which is expected. What is interesting to note here is that the registration error is essentially constant with respect

Standard Deviation	Percent Observed	Registration Error
0.0	75%	3.493686e−11
0.0	45%	4.400718e−11
0.25	75%	0.170252
0.25	45%	0.122155
0.5	75%	0.344568
0.5	45%	0.364317

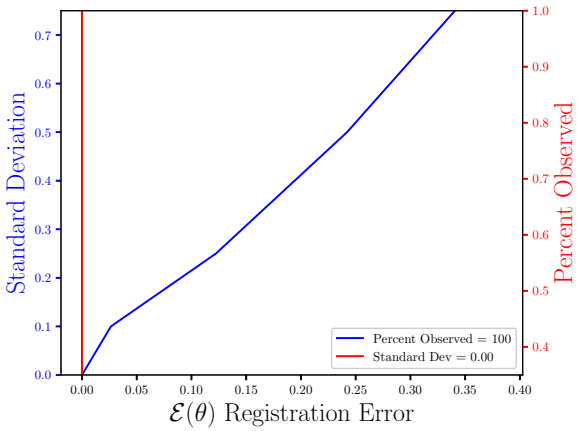
Table 3.1: $\mathcal{E}(\theta)$ Registration Errors

to the percentage of observed data, for a fixed standard deviation of the noise. More information will lead to a lower variance in the posterior on the transformation θ, following from standard statistical intuition. However, the important point to note is that, as mentioned above, for exact transformation, and infinite points, Equation (3.3.1) will equal γ^2. So, for sufficiently accurate transformation, one can expect a sample approximation thereof. Sufficient accuracy is found here with very few observed points, which is reasonable considering that in the zero noise case 2 points is sufficient to fit the 6 parameters exactly.

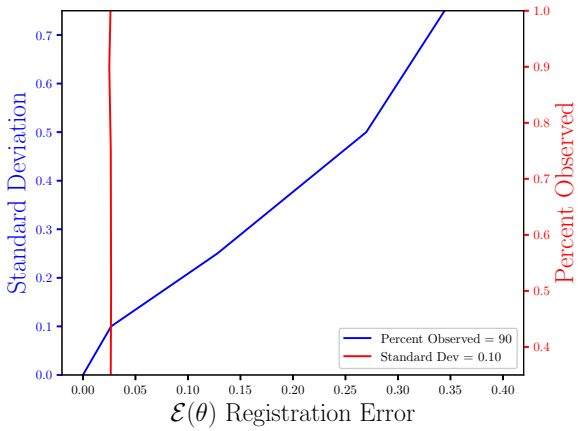
The MSE registration errors shown in Figs. 3.3a–3.3d, show the error remains essentially constant with respect to the percent observed. Consequently, if we consider only Fig. 3.3b, we observe that the blue and red lines intersect, when the blue has a standard deviation of 0.1, and the associated MSE is approximately 0.05. This same error estimate holds for all tested percentages of observed data having a standard deviation of 0.1. Similar results hold for other combinations of noise and percent observed, when the noise is fixed.

Furthermore, the results shown in Figs. 3.3a–3.3d are independent of the algorithm, as the plots in Figs. 3.3e–3.3f show. For the latter, we ran a similar experiment with 125 i.i.d. observation sets, but to compute the registration, we used the MALA [40] sampling algorimth detailed in Section 2.3.2, as opposed to HMC in Figs. 3.3a–3.3d. Both algorithms solve the same problem and use information from the gradient of the log density. In the plots shown in Figs. 3.3a–3.3d, we see the same constant error with respect to the percent observed and the error increasing with the noise, for a fixed percent observed. The MSE also appears to be proportional to γ^2, which is expected, until some saturation threshold of $\gamma \geq 0.5$ or so. This can be understood as a threshold beyond which the observed points will tend to get assigned to the wrong reference point.

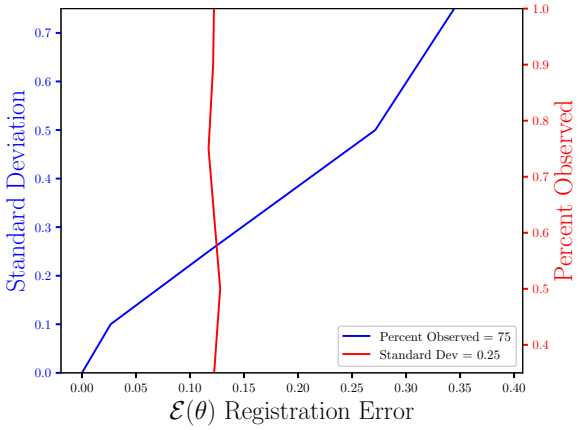
To examine the contours of our posterior described by Equation (3.2.8), we drew 10^5 samples from the density using the HMC methodology described previously. For this simulation we set the noise to have standard deviation of 0.25 and the percent observed was 35%, similar values to what we expect from



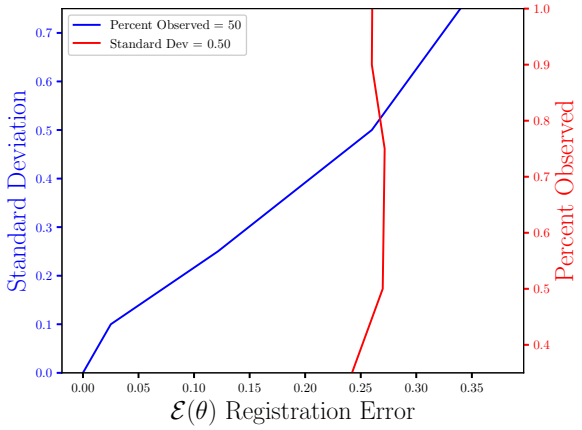
(a) Blue: Full data, Red: Noiseless data



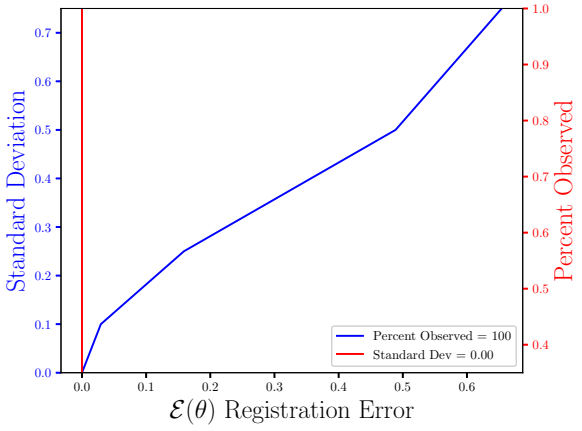
(b) Blue: 90% Observed, Red: $\gamma = 0.1$



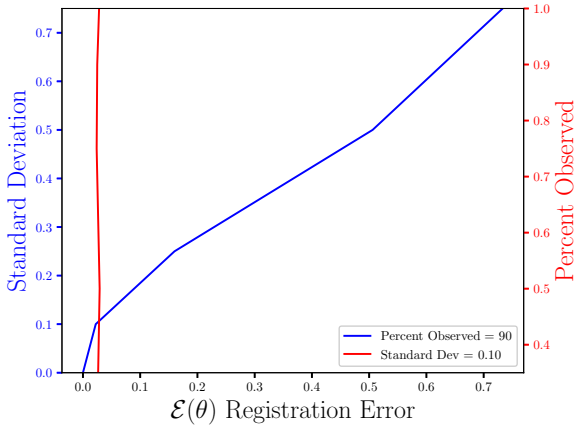
(c) Blue: 75% Observed, Red: $\gamma = 0.25$



(d) Blue: 50% Observed, Red: $\gamma = 0.5$



(e) Blue: Full data, Red: Noiseless data (MALA)



(f) Blue: 90% Observed, Red: $\gamma = 0.1$ (MALA)

Figure 3.3: Error as plotted against various combinations of noise and sparsity.

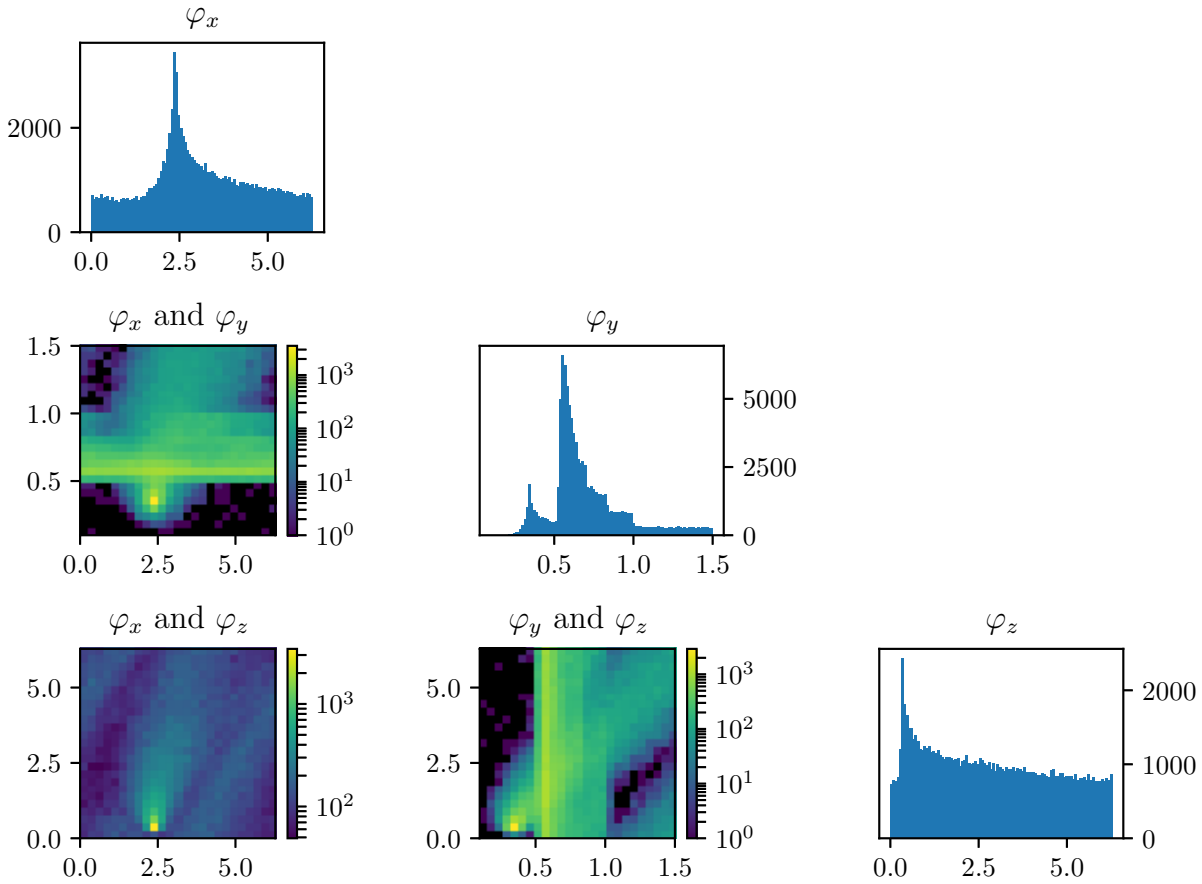


Figure 3.4: Histograms of φ parameters, 100000 samples, $\gamma = 0.25$, Observed = 35%

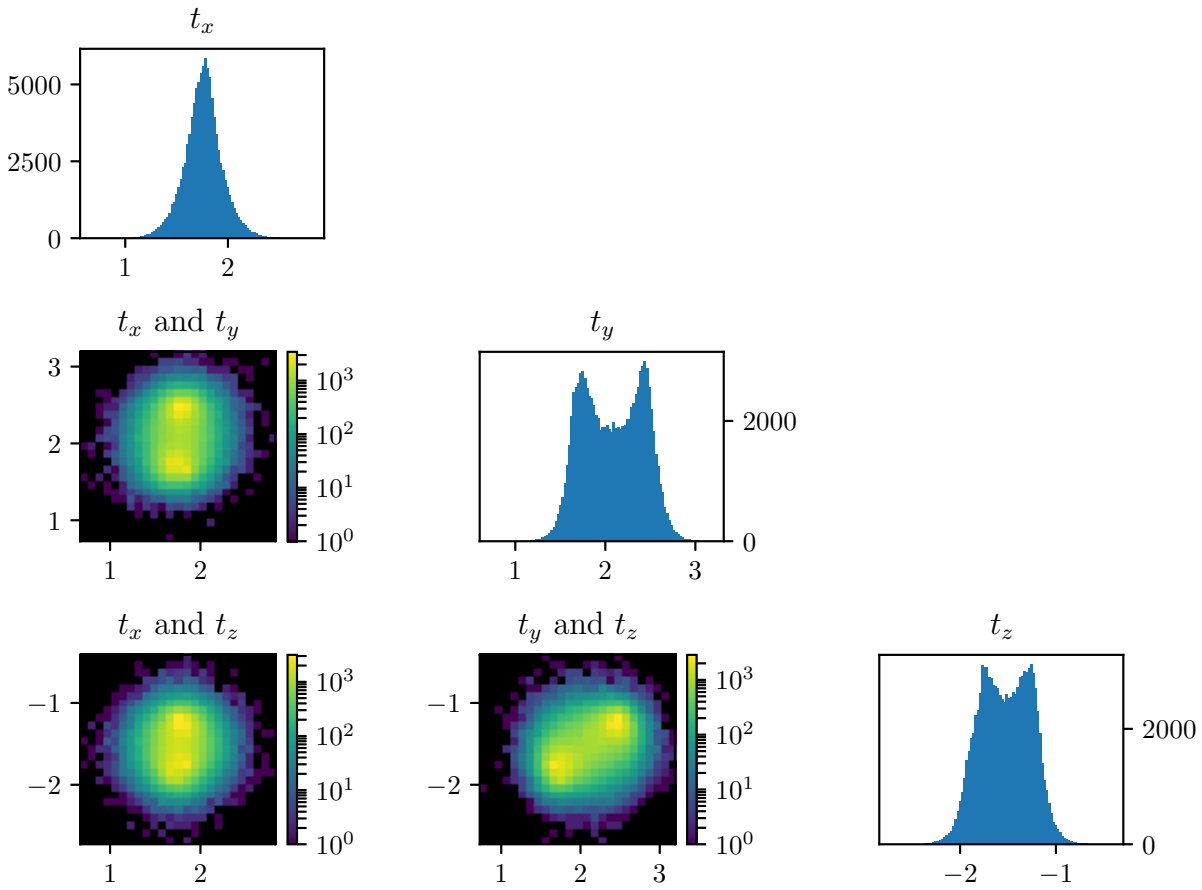


Figure 3.5: Histograms of θ parameters, 100000 samples, $\gamma = 0.25$, Observed = 35%

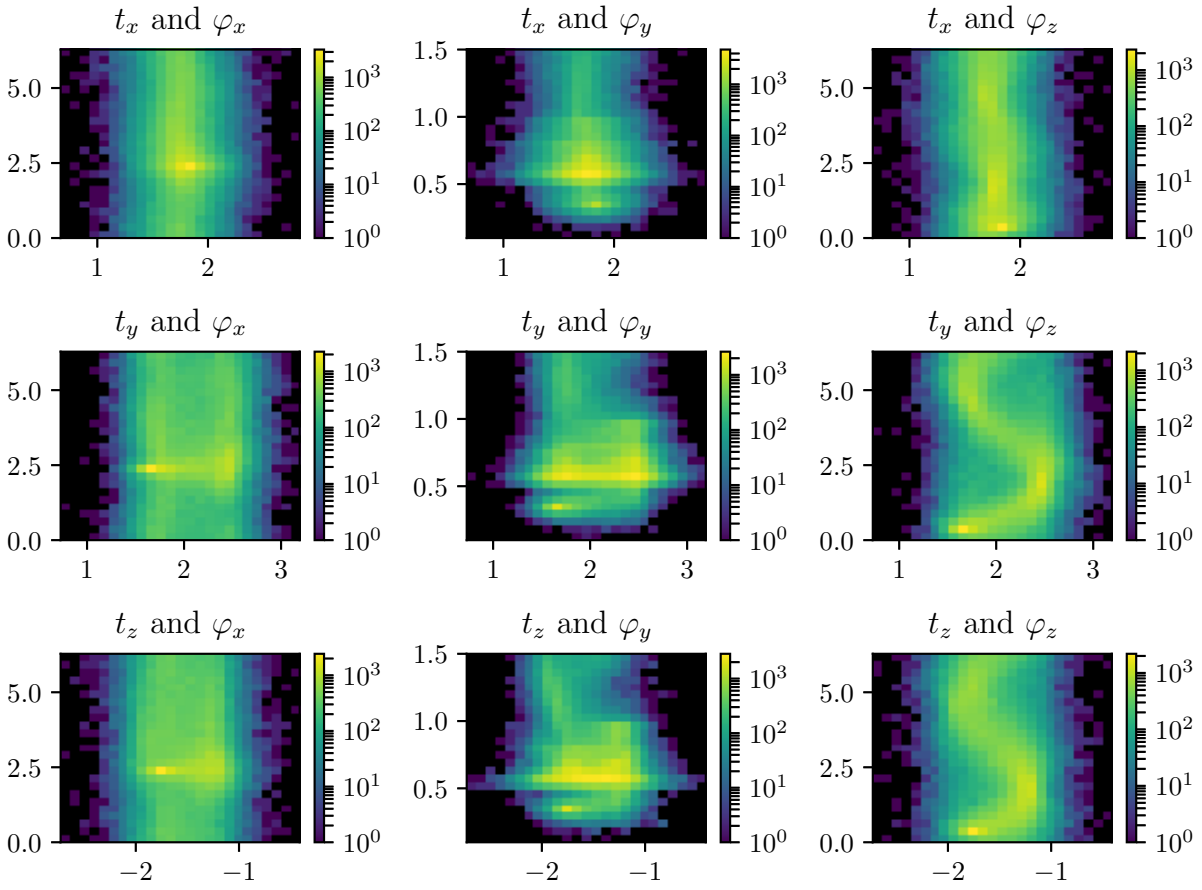


Figure 3.6: Histograms of θ parameters, 100000 samples, $\gamma = 0.25$, Observed = 35%

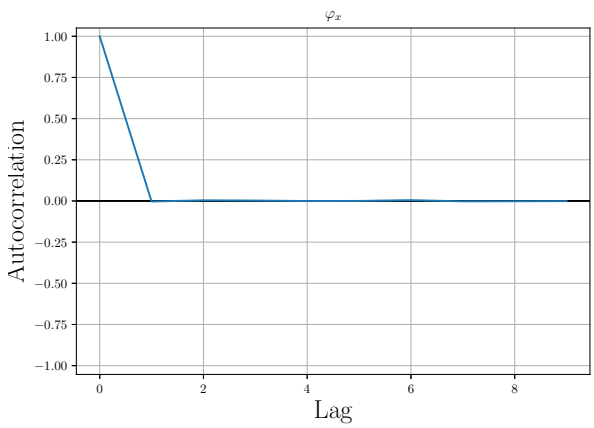
Standard Deviation	Percent Observed	Error
0.25	75%	0.049096
0.5	75%	0.079345
0.25	45%	0.074600
0.5	45%	0.119786

Table 3.2: Errors for 125 Completed Registrations

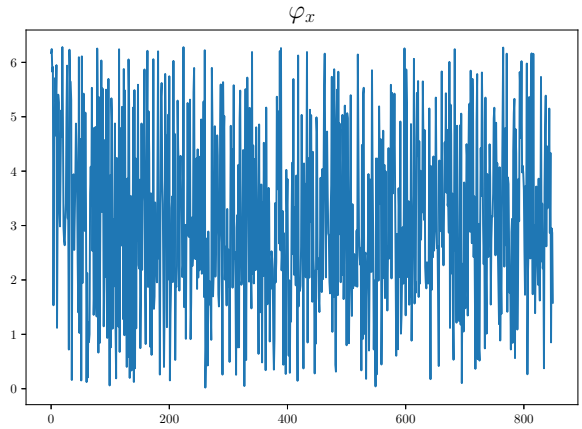
real APT datasets. The rotation matrix R is constructed via Euler angles denoted: $\varphi_x, \varphi_y, \varphi_z$, where $\varphi_x \in [0, 2\pi), \varphi_y \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and $\varphi_z \in [0, 2\pi)$. These parameters are especially important to making the correct atomic identification, which is crucial to the success of our method.

In Figures 3.4–3.6, we present marginal single variable histograms and all combinations of marginal two-variable joint histograms for the individual components of θ. We observe multiple modes in a number of the marginals. In Figs. 3.7a–3.7f we present autocorrelation and trace plots for the rotation parameters from the same instance of the HMC algorithm as presented in the histograms above in Figures 3.4–3.6. We focus specifically on the rotation angles, to ensure efficient mixing of the Markov chain as these have thus far been more difficult for the algorithm to optimize. We see the chain is mixing well with respect to these parameters and appears not to become stuck in local basins of attraction.

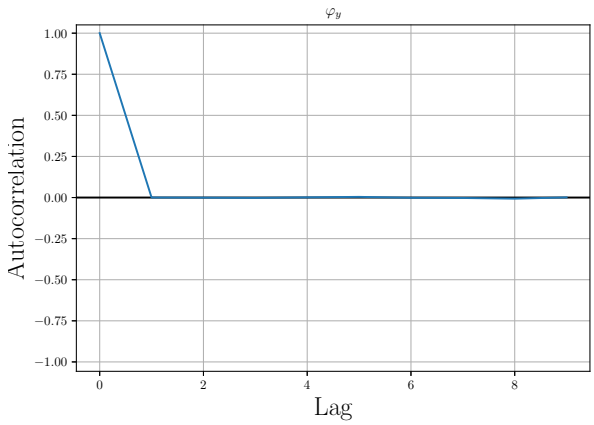
Additionally, we consider the following. Define null sets $A_1, \ldots, A_N$. For each $j = 1, \ldots, M$ and $l = 1, \ldots, L$, let $i^*(j,l) := \text{argmin}_{i \in \{1,\ldots,N\}} |R_{\varphi^l}^T (Y_j^l - t^l) - X_i|^2$, and increment $A_{i^*(j,l)} = A_{i^*(j,l)} \cup Y_j^l$. This provides a distribution of registered points for each index i, A_i, from which we estimate various statistics such as mean and variance. However, note that the cardinality varies between $|A_i| \in \{0, \ldots, L\}$. We are only be concerned with statistics around reference points i such that $|A_i| > L/10$ or so, assuming that the other reference points correspond to outliers which were registered to by accident. Around each of these $N' \leq N$ reference points X_i, we have a distribution of some $K \leq L$ registered points. We then computed the mean of these K points, denoted by $\bar{X}_i$ and finally we compute the MSE $\frac{1}{N'} \sum_{i=1}^{N'} |X_i - \bar{X}_i|^2$. The RMSE is reported in Section 3.3.1. Here we note that a lower percentage observed p is correlated with a larger error. Coupling correct inferences about spatial alignment with an ability to find distributions of atoms around each lattice point is a transformative tool for understanding high entropy alloys.



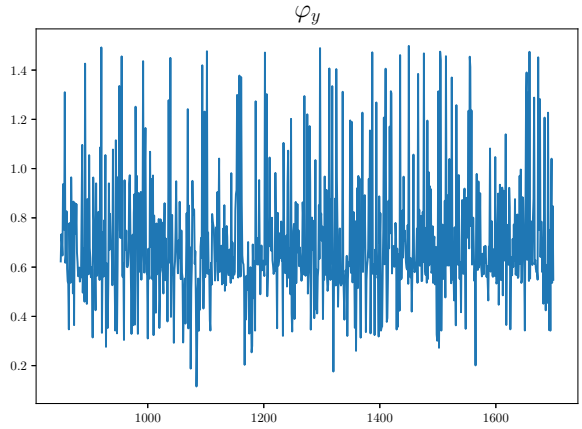
(a) Autocorrelation plot, φ_x



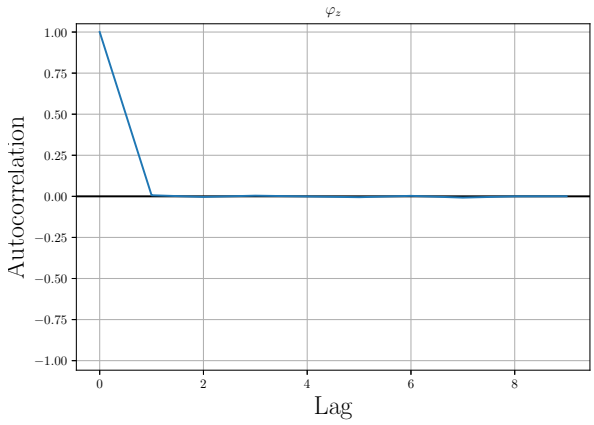
(b) Trace plot, φ_x



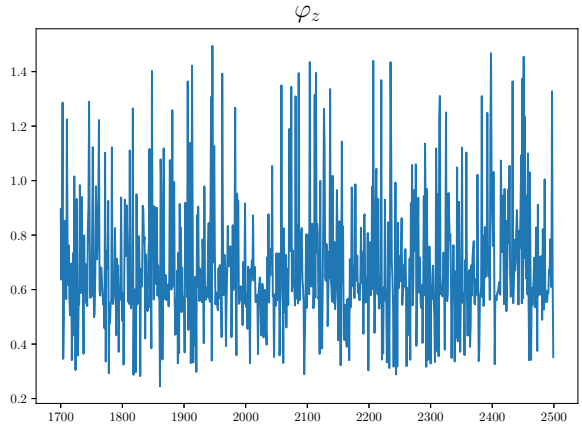
(c) Autocorrelation plot, φ_y



(d) Trace plot, φ_y



(e) Autocorrelation plot, φ_z



(f) Trace plot, φ_z

Figure 3.7: Autocorrelation and trace plots for φ for our MCMC Bayesian registration method.

Chapter 4

Unknown Reference

We now shift or focus to the case where we do not know the lattice structure of a material beforehand. We now seek to discover the crystal lattice, and neighbor relationships between the atoms in an atomic neighborhood. As posed here, the problem has two components: *i.* classification and *ii.* labeled point set registration. For the former, we present a topologically informed classification scheme, employing a distance on the space of persistence diagrams that leverages both differences in topology and cardinality of the persistence diagrams. We also show a stability property for this distance. Having inferred the correct lattice structure, we use this to inform our variational approach to the registration problem.

The version of the point set registration problem that we discuss, and present an algorithmic solution to, is different from all existing settings. The points we consider have labels, specifically atomic type, whereas all other existing algorithms make no distinction between points in the sets they consider. While this increases the computational complexity of the algorithm, we may ameliorate these considerations by devising a MCMC sampling scheme to find the optimal, i.e., lowest, energy configuration of atoms in a neighborhood.

4.1 Materials Fingerprinting

A crucial first step in understanding properties of a crystalline material is determining its crystal structure. For highly disordered metallic alloys, such as high-entropy alloys, atom probe tomography gives a snapshot of the local atomic environment. APT has the potential to quantify distributions of lattice parameters and atomic composition within HEAs. Indeed, analysis of HEAs are amenable to the APT experiment as the process is able to recover elemental type in addition to approximating the lattice sites in a material where the atoms sit.

An experimental process that determines the position, identity of each atom, and structure of a material is currently nonexistent [21, 69]. Indeed, unambiguous quantification of different lattice parameters and unit-cell compositions has not previously been reported due to data quality issues inherent to APT [24, 25]. While these experiments are able to discern elemental types at a high resolution, the process has two drawbacks, experimental noise and missing data. Approximately 65% of the atoms in a sample are not registered in a typical experiment, and those atoms that are captured have their spatial coordinates corrupted by experimental noise. As noted by [24] and [25], APT has a spatial resolution approximately the length of the unit cell we consider, as seen in Figure 4.2. Hence the process is unable to see the finer details of a material, making the determination of a lattice structure a challenging problem. Existing algorithms for detecting the crystal structure [70, 71, 72, 73, 69, 74] are not able to establish the crystal lattice of an APT dataset, as they rely on symmetry arguments. Consequently, the field of atom probe crystallography, i.e., determining the crystal structure from APT data, has emerged in recent years [26] and [69]. These algorithms rely on knowing the global lattice structure *a priori* and aim to determine local small-scale structures within a larger sample. For some materials this information is readily known, for others, such as HEAs, the global structure is unknown and must be inferred. A recent work by [75] proposes a machine-learning approach to classifying crystal structures of a noisy and sparse materials dataset, without knowing the global structure *a priori*. The authors employ a convolutional neural network for classifying the crystal structure by looking at a diffraction image, a computer-generated diffraction pattern. The authors suggest their method could be used to determine the crystal structure of APT data or other noisy and sparse data from materials science. However, the synthetic data considered in [75] is not a realistic representation of experimental APT data, where about 65% of the data is missing [1] and is corrupted by more observational noise [25] than is considered in [75]. Most importantly, their synthetic data is either sparse or noisy, not a combination of both. We consider a combination of noise and sparsity, such as is the case in real APT data.

The field of atom probe crystallography has emerged in recent years [26, 69], and existing methodologies in this area seek to discover local structures when the global structure is known *a priori*. In the case of HEAs, the global lattice structure is unknown and must be discovered through further analysis. Indeed, drawing correct conclusions about the material's crystal structure is virtually impossible from APT analysis using current techniques [25]. Even in the case of noiseless and complete data, symmetry-based algorithms for determining the crystal structure may fail due to the non-symmetric nature of HEA crystal structures. Their lattice structures become distorted due to the many types of differently-sized atoms that are randomly distributed throughout the material with equal probability. The net effect of neighboring atoms with different radii in a lattice structure is to deform the symmetry of the cubic lattice, Figure 4.1(a) shows a symmetric

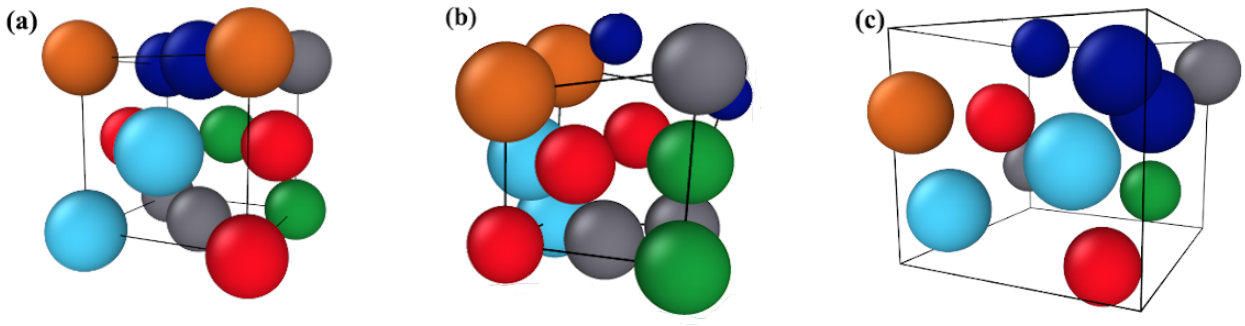
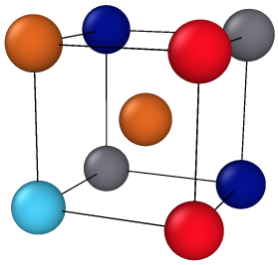


Figure 4.1: An example of face-centered cubic lattices showing the similarities and differences between an ideal, noiseless lattice structure in (a) and the data retrieved from an APT experiment in (c). Atoms in (a) sit precisely at their lattice points and each side of the cube is equal in length. The lattice in (b) shows the distorted lattice structure of an FCC HEA. The atomic positions no longer form a symmetric lattice and the sides are unequal in length. These local distortions are due to different sized atoms sitting at lattice positions and break the symmetry of the idealized FCC lattice in (a). These local lattice distortions make identification of the crystal structure by existing symmetry-based algorithms a challenging problem. In spite of these distortions, the unit cell retains the essential characteristics of an FCC cell: (*i*) number of atoms in the unit cell and (*ii*) atoms on the cube's faces and hollow in the center. We also note the different sized cubes in each one of the cells due to the random distribution of atoms throughout the material. The cell in (c) indicates the sparsity and atomic displacements due to the resolution of APT. Importantly, there are fewer atoms in (c) than in the idealized representation (a).

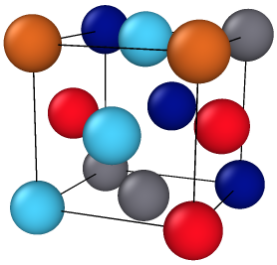
FCC lattice, into one that defies easy characterization; see Figure 4.1(b) for an example of the distorted lattice of an HEA. As seen through the lens of APT in Figure 4.1(c), the distorted FCC lattice structure of an HEA looks unrelated to its idealized version in Figure 4.1(a). Consequently, this deformation of the local crystal structure makes any determination of the lattice a challenging problem for any symmetry-based algorithm, such as [71, 72, 73].

We provide a machine learning approach to classify the crystal structure of a noisy and sparse materials dataset. Specifically, we consider materials that are either body-centered cubic (BCC) or face-centered cubic (FCC), as these lattice structures are the essential building blocks of HEAs [9] and have fundamental differences that set them apart in the case of noise-free, complete materials data. The BCC structure has a single atom in the center of the cube, while the FCC has a void in its center but has atoms on the center of the cubes' faces, see Figure 4.2 for a visual representation of these crystal lattices.

The BCC and FCC crystal structures are distinct when viewed through the lens of topological data analysis (TDA). Differentiating between the holes and connectedness of these two lattice structures allows us to create an accurate classification rule. This fundamental distinction between BCC and FCC point clouds is captured well by topological methods and explains the high degree of accuracy in the classification scheme presented herein. TDA provides input features for machine learning algorithms, as well as a useful toolbox



(a) BCC cell



(b) FCC cell

Figure 4.2: Example of body-centered cubic, (BCC), (a) and face-centered cubic, (FCC), (b) unit cells without additive noise or sparsity. Notice there is an essential topological difference between the two structures: The body-centered cubic structure has one atom at its center, whereas the face-centered cubic is hollow in its center, but has one atom in the middle of each of its faces.

for classification. Several authors have used TDA on real-world problems, see [76, 77, 78, 79, 80, 81, 82, 83, 32, 84] and the references therein. Persistent homology, which measures changes in topological features over different scales, is the main framework considered by these authors.

Persistent homology is used to create features used as input to machine learning algorithms, and as such, it is applicable to classification problems. The technique provides a multi-scale analysis of data as it differentiates holes within the data as viewed in different dimensions, e.g., the space enclosed by a loop is a one-dimensional hole., This methodology provides a summary of the connectedness and holes (empty space in atomic cells) of data, which indirectly gives information about the shape of the data as well and its subsequent analysis quantifies the significance of a homological feature and provides a tool to contend with noisy data. Persistent homology records when different homological features emerge and vanish in the data. The appearance and disappearance of a homological feature is calculated and recorded in a persistence diagram or barcode plot. The persistence diagram yields a topological summary of the persistent homology of a dataset and are rich sources of detail about underlying topological features. These diagrams may be used in distance-based classifiers [85, 35] or vectorized and input into standard classification algorithms, such as support vector machines [86, 87].

Distances on the space of persistence diagrams yield a means of comparison between diagrams, from which we choose to create features for our classification scheme. Motivated by [35], we consider the d_p^c distance, a distance on the space of persistence diagrams, as opposed to the Wasserstein and bottleneck distances, which are traditionally used to compute distances between persistence diagrams. These two distance metrics compute the cost of the optimal matching between the points in each persistence diagram, while allowing matching to additional points on the diagonal to allow for cardinality differences and to prove

stability properties as in [88]. The d_p^c distance however, employs the cardinality of the persistence diagrams, as well as distances between points in the diagrams to compute their similarity. It calculates the cost of an optimal matching between the persistence diagrams without any points added to the diagonal, as opposed to the Wasserstein or bottleneck distances. A regularization term then adds a penalty for differences between the cardinality of the persistence diagrams.

4.1.1 Stability

Here we will prove a stability theorem for the d_p^c distance. Stability of the distance under investigation means that small perturbations in the underlying space result in small perturbations of the generated persistence diagrams. This property guarantees that when the distances between point clouds go to zero, the distances between the associated persistence diagrams go to zero as well. Another formulation of this stability is given in [89]; using a related approach, we show continuity of the mapping of point cloud to persistence diagram under the d_p^c distance. This analysis provides insight into how the cardinality of the diagrams changes with the size of the input point clouds. First, let us recall the definition of the d_p^c distance.

Definition 4.1. *Let D^1 and D^2 be two persistence diagrams with cardinalities n and m respectively such that $n \leq m$ and denoted $D^1 = \{x_1, \ldots, x_n\}$, $D^2 = \{y_1, \ldots, y_m\}$. Let $c > 0$ and $1 \leq p < \infty$ be fixed parameters. The d_p^c distance between two persistence diagrams D^1 and D^2 is*

$$d_p^c(D^1, D^2) = \left(\frac{1}{m} \left(\min_{\pi \in \Pi_m} \sum_{\ell=1}^{n} \min(c, \|x_\ell - y_{\pi(\ell)}\|_\infty)^p + c^p |m - n| \right) \right)^{\frac{1}{p}}, \tag{4.1.1}$$

where Π_m is the set of permutations of $(1, \ldots, m)$. If $m < n$, define $d_p^c(D^1, D^2) := d_p^c(D^2, D^1)$.

Considering discrete point clouds whose distances shrink to zero, Theorem 4.1 shows that the distance between persistence diagrams goes to zero as well.

Theorem 4.1 (Stability Theorem)**.** *Consider $c > 0$ and $1 \leq p < \infty$. Let A be a finite nonempty point cloud in $\mathbb{R}^d$. Suppose that $\{A_i\}_{i \in \mathbb{N}}$ is a sequence of finite nonempty point clouds such that $d_p^c(A, A_i) \to 0$ as $i \to \infty$. Let D^k and D_i^k be the k-dim persistence diagrams created from the Vietoris-Rips complex for A and A_i respectively. Then $d_p^c(D^k, D_i^k) \to 0$ as $i \to \infty$.*

Note that Theorem 4.1 does not depend on a function created from the points such as a kernel density estimator as in [33], but simply on the points themselves and the Vietoris-Rips complex generated from these points. In fact, Theorem 4.1 shows that the mapping from a point cloud to the persistence diagram of its

Vietoris-Rips complex is continuous under the d_p^c distance. This continuous-type stability result is weaker than Lipschitz stability. In order to prove Theorem 4.1, we first show that if the d_p^c distance between the underlying point clouds goes to 0, then eventually the size of the point clouds must be the same.

Lemma 4.2. *Let A and A_i be as in Theorem 4.1 such that $d_p^c(A, A_i) \to 0$ as $i \to \infty$. Then A_i and A have the same number of points for $i \geq N_0$ for some $N_0 \in \mathbb{N}$.*

Proof. Denote by |A| the number of points in the point cloud A. Suppose that $|A_i| \neq |A|$ infinitely often. Since $d_p^c(A, A_i) \to 0$, for every $\epsilon > 0$, there is an $N \in \mathbb{N}$ such that $i \geq N$ implies that $d_p^c(A, A_i) < \epsilon$. Let $\epsilon = \frac{c}{|A|+1}$, noting that $|A|$ is fixed. By assumption $|A_i| < |A|$, $|A_i| > |A|$, or both, infinitely often. If $|A| < |A_i|$, then by Section 2.2.1

$$d_p^c(A, A_i) \geq \left(c^p \frac{|A_i| - |A|}{|A_i|} \right)^{\frac{1}{p}} \geq c \frac{|A_i| - |A|}{|A_i|}. \tag{4.1.2}$$

The function $h : \mathbb{N} \to \mathbb{R}$ given by $h(z) = \frac{z-|A|}{z}$ is strictly increasing. Whenever $|A| < |A_i|$, we have $|A_i| \geq |A|+1$. The restriction of h to $\{|A|+1, |A|+2, |A|+3, \ldots\}$ achieves its minimum at $|A|+1$. This shows that the RHS of Equation (4.1.2) is greater than or equal to $\frac{c}{|A|+1}$, whenever $|A| < |A_i|$, which by assumption happens infinitely often. This contradicts $d_p^c(A, A_i) < \epsilon$ for all $i \geq N$. The case where $|A| > |A_i|$ follows similarly. ∎

Lemma 4.3. *Let A and A_i be as in Theorem 4.1. Suppose the points of each point cloud A_i are ordered so that $A_i = \{a_{\pi_i(1)}, a_{\pi_i(2)}, \ldots, a_{\pi_i(|A|)}\}$, where π_i is the permutation used to calculate the d_p^c distance between A_i and A as in Equation (2.2.1). Let D_A and D_{A_i} be the distance matrices for the points of A and A_i respectively, i.e., the kl-th entry of D_A is $\|a_k - a_l\|_d$. Then,*

(i) $\|D_A - D_{A_i}\|_\infty \to 0$ as $i \to \infty$, and

(ii) for some $N_1 \in \mathbb{N}$, the order of the entries of the upper triangular portion of D_A and D_{A_i} is the same for $i \geq N_1$, up to permutation when either D_A or D_{A_i} have duplicate entries.

Proof. (i) Let $A = \{a_1, \ldots a_k\}$, $A_i = \{a_1^i, \ldots a_k^i\}$, and $\lambda_\alpha^i = \|a_\alpha - a^i_{\pi_i(\alpha)}\|_d$ for the permutation π_i in the d_p^c distance between A_i and A. Suppose that $d_p^c(A, A_i) \to 0$. Note that since c is fixed, then by Theorem 4.2, there is some N_c such that eventually $d_p^c(A_i, A) = \left(\frac{1}{|A|} \min_{\pi_i \in \prod_{|A|}} \sum_{\ell=1}^{|A|} \|a_\ell - a_{\pi_i(\ell)}\|_d^p \right)^{\frac{1}{p}}$ for $i \geq N_c$. By assumption $d_p^c(A, A_i) \to 0$, which shows that $|A|^{-\frac{1}{p}} \|\lambda\|_p \to 0$ as $i \to \infty$. Thus $\|\lambda^i\|_p \to 0$ as $i \to \infty$.

Now, let $E = D_A - D_{A_i}$.

$$\begin{aligned}
\|E\|_\infty &= \max_{k,l} \left| \|a_k - a_l\|_d - \|a_k^i - a_l^i\|_d \right| \\
&= \max_{k,l} \left| \|a_k - a_l\|_d + \|a_l - a_k^i\|_d - \|a_l - a_k^i\|_d - \|a_k^i - a_l^i\|_d \right| \\
&\leq \left| \|a_k - a_l\|_d - \|a_l - a_k^i\|_d \right| + \left| \|a_k^i - a_l^i\|_d - \|a_l - a_k^i\|_d \right| \\
&\leq \|a_k - a_k^i\|_d + \|a_l - a_l^i\|_d \qquad (4.1.3)
\end{aligned}$$

The last term in Equation (4.1.3) goes to 0 as $i \to \infty$, proving (i).

(ii) Suppose that the m distinct upper triangular entries of D_A are ordered from smallest to largest, say $d_1^A < d_2^A < \cdots d_m^A$, where $m \leq |A|(|A|-1)/2$. For $\eta \in \{1, \ldots, m+1\}$ let $h_\eta \subset [0, \infty)$ be a sequence such that $h_1 < d_1^A < h_2 < d_2^A < \cdots < h_m < d_m^A < h_{m+1}$. Let $\|D_A - D_{A_i}\|_\infty < \frac{h}{2}$, where $h = \min_{\eta_1, \eta_2 \in \{1,\ldots,m+1\}} \{|h_{\eta_1} - h_{\eta_2}|\}$. We show that there exists a sequence g_η such that $|h_\eta - g_\eta| < 2h$ for each $\eta \in \{1, \ldots, m+1\}$ and $h_\eta < d_j^A < h_{\eta+1}$ implies $g_\eta < d_j^{A_i} \leq g_{\eta+1}$. Let $h_\eta < d_j^A < h_{\eta+1}$, and suppose that it is not the case that $h_\eta < d_j^{A_i} \leq h_{\eta+1}$. Since $\|D_A - D_{A_i}\|_\infty < \frac{h}{2}$, then either $d_j^{A_i} \in (h_{\eta-1}, h_\eta]$ or $d_j^{A_i} \in (h_{\eta+1}, h_{\eta+2}]$. If the first case is true, then take $g_\eta = d_j^A - \frac{h}{2}$. If the second, then take $g_\eta = d_j^A + \frac{h}{2}$. This proves the existence of the sequence. Now proceeding by contradiction, if the lemma does not hold for some entries $d_j^A \in D_A$ and $d_j^{A_i} \in D_{A_i}$, then take $\|D_A - D_{A_i}\|_\infty < \frac{1}{2}|d_j^A - d_j^{A_i}|$. ∎

Proof of Theorem 4.1. By Lemma 4.2, take $|A_i| = |A|$ without loss of generality. By Lemma 4.3 (i), $\|D_A - D_{A_i}\|_\infty \to 0$ as $i \to \infty$. If the Vietoris-Rips complex were computed at every threshold value in $[0, \infty)$, then the birth and death times of all features of all dimensions would be distances between points in the underlying point cloud (including the birth time of 0 in the 0-dim diagram). Since the order of the entries of D_A and D_{A_i} may be taken to be the same from Lemma 4.3 (ii), the same number of simplices are formed in the complexes for A and A_i for each dimension of simplex. Also, the labels of the simplices according to the points of A and A_i are given from the permutation π_i in Lemma 4.3 (i).

Now, for 0-dim it is clear that for the cardinalities of the persistence diagrams, $|D^0| = |D_i^0|$ since for the sizes of their associated point clouds, $|A_i| = |A|$. For a higher dimensional feature ($k \geq 1$) to appear in the complex, we must have that a certain number of the distances are less than or equal to the threshold ϵ and a certain number of the distances are greater than ϵ. Lemma 4.3 (ii) shows that although the thresholds where the features are created may be different, the same number of features are formed in the Vietoris-Rips

complexes of A and A_i, and these features are formed in the same order and with the points that correspond under π_i.

If $D^k = \{x_1, x_2, \ldots, x_{|D^k|}\}$ and $D_i^k = \{x_1, x_2, \ldots, x_{|D_i^k|}\}$, then we have that $|D^k| = |D_i^k|$ and that $d_p^c(D^k, D_i^k) < 2h$. Thus $d_p^c(D^k, D_i^k) \to 0$ as $i \to \infty$. ∎

To provide a practical way to control c in computing the d_p^c distance of Equation (2.2.1) and consequently compute the possible fluctuations of the d_p^c distance, a probabilistic upper bound, which relies on least squares, is provided. Specifically, the following analysis gives predictions on the number of 1-dim holes represented in the persistence diagram, which we denote by b_1. The parameter b_1 relies on the number of connected components (or equivalently the number of points in the point cloud) represented in the persistence diagram, denoted by b_0.

Definition 4.2 ([90]). *The kissing number in $\mathbb{R}^d$ is the maximum number of nonoverlapping unit spheres that can be arranged so that each touches another common central unit sphere.*

Lemma 4.4 ([91]). *For a finite point cloud with no more than ρ points in $\mathbb{R}^d$ under the Euclidean distance, let $M_d(\rho)$ denote the maximum possible number of 1-dim holes in the Vietoris-Rips complex for the point cloud for a given threshold. Then*

$$M_d(\rho) \leq (K_d - 1)\rho. \tag{4.1.4}$$

Proposition 4.5. *Consider a point cloud in $\mathbb{R}^d$ with ρ points and its associated persistence diagram. Let B_1 denote the possible range of the number of 1-dim holes b_1. Then B_1 is such that $\{0, 1, \ldots, \lfloor \frac{\rho}{2} \rfloor - 1\} \subseteq B_1 \subseteq \{0, 1, \ldots, \frac{1}{2}(K_d - 1)\rho^2(\rho - 1)\}$, i.e., the possible range of b_1 is expanding as the number of points, b_0, in the point cloud increases.*

Proof. We first show the inclusion $\{0, 1, \ldots, \lfloor \frac{\rho}{2} \rfloor - 1\} \subseteq B_1$. To form a point cloud with ρ points that has $b_1 = 0$, simply take the ρ points and arrange them on a line. To form a point cloud with ρ points that has $b_1 = \lfloor \frac{\rho}{2} \rfloor - 1$, arrange the ρ points in two rows each with $\lfloor \frac{\rho}{2} \rfloor$ points. Set the spacing between adjacent points in each of the rows to be 1 and then place the two rows directly beside each other so that for each point in the first row, there is exactly one point in the second row at a distance of 1. Adding edges appropriately creates $b_1 = \lfloor \frac{\rho}{2} \rfloor - 1$ squares with side length 1. Thus, creating the Vietoris-Rips complex and corresponding diagram gives $b_1 = \lfloor \frac{\rho}{2} \rfloor - 1$. For an illustration of the arrangement, see Fig. 4.3a.

(a) (b)

Figure 4.3: An example of 8-point arrangements to visualize the proof of Proposition 4.5. (a) A 3-hole configuration vs. (b) a 2-hole configuration.

To form a point cloud with ρ points that has $b_1 \in \{1, 2, \ldots \lfloor \frac{\rho}{2} \rfloor - 2\}$, arrange $2(b_1 + 1)$ points in two rows as in Fig. 4.3a. Arrange the other $\rho - 2(b_1 + 1)$ points in a line with the minimum distance from any points in the line to points of the two rows such that it is greater than or equal to 1. Then exactly b_1 holes are formed from the two rows, with no holes formed by the line. For an illustration, see Fig. 4.3b.

Next, we verify the inclusion $B_1 \subseteq \{0, 1, \ldots, \frac{1}{2}(K_d - 1)\rho^2(\rho - 1)\}$. By Lemma 4.4, the number of 1-dim holes in the Vietoris-Rips complex for a fixed radius ϵ for the point cloud is bounded above by $(K_d - 1)\rho$. The homology of the Vietoris-Rips complex changes at most $\binom{\rho}{2}$ times as the radius ϵ increases due to the maximum of $\binom{\rho}{2}$ distinct distances between points in the point cloud. Therefore, there can be at most $\frac{1}{2}(K_d - 1)\rho^2(\rho - 1)$ 1-dim holes formed over the entire evolution of the Vietoris-Rips complex. This gives the desired bound of $b_1 \leq \frac{1}{2}(K_d - 1)\rho^2(\rho - 1)$. ∎

Now, let N point clouds be generated from some process, and N corresponding persistence diagrams be created. For each persistence diagram $D_i^k, k \in \{0, 1\}, i = 1, \ldots, N$, record the cardinality b_0^i of the 0-dim diagram and the cardinality b_1^i of the 1-dim diagram. Let $\boldsymbol{b_0} \in \mathbb{R}^{N \times 2}$ be the predictor matrix whose rows are $[1 \ b_0^i]$ and $\boldsymbol{b_1} \in \mathbb{R}^N$ be the vector of responses with entries b_1^i. Proposition 4.5 gives that the possible range of $\boldsymbol{b_1}$ is increasing as $\boldsymbol{b_0}$ grows, which yields that an increase in variance as $\boldsymbol{b_0}$ grows may be present, i.e., heteroscedasticity exists. Thus the analysis of the change in number of 1-dim holes as the size of the point cloud changes needs to account for heteroscedasticity in order to capture the non-constant variance behavior. Therefore to estimate the number of 1-dim holes, we use weighted least squares as in [51]. If $\mathbf{W} \in \mathbb{R}^{N \times N}$ is the weight matrix $\mathbf{W} = \text{diag}(a_1, \ldots, a_N)$, then a weighted least-squares regression

can be found for $\boldsymbol{b_1} = \boldsymbol{b_0}\boldsymbol{\gamma} + \boldsymbol{\epsilon}$, where $\epsilon_i \sim \mathcal{N}(0, \sigma_i^2)$. The approximation is then given by $\mathbf{b_0}\hat{\boldsymbol{\gamma}} = \boldsymbol{b_1}$, with $\hat{\boldsymbol{\gamma}} = (\mathbf{b_0}^T \mathbf{W} \mathbf{b_0})^{-1} \mathbf{b_0}^T \mathbf{W} \boldsymbol{b_1}$. In turn, Proposition 4.6 provides bounds from prediction intervals using weighted least squares for the d_p^c distance.

Proposition 4.6. *Suppose N point clouds are generated from a process, and N corresponding persistence diagrams are created. For each persistence diagram $D_i^k, k \in \{0, 1\}$, record the cardinality of the 0-dim diagram b_0^i and of the 1-dim diagram b_1^i. Let $\boldsymbol{b_0} \in \mathbb{R}^{N \times 2}$ be the predictor matrix whose rows are $[1\ b_0^i]$ and $\boldsymbol{b_1} \in \mathbb{R}^N$ be the vector of responses of b_1^i. Assume the model $\boldsymbol{b_1} = \mathbf{b_0}\boldsymbol{\gamma} + \boldsymbol{\epsilon}$, where $\epsilon_i \sim \mathcal{N}(0, \sigma_i^2)$ depends on the value of the input b_0^i. Let D^1 and $\widetilde{D}^1$ be persistence diagrams generated from the same process as $\boldsymbol{b_0}$ with $|D^0| = \mu$. Considering the $(1-\alpha)\cdot 100\%$-level prediction interval for $\boldsymbol{b_1}$, the distance $d_p^c(D^1, \widetilde{D}^1)$ is bounded above by*

$$\left(\min_{\pi \in \Pi_m} \sum_{\ell=1}^{n} \min(c, \|d_\ell^1 - \tilde{d}_{\pi(\ell)}^1\|_\infty)^p + c^p 2 t_{1-\alpha, N-2} s \sqrt{[1\ \mu](\mathbf{b_0}^T \mathbf{W} \mathbf{b_0})^{-1}[1\ \mu]^T + \mu} \right)^{\frac{1}{p}}.$$

Proof. Prediction intervals can be constructed for the cardinality of a 1-dim diagram for an instance of point cloud size ${b_0}^*$ using standard results on weighted least squares. Specifically, for level $(1-\alpha)\cdot 100\%$ a prediction interval for the new response $\widehat{b_1}^*$ is sought. To calculate this interval for a new response from the mean predicted response $\widehat{b_1}^* = \hat{\boldsymbol{\gamma}} {b_0}^*$, note that $\widehat{b_1}^* - {b_1}^*$ has the distribution $\frac{\widehat{b_1}^* - b_1^*}{\mathrm{Var}(\widehat{b_1}^* - {b_1}^*)} \sim t_{N-2}$. Also, $\mathrm{Var}(\widehat{b_1}^* - {b_1}^*) = \mathrm{Var}(\boldsymbol{\epsilon})[1\ {b_0}^*](\mathbf{b_0}^T \mathbf{W} \mathbf{b_0})^{-1}[1\ {b_0}^*]^T + \frac{\mathrm{Var}(\boldsymbol{\epsilon})}{w^*}$, where $w^* = \frac{1}{b_0^*}$, the weight corresponding to ${b_0}^*$. Prediction intervals for b_1^* are thus $\widehat{b_1}^* \pm t_{1-\alpha/2, N-2} s \sqrt{[1\ {b_0}^*](\mathbf{b_0}^T \mathbf{b_0})^{-1}[1\ {b_0}^*]^T + {b_0}^*}$, where $s^2 = \frac{\hat{\boldsymbol{\epsilon}}^T \mathbf{W} \hat{\boldsymbol{\epsilon}}}{N-2}$, the unbiased estimator for $\mathrm{Var}(\boldsymbol{\epsilon})$, using the residuals $\hat{\boldsymbol{\epsilon}}$. Thus the cardinality difference term in the calculation of the d_p^c distance as in Equation (2.2.1) is bounded above by the length of the prediction interval with $(1-\alpha)\cdot 100\%$-level confidence. Substituting this length into Equation (2.2.1) gives the result. ∎

The space of persistence diagrams endowed with the d_p^c metric is complete and separable [35], and so it admits Fréchet means and variances. In the context of persistence diagrams, the Fréchet mean of any persistence diagram minimizes the Fréchet variance, and we use these moments to create features for our classification methodology to infer the lattice structure of the materials data.

4.2 Classification of Materials Data

Here we describe the d_p^c-distance based classification of crystal structures of high-entropy alloys using data from atom probe tomography experiments. Recall that the building blocks of HEAs are either body-centered cubic or face-centered cubic. Topological considerations are a natural fit for this problem since BCC and FCC crystal structures enjoy a different atomic configuration within a unit cell. Indeed, the BCC structure has one atom at its center, but the FCC contains a void (recall Figs. 4.2a and 4.2b). This distinction is important from the viewpoint of persistent homology.

Each of these topological features has a birth and death time associated with specific radii. These birth and death pairs, written $(birth, death)$, are recorded in a persistence diagram. Thus, each point in a persistence diagram is generated by a topological feature whose birth and death radii are the point's (x, y)-coordinates respectively. The corresponding diagram for the atomic neighborhood in Figure 4.9(a) is shown in Figure 4.9(e). The persistence diagram encodes information about the structure of each neighborhood by providing insight about the number of atoms, the size and distance among atoms, possible configuration of the faces, and geometric structure. The persistence diagram then functions as a proxy for data by reducing an atomic neighborhood to its most pertinent qualities.

To study the persistent homology of atomic structures extracted by HEAs, we create spheres of increasing radii around each atom in a neighborhood and record when homological features emerge and disappear. Taking the atoms' spatial positions in the xyz-coordinate system, we begin by drawing a sphere of radius ϵ around each atom, see Figure 4.9(a). Increasing the radii of the spheres, we note the associated radii at which they intersect and form new topological features, e.g., a 1-dim hole has emerged as noted by the arrow in Figure 4.9(b). Further increasing the radii, we observe that more of the spheres have merged and see where a 2-dim void will form in Figure 4.9(c). Continuing the process, we increase the radii until all spheres mutually intersect, i.e., all the holes are closed (the feature is said to have died), and no additional information is gained by increasing the radii further, as seen in Figure 4.9(d). By examining homological changes to the neighborhood for increasing radii instead of a single value, we are able to capture information about the shape of the neighborhood itself. This type of multiscale analysis is key to bypassing the noise and sparsity present in the data and to extract meaningful details about the configuration of the neighborhood.

The extracted persistence diagrams generated by APT experiments summarize the shape peculiarities of each atomic neighborhood. Different types of lattice structures yield persistence diagrams with various identifying features. Figure 4.4 displays the difference between persistence diagrams for BCC and FCC

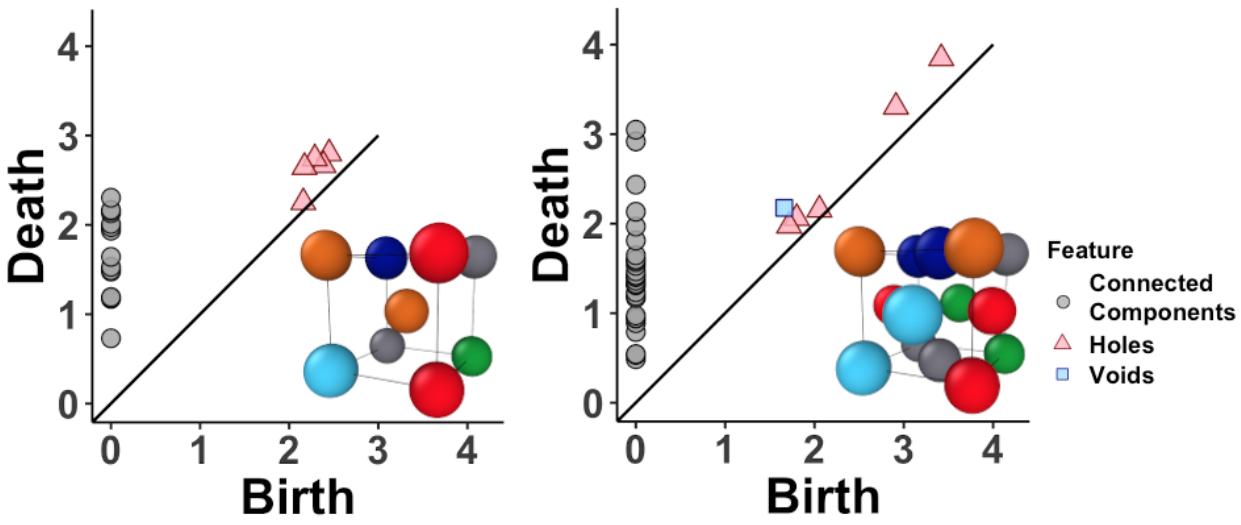


Figure 4.4: Sample persistence diagrams of a material from APT data of the alloys $Al_{1.3}$CoCrCuFeNi and $Al_{0.3}$CoCrFeNi for the two lattice types considered here: BCC (a) and FCC (b), respectively. Notice the distinguishing 2-dim feature, the blue square, in the diagram derived from an FCC lattice, and the diagram generated from the BCC structure has fewer 0-dim features.

structures. The persistence diagrams capture differences in (*i*) the number of atoms (8 for BCC and 12 for FCC), (*ii*) the spacing between neighbors i.e., packing density, and (*iii*) the arrangement of neighbors.

For a given configuration, the persistence diagram can be compared to a reference persistence diagram for BCC and FCC via a similarity metric. As different crystal structures produce different size point clouds [92], this information must be considered when creating our materials fingerprint. To properly account for differences in the number of points when comparing two persistence diagrams, we employ the d_p^c distance, introduced in [35]. This distance matches points between the persistence diagrams being compared, and those that are unmatched are penalized by a regularization term. For a visual representation of how the distance works, see Figure 4.9(f). In developing our materials fingerprint, we compute the d_p^c distance between persistence diagrams with respect to 0, 1, and 2−dim homological features, i.e., connected components, holes, and voids. We then compute summary statistics (mean, variance) from these distances to create features for the classification algorithm.

However, topology alone is insufficient to distinguish between noisy and sparse BCC and FCC lattice structures accurately. If we count the number of atoms in a unit cell (see Figs. 4.2a and 4.2b) one may observe that a BCC unit cell has two atoms, one at the center and $1/8^{th}$ of an atom at the unit cell's corners, as it shares part of these corner atoms with its neighboring cells. Similarly, an FCC unit cell has four atoms; the same $1/8^{th}$ of the corner atoms plus one-half of each of the six atoms on the cell's faces. In both cases, the

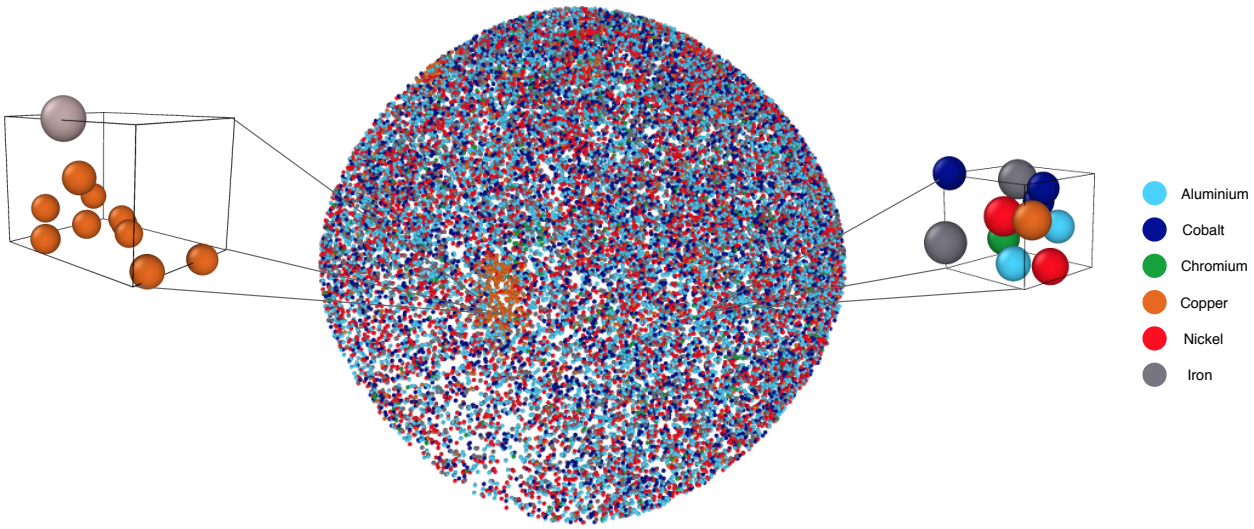


Figure 4.5: Image of APT data with atomic neighborhoods shown in detail on the left and right. Each pixel represents a different atom, the neighborhood of which is considered. Certain patterns with distinct crystal structures exist, e.g., the orange region is copper-rich (left), but overall no pattern is identified. Putting a single atomic neighborhood under a microscope, the true crystal structure of the material, which could be either BCC (Fig. 4.2a) or FCC (Fig. 4.2b), is not revealed. This distinction is obscured due to experimental noise and sparsity present in the dataset.

atoms on the faces and lattice points are shared with the cell's neighbors and are only counted as a proportion contributing to the unit cell.

Another way to see this difference in cardinality is by plotting the number of connected components against the number of holes for both BCC and FCC crystal structures. Figs. 4.7c and 4.7d depict that FCC structures have larger point clouds, and consequently, a greater number of connected components. Observe in Figure 4.6 that the number of connected components and 1-dim holes are greater in the FCC diagrams than the BCC diagrams. Consequently, we must account for more than just homological differences when considering persistence diagrams derived from these atomic neighborhoods. Variability in the size of the underlying point clouds must be considered, as verified in Proposition 4.6. Given the salient topological and cardinality differences between these two crystal structures, we seek to classify their associated persistence diagrams via these essential differences. To that end, we consider the d_p^c distance defined for two persistence diagrams by Definition 4.1.

4.2.1 Classification Model

In the numerical experiments, the point clouds (atomic neighborhoods) are either extracted from a sample containing approximately 10^7 atoms, or created to model the real APT data for a sensitivity analysis. In the

latter case, we remove atoms, to create spasity, and add Gaussian noise to the larger sample mirroring those levels found in true APT experimental data. In each case, to create these neighborhoods, we consider a fixed volume around each atom in the perturbed sample and those atoms within the volume are recorded for our classification methodology.

We write D_i as the persistence diagram generated by atom positions in an atomic neighborhood retrieved by the APT experiment as seen in Figure 4.5. Note that the number of atoms in a neighborhood is not constant, but varies between atomic neighborhoods in a sample. For our classification problem, we are interested in modeling the conditional probability $\pi(X) = \mathbb{P}(Y = 1 \mid X)$ for a given input X, and associated response Y. We write $Y = -1$ or $Y = 1$ to denote a BCC or FCC lattice respectively. To that end, we consider a generalized additive regression model [48, 49]. Choosing this type of model gives us the flexibility to let our data determine the correct functional form, as opposed to imposing a linear model as in traditional logistic regression. Our model is thus written

$$\log\left(\frac{\pi(X)}{1-\pi(X)}\right) = \sum_{j=1}^{P} f_j(X_j), \tag{4.2.1}$$

where f_j is some pertinent smooth function and $X = (X_1, \ldots, X_P)$ is a random vector, i.e., that is a vector where each entry is a random variable. Concatenating the random vectors we create our feature matrix, $\mathbf{X} \in \mathbb{R}^{N\times P}$, where $N = N_1 + N_2$. Here N_1 is the number of BCC persistence diagrams and N_2 denotes the number of FCC diagrams, for $P = 4(k+1)$. Indeed, an arbitrary row of $\mathbf{X}$ is

$$X_i = (\mathbb{E}^0_{i,B}, \mathbb{E}^1_{i,B}, \mathbb{E}^2_{i,B}, \mathrm{Var}^0_{i,B}, \mathrm{Var}^1_{i,B}, \mathrm{Var}^2_{i,B}, \mathbb{E}^0_{i,F}, \mathbb{E}^1_{i,F}, \mathbb{E}^2_{i,F}, \mathrm{Var}^0_{i,F}, \mathrm{Var}^1_{i,F}, \mathrm{Var}^2_{i,F}). \tag{4.2.2}$$

for any persistence diagram with $k-$dim homology, $k = \{0, 1, 2\}$. We write $\mathbb{E}^k_{i,B} = \frac{1}{N_1}\sum_{j=1}^{N_1} d_p^c(D_i^k, D_j^k)$ and $\mathrm{Var}^k_{i,B} = \frac{1}{N_1-1}\sum_{j=1}^{N_1}(d_p^c(D_i^k, D_j^k) - \mathbb{E}^k_{i,B})^2$ as the mean and variance respectively, as measured by the d_p^c distance given in Definition 4.1, between any diagram D_i^k and the collection of all BCC persistence diagrams. Similarly, we write $\mathbb{E}^k_{i,F}$ and $\mathrm{Var}^k_{i,F}$ as the mean and variance of any diagram compared with all diagrams in the FCC collection.

Having the persistence diagrams, we next compute the feature matrix according to Equation (4.2.2). This matrix is used as input to the AdaBoost algorithm as implemented in the scikit-learn library [93]. The pseudo-code for our fingerprinting method is shown in Algorithm 4.1. Having computed the feature matrix, we employ 10-fold cross validation on the entire dataset to control for overfitting and to obtain an estimate of the generalization ability of our model. To do this, we split the dataset into 10 partitions. For each partition,

τ	c-value	*Accuracy*
0.0	0.01	99%
0.25	0.05	99.4%
0.75	0.03	96.5%
1.0	0.13	96.4%

Table 4.1: The atomic positions in the APT data is $\mathcal{N}(0, \tau^2)$ distributed with 67% of the atoms missing. We employ the d_p^c classifier, where c has been optimized in each noise level case. The accuracy in the 10-fold cross validation is listed in the third column.

we create a classification rule from the other 9 partitions, and use the remaining one as a test set. Our accuracy, defined here as (1 - Misclassification rate), is recorded for each partition as it is used as the test set. Each partition is used for the testing phase only once, and the accuracy rate is averaged over all 10 partitions. For each persistence diagram in the training set, we compute the d_p^c distance among all diagrams with $k-$dim homology, $(k = 0, 1, 2)$, and the associated moments according to Equation (4.2.2). These moments are used to create the ensemble classifier. Next, for any unknown crystal structure in the test set, the associated feature vector is computed according to Equation (4.2.2) and used as input to the ensemble classifier. The classifier finds the best fit for the unknown crystal structure from our additive model and returns the class probabilities of the unknown structure.

4.2.2 Sensitivity Analysis

For our numerical experiments, the persistence diagrams are constructed using the C++ Ripser software, and the scikit-learn decision tree implementation. The studies [25, 1] estimate that approximately 65% of the data is missing. However, an estimate of the experimental noise is not provided. In fact, as noted by [22, 23], the noise varies between experiments and specimens. The data used in our sensitivity analysis replicates this resolution by drawing from a Gaussian [27, 94, 69], $\mathcal{N}(0, \tau^2)$, with four different levels of variance to give a more representative approximation of true APT datasets. Computing the d_p^c distances for 0- and 1-dim homology with $p = 2$ to imitate typical Euclidean distance, we find different values of c via a grid search for these four different levels of variance, τ^2, for both 0- and 1-dim homology, employing a different dataset than is used for the classification. In each case, a geometric sequence of 10 values between 0.01 and 1 is taken into account. The results and the associated algorithmic accuracy are presented in Table 4.1.

As a comparison the feature matrix in Equation (4.2.2) is also calculated using the Wasserstein distance, choosing $p = 2$. Moreover, we adopt a counting classifier which takes into account only the number of points in an atomic neighborhood as the input feature in the tree classifier. Our d_p^c classifier successfully

Algorithm 4.1 Materials Fingerprinting

Training Step

Read in labeled APT data (training set), both FCC and BCC, and compute persistence diagrams in the training set $\mathcal{D}_{train}$, which has N_1 diagrams from BCC data and N_2 diagrams from FCC data.

Read in response vector $Y = (-\mathbf{1}, \mathbf{1})^T$ where $-\mathbf{1}$ is a vector of -1's in $\mathbb{R}^{N_1}$ and analogously for $\mathbf{1} \in \mathbb{R}^{N_2}$.

for $i = 1, \ldots, N_1 + N_2$ **do**

Compute feature matrix $\mathbf{X}$ according to Equation (4.2.2)

$$\mathbf{X}_i = (\mathbb{E}^0_{i,B}, \mathbb{E}^1_{i,B}, \mathbb{E}^2_{i,B}, \mathrm{Var}^0_{i,B}, \mathrm{Var}^1_{i,B}, \mathrm{Var}^2_{i,B}, \mathbb{E}^0_{i,F}, \mathbb{E}^1_{i,F}, \mathbb{E}^2_{i,F}, \mathrm{Var}^0_{i,F}, \mathrm{Var}^1_{i,F}, \mathrm{Var}^2_{i,F}),$$

where

$$\mathbb{E}^k_{i,B} = \frac{1}{N_1}\sum_{j=1}^{N_1} d_p^c(D_i^k, D_j^k), \quad \mathbb{E}^k_{i,F} = \frac{1}{N_2}\sum_{j=N_1+1}^{N_1+N_2} d_p^c(D_i^k, D_j^k),$$

$$\mathrm{Var}^k_{i,B} = \frac{1}{N_1 - 1}\sum_{j=1}^{N_1} (d_p^c(D_i^k, D_j^k) - \mathbb{E}^k_{i,B})^2,$$

$$\mathrm{Var}^k_{i,F} = \frac{1}{N_2 - 1}\sum_{j=N_1+1}^{N_1+N_2} (d_p^c(D_i^k, D_j^k) - \mathbb{E}^k_{i,F})^2,$$

for $k = \{0, 1, 2\}$.

end for

$C(\mathbf{X}) = \texttt{ADABOOST}(\mathbf{X}, Y)$ ▷ Obtain a classification rule C from AdaBoost ensemble classification algorithm

Testing Step

Read in unlabeled APT point cloud data and compute persistence diagrams $\mathcal{D}_{test} = \{\widehat{D}_j\}_{j=1}^J$.

for $i = 1, \ldots, J$ **do**

Compute

$$\widehat{\mathbf{X}}_i = (\widehat{\mathbb{E}}^0_{i,B}, \widehat{\mathbb{E}}^1_{i,B}, \widehat{\mathbb{E}}^2_{i,B}, \widehat{\mathrm{Var}}^0_{i,B}, \widehat{\mathrm{Var}}^1_{i,B}, \widehat{\mathrm{Var}}^2_{i,B}, \widehat{\mathbb{E}}^0_{i,F}, \widehat{\mathbb{E}}^1_{i,F}, \widehat{\mathbb{E}}^2_{i,F}, \widehat{\mathrm{Var}}^0_{i,F}, \widehat{\mathrm{Var}}^1_{i,F}, \widehat{\mathrm{Var}}^2_{i,F})$$

where

$$\widehat{\mathbb{E}}^k_{i,B} = \frac{1}{N_1}\sum_{j=1}^{N_1} d_p^c(\widehat{D}_i^k, D_j^k), \quad \widehat{\mathbb{E}}^k_{i,F} = \frac{1}{N_2}\sum_{j=N_1+1}^{N_1+N_2} d_p^c(\widehat{D}_i^k, D_j^k),$$

$$\widehat{\mathrm{Var}}^k_{i,B} = \frac{1}{N_1 - 1}\sum_{j=1}^{N_1} (d_p^c(\widehat{D}_i^k, D_j^k) - \widehat{\mathbb{E}}^k_{i,B})^2,$$

$$\widehat{\mathrm{Var}}^k_{i,F} = \frac{1}{N_2 - 1}\sum_{j=N_1+1}^{N_1+N_2} (d_p^c(\widehat{D}_i^k, D_j^k) - \widehat{\mathbb{E}}^k_{i,F})^2,$$

for $k = \{0, 1, 2\}$.

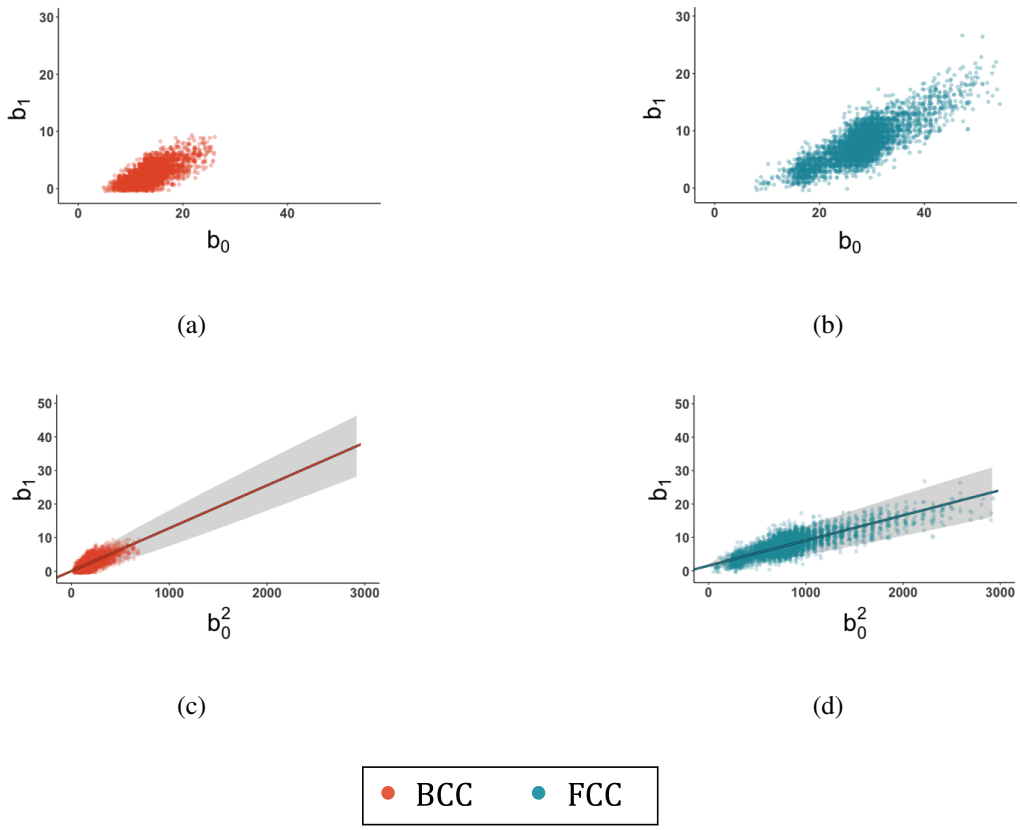
end for

Classify unlabeled APT data

$\widehat{Y} = C(\widehat{\mathbf{X}})$ ▷ Yields class labels for $\mathcal{D}_{test}$ as $\hat{Y} \in \{-1, 1\}^J$.

(a) (b)

Figure 4.6: Example of persistence diagrams generated by (a) a BCC lattice, and (b) FCC lattice. The data has a noise standard deviation of $\tau = 0.75$ and 67% of the atoms are missing. Note that the BCC diagram has two prominent (far from the diagonal) points representing 1-dim holes and fewer connected components and 1-dim holes than does the FCC diagram.



(a) (b)

(c) (d)

Figure 4.7: *Top:* Number of connected components (in this case atoms), $\mathbf{b_0}$, against the number of 1-dim homological features, $\mathbf{b_1}$, of the persistence diagrams. One can see the presence of heteroscedasticity since the variance of $\mathbf{b_1}$ increases as $\mathbf{b_0}$ increases. *Bottom:* Same as in top but using a quadratic transformation of the predictor variable, along with the weighted least squares fit line and 95% prediction intervals provided by Proposition 4.6.

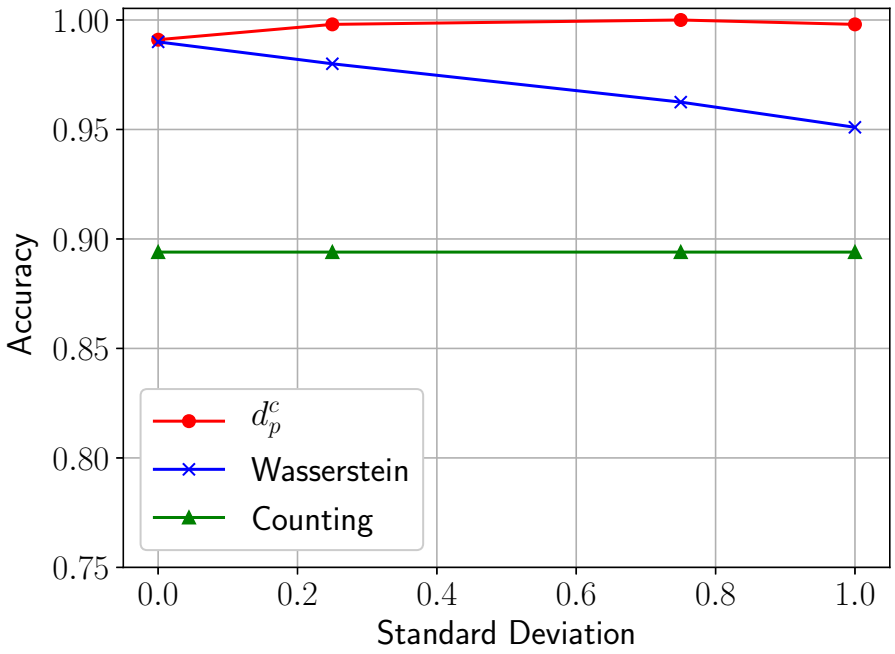


Figure 4.8: 10-fold cross validation accuracy scores for d_p^c (red), Wasserstein (blue), and counting (green) classifiers, plotted against different standard deviations, τ, (see Table 4.1) of the normally distributed noise of the atomic positions. In each instance, the sparsity has been fixed at 67% of the atoms missing, as in a true APT experiment.

dichotomizes these 1,000 persistence diagrams generated by BCC and FCC lattice structures at better than 96% accuracy, where accuracy is measured as (1 - Misclassification rate). The d_p^c classifier outperforms both the Wasserstein and the counting classifier, see Figure 4.8. These results demonstrate that using just the differences in cardinality between the two classes of crystal structures is insufficient to distinguish between them.

As demonstrated in Proposition 4.6, there is a relationship between the number of connected components, $\mathbf{b_0}$, (number of atoms in this case) and the number of 1-dim homological features, $\mathbf{b_1}$, in the persistence diagrams Figs. 4.7a and 4.7b demonstrate this relationship, as well as the presence of heteroscedasticity between $\mathbf{b_0}$ and $\mathbf{b_1}$, also verified by the Breusch-Pagan test [95] with a $p-$value of 9.3×10^{-54} for FCC cells and a $p-$value of 2.01×10^{-47} for BCC cells. Figs. 4.7a and 4.7b also provide 95% prediction intervals for $\mathbf{b_1}$ based on the weighted least squares regression analysis of Proposition 4.6. Additionally, these statistics on the diagram's cardinality generates corresponding prediction intervals, which give probabilistic bounds on the d_p^c distances between persistence diagrams, and we see that point clouds generated from the same process have small variability with respect to cardinality of the persistence diagrams. To that end, this exact fine balance between the number of atoms in a neighborhood and the associated topology created by the positions of these atoms in the cubic cell is captured by the d_p^c distance.

4.3 Materials Fingerprinting

In this section, we describe our novel machine-learning approach, a materials fingerprint, to classify the crystal structure of a material by looking at local atomic neighborhoods through the lens of topological data analysis. As previously discussed, TDA is a field that uses topological features within data for machine learning tasks. It has found other applications in materials science, such as the characterization of amorphous solids [96], equilibrium phase transitions [97], and similarity of pore-geometry in nanomaterials [98]. Our motivation is to encode the geometric peculiarities of HEAs by considering atomic positions within a neighborhood and looking at the neighborhood's topology.

Our fingerprinting process allows us to see the lattice structure of a noisy and sparse dataset. Particularly, key differences between atomic neighborhoods are encoded in the empty space, e.g., holes and voids, between atoms, as well as clusters of atoms in the neighborhood. These clusters of atoms, holes, and voids in an atomic neighborhood can be calculated through the concept of homology, which is the mathematical study of 'holes' in different dimensions. Extracting this homological information from each atomic neighborhood, we can distinguish between the different lattice structures we consider. These topological features differentiate the

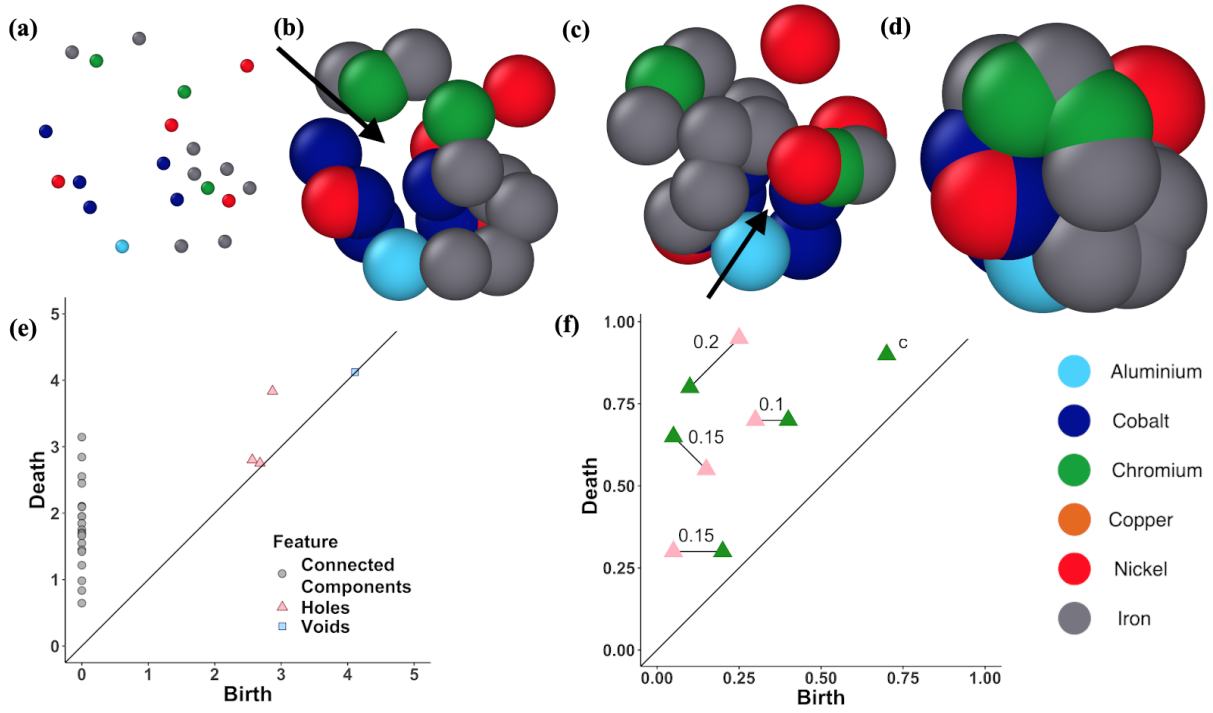


Figure 4.9: Atomic neighborhood from an APT experiment (Figure 4.5); for the alloy $Al_{1.3}CoCrCuFeNi$ where the atomic type is illustrated by the color. (a) shows each atom as a point cloud in $\mathbb{R}^3$. As the radius of the sphere centered at each atom increases in (b), a 1-dim hole forms in the atomic structure. Increasing the radii further, in (c) the formation of a 2-dim hole, a void, is evident. Continuing to increase the radii, in (d) the radii have increased such that all atoms form one cluster. The persistence diagram for this structure is shown in (e). In (f) the d_p^c metric computes the distance between two persistence diagrams. Consider two 1-dim persistence diagrams generated by atomic neighborhoods, one shown by the pink triangles, the other by the green triangles. The d_p^c metric measures the distance between the diagrams by first finding the best matching between points. Any unmatched points are then penalized by the regularization term c.

shape and structure of the neighborhoods and often retain this property even in the presence of experimental noise. Adopting this approach, we are able to classify the crystal structure of HEAs from the APT data with accuracy approaching 100%.

Experimental APT Data

We are interested in developing an automated methodology for classifying the crystal structure of disordered metallic alloys from the APT datasets to inform our registration process. APT is an experimental characterization technique used to determine the local elemental structure. Indeed, the technique produces a dataset of approximately 10^8 atoms [25], from which fundamental information about the structure of a material can be obtained. Combining the elemental information from an APT experiment with the lattice structure provided by our methodology yields an unambiguous representation of local neighborhoods at the atomic level.

Researchers are actively working to improve the APT detection process [24, 25], but the state-of-the-art APT method captures at most 60% of the atoms in a sample [25]. This estimate is perhaps optimistic, as a previous work analyzed an HEA via APT and found that only 37% of the atoms were registered by the process [1]. Additionally, the spatial coordinates of the atoms recorded from a typical APT experiment have added experimental noise. For our problem, the data consists of spatial coordinates of approximately 10^8 atoms with elemental type [25], which compose a highly-disordered metallic alloy that is either BCC or FCC in its lattice structure. We consider these crystal types as they are the building blocks of HEAs, our ultimate object of interest [99, 13, 9].

Our BCC sample under investigation here was chosen because it has been previously well-characterized [1]. It consists of three phases, a Cu-rich FCC phase, an Fe-Cr rich BCC phase and a remaining phase that incorporates all six elements, though the proportions of Cu, Fe and Cr are depleted due to segregation in the other phases. Importantly all three phases are present in the APT sample. When viewing this entire data set with atoms identified by color, some nanoscale information is immediately evident to the eye, see Figure 4.5. The eye perceives elemental segregation of the Cu-rich and Fe-Cr rich phases into nanoscale domains. However, one cannot infer any meaningful structure at a finer scale when viewing the entire dataset from a typical APT experiment. Instead, further scrutiny requires that individual atomic neighborhoods be extracted from the larger sample. Viewing each neighborhood individually, we can see that they contain a wealth of information about the shape of the material under investigation, despite the noise and sparsity present in a typical APT experiment.

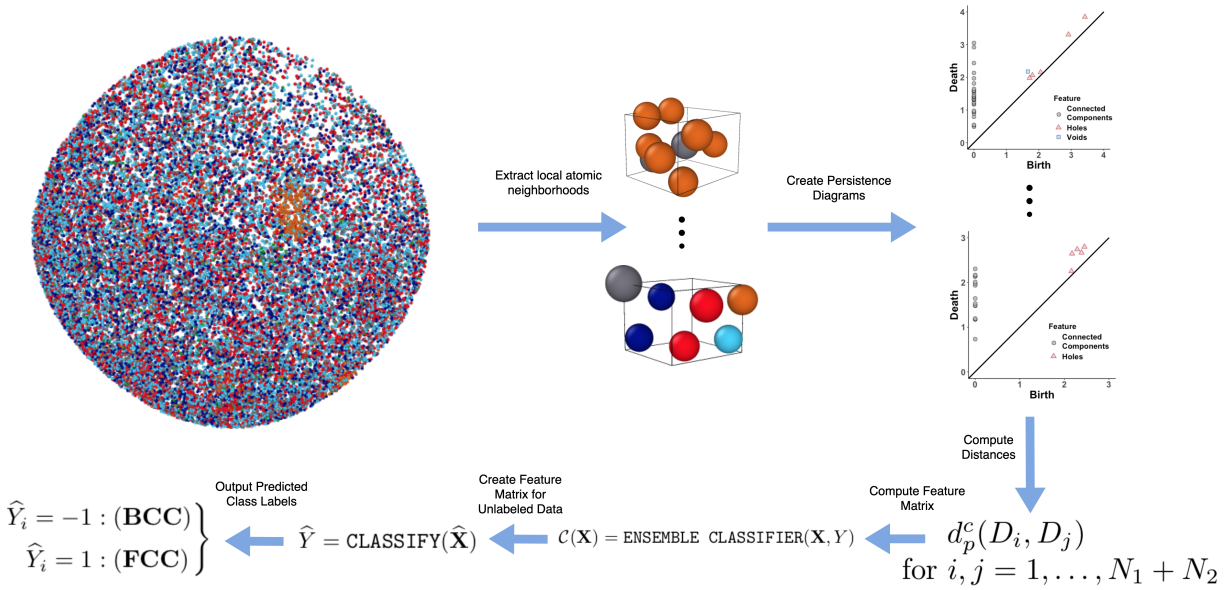


Figure 4.10: The materials fingerprinting methodology through which the APT data is processed. Individual atomic neighborhoods are extracted from an APT dataset. From these neighborhoods, we create a collection of persistence diagrams, each diagram associated with an atomic neighborhood. We then compute the d_p^c distance between all diagrams in the training set. We create a feature matrix composed of the summary statistics of these distances, which is used as input to the classification algorithm. The algorithm returns the class label for the atomic neighborhood associated to the persistence diagram D_i as either BCC or FCC in its structure, which is subsequently applied to new, unlabeled samples to automatically fingerprint them.

We first present our results on 200,000 atomic neighborhoods extracted from APT experiments. Then we conduct a sensitivity analysis of our method using synthetic datasets having varying percentages of sparsity, additive noise, or combinations of both. In these experiments with synthetic data, we compare our results using the same levels of experimental noise and sparsity as the neural net of [75] for our tests. In each of the experiments presented, we perform 10-fold cross validation on the entire dataset to control for overfitting of the model. For a schematic of our materials fingerprinting scheme, see Figure 4.10.

4.3.1 Numerical Results

We now turn to our original problem of determining the local lattice structure of an HEA from experimental APT data. Toward this end, we apply the materials fingerprinting method to APT experimental data from two HEAs, $Al_{1.3}CoCrCuFeNi$ and $Al_{0.3}CoCrFeNi$ (FCC). The former was previously studied through various diffraction techniques in [1] and was conclusively found through X-ray diffraction to have an FCC and BCC phase. The FCC phase is almost exclusively copper, so by excluding the copper-rich regions, we have an

alloy that is BCC in its structure. The latter HEA was also determined to be FCC through X-ray diffraction experiments.

As previously noted, these APT datasets are missing a significant proportion of the atoms from a sample, and the data that is recovered has spatial coordinates corrupted by additive noise. An example atomic neighborhood can be seen in Figure 4.1(c). Additionally, the sparsity is unevenly distributed throughout the sample [69]. Each atomic neighborhood contains a different number of atoms so that the overall effect is to have approximately 65% of the atoms missing, but consecutive neighborhoods contain different numbers of atoms. The challenge is to uncover the true atomic-level structure amid the noise and missing data, thus giving material scientists an unambiguous description of the lattice structure of these novel alloys. Using our materials fingerprinting methodology, we are able to classify the lattice structure of 200,000 atomic neighborhoods, split evenly between BCC and FCC lattice types, from these APT datasets at **99**% accuracy with 10-fold cross validation.

4.3.2 Sensitivity Analysis

In order to better understand the effect of differing levels of noise and sparsity in the data, the materials fingerprint was applied to synthetic data having different levels of sparsity or noise, and combinations thereof. Such computational experiments guide the development of the fingerprinting process since APT data has experimental noise and a significant percentage of the atoms missing. The synthetic datasets serve as a sensitivity analysis for the fingerprinting process and for determining optimal values of c and p, the two parameters of the d_p^c distance.

Fingerprinting was applied to a dataset of 10,000 synthetic crystal structures split evenly between BCC and FCC structures, varying individually the amount of additive noise or percentage sparsity in each instance. In all cases, the degree of accuracy was greater than **99%**. Accuracy scores reflect 10-fold cross validation for datasets with varying standard deviations of Gaussian additive noise, $\mathcal{N}(0, \sigma^2)$ (σ of 0.04, 0.06, 0.08 and 0.1 Å) or various percentages of sparsity (40%, 50%, 60% and 70%). Fingerprinting was also applied to synthetic datasets containing a combination of noise and sparsity (σ, sparsity) of (0.08, 60%), (0.1, 60%), (0.08, 70%) and (0.1, 70%). All accuracy scores exceed 99%.

Computational and Storage Considerations

Computing entries of the feature matrix $\mathbf{X}$, Equation (4.2.2), requires computing the mean and variance of d_p^c distances with $k-$dim persistence homology, $(k = 0, 1, 2)$. For each BCC persistence diagram, each $\mathbb{E}_{i,B}^k$ computation requires N_1 steps, while for FCC, it is N_2 steps. Similarly, computing the variance

accurately in a numerically stable fashion, e.g., when the size of the dataset is large and the variance is small, for each BCC diagram takes $2 \times N_1$ steps for the two pass algorithm [100]. In total, each row of $\mathbf{X}$ has complexity $O_i(N_1, N_2) = 9 \times (N_1 + N_2)$ and the entire feature matrix ends up with quadratic complexity: $O(N_1, N_2) = 9 \times (N_1 + N_2)^2$. With the atomic counts on the order of hundreds of thousands: $N_1, N_2 \approx O(10^5)$, the quadratic component clearly dominates with 10^{10} computational steps. Each of these steps requires the d_p^c distance computation given by Equation (2.2.1), which is computationally non-trivial for the majority of the diagrams due to the identification of the optimal permutation between the diagrams being compared. In order to reduce the total elapsed time of the computation, we used over 1000 x86 cores that ranged from Intel Westmere to Intel Skylake, ranging in cores per socket from 8 to 36 with up to 72 cores per node. Additional speedup of about 20% came from porting the code for computing the feature matrix from Python to C.

4.4 Variational Atomic Sequencing

Recall that our primary object of study is the atomic structure of HEAs, and our goal is to detect patterns from their noisy and sparse representation. From these patterns, we seek to derive meaningful statistical information about their composition and atomic-level structure. To infer the fundamental characteristics of these alloys, we define a transformation from the observed atoms to a static reference, which we have *a priori* determined through our materials fingerprinting process, see Section 4.1 and [92, 101].

Recall that the point set registration problem that seeks to spatially align two point clouds, called the reference and observed, and make point-wise correspondences between the two point sets. We choose to model each of these sets by a Gaussian Mixture Model (GMM), and pose the registration problem as one where we seek to minimize the distance between distributions, specifically two GMMS. Our formulation is not the first one to use GMMs for the point set registration problem [102, 103, 104, 105]. However, to our knowledge, we present the first proof of convergence for the point set registration problem using GMMs and give guidance on choosing a convergence criterion for the algorithm.

Our proposed algorithmic solution to this problem accounts for the points in our observation set being labeled by atomic type. This contrasts starkly with all other settings where the points are unlabeled, and any assignment is possible, but not necessarily favorable. The labeling of points in the reference lattice are precisely what we seek, and yields invaluable information to material science researchers. Thus we not only seek to align, and make point-wise correspondences between the point sets, we additionally seek the one yielding the lowest energy configuration of the atoms. In the present section, we present a statistical

formulation of the labeled point set registration problem, an algorithmic solution, and proof of convergence for our algorithm.

4.4.1 Statistical Model

Before presenting the details of our methodology, first recall our statistical model from Chapter 3. We assumed that

$$Y = C\mathcal{T}(x;\theta) + \Gamma, \tag{4.4.1}$$

for a rigid transformation $\mathcal{T}$ parameterized by θ for C, θ, and Γ independent. Here we relax the requirement that $\mathcal{T}$ is a rigid motion and assume that it describes some affine motion, which allows for scaling, shearing, and rotations. We also defined a matrix of correspondences $C \in \{0,1\}^{M\times N}$, such that $\sum_{k=1}^{N} C_{jk} = 1, 1 \leq j \leq M$, and each observation point corresponds to only one reference point. We now want to devise an algorithm to simultaneously infer both θ and C where previously C was not labeled, but now the points are labeled with their atomic identity. To do so, we will introduce a binary latent variable $z_m \in \{0,1\}^N$ that is a 1-of-N binary vector with elements z_{mn}, such that $\sum_{n=1}^{N} z_{mn} = 1$, for each $1 \leq m \leq M$ and make the assumption that both the reference and observed point sets may be represented by Gaussian mixture models.

Following these assumptions, we pose the point set registration problem as a minimization problem where we seek to minimize the the Kullback-Leibler divergence between two distributions, which is given by

$$\mathrm{KL}(\mathcal{P} \parallel \mathcal{Q}) = \int_{\Omega} p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x,$$

A GMM is written as a linear sum of say J Gaussian densities, each with their own mean, μ_j, and covariance, Λ_j.

$$p(x) = \sum_{j=1}^{J} \omega_j \mathcal{N}(x \mid \mu_j, \Lambda_j),$$

with mixing coefficient $\{\omega_j\}_{j=1}^{J}$, such that $0 \leq \omega_j \leq 1$ for all $1 \leq j \leq J$, and $\sum_{j=1}^{J} \omega_j = 1$.

In our labeled point set registration problem, we seek to spatially align and find the correspondence between two GMMs and label the points in the observation mixture model Y. To do this, for each observation Y_m, we introduce a corresponding latent variable $z_m \in \{0,1\}^N$ that is a 1-of-N binary vector with elements z_{mn} for each $1 \leq m \leq M$. This formulation leads us to consider a variational approach, as this methodology allows us to more readily explore different models and is well-suited to large data sets, as are typically retrieved from APT experiments.

Defining a joint distribution $p(Y, Z) = p(Y \mid Z)p(Z)$ and considering the marginal distribution over Z with respect to the mixing coefficients of the GMM we see that

$$p(z_m = 1) = \omega_m, \tag{4.4.2}$$

where $0 \leq \omega_m \leq 1$ and $\sum_{m=1}^{M} \omega_m = 1$, in order for the marginal, Equation (4.4.2), to be a valid probability. Thus we may write

$$p(Z) = \prod_{m=1}^{M} \omega_m^{z_m}, \tag{4.4.3}$$

and by the binary construction of z_m

$$p(Y \mid z_m = 1) = \mathcal{N}(Y \mid \mu_m, \Lambda_m),$$

which yields the conditional

$$p(Y \mid Z) = \prod_{m=1}^{M} \mathcal{N}(Y \mid \mu_m, \Lambda_m)^{z_m}. \tag{4.4.4}$$

Recall we have the observed data $Y = \{Y_1, \ldots, Y_M\}$ and to each $Y_m \in Y$ we associate a latent variable $Z_m \in Z$ where $Z = \{z_1, \ldots, z_M\}, Z \in \{0, 1\}^{M \times N}$. From these and an expectation-maximization procedure we may construct an explicit representation of the correspondence matrix. Assuming some prior information on the latent variables given by $p(z_{mn} = 1) = \omega_{mn}$ and for each individual observation $Y_m \in Y$ we have an associated mixing weight ω_m. Thus, the conditional probability of a matching, given the prior information $\boldsymbol{\omega}$ is

$$p(Z \mid \omega) = \prod_{m=1}^{M} \prod_{n=1}^{N} \omega_n^{z_{mn}} \tag{4.4.5}$$

and, suppressing the dependence on X, our likelihood model is

$$p(Y, Z \mid \mu, \Lambda) = \prod_{m=1}^{M} \prod_{n=1}^{N} \omega_n^{z_{mn}} \mathcal{N}(Y_m \mid \mu_n, \Lambda_n^{-1})^{z_{mn}} \tag{4.4.6}$$

where $\Lambda = \{\Lambda_i\}_{i=1}^{N}$ is the set of precision matrices of the Gaussian mixture model, and $\mu = \{\mu_i\}_{i=1}^{N}$ are the associated means. Also note that $\mu_i = [C\mathcal{T}(X; \theta)]_i$, which follows from our assumption that the reference is a GMM with components centered at the lattice points, and Y is a transformed version of the reference. We will show in the following section how we may explicitly construct C and an estimate of θ using a variational Bayesian approach.

4.5 Variational Posterior

We adopt a Bayesian viewpoint and incorporating the chemical ordering of the atoms in the neighborhood, denoted by λ, we want to sample from

$$p(Z, \omega, \mu, \Lambda, \lambda \mid X, Y) \propto \mathcal{L}(Y \mid X, Z, \mu, \Lambda, \lambda) p(\lambda \mid \omega, Z) p(Z \mid \omega) p(\omega) p(\mu \mid \Lambda) p(\Lambda). \tag{4.5.1}$$

Comparing Equation (4.5.1) with Equation (2.5.15), we see the pertinent difference is the inclusion here of the chemical ordering, i.e., the dependence on λ in the above posterior. To define this additional term accounting for the energy in a given configuration, we denote the set of individual elemental types present in a material by $\mathcal{A} = \{A, B, \dots\}$. Then $p(\lambda) \propto \exp\{-E(\lambda)\}$, where $E(\lambda(\mathcal{A})) = \sum_{A,B \in \mathcal{A}} w_{AB} p_{AB}$ is the Helmholtz free energy and $\mathcal{A}$ is the set of all atoms present in a neighborhood. In the energy function $E(\lambda(\mathcal{A}))$, w_{AB} is the interaction potential between atoms A and B, and p_{AB} denotes the probability of finding an AB elemental neighbor pair in the neighborhood. In the following derivation, we will simply write the chemical ordering as λ and suppress the $\mathcal{A}$ in the notation unless we want to make the functional dependence explicit.

4.5.1 Optimal $q(Z)$ (Variational E-Step)

We again make the standard variational Bayesian assumption on the form of our approximation, that we seek a density in the family of all densities $\mathcal{Q}$ that factor as $q(Z, \omega, \mu, \Lambda, \lambda) = q(Z) q(\omega, \mu, \Lambda, \lambda)$. Taking the log of the optimal distribution q^* yields

$$\begin{aligned} \log(q^*(Z)) &= \mathbb{E}_{\omega,\mu,\Lambda,\lambda}[\log(p(Z, \omega, \mu, \Lambda, \lambda \mid X, Y))] \\ \log(q^*(Z)) &= \sum_{m=1}^{M} \sum_{n=1}^{N} z_{mn} \log(\rho_{mn}) + \zeta, \end{aligned}$$

by incorporating all terms independent of Z into ζ, and by defining

$$\log(\rho_{mn}) := \mathbb{E}[\log(\omega_n)] + \frac{1}{2}\mathbb{E}[\log|\Lambda_n|] - \frac{d}{2}\log 2\pi - \frac{1}{2}\mathbb{E}[\|Y_m - \mathcal{T}(X_n; \theta)\|^2_{\Lambda_n}] - \mathbb{E}[\log(p(\lambda \mid \omega, Z))]. \tag{4.5.2}$$

We have previously evaluated a similar expression, Equation (2.5.19), which is identical save for the last term. This final expectation accounts for the elemental type of each atom in the configuration. Now the probability of finding an AB elemental pair as first neighbors is p_{AB} and we write the interaction potential between them as w_{AB}. From the statistical thermodynamic lattice model, i.e., the quasi-chemical

approximation [2], these terms are related by the expression

$$p_{AB} = \frac{1}{\mathcal{Z}} \exp\left\{-\frac{w_{AB}}{kT}\right\} = \frac{N_{AB}}{\sum\limits_{A,B\in\mathcal{A}} N_{AB}}, \tag{4.5.3}$$

for a given pressure k and temperature T, and $\mathcal{Z}$ is the partition function in the first equality. It is clear that we can estimate the interaction potential between atoms A and B by the relation

$$w_{AB} \propto -\log(N_{AB}), \tag{4.5.4}$$

assuming that the atomic compositions of A and B are equal, and we assume that $kT = 1$ without loss of generality. This approximation is hindered by the sparsity present in the APT data. We will discuss a method of successive approximation to refine our approximations of both the interaction potentials and the sufficient statistics of the GMM governing the atomic environment.

To evaluate this final expectation in Equation (4.5.2) and recalling that $z_{mn} = \{0, 1\}$

$$\mathbb{E}[\log(p(\lambda(\mathcal{A}) \mid \omega, Z))] = \mathbb{E}\left[\sum_{m,n=1}^{M,N} z_{mn} \sum_{\mathcal{A}} \log(p(\lambda(\mathcal{A}) \mid \omega, Z))\right], \tag{4.5.5}$$

$$= -\mathbb{E}\left[\sum_{m,n=1}^{M,N} z_{mn} \sum_{\mathcal{A}} E(\lambda(\mathcal{A}))\right] \tag{4.5.6}$$

We recognize Equation (4.5.6) as the expectation of the Helmholtz free energy, which is just the average internal energy of the system, given that the mn-th position in the lattice is occupied and its elemental type is known. This expectation can be easily evaluated via standard Monte Carlo integration techniques, i.e., Equation (2.3.1) or its variants [54].

4.5.2 Optimal $q(\omega, \mu, \Lambda, \lambda)$ (Variational M-Step)

Having defined the factor $q(Z)$, we now consider the remaining term $q(\omega, \mu, \Lambda, \lambda)$ in the variational approximation. Taking the log of the optimized distribution we have

$$\begin{aligned}\log q^*(\omega, \mu, \Lambda, \lambda) = &\sum_{i=1}^{N} \log(\mathbb{P}(\mu_i, \Lambda_i)) + \mathbb{E}[\log(\mathbb{P}(\lambda \mid Z, \omega))] + \log(\mathbb{P}(\omega)) + \\ &\sum_{i=1}^{N}\sum_{j=1}^{M} \mathbb{E}[c_{ji}] \log\left(\mathcal{N}(Y_j \mid \mu_i, \Lambda_i^{-1})\right) + \text{const.}\end{aligned} \tag{4.5.7}$$

Observing that the right-hand side of Equation (4.5.2) decomposes into terms with either ω, μ, or Λ, the variational posterior then factors as $q(\omega, \mu, \Lambda, \lambda) = q(\lambda \mid \omega)q(\omega)q(\mu \mid \Lambda)q(\Lambda)$. Furthermore, the expectations involving μ and Λ are composed of a sum over all N components of the reference, which implies that

$$q(\lambda, \omega, \mu, \Lambda) = q(\lambda \mid \omega)q(\omega) \prod_{i=1}^{N} q(\mu_i, \Lambda_i) \tag{4.5.8}$$

$$= q(\lambda \mid \omega)q(\omega) \prod_{i=1}^{N} q(\mu_i \mid \Lambda_i)q(\Lambda_i). \tag{4.5.9}$$

Parameter Update

To update the model parameters in the algorithm, we first compute the sufficient statistics of the observed data, given the correspondence. These are given by

$$M_i = \sum_{j=1}^{M} c_{ji} \tag{4.5.10}$$

$$\overline{Y}_i = \frac{1}{M_i} \sum_{j=1}^{M} c_{ji} Y_j \tag{4.5.11}$$

$$S_i = \frac{1}{M_i} \sum_{j=1}^{M} c_{ji} (Y_j - \bar{Y}_i)(Y_j - \bar{Y}_i)^T, \tag{4.5.12}$$

for $1 \leq i \leq N$. By our choice of conjugate prior densities, we find the update equations for each of the model parameters are given by

$$q^*(\mu_i, \Lambda_i) = \mathcal{N}(\mu_i \mid m_i, (\beta_i \Lambda_i)^{-1}) \mathcal{W}(\Lambda_i \mid W_i, \nu_i) \tag{4.5.13}$$

by the update formulas

$$\beta_i = \beta_0 + M_i \tag{4.5.14}$$

$$m_i = \frac{1}{\beta_i}(\beta_0 m_0 + M_i \overline{Y}_i) \tag{4.5.15}$$

$$W_i^{-1} = W_0^{-1} + M_i S_i + \frac{\beta_0 M_i}{\beta_0 + M_i} (\overline{Y}_i - m_0)(\overline{Y}_i - m_0)^T \tag{4.5.16}$$

$$\nu_i = \nu_0 + M_i, \tag{4.5.17}$$

where the pertinent parameters for the priors are as defined in Equations (2.5.12)–(2.5.14). Each of these update equations depends on the correspondence matrix C. Recalling Equation (4.5.2), we must compute the normalizing constant for the ρ_{mn}, and the individual terms are given below in Equations (4.5.18)–(4.5.20) and Equation (4.5.6).

$$\mathbb{E}[\|Y_m - \mathcal{T}(X_n;\theta)\|^2_{\Lambda_n}] = \frac{d}{\beta_n} + \nu_n (Y_m - X_n)^T W_n (Y_m - X_n) \tag{4.5.18}$$

$$\mathbb{E}[\log|\Lambda_n|] = \sum_{i=1}^{d} \left(\frac{\nu_n + 1 - i}{2} \right) + d\log(2) + \log|W_n| \tag{4.5.19}$$

$$\mathbb{E}[\log(\omega_n)] = \psi(\alpha_n) - \psi(\hat{\alpha}), \tag{4.5.20}$$

where $\psi(\cdot)$ is the digamma function, $\hat{\alpha} = \sum_{k=1} \alpha_k$, and Equations (4.5.19)–(4.5.20) follow from properties of the Wishart and Dirichlet distributions respectively [53, 59].

By construction of the variational posterior and priors distributions as given in Equations (4.4.3)–(4.4.4), the optimization involves iterating between finding the optimal correspondence matrix and evaluating the variational distribution over the model parameters. These iterations are equivalent to the maximum likelihood expectation-maximization (EM) algorithm [52, 55, 54] targeting our posterior Equation (4.5.1). We are now ready to present our variational Bayesian registration algorithm.

Algorithm 4.2 Variational Atomic Sequencing

Set initial values: $W_0 = \text{diag}(1,1,5), \beta_0 = 0.1, \eta_0 = \frac{1}{d}, m_0 = X$
for all $k > 0$ **do**
    E-Step: Compute correspondence matrix $C^{(k)}$
        $c_{mn} = \frac{\rho_{mn}}{\sum_{i=1}^{N} \rho_{mi}}$, where $\log \rho_{mn}$ is given in eq. (4.5.2).
    M-Step:
    Compute $\hat{N} = \mathbf{1}^T C^{(k)} \mathbf{1}, \mu_X = \frac{1}{\hat{N}} X^T (C^{(k)})^T \mathbf{1}, \mu_Y = \frac{1}{\hat{N}} Y^T C^{(k)} \mathbf{1}$
    $\hat{X} = X - \mathbf{1}\mu_X, \hat{Y} = Y - \mathbf{1}\mu_Y$
    Update means $\{m_i^{(k)}\}_{i=1}^{M}$
        Solve $A = (\hat{Y}^T C^{(k)} \hat{X})(\hat{X}^T \text{diag}(\mathbf{1}^T \cdot C^{(k)})\hat{X}))^{-1}$
        Set $m_i^{(k)} = \mathcal{T}^{-1}(\theta_i) = X_i A^T$
    Update Precisions $\{W_i^{(k)}\}_{i=1}^{M}$ per eq. (4.5.16)
    Update λ via simulated annealing to find $\text{argmin}_\lambda E(\lambda(\mathcal{A}))$
    Update model parameters and sufficient statistics per eqs. (4.5.10)–(4.5.17)
    Compute $d^{(k)} := \frac{1}{M^{(k)}} \sum_{j=1}^{M^{(k)}} \|\mathcal{T}^{-1}(C^{(k)^T} Y;\theta)_j - X_{i(j)}\|$, where $i(j)$ denotes that X_i matches Y_j
    if $d^{(k-1)} - d^{(k)} < \epsilon$ **then** Break
    end if
end for

In line 3 of our algorithm 4.2, we compute the correspondence matrix, given the current parameter estimates, and the parameters are updated in the M-step. We try and find the best affine transformation to fit the data, but if such a transformation is not invertible, i.e., $\texttt{rank}(D) < d$, where $UDV^H = \texttt{SVD}(A)$, then we instead find the best rigid rotation and translation. These parameters have an optimal closed-form solution given by [106] in the case where X and Y have the same number of points. If they do not, as is often the cases we consider with APT data, we use the correspondence matrix computed in the previous E-step to determine the point matching. Lastly, to find the lowest energy configuration in line 12, we employ the simulated annealing methodology of [107, 54], which is globally convergent under certain assumptions [108].

4.5.3 Convergence

While in general, as discussed in Section 2.5.1, checking for convergence involves maximizing a lower bound on the marginal likelihood, which is equivalent to minimizing the KL-divergence between our approximation and the target posterior. In our setting however, we can directly check for convergence by noting a few important facts. First, if one wants to measure the KL-divergence between two Gaussian densities, one only needs to consider the first two moments [109, 110]. In fact, [109] gives closed-from updates to the mean and variance to find the optimal parameters to minimize the KL-divergence. We can use this equivalence of minimizing the divergence to a moment matching problem to find the optimal parameters for our minimization process. However in the case of GMMs, this moment matching is not applicable, and the KL-divergence does not admit an analytical solution [111]. Although our model is a GMM, we can bypass lack of a closed-form solution by noting that the correspondence matrix C assigns each observed density to a corresponding one in the reference. Thus we may equivalently match moments between associated densities as a test of convergence. This does not alleviate all issues however, while the means of the GMM are the points themselves, the covariance matrices remain unknown, and the closed form solutions for the optimal values remain intractable.

We will now prove a convergence theorem for our method.

Theorem 4.7. *The variational point-set registration algorithm, Algorithm 4.2, converges montonically, as measured by the Kullback-Leiber divergence, to a global minimum.*

Proof. Given the reference $\{X_i\}_{i=1}^N = X, X_i \in \mathbb{R}^d$ and observation $\{Y_i\}_{i=1}^M = Y, Y_i \in \mathbb{R}^d, M \leq N$ GMMs, compute an initial correspondence matrix $C^{(0)} \in \{0,1\}^{M\times N}$. Denote $X_{i(j)}$ as the point in X matched to $Y_j \in Y$. Write $\{\mu_i\}_{i=1}^N$, and $\{\Lambda_i\}_{i=1}^N$ as the set of means and variances of the GMM for the reference. Similarly

write $\{m_i^{(k)}\}_{i=1}^{M}$, and $\{S_i^{(k)}\}_{i=1}^{M}$ as the set of estimated means and variances for the observed GMM at iteration k. Define the error between approximated and ground-truth moments at iteration k as

$$e^{(k)} := \max\left\{ \frac{1}{M^{(k)}} \sum_{j=1}^{M^{(k)}} \|m_j^{(k)} - \mu_{i(j)}\|, \frac{1}{M^{(k)}} \sum_{j=1}^{M^{(k)}} \|W_j^{(k)} - \Lambda_{i(j)}\| \right\},$$

where $M^{(k)} = \mathbf{1}^T \cdot C^{(k)} \cdot \mathbf{1}$. Lastly, define the mean-square error of the transformation as

$$d^{(k)} := \frac{1}{M^{(k)}} \sum_{j=1}^{M^{(k)}} \|[\mathcal{T}^{-1}(C^{(k)})^T Y; \theta^{(k)}]_j - X_{i(j)}\|.$$

Now it is the case that $d^{(k)} \leq e^{(k)}, \forall k$. To see this, suppose that

$$\begin{aligned} e^{(k)} =& \frac{1}{M^{(k+1)}} \sum_{j=1}^{M^{(k+1)}} \|W_j^{(k)} - \Lambda_{i(j)}\| \\ \geq& \frac{1}{M^{(k)}} \sum_{j=1}^{M^{(k)}} \|m_j^{(k)} - \mu_{i(j)}\| \\ =& d^{(k)}, \end{aligned}$$

where the last line follows from the assumption that X is a GMM with components at each $X_i \in X$. If it is the case that $e^{(k)} = \frac{1}{M^{(k)}} \sum_{j=1}^{M^{(k)}} \|m_j^{(k)} - \mu_{i(j)}\|$, then equality holds.

We claim that $d^{(k+1)} \leq e^{(k+1)} \leq d^{(k)} \leq e^{(k)}$. At each iteration the correspondence matrix C and transformation parameters are updated, each yielding the conditional distribution for the latent variables, in the E-step, or the maximum likelihood estimate, in the M-step. Now let $m^{(k+1)} = T(X; \theta^{k+1})$ and if $C^{(k+1)} = C^{(k)}$, then $d^{(k+1)} = d^{(k)}$, as the transformation parameters θ are maximized at each M-step of algorithm. We also obtain an update $m^{(k+1)}$, and it then follows that

$$\|\mu_{i(j)} - m_j^{(k+1)}\| \leq \|\mu_{i(j)} - m_j^{(k)}\| \, \forall j \tag{4.5.21}$$

and that

$$\|\Lambda_{i(j)} - W_j^{(k+1)}\| \leq \|\Lambda_{i(j)} - W_j^{(k)}\| \, \forall j \tag{4.5.22}$$

as the algorithm monotonically decreases the KL-divergence, which is equivalent to moment matching. Now it is clear that if $e^{(k+1)}$ is the difference between means, that $e^{(k+1)} \leq d^{(k)}$. Consider the case where $e^{(k+1)}$ is the difference between precision matrices. Then since $e^{(k+1)} \leq e^{(k)} = d^{(k)}$, the error sequences must obey

the desired inequality

$$0 \leq d^{(k+1)} \leq e^{(k+1)} \leq d^{(k)} \leq e^{(k)} \tag{4.5.23}$$

for all iterations k. The lower bound occurs since norms are positive semi-definite operators, and the algorithm converges monotonically to the global minimum. □

Iterating through our methodology not only yields the lowest energy configuration of the observed atoms, and the best alignment between the observation and reference point sets, but we can extract additional information as well. If we recall Equation (4.5.3)

$$p_{AB} = \frac{1}{\mathcal{Z}} \exp\left\{-\frac{w_{AB}}{kT}\right\} = \frac{N_{AB}}{\sum\limits_{A,B \in \mathcal{A}} N_{AB}},$$

from this relationship we can extract an approximation of the interaction potential as well. That is,

$$w_{AB} \propto -\log(N_{AB}), \tag{4.5.24}$$

assuming elements A and B have the same compositional proportion. Now this potential w_{AB} is a global parameter governing the material, and due to the sparsity present in the data, we cannot find an accurate value by examining a single configuration. Instead, we compute a single iteration of Algorithm 4.2, and find the average pair-wise composition of each neighborhood. We then refine this average composition in subsequent iterations for computing the lowest energy configuration of each observed point set.

4.5.4 Estimating Interaction Potentials

In addition to inferring the occupied positions in the lattice and the lattice spacing, we can gain additional information about the material in question by considering a quasi-chemical approximation for a lattice model. Recalling Equation (4.5.4), we see that the neighborhood composition is proportional to the negative log of the pairwise interaction between atoms present in a neighborhood. The interaction potential w_{AB} is a global parameter that governs how atoms in the entire HEA sample interact, and is directly related to the unknown chemical ordering. This estimate is affected by the sparsity present in the data, and by taking a sufficiently large sample, we may recover a sufficiently accurate estimate of this global parameter. To approximate these parameters we employ an iterative approximation method. To be concrete, we iterate through the entire dataset, computing one iteration of Algorithm 4.2 for each atomic neighborhood. We then find the average number of pairs in each neighborhood, and from this average, we estimate the interaction potentials according

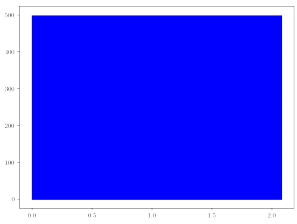
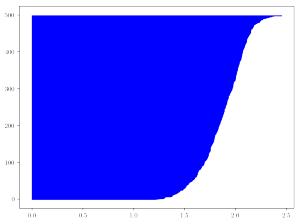
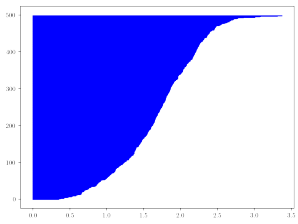
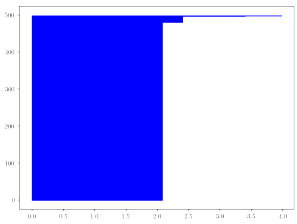
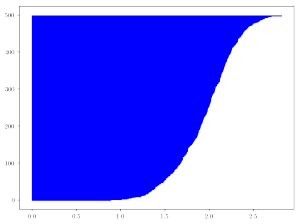
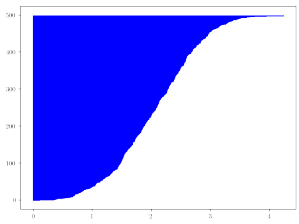
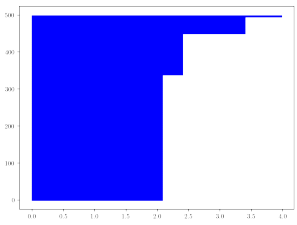
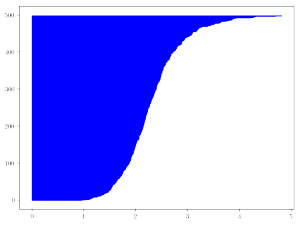
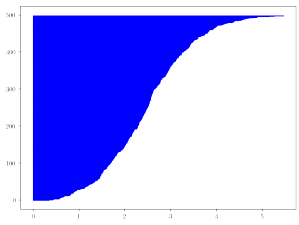
	$\mathcal{N}(0,0)$	$\mathcal{N}(0,0.25^2)$	$\mathcal{N}(0,1^2)$
0%			
33%			
67%			

Table 4.2: H_0 analysis of varying levels of sparsity and Gaussian noise to inform our choice of radii in the atomic neighborhoods we consider.

to Equation (4.5.4). Continuing in the same fashion, we loop over the entire data set to find the composition and neighbor types present in each configuration. We then update average composition of each neighborhood and use this empirical mean to estimate the unknown potentials. We will iterate this local-to-global process until convergence of both the interaction potentials and the labeled registration process.

4.6 Variational Atomic Sequencing Numerics

In this section we will present the results of numerical experiments using our methodology for synthetic APT data. Primarily, to verify the correctness of our method, observe the neighbor analysis in Figure 4.11.

4.6.1 Atomic Neighborhood Volume

Before proceeding with results from our algorithm, there is a parameter that impacts all aspects of our variational method that requires more discussion. We choose all points within some radius for the atomic neighborhoods that we map onto a reference lattice and find the chemical ordering. This parameter did

not impact the results in previous sections since we did not consider elemental type of the atoms, only their geometric coordinates. Indeed, we proposed a Bayesian formulation of the point-set registration problem Chapter 3 and developed a topologically-informed classification algorithm Section 4.1, neither of which considered elemental type. In both of these settings, the volume of these atomic neighborhoods was selected by the 'Goldilocks' criteria: to be large enough to yield sufficient information about the material, but not so large as to be computationally inefficient. In the present section, we would like these neighborhoods to capture first and second neighbor relationships, but not more. Recalling the fact that the neighborhood's volume ha a direct relationship on the interaction potential estimate Equation (4.5.3). Recall that this relationship from statistical mechanics implies

$$w_{AB} \propto -\log(N_{AB}). \tag{4.6.1}$$

From the above relation, we can clearly see how the interaction potential between two atoms of type A and B is a function of the number of atoms in a neighborhood. What we can see in the barcode plots of the H_0 features Table 4.2. The longest barcode at the top of each plot may be safely ignored, as it persists over all length scales and describes the connectedness of the complete simplicial complex, i.e., the complex formed when all data points are connected.

Recalling the definitions and notions of TDA from Section 2.2, the H_0 barcodes yield the connectivity of the 0-dim features, the connected components, which are the atoms themselves. This multiscale analysis provides a topological signature of the material which encodes the homology of the neighborhood for all ϵ-balls created by the Vietoris-Rips complex over all values of ϵ. As the radii of the balls increases and the balls intersect, this death time is noted and the associated barcode ceases to increase. Hence, we may quantify connections between the atoms, and begin to understand neighbor relationships. The plots in Table 4.2 show the connectivity of the datasets, and from which we may see first and second neighbor relationships. These are clearly seen in the first column, which has no noise added to the atoms. In the plots with 33% and 67% sparsity, we can clearly see the first, second, and third neighbor distances between the atoms in a material. These relationships are obfuscated by the noise present in the other columns where we have added some Gaussian noise to the points. Notice is that the lattice parameter, 2.4Å, is revealed by looking at the second neighbor distances in the first column plots with either 33% or 67% sparsity. Also, note that the distances between points increases as the noise increases for a fixed sparsity. This topological perspective guides our selection of radii of the atomic neighborhoods that we map onto the reference lattice. If we choose too large

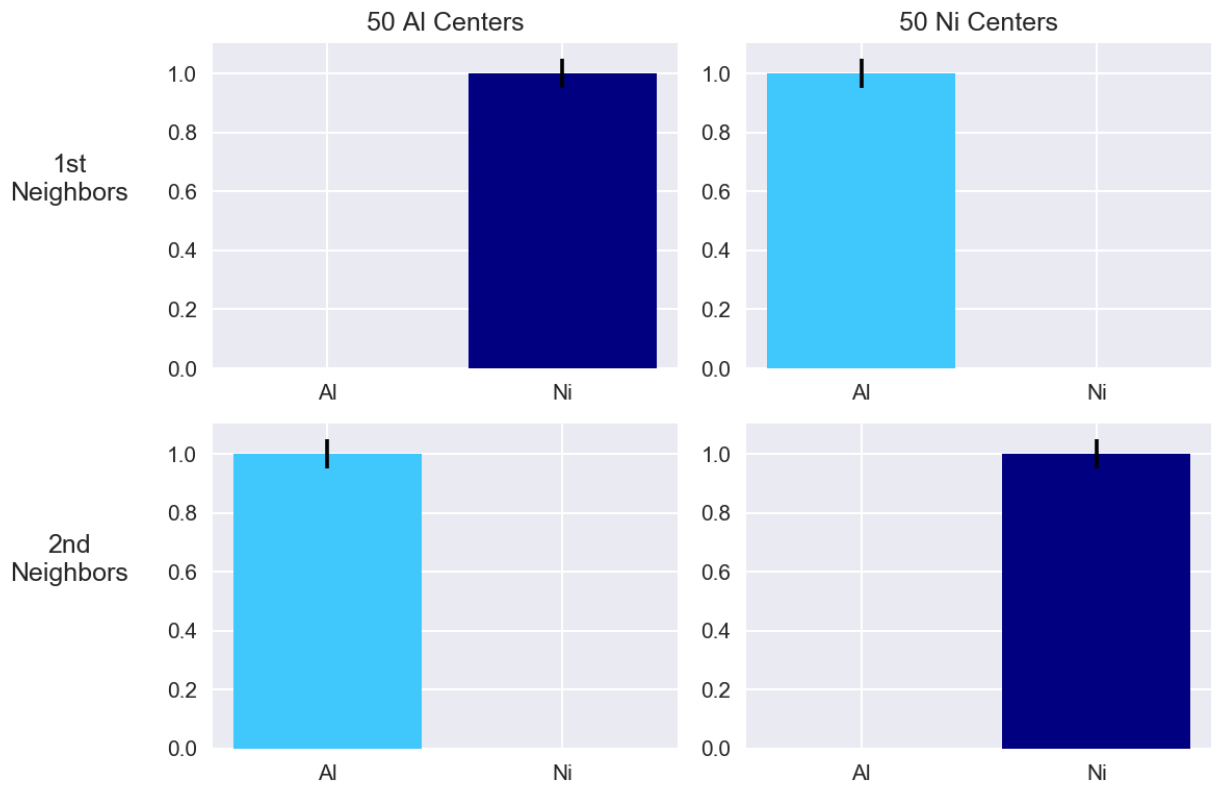


Figure 4.11: Variational inference applied to synthetic APT data of a binary NiAl alloy. The process is able to recover the chemical ordering, i.e., the aluminium center has only nickel first neighbors and aluminium second neighbors, and vise versa when nickel is the center atom.

a volume, we will be unable to map, in a one-to-one fashion, onto the reference, and conversely, if there are too few, we will not have enough information for the interaction potential calculation.

4.6.2 Sensitivity Analysis

These results for the sensitivity analysis were obtained from our method using chemically ordered complete, noiseless data. By chemically ordered we mean that each neighborhood with aluminium at its center has only nickel first neighbors and aluminium second neighbors, see Figure 4.12 for a visual representation. It follows then that the neighbor relationships are reversed if nickel is at the center of the neighborhood. This relationship is perfectly preserved in the case of an aluminium or nickel center, see Figure 4.11.

We now consider another asymptotic case, one in which elemental ordering is *not* present. This data was originally ordered BCC, as in Figure 4.11, but importantly, we set all interaction potentials to zero, for the entire simulation, indicating no preference for either element between neighbor shells. Upon first glance, one might expect the probabilities all to equal 0.5 in Figure 4.13. This is reasonable, *if the number*

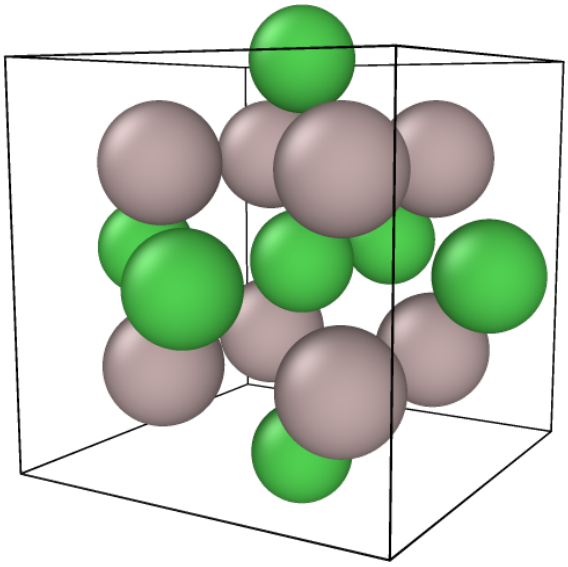


Figure 4.12: Example of a chemically ordered BCC lattice used in our sensitivity analysis, where color denotes elemental type. The first neighbors of the center atom are exclusively the other type, whereas the second neighbors are of the same species.

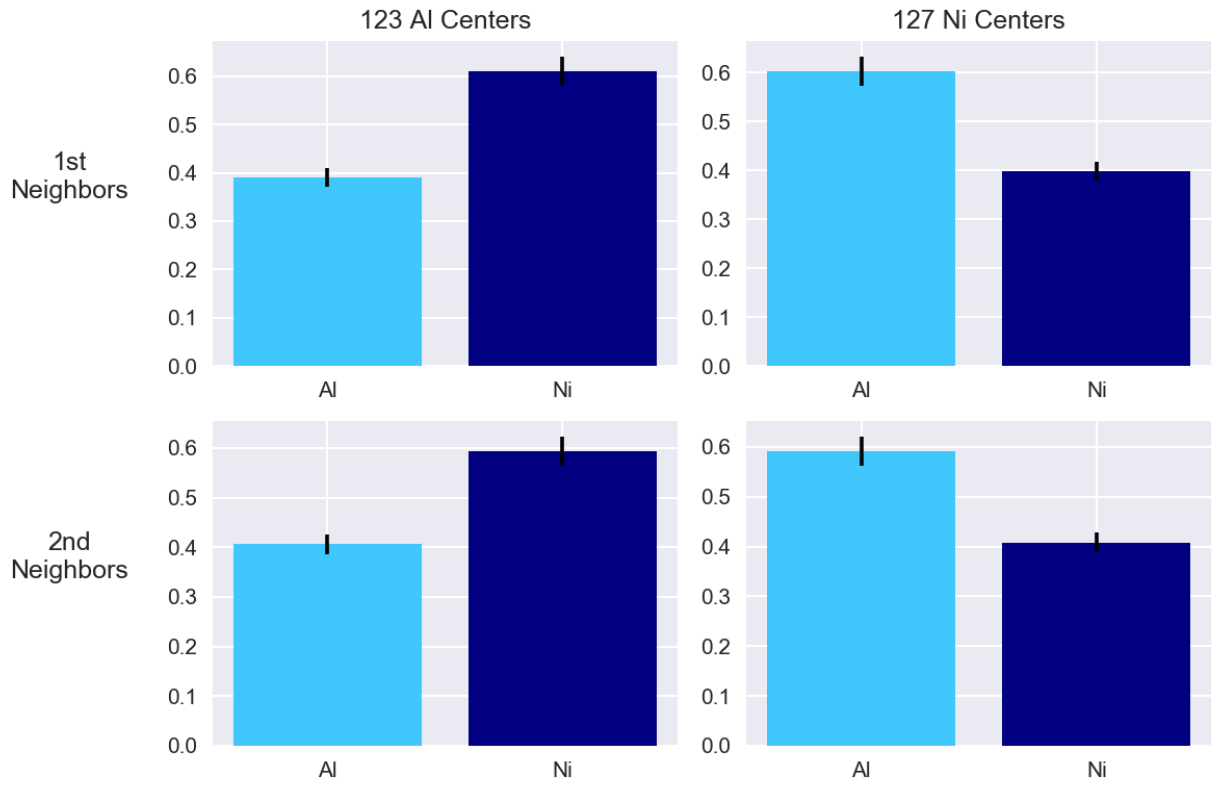


Figure 4.13: Variational inference applied to synthetic APT data of a binary NiAl alloy, *without* preferential ordering. As expected, the process is not able to recover the chemical ordering as in Figure 4.11, and the elemental distribution is random. The total number of atoms, not counting the center, in the aluminium neighborhoods is 1551: 615 aluminium and 936 nickel. Similarly the nickel neighborhoods contained 1564 atoms, 935 were aluminium a and 629 were nickel.

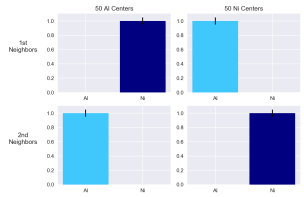
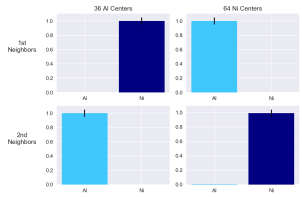
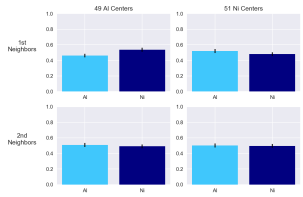
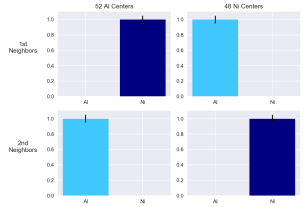
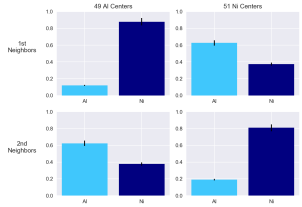
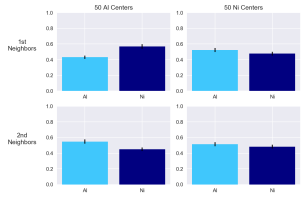
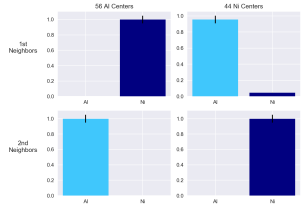
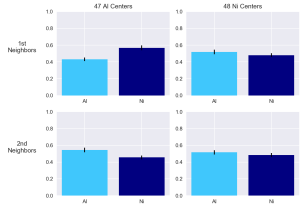
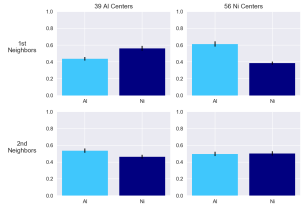
	$\mathcal{N}(0,0)$	$\mathcal{N}(0,0.25^2)$	$\mathcal{N}(0,1^2)$
0%			
33%			
67%			

Table 4.3: Neighbor analysis with varying levels of sparsity and Gaussian noise.

of atoms in the neighborhood are equal. Since the data is initially ordered, we must consider the atomic proportions in each neighborhood. Considering only the first and second neighbors, of which there are 8 and 6 respectively, per BCC unit cell, we immediately see that if we have an aluminium atom at the center, there are then 8 nickel atoms and 6 additional aluminium in the neighborhood. In our process, we do not perturb the center atom in each neighborhood, and so there are 14 atoms to vary. Consequently, if we assume perfect mixing and no elemental preference, there is a $4/7 \approx 58\%$ chance of finding an atom of the opposite type, opposite of the center atom, at either the first or second neighbor positions and a $3/7 \approx 42\%$ chance of finding a like type as either a first or second neighbor. Our empirical results agree with these percentages. We found that the aluminium neighborhoods were approximately 40% aluminium and 60% nickel, and the nickel neighborhoods we considered are approximately 59% aluminium and 41% nickel. Comparing these compositional proportions with the histograms in Figure 4.13, we see that our method does not impose chemical ordering if none exists.

Proceeding now with synthetic data more representative of that retrieved from an APT experiment, we see in the second and third columns of Table 4.3 that the process is in fact able to infer the chemical ordering from the noisy and sparse dataset. Two important issues require comment at this juncture. Primarily, this result is *not* by chance, as multiple runs with the same synthetic data yielded a similar plot, showing evidence

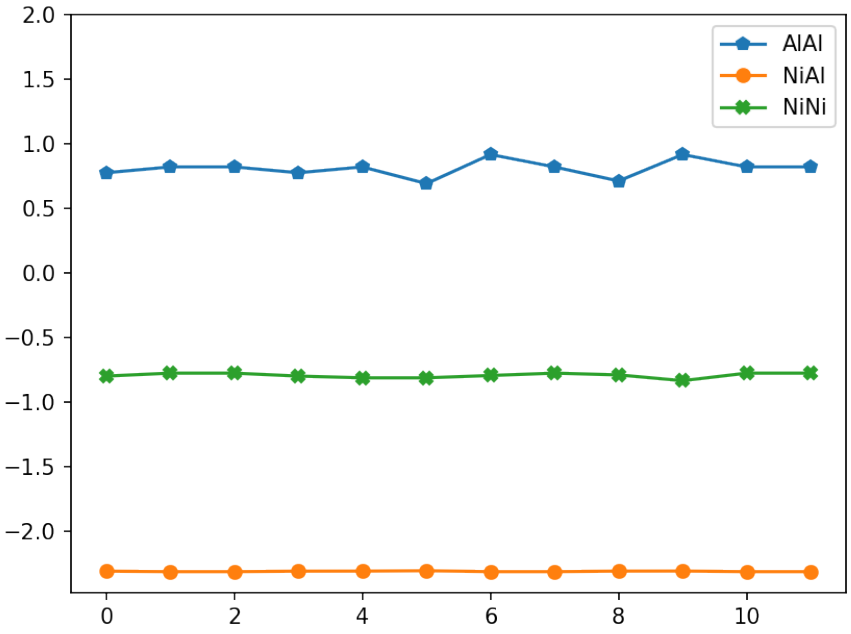


Figure 4.14: Extracted interaction potentials from our method applied to synthetic data with 33% missing and $\mathcal{N}(0, 0.25^2)$ added noise.

.

of a chemically ordered alloy. Secondly, if we compare Figure 4.11 with Figure 4.13, it is immediately apparent that the result in Figure 4.13 yields an incorrect result. This plot was generated from the same synthetic data, but in our algorithm, we did not account for elemental type. To be explicit, we omitted the final expectation in Equation (4.5.1) and Line 12 of Algorithm 4.2.

In Figure 4.15 we see a convergence plot detailing the mean-squared error, mean-squared difference between precision matrices, and energy calculation. We observe for the mean difference between precision

	Mean	Variance
Lattice parameter	3.041262	0.357354
Initial MSE	1.067046	0.029515
Final MSE	0.998185	0.023943
Error reduction	0.068861	0.017893

Table 4.4: Error Statistics, synthetic APT data with 33% missing and $\mathcal{N}(0, 0.25^2)$ added noise, for 250 neighborhoods. The true lattice parameter is 2.8.

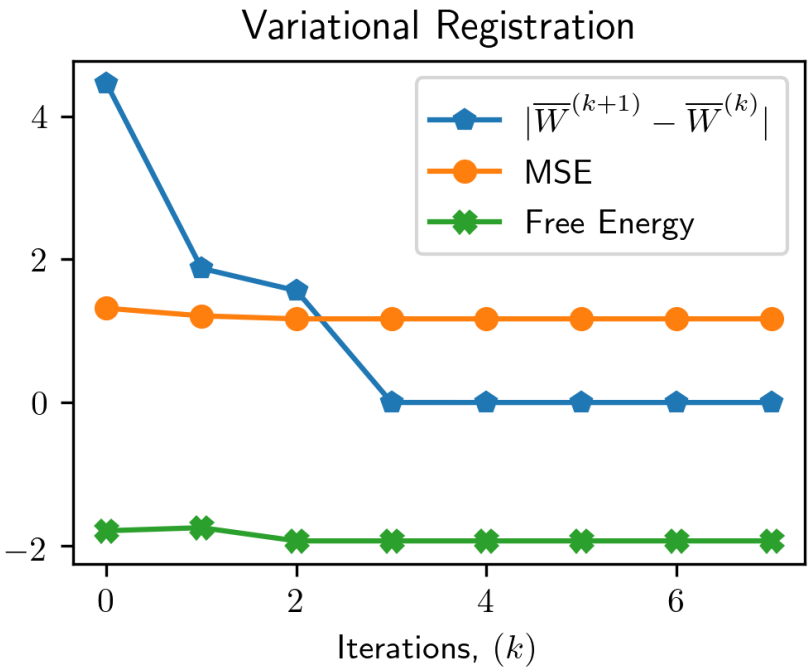


Figure 4.15: Variational inference applied to synthetic data with 33% missing and $\mathcal{N}(0, 0.25^2)$ added noise.

matrices and mean-squared error, that we have a decreasing sequence, as predicted by Section 4.5.3. Furthermore, the energy is decreasing as well, as expected. A plot of the estimated interaction potentials is shown in Figure 4.14. Summary statistics of our methodology are presented in Table 4.4. as applied to 250 atomic neighborhoods that compose a subset of synthetic data with 33% missing and have $\mathcal{N}(0, 0.25^2)$ added to each point. We can clearly see here the average decrease in mean-squared error and that the methodology inferred the lattice parameter to less than one standard deviation.

4.6.3 Real APT Data

We apply our variational Bayesian methodology both with and without consideration of chemical ordering on two different datasets and comment on the results.

If we first apply the variational registration methodology, as described in Section 2.5, without consideration of chemical ordering to the 100,000 atomic neighborhoods extracted from the multi-component alloy Al_1.3CoCrCuFeNi. We chose this alloy as it has been well-characterized in previous studies [1], and has some interesting features that we seek to infer from the data. The results of our variational registration and corresponding neighbor analysis is shown in Figure 4.16. We see that the bulk stoichiometry is preserved

	Mean	Variance
Lattice parameter	3.682982	0.4935562
Initial MSE	1.59352	0.114241
Final MSE	0.910836	0.290026
Error reduction	0.682685	0.353127

Table 4.5: Error Statistics, real APT data.

and some larger-scale trends can be seen. As previously analyzed in [1], the copper has almost entirely accumulated in an FCC phase, we extracted the BCC phase for this study, and we see the presence of a chromium-iron preference. The previous work [1] corroborates these findings, but our methodology is unable to see finer details of the chemical arrangements due to the noise and sparsity of the data. We find that the lattice parameter is 3.17Å with a standard deviation of 0.2565, which has not been previously reported.

Lastly, we applied our method, including elemental type, to APT data that is a chemically ordered Ni_3Al FCC alloy. In choosing the radii for our atomic neighborhood, we again considered the connectivity of the H_0 barcodes, shown in Figure 4.17. The barcode plot here looks very similar to the synthetic data case of 67% missing and $\mathcal{N}(0,1)$ added noise, so we chose a similar radius as in the synthetic data test. The true lattice parameter for this alloy is 3.58Å [112].

The Ni_3Al alloy analyzed in Figure 4.18 has no aluminium first neighbors with aluminium at the center, and should have 33% aluminium first neighbors with nickel at the center.

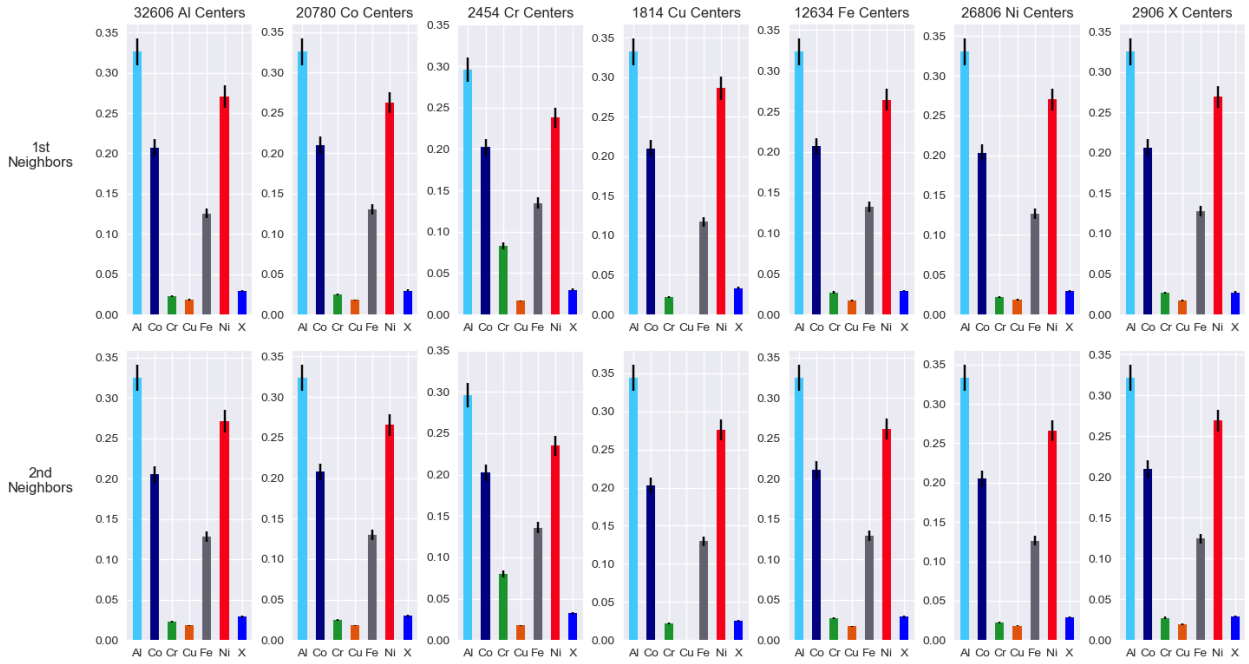


Figure 4.16: Neighbor analysis of the BCC phases in the multi-component alloy $Al_1.3CoCrCuFeNi$[1] considering only the geometry of the two point sets.

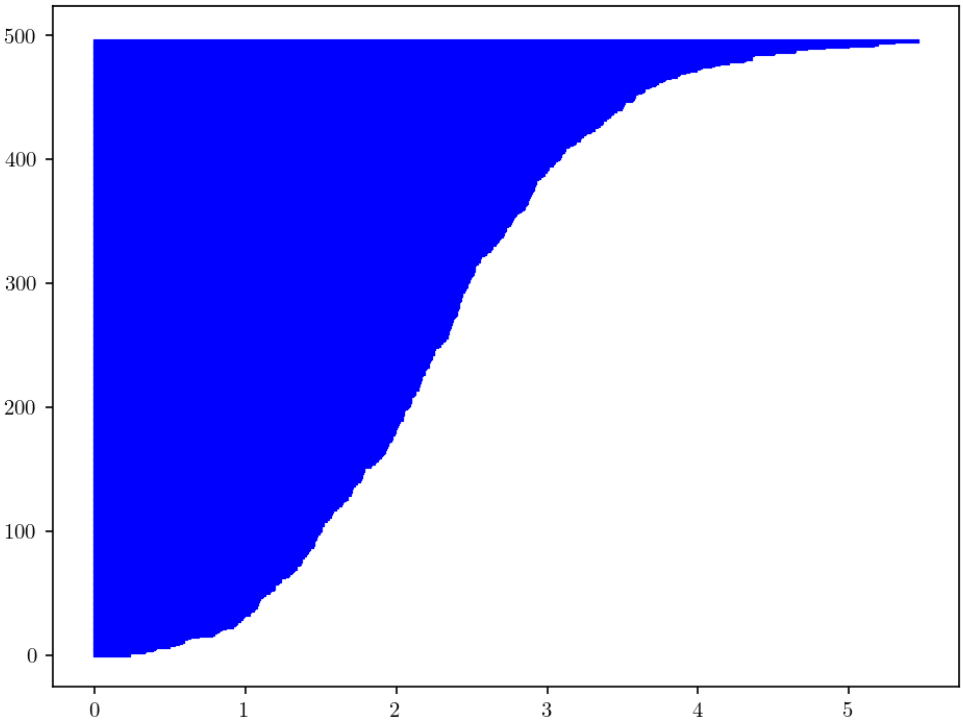


Figure 4.17: H_0 homology of real APT data.

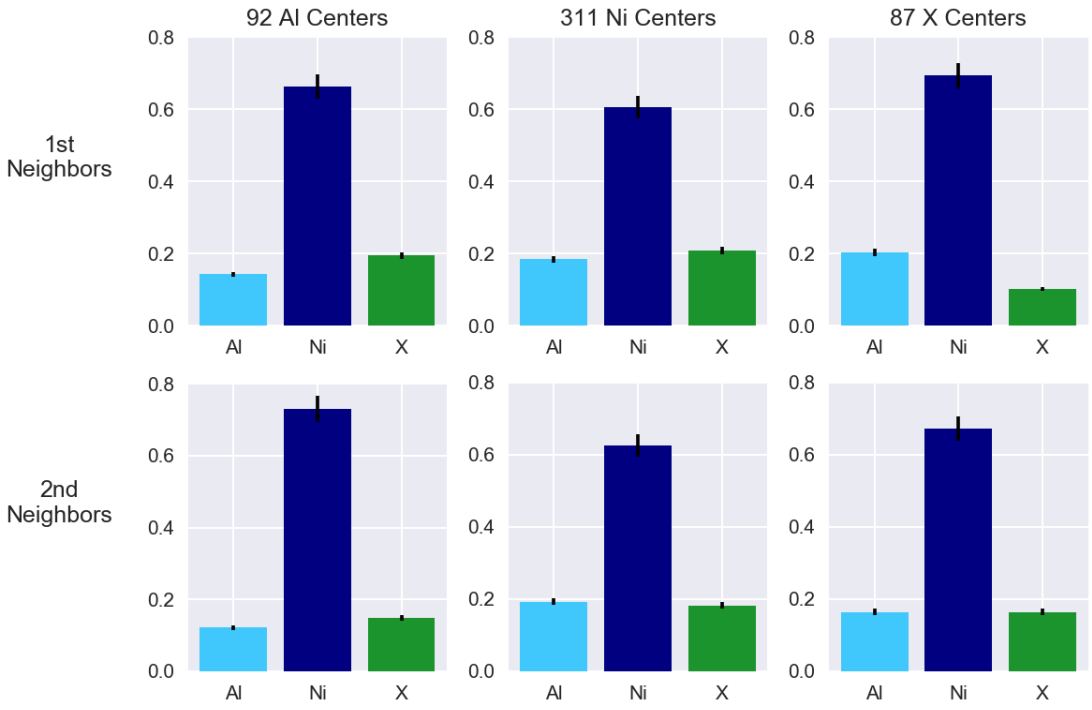


Figure 4.18: Variational inference applied to real APT data of a Ni_3Al alloy that exhibits preferential ordering. The method is unable to recover the ordering due to the noise and sparsity of the data.

Chapter 5

Conclusions

In this final chapter, we give a summary of the previous chapters to place it in context of our motivating problem. We conclude and present directions for further research.

5.1 Chapter Summaries

5.1.1 Introduction

In Chapter 1, we presented our motivation for developing the methodologies presented herein. Our goal is to discover the nanoscale atomic structure of a disordered material given a noisy and sparse representation, such as is typically collected via atomic probe tomography experiments. This process yields geometric coordinates in $\mathbb{R}^3$ in addition to elemental type for each atom registered by the detector. From these datasets large scale trends, such as areas of elemental accumulation may be seen. However, in order for materials scientists to accurately make structure-property predictions, further investigation and analysis are required. Specifically, we seek to infer the presence, or absence, of any chemical ordering, the crystal lattice type, and its spacing. From such fundamental information about a material, materials scientists may use these experimentally-validated structures to develop the unknown interaction potentials for multi-component systems as input to molecular dynamics simulations, which in turn lead to unique insight into the functional mechanisms in these materials.

Chapter 2 gives general mathematical details about the methods we used in our analysis of the APT data. We discuss the fundamentals of MCMC sampling and variational Bayesian methods for Gaussian mixture models. We give the necessary definitions from topological data analysis, and background material for the classification scheme used in our materials fingerprinting methodology. As presented, these definitions and

notions are somewhat general, and give only the necessary background for our specific methodologies presented in subsequent chapters. The interested reader is referred to the references provided in the appropriate sections for further investigation.

5.1.2 Known Reference

Our statistical methodology for the case of a known reference is presented in Chapter 3, where our goal is to find the optimal mapping between noisy observations from an APT experiment and a known reference lattice structure. In this section, we derive a Bayesian formulation of the classical point set registration problem, and detail a Markov chain Monte Carlo (MCMC) sampling scheme to find the optimal transformation, as measured by the mean-squared distance, mapping the reference onto the observed points. We employed a gradient-based MCMC sampling scheme to efficiently locate the mode of our target distribution, which by construction, is the global minimizer of our objective function. We are able to recover a good estimate of the correspondence and spatial alignment between synthetic materials datasets despite missing data and added noise. As a continuation of this work, we could extend the Bayesian framework presented in section to incorporate the case of an unknown reference. In such a setting, we would seek not only the correct spatial alignment and correspondence, but the reference point set as well. The efficiency of our algorithm could be improved through a tempering scheme, allowing for easier transitions between modes, or an adaptive HMC scheme, where the chain learns about the sample space in order to make more efficient moves.

In the formulation presented in Chapter 3, we did not consider elemental type, nor did we explicitly construct the correspondence matrix between the observed and reference point sets. Rather, it was inferred through the point set registration process. We provided ideas for how it may be constructed, via an assignment algorithm or a closest point process, if an explicit form is needed for further analysis.

5.1.3 Unknown Reference

The Markov chain Monte Carlo based approach of Chapter 3 requires *a priori* knowledge of the reference structure. While such information may be gained via X-ray diffraction or neutron scattering experiments and their subsequent analysis, the datasets provided by APT contain more information than an XRD experiment, that may be leveraged into further insights. From these information-rich materials descriptors that we may construct a richer picture of the nanoscale structure of a material. We explored the idea of employing the geometry of a crystal lattice to create a topologically-informed machine learning approach to infer the true crystal lattice structure directly from the APT data.

We have developed an automated methodology for classifying the crystal structure of the HEA APT data with near-perfect accuracy. Starting from a collection of atomic neighborhoods generated by an APT experiment, from which extract atomic neighborhoods, i.e., atoms within a fixed volume forming a point cloud, and apply the machinery of topological data analysis to these point clouds. TDA extracts the fundamental topology of the structure and record the information in a persistence diagram. These diagrams succinctly encode the essential topology of an atomic neighborhood over different length scales in different dimensions. It is by computing the persistent homology of the data that we are able to see through the noise and fill in the sparsity to see where these lattice structures are connected and where they are not. Basing our materials fingerprint on topological features such as connected components, holes, and voids, in conjunction with the number of atoms in each neighborhood, we represent the essential topological and numeric information necessary to differentiate between the lattice structures considered here, with the appropriate choice of metric.

Our machine learning methodology leverages the fundamental differences between the primary building blocks of high-entropy alloys: the topology and cardinality of their unit cells. We use a distance on the space of persistence diagrams that incorporates both of these elements, specifically the d_p^c metric. We proved a stability result for this metric, implying continuity of the Vietoris-Rips complex, which describes the mapping from point cloud to persistence diagram. Our materials fingerprinting methodology uses the mean and variance of the d_p^c distance between persistence diagrams derived from both body-centered and face-centered cubic lattices as to create features for a classification algorithm input for a machine learning algorithm. This distance not only measures differences in the diagrams but accounts for different numbers of points between diagrams being compared. This latter point is salient, as BCC and FCC unit cells each contain a different number of atoms, and this distinction must be taken into account. Using this distance, we are able to classify persistence diagrams as being created from either BCC or FCC lattice structures with better than 93% accuracy, considering both synthetic and real APT data.

Lastly, in Section 4.4, we seek to infer not only the mapping, including any scaling, of the noisy APT data onto the reference lattice, but the correspondence between points as well. Into this correspondence matrix we include the elemental type of each atom, information we have until this point not used. We construct the correspondence between the point sets by introducing a latent variable into our model, then take the marginal over this variable to recover the correspondence between the point sets. From this labeled correspondence matrix, we seek to infer the presence of any short-range chemical ordering within the constituent elements of the alloy.

Adopting a Bayesian perspective, we view the inference problem as one of minimizing the relative entropy between two Gaussian mixture models, the reference and the observation, the latter of which we assume to be a transformed version of the reference. The restriction imposed in Chapter 3 that we assume the transformation to be a rigid rotation and translation is relaxed to allow for affine transformations, thus allowing for more flexibility to find the best mapping, in the mean-squared error sense, between the point sets. We construct a novel algorithm for this inference problem, and provide the first proof of convergence to our knowledge for this setting of the point set registration algorithm. Additionally, we construct an explicit representation of the correspondence matrix between the reference and observation that accounts for the elemental types present in the data.

Through this construction, we define a distribution over the lattice scaling between point sets. This distribution is pertinent, as HEAs do not enjoy a uniform lattice spacing due to the distribution of atoms throughout the material, thus giving rise to the deformed lattice structure. Lastly, we are able to self-consistently identify the ordering of atoms, despite the noisy and sparsity present in the data. Identification of such ordering leads directly to pairwise interaction potentials from the quasi-chemical lattice model in statistical thermodynamics. To our knowledge, no such previous approximations have been reported for HEAs.

5.2 Conclusions

We have developed novel statistical methodologies and implementations to extract fundamental information from APT data that had not been previously reported. Relying on the methodologies presented herein, one may infer the crystal structure, lattice parameter, first and second neighbors of an atomic neighborhood, and identify short-range ordering of the component elements in a material. Being able to infer such fundamental information about a material provides materials science researchers with an invaluable tool to bridge the gap between theory and experiment. While our primary object of study is APT data of HEAs, our methodologies are not specific to these materials. They may be applied to any material open to characterization via APT, such as entropy-stabilized oxides [19].

Our methodology not only finds the mapping from the noisy and sparse APT data onto the reference lattice, it yields a first approximation of the interaction potentials between the multiple elemental types present in HEAs. Presently, the bottleneck for understanding materials with nano-scale engineered disorder is the lack of appropriate interaction potentials. Consequently, researchers are unable to perform relevant modeling of these materials, and the property-structure relationships present in these materials remain

unknown. While the promise of HEAs remains great for society, we have yet to unlock these relationships and thus cannot tailor an alloy to any specific application. Furthermore, the methodologies presented herein may serve as a guide to researchers working to improve the resolution of the atom probe tomography technique. Our sensitivity analysis yields quantitative bounds on the necessary resolution to quantify first and second neighbor relationships within a material.

Outside the field of materials science, our Bayesian point set methodology may be applied to computer vision problems arising in medical imaging and autonomous driving (LiDAR), to problems arising in chemistry, such as molecule representation, which may lead to new drug discoveries, or in bioinformatics for protein visualization.

5.3 Future Research

The primary area of future work with our methodologies is to improve the inference of short-range ordering within the variational Bayesian registration methodology. To model the chemical ordering, we employ the quasi-chemical lattice model. More sophisticated models for the interaction potentials exist, and should be explored, such as the embedded-atom type potentials, which are commonly used to describe metallic systems composed of at most, three different atomic types [113, 114], whereas the HEA systems we consider contain at least five different atomic types.

The quasi-chemical model employed herein accounts for pairwise interactions between all of the atoms in a neighborhood, and is a reasonable first approximation. An accurate estimation of the interaction potentials is dependent on a representative number of atoms in the configurations. In the real APT data though, we have seen configurations that are exclusively composed of a single atomic type. Note however that we did not however consider any single element alloy in our studies. In order to construct the approximate interaction potentials, we should employ some outlier detection and exclude such neighborhoods from the potential calculations.

The energy term in our Bayesian formulation is the Helmholtz free energy, $E = U - TS$, with temperature T, entropy S, and enthalpy U. We compute this term at at $T = 0$ K, and consequently do not account for the entropic term, but only consider the enthalpy. Recall that these alloys are designed with some configurational entropy, which influences the mixing of the component elements. Thus it is reasonable that we should account for the entropic term in some fashion. Finding the correct temperature to properly weigh the energy and mixing terms merits further investigation, and may help to refine our model, and potentially leading to a more robust algorithmic process against the noise and sparsity present.

The variational Bayesian framework of Section 4.4 yields a predictive density for new data and could be used to impute the missing atoms and their position within the lattice. Indeed, by adopting our Bayesian approach, we can construct the predictive posterior density for new data $\hat{Y}$ and its associated latent variable $\hat{z}$ by

$$p(\hat{Y} \mid Y) = \sum_{\hat{z}} \sum_{\lambda} \int p(\hat{Y} \mid \hat{z}, \mu, \Lambda, \lambda) p(\lambda \mid \omega, \hat{z}) p(\hat{z} \mid \omega) p(\omega) p(\omega, \mu, \Lambda, \lambda \mid Y) \mathrm{d}\omega \mathrm{d}\mu \mathrm{d}\Lambda.$$

By employing our approximation q to the true posterior, we obtain

$$p(\hat{Y} \mid Y) = \sum_{i=1}^{N} \sum_{\lambda} \int \omega_n \mathcal{N}(\hat{Y} \mid \mu_n, \Lambda_n^{-1}) q(\omega) q(\mu_n, \Lambda_n) p(\lambda \mid \omega_n, \hat{z}) \mathrm{d}\omega \mathrm{d}\mu_n \mathrm{d}\Lambda_n.$$

Without the energy term, the above distribution has a closed-form expression which yields a mixture of Student's t-distributions [53, 45]. Including the energetic contribution, it is not clear that an analytic solution exists, and we would have to estimate the quantity via MCMC methods.

Our methodology may also be used to give direction to APT instrument scientists actively working to improve the technique. Indeed, by performing a sensitivity analysis by varying either the noise present or sparsity of the data, we may readily quantify the error present in the resulting data from an APT experiment and say to what level the resolution needs to be increased to obtain a specific level of accuracy in our registration process.

Lastly, this work yields an unprecedented atomic-level view of HEAs for material science researchers. By providing a probabilistic solution, we may provide to materials science researchers a definitive distribution of configurations from APT experiments that quantify both spatial and compositional distributions of the data. By providing the empirical probability of any state, compositional or spatial, in an alloy, materials scientists can use these probabilites as input to an optimization routine, such as GARFfield [115], to generate interaction potentials. From these interaction potentials, materials science researchers may run molecular dynamics simulations, thus accelerating the materials discovery process for these transformative, complex materials. The methodologies described herein bring their goal of understanding HEAs at the atomic level within reach, furthering their understanding, and making structure-property relationships within our grasp.

Bibliography

[1] L. J. Santodonato, Y. Zhang, M. Feygenson, C. M. Parish, M. C. Gao, R. J. Weber, J. C. Neuefeind, Z. Tang, and P. K. Liaw, "Deviation from high-entropy configurations in the atomic distributions of a multi-principal-element alloy," *Nature communications*, vol. 6, p. 5964, 2015. x, xiv, 2, 8, 56, 69, 75, 76, 94, 95, 96

[2] T. L. Hill, *An introduction to statistical thermodynamics*. Courier Corporation, 1986. 3, 82

[3] C. A. Becker, F. Tavazza, Z. T. Trautt, and R. A. B. de Macedo, "Considerations for choosing and using force fields and interatomic potentials in materials science and engineering," *Current Opinion in Solid State and Materials Science*, vol. 17, no. 6, pp. 277–283, 2013. 3

[4] H. L. Becker C., Trautt Z., "Nist interatomic potentials repository," webpage, DOI: 10.18434/m37, 2010. 3

[5] L. M. Hale, Z. T. Trautt, and C. A. Becker, "Evaluating variability with atomistic simulations: the effect of potential and calculation methodology on the modeling of lattice and elastic constants," *Modelling and Simulation in Materials Science and Engineering*, vol. 26, no. 5, p. 055003, 2018. 3

[6] J.-W. Yeh, S.-K. Chen, S.-J. Lin, J.-Y. Gan, T.-S. Chin, T.-T. Shun, C.-H. Tsau, and S.-Y. Chang, "Nanostructured high-entropy alloys with multiple principal elements: novel alloy design concepts and outcomes," *Advanced Engineering Materials*, vol. 6, no. 5, pp. 299–303, 2004. 7, 8

[7] J.-W. Yeh, "Physical metallurgy of high-entropy alloys," *Jom*, vol. 67, no. 10, pp. 2254–2261, 2015. 7

[8] Y. Shi, B. Yang, and P. K. Liaw, "Corrosion-resistant high-entropy alloys: A review," *Metals*, vol. 7, no. 2, p. 43, 2017. 7

[9] Y. Zhang, T. T. Zuo, Z. Tang, M. C. Gao, K. A. Dahmen, P. K. Liaw, and Z. P. Lu, "Microstructures and properties of high-entropy alloys," *Progress in Materials Science*, vol. 61, pp. 1–93, 2014. 7, 8, 57, 75

[10] B. Gludovatz, A. Hohenwarter, D. Catoor, E. H. Chang, E. P. George, and R. O. Ritchie, "A fracture-resistant high-entropy alloy for cryogenic applications," *Science*, vol. 345, no. 6201, pp. 1153–1158, 2014. 7

[11] Z. Lei, X. Liu, Y. Wu, H. Wang, S. Jiang, S. Wang, X. Hui, Y. Wu, B. Gault, and P. Kontis, "Enhanced strength and ductility in a high-entropy alloy via ordered oxygen complexes," *Nature*, vol. 563, no. 7732, p. 546, 2018. 7

[12] Z. Li, K. G. Pradeep, Y. Deng, D. Raabe, and C. C. Tasan, "Metastable high-entropy dual-phase alloys overcome the strength–ductility trade-off," *Nature*, vol. 534, no. 7606, p. 227, 2016. 7

[13] M.-H. Tsai and J.-W. Yeh, "High-entropy alloys: a critical review," *Materials Research Letters*, vol. 2, no. 3, pp. 107–123, 2014. 7, 75

[14] M. A. Hemphill, T. Yuan, G. Wang, J. Yeh, C. Tsai, A. Chuang, and P. K. Liaw, "Fatigue behavior of $Al_{0.5}$CoCrCuFeNi high entropy alloys," *Acta Materialia*, vol. 60, no. 16, pp. 5723–5734, 2012. 7

[15] Z. Tang, T. Yuan, C. Tsai, J. Yeh, C. D. Lundin, and P. K. Liaw, "Fatigue behavior of a wrought $Al_{0.5}$CoCrCuFeNi two-phase high-entropy alloy," *Acta Materialia*, vol. 99, pp. 247–258, 2015. 7

[16] J. Guo, H. Wang, F. von Rohr, Z. Wang, S. Cai, Y. Zhou, K. Yang, A. Li, S. Jiang, and Q. Wu, "Robust zero resistance in a superconducting high-entropy alloy at pressures up to 190 gpa," *Proceedings of the National Academy of Sciences*, vol. 114, no. 50, pp. 13 144–13 147, 2017. 7

[17] P. Koželj, S. Vrtnik, A. Jelen, S. Jazbec, Z. Jagličić, S. Maiti, M. Feuerbacher, W. Steurer, and J. Dolinšek, "Discovery of a superconducting high-entropy alloy," *Physical Review Letters*, vol. 113, no. 10, p. 107001, 2014. 7

[18] Y. Jien-Wei, "Recent progress in high entropy alloys," *Ann. Chim. Sci. Mat*, vol. 31, no. 6, 2006. 7

[19] C. M. Rost, E. Sachet, T. Borman, A. Moballegh, E. C. Dickey, D. Hou, J. L. Jones, S. Curtarolo, and J.-P. Maria, "Entropy-stabilized oxides," *Nature communications*, vol. 6, p. 8485, 2015. 8, 101

[20] Y. Zhang, Y. J. Zhou, J. P. Lin, G. L. Chen, and P. K. Liaw, "Solid-solution phase formation rules for multi-component alloys," *Advanced Engineering Materials*, vol. 10, no. 6, pp. 534–538, 2008. 8, 9

[21] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, "Machine learning for molecular and materials science," *Nature*, vol. 559, no. 7715, p. 547, 2018. 8, 56

[22] D. Larson, T. Prosa, R. Ulfig, B. Geiser, and T. Kelly, *Local Electrode Atom Probe Tomography: A User's Guide*. Springer, 2013. 8, 69

[23] M. K. Miller, *Atom-Probe Tomography : The Local Electrode Atom Probe*. Springer, 2014. 8, 9, 69

[24] T. F. Kelly, M. K. Miller, K. Rajan, and S. P. Ringer, "Atomic-scale tomography: A 2020 vision," *Microscopy and Microanalysis*, vol. 19, no. 3, pp. 652–664, 2013. 9, 56, 75

[25] M. K. Miller, T. F. Kelly, K. Rajan, and S. P. Ringer, "The future of atom probe tomography," *Materials Today*, vol. 15, no. 4, pp. 158–165, 2012. 9, 56, 69, 75

[26] B. Gault, M. P. Moody, J. M. Cairney, and S. P. Ringer, "Atom probe crystallography," *Materials Today*, vol. 15, no. 9, pp. 378–386, 2012. 9, 56

[27] B. Gault, M. P. Moody, F. De Geuser, A. La Fontaine, L. T. Stephenson, D. Haley, and S. P. Ringer, "Spatial resolution in atom probe tomography," *Microscopy and Microanalysis*, vol. 16, no. 1, pp. 99–110, 2010. 9, 69

[28] H. Edelsbrunner and J. Harer, "Persistent homology-a survey," *Contemporary Mathematics*, vol. 453, pp. 257–282, 2008. 10

[29] ——, *Computational Topology: An Introduction*. Providence, RI: American Mathematical Society, 2010. 10

[30] T. Kaczynski, K. Mischaikow, and M. Mrozek, *Computational homology*. Springer Science & Business Media, 2006, vol. 157. 10

[31] R. W. Ghrist, *Elementary Applied Topology*. Createspace Seattle, 2014, vol. 1. 10

[32] L. Wasserman, "Topological data analysis," *Annual Review of Statistics and Its Application*, vol. 5, pp. 501–532, 2018. 11, 58

[33] B. T. Fasy, F. Lecci, A. Rinaldo, L. Wasserman, S. Balakrishnan, A. Singh *et al.*, "Confidence sets for persistence diagrams," *The Annals of Statistics*, vol. 42, no. 6, pp. 2301–2339, 2014. 11, 59

[34] F. Chazal, D. Cohen-Steiner, and Q. Mérigot, "Geometric inference for probability measures," *Foundations of Computational Mathematics*, vol. 11, no. 6, pp. 733–751, 2011. 11

[35] A. Marchese and V. Maroulas, "Signal classification with a point process distance on the space of persistence diagrams," *Advances in Data Analysis and Classification*, vol. 12, no. 3, pp. 657–682, 2018. 13, 58, 64, 66

[36] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, "Hybrid monte carlo," *Physics letters B*, vol. 195, no. 2, pp. 216–222, 1987. 15

[37] R. M. Neal, "Bayesian learning for neural networks," New York, 1996. 15, 16

[38] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, *Handbook of markov chain monte carlo*. CRC press, 2011. 15, 16

[39] G. Teschl, *Ordinary differential equations and dynamical systems*. American Mathematical Society Providence, 2012, vol. 140. 15

[40] G. O. Roberts and J. S. Rosenthal, "Optimal scaling of discrete approximations to langevin diffusions," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 60, no. 1, pp. 255–268, 1998. 17, 48

[41] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997. 18, 20, 23

[42] T. Hastie, S. Rosset, J. Zhu, and H. Zou, "Multi-class adaboost," *Statistics and its Interface*, vol. 2, no. 3, 2009. 18, 20

[43] P. Domingos and M. Pazzani, "On the optimality of the simple bayesian classifier under zero-one loss," *Machine learning*, vol. 29, no. 2-3, pp. 103–130, 1997. 19

[44] H. Zhang, "The optimality of naive bayes," *AA*, vol. 1, no. 2, p. 3, 2004. 19

[45] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012. 19, 33, 103

[46] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*, ser. Springer Series in Statistics. New York, NY: Springer-Verlag New York, 2009. 19, 23, 26, 27, 32

[47] R. E. Schapire and Y. Freund, *Boosting: Foundations and algorithms*. MIT press, 2012. 19, 20, 23, 26

[48] J. Friedman, T. Hastie, R. Tibshirani *et al.*, "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)," *The Annals of Statistics*, vol. 28, no. 2, pp. 337–407, 2000. 20, 26, 27, 68

[49] T. Hastie, "Generalized additive models," London ; New York, 1990. 22, 68

[50] L. Breiman, *Classification and Regression Trees*, ser. Wadsworth statistics/probability series. Wadsworth International Group, 1984. 23

[51] B. Efron and T. Hastie, *Computer age statistical inference*. Cambridge University Press, 2016, vol. 5. 23, 63

[52] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977. 32, 35, 84

[53] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006. 32, 36, 40, 84, 103

[54] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods; 2nd ed.*, ser. Springer texts in statistics. Berlin: Springer, 2004. 32, 33, 82, 84, 85

[55] R. M. Neal and G. E. Hinton, "A view of the em algorithm that justifies incremental, sparse, and other variants," in *Learning in graphical models*. Springer, 1998, pp. 355–368. 32, 33, 34, 35, 84

[56] M. J. Beal *et al.*, *Variational algorithms for approximate Bayesian inference*. University of London London, 2003. 33

[57] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American Statistical Association*, vol. 112, no. 518, pp. 859–877, Feb 2017. [Online]. Available: http://dx.doi.org/10.1080/01621459.2017.1285773 33, 36

[58] J. L. W. V. Jensen *et al.*, "Sur les fonctions convexes et les inégalités entre les valeurs moyennes," *Acta mathematica*, vol. 30, pp. 175–193, 1906. 34

[59] T. Minka, "Estimating a dirichlet distribution," 2000. 40, 84

[60] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, Feb 1992. 41

[61] L. Cheng, S. Chen, X. Liu, H. Xu, Y. Wu, M. Li, and Y. Chen, "Registration of laser scanning point clouds: A review," *Sensors*, vol. 18, no. 5, p. 1641, 2018. 41

[62] S. Granger and X. Pennec, "Multi-scale em-icp: A fast and robust approach for surface registration," in *European Conference on Computer Vision*. Springer, 2002, pp. 418–432. 41

[63] S. Klein, J. P. Pluim, M. Staring, and M. A. Viergever, "Adaptive stochastic gradient descent optimisation for image registration," *International journal of computer vision*, vol. 81, no. 3, p. 227, 2009. 41

[64] H. Li and R. Hartley, "The 3d-3d registration problem revisited," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8. 41

[65] C. Papazov and D. Burschka, "Stochastic global optimization for robust point set registration," *Computer Vision and Image Understanding*, vol. 115, no. 12, pp. 1598–1609, 2011. 41

[66] L. Qi and Y. Song, "An even order symmetric b tensor is positive definite," *Linear Algebra and Its Applications*, vol. 457, pp. 303–312, 2014. 41

[67] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, vol. 89, no. 2, pp. 114–141, 2003. 42

[68] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010. 42

[69] M. P. Moody, B. Gault, L. T. Stephenson, R. K. Marceau, R. C. Powles, A. V. Ceguerra, A. J. Breen, and S. P. Ringer, "Lattice rectification in atom probe tomography: Toward true three-dimensional atomic microscopy," *Microscopy and Microanalysis*, vol. 17, no. 2, pp. 226–239, 2011. 56, 69, 77

[70] J. A. Chisholm and S. Motherwell, "A new algorithm for performing three-dimensional searches of the cambridge structural database," *Journal of applied crystallography*, vol. 37, no. 2, pp. 331–334, 2004. 56

[71] D. Hicks, C. Oses, E. Gossett, G. Gomez, R. H. Taylor, C. Toher, M. J. Mehl, O. Levy, and S. Curtarolo, "Aflow-sym: platform for the complete, automatic and self-consistent symmetry analysis of crystals," *Acta Crystallographica Section A: Foundations and Advances*, vol. 74, no. 3, pp. 184–203, 2018. 56, 57

[72] J. D. Honeycutt and H. C. Andersen, "Molecular dynamics study of melting and freezing of small lennard-jones clusters," *Journal of Physical Chemistry*, vol. 91, no. 19, pp. 4950–4963, 1987. 56, 57

[73] P. M. Larsen, S. Schmidt, and J. Schiøtz, "Robust structural identification via polyhedral template matching," *Modelling and Simulation in Materials Science and Engineering*, vol. 24, no. 5, p. 055007, 2016. 56, 57

[74] A. Togo and I. Tanaka, "Spglib : a software library for crystal symmetry search," *arXiv preprint arXiv:1808.01590*, 2018. 56

[75] A. Ziletti, D. Kumar, M. Scheffler, and L. M. Ghiringhelli, "Insightful classification of crystal structures using deep learning," *Nature communications*, vol. 9, no. 1, p. 2775, 2018. 56, 76

[76] G. Carlsson, A. Zomorodian, A. Collins, and L. J. Guibas, "Persistence barcodes for shapes," *International Journal of Shape Modeling*, vol. 11, no. 02, pp. 149–187, 2005. 58

[77] H. Edelsbrunner, D. Letscher, and A. Zomorodian, "Topological persistence and simplification," in *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*. IEEE, 2000, pp. 454–463. 58

[78] A. Marchese and V. Maroulas, "Topological learning for acoustic signal identification," in *Information Fusion (FUSION), 2016 19th International Conference on*. IEEE, 2016, pp. 1377–1381. 58

[79] A. Marchese, V. Maroulas, and J. Mike, "K-means clustering on the space of persistence diagrams," in *Wavelets and Sparsity XVII*, vol. 10394. International Society for Optics and Photonics, 2017, p. 103940W. 58

[80] V. Maroulas, J. L. Mike, and C. Oballe, "Nonparametric estimation of probability density functions of random persistence diagrams," *Journal of Machine Learning Research*, vol. 20, no. 151, pp. 1–49, 2019. [Online]. Available: http://jmlr.org/papers/v20/18-618.html 58

[81] V. Maroulas, F. Nasrin, and C. Oballe, "Bayesian inference for persistent homology," *SIAM Jounral on Mathematics of Data Science*, DOI: 10.1137/19M1268719, 2020. 58

[82] I. Sgouralis, A. Nebenführ, and V. Maroulas, "A bayesian topological framework for the identification and reconstruction of subcellular motion," *SIAM Journal on Imaging Sciences*, vol. 10, no. 2, pp. 871–899, 2017. 58

[83] J. Townsend, C. P. Micucci, J. H. Hymel, V. Maroulas, and K. Vogiatzis, "Representation of Molecular Structures with Persistent Homology Leads to the Discovery of Molecular Groups with Enhanced CO2 Binding," 11 2019. [Online]. Available: https://chemrxiv.org/articles/Representation_of_Molecular_Structures_with_Persistent_Homology_Leads_to_the_Discovery_of_Molecular_Groups_with_Enhanced_CO2_Binding/10263293 58

[84] A. Zomorodian and G. Carlsson, "Computing persistent homology," *Discrete & Computational Geometry*, vol. 33, no. 2, pp. 249–274, 2005. 58

[85] M. Carriere, M. Cuturi, and S. Oudot, "Sliced wasserstein kernel for persistence diagrams," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 664–673. 58

[86] H. Adams, T. Emerson, M. Kirby, R. Neville, C. Peterson, P. Shipman, S. Chepushtanova, E. Hanson, F. Motta, and L. Ziegelmeier, "Persistence images: A stable vector representation of persistent homology," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 218–252, 2017. 58

[87] P. Bubenik, "Statistical topological data analysis using persistence landscapes," *The Journal of Machine Learning Research*, vol. 16, no. 1, pp. 77–102, 2015. 58

[88] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer, "Stability of persistence diagrams," *Discrete & Computational Geometry*, vol. 37, no. 1, pp. 103–120, 2007. 59

[89] F. Chazal, V. de Silva, and S. Oudot, "Persistence stability for geometric complexes," *Geometriae Dedicata*, vol. 173, no. 1, pp. 193–214, Dec 2014. 59

[90] F. Pfender and G. M. Ziegler, "Kissing numbers, sphere packings, and some unexpected proofs," *Notices-American Mathematical Society*, vol. 51, pp. 873–883, 2004. 62

[91] M. Goff, "Extremal betti numbers of vietoris–rips complexes," *Discrete & Computational Geometry*, vol. 46, no. 1, pp. 132–155, 2011. 62

[92] V. Maroulas, C. P. Micucci, and A. Spannaus, "A stable cardinality distance for topological classification," *Advances in Data Analysis and Classification*, pp. 1–18, DOI: 10.1007/s11634-019-00378-3, 2019. 66, 78

[93] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. 68

[94] N. W. McNutt, O. Rios, V. Maroulas, and D. J. Keffer, "Interfacial li-ion localization in hierarchical carbon anodes," *Carbon*, vol. 111, pp. 828–834, 2017. 69

[95] T. S. Breusch and A. R. Pagan, "A simple test for heteroscedasticity and random coefficient variation," *Econometrica: Journal of the Econometric Society*, pp. 1287–1294, 1979. 73

[96] Y. Hiraoka, T. Nakamura, A. Hirata, E. G. Escolar, K. Matsue, and Y. Nishiura, "Hierarchical structures of amorphous solids characterized by persistent homology," *Proceedings of the National Academy of Sciences*, p. 201520877, 2016. 73

[97] I. Donato, M. Gori, M. Pettini, G. Petri, S. De Nigris, R. Franzosi, and F. Vaccarino, "Persistent homology analysis of phase transitions," *Physical Review E*, vol. 93, no. 5, p. 052138, 2016. 73

[98] Y. Lee, S. D. Barthel, P. Dłotko, S. M. Moosavi, K. Hess, and B. Smit, "Quantifying similarity of pore-geometry in nanoporous materials," *Nature Communications*, vol. 8, p. 15396, 2017. 73

[99] D. B. Miracle and O. N. Senkov, "A critical review of high entropy alloys and related concepts," *Acta Materialia*, vol. 122, pp. 448–511, 2017. 75

[100] T. F. Chan, G. H. Golub, and R. J. LeVeque, "Algorithms for computing the sample variance: Analysis and recommendations," *The American Statistician*, vol. 37, no. 3, pp. 242–247, 1983. 78

[101] A. Spannaus, V. Maroulas, C. Putman Micucci, D. J. Keffer, and K. J. H. Law, "Materials fingerprinting classification," 2019, *Proceedings of the National Academy of Science*, submitted. 78

[102] B. Jian and B. C. Vemuri, "Robust point set registration using gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1633–1645, 2011. 78

[103] S. Koo, D. Lee, and D.-S. Kwon, "Gmm-based 3d object representation and robust tracking in unconstructed dynamic environments," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 1114–1121. 78

[104] J. Ma, J. Chen, D. Ming, and J. Tian, "A mixture model for robust point matching under multi-layer motion," *PloS one*, vol. 9, no. 3, 2014. 78

[105] J. Ma, J. Zhao, and A. L. Yuille, "Non-rigid point set registration by preserving global and local structures," *IEEE Transactions on image Processing*, vol. 25, no. 1, pp. 53–64, 2015. 78

[106] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, Apr 1991. 85

[107] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983. 85

[108] B. Hajek, "Optimization by simulated annealing: a necessary and sufficient condition for convergence," *Lecture Notes-Monograph Series*, pp. 417–427, 1986. 85

[109] R. Herbrich, "Minimising the Kullback–leibler divergence," *Microsoft, Tech. Rep.*, 2005. 85

[110] G. Kurz, F. Pfaff, and U. D. Hanebeck, "Kullback-leibler divergence and moment matching for hyperspherical probability distributions," in *2016 19th International Conference on Information Fusion (FUSION)*. IEEE, 2016, pp. 2087–2094. 85

[111] J. R. Hershey and P. A. Olsen, "Approximating the kullback leibler divergence between gaussian mixture models," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 4. IEEE, 2007, pp. IV–317. 85

[112] J. Wang and H. Sehitoglu, "Dislocation slip and twinning in ni-based l12 type alloys," *Intermetallics*, vol. 52, pp. 20–31, 2014. 95

[113] M. I. Baskes, "Modified embedded-atom potentials for cubic materials and impurities," *Physical review B*, vol. 46, no. 5, p. 2727, 1992. 102

[114] B.-J. Lee and M. Baskes, "Second nearest-neighbor modified embedded-atom-method potential," *Physical Review B*, vol. 62, no. 13, p. 8564, 2000. 102

[115] A. Jaramillo-Botero, S. Naserifar, and W. A. Goddard III, "General multiobjective force field optimization framework, with application to reactive force fields for silicon carbide," *Journal of chemical theory and computation*, vol. 10, no. 4, pp. 1426–1439, 2014. 103

Vita

Adam Spannaus was born in Detroit, MI and his interest in mathematics and computing was fostered by his teacher Michael McGaw. Adam attended the University of Michigan, Ann Arbor for his undergraduate education, pursing a dual-degree in mathematics and music performance. He graduated with honors with a Bachelor of Fine Arts degree. After a hiatus, Adam returned to school at the University of Tennessee, Knoxville, and finished the undergraduate coursework requisite for any graduate work. Always with an eye towards applications and computation, Adam began to pursue his interests under the supervision of Professor Vasileios Maroulas, engaging computational techniques with machine learning and topology, and leading to the results contained in his dissertation. He will continue his research as a post-doctoral research associate with the Biomedical Science, Engineering, and Computing group at Oak Ridge National Laboratory.