

ABSTRACT

WANG, HAOYU. *Advances and Applications of Nonparametric Statistics*. (Under the direction of Arnab Maity and Brian Reich).

Nonparametric modeling provides a suite of statistical methods that require very limited assumptions to be made about the data. As nonparametric methods make fewer assumptions, their applicability is much wider than the corresponding parametric methods and are more robust. In this thesis, we study the properties and applications of three nonparametric modeling methods. First, we propose a new variable selection technique for the case where the predictors are functional and the response is scalar. The method is based on a flexible nonparametric model. We implement functional PCA to convert functional predictors to multivariate vector representatives, then assume a Gaussian process regression model to capture the nonlinear relationships between response and functional predictors. We then conduct similarity-based regression with elastic net penalty to select important variables. The method is characterized by low specificity rate and computational efficiency demonstrated by a simulation study. We apply the proposed algorithm to select important forearm muscles driving hand movement to aid in producing a high-functioning, EMG-based controller of a robotic hand prosthetic.

Second, we develop a Bayesian nonparametric regression model for grain boundary (GB) energy predictions. Grain boundary (GB) energy is a fundamental property that affects the form of grain boundary and plays an important role to unveil the behavior of polycrystalline materials. With a better understanding of grain boundary energy distribution (GBED), we can produce more durable and efficient materials that will further improve productivity and reduce loss. The lack of robust GB structure-property relationships still remains one of the biggest obstacles towards developing true bottom-up models for the behavior of polycrystalline materials. Progress has been slow because of the inherent complexity associ-

ated with the structure of interfaces and the vast five-dimensional configurational space in which they reside. Estimating the GBED is challenging from a statistical perspective because there are not direct measurements on the grain boundary energy. We only have indirect information in the form of an unidentifiable homogeneous set of linear equations. In this paper, we propose a new statistical model to determine the GBED from the microstructures of polycrystalline materials. We apply spline-based regression with constraints to successfully recover the GB energy surface. Hamiltonian Monte Carlo and Gibbs sampling are used for computation and model fitting. Compared with conventional methods, our method not only gives more accurate predictions but also provides prediction uncertainties.

Finally, we apply deep learning to geostatistical prediction. Kriging is the predominant method used for spatial prediction, but relies on the assumption that predictions are linear combinations of the observations. Kriging often also relies on additional assumptions such as normality and stationarity. We propose a more flexible spatial prediction method based on the Nearest-Neighbor Neural Network (4N) process that embeds deep learning into a geostatistical model. We show that the 4N process is a valid stochastic process and propose a series of new ways to construct features to be used as inputs to the deep learning model based on neighboring information. Our model framework outperforms some existing state-of-art geostatistical modelling methods for simulated non-Gaussian data and is applied to a massive forestry dataset.

© Copyright 2019 by Haoyu Wang

All Rights Reserved

Advances and Applications of Nonparametric Statistics

by
Haoyu Wang

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Statistics

Raleigh, North Carolina
2019

APPROVED BY:

Jonathan Stallrich

Srikanth Patala

Arnab Maity
Co-chair of Advisory Committee

Brian Reich
Co-chair of Advisory Committee

DEDICATION

To my beloved family.

BIOGRAPHY

The author was born in Dalian, Liaoning, China in April 1992. In 2010, he was admitted to Nankai University, where he spent four years on studying Mathematics and Statistics. After receiving Bachelor's degree of Statistics from Nankai University in 2014, he attended North Carolina State University for a Ph.D. in Statistics. In the city of Raleigh, he met his wife Xingqi Du, who was his stat buddy. Under the direction of Dr. Arnab Maity and Dr. Brian Reich, he will complete his Ph.D. in Statistics in May 2019.

ACKNOWLEDGEMENTS

First, I would like to express my deepest gratitude to my advisors, Dr. Arnab Maity and Dr. Brian Reich, for their kind guidance, excellent inspiration and continued support. Dr. Maity shows great patience and kindness, I really appreciate all his dedication of time and mentoring. I am also grateful to have the opportunity to work with Dr. Reich, his creativity and passion always inspire me. I am so lucky to have them as my advisors, without whom this dissertation would not be possible.

I would like to thank Dr. Jonathan Stallrich and Dr. Srikanth Patala for taking precious time to serve as my Ph.D. committee members. It has been a pleasure of mine to work with them, and the work done with them become important parts of my thesis.

I appreciate the study resources provided by the Department of Statistics. I am grateful for every faculty member and staff for their great services and dedication. My graduate experience benefit greatly from the friendly environment they create. I thank our department head Dr. Len Stefanski, director of statistics graduate program Dr. Wenbin Lu and program assistant Alexander Lanakila for their generous help. Special thanks to all my friends who support and encourage me as always.

I am also grateful to the internship experiences at Pfizer and SAS. Special thanks to my supervisors, Brian Corrigan and Arin Chaudhuri for sharing with me many practical knowledge and skills.

Last but not least, none of this would have been possible without the love of my family. My parents have always been supportive, and given me great freedom since I was a kid. My greatest appreciation goes to the most important person in my life, my wife, Xingqi Du. She is such a wonderful woman who has shared all the ups and downs with me in the past years. I could not be too grateful to have her in my life.

TABLE OF CONTENTS

List of Tables	vii
List of Figures	viii
Chapter 1 Variable Selection for Functional Data Using Kernel Machine	1
1.1 Introduction	1
1.2 Methodology	4
1.2.1 Nonparametric Model for Functional Data	4
1.2.2 Variable Selection Procedure	5
1.3 Simulation Study	8
1.4 Real Data Application	12
1.4.1 Overview of Dataset	12
1.4.2 Results	15
1.5 Discussion	17
Chapter 2 Constrained Bayesian Nonparametric Regression for Grain Bound- ary Energy Predictions	19
2.1 Introduction	20
2.2 Analyzing the 3D EBSD Triple Junction Data	22
2.3 Statistical model	25
2.3.1 Model description and prior specification	25
2.3.2 Constraints	27
2.3.3 Posterior computation and model fitting	28
2.4 Simulation Study	29
2.4.1 One-dimensional simulation	30
2.4.2 Two-dimensional simulation	31
2.4.3 Five-dimensional simulation	33
2.5 Real Data	34
2.6 Discussion	39
Chapter 3 Nearest-Neighbor Neural Networks for Geostatistics	42
3.1 Introduction	42
3.2 Methodology	45
3.2.1 Nearest-Neighbor Gaussian Process	45
3.2.2 Nearest-Neighbor Neural Network (4N) model	46
3.2.3 Feature Engineering	48
3.2.4 Computational Details	50
3.3 Simulation Study	51
3.3.1 Mean Prediction	53
3.3.2 Quantile Prediction	54

3.3.3	Variable Importance	56
3.4	Canopy height data analysis	58
3.5	Conclusions	60
References		62
APPENDICES		73
Appendix A	Supplementary analysis results	74
Appendix B	Hamiltonian Monte Carlo	80

LIST OF TABLES

Table 1.1	Simulation Results for Linear Case	10
Table 1.2	Simulation Results for Nonlinear Case	10
Table 1.3	Method Comparisons for Linear Case. KM refers to our Kernel Machine method, FAR refers to Functional Additive Regression method and grLasso refers to Gertheiss et al. (2013)'s method. False negative rate, false positive rate, average model size and Matthews correlation coefficient are compared.	11
Table 1.4	Method Comparison for Nonlinear Case. KM refers to our Kernel Machine method, FAR refers to Functional Additive Regression method and grLasso refers to Gertheiss et al. (2013)'s method. False negative rate, false positive rate, average model size and Matthews correlation coefficient are compared.	12
Table 1.5	Variable Selection Results for EMG Data	17
Table 2.1	Mean squared error and 95% confidence interval coverage comparison between Constrained Bayesian Nonparametric Regression (CBNR) and Block Aggregation (BA) for simulated data. Case 1 and 2 refer to the one-dimensional simulations, Case 3 refers to the two-dimensional simulation and Case 4 refers to the five-dimensional simulation.	34
Table 2.2	Relative prediction MSE of CBNR to BA	35
Table 2.3	Correlations between δ_{ij} 's and variances of δ_{ij} 's, where r_{jk} is the correlation estimate between δ_{ij} and δ_{ik} , σ_j^2 is the variance estimate of δ_{ij} . All the numbers in the table are multiplied by 10^{-3}	38
Table 3.1	Mean squared error comparison between 4N models and NNGP (standard error shown in parenthesis) for simulated data.	55
Table 3.2	Check loss with different quantiles comparison between 4N models and ALP model, with $n = 1,000$ observations (standard error shown in parenthesis) for simulated data.	56
Table 3.3	Check loss with different quantiles comparison between 4N models and ALP model, with $n = 10,000$ observations (standard error shown in parenthesis) for simulated data.	57
Table 3.4	Prediction performance for 4N models, NNGP and ALP for the canopy height data.	60
Table A.1	Proportion of Variance Explained by the First Five Principle Components	76
Table A.2	Variable Selection Results for Hand Movement (4 signals)	77
Table A.3	Variable Selection Results for Wrist Movement (4 signals)	77
Table A.4	Variable Selection Results for Hand Movement(16 variables)	77
Table A.5	Variable Selection Results for Hand Movement(16 variables) Cont.	79

LIST OF FIGURES

Figure 1.1	The biomechanical system for hand movement for able-bodied subject and transradial amputee.	14
Figure 1.2	Joint finger angle (radians) for flexion and extension movements, in black, and two normalized EMG signals, in orange and green.	15
Figure 2.1	Illustrative example of one triple junction consisting of three grain boundaries: A (grey), B (red) and C (green). \mathbf{O}^s is the grain orientation matrix, $\hat{\mathbf{n}}^s$ is the boundary-plane orientation vector and $\hat{\mathbf{l}}^s$ is the tangent vector for the triple junction line.	24
Figure 2.2	Analysis of one simulated dataset for Case 1. Plotted is the true curve ξ (black), the Constrained Bayesian Nonparametric Regression (CBNR) Method (red), 95% credible set (dashed red) and the Block Aggregation method (blue).	31
Figure 2.3	Analysis of one simulated dataset for Case 2. Plotted is the true curve ξ (black), the Constrained Bayesian Nonparametric Method (red), 95% credible set (dashed red) and the Block Aggregation method (blue).	32
Figure 2.4	Analysis of one simulated dataset for Case 3. Panels (a) and (c) show the true values and panels (b) and (d) show the recovered values by Constrained Bayesian Nonparametric Regression method.	33
Figure 2.5	Stereographic projections of the grain boundary energy distribution for the $\Sigma 3$ misorientation. Posterior mean of the unit-less capillary vectors ξ is 0.61.	36
Figure 2.6	One dimensional prediction of the grain boundary energy distribution with 95% prediction interval for the $\Sigma 3$ misorientation fixing one boundary-plane parameters	37
Figure 2.7	Histograms of δ_i for different values of γ^2	40
Figure 2.8	QQ plots for $\delta_i, \gamma^2 = 0.5$	41
Figure 3.1	Data (\mathbf{Y}) generated from different simulation settings.	53
Figure 3.2	Relative prediction MSE of 4N (Kriging + Nonparametric) to NNGP as a function of sample size n , for data generated from GP and max stable models.	55
Figure 3.3	Relative variable importance (averaged over simulated datasets) for the Kriging feature, spatial location of the prediction site, and the total importance of the three features (latitude and longitude, difference to the prediction location, and response) for each neighbor, plotted so that $l = 1$ is the closest neighbor and $l = 10$ is the most distant.	58
Figure 3.4	(a) Satellite image from Google map, with the blue diamond showing the area where the data are collected and (b) canopy height (meters) in the study area.	59

Figure 3.5	Relative Variable Importance for CHM Data	60
Figure A.1	Hand Positions(4 Variables)	75
Figure A.2	4 Variables(Subtract from mean)	75
Figure A.3	Wrist Positions(4 Variables)	76
Figure A.4	4 Variables(Subtract from mean)	76
Figure A.5	Hand Positions (16 Variables)	78
Figure A.6	EMG 1 - 16 (Subtracted from mean)	78
Figure A.7	Hierarchical Clustering on 16 EMG Signals	78
Figure A.8	Variable Selection Result for Hand Movement (by variable)	79
Figure A.9	Variable Selection Result for Hand Movement (by cluster)	79
Figure B.1	Leapfrog method with different stepsizes	81
Figure B.2	Sampling from a two-dimensional distribution with $\log p(x, y) \propto -\frac{1}{2}(x-4)^2 - \frac{1}{2}(y-4)^2$, constrained to $x \leq y \leq 1.1x$ and $x, y \geq 0$, using Exact Hamiltonian Monte Carlo method.	82

CHAPTER

1

VARIABLE SELECTION FOR FUNCTIONAL DATA USING KERNEL MACHINE

1.1 Introduction

Functional regression is widely used to explore the relationship between response and functional predictors. There have been a number of studies in the area of functional linear regression with scalar response and functional predictors. Cardot et al. (1999), Reiss and

Ogden (2007) apply functional principal component analysis to univariate functional linear regression with scalar response. Muller and Stadtmuller (2005) discuss generalized linear regression with scalar response and univariate functional predictor in detail. Crainiceanu et al. (2009) propose generalized multilevel functional regression following the work from Di et al. (2009), where multilevel functional principal component analysis is applied to the Sleep Heart Health Study. Goldsmith et al. (2011) propose a penalized functional regression model, where the coefficient function is estimated using penalized spline regression. Delaigle and Hall (2012) discuss methodology and theory for partial least squares applied to functional data in detail. However, when the true relationship between response and predictors is not linear, the assumption of functional linear relationship may lead to inaccurate estimation and prediction. Therefore, some studies focus on functional nonlinear regression models as well. James and Silverman (2005) propose a kernel based index model to implement a nonlinear functional regression. Chen et al. (2011) and Ferraty et al. (2013) extend the work to fully nonparametric settings. Shi et al. (2007) propose a Gaussian process functional regression (GPFR) model with functional response and extend the model to situations where the response is non-Gaussian, see Wang and Shi (2014) for details. Lian (2011) proposes a functional partial linear model to address the case where some functional variables are related to responses linearly while other variables have more complicated relationship with the responses. Jiang and Wang (2011) propose a functional single index model for longitudinal and functional data. However, these mentioned methods mainly focus on modeling and estimation, but not on variable selection.

Under the setting of functional covariates and scalar response, several variable selection methods have been proposed. Matsui and Konishi (2011) apply the group SCAD to select functional variables each of which is controlled by multiple parameters. Gertheiss et al. (2013) propose a variable selection method based on group LASSO proposed by Yuan and Lin (2006) that can simultaneously control the sparsity of the model and the smoothness of

the corresponding coefficient functions by adequate penalization. Gareth M. James and Zhu (2009) introduce a method called "Functional Linear Regression That's Interpretable" (FLIRTI), which can estimate coefficient functions and do variable selection simultaneously. However, these methods are based on the assumption of linear relationship between response and functional predictors. Yingying Fan and Radchenko (2015) propose a variable selection algorithm for functional additive models based on penalized least square. They first get linear probes, and then model the linear probes with nonparametric functions. Their algorithm can handle both linear and nonlinear problems, and can select a large number of candidate variables. In our paper, we use kernel machine and similarity regression to conduct variable selection.

Kernel machine method has been developed in machine learning as a powerful learning technique for multi-dimensional data. Some examples are Vapnik (1998); Scholkopf and Smola (2002); Suykens et al. (2002); Rasmussen and Williams (2006). It is also well known that it can handle problem of nonlinearity. Popular examples of kernel machine methods include support vector machine (SVM) and Gaussian process. Liu et al. (2007) consider a semiparametric regression model using Least Square Kernel Machine (LSKM) and show that the LSKM semiparametric regression can be formulated using a linear mixed model. Another application of kernel machine is in the area of genetics, where kernels are used to capture similarity between genotypes. Tzeng et al. (2009) propose a similarity-based regression method based on kernel machine technique to detect associations between traits and multimarker genotypes. The advantage of similarity-based regression is that it applies kernel machine to both response and predictors, so that it can handle the case where the response is functional or multi-dimensional. In our paper, we apply similarity-based regression with elastic net regularization (Zou and Hastie 2005).

In this paper, we consider a nonparametric model with scalar response and functional predictors. We transform functional data to multivariate data using functional Principal

Component Analysis, and thus get scores as representative for the functional predictors. Instead of directly modeling the relationship between response and predictors, we assume a Gaussian process to connect response and scores of these predictors. We then use kernel machine framework and penalized regression to conduct variable selection. The algorithm is shown to be computationally fast and have high specificity rate, that is, noises are rarely chosen by our algorithm. To the best of our knowledge, this is the first time that kernel machine and similarity-based method are used in variable selection for nonlinear scalar-on-functional model.

The remainder of the paper is organized as follows. Section 1.2 presents the methodology for variable selection. Section 1.3 shows simulation results under both functional linear and nonlinear setting. In section 1.4, the algorithm is applied to prosthetic arm control data. At last, section 1.5 discusses some advantages and disadvantages of our proposed method, future work is also mentioned in this section.

1.2 Methodology

1.2.1 Nonparametric Model for Functional Data

Suppose for subject $i \in \{1, \dots, n\}$, we observe a scalar response y_i , and p functional covariates $X_{i1}(\cdot), \dots, X_{ip}(\cdot)$ within the domain of $[0, \tau_j]$, $j = 1, 2, \dots, p$. We assume that $X_{ij}(\cdot)$, $i = 1, \dots, p$ are independent realizations of the squared integrable processes $X_1(\cdot), \dots, X_p(\cdot)$. In practice $X_{ij}(\cdot)$ is only observed on a finite set of points $\{t_{j1}, \dots, t_{jN_j}\}$. We assume X'_{ij} 's are observed at equally-spaced time points and are observed without measurement errors. The case where X'_{ij} 's are observed with measurement errors will be discussed later. The focus of this paper is to develop procedures that perform variable selection.

We consider a functional additive regression model with scalar response variable defined

as

$$Y_i = \alpha + \sum_{j=1}^p f_j\{X_{ij}(\cdot)\} + \epsilon_i,$$

where $Y_i, i \in \{1, \dots, n\}$ stands for n responses, $f_j(\cdot), j \in \{1, \dots, p\}$ is unknown nonlinear function, depending on functional covariate $X_{ij}(\cdot)$, ϵ_i is random error with mean 0 and variance σ^2 , typically assumed to be normally distributed and independent from each other. We assume $E[f_j\{X_{ij}(\cdot)\}] = 0$ so that $f_j(\cdot)$ s are identifiable.

1.2.2 Variable Selection Procedure

For such a flexible regression model, it is challenging to directly model the functional predictors $X_{ij}(\cdot)$ s, since the integrable processes X_1, \dots, X_p are infinite-dimensional. In addition, we need to capture the nonlinear relationship between response and predictors and conduct variable selection. To address these issues, we propose a three-step procedure.

Functional Principal Component Analysis

The first step of our variable selection procedure is to conduct functional PCA on observed data. Functional principal component analysis has been widely used and become one of the most common tools to analyse functional data. Some examples are Besse et al. (1997); Cardot et al. (2004); Locantore et al. (1999); James et al. (2000); Viviani et al. (2005). More comprehensive application examples for fPCA could be found in Ramsay and Silverman (2005).

We implement functional principal component analysis on the functional predictors $X_{ij}(\cdot)$'s, so that each predictor $X_{ij}(\cdot)$ has a unique score vector, $\xi_{ij} = (\xi_{ij1}, \xi_{ij2}, \dots, \xi_{ijL_j})$, where $\xi_{ijk} = \int \phi_k(s)X_{ij}(s)ds, k = 1, 2, \dots, L_j$. Here, $\phi_k(\cdot)$'s are orthogonal basis functions. In this way, each curve could be approximated by $X_{ij}(\cdot) = \sum_{k=1}^{L_j} \xi_{ijk}\phi_k(\cdot)$, where L_j could be chosen such that the first L_j principal components can explain 95% variance. More

technical details about functional PCA could be found in Ramsay and Silverman (2005). By using functional PCA, we transform the infinite-dimensional predictors $X_{ij}(\cdot)$'s to multi-dimensional score vectors. Next, we will consider how to capture the nonlinear relationship between response and functional predictors.

Kernel Machine

Now, each $X_{ij}(\cdot)$ is represented by its own score vector ξ_{ij} , and next we need to handle the nonlinear relationship between response and functional predictors. Instead of modeling the relationship between Y_i and $X_{ij}(\cdot)$'s, we model the relation between Y_i and the score vector f_{ij} . We consider the following model:

$$Y_i = \alpha + \sum_{j=1}^p g_j(\xi_{ij}) + \epsilon_i,$$

where $g_j(\cdot)$'s are unknown functions that quantify the relationship between Y_i and the score vector of $X_{ij}(\cdot)$ independent Gaussian processes with mean zero and covariance function $K(\cdot, \cdot)$, that is, $g_j(\cdot) \sim GP(0, \tau_j K_j(\cdot, \cdot))$. Without loss of generality, we assume Y_i 's have mean zero so that $\alpha = 0$.

A Gaussian process is a statistical distribution for which any finite linear combination of samples has a joint Gaussian distribution. It has been widely used in machine learning (Rasmussen and Williams 2006), functional data analysis (Shi and Wang 2008; Shi and Choi 2008; Gramacy and Lian 2012; Wang and Shi 2014) and spatial statistics (Moller and Waagepetersen 2003; Rue and Held 2005; Banerjee et al. 2014). We assume a Gaussian process model here to capture both linear and nonlinear relationship between response and functional covariates.

Here, we use kernel function as covariance function $K(\cdot, \cdot)$ to measure pairwise similarity between predictors among all subjects. Notice that we won't focus on the estimation of

the Gaussian process. Commonly used kernels are the d th Polynomial Kernel: $K(z_1, z_2) = (z_1^T z_2 + 1)^d$ and the Gaussian Kernel: $K(z_1, z_2) = \exp(-\|z_1 - z_2\|^2/\rho)$. Here, we use the d th Polynomial Kernel with $d = 1, 2, 3$.

Penalized Similarity-based Regression

So far, Similarity-based regression has been widely used in genetics. Qian and Thomas (2001) quantify the similarities of phenotypes and of haplotypes. Beckmann et al. (2005) and Wessel and Schork (2006) propose regression models that correlate trait similarity with genetic similarity. Tzeng and Zhang (2007) propose a similarity-based regression method to detect associations between traits and multimarker genotypes. As far as we are concerned, this is the first time that similarity regression is used to functional data analysis.

Assuming $Y_i, i \in \{1, \dots, n\}$ has mean zero, we notice that covariance between Y_i and Y_j has the following form:

$$E(Y_i Y_j) = E\left[\left(\sum_{l=1}^p g_l(f_{il}) + \epsilon_i\right)\left(\sum_{l=1}^p g_l(f_{jl}) + \epsilon_j\right)\right] = \sum_{l=1}^p \tau_l^2 K_l(f_{il}, f_{jl}) \quad (1.1)$$

With this variance component relationship, the covariance of Y_i and Y_j is $E[Y_i Y_j | \tilde{f}] = \tau_1^2 S_{ij,1} + \dots + \tau_p^2 S_{ij,p}$, where $S_{ij,l} = K_l(f_{il}, f_{jl}), l = 1, 2, \dots, p$.

We thus propose a similarity regression model of the following form to study the variance component relationship:

$$Z_{ij} = \tau_1^2 S_{ij,1} + \dots + \tau_p^2 S_{ij,p} + \epsilon_{ij}, \forall i \leq j,$$

where $Z_{ij} = Y_i Y_j$ and ϵ_{ij} has mean zero and finite variance. For some kernels such as quadratic kernel or Gaussian kernel, the kernel value $K_l(\cdot)$ can only be positive, considering the fact that τ^2 is non-negative, here we center $S_{ij,l}$. Otherwise, the relationship between

the negative part of left-hand side and right-hand side can not be captured by the model.

Now we transform the nonparametric Gaussian process model to the parametric similarity regression model, and then we conduct variable selection via the elastic net regularization (Zou and Hastie 2005). If there is a group of variables among which the pairwise correlations are very high, such as some variables in our real data application, then the elastic net tends to select groups of correlated variables. In addition, since we are modeling variance components, we constrain the coefficients to be non-negative.

Remark: So far, we assume X'_{ij} s are observed without measurement errors. In fact, functional PCA will deal with the case where X'_{ij} s are observed with measurement errors.

1.3 Simulation Study

We conduct simulations in a variety of settings to illustrate the performance of the proposed method in terms of sensitivity and specificity. In this section, we summarize results based on data sets of the form $[\{X_{ij}(t) : t \in T_j\}]$, where T_j is taken to be the set of 300 equidistant points in $(0,300)$. Define for $i = 1, \dots, 300$ and $j = 1, \dots, 30$,

$$X_{ij}(t) = \{\sigma(t)^{-1} \sum_{r=1}^5 (a_{ijr} \sin(\pi t(5 - a_{ijr})/150) - m_{ijr})\}$$

, where $a_{ijr} \sim U(0, 5)$, $m_{ijr} \sim U(0, 2\pi)$, with $U(a, b)$ denoting the uniform distribution on interval $[a, b]$. Here $\sigma(t)$ is defined such that $\text{var}\{X_{ij}(t)\} = 0.01$ for all $t \in (0, 300)$.

We consider both functional linear model and functional nonlinear model with scalar response.

1. Functional linear model, and in particular $Y_i = \sum_{j=1}^p \int X_{ij}(t) \beta_j(t) dt + \epsilon_i$, where $\epsilon_i \sim N(0, 1)$, and $\beta_1(t), \beta_2(t)$ have Gamma-density like shape with effect sizes decreasing with increasing j .

2. Functional nonlinear model, $Y_i = (\int X_{i1}(t)\beta_1(t)dt)^2/5 + (\int X_{i2}(t)\beta_2(t)dt)^3/60 + \epsilon_i$. $\beta_j(t)$'s are chosen as same as in (1).

We set $n = 50, 100$, respectively. Note that only signals $j = 1, 2$ are assumed to be relevant, we simulate with 3 noises, 8 noises, respectively. Therefore, there are in total 8 simulation settings. For each simulation setting, we use 4 different kernels (linear, quadratic, cubic and Gaussian kernel) and run 1000 replications. We also consider the combined results from the 4 kernels, that is, if a signal is picked up by at least one of the 4 kernel, then it adds up to a count. We calculate the proportion of each variable being picked up and the average model size (average number of selected variables, true model size should be 2). The results are listed in Table 1 and Table 2. From Table 1, when the relationship between response and functional predictors are linear, linear kernel gives the best results. Quadratic and cubic kernel also works but they both lose some power compared with linear kernel. In general, our method works very well for linear case, the average size is close to 2, which is the number of the true signals in the simulation setting. From Table 2, for nonlinear case, linear kernel can barely capture the signals, while quadratic and cubic kernel work well. Especially when quadratic kernel is used, it can capture the quadratic-form signal well while can't pick up cubic-form signal so often. However, if cubic kernel is used, it can capture both quadratic-form signal and cubic-form kernel, but somehow loses some power. In general, cubic kernel works best in the nonlinear case. We also find that by combining the three kernels, we can get a better average size, meaning true signals will be chosen more often. With that being said, we recommend running algorithms with choice of all the four kernels and make variable selection decisions based on the combined results.

We then compare our method (KM) with Gertheiss et al. (2013) (grLasso) and Yingying Fan and Radchenko (2015) (FAR), with respect to false positive rate (FPR), false negative rate (FNR), average model size and Matthews correlation coefficient (MCC). False positive

Table 1.1: Simulation Results for Linear Case

p	n	Kernel	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	Size
5	50	Linear	0.965	0.98	0	0	0	-	-	-	-	-	1.945
		Quadratic	0.428	0.474	0	0.001	0	-	-	-	-	-	0.903
		Cubic	0.670	0.648	0.002	0	0.002	-	-	-	-	-	1.322
		Gaussian	0.766	0.773	0.021	0.020	0.014	-	-	-	-	-	1.594
		Combined	0.979	0.986	0.021	0.021	0.014	-	-	-	-	-	2.022
100	50	Linear	1	1	0	0	0	-	-	-	-	-	2
		Quadratic	0.753	0.784	0	0	0	-	-	-	-	-	1.537
		Cubic	0.880	0.890	0	0	0	-	-	-	-	-	1.770
		Gaussian	0.941	0.944	0.006	0.006	0.011	-	-	-	-	-	1.908
		Combined	1	1	0.006	0.006	0.011	-	-	-	-	-	2.023
10	50	Linear	0.957	0.975	0	0	0	0	0	0	0	0	1.933
		Quadratic	0.436	0.461	0	0.001	0	0	0	0.001	0.001	0	0.9
		Cubic	0.652	0.647	0.003	0.002	0.002	0.004	0.002	0.003	0.002	0	1.317
		Gaussian	0.770	0.764	0.020	0.022	0.013	0.023	0.022	0.015	0.015	0.013	1.677
		Combined	0.983	0.986	0.021	0.023	0.013	0.023	0.022	0.016	0.016	0.013	2.116
100	50	Linear	0.998	1	0	0	0	0	0	0	0	0	1.998
		Quadratic	0.778	0.795	0	0	0	0	0	0	0	0	1.573
		Cubic	0.880	0.882	0.001	0	0	0	0	0	0.001	0	1.764
		Gaussian	0.946	0.936	0.006	0.005	0.010	0.002	0.005	0.010	0.006	0.009	1.935
		Combined	0.999	1	0.006	0.005	0.010	0.002	0.005	0.010	0.006	0.009	2.052

Table 1.2: Simulation Results for Nonlinear Case

p	n	Kernel	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	Size
5	50	Linear	0.026	0.482	0.005	0.007	0.005	-	-	-	-	-	0.525
		Quadratic	0.820	0.284	0.020	0.021	0.010	-	-	-	-	-	1.155
		Cubic	0.595	0.605	0.011	0.018	0.018	-	-	-	-	-	1.247
		Gaussian	0.579	0.589	0.012	0.018	0.013	-	-	-	-	-	1.211
		Combined	0.874	0.798	0.032	0.038	0.027	-	-	-	-	-	1.769
100	50	Linear	0.01	0.673	0.002	0.005	0.005	-	-	-	-	-	0.695
		Quadratic	0.902	0.339	0.005	0.010	0.006	-	-	-	-	-	1.262
		Cubic	0.718	0.817	0.003	0.006	0.003	-	-	-	-	-	1.547
		Gaussian	0.724	0.813	0.007	0.008	0.003	-	-	-	-	-	1.555
		Combined	0.932	0.925	0.008	0.019	0.013	-	-	-	-	-	1.897
10	50	Linear	0.026	0.511	0.004	0.008	0.008	0.005	0.013	0.007	0.010	0.004	0.596
		Quadratic	0.828	0.294	0.013	0.018	0.011	0.024	0.012	0.019	0.016	0.013	1.248
		Cubic	0.564	0.582	0.011	0.018	0.013	0.017	0.015	0.016	0.012	0.015	1.263
		Gaussian	0.565	0.586	0.013	0.016	0.011	0.014	0.014	0.014	0.011	0.014	1.258
		Combined	0.875	0.791	0.022	0.039	0.027	0.037	0.030	0.031	0.027	0.027	1.906
100	50	Linear	0.013	0.656	0.004	0.006	0.003	0.006	0.004	0.006	0.005	0.006	0.709
		Quadratic	0.902	0.342	0.005	0.011	0.006	0.009	0.009	0.007	0.010	0.007	1.308
		Cubic	0.711	0.811	0.002	0.005	0.004	0.007	0.006	0.005	0.007	0.007	1.565
		Gaussian	0.718	0.802	0.002	0.003	0.004	0.006	0.005	0.007	0.004	0.009	1.56
		Combined	0.937	0.908	0.011	0.020	0.012	0.019	0.017	0.014	0.020	0.015	1.9

Table 1.3: Method Comparisons for Linear Case. KM refers to our Kernel Machine method, FAR refers to Functional Additive Regression method and grLasso refers to Gertheiss et al. (2013)’s method. False negative rate, false positive rate, average model size and Matthews correlation coefficient are compared.

p	n	Method	FNR	FPR	Size	MCC
5	50	KM	0.018	0.019	2.022	0.96
		FAR	0	0.073	2.220	0.91
		grLasso	0	0.349	3.048	0.65
100	50	KM	0	0.008	2.023	0.99
		FAR	0	0.030	2.090	0.96
		grLasso	0	0.529	3.588	0.51
10	50	KM	0.016	0.018	2.116	0.95
		FAR	0	0.090	2.720	0.82
		grLasso	0	0.344	4.753	0.53
100	50	KM	0.001	0.007	2.052	0.98
		FAR	0	0.009	2.070	0.98
		grLasso	0	0.528	6.227	0.39

rate records the fraction of noise predictors incorrectly included in the model. False negative rate corresponds to the fraction of signal variables incorrectly excluded. Matthews correlation coefficient is defined as followed:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

where TP, TN, FP and FN represent true positive rate, true negative rate, false positive rate and false negative rate, respectively. The results are shown in Tables 3 - 4. For linear case, both FAR and grLasso have FN rate zero, while our KM has a small FN rate close to zero. KM outperforms FAR and grLasso with respect to FPR, model size and MCC. As for nonlinear case, since grLasso is not applicable, we compare KM only with FAR. FAR always has smaller FN rate than KM while KM has smaller FP rate than FAR. In addition, KM has a model size closer to 2 and has a higher MCC than FAR.

Table 1.4: Method Comparison for Nonlinear Case. KM refers to our Kernel Machine method, FAR refers to Functional Additive Regression method and grLasso refers to Gertheiss et al. (2013)’s method. False negative rate, false positive rate, average model size and Matthews correlation coefficient are compared.

p	n	Method	FNR	FPR	Size	MCC
5	50	KM	0.164	0.032	1.769	0.82
		FAR	0.015	0.277	2.800	0.70
	100	KM	0.072	0.013	1.897	0.92
		FAR	0	0.093	2.280	0.89
10	50	KM	0.167	0.030	1.906	0.82
		FAR	0.005	0.196	3.560	0.67
	100	KM	0.078	0.016	1.900	0.88
		FAR	0	0.090	2.720	0.82

1.4 Real Data Application

1.4.1 Overview of Dataset

We are interested in identifying forearm muscles that contribute to specific hand movements. These muscles are known for subjects with an intact limb but they may no longer hold for subjects amputated below the wrist. Such amputees will still have residual forearm muscles that previously contributed to hand movement and will experience sensations that their missing hand is moving. This is proof that after amputation, amputees still have a mental model for controlling their amputated hand. This mental model when physically transferred to the amputated limb, however, may be altered compared to when the limb was intact.

Our goal is to develop an objective muscle-selection algorithm that leads to successful identification of the hand-movement muscles for amputees. The primary application is to create a high-fidelity robotic prosthetic for amputees utilizing EMG from the intact forearm muscles. This algorithm should work well for both able-bodied and amputated subjects

and we focus our attention on the former for this paper.

Any particular hand motion can involve a complicated sequence of muscle contractions. To simplify the problem, we chose to focus on two degrees of freedom of movement: finger/wrist flexion and extension. More specifically, finger movement is simultaneous flexion/extension of all fingers except for the thumb. We ignore individual finger movement because it involves more muscles than those found in the forearm. Finger flexion results in a fist and extension yields an open hand. Wrist flexion brings the hand closer to the inside of the forearm and extension moves it away from the inside forearm. These two degrees of freedom of hand movement account for most grasping movements and other useful day-to-day movements that an amputee may not be able to perform with a static hand prosthesis.

For an able-bodied person, intentional (i.e. premeditated) hand movement starts with a neural signal sent from the brain to muscles in the forearm leading to muscle contraction. These muscles are connected to tendons which travel through the wrist and are attached to bones in the hand. Contraction of a muscle moves the tendons which potentially leads to finger or wrist movement. We say potentially because in some cases muscle contraction is necessary to stabilize the hand.

It is best to think of a muscle contraction as generating a force rather than thinking of the muscle contraction as leading to a specific position or velocity of the fingers or hand. For example, a muscle controlling finger flexion could be contracted to decelerate the rate of finger extension. That is, we are simultaneously flexing and extending our fingers. The hand and wrist movements also clearly have an upper and lower bound; the fingers and wrist may be flexed/extended so much. To maintain a full flexion or extension, one must maintain muscle contraction that got the fingers/wrist to the position.

Contraction of a muscle is accompanied by an electrical signal, called electromyography (EMG), which can be measured using a surface EMG sensor. Accurately measuring EMG

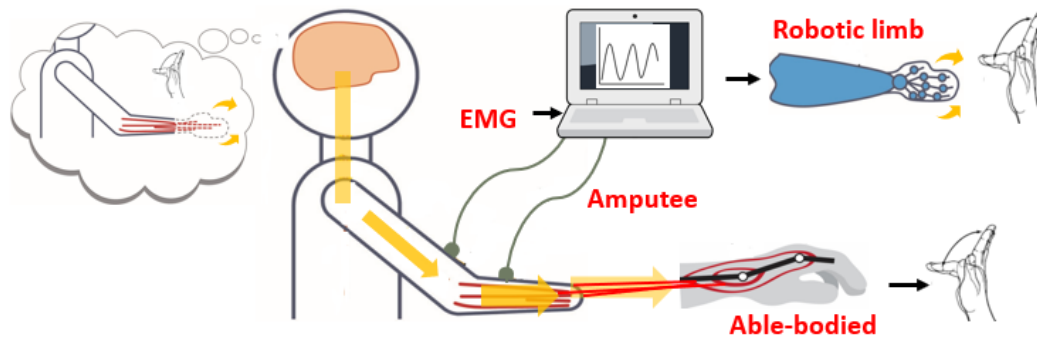


Figure 1.1: The biomechanical system for hand movement for able-bodied subject and transradial amputee.

can be a problem because forearm muscles are overlapping. The sensor will be better able to detect EMG from surface muscles, but it may also pick up EMG from neighboring muscles and muscles underneath these surface muscles, a problem often referred to as “crosstalk”. Fine-wire EMG sensors which are directly inserted into the surface muscles are able to avoid this issues but these are very intrusive and impractical for daily prostheses. Fortunately, muscles that are unintentionally stretched do not yield EMG.

Data were collected at 120Hz (120 measurements per second, equally spaced) from an able-bodied subject asked to repeatedly perform specific hand movements for a certain period of time. The subject had 16 EMG sensors attached to their forearm and motion capture sensors were placed on their corresponding hand and wrist. The motion capture software calculated the position (in radians) of the hand and wrist after choosing a reference location (need picture). Two types of movements were performed: intentional finger flexion/extension and intentional wrist flexion/extension. In both cases, the other type of movement was not intentionally manipulated. Furthermore, the movement was done so that at a given time only the corresponding flexor/extensor muscle was contracted. This allowed us to focus on modeling the movement’s velocity instead of acceleration.

Raw EMG signals were collected at a higher frequency than 120Hz but were very noisy.

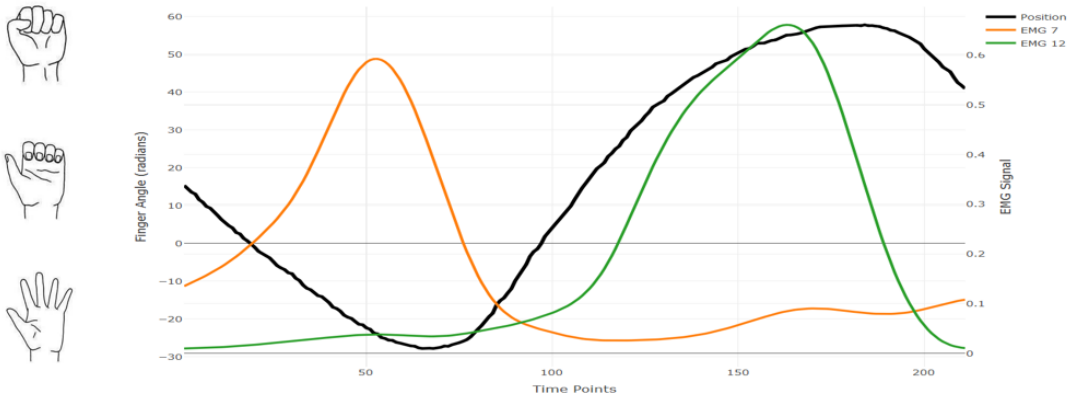


Figure 1.2: Joint finger angle (radians) for flexion and extension movements, in black, and two normalized EMG signals, in orange and green.

The signals were preprocessed prior to analysis using an envelope function. That is, raw EMG signals are pre-smoothed and it is these pre-smoothed values (at concurrent time points as the 120Hz motion data) that are used in the analysis. Part of the smoothing of the EMG signals is normalization between 0 and 1 for each subject who are asked to generate maximal force for hand and wrist movement prior to movement data collection. An EMG signal of 0 means no muscle contraction while a signal of 1 means the subject is maximally contracting that muscle.

1.4.2 Results

In this section, we present the variable selection results for our data collected from an able-bodied subject across multiple movement patterns as described in section 4.1. Recall there are 16 EMG signals: 14 coming from different forearm muscles that could potentially contribute to finger or wrist movements and 2 that are randomly generated noise. The relevant muscles for finger and wrist movements for an able-bodied person are discovered clinically. For finger movements, the EMG signal contributing to flexion is EMG 12; EMG 5 and 7 contribute to finger extension. For wrist movement, EMG 8,10,11,14 correspond

to flexion and EMG 2,7,13,15 are important to extension. In this application, signals from a specific movement class are highly correlated. The expert consensus is that the ideal selection should pick at least one signal for flexion and at least one for extension as representatives, for both finger and wrist movements. We are not intended to select all the important variables, one from each category is enough to depict the relationship between hand movement and EMG signals.

One constraint we consider when post-processing the data is that, muscles generate passive movement forces when stretched, which may produce movement in the absence of EMG information. For instance, if one relaxes their forearm muscles following a contraction, the tendons will return back to their resting length, generating a passive force, and the hand will return to a neutral configuration. Therefore, we may observe movement without observing concurrent EMG activation. While this tendon connection no longer exists for a TRA, they may still anticipate these passive forces. To accommodate this constraint, we use finger/wrist velocity as response and utilize recent past behavior of EMG signals. The velocity values are estimated from a penalized smoother of the entire series of recorded position data across time using the R package `fda`. In particular, we use a second-order smooth regularization penalty to control the goodness of fit and smoothness of the fitted curve, where the smoothing parameter is selected by cross-validation. We then generate a velocity estimate for each observed position data point. For each velocity data point, we extract the previous EMG observations that end with the concurrent EMG value to the velocity value. In this application, we choose a past time window of roughly 0.33 seconds. The value is chosen based on observed passive force movement. This is done for all EMG signals, so each velocity estimate is associated with past measurements across 0.33 seconds for all 16 EMG signals collected.

As suggested in simulation study, we run algorithms with choice of all the four kernels and make variable selection decisions based on the combined results. There are 6 datasets

Table 1.5: Variable Selection Results for EMG Data

Signal	Finger	Wrist
1	0.07	0.27
2	0.29	0.54
3	0.09	0.04
4	0	0.12
5	1	0.36
6	0.13	0.13
7	1	0
8	0	0.21
9	0.05	0.09
10	0	0.80
11	0.02	1
12	1	0
13	0	0
14	0	0
15	0	1
16	0	0.04

for finger movement and 6 datasets for wrist movement. We record proportion of pickups for each of the 16 signals in Table 5. Our KM algorithm pick up all the three signals that contribute to finger movement, and it selects only a small proportion of uncorrelated signals. For wrist flexion, KM selects EMG 10 80% of the time and EMG 11 all the time and for wrist extension, KM selects EMG 2 54% of the time and EMG 15 all the time.

1.5 Discussion

In this paper, we consider a flexible functional additive model, use functional PCA to convert functional predictors to multivariate score vectors. Then we use kernel machine to capture similarity between those functional predictors among all subjects. At last, we model the variance component relationship using penalized similarity-based regression to conduct

variable selection. Systematic simulation studies show that our method can handle both linear and nonlinear problems, and outperform the state-of-the-art variable selection methods. Our algorithm is applied to the prosthetic arm control data, and is able to select desired variables. With those advantages and accomplishment being said, there are some aspects in the paper that need further study. In Section 1.2.2, Z_{ij} is defined as $Z_{ij} = Y_i Y_j$, which means Z_{ij_1} and Z_{ij_2} for some j_1, j_2 can be correlated. We thus assume the error term ϵ_{ij} has mean zero and finite variance, but not independent. From an optimization perspective, this assumption of ϵ_{ij} does not affect the numerical solution for Lasso or Lasso related models, such as SCAD (Fan and Li 2001), Elastic net (Zou and Hastie 2005) and the adaptive Lasso (Zou 2006). However, a fair amount of oracle properties of L1 regularization rely on the assumption that the error term is independent, identical and has finite variance. Few work has been done under the setting of correlated random errors. Therefore, properties under this assumption need further study. In addition, our method requires computation of $n(n+1)/2$ responses with a sample size of n , which can be massive for moderately large data. A potential direction for solving this issue is the orthogonalizing EM algorithm (Xiong et al. 2016), which is designed for massive tall data. What is more, the choice of kernel in the algorithm is somehow ad-hoc and can affect the accuracy of variable selection results. Also, this method focuses only on variable selection, future work will consider estimation of parameters in the model.

CHAPTER

2

CONSTRAINED BAYESIAN
NONPARAMETRIC REGRESSION FOR
GRAIN BOUNDARY ENERGY
PREDICTIONS

2.1 Introduction

While the role of the structure of grain boundaries (GBs) in various transport and failure mechanisms in polycrystalline materials has been investigated for more than half a century (Forsyth et al. 1946; Smoluchowski 1952; Haynes and Smoluchowski 1955; Hirth 1972; Hunderi 1973; Chadwick and Smith 1976; Gleiter 1981, 1982; Dimos et al. 1990; Sutton and Balluffi 1995; Gottstein and Shvindlerman 2009), the lack of robust GB structure-property relationships still remains one of the biggest obstacles towards developing true bottom-up models for the behavior of polycrystalline materials (Panchal et al. 2013). This is because of the inherent complexity associated with the structure of interfaces and the vast five-dimensional configurational space in which they reside (Morawiec 2003; Patala et al. 2012; Patala and Schuh 2013). Reliable crystallography-structure-property relationships for interfaces are particularly important for structural materials operating under extreme environments, such as high temperatures, high strain rates and dynamic loading conditions.

More recently, however, advances in both experimental and computational techniques have facilitated large databases of GB properties (Olmsted et al. 2009a,b; Holm et al. 2010; Rohrer 2011; Homer et al. 2014) in the five-parameter crystallographic phase-space. The five macroscopic degrees of freedom (d.o.f) refer to the misorientation (three parameters) and the boundary-plane orientation (two parameters) aspects of the GB. With the advent of modern high-throughput algorithms (Jain et al. 2011; Curtarolo et al. 2013; Jain et al. 2013; Saal et al. 2013) and sophisticated experimental techniques (Seita et al. 2016), we have reached a point where the development of new statistical tools is critical for the analysis of the vast amounts of data being generated, for developing novel scientific insights and for building predictive models essential for the advancement of the field of GB science and engineering.

One of the earliest high-throughput experimental techniques for the measurement of

GB properties is related to the relative energy distributions of GBs in the five-parameter crystallographic phase-space. Experimental measurements of GB energies rely on the Herring equation (Herring 1951) that describes the equilibrium condition of a triple-junction. For example, GB energies for copper and aluminum were computed at high temperatures using the thermal grooving measurements, where two free surfaces and a GB are in equilibrium (Hasson and Goux 1971; Miura et al. 1994). Similarly, triple junction geometries are determined to compute relative energies of experimentally observed interfaces (Adams et al. 1999; Rollett et al. 2001). The advent of automated acquisition of large data sets of 3D EBSD data has facilitated a sampling of triple junction geometries to evaluate the relative energies of a large number of GBs (Morawiec 2000). Using this technique relative GB energies have been computed for different structural metallic systems such as nickel (Li et al. 2009), aluminum Barmak et al. (2005, 2006), ferritic and austenitic steels (Beladi and Rohrer 2013; Beladi et al. 2014); and ceramic materials including magnesia (MgO) (Saylor et al. 2002, 2000, 2003) and yttria (Y_2O_3) (Dillon and Rohrer 2009; Bojarski et al. 2013, 2012). The statistical distributions of different GB types (the GB character distribution) and the relative energies are hosted online by Prof. Gregory S. Rohrer's group at http://mimp.materials.cmu.edu/~gr20/Grain_Boundary_Data_Archive/. Morawiec (2000) presents a numerical method (referred to as block aggregation in this paper) for reconstructing the grain boundary energy distribution over the complete space of macroscopic boundary parameters. The method assumes that triple junctions are in local equilibrium, which is described by the Herring equation. The method discretizes the five-dimensional space and solves a homogeneous system of algebraic linear equations.

In this paper, we propose a new nonparametric Bayesian model to reconstruct and predict grain boundary energy. The method is based on generalized additive model (Hastie 2017), which is a generalized linear model with a linear predictor involving a sum of smooth functions of covariates. Each smooth function is defined by some basis functions, such

as B-spline basis, polynomial basis and Gaussian basis. GAMs have been proven to be extremely useful in analyzing data in complex domains (Walczak and Massart 1996; Schölkopf et al. 1997; Ramamoorthi and Hanrahan 2001). However, applying GAM to this problem is challenging because the physical properties of the GBs imply numerous constraints in the response surface that must be incorporated into the GAM model. In addition, there are not direct measurements on the grain boundary energy. We only have indirect information in the form of a homogeneous set of linear equations. We incorporate the constraints by implementing Hamiltonian Monte Carlo (Duane et al. 1987) sampling and Gibbs sampling for posterior computation. The constraints enable the estimates of grain boundary energy identifiable. Our constrained Bayesian nonparametric regression (CBNR) model outperforms the block aggregation (BA) method with respect to prediction accuracy. Our method also gives prediction intervals. This is the first time that GB energy uncertainties are quantified.

The remainder of the paper is organized as follows. Section 2.2 introduces notation and the equations that define triple junctions. Section 3 presents our model and computational details. In Section 4, the method is compared with BA via a simulation study. Section 5 analyzes experimental data. Section 6 summarizes the paper and discusses future work.

2.2 Analyzing the 3D EBSD Triple Junction Data

The dataset has $n = 19,094$ triple junctions. Figure 3.1 shows an example of one triple junction. For each triple junction, three grain boundaries are involved. For each grain boundary, the 3×3 grain orientation matrix \mathbf{O} , the 3×1 boundary-plane orientation vector $\hat{\mathbf{n}}$ and the 3×1 tangent vector for the triple junction line \mathbf{l} are given in the dataset. The grain boundary misorientation matrix between grain boundary \mathbf{b}_i and \mathbf{b}_j is then defined as $\mathbf{M}_{ij} = \mathbf{O}_i^{-1}\mathbf{O}_j$. The boundary-plane crystallography is thus defined by \mathbf{M} and $\hat{\mathbf{n}}$. Further, the grain

boundary misorientation matrix \mathbf{M} can be transformed to a 3×1 vector \mathbf{m} , therefore, each grain boundary can be defined by five parameters: $\mathbf{b} = (\mathbf{m}, \hat{\mathbf{n}})$. We use the five-dimensional parameter \mathbf{b} for the data analysis. Finally, the grain boundary $\gamma(\mathbf{b})$ is defined as

$$\gamma(\mathbf{b}) = \hat{\mathbf{n}} \cdot \xi(\mathbf{b}),$$

where $\xi(\mathbf{b})$ is the 3×1 capillarity vector, which is unknown.

In addition, we consider two reference frames in this paper:

1. *Crystal Reference Frame*: This refers to the Cartesian coordinate axes of the crystal (or grain) from which the vector, rotation matrices are described. We use a superscript c to denote such quantities (e.g. $\mathbf{O}^c, \hat{\mathbf{n}}^c, \xi^c$ etc.)
2. *Lab/sample Reference Frame*: This refers to the fixed lab reference frame (usually fixed internally by the sample/detector geometry) from which the vector, rotation matrices are described. We use a superscript s to denote such quantities (e.g. $\mathbf{O}^s, \hat{\mathbf{n}}^s, \xi^s$ etc.)

The equation relating the capillarity vectors ξ and the tangent vector \mathbf{t} of the junction

$$(\xi^s(\mathbf{b}_1) + \xi^s(\mathbf{b}_2) + \xi^s(\mathbf{b}_3)) \times \mathbf{I}^s = 0 \quad (2.1)$$

The energy of the grain boundary \mathbf{b}_i is given by $\gamma(\mathbf{b}_i) = \xi^s(\mathbf{b}_i) \cdot \hat{\mathbf{n}}_i^s = \xi^c(\mathbf{b}_i) \cdot \hat{\mathbf{n}}_i^c$.

In order to apply the condition that the function defining the capillarity vectors is continuous, they have to be expressed in the crystal reference frame, i.e., we rewrite the capillarity vector in (1) using the crystal reference frame. This is given by:

$$((\mathbf{O}_1^s)^T \xi^s(\mathbf{b}_1) + (\mathbf{O}_2^s)^T \xi^s(\mathbf{b}_2) + (\mathbf{O}_3^s)^T \xi^s(\mathbf{b}_3)) \times \mathbf{I}^s = 0 \quad (2.2)$$

We will not use superscript c and s representing reference frame for the rest of the paper. Our

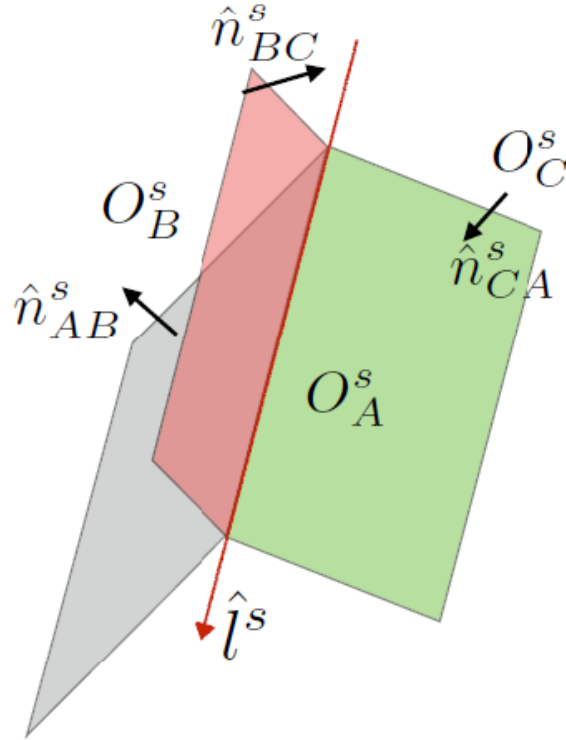


Figure 2.1: Illustrative example of one triple junction consisting of three grain boundaries: A (grey), B (red) and C (green). \mathbf{O}^s is the grain orientation matrix, $\hat{\mathbf{n}}^s$ is the boundary-plane orientation vector and $\hat{\mathbf{l}}^s$ is the tangent vector for the triple junction line.

goal is to recover and predict grain boundary energies over the five-dimensional parameter space. Notice that the grain boundary energy is defined by the unknown capillarity vector ξ and known boundary-plane orientation vector $\hat{\mathbf{n}}$. Therefore, instead of modeling directly on grain boundary energy $\gamma(\cdot)$, we model on the capillarity vector ξ . The challenge is that there are not direct measurements on either the grain boundary energy or the capillarity vector. We only have indirect information in the form of a homogeneous set of linear equations as in (1) and (2). In next section, we will present our model and computation details to solve this problem.

2.3 Statistical model

2.3.1 Model description and prior specification

Using the notation defined in Section 2, for triple junction $i = 1, 2, \dots, n$ we have

$$[\mathbf{O}_{i1}^T \boldsymbol{\xi}(\mathbf{b}_{i1}) + \mathbf{O}_{i2}^T \boldsymbol{\xi}(\mathbf{b}_{i2}) + \mathbf{O}_{i3}^T \boldsymbol{\xi}(\mathbf{b}_{i3})] \times \mathbf{l}_i = \mathbf{0}, \quad (2.3)$$

where \mathbf{O}_{ij} is the 3×3 grain orientation matrix, \mathbf{b}_{ij} is the 5×1 boundary-plane crystallography vector, and \mathbf{l}_i is the 3×1 tangent vector for triple junction i , $u \times v$ is the outer product of u and v . The goal is to estimate the unknown capillarity vectors $\boldsymbol{\xi}(\mathbf{b}) = [\xi_1(\mathbf{b}), \xi_2(\mathbf{b}), \xi_3(\mathbf{b})]^T$.

In matrix form, we can re-write (3) as

$$\mathbf{A}_i [\boldsymbol{\xi}(\mathbf{b}_{i1})^T, \boldsymbol{\xi}(\mathbf{b}_{i2})^T, \boldsymbol{\xi}(\mathbf{b}_{i3})^T]^T = \mathbf{0}, \quad (2.4)$$

where A_i is a 3×9 matrix consisting of coefficients corresponding to the i^{th} triple junction. Combining all the n triple junctions, we have:

$$\mathbf{A} \boldsymbol{\xi} = \mathbf{0}, \quad (2.5)$$

where \mathbf{A} is a $3n \times 9n$ diagonal block matrix with diagonal blocks \mathbf{A}_i and

$\boldsymbol{\xi} = [\xi^T(\mathbf{b}_{11}), \xi^T(\mathbf{b}_{12}), \xi^T(\mathbf{b}_{13}), \dots, \xi^T(\mathbf{b}_{n1}), \xi^T(\mathbf{b}_{n2}), \xi^T(\mathbf{b}_{n3})]^T$ is the $9n \times 1$ vector of unknown capillary.

We assume the components \mathbf{O}_{ij} , \mathbf{b}_{ij} and \mathbf{l}_i are measured with error. Since it is impossible to estimate the errors associated with each component, we simply assume the random error model

$$\mathbf{A} \boldsymbol{\xi} + \boldsymbol{\delta} = \mathbf{0},$$

where $\boldsymbol{\delta} \sim N(0, \sigma^2 \mathbf{I})$. This is equivalent to the model

$$\mathbf{Y}|\boldsymbol{\xi} \sim N(\mathbf{A}\boldsymbol{\xi}, \sigma^2 \mathbf{I}),$$

where \mathbf{Y} is a zero vector.

To model the underlying capillary process, suppose

$$\boldsymbol{\xi}(\mathbf{b}) = \mathbf{f}(\mathbf{b}) + \boldsymbol{\epsilon}, \quad (2.6)$$

where $\mathbf{f}(\mathbf{b}) = (f_1(\mathbf{b}), f_2(\mathbf{b}), f_3(\mathbf{b}))^T$ and $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_{3 \times 3})$. The smooth function \mathbf{f} can be approximated by any nonparametric basis functions, such as B-splines and Fourier functions, etc. For example, we decompose \mathbf{f} as the sum of main-effect functions: $f(\mathbf{b}) = \sum_{i=1}^{L=5} g_i(b_i)$, where $g_i(\mathbf{b}) = (g_{i1}(\mathbf{b}), g_{i2}(\mathbf{b}), g_{i3}(\mathbf{b}))$, is the i^{th} additive main effect; here $L = 5$ since \mathbf{b} is five dimensional. Assuming the main-effect functions are sufficiently smooth, they can be approximated by B-spline basis expansions with B-spline basis functions $B_1(x), \dots, B_m(x)$. The main effect approximation is then $g_i(b_j) \approx \sum_{r=1}^m B_r(b_j) \beta_{ijr}$. Therefore, the regression of $\xi_i(\cdot)$, $i = 1, 2, 3$ can be modeled as

$$\xi_i(\mathbf{b}) = \sum_{j=1}^5 \sum_{r=1}^m B_r(b_j) \beta_{ijr} + \epsilon_i = \mathbf{B}(\mathbf{b}) \boldsymbol{\beta}_i + \epsilon_i.$$

The unknown coefficients are assigned non-informative normal priors, $\beta_{ijr} \sim N(0, \lambda)$ for large λ .

The complete Bayesian hierarchical model is

$$\mathbf{Y}|\boldsymbol{\xi}, \boldsymbol{\beta}, \sigma^2 \sim N(\mathbf{A}\boldsymbol{\xi}, \sigma^2 \mathbf{I})$$

$$\boldsymbol{\xi} \sim N(\mathbf{B}(\mathbf{b})\boldsymbol{\beta}, \mathbf{I} \otimes \boldsymbol{\Sigma}),$$

where \otimes represents Kronecker product, $\sigma^2 \sim \text{InvGamma}(a, b)$, $\Sigma \sim \text{InvWishart}(\Phi, df)$, and $\beta_{ijr} \sim N(0, \lambda)$, $i = 1, 2, 3$, $j = 1, \dots, 5$, $r = 1, \dots, m$. The hyperparameters a , b , Φ , df and λ are set to give uninformative priors, as described in Section 3.3.

2.3.2 Constraints

In (3), the scale of ξ is not identified because if $\mathbf{A}\xi_0 = 0$, then $\mathbf{A}(c\xi_0) = 0$ for any c . Also, depending on the rank of \mathbf{A} the linear equation system may have infinitely many solutions and $\xi = \mathbf{0}$ is always a solution. We impose the following constraints to ensure the capillary vector identified.

1. Assume the L_2 norm of capillary vector is set to be greater or equal to a constant, that is,

$$\|\xi_{9n \times 1}\|_2 \geq D > 0, \quad (2.7)$$

D can be chosen arbitrarily, here we use $D = 9n$.

2. Physical principles dictate that, for $\forall i \in \{1, 2, \dots, n\}$,

$$\gamma(\mathbf{b}_i) = \hat{n}(\mathbf{b}_i) \cdot \xi(\mathbf{b}_i) \geq 0. \quad (2.8)$$

Therefore, we have $\|\xi\|_2 \geq D$ and $[\mathbf{CB}(\mathbf{b})]\boldsymbol{\beta} \geq 0$, where \mathbf{C} is the $3n \times 9n$ constraint matrix defined by the $\hat{n}(\mathbf{b}_i)$ in the second constraint. The constraints clearly rule out the zero solution but do not fix the scale of ξ . For this, we rescale the posterior of ξ as described in the next section.

2.3.3 Posterior computation and model fitting

We now describe the computational algorithm for our model. We sample the parameters using a combination of Gibbs sampling and Hamiltonian Monte Carlo sampling. The full conditional posterior distribution of $\boldsymbol{\beta}$ is truncated multivariate normal with mean $(\mathbf{B}^T \boldsymbol{\Omega}^{-1} \mathbf{B} + \mathbf{I}/\lambda^2)^{-1} \mathbf{B}^T \boldsymbol{\Omega}^{-1} \boldsymbol{\xi}$ and covariance $(\mathbf{B}^T \boldsymbol{\Omega}^{-1} \mathbf{B} + \mathbf{I}/\lambda^2)^{-1}$, constrained to $\|\boldsymbol{\xi}\|_2 \geq D$, $\mathbf{C} \mathbf{B} \boldsymbol{\beta} \geq 0$. We sample $\boldsymbol{\beta}$ using the Exact Hamiltonian Monte Carlo (Pakman and Paninski 2014). Hamiltonian Monte Carlo (HMC) was first introduced by Duane et al. (1987), where they united the MCMC and the molecular dynamics approach Hamiltonian dynamics (Alder and Wainwright 1959) to address lattice field theory simulations. Not long after, HMC began to be applied to statistical problems. Examples are Neal (1996b); Ishwaran (1999); Schmidt (2009). There have also been some tutorial and reviews on HMC such as Neal (1993); Liu (2008); Neal (2012). Pakman and Paninski (2014) presented Exact HMC algorithm to sample from multivariate normal distribution in which the target space is constrained by linear and quadratic inequalities.

We sample other parameters using Gibbs sampling. The complete MCMC sampling scheme follows the given process:

- $\boldsymbol{\beta} | \boldsymbol{\xi}, \lambda, \boldsymbol{\Omega} \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, where $\boldsymbol{\mu}_1 = (\mathbf{B}^T \boldsymbol{\Omega}^{-1} \mathbf{B} + \mathbf{I}/\lambda^2)^{-1} \mathbf{B}^T \boldsymbol{\Omega}^{-1} \boldsymbol{\xi}$ and $\boldsymbol{\Sigma}_1 = (\mathbf{B}^T \boldsymbol{\Omega}^{-1} \mathbf{B} + \mathbf{I}/\lambda^2)^{-1}$
- $\boldsymbol{\xi} | \mathbf{Y}, \sigma^2, \boldsymbol{\Omega}, \boldsymbol{\beta} \sim N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, where $\boldsymbol{\mu}_2 = (\mathbf{A}^T \mathbf{A} / \sigma^2 + \boldsymbol{\Omega}^{-1})^{-1} (\mathbf{A}^T \mathbf{Y} / \sigma^2 + \boldsymbol{\Omega}^{-1} \mathbf{B} \boldsymbol{\beta})$ and $\boldsymbol{\Sigma}_2 = (\mathbf{A}^T \mathbf{A} / \sigma^2 + \boldsymbol{\Omega}^{-1})^{-1}$
- $\sigma^2 | \mathbf{Y}, \boldsymbol{\xi} \sim \text{InvGamma}(a + \frac{3n}{2}, b + \frac{(\mathbf{Y} - \mathbf{A} \boldsymbol{\xi})^T (\mathbf{Y} - \mathbf{A} \boldsymbol{\xi})}{2})$
- $\boldsymbol{\Sigma} | \boldsymbol{\xi}, \boldsymbol{\beta} \sim \text{InvWishart}(\phi + \sum_{i=1}^{3n} \mathbf{a}_i, df + 3n)$, where $\mathbf{a}_{i_{3 \times 3}}$ is diagonal block matrix of $(\boldsymbol{\xi} - \mathbf{B} \boldsymbol{\beta})(\boldsymbol{\xi} - \mathbf{B} \boldsymbol{\beta})^T$

As described in section 3.2, the scale of $\boldsymbol{\xi}$ is not identifiable. Therefore for each MCMC iteration, we compute $\tilde{\boldsymbol{\xi}}(\mathbf{b}) = \mathbf{B}(\mathbf{b}) \boldsymbol{\beta}$ and rescale as $\boldsymbol{\xi}(\mathbf{b}) = \tilde{\boldsymbol{\xi}}(\mathbf{b}) / \|\tilde{\boldsymbol{\xi}}(\mathbf{b})\|_2$. The capillary vector

estimates are then the sample mean of the draws of $\xi(\mathbf{b})$ and the pointwise 95% prediction intervals are given by the sample quantiles of the MCMC draws of $\xi(\mathbf{b})$.

2.4 Simulation Study

We use one-dimensional and two-dimensional examples to illustrate our method in Sections 5.1 and 5.2, respectively. In Section 5.3, we conduct a five-dimensional simulation study to mimic the real data. For each simulation design we generate 100 datasets and compare our constrained Bayesian nonparametric regression (CBNR) model with block aggregation (BA). Methods are compared with respect to prediction mean squared errors (MSE) and 95% credible set coverage (not available for BA). In the BA method, the parameter space is divided into discrete bins (so that each bin has 3 - 5 data points) and each bin is associated with one unknown capillarity vector ξ . For every grain boundary in the experimental dataset, its capillarity vector is calculated by averaging all the capillarity vector of the bins that contain \mathbf{b} 's equivalences. To compare with the ground truth, we rescale the reconstructed responses so that the maximum is the same as the true maximum response. Prediction MSE is calculated using formula $MSE = \frac{1}{n} \sum_{i=1}^n (\hat{\xi}(\mathbf{b}) - \xi)^2$. Simulation results are shown in Table 1, both MSE and 95% credible set coverage are the mean over the 100 simulated datasets.

2.4.1 One-dimensional simulation

First, we demonstrate our method using a one-dimensional example. Assume the true capillary function is

$$\xi(b) = \begin{cases} \max[\sin(b), 1/b], & \text{if } b \in [0, 2\pi] \\ -5 \sin(b/2), & \text{if } b \in [2\pi, 4\pi] \\ (b - 4\pi)^2/10 & \text{otherwise.} \end{cases}$$

The constraint here is $\sum_{i=1}^n \xi_i^2 \geq n$ and $\xi_i \geq 0$ for $i = 1, 2, \dots, n = 100$. We assume we only know linear combinations of the corresponding responses are zeros. That is, we generate coefficient matrix \mathbf{A} , such that $\mathbf{A}\boldsymbol{\xi} = \boldsymbol{\delta}$, where $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_n)$, $\mathbf{b} \in [0, 20]$, $\boldsymbol{\delta} \sim N(0, 0.1)$ is random noise. To make $\mathbf{A}\boldsymbol{\xi} = \boldsymbol{\delta}$, we first generate random matrix \mathbf{A} with standard normal elements. Denote the last column of \mathbf{A} as \mathbf{a} and the last element of $\boldsymbol{\xi}$ as ϕ , we then replace the final column of \mathbf{A} with $(\mathbf{a} - \mathbf{A}\boldsymbol{\xi} - \boldsymbol{\delta})/\phi$. We denote this simulation as Case 1. Second, we generate 500 data points from one-dimensional mean-zero Gaussian process (GP) with Matern correlation function $cov(\xi_i, \xi_j) = \sigma^2(2^{\kappa-1}\Gamma(\kappa))^{-1}(\|b_i - b_j\|/\phi)^\kappa K_\kappa(\|b_i - b_j\|/\phi)$, where $\Gamma(\cdot)$ is the gamma function, K_κ is the modified Bessel function of the third kind of order κ , ϕ is the range parameter and κ is the smoothness parameter. We then generate coefficient matrix \mathbf{A} as in Case 1. For the parameters used in generating Gaussian process data points, we set $\sigma^2 = 0.9$, $\phi = 0.1$ and $\kappa = 2$, and we do not include nugget. We denote this simulation as Case 2.

Figures 2.2 and 3.3 show the results of one simulated dataset for Case 1 and Case 2, respectively. Our CBNR model recovers the true curve smoothly because of the spline basis functions; the BA is not smooth, each bin shares the same estimated value. Note that CBNR can make predictions at new locations. However, since BA method uses numerical

optimization to solve the linear equation system, it can only provide estimates at existing locations. In addition, according to Table 1, CBNR outperforms BA with MSE 3.6 times less for Case 1 and 6 times less for Case 2, which means CBNR gives more accurate predictions than BA. Furthermore, another advantage of CBNR over BA is that CBNR can provide prediction intervals. The 95% CI coverage for Case 1 and 2 are 92.35% and 92.51%.

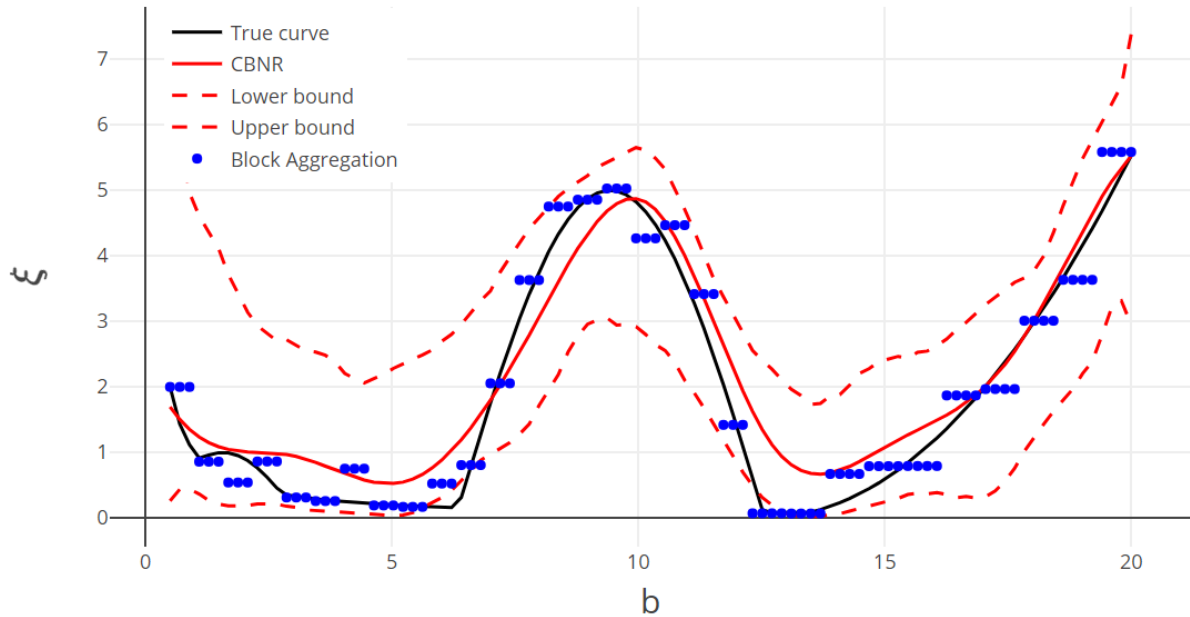


Figure 2.2: Analysis of one simulated dataset for Case 1. Plotted is the true curve ξ (black), the Constrained Bayesian Nonparametric Regression (CBNR) Method (red), 95% credible set (dashed red) and the Block Aggregation method (blue).

2.4.2 Two-dimensional simulation

We then conduct simulations for two dimensional case. Assume the true curves are

$$\begin{cases} c_1 = 75 \cos \theta + 354 \sin \theta \cos \phi + 206 \sin \theta \sin \phi \\ c_2 = 75 \sin \theta + 354 \cos \theta \sin \phi + 206 \cos \theta \cos \phi \end{cases},$$

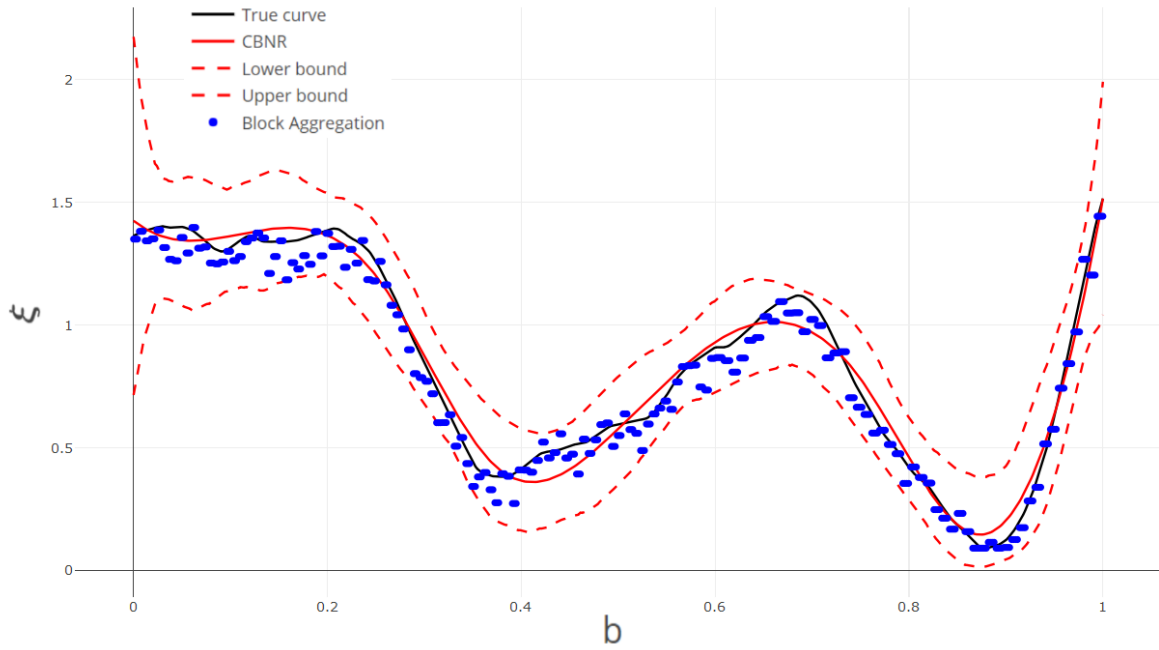


Figure 2.3: Analysis of one simulated dataset for Case 2. Plotted is the true curve ξ (black), the Constrained Bayesian Nonparametric Method (red), 95% credible set (dashed red) and the Block Aggregation method (blue).

and the constraints are $\sum_{i=1}^{294} (c_{1i}^2 + c_{2i}^2) \geq 294$, and $\theta_i c_{1i} + \phi_i c_{2i} \geq 0$, $i = 1, 2, \dots, 294$. Here, $\xi = (c_1, c_2)$ and $\mathbf{b} = (\theta, \phi)$, $\theta \in [0, \pi/2]$, $\phi \in [0, \pi/6]$. As in the one-dimensional case, we assume we only know linear combinations of the corresponding responses are zeros. Here, we generate 100 different coefficient matrix \mathbf{A} , such that $\mathbf{A}\mathbf{c} = \mathbf{0}$, where $\mathbf{c} = (c_{11}, \dots, c_{1n}, c_{21}, \dots, c_{2n})$, $n = 294$. We denote this simulation as Case 3. We project results on the sphere onto two-dimensional plane, Figure 3.4 shows the true values of c_1 and c_2 , as well as fitted values on the 2-d plane, the CBNR model recovers the two-dimensional surfaces c_1 and c_2 precisely. Similar to one-dimensional case, our method has more accurate predictions than BA, with MSE being 86% smaller than BA. In this case, the 95% CI coverage is 99.65%.

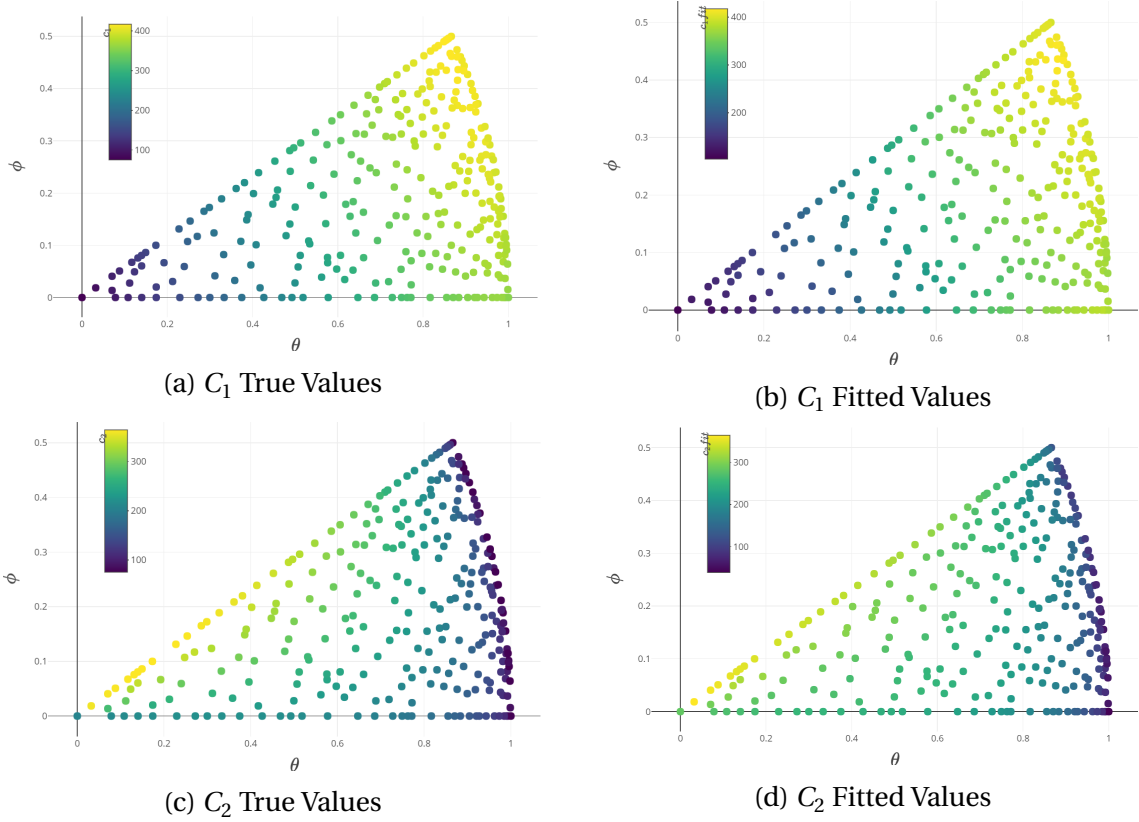


Figure 2.4: Analysis of one simulated dataset for Case 3. Panels (a) and (c) show the true values and panels (b) and (d) show the recovered values by Constrained Bayesian Nonparametric Regression method.

2.4.3 Five-dimensional simulation

Finally, we conduct a systematic simulation study based on real data. Since we suspect the capillarity vectors $\xi(\cdot) = [\xi_1(\cdot), \xi_2(\cdot), \xi_3(\cdot)]^T$ in the application come from a smooth multivariate curve, to evaluate the performance of our method, we assume

$$\xi(\mathbf{b}) \sim GP(\boldsymbol{\mu}, \boldsymbol{\Omega}),$$

where $\boldsymbol{\mu}$ is the mean of Gaussian process (GP), and $\boldsymbol{\Omega}$ is the covariance matrix with exponential correlation function $cov(\xi_i, \xi_j) = \sigma^2 \exp(-\|\mathbf{b}_i - \mathbf{b}_j\|/\rho)$, where σ^2 is the variance, ρ

Table 2.1: Mean squared error and 95% confidence interval coverage comparison between Constrained Bayesian Nonparametric Regression (CBNR) and Block Aggregation (BA) for simulated data. Case 1 and 2 refer to the one-dimensional simulations, Case 3 refers to the two-dimensional simulation and Case 4 refers to the five-dimensional simulation.

	CBNR		BA
	MSE	95% CI Coverage	MSE
Case 1	0.19	92.35%	0.87
Case 2	0.01	92.51%	0.08
Case 3	19.50	99.65%	136.42
Case 4	0.32	86.67%	1.59

is the range parameter controlling spatial dependence and $\|\cdot\|$ is the Euclidean distance. Without loss of generality, we let $\boldsymbol{\mu} = \mathbf{0}$. Therefore, using the 5×1 boundary-plane crystallography vector \mathbf{b} in the real dataset, we generate capillarity vectors $\boldsymbol{\xi}$ from a GP with variance $\sigma^2 = 5$ and range parameter $\rho = 1$ and we do not add a nugget term. and coefficient matrix \mathbf{A} , such that $\mathbf{A}\boldsymbol{\xi} = \mathbf{0}$. We subsample 100 different sets of \mathbf{b} in the real data. We denote this simulation as Case 4. As shown in Table 1, CBNR has prediction MSE 0.32, while MSE for BA is 1.59. In this case, the 95% CI coverage drops down to 86.67%.

In conclusion, the simulation study shows three advantages of CBNR to BA. First, CBNR can make predictions at new data points, while BA can only recover at the existing points. Second, CBNR make much more accurate predictions than BA. Third, CBNR can provide prediction intervals, which BA can not offer.

2.5 Real Data

In the dataset, there are $n = 19,094$ triple junctions. By imposing (7) in Section 3.2, all grain boundary energies can be determined up to a constant factor. Since we do not know the ground truth about grain boundary energy, metrics such as MSE can not be used to compare

Table 2.2: Relative prediction MSE of CBNR to BA

No. of basis functions	5	10	20	30
Relative MSE	0.22	0.20	0.14	0.13

the two methods. However, according to Equation 5, the better method should have predictions with $\mathbf{A}\xi$ closer to zero. Another factor we explore is the number of basis functions. We set number of basis functions to be $m = 5, 10, 20, 30$, and calculate $\frac{1}{3n} \sum_{i=1}^{3n} (\mathbf{A}_i \hat{\xi}(\mathbf{b}) - 0)^2$ for each m , where \mathbf{A}_i is the i^{th} row of coefficient matrix \mathbf{A} . We use five-fold cross validation to avoid overfitting. Table 2 shows the relative prediction MSE of CBNR to BA. As the number of basis functions increases, the ratio decreases, meaning that more basis functions lead to better results. The ratio is always much smaller than one, meaning that our CBNR model outperforms BA with respect to prediction accuracy.

Visualizing the fitted surface of grain boundary energy in five-dimensional space is challenging. One approach is to fix the three misorientation parameters and plot GB energy in the remaining two dimensions. One specific misorientation we are interested in is called $\Sigma 3$, with misorientation parameters (0.52, 0.96, 0.79). The two boundary-plane orientation parameters can be gridded on a 3D sphere, we predict GB energies and project them on the sphere. Figure 3.5 shows the results. The GB energy is strongly peaked at the position of the so-called coherent twin, which corresponds to boundary-plane parameters $(\arccos 1, \pi/4)$. This is due to the fact that the boundary-plane population is maximized at the coherent twin. Another advantage of CBNR against the conventional method BA is that we can provide prediction intervals. For instance, as shown in Figure 2.6, we in further fix one of the two boundary-plane parameters and get one dimensional predictions with 95% prediction interval. the two rock bottom points at two sides correspond to the blue area the opposite side in Figure 3.5. The prediction intervals are narrower than others because boundary-plane population is maximized, in other words, we have more samples around

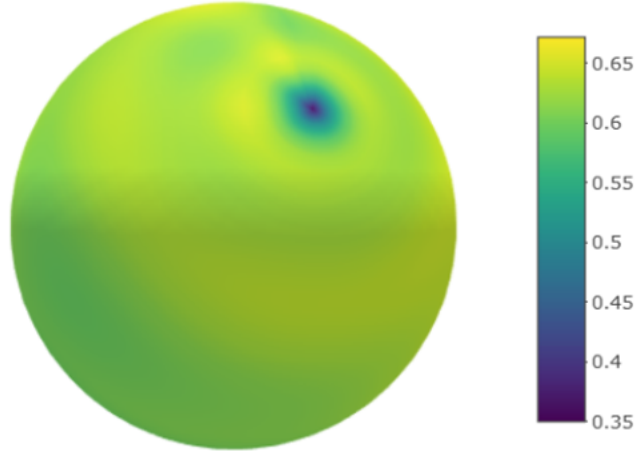


Figure 2.5: Stereographic projections of the grain boundary energy distribution for the $\Sigma 3$ misorientation. Posterior mean of the unit-less capillary vectors ξ is 0.61.

these two areas than other areas.

Our Constrained Bayesian Nonparametric Regression (CBNR) model can recover and predict grain boundary energy surface and also give prediction intervals. The computation scheme is based on the simplified random error model

$$\mathbf{A}\xi + \delta = 0,$$

where $\delta \sim N(0, \sigma^2 \mathbf{I})$. Recall that we assume the components orientation matrix \mathbf{O}_{ij} , boundary-plane vector \mathbf{b}_{ij} and tangent vector \mathbf{l}_i are measured with error. However, since it is challenging to estimate the measurement error associated with each component, we simplify the process and propose the above model. That said, the justification of this simplified assumption on the measurement error remains unverified.

Now we conduct a simulation study to justify this model assumption. From the scientific fact we know that the measurement error associated with measuring the tangent vector \mathbf{l}_i is larger than those with measuring \mathbf{O}_{ij} and \mathbf{b}_{ij} . It is reasonable to assume that measurement

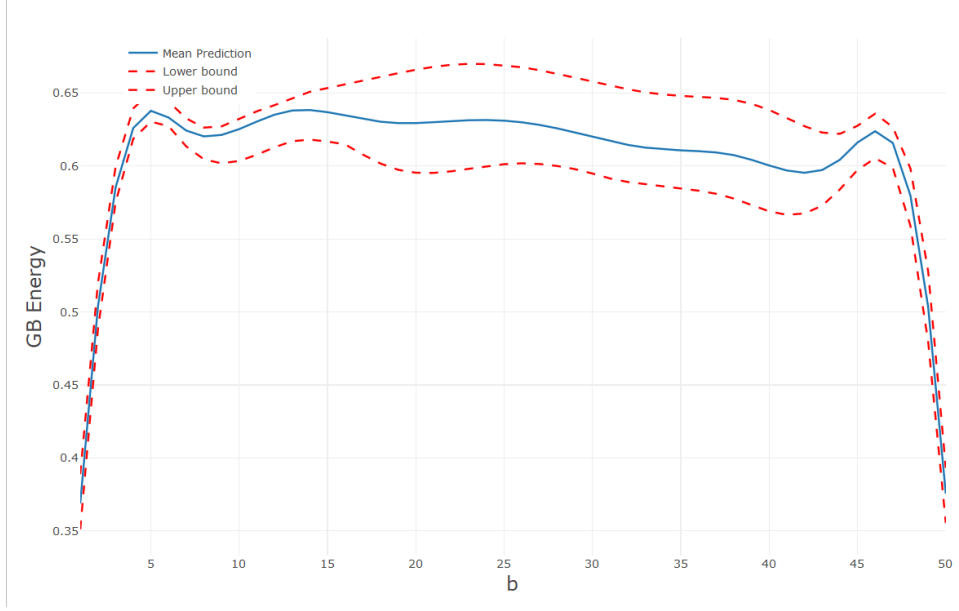


Figure 2.6: One dimensional prediction of the grain boundary energy distribution with 95% prediction interval for the $\Sigma 3$ misorientation fixing one boundary-plane parameters

error with \mathbf{l}_i has more impact than those with \mathbf{O}_{ij} and \mathbf{b}_{ij} and we thus further assume that measurement errors with \mathbf{O}_{ij} and \mathbf{b}_{ij} are negligible. Therefore, we fix \mathbf{O}_{ij} , \mathbf{b}_{ij} as in the real data and $\hat{\xi}$ as estimated value. We then randomly generate error $\epsilon_{ij} \sim N(0, \gamma^2)$, $j = 1, 2, 3$ as the measurement error associated with \mathbf{l}_{ij} , $j = 1, 2, 3$. In this way, we obtain the tangent vector with measurement error, and the element in the tangent vector is $\tilde{l}_{ij} = \frac{l_{ij} + \epsilon_{ij}}{\sqrt{\sum_{j=1}^3 (l_{ij} + \epsilon_{ij})^2}}$. Denote the new coefficient matrix for the i^{th} triple function as $\tilde{\mathbf{A}}_i$, we then calculate

$$\delta_i = (\tilde{\mathbf{A}}_i - \mathbf{A}_i) [\hat{\xi}(\mathbf{b}_{i1})^T, \hat{\xi}(\mathbf{b}_{i2})^T, \hat{\xi}(\mathbf{b}_{i3})^T]^T.$$

We are interested in the empirical distributions of δ_i with different amount of error variance γ^2 . In addition, we would also like to see if the correlations between δ_{ijs} , $j = 1, 2, 3$ are strong. Since the tangent vector \mathbf{l}_i has norm one, we set γ^2 to be 0.01, 0.1, 0.05, 0.1, 0.5 and for each value of γ^2 , we generate 100 replications and the results are shown in Table

Table 2.3: Correlations between δ_{ij} 's and variances of δ_{ij} 's, where r_{jk} is the correlation estimate between δ_{ij} and δ_{ik} , σ_j^2 is the variance estimate of δ_{ij} . All the numbers in the table are multiplied by 10^{-3} .

γ^2	r_{12}	r_{13}	r_{23}	σ_1^2	σ_2^2	σ_3^2
0.01	0.5(1)	-0.2(1)	-1(1)	0.19(4.3×10^{-2})	0.20(4.3×10^{-2})	0.27(5.5×10^{-2})
0.05	-1(1)	2(1)	0.4(1)	0.97(0.08)	1.02(0.08)	1.34(0.09)
0.1	4(1)	5(2)	7(2)	1.97(0.02)	2.09(0.02)	2.15(0.02)
0.5	-7(2)	20(2)	9(2)	8.21(0.5)	8.43(0.7)	8.08(0.6)

1 and Figure 1. From Table 1, even though the correlations increase as the variance γ^2 increases, the absolute values of these correlations are very small and close to zero. The variances of δ_{ij} s are similar and small, indicating that the constant variance assumption seems to be appropriate. What is more, according to Figure 1, the histograms of δ_i s under different values of γ^2 show that each of δ_{ij} is normal-like, with relatively small variances. The correlations between δ_i s and the histograms of δ_{ij} s are not sensitive to error variance γ^2 . The results from Table 1 and Figure 1 indicate that our assumption about the simplified random error makes sense.

We further investigate QQ plots for δ_{ij} s. Figure 2 shows the results for $\gamma^2 = 0.5$ as an example. Compared with normal assumption for δ , Laplace distribution seems to be a more appropriate assumption, even though it is less commonly used in the literatures as an error distribution than normal distribution. If we assume Laplace random error, then since δ_{ij} s are uncorrelated and constant variance holds, the posterior distribution for variance σ^2 is still inverse Gamma. However, posterior distribution for ξ is no longer conjugate, we need to implement Metropolis sampling instead of Gibbs sampling to sample ξ . Other MCMC sampling scheme remains the same. Especially for β , we can still use exact Hamiltonian Monte Carlo to sample from the truncated normal distribution. Therefore, the key posterior computation remains to be the same.

2.6 Discussion

In this paper, we apply nonparametric Bayesian regression to grain boundary energy prediction. We formulate a new model that can recover and predict grain boundary energy relying on Herring's equation. Our method outperforms conventional numerical method with respect to prediction accuracy for both simulated and real data, and can provide prediction intervals.

In material science it is often desirable to take symmetry operations in the five-dimensional parameter space into account, as stated in Section 1. Therefore, a capability of describing arbitrary functions in the five-parameter crystallographic phase-space of GBs will pave the way for applying statistical regression-based techniques and machine learning algorithms for the analysis of interface crystallography-structure and crystallography-property relationships. The numerical results are expected to be improved if we can extend the current spherical harmonics basis functions in the rotation space $SO(3)$ to the five-dimensional space.

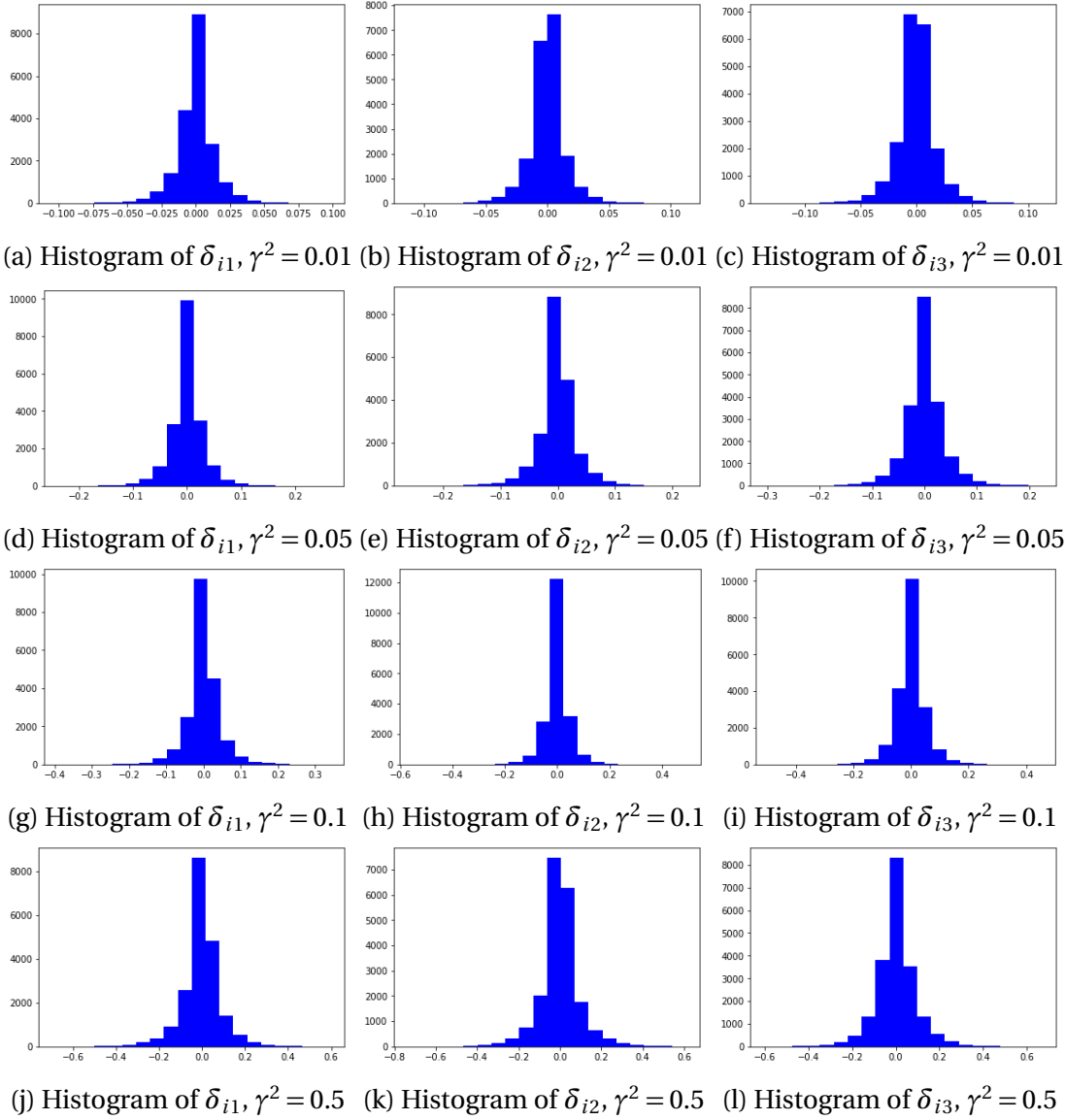
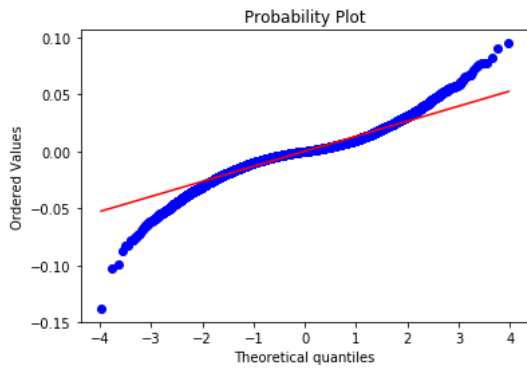
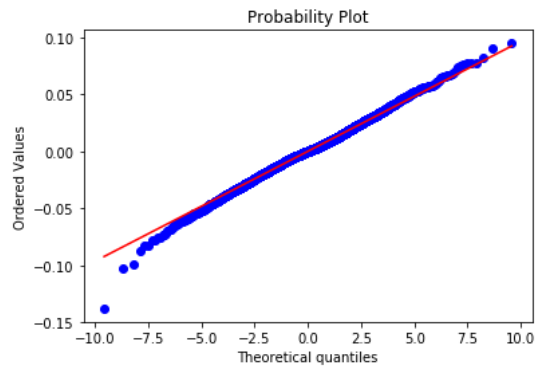


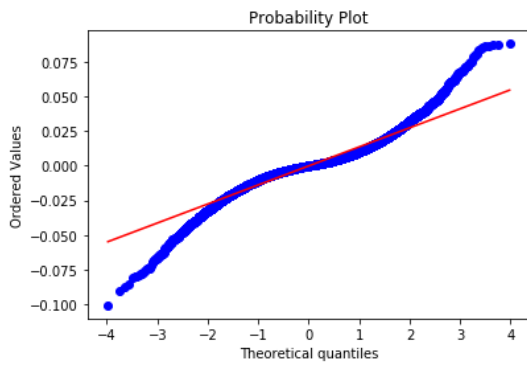
Figure 2.7: Histograms of δ_i for different values of γ^2



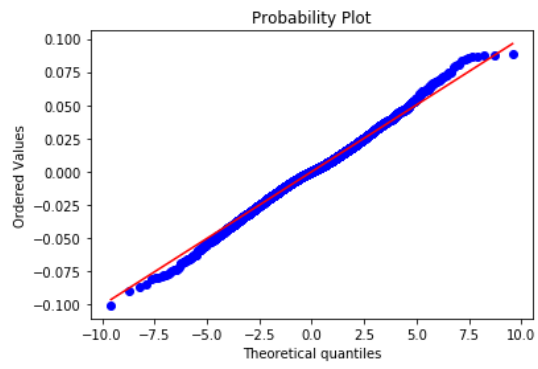
(a) Normal QQ Plot for δ_{i1}



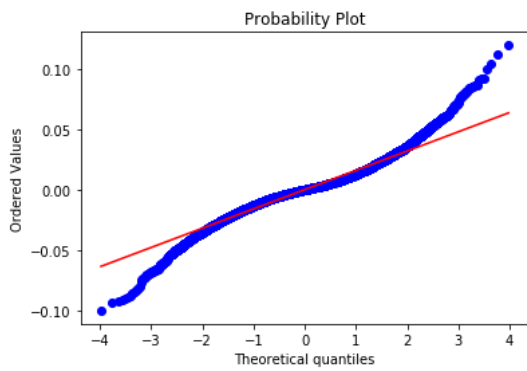
(b) Laplace QQ Plot for δ_{i1}



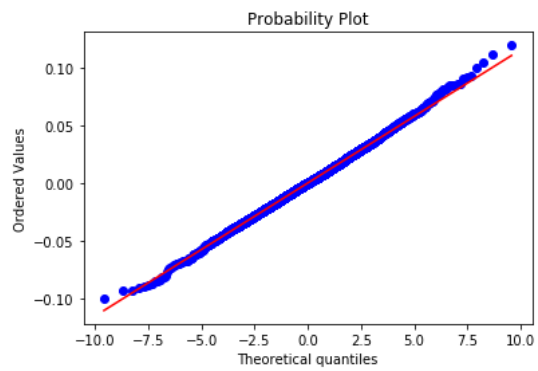
(c) Normal QQ Plot for δ_{i2}



(d) Laplace QQ Plot for δ_{i2}



(e) Normal QQ Plot for δ_{i3}



(f) Laplace QQ Plot for δ_{i3}

Figure 2.8: QQ plots for $\delta_i, \gamma^2 = 0.5$

CHAPTER

3

NEAREST-NEIGHBOR NEURAL NETWORKS FOR GEOSTATISTICS

3.1 Introduction

The Gaussian process (GP) (Rasmussen 2004) is the foundational stochastic process used in geostatistics (Cressie 1992; Stein 2012; Gelfand et al. 2010; Gelfand and Schliep 2016). GPs are used directly to model Gaussian data and as the basis of non-Gaussian models such as generalized linear (e.g., Diggle et al. 1998), quantile regression (e.g., Lum et al. 2012; Reich 2012) and spatial extremes (e.g., Cooley et al. 2007; Sang and Gelfand 2010)

models. Similarly, Kriging is the standard method for geostatistical prediction. Kriging can be motivated as the prediction that arises from the conditional distribution of a GP or more generally as the best linear unbiased predictor under squared error loss (Ferguson 2014; Whittle 1954; Ripley 2005). Both GP modeling and Kriging require estimating the spatial covariance function, which often relies on assumptions such as stationarity. Parametric assumptions such as linearity, normality and stationary can be questionable and difficult to verify.

Flexible methods have been developed to overcome these limitations (Reich and Fuentes 2015, provide a review). There is an extensive literature on nonstationary covariance modelling (Risser 2016, provides a recent review). Another class of methods estimates the covariance function nonparametrically but retains the normality assumption (e.g., Huang et al. 2011; Im et al. 2007; Choi et al. 2013). Fully nonparametric methods that go beyond normality have also been proposed (e.g., Gelfand et al. 2005; Duan et al. 2007; Reich and Fuentes 2007; Rodriguez and Dunson 2011), but are computationally intensive and often require replication of the spatial process.

Deep learning has emerged as a powerful alternative framework for prediction. The ability of deep learning to reveal intricate structures in high-dimensional data such as images, text and videos make them extremely successful for complex tasks (Schmidhuber 2015; Goodfellow et al. 2016; LeCun et al. 2015, provide detailed reviews). For similar reasons, researchers have begun to explore the application of deep learning to model complex spatial-temporal data. Xingjian et al. (2015) propose the convolutional long short term memory (ConvLSTM) structure to capture spatial and temporal dependencies in precipitation data. Zhang et al. (2017) design an end-to-end structure of ST-ResNet based on original deep residual learning (He et al. 2016) for citywide crowd flows prediction. Fouladgar et al. (2017) propose a convolutional and recurrent neural network based method to accurately predict traffic congestion state in real-time based on the congestion state of the neigh-

boring information. Rodrigues and Pereira (2018) propose a multi-output multi-quantile deep learning approach for jointly modeling several conditional quantiles together with the conditional expectation for spatial-temporal problems. One common structure these previous work apply is convolutional neural network, which requires input data presented in grid cell form, i.e., an image. However, less attention in this field has been drawn to point-referenced geostatistical data at irregular locations (non-gridded). Di et al. (2016) establish a hybrid geostatistical model that incorporates multiple covariates (including a convolutional layer to capture neighboring information) to improve model performance. Unlike the proposed method, neighboring observations enter the predictive model of Di et al. (2016) only through a linear combination (similar to the 4N-Kriging model described below in Section 2.3).

In this paper, we establish a more flexible spatial prediction method by embedding deep learning into a geostatistical model. The method uses deep learning to build a predictive model based on neighboring values and their spatial configuration. We show that our Nearest-Neighbor Neural Network (4N) is a valid stochastic process model that does not rely on assumptions of stationarity, linearity or normality. Different sets of features are created as inputs to the 4N model including (but are not restricted to) Kriging predictions, neighboring information and spatial locations. By including different features, 4N spans methods that are anchored on parametric models to methods that are completely nonparametric and make no assumptions about the relationship between nearby observations. By construction, the flexible method can be implemented using standard and highly-optimized deep learning software and can thus be applied to massive datasets.

The remainder of the paper is organized as follows. Section 2 presents the 4N model. Section 3 compares the proposed model with the Nearest-Neighbor Gaussian Process model (Datta et al. 2016) and the spatial Asymmetric Laplace Process model (Lum et al. 2012) via a simulation study. In Section 4, the algorithm is applied to Canopy Height Model

(CHM) data. Section 5 summarizes the results and discusses a potential method to obtain prediction uncertainties.

3.2 Methodology

3.2.1 Nearest-Neighbor Gaussian Process

In this section we review the Nearest-Neighbor Gaussian Process (NNGP) of Datta et al. (2016). Let Y_i be the response at spatial location s_i , and denote $Y_{\mathcal{N}} = \{Y_i; i \in \mathcal{N}\}$ for any set of indices \mathcal{N} . The NNGP is defined in domain \mathcal{D} and is specified in terms of a reference set $\mathcal{S} = \{\infty, \dots, \backslash\}$, which we take to be observation locations, and locations outside of \mathcal{S} , $\mathcal{U} = \{\backslash + \infty, \dots, \backslash + \|\}$. The joint density of $Y_{\mathcal{S}}$ can be written as the product of conditional densities,

$$p(Y_{\mathcal{S}}) = \prod_{i=1}^n p(Y_i | Y_1, \dots, Y_{i-1}). \quad (3.1)$$

When n is large this joint distribution is unwieldy and Datta et al. (2016) use the conditioning set approximation (Vecchia 1988; Stein et al. 2004; Gramacy and Apley 2015; Gramacy et al. 2014; Datta et al. 2016)

$$p(Y_{\mathcal{S}}) \approx \tilde{p}(Y_{\mathcal{S}}) = \prod_{i=1}^n p(Y_i | Y_{\mathcal{N}_i}) \quad (3.2)$$

where $\mathcal{N}_i \subset \{\infty, \dots, \backslash - \infty\}$ is the conditioning set (also referred to as the neighboring set) for observation i of size $m \ll n$. Datta et al. (2016) and Lauritzen (1996) prove that $\tilde{p}(Y_{\mathcal{S}})$ constructed in this way is a proper joint density.

We follow Vecchia (1988)'s choice of neighboring set and take \mathcal{N}_i to be the indices of the m nearest neighbors of s_i in the set $\{s_1, \dots, s_{i-1}\}$. The motivation for this approximation is that after conditioning on the m nearest (and thus most strongly correlated) observations, the remaining observations do not contribute substantially to the conditional distribution

and thus the approximation is accurate. This approximation is fast because it only requires dealing with matrices with size of $m \times m$.

Datta et al. (2016) further define the conditional density of $Y_{\mathcal{Q}}$ as

$$p(\mathbf{Y}_{\mathcal{Q}}|\mathbf{Y}_{\mathcal{S}}) = \prod_{i=n+1}^{n+k} p(Y_i|Y_{\mathcal{N}_i}), \quad (3.3)$$

where \mathcal{N}_i for $i > n$ is comprised of the m nearest neighbors to s_i in \mathcal{S} . This is a proper conditional density. Equations (2) and (3) are sufficient to describe the joint density of any finite set over domain \mathcal{D} . Datta et al. (2016) show that these densities satisfy the Kolmogorov consistency criteria and thus correspond to a valid stochastic process.

3.2.2 Nearest-Neighbor Neural Network (4N) model

While the sparsity of the NNGP delivers massive computational benefits, it remains a parametric Gaussian model. We note that the proof in Datta et al. (2016) that the NNGP model for $(Y_{\mathcal{S}}, Y_{\mathcal{Q}})$ yields a valid stochastic process holds for any conditional distribution $p(Y_i|Y_{\mathcal{N}_i})$, including non-linear and non-Gaussian conditional distributions. Therefore, in this paper we extend the nearest neighbor process to include a neural network in the conditional distribution and thus achieve a more flexible modeling framework.

Let $\mathbf{X}_i = (X_{i1}, \dots, X_{ip})$ be features constructed from the available information s_i , \mathcal{N}_i and $Y_{\mathcal{N}_i}$ (this can include spatial covariates such as the elevation corresponding to s_i). The features are then related to the response via the link function $f(\mathbf{X}_i)$, that is, $Y_i = f(\mathbf{X}_i) + \epsilon_i$, where ϵ_i is additive error. The NNGP model assumes that $f(\cdot)$ is a linear combination of $Y_{\mathcal{N}_i}$ with weights determined by the configuration of s_i and locations in its neighboring set. We generalize this model using a multilayer perceptron for $f(\mathbf{X}_i)$. A multilayer perceptron (MLP) is a class of feedforward neural network. It consists of at least three layers of nodes: an input layer, one or more than one hidden layers, and an output layer. Nodes in different

layers are connected with activation function such as Relu, tanh and sigmoid. The hidden layers and nonlinear activation function enable multilayer perceptron capture nonlinear relationships.

Let $\mathbf{X} = (\mathbf{X}_1^T, \mathbf{X}_2^T, \dots, \mathbf{X}_n^T)^T$ be the $n \times p$ covariate matrix, where $\mathbf{X}_i, i = 1, 2, \dots, n$ is associated with corresponding $Y_{i\mathcal{N}}$ (we will explain how to generate these features in the next section). Our 4N model is then:

$$Y_i = f(\mathbf{X}_i) + \epsilon_i, \quad (3.4)$$

where $f(\cdot)$ is a multilayer perceptron and ϵ_i is iid random error. In matrix form, an L -layer perceptron can be written as follows:

$$\begin{aligned} \mathbf{Z}^1 &= \mathbf{W}^1 \mathbf{X}^T + \mathbf{b}^1, \quad \mathbf{A}^1 = \psi_1(\mathbf{Z}^1) \\ \mathbf{Z}^2 &= \mathbf{W}^2 \mathbf{A}^1 + \mathbf{b}^2, \quad \mathbf{A}^2 = \psi_2(\mathbf{Z}^2) \\ &\dots \\ \mathbf{Z}^L &= \mathbf{W}^L \mathbf{A}^{L-1} + \mathbf{b}^L, \quad f(\mathbf{X}) = \psi_L(\mathbf{Z}^L), \end{aligned}$$

where $\psi_l(\cdot)$ is an activation function for layer l and $f(\mathbf{X}_i)$ is the i^{th} element of $f(\mathbf{X})$. The activation function for the L th layer, also known as the output layer, is chosen according to the problems at hand. For instance, in this paper, we focus on regression with real-valued responses, thus we set ψ_L to be the linear activation function. Alternatively, the sigmoid function can be used for binary classification and the softmax function can be used for multi-class classification. The unknown parameters are weight matrices W^l and bias vectors b^l . The vectors $\mathbf{A}^1, \dots, \mathbf{A}^{L-1}$ constitute the hidden layers.

The network is trained by minimizing $\frac{1}{n} \sum_{i=1}^n l(Y_i - f(\mathbf{X}_i))$, where $l(u)$ is loss function. Different distributions of random error ϵ_i lead to different loss functions. If ϵ_i is assumed to be normal, then the likelihood is normally distributed and maximizing the log-likelihood

is then equivalent to minimizing the squared loss function $l(u) = u^2$, and $f(\cdot)$ models the mean of the response values. If ϵ_i follows an asymmetric Laplace distribution, then maximizing the log-likelihood is then equivalent to minimizing the check loss function $l_\gamma(u) = u(\gamma - 1_{\{u < 0\}})$, where $\gamma \in (0, 1)$ refers to the quantile level, and $f(\cdot)$ models the γ quantile of the responses. In this paper, we consider these two assumptions about the random errors, and compare our 4N models with corresponding competitors. More details are given in Section 3. Note that even when ϵ_i is normally distributed, 4N is different than the NNGP model. 4N uses a multilayer perceptron for the link function $f(\cdot)$ to capture nonlinear dependence whereas NNGP restricts $f(\cdot)$ to be a linear combination of $Y_{\mathcal{N}_i}$. In other words, NNGP assumes a Gaussian joint density but 4N does not assume a parametric joint density.

3.2.3 Feature Engineering

4N models with different properties can be obtained by taking different summaries of the m observations of the neighboring sets as features in \mathbf{X}_i . The complete information in the neighboring set (assuming there are no covariates) are the m observations $Y_{\mathcal{N}_i}$ and their $2m$ (m latitudes and m longitudes) locations s_l for $l \in \mathcal{N}_i$. We assume that the conditioning set is ordered by distance to location s_i , so that the location closest to s_i appears first in \mathcal{N}_i and the location farthest to s_i appears last. This ordering encourages the entries in \mathbf{X}_i to have similar interpretation across observations.

A key feature that we use in X_i is the Kriging prediction. The Kriging prediction is based on the model $Y_i = \mu + W_i + \epsilon_i$, where μ is the mean, W_i is a mean-zero Gaussian process with correlation function $\phi(\cdot)$ and $\epsilon_i \sim N(0, \tau^2)$. The Kriging prediction at a new location s_i is

$$\hat{Y}_i = \hat{\mu} + \mathbf{C}_{\mathbf{s}_i} \mathbf{C}_{\mathcal{N}_i}^{-1} (\mathbf{Y}_{\mathcal{N}_i} - \hat{\mu}),$$

where $\mathbf{C}_{\mathbf{s}_i}$ is the $1 \times m$ covariance matrix (vector) between W_i and $W_{\mathcal{N}_i}$, and $\mathbf{C}_{\mathcal{N}_i}$ is the $m \times m$

covariance matrix of $W_{\mathcal{N}_i}$. Since \hat{Y}_i is a function only of information in the neighboring set, using it as a feature fit in the 4N framework.

Although the user is free to construct any features that are thought to be useful, we consider the following three sets of features:

- **Kriging only:** In this case, $p = 1$ and the feature is the Kriging prediction of Y_i given $Y_{\mathcal{N}_i}$, denoted $X_{i1} = \hat{Y}_i$. We use the exponential correlation function $cov(W_i, W_j) = \sigma^2 \exp(-\|\mathbf{s}_i - \mathbf{s}_j\|/\rho)$, where σ^2 is the variance, ρ is the range parameter controlling spatial dependence and $\|\cdot\|$ is the Euclidean distance. Other correlation functions such as Matern and double exponential functions can also be used. We estimate the spatial covariance parameters ρ, σ^2 and τ^2 using Vecchia's approximation (Vecchia 1988) and Permutation and Grouping Methods (Guinness 2018). We then derive the Kriging prediction using the equation above. This gives a stationary model but with potential non-linearity between the Kriging prediction and expected response. Since Kriging gives the best predictions under normality, when data are generated from a Gaussian process and sample size is relatively small, 4N with Kriging as feature should give predictions close to ordinary Kriging.
- **Nonparametric features:** In this case, $p = 3m + 2$ and the features are $s_l, s_l - s_i$ (m differences in latitudes and m differences in longitudes) for $l \in \mathcal{N}_i$ and $Y_{\mathcal{N}_i}$. Including the differences $s_l - s_i$ allows for dependence as a function of distance, while including s_l allows for nonstationarity, that is, different response surfaces in different locations. The Kriging prediction \hat{Y}_i is a function of these p features and thus the model includes the parametric special case. However, by including all $3m$ variables from the neighboring sites without any predefined structure we allow the neural network model to estimate the prediction equation nonparametrically. When data are not normally distributed, nonparametric features including neighboring information are

anticipated to play a key role in prediction. Even if data are normally distributed, 4N with nonparametric features should approximate Kriging predictions as sample size increases.

- **Kriging + nonparametric (NP) features:** In this case, $p = 3m + 3$ as we add the Kriging prediction \hat{Y}_i to the $3m + 2$ features in the nonparametric model. We anticipate that the combination of the two sets of features above can allow 4N to have their respective advantages. That is, if data are normal, including the Kriging feature should make the method competitive with Kriging, and if the true process is non-Gaussian and the sample size is large then the model should be able to learn complicated non-linear relationships and improve prediction.

3.2.4 Computational Details

We use the R package "GpGp" (<https://cran.r-project.org/web/packages/GpGp/GpGp.pdf>) to get estimates of the spatial parameters and subsequent Kriging prediction \hat{Y}_i (Section 2.3). To implement 4N we use the deep learning package "keras" with GPU acceleration (Nvidia GeForce GTX 980Ti) on an 8-core i7-6700k machine with 16GB of RAM. We use Relu activation function for the hidden layers and linear activation for the output layers. For simulated and real data we set number of epochs to be 50 and impose early stopping and dropout (ranging from 0.1 to 0.5) to avoid overfitting. Early stopping is a technique to stop the training process early if the value of loss function does not decrease too much for several epochs. Dropout randomly makes nodes in the neural network dropped out by setting them to zero, which encourages the network to rely on other features that act as signals. That, in turn, creates more generalizable representations of the data. We tune other hyperparameters using five-fold cross validation. For the simulation study, we tune the parameters with one dataset and use these pre-set tuning parameters for the other

datasets. The parameters we tune are the learning rate, mini-batch size, number of layers and hidden units per layer. Learning rate is a hyperparameter that controls how much we are adjusting the weights of our network with respect to the loss gradient. We try different values for the learning rate in the range of 0.1 to 1e-6. A mini-batch refers to the number of examples used at a time, when computing gradients and parameter updates. Mini-batch size in the range of 16-128 are tried in our simulations and real data. We keep our number of layers to no more than three. As suggested in Liang et al. (2018), a network with one hidden layer is usually large enough to approximate the system. We try a range of 100 to 500 hidden units for the first hidden layer and make number of hidden units gradually decrease for each layer. In general, number of hidden layers and hidden units per layer have less impact for model performance than learning rate, mini-batch size and regularization such as dropout and early stopping. All the parameters are trained using ADAM algorithm (Kingma and Ba 2014).

3.3 Simulation Study

We conduct simulations in a variety of settings to evaluate the performance of the 4N model with the three different sets of features. Here we focus on the 4N performance with respect to mean and quantile prediction by using squared loss function and check loss function, respectively. We give the details of the four simulation cases below. We do not include any covariates to focus on modeling spatial dependence. Figure 1 plots a realization for each simulation case.

- **Gaussian Process:** We generate data from the GP with mean $\mu = 0$ and exponential correlation function with variance $\sigma^2 = 5$, nugget variance $\tau^2 = 1$ and range parameter $\rho = 0.16$.

- **Cubic and exponential transformation:** After generating data from the GP, we take $Y_i^3/100 + \exp(Y_i/5)/10$ as new response value. All the parameter settings for generating Gaussian process data are the same as above.
- **Max stable process:** The max stable process is used for modeling spatial extremes. We generate data from the max stable process (in particular, the Schlather process of Schlather (2002)) using R package "spatialExtremes". The marginal distribution is the Frechet distribution with location parameter 1, scale parameter 2, shape parameter 0.3, and range parameter of the exponential correlation function 0.5.
- **Potts model:** The Potts model (Potts 1952) randomly assigns observations to clusters. Let $g_i \in \{1, \dots, G\}$ be the cluster label for the i^{th} observation. The Potts model can be defined by the distribution of g_i given its neighboring set \mathcal{N}_i as $\text{Prob}(g_i = k | g_l \text{ for } l \in \mathcal{N}_i) \propto \exp(\beta \sum_{l \in \mathcal{N}_i} \mathbf{1}\{g_l = k\})$, which encourages spatial clustering of the labels if $\beta > 0$. We simulate Potts model using Markov chain Monte Carlo algorithm (R package "potts"). In the simulation, we generate $G = 8$ blocks and set $\beta = \log(1 + \sqrt{8})$. Given the cluster labels, the responses are generated as $Y_i | g_i \sim N(g_i^2 + 5g_i, \sqrt{g_i})$.

For each simulation setting data are generated on the unit square with the sample size n set to be either 1,000 or 10,000. For each model setting and sample size, we generate 100 datasets. The mean predicted mean squared error, check loss and the corresponding standard error based on the 100 replications are shown in Tables 1 - 3. We compare the results from the three 4N models with those from NNGP and ALP. We implement Nearest-Neighbor Gaussian Process using R package "spNNGP" (<https://cran.r-project.org/web/packages/spNNGP/spNNGP.pdf>) for both parameter estimates and predictions. For Asymmetric Laplace Process, we use package "baquantreg" (<https://github.com/brsantos/baquantreg/>) to get the parameter estimates and write our own codes to get predictions.

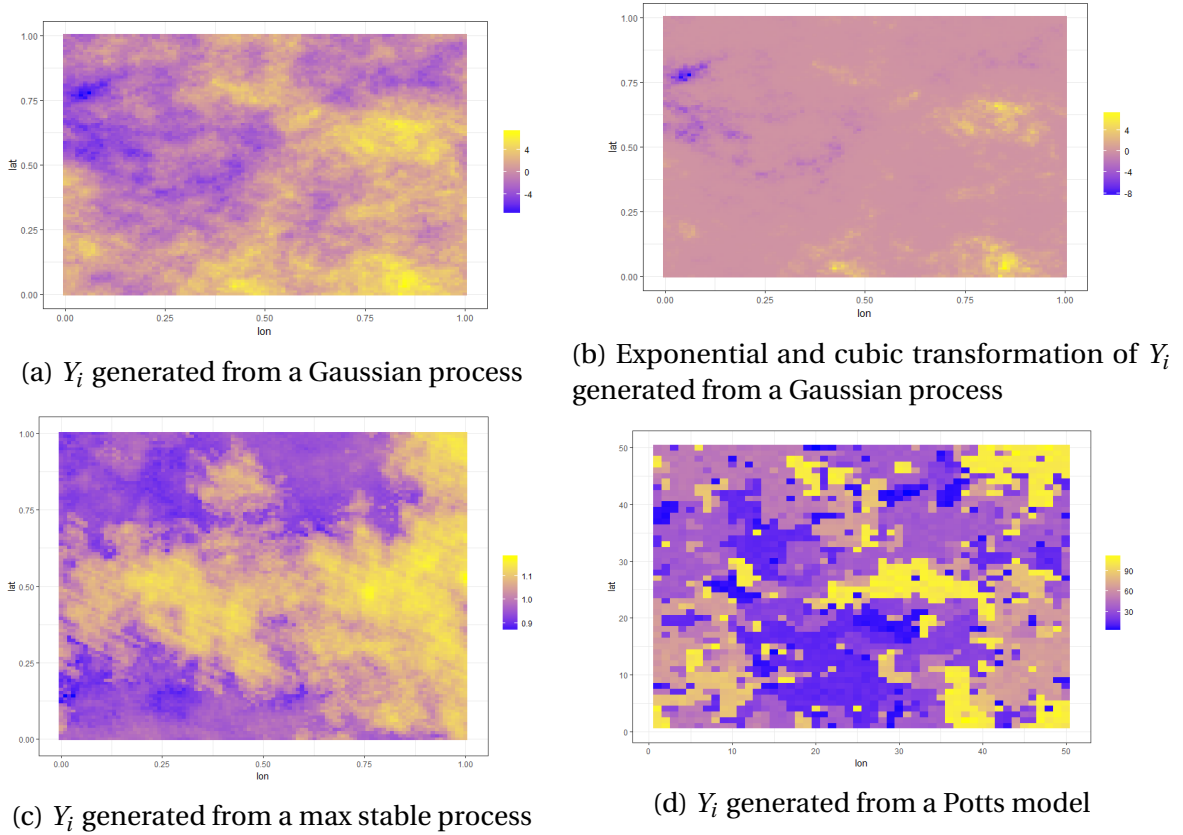


Figure 3.1: Data (\mathbf{Y}) generated from different simulation settings.

3.3.1 Mean Prediction

We first compare our 4N models with NNGP with respect to prediction mean squared error:

$$MSE = \frac{1}{k} \sum_{i=1}^k (Y_i - \hat{Y}_i)^2,$$

where k is number of locations in testing dataset and \hat{Y}_i is the predicted value at location s_i . Table 1 shows the results. For all methods we use $m = 10$. When the data are generated from the GP, NNGP always has the best prediction results. The prediction MSE is at least 4.5% smaller than the 4N model. Among the three 4N models, the Kriging only model gives the best results and the nonparametric features is the worst, as expected. However, when we

take cubic and exponential transformation, most of our 4N models have smaller MSE than NNGP. For both $n = 1,000$ and $n = 10,000$ cases, 4N with only the Kriging prediction as a feature gives the best results, and MSE is 23% and 28% smaller than NNGP. This is expected because the data are generated from a transformation of GP and so the Kriging feature captures most of the spatial dependence. For the max stable process and Potts models our 4N model with Kriging + nonparametric features gives the best prediction results. These processes are unrelated to a GP so the flexibility of including the nonparametric features improves prediction.

Figure 2 further illustrates the role of sample size in the comparison between methods. We generate 50 replications of GP data using same parameters as above, but let $n = 50,000$, 100,000 and 500,000. As sample size increases, the relative MSE of 4N over NNGP converges to one. For other models where 4N outperforms NNGP, such as the max stable process, the ratio is smaller than one and decreases as sample size increases.

We conclude from the simulation results that when data are normally distributed, NNGP gives better predictions than our 4N models, but the 4N models with the Kriging feature remain competitive. In both cases, the relative performance of 4N to NNGP improves as the sample size increases. However, for cases where data are not normally distributed, 4N can outperform NNGP.

3.3.2 Quantile Prediction

We also compare 4N models with the ALP model with respect to check loss:

$$l_\gamma = (\gamma - 1) \sum_{Y_i < \hat{Y}_i} (Y_i - \hat{Y}_i) + \gamma \sum_{Y_i \geq \hat{Y}_i} (Y_i - \hat{Y}_i),$$

where γ is the quantile level of interest. We consider $\gamma \in \{0.25, 0.50, 0.75, 0.95\}$. Tables 2 ($n = 1,000$) and 3 ($n = 10,000$) show the results. If data are generated from a Gaussian

Table 3.1: Mean squared error comparison between 4N models and NNGP (standard error shown in parenthesis) for simulated data.

n	Method	GP	Transformed GP	Max Stable	Potts
1,000	4N(Kriging only)	2.11(0.18)	0.71(0.06)	4.85(0.11)	5.62(0.33)
	4N(Nonparametric)	2.56(0.58)	0.89(0.08)	4.72(0.17)	5.43(0.71)
	4N(Kriging+NP)	2.33(0.29)	0.72(0.07)	4.64(0.18)	5.13(0.34)
	NNGP	1.99(0.14)	0.88(0.05)	4.81(0.19)	5.96(1.45)
10,000	4N(Kriging only)	2.06(0.13)	0.70(0.06)	4.71(0.10)	5.55(0.23)
	4N(Nonparametric)	2.48(0.48)	0.83(0.06)	4.57(0.16)	5.29(0.17)
	4N(Kriging+NP)	2.23(0.23)	0.71(0.08)	4.42(0.20)	5.12(0.19)
	NNGP	1.97(0.13)	0.85(0.09)	4.70(0.15)	5.92(0.18)

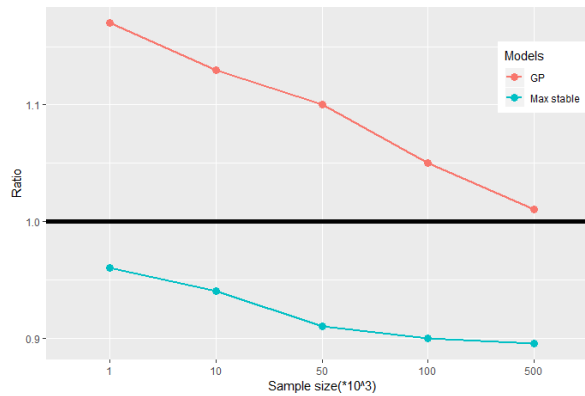


Figure 3.2: Relative prediction MSE of 4N (Kriging + Nonparametric) to NNGP as a function of sample size n , for data generated from GP and max stable models.

process, for $\gamma = 0.5$, ALP model has better prediction results than the 4N model with MSE 3% to 8% smaller than those from 4N models. However, for all other quantile levels, most of the 4N models are superior to the ALP model, with Kriging predictions + nonparametric features giving the best MSE, which is 5% to 12% smaller than the ALP model. Similar conclusions can be drawn for the max stable process except that when $\gamma = 0.95$, 4N with nonparametric features has slightly smaller MSE than 4N with Kriging predictions + nonparametric features (1% difference). For data generated as exponential and cubic transformations of Gaussian data, 4N with Kriging predictions + nonparametric features almost always have the best

Table 3.2: Check loss with different quantiles comparison between 4N models and ALP model, with $n = 1,000$ observations (standard error shown in parenthesis) for simulated data.

Quantile Level	Method	GP	Transformed GP	Max Stable	Potts
0.95	4N(Kriging only)	0.159(0.006)	0.477(0.029)	0.408(0.031)	0.273(0.045)
	4N(Nonparametric)	0.150(0.005)	0.409(0.031)	0.337(0.034)	0.269(0.029)
	4N(Kriging+NP)	0.147(0.006)	0.370(0.027)	0.341(0.029)	0.267(0.023)
	ALP	0.154(0.005)	0.487(0.043)	0.381(0.039)	0.276(0.034)
0.75	QuadN(Kriging only)	0.475(0.015)	1.121(0.054)	1.031(0.041)	0.695(0.025)
	4N(Nonparametric)	0.501(0.026)	1.134(0.067)	1.034(0.034)	0.705(0.013)
	4N(Kriging+NP)	0.450(0.013)	1.083(0.035)	1.013(0.029)	0.680(0.028)
	ALP	0.506(0.014)	1.131(0.047)	1.118(0.032)	0.724(0.024)
0.5	QuadN(Kriging only)	0.570(0.017)	1.713(0.021)	1.237(0.024)	0.770(0.031)
	4N(Nonparametric)	0.593(0.026)	1.819(0.013)	1.264(0.033)	0.705(0.019)
	4N(Kriging+NP)	0.566(0.015)	1.772(0.033)	1.226(0.012)	0.765(0.026)
	ALP	0.550(0.024)	1.798(0.024)	1.213(0.014)	0.835(0.034)
0.25	QuadN(Kriging only)	0.451(0.018)	0.928(0.029)	1.131(0.040)	0.671(0.035)
	4N(Nonparametric)	0.472(0.058)	0.953(0.031)	1.127(0.040)	0.683(0.017)
	4N(Kriging+NP)	0.445(0.029)	0.902(0.027)	1.057(0.047)	0.647(0.022)
	ALP	0.466(0.014)	0.960(0.035)	1.100(0.019)	0.713(0.015)

prediction results, except for $\gamma = 0.5$ where 4N with only Kriging prediction as feature has smaller MSE. Finally, for Potts model 4N models always outperform ALP model.

3.3.3 Variable Importance

In addition to prediction accuracy, we would like to understand the dependence structure estimated by the 4N model for each simulation scenario. To explore the dependence structure we compute the relative importance of each feature (for a review of importance measures, see Gevrey et al. 2003). We use Garson (1991)'s method to partition the connection weight \mathbf{W}^l and bias \mathbf{b}^l to determine the relative importance of the input variables. The method essentially involves partitioning the hidden-output connection weights of each hidden neuron into components associated with each input neuron. By construction, the sum of the relative importance over the p features is one.

For the simulation with $n = 10,000$, we calculate the relative importance for each input variable using Kriging + nonparametric model and nonparametric model averaged over

Table 3.3: Check loss with different quantiles comparison between 4N models and ALP model, with $n = 10,000$ observations (standard error shown in parenthesis) for simulated data.

Quantile Level	Method	GP	Transformed GP	Max Stable	Potts
0.95	4N(Kriging only)	0.154(0.007)	0.472(0.028)	0.409(0.034)	0.271(0.043)
	4N(Nonparametric)	0.147(0.008)	0.407(0.030)	0.335(0.024)	0.264(0.023)
	4N(Kriging+NP)	0.143(0.009)	0.371(0.028)	0.340(0.019)	0.263(0.021)
	ALP	0.152(0.004)	0.480(0.047)	0.383(0.034)	0.277(0.038)
0.75	QuadN(Kriging only)	0.471(0.012)	1.122(0.051)	1.029(0.044)	0.690(0.022)
	4N(Nonparametric)	0.475(0.029)	1.110(0.063)	1.031(0.033)	0.680(0.019)
	4N(Kriging+NP)	0.451(0.015)	1.080(0.031)	1.011(0.022)	0.676(0.029)
	ALP	0.501(0.013)	1.130(0.044)	1.112(0.031)	0.720(0.022)
0.5	QuadN(Kriging only)	0.567(0.017)	1.708(0.022)	1.233(0.022)	0.751(0.030)
	4N(Nonparametric)	0.570(0.023)	1.810(0.018)	1.260(0.032)	0.700(0.011)
	4N(Kriging+NP)	0.562(0.017)	1.773(0.032)	1.222(0.011)	0.723(0.029)
	ALP	0.547(0.021)	1.792(0.023)	1.210(0.012)	0.732(0.032)
0.25	QuadN(Kriging only)	0.444(0.016)	0.950(0.023)	1.132(0.042)	0.672(0.032)
	4N(Nonparametric)	0.450(0.054)	0.923(0.032)	1.123(0.044)	0.650(0.015)
	4N(Kriging+NP)	0.443(0.023)	0.901(0.023)	1.054(0.043)	0.642(0.021)
	ALP	0.463(0.014)	0.961(0.033)	1.004(0.013)	0.710(0.013)

the 100 replications. Figure 3 shows the results. The importance of s_i and Y_i are combined for each neighbor, and the results are plotted so $l = 1$ is the nearest neighbor and $l = 10$ is the most distant neighbor. Figure 3 also includes the importance of the Kriging feature \hat{Y}_i and the prediction location s_i to measure the effect of nonstationarity. For the Kriging + nonparametric model, if the data are generated from a GP or max stable process, then the Kriging feature is the most important. However, for the transformed GP or Potts model importance of Kriging feature decreases and is less than most of the neighboring information. For the other features, importance decreases from the closest neighbor to the farthest neighbor, which is shown more clearly in Figure 3(b) for the nonparametric model. For GP data the closest neighbor is more important than the others, consistent with Kriging prediction. Similar conclusion can be made for data from max stable process. However, for the other scenarios the importance decreases more slowly so that distant neighbors are almost as important as the closer neighbors. Across all the models, the spatial location of the prediction site is the least important feature, which is as expected because the data are

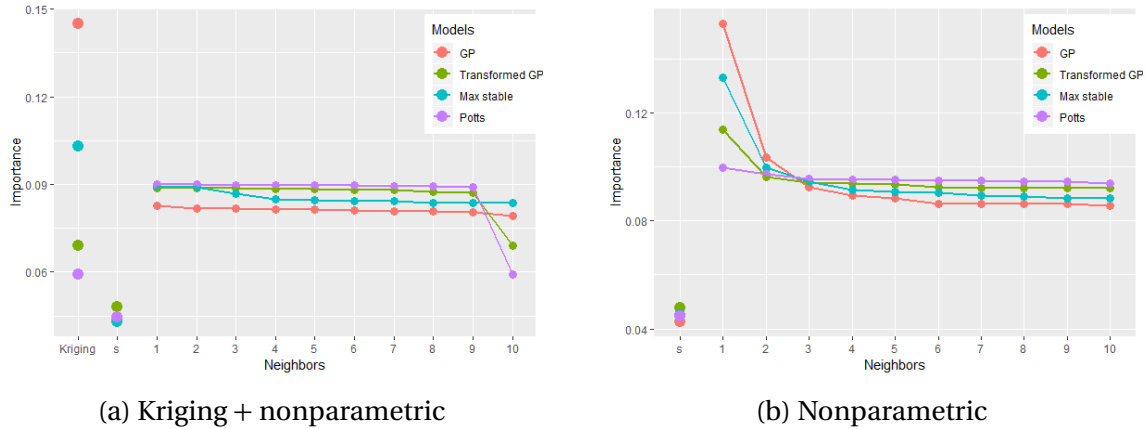


Figure 3.3: Relative variable importance (averaged over simulated datasets) for the Kriging feature, spatial location of the prediction site, and the total importance of the three features (latitude and longitude, difference to the prediction location, and response) for each neighbor, plotted so that $l = 1$ is the closest neighbor and $l = 10$ is the most distant.

generated from stationary processes.

3.4 Canopy height data analysis

In an effort to quantify forest carbon baseline and change, researchers need complete maps of forest biomass. These maps are used in forest management and global carbon monitoring and modeling efforts. Globally, canopy height represents a substantial source and sink in the global carbon cycle. Canopy Height Model (CHM) from NASA Goddard’s LiDAR Hyperspectral and Thermal (G-LiHT) (Cook et al. 2013) Airborne Imager over a subset of Harvard Forest Simes Tract, MA, was collected in Summer 2012. G-LiHT is a portable multi-sensor system that is used to collect fine spatial-scale image data of canopy structure, optical properties, and surface temperatures. Data we analyze here are downloaded from <https://gliht.gsfc.nasa.gov> and contain 1,723,137 observations. Locations are in UTM Zone 18 in the raw data, we convert them into standard longitude and latitude convention in the analysis. Figure 3.4 plots the data.

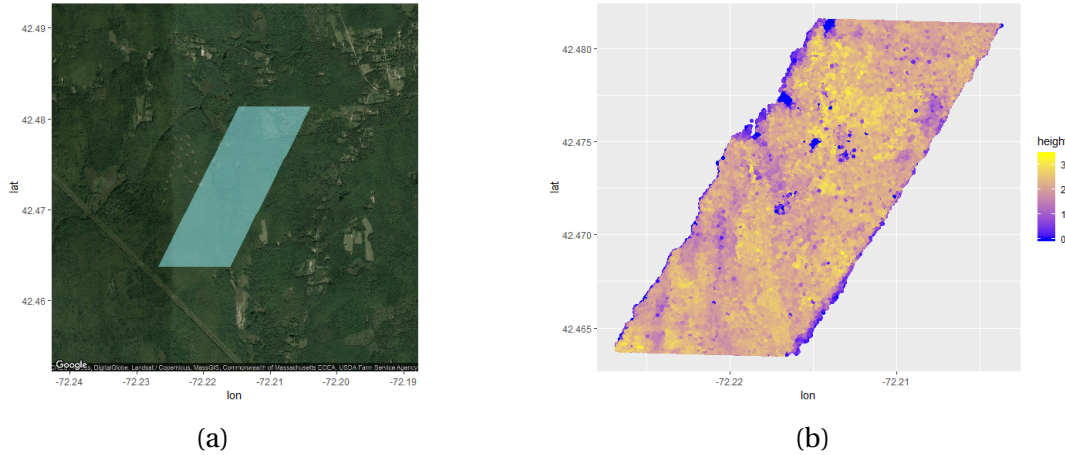


Figure 3.4: (a) Satellite image from Google map, with the blue diamond showing the area where the data are collected and (b) canopy height (meters) in the study area.

The same models, fitting procedures and performance metrics are applied to these data as in the simulation study. We subset 500,000 observations of the original dataset as training data and take the rest as testing data. The training dataset is further split with 90% for training and 10% for validation to tune the model. Table 4 shows the results. The NNGP method has the smallest MSE for estimating the mean, but all methods are similar. However, the 4N models outperform ALP model for quantile prediction in all cases. For $\gamma = 0.95$, the Kriging-only 4N model is the best with check loss 10% less than ALP model. For other γ values, Kriging + nonparametric model gives the best results, which are 10% to 14% better than ALP.

We also conduct a variable importance analysis for the CHM data (Figure 5). For all objective functions the Kriging feature is the most important, especially for MSE. The importance of neighboring information decreases with distance at a similar rate for all five objective functions. The spatial-location features account for approximately 5% of the total importance, suggesting nonstationarity is not strong for this analysis.

Table 3.4: Prediction performance for 4N models, NNGP and ALP for the canopy height data.

Loss function	4N(Kriging only)	4N(Nonparametric)	4N(Kriging+NP)	NNGP	ALP
MSE	1.435	1.436	1.433	1.430	NA
Check loss ($\gamma=0.25$)	1.283	1.184	1.157	NA	1.279
Check loss ($\gamma=0.5$)	1.350	1.364	1.301	NA	1.489
Check loss ($\gamma=0.75$)	1.112	1.194	1.034	NA	1.156
Check loss ($\gamma=0.95$)	0.360	0.413	0.386	NA	0.401

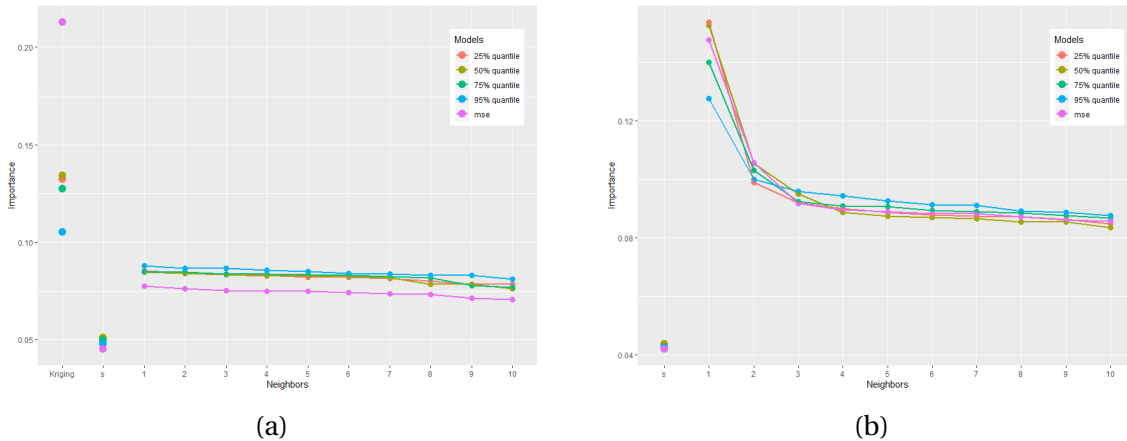


Figure 3.5: Relative Variable Importance for CHM Data

3.5 Conclusions

In this paper, we apply deep learning to geostatistical prediction. We formulate a flexible model that is a valid stochastic process and propose a series of new ways to construct features based on neighboring information. Our method outperforms state-of-art geostatistical methods for non-normal data. We also provide practical guidance on using MLP for modeling point-referenced geostatistical data that is fast and easy to implement. The 4N framework can be easily extended to other prediction tasks, such as classification, by modifying the activation and loss functions. In addition, the implementation of 4N models can utilize powerful GPU computing for further acceleration.

In geostatistics it is often desirable to attach a measure of uncertainty to spatial predictions and uncertainty quantification is known to be a challenge for deep learning methods.

A Bayesian implementation naturally provides measures of uncertainty, but at additional computational costs. Neural networks are related to Gaussian processes (Neal 1996a; Jaehoon et al. 2018) and this relationship can be used to provide uncertainty quantification (Graves 2011; Kingma et al. 2015; Blundell et al. 2015). Gal and Ghahramani (2016) prove dropout training in deep neural networks can be regarded as approximate Bayesian inference in deep Gaussian processes. One possibility that arises from our paper is to use quantile predictions resulting from check-loss optimization (e.g., Table 2) to form 95% prediction intervals. Applying this method in the simulation study gave coverage 98% for the GP, 94% for the transformed GP, 94% for the max stable process and 87% for the Potts model. For the CHM data, the 95% prediction interval coverage is exactly 95%. Therefore this approach warrants further study.

REFERENCES

- Adams, B. L., Ta'Asan, S., Kinderlehrer, D., Livshits, I., Mason, D., Wu, C.-T., Mullins, W., Rohrer, G., Rollett, A., and Saylor, D. (1999). Extracting grain boundary and surface energy from measurement of triple junction geometry. *Interface Science*, 7(3-4):321–337.
- Alder, B. J. and Wainwright, T. E. (1959). Studies in molecular dynamics. i. general method. *The Journal of Chemical Physics*, 31(2):459–466.
- Banerjee, S., Carlin, B. P., and E., A. (2014). *Hierarchical Modeling and Analysis for Spatial Data*. CRC Press.
- Barmak, K., Archibald, W., Kim, J., Kim, C. S., Rollett, A. D., Rohrer, G. S., Ta'asan, S., and Kinderlehrer, D. (2005). Grain boundary energy and grain growth in highly-textured al films and foils: experiment and simulation. In *Materials Science Forum*, volume 495, pages 1255–1260. Trans Tech Publ.
- Barmak, K., Kim, J., Kim, C.-S., Archibald, W., Rohrer, G., Rollett, A., Kinderlehrer, D., Ta'Asan, S., Zhang, H., and Srolovitz, D. (2006). Grain boundary energy and grain growth in al films: Comparison of experiments and simulations. *Scripta materialia*, 54(6):1059–1063.
- Beckmann, Thomas, Fischer, and Chang-Claude (2005). Haplotype sharing analysis using mantel statistics. *Human Heredity*, 59:67–78.
- Beladi, H., Nuhfer, N. T., and Rohrer, G. S. (2014). The five-parameter grain boundary character and energy distributions of a fully austenitic high-manganese steel using three dimensional data. *Acta Materialia*, 70:281–289.
- Beladi, H. and Rohrer, G. S. (2013). The relative grain boundary area and energy distributions in a ferritic steel determined from three-dimensional electron backscatter diffraction maps. *Acta materialia*, 61(4):1404–1412.
- Besse, P. C., Cardot, H., and Ferraty, F. (1997). Simultaneous non-parametric regressions of unbalanced longitudinal data. *Computational Statistics & Data Analysis*, 24(3):255 – 270.
- Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015). Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*.
- Bojarski, S. A., Knighting, J., Ma, S. L., Lenthe, W., Harmer, M. P., and Rohrer, G. S. (2013). The relationship between grain boundary energy, grain boundary complexion transitions, and grain size in ca-doped yttria. In *Materials Science Forum*, volume 753, pages 87–92. Trans Tech Publ.

- Bojarski, S. A., Ma, S., Lenthe, W., Harmer, M. P., and Rohrer, G. S. (2012). Changes in the grain boundary character and energy distributions resulting from a complexion transition in ca-doped yttria. *Metallurgical and Materials Transactions A*, 43(10):3532–3538.
- Cardot, H., Ferraty, F., and Sarda, P. (1999). Functional linear model. *Statistics & Probability Letters*, 45(1):11–22.
- Cardot, H., Goia, A., and Sarda, P. (2004). Testing for no effect in functional linear regression models, some computational approaches. *Communications in Statistics - Simulation and Computation*, 33(1):179–199.
- Chadwick, G. A. and Smith, D. A. (1976). *Grain boundary structure and properties*. Academic Press.
- Chen, D., Hall, P., and Muller, H.-G. (2011). Single and multiple index functional regression models with nonparametric link. *Annals of Statistics*, 39(3):1720–1747.
- Choi, I., Li, B., and Wang, X. (2013). Nonparametric estimation of spatial and space-time covariance function. *Journal of Agricultural, Biological, and Environmental Statistics*, 18(4):611–630.
- Cook, B. D., Nelson, R. F., Middleton, E. M., Morton, D. C., McCorkel, J. T., Masek, J. G., Ranson, K. J., Ly, V., Montesano, P. M., et al. (2013). NASA Goddard's LiDAR, Hyperspectral and Thermal (G-LiHT) Airborne Imager. *Remote Sensing*, 5(8):4045–4066.
- Cooley, D., Nychka, D., and Naveau, P. (2007). Bayesian spatial modeling of extreme precipitation return levels. *Journal of the American Statistical Association*, 102(479):824–840.
- Crainiceanu, C. M., Staicu, A. M., and Di, C. Z. (2009). Generalized multilevel functional regression. *Journal of the American Statistical Association*, 104(488):1550–1561.
- Cressie, N. (1992). Statistics for Spatial Data. *Terra Nova*, 4(5):613–617.
- Curtarolo, S., Hart, G. L., Nardelli, M. B., Mingo, N., Sanvito, S., and Levy, O. (2013). The high-throughput highway to computational materials design. *Nature materials*, 12(3):191–201.
- Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E. (2016). Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets. *Journal of the American Statistical Association*, 111(514):800–812.
- Delaigle, A. and Hall, P. (2012). Methodology and theory for partial least squares applied to functional data. *Annals of Statistics*, 40(1):322–352.
- Di, C.-Z., Crainiceanu, C., Caffo, B., and Punjabi, N. (2009). Multilevel functional principal component analysis. *The annals of applied statistics.*, 3(1):458–488.

- Di, Q., Kloog, I., Koutrakis, P., Lyapustin, A., Wang, Y., and Schwartz, J. (2016). Assessing PM_{2.5} exposures with high spatiotemporal resolution across the continental United States. *Environmental Science & Technology*, 50(9):4712–4721.
- Diggle, P. J., Tawn, J., and Moyeed, R. (1998). Model-based geostatistics. *Journal of the Royal Statistical Society: Series C*, 47(3):299–350.
- Dillon, S. J. and Rohrer, G. S. (2009). Characterization of the grain-boundary character and energy distributions of yttria using automated serial sectioning and ebsd in the fib. *Journal of the American Ceramic Society*, 92(7):1580–1585.
- Dimos, D., Chaudhari, P., and Mannhart, J. (1990). Superconducting transport properties of grain boundaries in $\text{YBa}_2\text{Cu}_3\text{O}_7$ bicrystals. *Physical Review B*, 41(7):4038.
- Duan, J. A., Guindani, M., and Gelfand, A. E. (2007). Generalized spatial Dirichlet process models. *Biometrika*, 94(4):809–825.
- Duane, S., Kennedy, A., Pendleton, B. J., and Roweth, D. (1987). Hybrid monte carlo. *Physics Letters B*, 195(2):216 – 222.
- Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360.
- Ferguson, T. S. (2014). *Mathematical statistics: A decision theoretic approach*, volume 1. Academic Press.
- Ferraty, F., Goia, A., and Salinelli, E. (2013). Functional projection pursuit regression. *TEST*, 22:293.
- Forsyth, P., King, R., Metcalfe, G., and Chalmers, B. (1946). Grain boundaries in metals. *Nature*, 158(4024):875–876.
- Fouladgar, M., Parchami, M., Elmasri, R., and Ghaderi, A. (2017). Scalable deep traffic flow neural networks for urban traffic congestion prediction. In *Neural Networks (IJCNN), 2017 International Joint Conference on*, pages 2251–2258. IEEE.
- Gal, Y. and Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059.
- Gareth M. James, J. W. and Zhu, J. (2009). Functional linear regression that’s interpretable. *Annals of Statistics*, 37(5A):2083–2108.
- Garson, G. D. (1991). Interpreting Neural-network Connection Weights. *AI Expert*, 6(4):46–51.

- Gelfand, A. E., Diggle, P., Guttorp, P., and Fuentes, M. (2010). *Handbook of spatial statistics*. CRC Press.
- Gelfand, A. E., Kottas, A., and MacEachern, S. N. (2005). Bayesian nonparametric spatial modeling with Dirichlet process mixing. *Journal of the American Statistical Association*, 100(471):1021–1035.
- Gelfand, A. E. and Schliep, E. M. (2016). Spatial Statistics and Gaussian Processes: A Beautiful Marriage. *Spatial Statistics*, 18:86–104.
- Gertheiss, J., Maity, A., and Staicu, A.-M. (2013). Variable selection in generalized functional linear models. *Stat*, 2(1):86–101.
- Gevrey, M., Dimopoulos, I., and Lek, S. (2003). Review and comparison of methods to study the contribution of variables in artificial neural network models. *Ecological Modelling*, 160(3):249–264.
- Gleiter, H. (1981). The interaction of point defects, dislocations and two-dimensional defects with grain boundaries. *Progress in Materials Science—Chalmers Anniversary Volume*, pages 125–183.
- Gleiter, H. (1982). On the structure of grain boundaries in metals. In *Interfacial Aspects of Phase Transformations*, pages 199–222. Springer.
- Goldsmith, J., Bobb, J., Crainiceanu, C. M., Caffo, B., and Reich, D. (2011). Penalized functional regression. *Journal of Computational and Graphical Statistics*, 20(4):830–851.
- Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT Press Cambridge.
- Gottstein, G. and Shvindlerman, L. S. (2009). *Grain boundary migration in metals: thermodynamics, kinetics, applications*. CRC press.
- Gramacy, R. B. and Apley, D. W. (2015). Local Gaussian process approximation for large computer experiments. *Journal of Computational and Graphical Statistics*, 24(2):561–578.
- Gramacy, R. B. and Lian, H. (2012). Gaussian process single-index models as emulators for computer experiments. *Technometrics*, 54(1):30–41.
- Gramacy, R. B., Niemi, J., and Weiss, R. M. (2014). Massively parallel approximate Gaussian process regression. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):564–584.
- Graves, A. (2011). Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*, pages 2348–2356.
- Guinness, J. (2018). Permutation and Grouping Methods for Sharpening Gaussian Process Approximations. *Technometrics*, (just-accepted).

- Hasson, G. and Goux, C. (1971). Interfacial energies of tilt boundaries in aluminium. experimental and theoretical determination. *Scripta metallurgica*, 5(10):889–894.
- Hastie, T. J. (2017). Generalized additive models. In *Statistical models in S*, pages 249–307. Routledge.
- Haynes, C. and Smoluchowski, R. (1955). Grain boundary diffusion in a body-centered cubic lattice. *Acta metallurgica*, 3(2):130–134.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778.
- Herring, C. (1951). Some theorems on the free energies of crystal surfaces. *Physical Review*, 82(1):87.
- Hirth, J. P. (1972). The influence of grain boundaries on mechanical properties. *Metallurgical Transactions*, 3(12):3047–3067.
- Holm, E. A., Olmsted, D. L., and Foiles, S. M. (2010). Comparing grain boundary energies in face-centered cubic metals: Al, au, cu and ni. *Scripta Materialia*, 63(9):905–908.
- Homer, E. R., Holm, E. A., Foiles, S. M., and Olmsted, D. L. (2014). Trends in grain boundary mobility: Survey of motion mechanisms. *JOM*, 66(1):114–120.
- Huang, C., Hsing, T., and Cressie, N. (2011). Nonparametric estimation of the variogram and its spectrum. *Biometrika*, 98(4):775–789.
- Hunderi, O. (1973). Influence of grain boundaries and lattice defects on the optical properties of some metals. *Physical Review B*, 7(8):3419.
- Im, H. K., Stein, M. L., and Zhu, Z. (2007). Semiparametric estimation of spectral density with irregular observations. *Journal of the American Statistical Association*, 102(478):726–735.
- Ishwaran, H. (1999). Applications of hybrid monte carlo to bayesian generalized linear models: Quasicomplete separation and neural networks. *Journal of Computational and Graphical Statistics*, 8(4):779–799.
- Jaehoon, L., Yasaman, B., Roman, N., Sam, S., Jeffrey, P., and Jascha, S.-d. (2018). Deep Neural Networks as Gaussian Processes. *International Conference on Learning Representations*.
- Jain, A., Hautier, G., Moore, C. J., Ong, S. P., Fischer, C. C., Mueller, T., Persson, K. A., and Ceder, G. (2011). A high-throughput infrastructure for density functional theory calculations. *Computational Materials Science*, 50(8):2295–2310.

- Jain, A., Ong, S. P., Hautier, G., Chen, W., Richards, W. D., Dacek, S., Cholia, S., Gunter, D., Skinner, D., Ceder, G., et al. (2013). Commentary: The materials project: A materials genome approach to accelerating materials innovation. *Apl Materials*, 1(1):011002.
- James, G., Hastie, T., and Sugar, C. (2000). Principal component models for sparse functional data. *Biometrika*, 87(3):587–602.
- James, G. M. and Silverman, B. W. (2005). Functional adaptive model estimation. *Journal of the American Statistical Association*, 100(470):565–576.
- Jiang, C.-R. and Wang, J.-L. (2011). Functional single index models for longitudinal data. *Annals of Statistics*, 39(1):362–388.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kingma, D. P., Salimans, T., and Welling, M. (2015). Variational dropout and the local reparameterization trick. In *Advances in Neural Information Processing Systems*, pages 2575–2583.
- Lauritzen, S. L. (1996). *Graphical models*, volume 17. Clarendon Press.
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436.
- Li, J., Dillon, S. J., and Rohrer, G. S. (2009). Relative grain boundary area and energy distributions in nickel. *Acta Materialia*, 57(14):4304–4311.
- Lian, H. (2011). Functional partial linear model. *Journal of Nonparametric Statistics*, 23(1):115–128.
- Liang, F, Li, Q., and Zhou, L. (2018). Bayesian neural networks for selection of drug sensitive genes. *Journal of the American Statistical Association*, 113(523):955–972.
- Liu, D., Lin, X., and Ghosh, D. (2007). Semiparametric regression of multidimensional genetic pathway data: least-squares kernel machines and linear mixed models. *Biometrics*, 63(4):1079–88.
- Liu, J. S. (2008). *Monte Carlo strategies in scientific computing*. Springer Science & Business Media.
- Locantore, N., Marron, J., and Simpson, D. (1999). Robust principal component analysis for functional data. *TEST*, 8(1):1–73.
- Lum, K., Gelfand, A. E., et al. (2012). Spatial quantile multiple regression using the asymmetric Laplace process. *Bayesian Analysis*, 7(2):235–258.

- Matsui, H. and Konishi, S. (2011). Variable selection for functional regression models via the l1 regularization. *Computational Statistics & Data Analysis*, 55(12):3304 – 3310.
- Miura, H., Kato, M., and Mori, T. (1994). Temperature dependence of the energy of cu [110] symmetrical tilt grain boundaries. *Journal of materials science letters*, 13(1):46–48.
- Moller, J. and Waagepetersen, R. P. (2003). *Statistical Inference and Simulation for Spatial Point Processes*. CRC Press.
- Morawiec, A. (2000). Method to calculate the grain boundary energy distribution over the space of macroscopic boundary parameters from the geometry of triple junctions. *Acta Materialia*, 48(13):3525–3532.
- Morawiec, A. (2003). *Orientations and rotations*. Springer.
- Muller, H. G. and Stadtmuller, U. (2005). Generalized functional linear models. *Annals of Statistics*, pages 774–805.
- Neal, R. M. (1993). Probabilistic inference using markov chain monte carlo methods.
- Neal, R. M. (1996a). Priors for infinite networks. In *Bayesian Learning for Neural Networks*, pages 29–53. Springer.
- Neal, R. M. (1996b). Sampling from multimodal distributions using tempered transitions. *Statistics and Computing*, 6(4):353–366.
- Neal, R. M. (2012). *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media.
- Olmsted, D. L., Foiles, S. M., and Holm, E. A. (2009a). Survey of computed grain boundary properties in face-centered cubic metals: I. grain boundary energy. *Acta Materialia*, 57(13):3694–3703.
- Olmsted, D. L., Holm, E. A., and Foiles, S. M. (2009b). Survey of computed grain boundary properties in face-centered cubic metals - ii: Grain boundary mobility. *Acta materialia*, 57(13):3704–3713.
- Pakman, A. and Paninski, L. (2014). Exact hamiltonian monte carlo for truncated multivariate gaussians. *Journal of Computational and Graphical Statistics*, 23(2):518–542.
- Panchal, J. H., Kalidindi, S. R., and McDowell, D. L. (2013). Key computational modeling issues in integrated computational materials engineering. *Computer-Aided Design*, 45(1):4–25.
- Patala, S., Mason, J. K., and Schuh, C. A. (2012). Improved representations of misorientation information for grain boundary science and engineering. *Progress in Materials Science*, 57(8):1383–1425.

- Patala, S. and Schuh, C. A. (2013). Symmetries in the representation of grain boundary-plane distributions. *Philosophical Magazine*, 93(5):524–573.
- Potts, R. B. (1952). Some generalized order-disorder transformations. *Mathematical Proceedings of the Cambridge Philosophical Society*, 48(1):106 – 109.
- Qian and Thomas (2001). Genome scan of complex traits by haplotype sharing correlation. *Genetic Epidemiology*, 21(1):S582–S587.
- Ramamoorthi, R. and Hanrahan, P. (2001). An efficient representation for irradiance environment maps. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 497–500. ACM.
- Ramsay, J. and Silverman, B. (2005). *Functional data analysis*. Springer.
- Rasmussen, C. E. (2004). Gaussian processes in machine learning. In *Advanced Lectures on Machine Learning*, pages 63–71. Springer.
- Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press.
- Reich, B. J. (2012). Spatiotemporal quantile regression for detecting distributional changes in environmental processes. *Journal of the Royal Statistical Society: Series C*, 61(4):535–553.
- Reich, B. J. and Fuentes, M. (2007). A multivariate semiparametric Bayesian spatial modeling framework for hurricane surface wind fields. *The Annals of Applied Statistics*, 1(1):249–264.
- Reich, B. J. and Fuentes, M. (2015). Spatial Bayesian Nonparametric Methods. In *Nonparametric Bayesian Inference in Biostatistics*, pages 347–357. Springer.
- Reiss, P. T. and Ogden, R. T. (2007). Functional principal component regression and functional partial least squares. *Journal of the American Statistical Association*, 102(479):984–996.
- Ripley, B. D. (2005). *Spatial statistics*, volume 575. John Wiley & Sons.
- Risser, M. D. (2016). Nonstationary Spatial Modeling, with Emphasis on Process Convolution and Covariate-Driven Approaches. *arXiv preprint arXiv:1610.02447*.
- Rodrigues, F. and Pereira, F. C. (2018). Beyond expectation: Deep joint mean and quantile regression for spatio-temporal problems. *CoRR*, abs/1808.08798.
- Rodriguez, A. and Dunson, D. B. (2011). Nonparametric Bayesian Models Through Probit Stick-breaking Processes. *Bayesian Analysis*, 6(1).

- Rohrer, G. S. (2011). Grain boundary energy anisotropy: a review. *Journal of materials science*, 46(18):5881–5895.
- Rollett, A., Yang, C., Mullins, W., Adams, B., Wu, C., Kinderlehrer, D., Ta'asan, S., Manolache, F., Liu, C., Livshits, I., et al. (2001). Grain boundary property determination through measurement of triple junction geometry and crystallography. In *Proceedings of the First Joint International Conference on Grain Growth, Aachen, Germany, Eds. G. Gottstein and DA Molodov, (Springer Verlag, 2001) pp*, pages 165–175.
- Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*. CRC Press.
- Saal, J. E., Kirklın, S., Aykol, M., Meredig, B., and Wolverton, C. (2013). Materials design and discovery with high-throughput density functional theory: the open quantum materials database (oqmd). *Jom*, 65(11):1501–1509.
- Sang, H. and Gelfand, A. E. (2010). Continuous spatial process models for spatial extreme values. *Journal of Agricultural, Biological, and Environmental Statistics*, 15(1):49–65.
- Saylor, D. M., Morawiec, A., Adams, B. L., and Rohrer, G. S. (2000). Misorientation dependence of the grain boundary energy in magnesia. *Interface Science*, 8(2-3):131–140.
- Saylor, D. M., Morawiec, A., and Rohrer, G. S. (2002). Distribution and energies of grain boundaries in magnesia as a function of five degrees of freedom. *Journal of the American Ceramic Society*, 85(12):3081–3083.
- Saylor, D. M., Morawiec, A., and Rohrer, G. S. (2003). The relative free energies of grain boundaries in magnesia as a function of five macroscopic parameters. *Acta materialia*, 51(13):3675–3686.
- Schlather, M. (2002). Models for stationary max-stable random fields. *Extremes*, 5(1):33–44.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117.
- Schmidt, M. N. (2009). Function factorization using warped gaussian processes. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 921–928. ACM.
- Scholkopf, B. and Smola, A. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT Press.
- Schölkopf, B., Sung, K.-K., Burges, C. J., Girosi, F., Niyogi, P., Poggio, T., and Vapnik, V. (1997). Comparing support vector machines with gaussian kernels to radial basis function classifiers. *Signal Processing, IEEE Transactions on*, 45(11):2758–2765.

- Seita, M., Volpi, M., Patala, S., McCue, I., Schuh, C. A., Diamanti, M. V., Erlebacher, J., and Demkowicz, M. J. (2016). A high-throughput technique for determining grain boundary character non-destructively in microstructures with through-thickness grains. *npj Computational Materials*, Accepted.
- Shi, J. Q. and Choi, T. (2008). *Gaussian Process Regression Analysis for Functional Data*. CRC Press.
- Shi, J. Q. and Wang, B. (2008). Curve prediction and clustering with mixtures of gaussian process functional regression models. *Statistics and Computing*, 18(3):267–283.
- Shi, J. Q., Wang, B., Murray-Smith, R., and Titterton, D. M. (2007). Gaussian process functional regression modeling for batch data. *Biometrics*, 63(3):714–723.
- Smoluchowski, R. (1952). Theory of grain boundary diffusion. *Physical Review*, 87(3):482.
- Stein, M. L. (2012). *Interpolation of spatial data: some theory for Kriging*. Springer Science & Business Media.
- Stein, M. L., Chi, Z., and Welty, L. J. (2004). Approximating likelihoods for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(2):275–296.
- Sutton, A. P. and Balluffi, R. W. (1995). *Interfaces in crystalline materials*. Clarendon Press.
- Suykens, J., Gestel, T. V., and Brabanter, J. D. (2002). *Least Squares support vector machines*. World Scientific Publishing.
- Tzeng and Zhang (2007). Haplotype-based association analysis via variance-components score test. *American Journal of Human Genetics*, 81(5):927–938.
- Tzeng, J.-Y., Zhang, D., Chang, S.-M., Thomas, D. C., and Davidian, M. (2009). Gene-trait similarity regression for multimarker-based association analysis. *Biometrics*, 65(3):822–832.
- Vapnik, V. (1998). *Statistical learning theory*. Wiley-Interscience.
- Vecchia, A. V. (1988). Estimation and model identification for continuous spatial processes. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 297–312.
- Viviani, R., GrÅn, G., and Spitzer, M. (2005). Functional principal component analysis of fmri data. *Human Brain Mapping*, 24(2):109–129.
- Walczak, B. and Massart, D. (1996). The radial basis functionsâpartial least squares approach as a flexible non-linear regression technique. *Analytica Chimica Acta*, 331(3):177–185.

- Wang, B. and Shi, J. Q. (2014). Generalized gaussian process regression model for non-gaussian functional data. *Journal of the American Statistical Association*, 109(507):1123–1133.
- Wessel and Schork (2006). Generalized genomic distance-based regression methodology for multilocus association analysis. *The American Journal of Human Genetics*, 79:792–806.
- Whittle, P. (1954). On stationary processes in the plane. *Biometrika*, pages 434–449.
- Xingjian, S., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems*, pages 802–810.
- Xiong, S., Dai, B., Huling, J., and Qian, P. Z. (2016). Orthogonalizing em: A design-based least squares algorithm. *Technometrics*, 58(3):285–293.
- Yingying Fan, G. M. J. and Radchenko, P. (2015). Functional additive regression. *Annals of Statistics*, 43(5):2296–2325.
- Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.
- Zhang, J., Zheng, Y., and Qi, D. (2017). Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. In *AAAI*, pages 1655–1661.
- Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429.
- Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.

APPENDICES

APPENDIX

A

SUPPLEMENTARY ANALYSIS RESULTS

In Chapter 1, before we analyze the datasets with 16 EMG signals, we first apply our method on a subset of the 16 variables. We denote these 4 signals as EMG 1 - 4. We use a different way to split the data, we will describe it shortly. We concurrently capture signals from 16 EMG sensors (X_1, \dots, X_{16}) on a subject's arm and also use a motion capturing device to record the position of the subject's fingers(Y). These signals are recorded for a long sequence of time but we are able to chop up the entire time window into smaller time windows of any size that we want. Here, the figure below results from asking a subject to open and close their hand for a period of time. The entire time window was then split up into time periods of 210 time units. In addition, we also consider the wrist movement as our response. There are in total 14 observations. In Figure A.1, a value of $Y = 25$ radians is usually where our

hand is at a resting position (where our hand would be if we were not contracting muscles). Values less than 20 indicate our hand is in a more open position and values greater than 20 indicate our hand is in a close position.

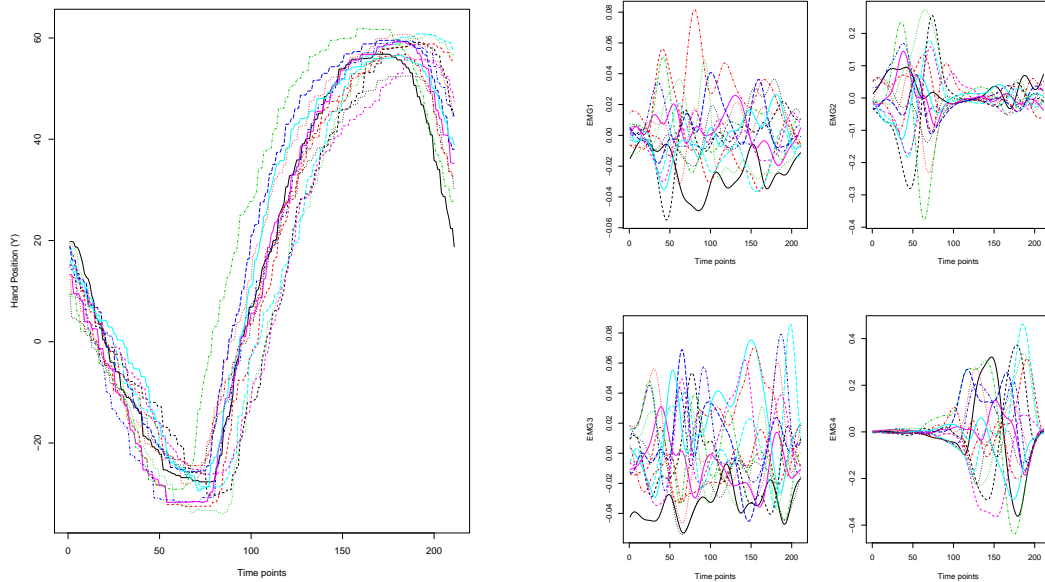


Figure A.1: Hand Positions(4 Variables) Figure A.2: 4 Variables(Subtract from mean)

We would like to determine whether the concurrent EMG signals can explain the variation of the curves. Figure A.2 shows the 4 different EMG signals from muscles. To begin with, we first apply our method on a subset of the 16 variables. We then use our variable selection method to pick up the important signals. We want to find out which signal is associated with hand opening and which signal is associated with hand closing, so we consider hand opening (the first half of data points in Figure A.1), hand closing (the second half of data points in Figure A.1) and both hand opening and hand closing, separately. First, we do Principle Component Analysis on the hand positions, and take the first score as our

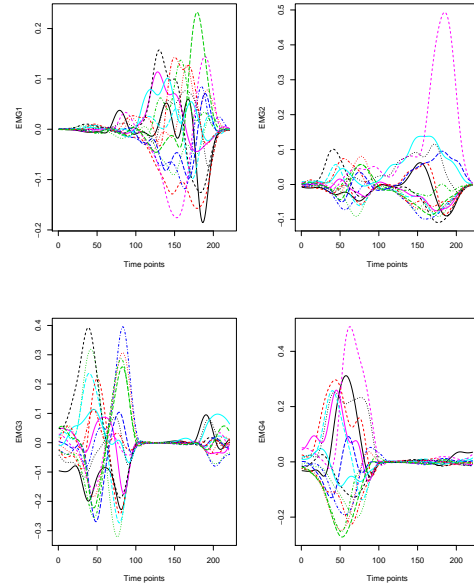
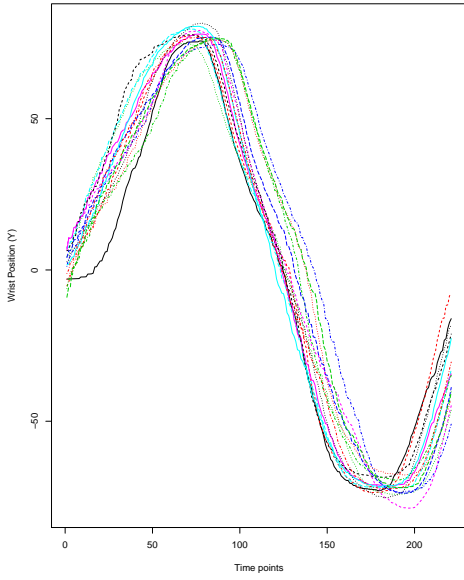


Figure A.3: Wrist Positions(4 Variables)

Figure A.4: 4 Variables(Subtract from mean)

Table A.1: Proportion of Variance Explained by the First Five Principle Components

	1st	2nd	3rd	4th	5th
Hand Open	73%	18%	5%	2%	1%
Hand Close	78%	17%	3%	1%	1%
Both	74%	11%	6%	4%	2%

response in the model. The reason to do this is that the first principle component of the hand position will explain most of the variance, as shown in Table A.1.

Here we use polynomial kernel $K(x, x') = (x^T x + 1)^d$, where d is the highest degree, which should be chosen properly. We thus try different highest degrees ranging from 1 to 3, and track the chosen variables under each condition. The results are listed in Table A.2. To get more stable results, we apply the procedure 20 times for each case. We do similar things on the wrist response, for the 4 signal case and the results are shown in A.3. EMG 2 and 4 are related to hand movement and EMG 1 and 3 are related to wrist movement.

Table A.2: Variable Selection Results for Hand Movement (4 signals)

Highest Degree	Hand Open	Hand Close	Both
1	EMG2(48%),EMG4(3%)	EMG4(100%)	EMG4(100%)
2	EMG2(100%)	EMG2(18%),EMG4(78%)	EMG1(1%),EMG2(99%)
3	EMG2(100%)	EMG4(100%)	EMG2(86%),EMG4(100%)

Table A.3: Variable Selection Results for Wrist Movement (4 signals)

Highest Degree	Hand Open	Hand Close	Both
1	EMG3(100%)	None	EMG3(100%)
2	EMG3(100%)	None	EMG3(90%)
3	EMG3(100%)	None	EMG1(29%),EMG3(46%)

We then conduct our algorithm to the full dataset with 16 EMG signals, and we focus on hand movement only. Figure A.5 shows hand positions. According to Figure A.6, some of the signals are highly correlated. Therefore, we conduct hierarchical cluster on the signals to have a better understanding of the signals. As shown in Figure A.7, the 16 signals are clustered in several groups. For example, Signal 1, 5, 7 are in the same cluster and signal 8, 10 and 12 are in another cluster. Tables A.4 and A.5 show the variable selection results for the 16 EMG data. EMG 7 is picked up almost every time, and is related to hand opening. EMG 8 and 12 are picked up quite frequently, and are related to hand closing. Figures A.8 and A.9 show the results by variable and by cluster, respectively.

Table A.4: Variable Selection Results for Hand Movement(16 variables)

Degree	Hand Open	Hand Close
1	EMG3(15%),EMG7(69%)	EMG8(100%),EMG12(100%)
2	EMG7(100%)	EMG1(23%),EMG10(93%),EMG12(100%)
3	EMG7(100%)	EMG1(92%),EMG8,11(3%),EMG12(89%)

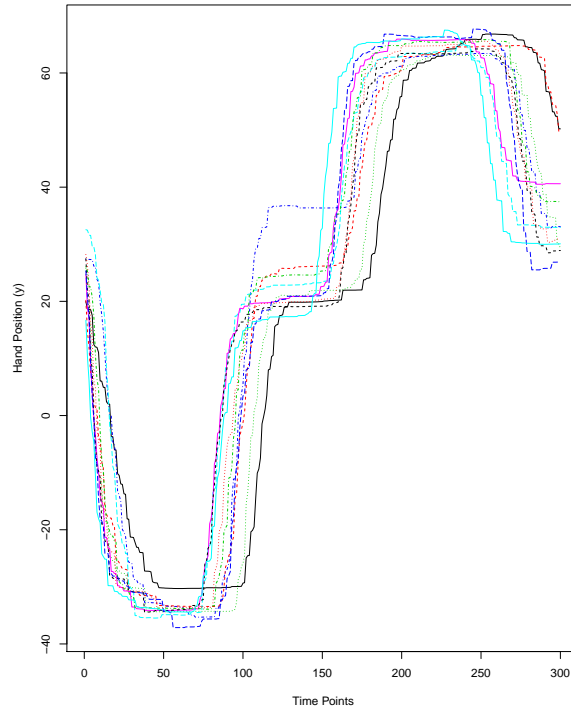


Figure A.5: Hand Positions (16 Variables)

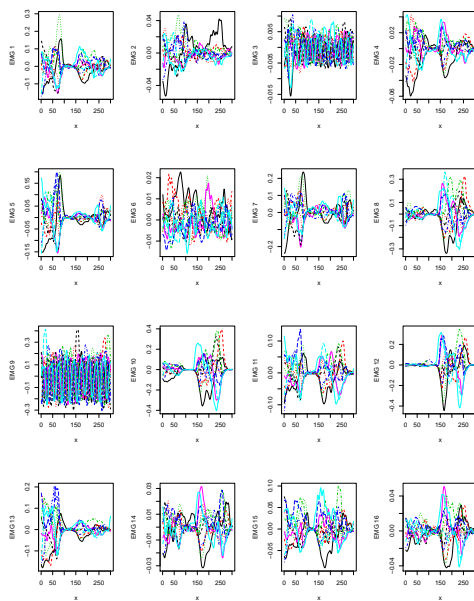


Figure A.6: EMG 1 - 16 (Subtracted from mean)

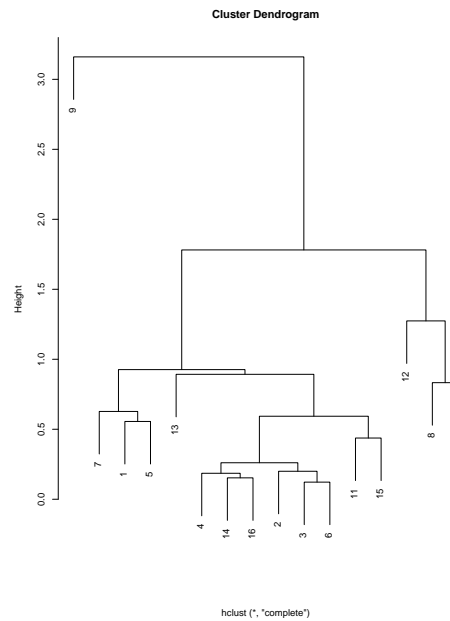


Figure A.7: Hierarchical Clustering on 16 EMG Signals

Table A.5: Variable Selection Results for Hand Movement(16 variables) Cont.

Degree	Both
1	EMG7(79%),EMG8(99%),EMG12(99%),EMG14(1%)
2	EMG3(1%),EMG7(99%),EMG10(22%)
3	EMG4(1%),EMG5(40%),EMG7(100%),EMG8(29%),EMG11(49%),EMG12(29%)

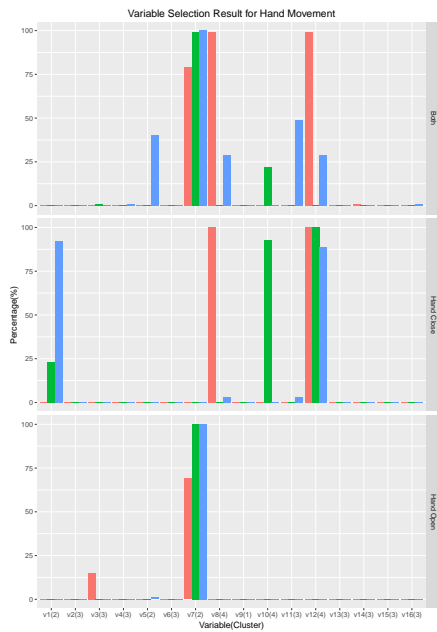


Figure A.8: Variable Selection Result for Hand Movement (by variable)

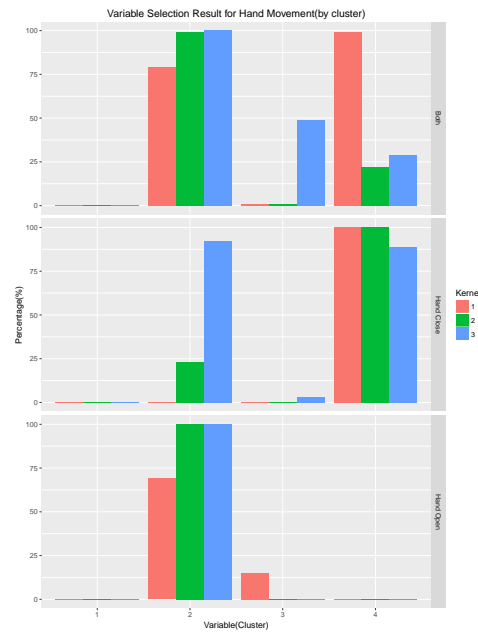


Figure A.9: Variable Selection Result for Hand Movement (by cluster)

APPENDIX

B

HAMILTONIAN MONTE CARLO

Hamiltonian Monte Carlo origins from a physics concept Hamiltonian dynamics:

$$H(q, p) = U(q) + K(p),$$

where $U(q)$ is potential energy, which is defined to be negative log pdf of q that we wish to sample; $K(p)$ is kinetic energy, p defined to be "momentum" variables; q and p are functions of time t . There are several methods to approximate the Hamiltonian dynamics, the most commonly used one is the Leapfrog method, which is defined as

$$p(t + \epsilon/2) = p(t) - (\epsilon/2) \frac{\partial U}{\partial q}(q(t)), q(t + \epsilon) = q(t) + \epsilon \frac{p(t + \epsilon/2)}{m},$$

where ϵ is stepsize. Figure B.1 shows an illustrative example of Leapfrog method with different stepsizes.

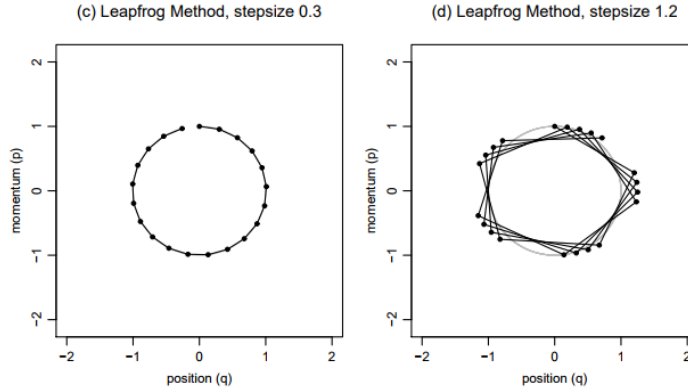


Figure B.1: Leapfrog method with different stepsizes

Suppose joint distribution for p and q is

$$P(q, p) = \frac{1}{Z} \exp(-H(q, p)/T),$$

where Z is a normalization factor and T is a constant value, Hamiltonian Monte Carlo consists of three steps. First, we randomly draw new values for momentum variables p from Gaussian distribution. Second, we propose a new state (q^*, p^*) for L steps with stepsize ϵ using the Leapfrog Method. Finally, we accept the proposed state with probability $\min[1, \exp(-H(q^*, p^*) + H(q, p))]$. We are interested in sampling from truncated multivariate Gaussian distributions. For instance, we consider sampling from

$$\log p(X) = -\frac{1}{2} X^T X + c,$$

subject to

$$F_j X + g_j \geq 0.$$

We introduce momentum variables S , so that $H = \frac{1}{2}X^T X + \frac{1}{2}S^T S$. We then solve differential equations to get $x_i(t) = a_i \sin(t) + b_i \cos(t)$.

When the sample hits a wall, meaning that any of the linear inequalities is violated, the trajectory continues with a reflected velocity. Once the reflected velocity is computed, we use it as initial condition to continue the trajectory. Figure B.2 shows an example of sampling from a two-dimensional distribution with $\log p(x, y) \propto -\frac{1}{2}(x-4)^2 - \frac{1}{2}(y-4)^2$, constrained to $x \leq y \leq 1.1x$ and $x, y \geq 0$ using Exact Hamiltonian Monte Carlo method.

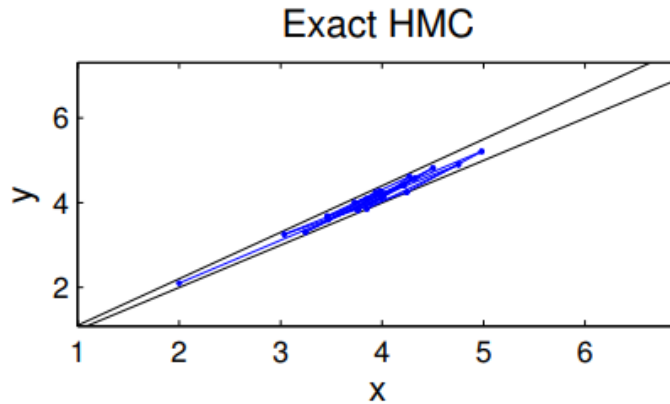


Figure B.2: Sampling from a two-dimensional distribution with $\log p(x, y) \propto -\frac{1}{2}(x-4)^2 - \frac{1}{2}(y-4)^2$, constrained to $x \leq y \leq 1.1x$ and $x, y \geq 0$, using Exact Hamiltonian Monte Carlo method.