Some Advanced Model Selection Topics for Nonparametric/Semiparametric Models with High-Dimensional Data

Zaili Fang

Dissertation submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Statistics

Inyoung Kim, Committee Chair
Eric P. Smith
George R. Terrell
Pang Du
Scotland C. Leman

November 3, 2012
Blacksburg, Virginia

KEYWORDS: Additive Model; Cluster Algorithm; Gaussian Random Process; Global-Local Shrinkage; Graphical Model; Ising Model; Kernel Machine; KM Model; LASSO; Long Tail Prior; Mixture Normals; Model Selection; Multivariate Smoothing Function; Nonnegative Garrote; Nonparametric Model; Pathway Analysis; Semiparametric Model; Sparsistency; Smoothing Splines; Variable Selection.

Some Advanced Model Selection Topics for Nonparametric/Semiparametric Models with High-Dimensional Data

Zaili Fang

(ABSTRACT)

Model and variable selection have attracted considerable attention in areas of application where datasets usually contain thousands of variables. Variable selection is a critical step to reduce the dimension of high dimensional data by eliminating irrelevant variables. The general objective of variable selection is not only to obtain a set of cost-effective predictors selected but also to improve prediction and prediction variance. We have made several contributions to this issue through a range of advanced topics: providing a graphical view of Bayesian Variable Selection (BVS), recovering sparsity in multivariate nonparametric models and proposing a testing procedure for evaluating nonlinear interaction effect in a semiparametric model.

To address the first topic, we propose a new Bayesian variable selection approach via the graphical model and the Ising model, which we refer to the "Bayesian Ising Graphical Model" (BIGM). There are several advantages of our BIGM: it is easy to (1) employ the single-site updating and cluster updating algorithm, both of which are suitable for problems with small sample sizes and a larger number of variables, (2) extend this approach to nonparametric regression models, and (3) incorporate graphical prior information.

In the second topic, we propose a Nonnegative Garrote on a Kernel machine (NGK) to recover sparsity of input variables in smoothing functions. We model the smoothing function by a least squares kernel machine and construct a nonnegative garrote on the kernel model as the function of the similarity matrix. An efficient coordinate descent/backfitting algorithm is developed.

The third topic involves a specific genetic pathway dataset in which the pathways interact with the environmental variables. We propose a semiparametric method to model the pathway-environment interaction. We then employ a restricted likelihood ratio test and a score test to evaluate the main pathway effect and the pathway-environment interaction.

To Father and Mother

Acknowledgments

I would like to express my sincere gratitude and special thanks to my supervisor, Dr. Inyoung Kim, who has provided her guidance, support, and encouragement throughout my graduate studies.

Also thanks to my committee members, Dr. Eric P. Smith, Dr. George Terrell, Dr. Pang Du and Dr. Scotland Leman, who offered value advice and support.

I have also had the privilege to talk and discuss with a number of talented individual in Department of Statistics at Virginia Tech. Thank you all the professors for your inspiring courses and guidance. Thank you all my friends for your generous collaboration and discussion on statistical issues that widened my vision. I also would like to thank all staff of Department of Statistics for the selfless support to the students.

Lastly, I would like to thank my beloved parents, for their great and never-ending support companying with me all the way. None of this would be possible without the love and support of my family.

Contents

List of Figures

List of Tables

Chapter 1

Outline of this Dissertation

Model selection is a very general statistical term to describe methods for selecting a statistical model from a set of candidates, which typically involves a set of regressor variables or perhaps model terms (transforms of the natural variables). One very popular model selection problem is variable and feature selection for selecting significant predictors from tens, hundreds or thousands of candidates, which has become the focus of much research in recently years. The most popular variable selection technique based on linear regression is Least Absolute Shrinkage and Selection Operator (LASSO) (Efron et al., 2004; Tibshirani, 1996). However, there are several issues associated with variable selection where LASSO does not work well, such as the function components selection of an additive model, or the input variable selection of a Gaussian process or neural network. Usually we apply nonparametric regression methods to model those nonlinear and nonadditive functions. To address variable selection in the nonparametric regression model, smoothing spline and kernel machine techniques have been employed in many ways. To develop a variable selection procedure that is practical for these situations is still an interesting and challenging topic. We explore some advanced variable selection

problems from both frequentist and Bayesian point of view. In the frequentist statistics side, employing penalized norm of the nonlinear function space is the standard approach to reduce the dimension, while in the Bayesian statistics side, finding an efficient updating sampler is the typical research goal for the stochastic searching of the binary random variables corresponding to the inclusion or exclusion of the predictors. Nevertheless, the model selection task can also involve the hypothesis testing of the main effects and interaction effects in a mixed model (Goeman et al., 2004; Zhang et al., 1998; Zhang and Lin, 2003), which becomes complicated when an semi/nonparametric mixed models are considered to establish the relationship between the response and the predictors.

Hence, this dissertation focuses on these three major problems in model selection. The outline of the dissertation is sketched as follows:

- In Chapter 2, we discuss a new Bayesian variable selection (BVS) approach via the graphical model and the Ising model, which we refer to the "Bayesian Ising Graphical Model" (BIGM). In this chapter, we are mainly interested in the following issues:
  - Model Description: we connect the regular linear regression model with the graphical model for purpose of variable selection.
  - Methodology: we generalize the cluster algorithm existing in the Ising model with fixed interaction to a cluster algorithm applicable to a complete binary random graphical model with random interactions.
  - Extension: we provide an extension of BIGM to Bayesian additive models for function component selection, and we improve the performance of BIGM by incorporating the network information of the graphical model's nodes.
  - Understanding BVS: we study the selection probability profile curves and evaluate the performance of different priors for the model coefficients with dif-

ferent characteristics in heavy mass around zero to long-tailed prior. We also discuss the connection between shrinkage parameters and tempering parameters.

  - Simulation studies and applications.

- In Chapter 3, we introduce a flexible variable selection approach for recovering the sparsity in the nonadditive or additive multivariate nonparametric models. In this chapter, the following major issues are discussed:

  - Model Description: we extend the nonnegative garrote method to a nonlinear nonparametric model.
  - Methodology: we propose a coordinate descent or backfitting algorithm to solve the problem.
  - Theoretical Properties: we provide theoretical results to show the sparsistency (sparsity consistency) of our method.
  - Simulation studies and applications.

- In Chapter 4, we discuss a semiparametric mixed model for evaluating pathway-environment interaction. In this chapter, we will focus on the following issues:

  - Model Description: we establish the semiparametric model to describe the environmental variable and the pathway covariates and their interaction.
  - Estimation: we estimate the parameters with two methods, Restricted Maximum Likelihood (REML) and profile REML.
  - Hypothesis Testing: a profile Restricted Likelihood Ratio Test (RLRT) and a REML score test are discussed.
  - Simulation studies and applications.

Chapter 2

Bayesian Ising Graphical Model for Variable Selection

2.1 Introduction

Let's start from the standard multiple linear regression model $[\mathbf{y}|\boldsymbol{\beta}, \phi] \sim N(X\boldsymbol{\beta}, \phi^{-1}I)$, where $\mathbf{y}$ is an $n \times 1$ vector of the response variables, $X = (\mathbf{x}_1, ..., \mathbf{x}_p)$ is an $n \times p$ matrix of predictors, $\boldsymbol{\beta} = (\beta_1, ..., \beta_p)^T$ is a $p \times 1$ model coefficient vector of the full model with β_j corresponding to the jth predictor, $j = 1, ..., p$, and ϕ is the precision parameter. The inclusion or exclusion of the jth predictor in the model is represented by a binary indicator random variable γ_j, where $\gamma_j \in (0, 1)$. We denote the inclusion of predictor $\mathbf{x}_j$ with $\gamma_j = 1$, and otherwise we exclude it from the model. In recent years, incorporating prior network information of predictors into BVS models has received substantial attention (Li and Zhang, 2010; Monni and Li, 2010; Stingo et al., 2011; Tai et al., 2010). In all these papers, the network information of the predictors are introduced through an informative prior for γ_j's, which is a binary random graph. However, none of these papers discuss

treating variable selection as a graphical model with a noninformative prior for γ_j's. A binary random graphical model for the random vector $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_p)^T$ is represented by an undirected graph $G = (V, E)$, where V represents the set of p vertices or nodes corresponding to p predictors and E is a set of edges connecting neighboring nodes. In this dissertation, we base our approach on a reparameterized BVS model known as the KM model (Kuo and Mallick, 1998). We develop the new BVS approach via the graphical model and the Ising model, which is referred to "Bayesian Ising Graphical Model" (BIGM) for variable selection. We demonstrate that with the noninformative prior for $\boldsymbol{\gamma}$, the linear regression model is essentially a complete graphical model. A nice review about Ising model can be found in (Iba, 2001; Newman and Barkema, 1999).

Our contributions to this topic are in several aspects: (1) we connect BVS on linear regression model with the Ising model with random interaction terms and propose the BIGM, (2) we develop an efficient cluster algorithm, (3) we extend the BIGM to the Bayesian sparse additive model (BSAM) for nonparametric function components selection, (4) we study the selection probability profiles under different shrinkage and (5) connect the BIGM with tempering algorithm. In the following we explain in detail why they are important contributions by itemizing each contribution separately.

- First, by revealing that the binary Markov chain random process for $\boldsymbol{\gamma}$ on a graph can be modeled by the Ising model conditional on $\boldsymbol{\beta}$ and ϕ, we propose the BIGM. In a BIGM, the interactions between nodes are random and long-range (each node is the neighbour of any other nodes). To have flexible interactions between nodes, we adopt the "shrink globally act locally" strategy (Polson and Scott, 2011, 2012), which assigned scale normal mixture priors for the β_j's (Barndorff-Nielsen et al., 1982; West, 1987). To our best knowledge, our work is the first one to directly connect BVS with the binary graphical model via the Ising model.

- Second, we develop a generalized cluster algorithm in which the cluster is formed with the random interactions among nodes. Possible approaches to explore the configuration space of $\boldsymbol{\gamma}$ in an Ising model are the cluster algorithm and a family of exchange Monte Carlo, parallel tempering and simulated tempering algorithm (Iba, 2001). However, the current cluster algorithms such as the Swendsen-Wang algorithm (Swendsen and Wang, 1987) and Wolff algorithm (Wolff, 1989) are constructed based on the graph prior for $\boldsymbol{\gamma}$ and only consider fixed interactions. Therefore, both are not applicable to the more general random complete graphical model. Furthermore, in our BIGM, it is straightforward to combine the graphical prior information of $\boldsymbol{\gamma}$.

- Third, we extend our BIGM to the BSAM. There are only few papers discussing BVS under nonparametric regression (Reich et al., 2009; Scheipl, 2011; Smith and Kohn, 1996). Based on the KM model, our BIGM is easily extended to BSAM. We employ the Lancaster and Šalkauskas (LS) spline basis (Chib and Greenberg, 2010; Lancaster and Šalkauskas, 1986) to express the nonparametric function components. To our best knowledge, our method is the first to connect the graphical model with the nonparametric regressors such that we can simultaneously select an appropriate subset of the function components and estimate the flexible function curves.

- Fourth, we address the dynamics of the selection probability under different shrinkage. For BVS with large p, we define the curves of the selection probabilities, $p(\gamma_j = 1|\mathbf{y})$, of all predictors against the global shrinkage parameter as the profile curves of the BIGM. These profile curves are important because (1) they provide a good visualization for how to select the shrinkage parameter, and (2) they can also be used to assess the performance of different priors for $\boldsymbol{\beta}$. There is no such study on addressing the overall profile of selection probability under a wide range of shrinkage

parameter, while Lykou and Ntzoufras (2012) studied using only the Laplace prior. We address this issue by focusing on the orthogonal design. Interestingly, instead of priors with high weight on large shrinkage, our results indicate that the best prior places substantial weight on small, nonzero shrinkage.

- Fifth, to our best knowledge, there are no publications discussing the connection between the tempering algorithm and BVS. By casting the BVS problem as an Ising model, we demonstrate that the shrinkage parameter in BVS is equivalent to the temperature parameter in an Ising model. However, in the regular tempering algorithm, only one global temperature random variable is considered.

This chapter is organized as follows. In Section 2.2, we first introduce the KM hierarchical model and full conditional distributions for sampling all of the parameters except γ, and then we discuss the connection between BVS and the binary random graphical model. We finally express our model as the Ising model with noninformative prior for γ. In Section 2.3, we explain the single-site algorithm for sampling γ, then present a cluster algorithm. In Section 2.4 we introduce two extensions: one incorporates prior network information for γ, while the other applies the results to BSAM using the spline basis. In Section 2.5, we focus on understanding the selection probability profile, including the dynamics of the selection probability under different shrinkage priors. In Section 2.6 we also discuss the connection between the simulated tempering algorithm and priors of the scale mixture of normals. In Section 2.7, we provide the proofs for all lemmas and theorems. Then in Section 2.8 and 2.9, we illustrate our model with simulations and real data analysis. Finally, in the last section, we provide concluding remarks and discuss other potential extensions of our model.

2.2 Model Description

2.2.1 Bayesian Variable Selection with Normal Mixture Priors

We are interested in selecting a subset of predictors from the p potential candidates by exploring the configuration space of $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_p)^T$. To implement the stochastic search for γ_j's, stochastic search variable selection (SSVS) considers a multi-mode point mass and Gaussian mixture prior for β_j's, $[\beta_j|\gamma_j, \tau_\beta] \sim (1-\gamma_j)\delta(0) + \gamma_j N(0, \tau_\beta^{-1})$, where $\delta(0)$ represents the point mass density at zero. In this dissertation we consider the KM model, which is expressed as

$$\mathbf{y} = \sum_{j}^{p} \gamma_j \mathbf{x}_j \beta_j + \boldsymbol{\epsilon}, \tag{2.1}$$

where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \phi^{-1} I)$ is an iid noise vector. We standardize the data set X and center the response $\mathbf{y}$ such that $\sum_{i=1}^{n} x_{ij}^2 = 1, \sum_{i=1}^{n} x_{ij} = 0, j = 1, ...p$ and $\sum_{i=1}^{n} y_i = 0$. We may also include an intercept term μ in Model (2.1) with a normal prior, which requires only a simple extra step in the sampling procedure.

The application of the KM model is appropriate in this context for two reasons: first, it is more natural as a variable selection model, where $\gamma_j = 0$ indicates that jth predictor has no effect on the responses; and second, spike and slab models such as SSVS consider a multi-mode prior for the β_j's, which may present a mixing problem in that the sampling algorithm for the β_j's may get trapped in the point mass mode. This problem worsens when we extend the SSVS to nonparametric additive model (Scheipl, 2011), because the transition probability between the point mass and the normal mode for $\boldsymbol{\beta}_j$ decreases in higher dimensional space. Another reason can be demonstrated in the following section, where we show that it is natural to express a KM model under an Ising model, while this is not so straightforward in SSVS models.

The prior of $\boldsymbol{\beta}$ plays an important role in our model. In the following section 2.2.2, we will see that the interactions within the Ising model are determined by β_j's. The larger the coefficients of any two given predictors, the stronger the interaction will be between them. This corresponds the nodes having a high probability of dependence, meaning that they are either "aligned" (both γ_j's are likely to be equal) or "anti-aligned" (the γ_j's are likely to be unequal). On the other hand, if the magnitude of the coefficients for two nodes is small, then the inclusion or exclusion of the predictors is independently determined. We seek a method that allows the β_j's to be as flexible as possible in order to explore the configuration space so the variation of each β_j's is modeled by an independent precision parameter. Meanwhile, we also seek to place a constrain on the overall variability of the interaction through a global parameter, which we refer to b. Therefore, we follow the "shrink globally act locally" scheme suggested by Polson and Scott (2011).

The "shrink globally act locally" scheme is easily implemented by imposing a hierarchical model:

$$\begin{aligned} [\beta_j|\tau_j, b] &\sim N(0, b^2\tau_j^{-1}), \\ [\tau_j] &\sim p(\tau_j), \\ [\phi] &\sim p(\phi), \end{aligned} \tag{2.2}$$

where τ_j is the precision parameter for the conditional normal prior of β_j and plays the role of local tempering, and b is the global shrinkage parameter to place a constrain on all τ_j's. $p(\tau_j)$ and $p(\phi)$ are the priors for τ_j's and ϕ respectively. Heaton and Scott (2010) have also discussed similar hierarchical model under SSVS. With these settings, we can easily

achieve the full conditional distribution for $\boldsymbol{\beta}_c \subseteq \boldsymbol{\beta}$

$$[\boldsymbol{\beta}_c|\mathbf{y}, \boldsymbol{\gamma}, \boldsymbol{\beta}_{\bar{c}}, \phi] \sim \begin{cases} N(\boldsymbol{\mu}_c, \Sigma_c) & \text{if } \boldsymbol{\gamma}_c = \mathbf{1} \\ N(\mathbf{0}, D_c^{-1}) & \text{if } \boldsymbol{\gamma}_c = \mathbf{0}. \end{cases} \tag{2.3}$$

Here we use "c" as a general subscript to stand for a subset of the index $\{1, ..., p\}$. We use $\bar{c}$ to present the complementary index set of c. In above expression, D_c is a $|c| \times |c|$ diagonal matrix with $\tau_j/b^2, j \in c$ as the diagonal elements, where $|c|$ refers for the cardinality of c. Σ_c and $\boldsymbol{\mu}_c$ are expressed as

$$\begin{aligned} \Sigma_c &= \left(\phi X_c^T X_c + D_c\right)^{-1}, \\ \boldsymbol{\mu}_c &= \phi \Sigma_c X_c^T \left(\mathbf{y} - X_{\boldsymbol{\gamma}_{\bar{c}}} \boldsymbol{\beta}_{\bar{c}}\right). \end{aligned} \tag{2.4}$$

With some abuse of notations here, X_c stands for the sub-matrix of X corresponding the predictors in c. Similarly $X_{\boldsymbol{\gamma}_{\bar{c}}}$ is the sub-matrix of $X_{\boldsymbol{\gamma}} = (\gamma_1 \mathbf{x}_1, ..., \gamma_p \mathbf{x}_p)$ corresponding to predictors indexed by elements of $\bar{c}$.

We simply assign a noninformative prior for $\phi : [\phi] \sim \phi^{-1}$, so that the full conditional distribution of ϕ becomes a gamma distribution,

$$[\phi|\mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\gamma}] \sim G\left(\frac{n}{2}, \frac{1}{2}\|\mathbf{y} - X_{\boldsymbol{\gamma}}\boldsymbol{\beta}\|^2\right). \tag{2.5}$$

Prior selection for the τ_j's is critical since it determines how the local behavior of the sampling process. There are several options for $p(\tau_j)$. In this dissertation we consider three widely known $p(\tau_j)$'s that result in three typical marginal β_j's priors with heavy tails and/or heavy mass near zero. These marginal priors of the β_j are the Cauchy, Laplace and horseshoe priors, which are achieved by assigning a Gamma prior $[\tau_j] \sim G(1/2, 1/2)$,

inverse gamma prior $[\tau_j] \sim IG(1, 1/2)$ and half Cauchy prior $[\tau_j^{1/2}] \sim C^+(0, 1)$ to τ_j respectively. The forms of the densities for these three normal mixture settings are listed in Table 2.1. Notice that we use the terms "Cauchy", "Laplace" and "horseshoe" not only to refer to the marginal priors of β_j's, but also to represent the normal mixture settings. For example, in our context, "Cauchy prior" stands for normal/gamma setting such that the marginal prior of β_j is Cauchy and the prior for $p(\tau_j)$ is $G(1/2, 1/2)$.

Table 2.1: Summary of the Cauchy, Laplace and horseshoe priors for the marginal prior of β_j's, corresponding to priors for $p(\tau_j)$ and the density functions of the shrinkage parameter κ_j.

Marginal prior	$p(\beta_j \mid b)$	Prior for τ_j	Distribution for κ_j
Cauchy	$\pi b(\beta_j^2 + b^2)^{-1}$	$\tau_j^{-\frac{1}{2}} \exp(-\frac{\tau_j}{2})$	$\kappa_j^{-\frac{1}{2}}(1-\kappa_j)^{-\frac{3}{2}} \exp(-\frac{b^2\kappa_j}{2(1-\kappa_j)})$
Laplace	$(2b)^{-1} \exp(-\lvert\beta_j\rvert/b)$	$\tau_j^{-2} \exp(-\frac{2}{\tau_j})$	$\kappa_j^{-2} \exp(-\frac{(1-\kappa_j)}{2b^2\kappa_j})$
Horseshoe	-	$\tau_j^{-\frac{1}{2}}(1+\tau_j)^{-1}$	$\kappa_j^{-\frac{1}{2}}(1-\kappa_j)^{-\frac{1}{2}}[1-\kappa_j+b^2\kappa_j]^{-1}$

By defining the parameter $b^* = b/\sqrt{\phi}$, the full conditional distribution for τ_j in the Cauchy and Laplace priors are

$$[\tau_j \mid \beta_j, b, \phi] \propto \tau_j^{-3/2} \exp\left[-\frac{(\tau_j - b^*/|\beta_j|)^2}{2\tau_j b^{*2}/\beta_j^2}\right], \qquad \text{Laplace prior} \tag{2.6}$$

$$[\tau_j \mid \beta_j, b, \phi] \propto \exp\left[-\frac{1}{2}(\beta_j^2/b^{*2} + 1)\tau_j\right], \qquad \text{Cauchy prior} \tag{2.7}$$

and the full condition distribution for τ_j in the horseshoe prior is obtained by

$$\begin{aligned} [u_j \mid \beta_j, b, \phi, v_j] &\propto \exp\left[-\frac{1}{2}\left(\frac{\beta_j^2}{b^{*2}v_j} + 1\right)u_j\right], \\ [v_j \mid \beta_j, b, \phi, u_j] &\propto v_j^{-1} \exp\left[-\frac{1}{2}\left(\frac{\beta_j^2 u_j}{b^{*2}}\frac{1}{v_j} + v_j\right)\right], \qquad \text{Horseshoe prior} \\ \tau_j &= u_j/v_j. \end{aligned} \tag{2.8}$$

(2.6) is an inverse Gaussian distribution $ING(b^*/|\beta_j|, 1)$ with mean $b^*/|\beta_j|$ and shape pa-

rameter 1. (2.7) is an exponential distribution or Gamma distribution $G\left(1, (\beta_j^2/b^{*2}+1)/2\right)$. The Gibbs sampler for the horseshoe prior is implemented by using the redundant multiplicative reparameterization technique similar to that found in Gelman (2006). If we reparameterize τ_j as $\tau_j = u_j/v_j$ where u_j and v_j are independently identically distributed with prior $G(1/2, 1/2)$, then a priori $[\tau_j^{1/2}] \sim C^+(0,1)$, and the prior for β_j is the horseshoe prior. In (2.8), the full conditional distributions for u_j and v_j are a Gamma distribution $G\left[1, (\beta_j^2/(b^{*2}v_j)+1)/2\right]$ and the generalized inverse Gaussian distribution $GING\left(0, \frac{\beta_j^2 u_j}{b^{*2}}, 1\right)$, respectively.

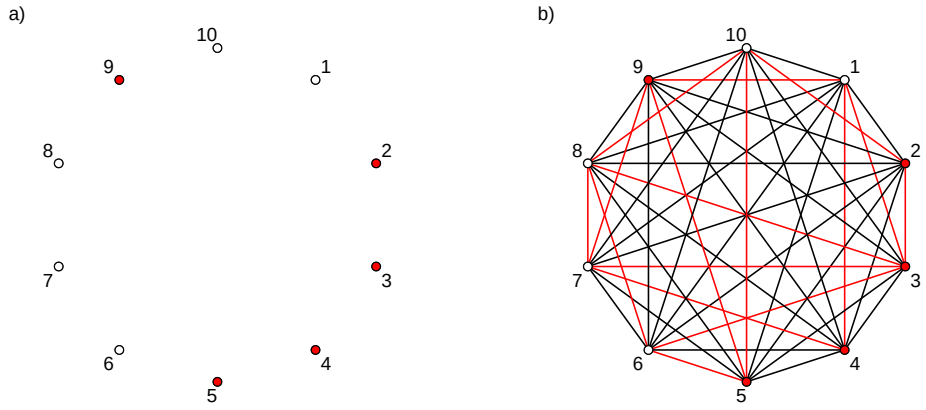

Figure 2.1: Diagram of variable selection as a random graph model with selected nodes (filled circles), excluded nodes (circles), edges of positive interaction (black lines), and edges of negative interaction (red lines). Independent variable selection: no interactions among nodes (a). General variable selection: a complete graph (b).

2.2.2 Bayesian Ising Graphical Model

The noninformative prior for $\boldsymbol{\gamma}$ is $[\boldsymbol{\gamma}] \sim \left(\frac{1}{2}\right)^p$. Thus the full conditional distribution of $\boldsymbol{\gamma}|\boldsymbol{\beta}, \phi$ is directly derived from the likelihood of $\boldsymbol{\gamma}$ given $\boldsymbol{\beta}$ and ϕ. Given $\boldsymbol{\beta}$, consider the

matrix of marginal regression functions $R = (\mathbf{r}_1, ..., \mathbf{r}_p) = (\beta_1 \mathbf{x}_1, ..., \beta_p \mathbf{x}_p)$, with each column as the marginal regression vector for jth predictor vector. In additive nonparametric model (see Section 2.4.2), $\mathbf{r}_j = f_j(\mathbf{x}_j) = Z_j \boldsymbol{\beta}_j$ is the nonparametric function component of $\mathbf{x}_j$ expanding on the $n \times M_j$ basis matrix Z_j with $1 \times M_j$ coefficient vector $\boldsymbol{\beta}_j$ (M_j is the dimension of the basis). Here we first consider a parametric regression model, thus the full conditional distribution of $\boldsymbol{\gamma}$ is

$$\begin{aligned} p(\boldsymbol{\gamma}|\mathbf{y}, \boldsymbol{\beta}, \phi) &\propto p(\mathbf{y}|\boldsymbol{\gamma}, \boldsymbol{\beta}, \phi) \\ &\propto \exp\left(-\frac{1}{2}\phi \boldsymbol{\gamma}^T R^T R \boldsymbol{\gamma} + \phi \mathbf{y}^T R \boldsymbol{\gamma}\right). \end{aligned} \tag{2.9}$$

This is the Boltzman distribution of the Ising model, $\frac{1}{Z}\exp(-U(\boldsymbol{\gamma}))$, with

$$\begin{aligned} U(\boldsymbol{\gamma}) &= -\boldsymbol{\gamma}^T J \boldsymbol{\gamma} - \mathbf{h}^T \boldsymbol{\gamma}, \\ J &= -\frac{\phi R^T R}{2}, \\ \mathbf{h} &= \phi R^T \mathbf{y}, \end{aligned} \tag{2.10}$$

where $Z = \sum_{\boldsymbol{\gamma}} \exp(-U(\boldsymbol{\gamma}))$ is called the partition (normalized) function and $U(\boldsymbol{\gamma})$ is called the "energy" of state $\boldsymbol{\gamma}$ given $\boldsymbol{\beta}$ and ϕ, J is the interaction matrix, and $\mathbf{h}$ is called an "external field". Expression (2.9) is equivalent to the following model:

$$p(\boldsymbol{\gamma}|\mathbf{y}, \boldsymbol{\beta}, \phi) \propto \exp\left(\sum_{i<j} J_{ij}\delta_{ij} + \sum_j h_j^* \gamma_j\right), \tag{2.11}$$

where the first summation is on all $i < j, j = 1, ..., p$, $\delta_{ij} = 1$ if $\gamma_i = \gamma_j$ otherwise $\delta_{ij} = 0$, $J_{ij} = \phi\beta_i(\mathbf{x}_i^T \mathbf{x}_j)\beta_j$ is the non diagonal element of matrix J and h_j^* is the jth element of vector $\mathbf{h}^* = \phi R^T(\mathbf{y} - R\mathbf{1}/2)$. Expression (2.11) is achieved by plugging in following

transformation into (2.9)

$$2\left[\frac{1}{4}+\left(\gamma_i-\frac{1}{2}\right)\left(\gamma_j-\frac{1}{2}\right)\right]=\delta_{ij}=\begin{cases}1 & \gamma_i=\gamma_j\\ 0 & \gamma_i\neq\gamma_j.\end{cases} \tag{2.12}$$

The model with $\boldsymbol{\gamma}$ distributed as (2.9) is called an "spin glass" model (consider γ_j as having two spin states, up and down, corresponding to 1 and 0 respectively) when the coupling parameter J_{ij} follows some random distribution with positive or negative values. However, in our BIGM model, because J_{ij} is the product of random variables, β_j's and ϕ each has a prior, the distribution for J_{ij} is some unknown distribution which is usually neither iid nor tractable. Therefore, a numerical method, such as an MCMC sampling procedure, to simulate the distribution of $\boldsymbol{\gamma}$ is required. Now we can see that the choice of the prior for β_j's is important, as it dictates the interaction among the nodes. The independent scale normal mixture prior for β_j's is a possible choice since it is similar to the known tempering algorithm in the Ising model, and we can also derive a cluster algorithm based on it. Both algorithms are expected to improve the mixing issue of the sampler (Nott and Green, 2004; Swendsen and Wang, 1987; Wolff, 1989).

Based on the Ising model, and considering the p predictors as a set of nodes, we assign a binary random variable γ_j for each nodes. Those nodes may interact or couple with each other as previously described, so we have the following proposition:

Proposition 2.2.1. *The p-dimensional binary random variable $\boldsymbol{\gamma}$ of the BVS problem based on the KM model (2.1) is a class of stochastic processes on a finite random undirected graph model $G=(V,E)$, where $V=\{1,...,p\}$ is the set of nodes, corresponding to p predictors, and $E\subset V\times V$ is the set of edges. $\boldsymbol{\gamma}\in\Gamma=\{(\gamma_1,...,\gamma_p):\gamma_j\in(0,1),j=1,...,p\}$ is indexed by V with probability measure on Γ as (2.11), in which J_{ij}'s and h_j^*'s are all random with some distributions determined by the priori distributions of β_j's and ϕ.*

This is a complete graphical model. Since we are free to include an interaction between any two nodes of V, the interactions among the nodes are long-range. Figure 2.1 shows the diagram of the graphical model for BVS. In Figure 2.1 (a), the interaction between any nodes $J_{ij} = 0$ thus implying that this is a completely independent setting with which the configuration of $\boldsymbol{\gamma}$ depends on the "external field" $\mathbf{h}$ only. Figure 2.1 (b) is a more general diagram for the complete graphical model. However, since any possible J is allowed, for a given J, a specific configuration of the edges will be given. For example, for a one-dimension Ising model, the nodes form a one-dimension chain, and one node only interacts with its two nearest neighbor nodes. This means the matrix J is a sparse matrix with nonzero elements in positions $|i - j| \leq 1$ only.

2.3 Algorithm for Updating $\boldsymbol{\gamma}$

2.3.1 Single-site Algorithm

Because we have the full conditional distributions under the Ising model (2.9) and (2.11), we are able to directly apply the Gibbs sampler for $\boldsymbol{\gamma}$. By assigning a noninformative prior for $\boldsymbol{\gamma}$, $[\boldsymbol{\gamma}] \sim \left(\frac{1}{2}\right)^p$, the full conditional distribution given the data for single-site updating becomes

$$\begin{aligned} [\gamma_j|\mathbf{y}, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\beta}, \phi] &\sim Ber\left(\frac{1}{1+\pi}\right), \\ \pi &= \exp\left\{-[U(\gamma_j = 1|\boldsymbol{\gamma}_{\bar{j}}) - U(\gamma_j = 0|\boldsymbol{\gamma}_{\bar{j}})]\right\} \\ &= \exp(-J_{jj} - 2J_{j\bar{j}}\boldsymbol{\gamma}_{\bar{j}} - h_j), \end{aligned} \tag{2.13}$$

where Ber stands for the Bernoulli distribution and $J_{j\bar{j}}$ is the jth row of J with jth column removed. $U(\gamma_j = 1|\boldsymbol{\gamma}_{\bar{j}}) - U(\gamma_j = 0|\boldsymbol{\gamma}_{\bar{j}})$ is the "energy" difference of two state configura-

tions, $\gamma_j = 0|\boldsymbol{\gamma}_{\bar{j}}$ and $\gamma_j = 1|\boldsymbol{\gamma}_{\bar{j}}$, where $\boldsymbol{\gamma}_{\bar{j}}$ is the vector of $\boldsymbol{\gamma}$ with γ_j removed. Therefore the complete full conditional distributions of Gibbs sampler for updating $\boldsymbol{\gamma}, \boldsymbol{\beta}$ and ϕ involves Expression (2.3), (2.4), (2.5), one of (2.6-2.8) and (2.13). This procedure is simple and works well for most cases with moderate size p.

The main advantage of the above procedure is there is only one tuning parameter b, and the "tuning" process is extremely simple: just choose a b that can separate the selected predictors from the rest to the max according to the selection probability, $p(\gamma_j = 1|\mathbf{y})$.

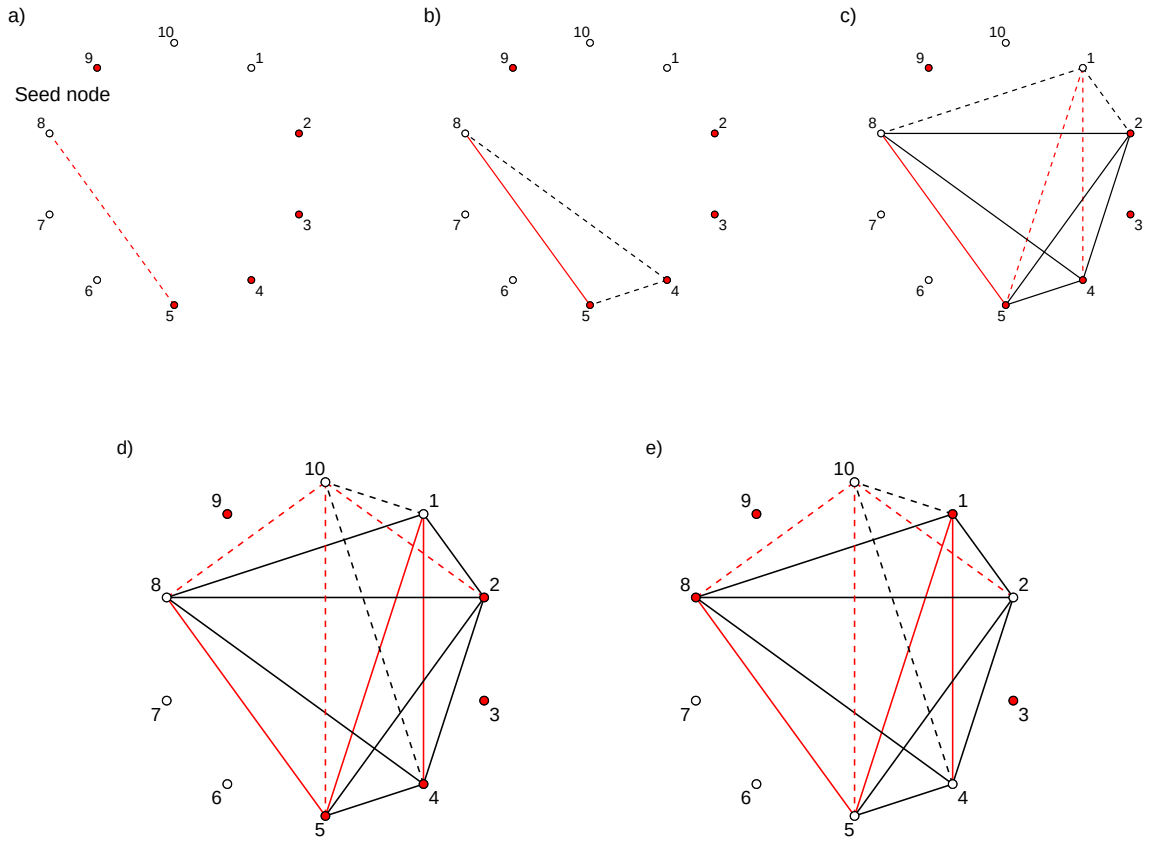

Figure 2.2: Diagram of the cluster algorithm. Forming the cluster (a-c). Flipping clustered nodes (d-e).

We can also consider a more general Matropolis-Hastings (MH) one-step updating procedure. Denote the current state for γ_j as $\gamma_j^0|\boldsymbol{\gamma}_{\bar{j}}$ and its flipped state $\gamma_j^*|\boldsymbol{\gamma}_{\bar{j}}$, whether or not we move from $\gamma_j^0|\boldsymbol{\gamma}_{\bar{j}}$ to $\gamma_j^*|\boldsymbol{\gamma}_{\bar{j}}$ depends on the "energy" difference $\Delta U = U(\gamma_j^*|\boldsymbol{\gamma}_{\bar{j}}) - U(\gamma_j^0|\boldsymbol{\gamma}_{\bar{j}})$. We prefer the system in lower "energy" state since the lower the energy, the higher the probability. Thus if $\Delta U \leq 0$, the flipped state is accepted with probability 1. We treat the case $\Delta U > 0$ probabilistically, that is, with the probability of acceptance as $p(\Delta U) = \exp(-\Delta U)$. These steps can be summarized by flipping the current state to its opposite with probability

$$\min\left(1, \exp(-\Delta U)\right). \tag{2.14}$$

The detailed balance maintains and this MH updating is used in the MCMC Ising model sampling (Newman and Barkema, 1999; Nott and Green, 2004). In this dissertation, unless otherwise specified, we adopt the one step MH updating (2.14) with other Gibbs samplers in all cases. This algorithm is the antithetic updating method discussed by Nott and Green (2004) since $\exp(-\Delta U)$ is the odds of flipping current state with the Gibbs-type proposal.

2.3.2 Cluster Algorithm

Beyond the single-site algorithm, the cluster algorithm is well established for simulating model (2.9) when J_{ij}'s and h_j's are fixed. In general, the cluster algorithm performs better than the single-site updating when J_{ij} is fixed. However, as pointed out before, it is difficult to apply the cluster algorithm to the model (2.9) as there is a random external field $\mathbf{h}^*$ and the coupling coefficients J_{ij}'s follow some unknown distribution with a non-negligible dependence structure. Additionally, the nodes are connected with each other by so called long-range interactions, and thus the system is a totally disordered complete graph. In this dissertation, we propose a generalized single-cluster Monte Carlo algorith-

m, which is similar to Wolff's clustering scheme but is also capable of handling long-range interactions and a random external field.

In the original SW and Wolff algorithm, clusters are formed through the bonding between paired nodes on a lattice with positive interactions. Unlike the usual Ising model on a one-dimension chain or two/three dimension lattice, the complete graph model of the binary random process is fully connected. This indicates that the behavior of each node is determined according to the overall effects of all other nodes. The clustering dynamics must therefore incorporate this consideration. The growth of a cluster (adding one new node to the existing cluster) should consider the coupling between the new node and all nodes in the cluster.

Before we introduce the cluster algorithm, we specify two types of clusters since the cluster is formed according to the coupling coefficient J_{ij} which can be either positive or negative; (1) one is a cluster with nodes aligned and (2) the other is a cluster with nodes aligned and anti-aligned.

We use c to denote the cluster, and $\bar{c}$ as the complement of c. The single node is considered to be a special case of the cluster with aligned nodes. Therefore, within the second type of cluster there are two sub clusters that are anti-aligned. We denote these two sub clusters as c_1 and c_0 with $\boldsymbol{\gamma}_{c_1} = \mathbf{1}$ and $\boldsymbol{\gamma}_{c_0} = \mathbf{0}$ respectively.

The question, then, is: given a particularly defined probability p_a of adding a node to the cluster, what is the acceptance ratio that makes the flip of the cluster satisfy detailed balance. Also, how does one choose p_a such that the average acceptance ratio is as large as possible? We derive following generalized Wolff algorithm based on these considerations.

1. Form the cluster.

   (a) Initialize the cluster set c by randomly picking a seed node.

(b) Examine the nodes in $\bar{c}$ one by one, add the node j in $\bar{c}$ to the cluster with the probability

$$p_{a,j} = \max\left\{1 - \exp\left[\lambda(-1)^{\gamma_j}\left(\sum_{k\in c_1} J_{jk} - \sum_{l\in c_0} J_{jl}\right)\right], 0\right\}, \tag{2.15}$$

and remove j from $\bar{c}$ if j added to c, where $0 \leq \lambda \leq 1$. Continue iteratively until no new sites added when each nodes in $\bar{c}$ has been examined.

2. Flip the nodes in cluster c with probability

$$\begin{aligned}
&\alpha(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*) \\
&= \min\left\{\exp\left[(1-\lambda)\sum_{j\in\bar{c}}(-1)^{\gamma_j}\left(\sum_{k\in c_1} J_{jk} - \sum_{l\in c_0} J_{jl}\right) + \sum_{j\in c_0} h_j^* - \sum_{j\in c_1} h_j^*\right], 1\right\} \\
&= \min\left\{\exp\left[(1-\lambda)(\mathbf{1}^T J_{c_0\bar{c}} - \mathbf{1}^T J_{c_1\bar{c}})(2\boldsymbol{\gamma}_{\bar{c}} - \mathbf{1}) + \mathbf{1}^T \mathbf{h}_{c_0}^* - \mathbf{1}^T \mathbf{h}_{c_1}^*\right], 1\right\}.
\end{aligned} \tag{2.16}$$

3. Flip the rest nodes in $\bar{c}$ (if any are left) by the single-site updating method (2.14).

4. Update β_j's, τ_j's and ϕ.

In (2.16), the last expression is for coding convenience, and therefore uses matrix expressions. As we can see, λ plays a role of partial clustering similar to Higdon (1998). When $\lambda = 1$, all interaction terms in (2.16) are annihilated, which means the cluster is completely decoupled from its neighbors. If $\lambda = 0$, then no clustering occurs, and the algorithm is reduced to single-site algorithm.

The cluster algorithm can be better explained using the diagram in Figure 2.2. Figure 2.2 (a-c) demonstrate the clustering process. First we randomly select a seed node (in this diagram, we use node 8). Then we throw the bond to all neighbors of node 8, and find that node 5 is bonded to 8 with probability $p_{a,5}$, forming the cluster (the dashed line

is turned into solid lines, which indicates that 5 is added to the cluster). We scan the remaining nodes again, but whether or not a new node should be added is determined by the bonding between the new node and node 5 and 8. For example in Figure 2.2 (b), the bond between the new node 4 and the cluster is composed of the bonds 4-5 and 4-8. In Figure 2.2 (c), after adding the last new node 1 to the cluster, we scan all the remaining nodes and find no new node added, which then terminates the clustering process.

Flipping of the cluster is demonstrated in Figure 2.2 (d-e). The cluster formed contains nodes $c = \{8, 5, 4, 2, 1\}$. To flip these nodes, we have to cut off the bonding of the cluster with all other nodes in $\bar{c}$ because in a complete graph the neighbors of a cluster are all the other nodes outside of the cluster. For example, the bond between the cluster and node 10 is demonstrated in Figure 2.2 (d), where we can see the bonds between 10 and all nodes in the cluster should be cut to flip the cluster. Thus to completely flip the cluster, the bonds between all other nodes in $\bar{c}$ and the nodes in c should be cut. Similarly, in the reverse process to flip the cluster to its original configuration, as shown in Figure 2.2 (e), all the bonds between cluster c and the nodes in $\bar{c}$ must be cut.

It is easy to show that our algorithm is more general then Wolff's in sense that it is applicable to complete graphs with random interaction. When applied to the Ising model on a lattice with positive fixed interaction J, where only interactions among the nearest neighbors count, our algorithm evaluates to the original Wolff algorithm: the cluster grows by adding bonds between nearest neighbors with probability $1 - \exp(-J)$.

Theorem 2.3.1. *With the probability of adding node to the cluster, $p_{a,j}$, and the probability of moving from the current configuration $\boldsymbol{\gamma}_c^0$ to the flipped configuration $\boldsymbol{\gamma}_c^*$, $\alpha(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)$, as defined as in the generalized Wolff algorithm, the algorithm is detail balanced and ergodic.*

Proof: See Section 2.7.1.

In this dissertation, we mainly focus on the noninformative prior for $\boldsymbol{\gamma}$. However, since the distribution of $\boldsymbol{\gamma}$ given $\boldsymbol{\beta}$ and ϕ is the Boltzman distribution, it is nature to assign a Boltzman prior or Ising prior for $\boldsymbol{\gamma}$ if such priori information is available. Another advantage of the cluster algorithm is it reveals the latent graph structure according to the frequencies of nodes that form a cluster, and this information may help us to distinguish the signals from the noise since they are likely to be anti-aligned.

2.4 Extensions

2.4.1 Incorporating Graph Prior Information

In this dissertation, we mainly discuss Model (2.1) as a graphical model with a noninformative prior for $\boldsymbol{\gamma}$. This approach works well for the case where n is large. However, the priori information about $\boldsymbol{\gamma}$ becomes important when n goes small. There are two purposes of incorporating graph prior information for $\boldsymbol{\gamma}$. First of all, doing so helps to improve mixing so that the model works for $n << p$. Second, it improves the power for detection of true signals. Since two connected nodes with positive interaction tend to be selected or excluded together, only the prior graph for $\boldsymbol{\gamma}$ with positive interaction is meaningful. If we have the information that some selected nodes and their neighbors are all true nodes, then incorporating a graph prior with those nodes connected will improve the power to identify the nodes with small signal. This is because the prior tells us that nodes with small signal have a higher chances of being selected with their neighbors, which give true signal. On the other hand, for nodes that are known to be false signal, we have a higher chance of excluding their neighbors, since the prior tells us they should be excluded together.

Even though the prior information is not exactly correct, it will help if the prior graph contains the true graph about which nodes are networked. Take, for example, a given true model, $\mathbf{y} = \sum_{j \in S} \mathbf{x}_j \beta_j$ where $V^* = 1, ..., k$ is sequential index up to k and $k < p$. Obviously there are some information about the true variables such that there are k sequential nodes that are true nodes, and $p - k$ sequential nodes which do not belong to the true model. Therefore, an Ising prior with a one dimensional linear chain will be a very efficient prior since this prior reflects the information that sequential nodes are selected or excluded together.

Another example is the genetic pathway data within which different sets of genes function together. Some gene sets are related to phenotype diseases, some are not. Therefore the prior with this pathway graph helps to distinguish different sets of genes in the pathway since among those genes if one node is selected then its connected neighbors have a high chance of being selected as well. Further example about incorporating prior graph information can be seen in Li and Zhang (2010); Monni and Li (2010); Stingo et al. (2011); Tai et al. (2010).

Since we are only interested in the network prior information, we only apply a graph prior for $\boldsymbol{\gamma}$ with the interaction matrix $W = \{W_{ij}\}$ without the external field:

$$p(\boldsymbol{\gamma}) \propto \exp\left(\sum_{i<j} W_{ij}\delta_{ij}\right),$$

where W_{ij} represents the prior coupling information between node i and j. According to W_{ij}, we can also define the adjacency matrix, $\Lambda = \{\lambda_{ij}\}$, with $\lambda_{ij} = 1$ if $W_{ij} \neq 0$ otherwise

$\lambda_{ij} = 0$ for $i \neq j$. With this prior, the posterior distribution for $\boldsymbol{\gamma}$ is modified as:

$$\begin{aligned} p(\boldsymbol{\gamma}|\mathbf{y}, \boldsymbol{\beta}, \phi) &\propto p(\mathbf{y}|\boldsymbol{\gamma}, \boldsymbol{\beta}, \phi)p(\boldsymbol{\gamma}) \\ &\propto \exp\left(\sum_{i<j} J^*_{ij}\delta_{ij} + \sum_j h^*_j \gamma_j\right), \end{aligned} \tag{2.17}$$

where $J^*_{ij} = (J_{ij} + W_{ij})$. J_{ij} and h^*_j are defined in (2.10) and (2.11).

Correspondingly, the two expressions for the cluster algorithm are modified as

$$p_{a,j} = \max\left\{1 - \exp\left[(-1)^{\gamma_j}\left(\sum_{k\in c_1}\lambda_{jk}J^*_{jk} - \sum_{l\in c_0}\lambda_{jl}J^*_{jl}\right)\right], 0\right\}, \tag{2.18}$$

$$\begin{aligned} &\alpha(\boldsymbol{\gamma}^0_c \to \boldsymbol{\gamma}^*_c) \\ &= \min\left\{\exp\left[\sum_{j\in\bar{c}}(-1)^{\gamma_j}\left(\sum_{k\in c_1}(1-\lambda_{jk})J^*_{jk} - \sum_{l\in c_0}(1-\lambda_{jl})J^*_{jl}\right) + \sum_{j\in c_0}h^*_j - \sum_{j\in c_1}h^*_j\right], 1\right\}. \end{aligned} \tag{2.19}$$

The above two expressions tell us that $p_{a,j}$ and $\alpha(\boldsymbol{\gamma}^0_c \to \boldsymbol{\gamma}^*_c)$ are also conditional on Λ.

2.4.2 Extension to Nonparametric Regression Models: Bayesian Sparse Additive Model (BSAM)

Although BIGM is based on the parametric linear regression model (2.1), it is easily extended to nonparametric regression models. Some similar approaches have been suggested, such as nonparametric regression using BVS (Smith and Kohn, 1996) and Bayesian Smoothing Spline ANOVA models (Reich et al., 2009), both of which use the spline tech-

niques.

Extending the multiple parametric linear regression model (2.1) to an additive model is straightforward. From a Bayesian point of view, there is no strict difference between parametric and nonparametric additive regression model in the sense that both assign priors to the basis coefficients. In general, both choose a basis to express the marginal regression predictor $f_j(\mathbf{x}_j)$. For linear parametric regression, $f_j(\mathbf{x}_j) = \beta_j \mathbf{x}_j$, where the predictor $\mathbf{x}_j$ itself can be considered as the basis to represent f_j, and β_j is a univariate random variable. This is a special case of nonparametric regression model considering $f_j(\mathbf{x}_j) = Z_j \boldsymbol{\beta}_j$, where Z_j is some basis matrix for the jth predictor and $\boldsymbol{\beta}_j$ is multivariate random variable, and the basis length $M_j \geq 1$ can vary for different predictor. Despite the variation of the basis chosen, each predictor corresponds to a univariate random vector $\mathbf{r}_j = f(\mathbf{x}_j) = Z_j \boldsymbol{\beta}_j$. Then the generalized additive model can be expressed as

$$\mathbf{y} = \mu + \sum_{j=1}^{p} \gamma_j f_j(\mathbf{x}_j) + \boldsymbol{\epsilon}.$$

For this model, similarly, we can consider following prior for $\boldsymbol{\beta}_j$'s

$$\begin{aligned} [\boldsymbol{\beta}_j | \tau_j] &\sim N(\mathbf{0}, b^2 \tau_j^{-1} I), \\ [\tau_j] &\sim p(\tau_j). \end{aligned} \tag{2.20}$$

where $N(\cdot)$ is multivariate M_j-dimensional normal distribution, and $p(\tau_j)$ is a prior similar to those in the previous discussions, such as $G(1/2, 1/2), IG(1, 1/2)$ or $C^+(0, 1)$. For some special bases, such as LS basis, the multivariate normal prior of $\boldsymbol{\beta}_j$ may have two variance components (see Appendix A). Note that because the dimension of $\boldsymbol{\beta}_j$ is changed, if we integrate out τ_j by assigning the same $p(\tau_j)$ as in parametric linear models, the marginal prior $p(\boldsymbol{\beta}_j | b)$ is no longer Cauchy, Laplace, or horseshoe, but rather it shares

similar properties with the linear parametric case.

Similarly we can define matrix $R = [\mathbf{r}_1, ..., \mathbf{r}_p]$, the design matrices $Z = [Z_1, ..., Z_p]$, $Z_{\boldsymbol{\gamma}} = [\gamma_1 Z_1, ..., \gamma_p Z_p]$ and the coefficients vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^T, ..., \boldsymbol{\beta}_p^T)^T$, but here we should treat $\boldsymbol{\beta}$ and Z as blocks. The total dimension for the design matrix Z is $n \times M$ and $M \times 1$ for $\boldsymbol{\beta}$, where $M = \sum_j M_j$. Without any further complications, we can use the same posterior distribution expressions in (2.3) and (2.5) to update $\boldsymbol{\beta}_c$ and ϕ, with the exception that $\boldsymbol{\beta}$ and Z are in blocks and D_c is a diagonal block matrix with block $\tau_j/b^2 I_{M_j}, j \in c$ along the diagonal, where I_{M_j} is the M_j-dimensional identity matrix. Σ_c and $\boldsymbol{\mu}_c$ are expressed as

$$\begin{aligned} \Sigma_c &= \left(\phi Z_c^T Z_c + D_c\right)^{-1}, \\ \boldsymbol{\mu}_c &= \phi \Sigma_c Z_c^T \left(\mathbf{y} - Z_{\boldsymbol{\gamma}_{\bar{c}}} \boldsymbol{\beta}_{\bar{c}}\right). \end{aligned} \tag{2.21}$$

The calculation for J_{ij}'s and h_j's is exactly the same as in (2.10), since those formulas involve R, a $n \times p$ matrix in both cases. In Appendix A we will introduce a specific additive model with the natural cubic spline represented by LS basis. Of course, other spline bases to define Z are possible.

2.5 Understanding the Mechanism of Bayesian Variable Selection

The purpose of this section is to understand the influence of the marginal prior for $\boldsymbol{\beta}$, given the tuning parameter b, on the marginal probability $p(\gamma_j|\mathbf{y})$. Although our Ising model is based on the KM model (2.1), the results of this section are also applicable to the SSVS model with the point mass mixture prior for $\boldsymbol{\beta}$. This is because if we integrate out $\boldsymbol{\beta}$, both models become identical. Note that the results in this section apply to the

parametric linear model (2.1) where β_j is scalar, but the main conclusions are similar to nonparametric linear model where $\boldsymbol{\beta}_j$ is $M_j \times 1$ vector.

Some extra notations are introduced here. Since $\mathbf{x}_j$'s are standardized, $C = X^T X = [\mathbf{c}_1, ..., \mathbf{c}_p]$ is the correlation matrix of $\mathbf{x}_j$'s and $\mathbf{c}_j$'s stands for the vector of the correlation between $\mathbf{x}_j$ with all predictors. For the orthogonal data set, $C = I_n$ or $\mathbf{x}_i^T \mathbf{x}_j = \delta_{ij}$ and $\mathbf{x}_j^T \boldsymbol{\epsilon} = 0, i, j = 1, ..., p$. The projection of $\mathbf{y}$ on $\mathbf{x}_j$ can be expressed as $\mathbf{a} = X^T \mathbf{y} = (\mathbf{x}_1^T \mathbf{y}, ..., \mathbf{x}_p^T \mathbf{y})^T = (a_1, ..., a_p)^T$, which are the estimates of the signals of the β_j's under an orthogonal design. We may also need the notation $\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}} = \boldsymbol{\gamma}_{\bar{j}} \circ \{\mathbf{c}_j\}_{\bar{j}}$, where "$\circ$" stands for the pointwise product of two vectors, "$\bar{j}$" stands for the jth element removed for corresponding vectors and matrices,

2.5.1 General Profile of the Marginal Selection Probability

With the hierarchical model defined as (2.1) and (2.2) in Section 2.2.1, the posterior distribution of $\boldsymbol{\beta}$ is multivariate normal given $\boldsymbol{\gamma}$ with mean $\boldsymbol{\mu}$ and variance Σ as.

$$\boldsymbol{\mu} = \phi \Sigma X_{\boldsymbol{\gamma}}^T \mathbf{y}; \Sigma = (\phi X_{\boldsymbol{\gamma}}^T X_{\boldsymbol{\gamma}} + D)^{-1},$$

where D is a diagonal matrix with diagonal element $\left\{\frac{\tau_j}{b^2}\right\}_{1 \le j \le p}$. Similar to Carvalho and Polson (2010); Polson and Scott (2011), it is convenient to introduce the shrinkage coefficient, $\kappa_j = \left(\frac{\tau_j}{b^2 \phi}\right) / \left(1 + \frac{\tau_j}{b^2 \phi}\right)$. Under the orthogonal design, the posterior mean and variance of β_j's corresponding to $j \in \{j : \gamma_j = 1\}$ are

$$\begin{aligned} E(\beta_j | \mathbf{y}, \phi, b) &= [1 - E(\kappa_j | \mathbf{y}, \phi, b)] a_j, \\ Var(\beta_j | \mathbf{y}, \phi, b) &= [1 - E(\kappa_j | \mathbf{y}, \phi, b)] \phi^{-1}. \end{aligned} \tag{2.22}$$

The coefficient κ_j's represent how much shrinkage is placed on the initial estimation of β_j's. $\kappa_j \to 0$, yields no shrinkage, and $\kappa_j \to 1$ yields near-total shrinkage. With this definition of κ_j, it is easy to derive the density function of κ_j, $p(\kappa_j)$. Table 2.1 lists $p(\kappa_j)$'s based on the three prior settings given $\phi = 1$.

In order to compare the performance of different variance mixture priors $p(\tau_j)$ on the marginal selection probability and avoid notational abuses, it is convenient to assume ϕ fixed and use a transformation $\sqrt{\phi}\mathbf{y} \to \mathbf{y}$ such that $a_j\sqrt{\phi} \to a_j$, $b\sqrt{\phi} \to b$ and $\sqrt{\phi}C \to C$. This is equivalent to assume $\phi = 1$, but keep in mind that a_j's, $\mathbf{c}_j$'s and b are scaled by $\sqrt{\phi}$ unless stated otherwise.

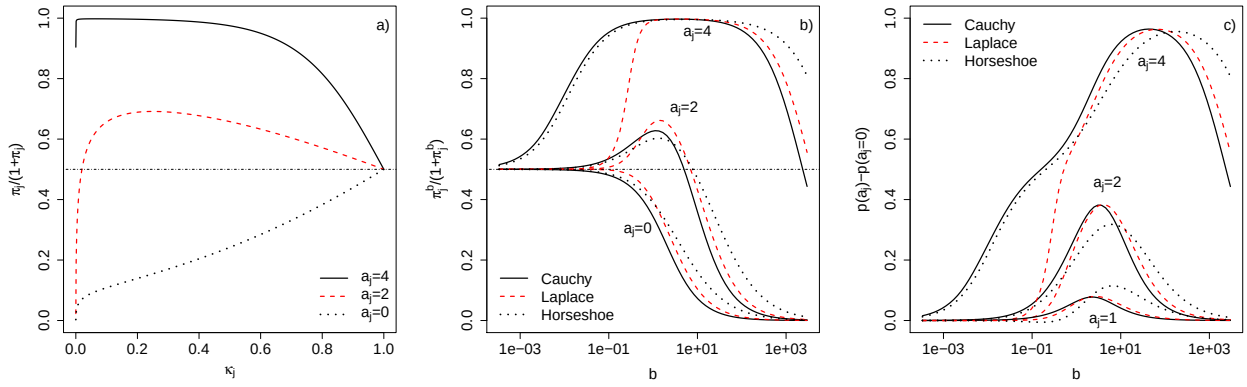

Figure 2.3: Selection probability curves against κ_j (a). The curves of marginal selection probability against global shrinkage parameter b (b). Marginal selection probabilities with baseline subtracted (c). All plots are under orthogonal designs.

With these coefficients now defined, the following theorems connect the marginal odds of γ_j given the data with κ_j's and b. Based on (2.1) and (2.2), the join distribution for $\boldsymbol{\gamma}, \boldsymbol{\beta}, \boldsymbol{\tau}$ given b is

$$p(\boldsymbol{\gamma}, \boldsymbol{\beta}, \boldsymbol{\tau}|\mathbf{y}, b) \propto p(\mathbf{y}|\boldsymbol{\gamma}, \boldsymbol{\beta})p(\boldsymbol{\beta}|\boldsymbol{\tau}, b)p(\boldsymbol{\tau}),$$

and the marginal probability for γ_j is $p(\gamma_j|\mathbf{y}, b) = \int \sum_{\boldsymbol{\gamma}_{\bar{j}}} p(\gamma_j, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\beta}, \boldsymbol{\tau}|\mathbf{y}, b) d\boldsymbol{\beta} d\boldsymbol{\tau}$. Thus the

marginal odds for $\gamma_j = 1$ given b is

$$\pi_j^b = \frac{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} p(\gamma_j = 1, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\beta}, \boldsymbol{\tau}|\mathbf{y}, b) d\boldsymbol{\beta} d\boldsymbol{\tau}}{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} p(\gamma_j = 0, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\beta}, \boldsymbol{\tau}|\mathbf{y}, b) d\boldsymbol{\beta} d\boldsymbol{\tau}}.$$

Theorem 2.5.1. *For the Bayesian model defined in (2.1) and (2.2), the marginal odds of γ_j, defined as π_j^b, has the following form*

$$\pi_j^b = \int \pi_j \xi_j p(\kappa_j) d\kappa_j, \tag{2.23}$$

where $p(\kappa_j)$ is the density function of κ_j,

$$\pi_j = \kappa_j^{\frac{1}{2}} \exp\left[\frac{a_j^2}{2}(1 - \kappa_j)\right], \tag{2.24}$$

and ξ_j is a positive real function of κ_j

$$\xi_j = \frac{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} \xi(\gamma_j = 1, \kappa_j, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\tau}_{\bar{j}}) p(\boldsymbol{\tau}_{\bar{j}}) d\boldsymbol{\tau}_{\bar{j}}}{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} \xi(\gamma_j = 0, \kappa_j, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\tau}_{\bar{j}}) p(\boldsymbol{\tau}_{\bar{j}}) d\boldsymbol{\tau}_{\bar{j}}}. \tag{2.25}$$

1. *For general cases*

   $$\xi(\gamma_j, \kappa_j, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\tau}_{\bar{j}}) = \exp\left[\frac{1}{2}\left(\mathbf{a}_{\bar{j}} - (1 - \kappa_j^{\gamma_j}) a_j \mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}\right)^T \Omega_j^{-1} \left(\mathbf{a}_{\bar{j}} - (1 - \kappa_j^{\gamma_j}) a_j \mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}\right)\right] |\Omega_j|^{1/2} |D_{\bar{j}}|^{1/2}$$

   with $\Omega_j = [D_{\bar{j}} + X_{\boldsymbol{\gamma}_{\bar{j}}}^T X_{\boldsymbol{\gamma}_{\bar{j}}} - (1 - \kappa_j)^{\gamma_j} (\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}} \mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}^T)]^{-1}$.

   For orthogonal designs, $\xi_j = 1$, and

   $$\pi_j^b = \int \pi_j p(\kappa_j) d\kappa_j. \tag{2.26}$$

2. *For orthogonal designs, if $\kappa_j \to 0$, then $\pi_j \to 0$, and if $\kappa_j \to 1$, then $\pi_j \to 1$. Similarly, if $b \to 0$, then $\pi_j^b \to 1$, and if $b \to \infty$, then $\pi_j^b \to 0$.*

Proof: See Section 2.7.2.

From Theorem 2.5.1 we can see that in general the marginal odds $\pi_j^b \neq \int \pi_j p(\kappa_j) d\kappa_j$, the marginal odds of the orthogonal design. According to Equation (2.23), when the correlation among predictors is not negligible, the odds will be "blurred" by the coefficient ξ_j, and the marginal selection probability is blurred too. Basically, ξ_j is a complex function of $a_j, \mathbf{c}_{\bar{j}}$, and $\tau_k/b^2, k \neq j$ or κ_j. Furthermore, it is infeasible to calculate ξ_j given large p with more than 2 predictors are correlated. However, by focusing on the orthogonal design, we are able to glean a better understanding of the marginal selection probability.

Combining Theorem 2.5.1 and Figure 2.3, we can get a better idea of the behaviors of π_j and π_j^b. Figure 2.3 (a) plots the selection probability as a function of κ_j according to odds π_j (2.24), and Figure 2.3 (b) plots the marginal selection probabilities according to π_j^b with different prior $p(\kappa_j)$'s. We can see that for the orthogonal design, Expression (2.24) and (2.26) indicate that π_j and π_j^b are monotone functions of a_j. This relationship is also demonstrated by the different selection probability curves in Figure 2.3. In the ideal scenario, all noise predictors will demonstrate the same selection probability since $a_j = 0$, which defines the baseline selection probability curve in Figure 2.3 (a-b).

Thus ideally, any signals with $a_j \neq 0$ deviate from the baseline curve. However, when correlations among variables exist, the situation becomes complicated. First, even though the correlation among variables are small enough so that π_j and π_j^b are still monotone function of a_j, the baseline will be blurred and extended to a band. To see this, consider $k \in \bar{V}^*$ and $j \in V^*$ where V^* is the set of true nodes and its complement is $\bar{V}^*$. Because $c_{kj} = \mathbf{x}_k^T \mathbf{x}_j \neq 0$, $\mathbf{x}_k$ will have a fake signal, we have that $a_k = \mathbf{x}_k^T \mathbf{y} = c_{kj} a_j$. Thus all the noise predictors will demonstrate false signals if they exhibit non-negligible correlations with the true signals. This makes separating the true predictors with small signals from the noise difficult. Secondly, because of ξ_j, even the selection probability for large true

signals will be distorted by their correlated fake signals. For example in Figure 2.3 (b), if the fake signal, $a_j = 2$, is correlated with the true signal, $a_j = 4$, then the profile curve of $a_j = 2$ and $a_j = 4$ will show some "interacting" behavior at $b \approx 1$ where the selection probability of fake signal reaches the maximum. We will expand on this behavior in the simulation analysis.

Thus in general the largest gap that separates $a_j = 2$ and $a_j = 4$ is not around a moderate b, say $b \approx 1$, but in two regions around $b \approx 0.1$ and $b \approx 1000$. Furthermore, the second result of Theorem 2.5.1 states some asymptotic behaviors of π_j and π_j^b as $\kappa_j \to 0$ or $b \to \infty$ and $\kappa_j \to 1$ or $b \to 0$. This can be clearly seen in Figure 2.3 (a-b) where with small shrinkage ($\kappa_j \to 0$ or $b \to \infty$), both π_j and π_j^b approach 0, and with large shrinkage ($\kappa_j \to 1$ or $b \to 0$), they approach 0.5. However, the dropping rate depends on the magnitude of the signal and the prior $p(\kappa_j)$. For example in Figure 2.3 (a) we can see for large signal $a_j = 4$, the selection probability maintains at 1 for $\kappa_j \to 0$ till the last point. In addition, Figure 2.3 (b) shows that the selection probability curves are different for different priors: some drop off quickly, such as in the case of Laplace prior, while otheres are robust to shrinkage, as in the case of the horseshoe prior.

Our next question will be how to choose an appropriate prior for τ_j or $p(\kappa_j)$. Based on Figure 2.3 and Theorem 2.5.1, we have some guidelines to follow in order to select $p(\tau_j)$: (1) The rate of increase for π_j^b must be large when the signal increases, so that the large true signal can be separated from the noise more easily. (2) π_j^b drops to 0.5 or 0 slowly when $b \to 0$ or $b \to \infty$ so we have a wider windows for b such that the true signals maintain a high selection probability.

The following theorems will further help us to understand the relationship between π_j^b and the shrinkage coefficient κ_j.

Theorem 2.5.2. *For the Bayesian model (2.1) and (2.2) with orthogonal design, suppose prior*

$p(\beta_j)$ is a zero mean scale mixture of normals: $[\beta_j|\tau_j, b] \sim N(0, b^2\tau_j^{-1})$, with τ_j having proper prior $p(\tau_j)$. Define the marginal density $m_j = m(\mathbf{y}|\gamma_j = 1)$ as

$$m_j = \int p(\mathbf{y}|\beta_j, \gamma_j = 1)p(\beta_j|\tau_j)p(\tau_j)d\beta_j d\tau_j.$$

where $p(\mathbf{y}|\beta_j, \gamma_j) \equiv p(\mathbf{y}|\beta_j, \gamma_j, \boldsymbol{\beta}_{\bar{j}}, \boldsymbol{\gamma}_{\bar{j}})$. If m_j is finite for all $\mathbf{y}$, then

1.
$$E(\beta_j|\mathbf{y}, \gamma_j = 1) = a_j + \frac{1}{m_j}\mathbf{x}_j^T\frac{\partial}{\partial \mathbf{y}}m_j = a_j + \mathbf{x}_j^T\frac{\partial}{\partial \mathbf{y}}\log m_j. \tag{2.27}$$

2.
$$E(\beta_j|\mathbf{y}, \gamma_j = 1) = \frac{d}{da_j}\log \pi_j^b = [1 - E(\kappa_j|\mathbf{y}, \gamma_j = 1)]a_j. \tag{2.28}$$

Proof: See Section 2.7.3

A similar result to the first part of Theorem 2.5.2 can be found in Pericchi and Smith (1992) and Polson and Scott (2010). We are more interested, however, in the second result, which gives the relationship among the expectation of β_j, the derivative of log π_j^b respect to a_j, and the shrinkage coefficient. We prefer a larger value of the derivative of log marginal odds so that the large signal can be further separated from the baseline. Equation (2.28) indicates that to achieve this, we not only require a large a_j, but also that the expectation of the shrinkage parameter κ_j be small. This is confirmed by Figure 2.3 (a), where we see that the largest separation between the signals and the baseline is on the side of $\kappa_j \to 0$. Thus if we were to integrate out κ_j to have π_j^b, we would want the density $p(\kappa_j)$ to have substantial mass around the region with largest separation. However, we don't want $\kappa_j = 0$ since this is equivalent to having exactly no shrinkage, and would result in all π_j's dropping to zero.

Therefore, the general requirement for $p(\kappa_j)$ based on Theorem 2.5.1, 2.5.2 and Fig-

ure 2.3 is to maintain substantial mass around a region of small shrinkage. Surprisedly, (2.28) seems contradict to the usual variable selection strategy to recover the sparsity in the region of large shrinkage. In fact, it is possible to separate the signal and noise in large shrinkage region. Furthermore, large shrinkage has some advantages, such as stability, faster mixing, lower sensitivity to nodes number p and so on. Therefore, large shrinkage is a second choice when the signals are robust to such conditions. However, in this dissertation we focus on the small shrinkage region where the consistency in variable selection seems to be satisfied more often.

2.5.2 Dynamic Properties of the Odds with Different Priors

To explain the different behaviors of the selection probability caused by different priors, we need to examine the details of the density distribution $p(\kappa_j)$. Carvalho and Polson (2010) and Polson and Scott (2010) discussed the performance of different types of priors in the Bayesian regularization with difference weight on shrinkage. They focus on the effects of the estimation of signals.

In the prior for β_j's, b is the global shrinkage parameter. As $b \to 0$, large global shrinkage effect is applied on all β_j's, and as $b \to \infty$, the global shrinkage effect becomes negligible. Table 2.1 lists the priors $p(\tau_j)$'s with their corresponding $p(\kappa_j)$'s and marginal prior $p(\beta_j|b)$s. The effect of modifying b depends on the type of prior used.

To see this, Figure 2.4 compares the density function $p(\beta_j|b)$'s around zero and along the tails, and the density function $p(\kappa_j)$'s given different b's. By examining $p(\kappa_j)$'s in Figure 2.4 together with Figure 2.3 (c), we see how b effects the selection probability profile by putting different weight on shrinkage. Figure 2.3 (c) plots the selection probability profile with the baseline subtracted. It can be seen that for an orthogonal design, the

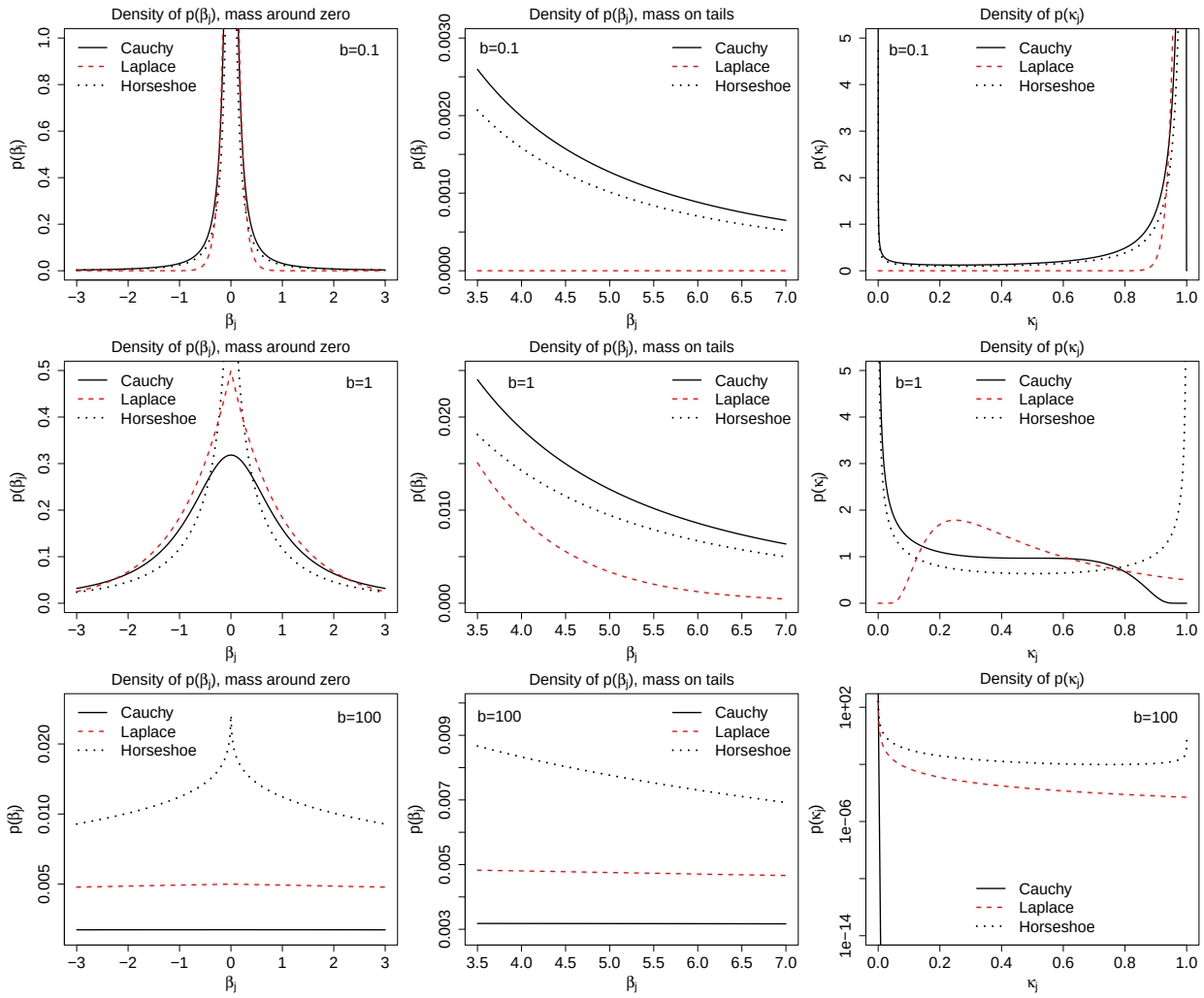

Figure 2.4: Marginal prior density functions of β_j and density functions of κ_j for different b.

larger the magnitude of the selection probability, the more markedly the true signal is distinguished from the baseline. In the small b ($b \leq 1$) region, the descending order of magnitude for a given signal is: the Cauchy prior, the horseshoe prior, and the Laplace prior, which is consistent with the $p(\kappa_j)$ plots in Figure 2.4 at $b = 0.1$, where the order of density masses on the small κ_j region also exhibits this ordering. In addition, since the Cauchy and horseshoe priors put similar mass around the small κ_j side, their selection probabilities behave almost identically for small b as shown in Figure 2.3 (c).

On the other hand, the order changes for $b = 100$ to: horseshoe, Laplace and Cauchy (Figure 2.4). Again this is consistent with the selection probability order in Figure 2.3 (c) for large b ($b \geq 100$). The reason that Cauchy prior becomes worse for large b is that all the mass of $p(\kappa_j)$ is absorbed to $\kappa_j = 0$, which is not we might have expected, as mentioned before. For a moderate b, such as $b = 1$, all priors have substantial mass around small κ_j side, as shown in Figure 2.4, thus all priors behave similarly. This is confirmed by Figure 2.3 (c), where the selection probabilities for different prior seems similar around $b = 1$ for large signals.

The above analysis also shows that more weight on large shrinkage is not as important as it is on small shrinkage for the purpose of distinguishing the signals. Therefore, we find that the horseshoe prior is superior to other two, even though the marginal prior $p(\beta_j|b)$ of the horseshoe prior is not as heavy-tailed as the Cauchy prior for small b. The horseshoe prior demonstrates that for a wide range of b, $p(\kappa_j)$ maintains substantial mass on the small shrinkage side of κ_j. On the other hand, the Laplace prior has almost zero mass around small κ_j when b is small, and Cauchy prior has all mass absorbed to $\kappa_j = 0$ when b is very large, each diminishing its respective performance for the particularly b values. Our argument to evaluate the priors is thus different from Carvalho and Polson (2010), where they conclude that the horseshoe prior is superior because it has substantial mass on both small shrinkage and large shrinkage for estimation. Of course, although we prefer small shrinkage for variable selection, large shrinkage does have its advantages, which we must consider. For example, we find in our simulation results that with large shrinkage, the Gibbs sampler converges faster even with very large p.

To further examine the dynamics of the selection probability profile, the following theorem gives some asymptotic behaviors about the derivation of log π_j^b with respect to $|a_j|$ and b, and hence it helps us with the evaluation of different priors.

Theorem 2.5.3. *Consider the inverse of τ_j, $\sigma_j^2 = \tau_j^{-1}$, with prior density, $p(\sigma_j^2)$. If we let $\sigma_j^2 \to \infty$,*

$$p(\sigma_j^2) \sim (\sigma_j^2)^{\alpha-1} e^{-\lambda\sigma_j^2} L(\sigma_j^2) d\sigma_j^2$$

for some slowly varying function $L(x)$ such that as $x \to \infty$ for all $t > 0$, $L(tx)/L(x) \to 1$, then

1. *as $a_j \to \infty$*

$$\frac{d}{d|a_j|} \log \pi_j^b \sim \begin{cases} |a_j| + \frac{2\alpha-1}{|a_j|} & if \;\; \lambda = 0 \\ |a_j| + \frac{\alpha-1}{|a_j|} - \frac{\sqrt{2\lambda}}{b} & if \;\; \lambda > 0. \end{cases} \tag{2.29}$$

2. *For large a_j, as $b \to 0$*

$$\frac{d}{db} \log \pi_j^b \sim \begin{cases} \frac{d}{db} \log L^b\left(a_j^2\right) & if \;\; \lambda = 0 \\ \frac{\sqrt{2\lambda}}{b^2}|a_j| + \frac{d}{db} \log L^b\left(|a_j|\right) & if \;\; \lambda > 0. \end{cases} \tag{2.30}$$

where $L^b(x)$ is a function conditioning on b (see Section 2.7.4)

Proof: See Section 2.7.4

Particulary, a $L^b(x)$ has forms of $b \exp\left(-\frac{b^2}{2x}\right), b^{-2}$, and $\frac{b}{b^2+x^2}$ for the Cauchy, Laplace and horseshoe prior, respectively. When $|a_j|$ is large, as Theorem 2.5.3 assumes, $\frac{d}{db} \log L^b$ is negligible.

Theorem 2.5.3 is similar to the tail robustness theorem discussed by Polson and Scott (2011) about the marginal density $m(\mathbf{y}|\gamma_j = 1)$, which implies that the shrinkage will vanish for any scale mixture normals with $p(\sigma_j^2)$ when using heavier tails (such as the Cauchy and horseshoe prior), while remaining non-diminishing for $p(\sigma_j^2)$ with exponential tails (such as Laplace prior). Combined with (2.27) of Theorem 2.5.2, we conclude that the estimation of β_j is robust if done with long-tailed priors. A similar form of robustness

can be found for π_j^b. The robustness of π_j^b implies a fast rate of change for π_j^b as the signal magnitude increases, and a smaller rate of change for π_j^b as shrinkage increases. These two characteristics are important since they make distinguishing the signals easier. Large $\frac{d}{d|a_j|}\log \pi_j^b$ helps to distinguish the signals from the baseline, while small $\frac{d}{db}\log \pi_j^b$ leads to a wide window of b such that the selection probability of the true signals remains highly.

Expression (2.29) indicates that for priors with exponential tails ($\lambda > 0$), the selection probability π_j^b increases with a smaller rate than when the signal magnitude increases for heavier tail priors. Meanwhile, Expression (2.30) shows that for priors with exponential tails, π_j^b drops at a faster rate ($\sim \sqrt{2\lambda}/b^2$) as $b \to 0$. We can also compare the dropping rates of $\log \pi_j^b$ of the Cauchy and horseshoe prior as $b \to 0$. Since $L^b(x) = \exp\left(-\frac{b^2}{2x^2}\right) b$ and $(1 + b^2/x^2)^{-1}b$ for the Cauchy and horseshoe priors respectively, it turns out that $d \log L^b(x^2)/db \approx 1/b$ as $b \to 0$ for both priors. This means that the dropping rates as $b \to 0$ are similar between the Cauchy and horseshoe priors, which is confirmed in Figure 2.3 (b). Note that the second conclusion of Theorem 2.5.3 does not apply to $b \to \infty$ unless $|a_j| \to \infty$ faster than $b \to \infty$, i.e., $b/|a_j| \to 0$. Therefore L^b remains a slowly varying function. However, as shown by the exact calculation in Figure 2.3 (b), the horseshoe prior is also the most robust as $b \to \infty$.

Figure 2.5 gives the exact calculation of $r_a = \frac{d}{d|a_j|}\log \pi_j^b$ and $r_b = \frac{d}{db}\log \pi_j^b$ for three priors. In Figure 2.5 (a) we can see that r_a is the nearly the same for three priors at large b. However, when b is small, r_a is reduced by certain values for the Laplace prior. Meanwhile, it remains the same for the Cauchy and horseshoe priors. So the exact calculation confirms Theorem 2.5.3. Similarly, the exact calculation confirms the result of Theorem 2.5.3 about r_b. As we see in Figure 2.5 (b), r_b increases exponentially as $b \to 0$ under the Laplace prior, which means that as $b \to 0$, the selection probability by the Laplace prior will exponentially drop to 0.5, a behavior that we have already observed in Figure 2.3 (b).

Based on the discussion in Section 2.5.1 and this section, the horseshoe prior performs the best in terms of the marginal selection probability, while the Cauchy prior performs second best, leaving the Laplace prior as the worst. This is due to the fact that the selection probability drops too quickly as the shrinkage parameter increases.

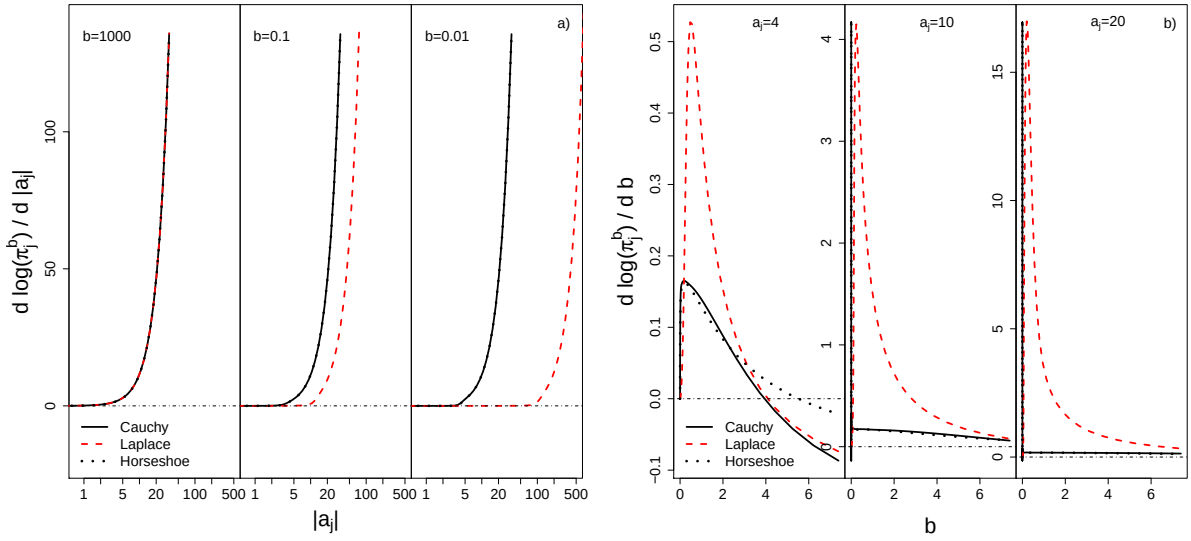

Figure 2.5: The derivative of log odds respect to $|a_j|$ against given different b (a). Note the curves of the Cauchy and horseshoe priors are overlapped. The derivative of log odds with respect to b given different a_j (b).

2.5.3 Expressions for π_j^b

In the above sections, we discussed the properties of the odds π_j^b for orthogonal designs, but did not show how to calculate π_j^b. Those curves are calculated via Monte Carlo simulations, which turn out to be very precise. In some cases, we may want to calculate π_j^b directly. There is no closed form of π_j^b for the three different priors. However, we can see that at least for the Laplace and horseshoe priors, π_j^b can be expressed by some special functions.

For Laplace prior,

$$\pi_j^b = \sqrt{\frac{\pi}{2}} b^{-1} \exp\left[\frac{(|a_j| - b^{-1})^2}{2}\right] \left\{\Phi\left(|a_j| - b^{-1}\right) + \exp\left(2|a_j|b^{-1}\right) \Phi\left(-|a_j| - b^{-1}\right)\right\}, \quad (2.31)$$

where $\Phi(\cdot)$ is the CDF of the standardized normal distribution. This expression can be directly used to calculate the marginal selection probability given b.

For horseshoe prior,

$$\pi_j^b = \frac{1}{\pi} b^{-1} Be\left(1, \frac{1}{2}\right) \Phi_1\left(\frac{1}{2}, 1, \frac{3}{2}, \frac{a_j^2}{2}, 1 - b^{-2}\right), \quad (2.32)$$

where $Be(\cdots)$ denotes the beta function, and $\Phi_1(\cdots)$ is the degenerate hypergeometric function of two variables (Gordy, 1998; Polson and Scott, 2010). Φ_1 can be evaluated by using a series of hypergeometric ${}_2F_1$ functions (Gordy, 1998).

The derivative of the above expressions is shown in Section 2.7.5. The odds π_j^b of the Cauchy prior does not have an analytic form and cannot be represented by a known set of special functions. Hence we use the Monte Carlo approach to calculate π_j^b under the Cauchy prior.

2.6 Connection of BIGM to Simulated Tempering and Generalization by Lévy Process

Li and Zhang (2010) discussed the difficulty of sampling around phase transition in a SSVS model by assigning a Ising prior for $\boldsymbol{\gamma}$. The difficulty is that, given an Ising model, there is a threshold for the interaction strength such that when the interaction magnitude is larger than this threshold, the MCMC sampling will dramatically slow down and result

in either an overwhelming number of nodes selected or extremely few ones. The problem becomes even worse when J_{ij}'s and h_j's are all random, such as our model. However, the family of exchange Monte Carlo and simulated tempering algorithm has be developed to handle the slow mixing problem (Geyer and Thompson, 1995; Iba, 2001; Lyubartsev et al., 1992). By introducing the scale normal mixture for $\boldsymbol{\beta}$, our model is an special simulated tempering algorithm, which thus improves the mixing issue.

To get a better understanding of the simulated tempering algorithm, consider the usual Ising model with $U(\boldsymbol{\gamma}, J) = \sum_{i<j} J_{ij}\delta_{ij}$ (for simplicity no external field h_i^* included), then the Boltzman distribution is expressed as

$$p(\boldsymbol{\gamma}|T, J) = \frac{1}{Z(T)} \exp\left[-T^{-1}U(\boldsymbol{\gamma}, J)\right],$$

where T represents the temperature (or the scale of variation), and J_{ij} is random and follows some distribution $p(J_{ij})$ such as standard Gaussian distribution. When $T \to 0$, the effective interaction $\tilde{J}_{ij} = T^{-1}J_{ij} \to \infty$. Thus if T is lower than some critical temperature, the strong interaction will lead to non-ergodic behavior such as the slowdown of the MCMC and extremely large proportion of $\gamma_j = 1$. The reason for this is that the low temperature phase of the disordered Ising model generally has numerous local minima, which are separated from each other by energy barriers. The characteristic time in which the system escapes from a local minimum, however, increases rapidly as the temperature decreases or the interaction increases. The family of tempering algorithm treats temperature T as a dynamical variable (Lyubartsev et al., 1992), and the joint distribution $p(\boldsymbol{\gamma}, T)$ is represented as

$$p(\boldsymbol{\gamma}, T, J) \propto p(\boldsymbol{\gamma}|T, J) \prod_{i<j} p(J_{ij})p(T), \tag{2.33}$$

where $p(T)$ is the distribution of T. The prior information for the T thus represents the

range and mass of the temperature to sample the MCMC. With some variable transformation done by replacing $T_b^{-1/2} J_{ij}$ with $\tilde{J}_{ij}$, where $T^2 = T_b$, the joint distribution (2.33) then becomes

$$p(\boldsymbol{\gamma}, T_b, \tilde{J}) \propto p(\boldsymbol{\gamma}|\tilde{J}) \prod_{i<j} p(\tilde{J}_{ij}|T_b) T_b^{-1/2} p(T_b),$$

where $p(\boldsymbol{\gamma}|\tilde{J}) \propto \exp\left[-\sum_{ij} \tilde{J}_{ij}\delta_{ij}\right]$, $p(\tilde{J}_{ij}|T_b) \propto p(T_b^{1/2}\tilde{J}_{ij}) T_b^{1/2}$ (with some notational abuse, the later $p(\cdot)$ represents the same density function of $p(J_{ij})$). Clearly, T_b is a global temperature parameter here. If we introduce the local temperature parameter T_b for each interaction J_{ij}, then the marginal prior for $\tilde{J}_{ij}$ is

$$p(\tilde{J}_{ij}) \propto \int_0^\infty p(\tilde{J}_{ij}|T_b) T_b^{-1/2} p(T_b) dT_b.$$

If $p(J_{ij}) \sim N(0,1)$, the above posterior for $\tilde{J}_{ij}$ is a normal scale-mixture whose mixing measure is expressible in terms of the density of the subordinator T_b. Hence according to the Theorem 3 of Polson and Scott (2011), with the simulated tempering algorithm, the interaction of the random Ising model (2.33) can be expressed as a Lévy process mixture scaled by T_b, and T_b is a nondecreasing pure-jump Lévy process with marginal density $p(T_b)$ at time b.

As another algorithm in the same family, the exchange monte carlo algorithms (or parallel tempering) simultaneously and independently simulate $K \geq 2$ replicas of the MCMC trace under different temperatures, and exchange the $\boldsymbol{\gamma}$ configurations of the replicas with certain acceptance probability by referring to the energy cost ΔU. The analogy between the exchanged monte carlo and simulated tempering algorithm is clear in terms of the mixture distribution of the interaction J_{ij}. For simulated tempering the mixture weight is the continuous prior $p(T_b)$ while the exchanged monte carlo is mixed with weight on a set of discrete temperatures. In both algorithms, the low temperature process

can access a representative set of local energy minimums with the accompany of the high temperature process which are generally able to sample large volumes of configuration space to keep the configuration from trapping in some local minimum.

We see that the best interpretation of the simulated tempering algorithm is as the Ising model with a normal scale-mixture prior mixed by the Lévy process. On the other hand, a BIGM with Lévy process mixtures can also be understood as an Ising model sampled by a simulated tempering algorithm. To see this, we can generalize both the Cauchy and Laplace prior into the framework of the normal/generalized inverse Gaussian mixture. The marginal prior of β_j for both priors can be expressed using one formula

$$\begin{aligned} p(\beta_j|u,v,w) &= \int_0^\infty p(\beta_j|\tau_j)\tau_j^{-1}g(\tau_j)d\tau_j = \int_0^\infty p(\beta_j|\tau_j)p(\tau_j|u,v,w)d\tau_j \\ &= \int_0^\infty \frac{1}{\sqrt{2\pi}}\exp\left(-\beta_j^2\tau_j/2\right)\tau_j^{1/2}\frac{(u/v)^w}{2K_w(uv)}\tau_j^{w-1}\exp\left[-\left(u^2\tau_j+v^2\tau_j^{-1}\right)/2\right]d\tau_j \\ &= \frac{1}{\sqrt{2\pi}}\cdot\frac{(u/v)^w}{K_w(uv)}\cdot\frac{K_{w+1/2}\left(v\sqrt{\beta_j^2+u^2}\right)}{\left(\sqrt{\beta_j^2+u^2}/v\right)^{w+1/2}}, \end{aligned} \tag{2.34}$$

where $K_w(x)$ denotes the modified Bessel function of the third kind with w. $p(\tau_j|u,v,w)$ is the generalized inverse Gaussian, $GING(w,u,v)$, distribution with parameters u,v and w such that $w \in \mathbb{R}$ while u and v are both nonnegative and not simultaneously 0. Note for the two special cases we adopted in this dissertation, the Cauchy and Laplace priors, the values for these three parameters found on the boundary. However, it turns out both the prior $p(\tau_j|u,v,w)$ and the marginal prior $p(\beta_j|u,v,w)$ exist when the parameters approach the boundary. For example, for the Cauchy prior, $w = 1/2$, $u = b$, and $v = 0$, thus $p(\tau_j|b,v,1/2) \to G(1/2,b^2/2)$ as $v \to 0$, where we use identity $K_{-1/2}(x) = \sqrt{\pi/2}x^{-1/2}\exp(-x)$. Hence the distribution of β_j reduces to the Cauchy dis-

tribution with scale parameter b, where we use $\lim_{x\to 0} x^w K_w(x) \to 2^{w-1}\Gamma(w)$ and $\Gamma(w)$ is the Gamma function. For the Laplace prior, $w = -1$, $u = 0$, and $v = b^{-1}$. the limit of prior $p(\tau_j|u, b^{-1}, -1) \to IG[1, (2b^2)^{-1}]$ as $u \to 0$, where $IG(\cdot)$ stands for the inverse gamma distribution, and we used the index symmetric index $K_{-w}(x) = K_w(x)$.

According to Theorem 3 of Polson and Scott (2011), we see that $\tau_j^{-1} g(\tau_j) = p(\tau_j|u, v, w)$ where $g(\tau_j)$ is the density of the subordinator τ_j at (u, v, w). Analogous to the discussion with the simulated tempering algorithm for the Ising model, we can see that the temperature parameter in our model is τ_j. However, there are several differences, such as in our model, $J_{ij} \propto \beta_i\beta_j$ and we assign normal mixture prior for β_j's, while in the simulated tempering algorithm of the regular Ising model, the prior is assigned to J_{ij} directly.

Now we can explore the temperature effect of τ_j. When $\tau_j \to \infty$, which is equivalent to $\kappa_j \to 1$, the system is in a high temperature state. This means that the MCMC is exploring the whole configuration space, and there is no precise sampling for a local energy minimum. This is also equivalent to saying that for each node, the odds of $\gamma_j = 1$ is equal to one, since under high temperature every node "heats up," and the chance of being "up" and "down" is even. This means that the marginal selection probability for all nodes is $1/2$, as shown in Figure 2.3 (a) when $\kappa_j \to 1$. On the other hand, when $\tau_j \to 0$, which is equivalent to $\kappa_j \to 0$, the system is in a low temperature state. Starting from some initial state, all nodes configurations will be trapped into their energy minimum (local maximum likelihood), which is $\gamma_j = 0$ for most nodes in the orthogonal design, unless the external field $h_j \propto a_j$ is strong enough to force the node into the state $\gamma_j = 1$. This is why we see in Figure 2.3 (a) for small κ_j the selection probability is 0 for nodes with small a_j and remains 1 for nodes with very large a_j.

In this scenario, it is easy to see the role of b too, which is opposite to τ_j if we look at the prior $p(\beta_j|\tau_j, b) \sim N(0, b^2/\tau_j)$. In fact, we can also see the role of b plays by representing

the hierarchical model as $p(\boldsymbol{\gamma}|\mathbf{y},\boldsymbol{\beta})\prod p(\beta_j|\tau_j)p(\tau_j|b)$, where $p(\tau_j|b) = \tau_j^{-1/2}\exp(-b^2\tau_j/2)$, $\tau_j^{-2}\exp[-(2b^2\tau_j)^{-1}]$ and $(\tau_j + b^{-2})^{-1}$ for the Cauchy, Laplace and horseshoe priors respectively. Thus we see that b controls how the local temperature parameter τ_j is distributed. Large b limits the variation of τ_j and increases the mass around zero, and small b means the range over which τ_j varies is large. This is also consistent to Figure 2.3 (b), where we see that when $b \to 0$, $\kappa_j \to 1$ and τ_j varies widely, thus the system is in a high temperature state. Alternately, if $b \to \infty$, $\kappa_j \to 0$ and τ_j will be limited to varying around 0, then the system is in a low temperature state.

The generalization of the Cauchy prior and Laplace prior into the Lévy process mixture not only shows that there is a deep connection between BVS and the Ising model with a tempering algorithm, it also provides flexibility to choose priors with different shrinkage characteristics. We did not discuss the generalization of the horseshoe prior as a Lévy process, but similar conclusion can be drawn about the horseshoe prior in terms of shrinkage or tempering.

2.7 Proofs of the Lemmas and Theorems

2.7.1 Proof of Theorem 2.3.1

In general, in the Markov chain of MH algorithm, the move from current state $\boldsymbol{\gamma}_c^0$ to the proposed state $\boldsymbol{\gamma}_c^*$ in the cluster has the transition probability, $P(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)$, which satisfies the detailed balance condition

$$\frac{P(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)}{P(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)} = \frac{p(\boldsymbol{\gamma}_c^*|\mathbf{y},\boldsymbol{\beta},\phi)}{p(\boldsymbol{\gamma}_c^0|\mathbf{y},\boldsymbol{\beta},\phi)} = \exp\left\{-\left[U(\boldsymbol{\gamma}_c^*) - U(\boldsymbol{\gamma}_c^0)\right]\right\}. \tag{2.35}$$

The transition probability can be broken down into two parts:

$$P(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*) = g(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)A(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*),$$

where $g(\cdot)$ is the selection probability, which is the probability given $\boldsymbol{\gamma}_c^0$ that the new target state generated, and $A(\cdot)$ is the acceptance ratio. Thus

$$\frac{g(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)A(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)}{g(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)A(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)} = \exp\left\{-\left[U(\boldsymbol{\gamma}_c^*) - U(\boldsymbol{\gamma}_c^0)\right]\right\}. \tag{2.36}$$

Now we consider the move $\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*$, starting with a particular cluster c and then adding the others to it in a particular order. Consider also the reverse move, which takes us back to $\boldsymbol{\gamma}_c^0$ from $\boldsymbol{\gamma}_c^*$, starting with exactly the same cluster (except the state in the cluster is flipped), and adding the others to it in exactly the same way as in the forward move. The probability of choosing the cluster (if the cluster is the seed node) is exactly the same in the two directions, as is the probability of adding each node to the cluster. The only difference between the two directions is the probability of "breaking" bonds around the edge of the cluster. Because the cluster couples with all $j \in \bar{c}$, for both directions, there are $|\bar{c}|$ bonds which have to be broken in order to flip the cluster. These broken bonds represent the affinity between the cluster and the spins which were not added to the cluster by the algorithm. We represent the probability of not adding such a node in forward move as $1 - p_{a,j}^0, j \in \bar{c}$ and in backward move as $1 - p_{a,j}^*, j \in \bar{c}$. Thus the probability of not adding all of them, which is proportional to the selection probability $g(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)$ for the forward move, is $\prod_{j\in\bar{c}}(1-p_{a,j}^0)$. In the reverse move then the probability of doing it is $\prod_{j\in\bar{c}}(1-p_{a,j}^*)$. The condition of detailed balance, Equation (2.35), along with Equation (2.36), then tells

us that

$$\frac{g(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)A(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)}{g(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)A(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)} = \prod_{j \in \bar{c}} \left(\frac{1 - p_{a,j}^0}{1 - p_{a,j}^*} \right) \cdot \frac{A(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)}{A(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)} = \exp\left\{ - \left[U(\boldsymbol{\gamma}_c^*) - U(\boldsymbol{\gamma}_c^0) \right] \right\}. \tag{2.37}$$

Note that the energy change $U(\boldsymbol{\gamma}_c^*) - U(\boldsymbol{\gamma}_c^0)$ is only determined by the bonds (the coupling between c and $\bar{c}$) and coupling of c with the external field $\mathbf{h}^*$, i.e.,

$$U(\boldsymbol{\gamma}_c^*) - U(\boldsymbol{\gamma}_c^0) = \sum_{j \in \bar{c}} (-1)^{\gamma_j} \left(\sum_{k \in c_0} J_{jk} - \sum_{l \in c_1} J_{jl} \right) + \sum_{j \in c_1} h_j^* - \sum_{j \in c_0} h_j^*. \tag{2.38}$$

The first part of right hand side of Equation (2.38) can be decomposed as

$$\lambda \sum_{j \in \bar{c}} (-1)^{\gamma_j} \left(\sum_{k \in c_0} J_{jk} - \sum_{l \in c_1} J_{jl} \right) + (1 - \lambda) \sum_{j \in \bar{c}} (-1)^{\gamma_j} \left(\sum_{k \in c_0} J_{jk} - \sum_{l \in c_1} J_{jl} \right).$$

With the probability of adding a node $j \in \bar{c}$ to the cluster, $p_{a,j}$, defined as (2.15),

$$\frac{1 - p_{a,j}^0}{1 - p_{a,j}^*} = \exp\left\{ -\lambda (-1)^{\gamma_j} \left(\sum_{k \in c_0} J_{jk} - \sum_{l \in c_1} J_{jl} \right) \right\}.$$

Substituting the above equation into Expression (2.37) and rearranging it, we derive the acceptance ratio for the moves in the two directions as

$$\frac{A(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)}{A(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)} = \exp\left[(1 - \lambda) \sum_{j \in \bar{c}} (-1)^{\gamma_j} \left(\sum_{k \in c_1} J_{jk} - \sum_{l \in c_0} J_{jl} \right) + \sum_{j \in c_0} h_j^* - \sum_{j \in c_1} h_j^* \right],$$

and the acceptance probability for move from $\boldsymbol{\gamma}_c^0$ to $\boldsymbol{\gamma}_c^*$ is

$$\alpha(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*) = \min\left\{ \frac{A(\boldsymbol{\gamma}_c^0 \to \boldsymbol{\gamma}_c^*)}{A(\boldsymbol{\gamma}_c^* \to \boldsymbol{\gamma}_c^0)}, 1 \right\}.$$

As well as satisfying the detailed balance, the algorithm also guarantees the ergodicity by the fact that there is always a finite chance that any spin will be chosen as the sole member of cluster of one, which is then flipped. The appropriate succession of such moves will get us from any state to any other in a finite time as ergodicity requires.

2.7.2 Proof of Theorem 2.5.1

The proof of the first part in Theorem 2.5.1 is simply algebra calculation. First define $\mathbf{y}^* = \mathbf{y} - \sum_{k\neq j} \gamma_k \mathbf{x}_k \beta_k$, and integrate out β_j and $\boldsymbol{\beta}_{\bar{j}}$ separately in following expression

$$\begin{aligned} p(\boldsymbol{\gamma}|\mathbf{y},\boldsymbol{\tau},b) &\propto \int p(\mathbf{y}|\boldsymbol{\gamma},\boldsymbol{\beta})p(\boldsymbol{\beta}|\boldsymbol{\tau},b)d\boldsymbol{\beta} \\ &\propto \int \exp\left[-\frac{1}{2}\left(\mathbf{y}^* - \gamma_j\mathbf{x}_j\beta_j\right)^2\right] p(\beta_j|\tau_j,b)d\beta_j \prod_{k\neq j} p(\beta_k|\tau_k,b)d\beta_k \\ &\propto \exp\left[\frac{\gamma_j}{2}a_j^2\left(\gamma_j + \frac{\tau_j}{b^2}\right)^{-1}\right]\left(\frac{\tau_j/b^2}{\gamma_j+\tau_j/b^2}\right)^{1/2} \cdot \xi(\gamma_j,\kappa_j,\boldsymbol{\gamma}_{\bar{j}},\boldsymbol{\tau}_{\bar{j}}), \end{aligned} \tag{2.39}$$

where $\xi(\gamma_j,\kappa_j,\boldsymbol{\gamma}_{\bar{j}},\boldsymbol{\tau}_{\bar{j}})$ is calculated by integrating out $\boldsymbol{\beta}_{\bar{j}}$:

$$\begin{aligned} \xi(\gamma_j,\kappa_j,\boldsymbol{\gamma}_{\bar{j}},\boldsymbol{\tau}_{\bar{j}}) &\propto \int \exp\left[\frac{\gamma_j}{2}\left(\gamma_j+\frac{\tau_j}{b^2}\right)^{-1}\left(\sum_{k\neq j;l\neq j}\beta_k\gamma_k\mathbf{x}_k^T\mathbf{x}_j\mathbf{x}_j^T\mathbf{x}_l\beta_l\gamma_l - 2\sum_{k\neq j}\mathbf{x}_k^T\mathbf{x}_j\mathbf{x}_j^T\mathbf{y}\beta_k\gamma_k\right)\right] \\ &\quad\times \exp\left(-\frac{\mathbf{y}^{*T}\mathbf{y}^*}{2}\right)\prod_{k\neq j}p(\beta_k|\tau_k,b)d\beta_k \\ &\propto \int \exp\left[\frac{\gamma_j}{2}\left(\gamma_j+\frac{\tau_j}{b^2}\right)^{-1}\left(\boldsymbol{\beta}_{\bar{j}}^T(\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}^T)\boldsymbol{\beta}_{\bar{j}} - 2a_j\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}^T\boldsymbol{\beta}_{\bar{j}}\right)\right] \\ &\quad\times \exp\left[-\frac{1}{2}\left(\mathbf{y} - X_{\boldsymbol{\gamma}_{\bar{j}}}\boldsymbol{\beta}_{\bar{j}}\right)^2\right]p(\boldsymbol{\beta}_{\bar{j}}|\boldsymbol{\tau}_{\bar{j}},b)d\boldsymbol{\beta}_{\bar{j}} \\ &\propto \exp\left[\frac{1}{2}\left(\mathbf{a}_{\bar{j}} - (1-\kappa_j^{\gamma_j})a_j\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}\right)^T\Omega_j^{-1}\left(\mathbf{a}_{\bar{j}} - (1-\kappa_j^{\gamma_j})a_j\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}\right)\right]|\Omega_j|^{1/2}|D_{\bar{j}}|^{1/2}, \end{aligned} \tag{2.40}$$

where $\Omega_j = [D_{\bar{j}} + X_{\boldsymbol{\gamma}_{\bar{j}}}^T X_{\boldsymbol{\gamma}_{\bar{j}}} - (1-\kappa_j)^{\gamma_j}(\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}}^T)]^{-1}$ and is easy to show it is positive definite. We also used the identity $\gamma_j / \left(\gamma_j + \frac{\tau_j}{b^2}\right) = (1-\kappa_j^{\gamma_j})$. Note if $\gamma_j = 0$, $\xi(\cdots)$ does not depend on κ_j or τ_j.

Then by definition and above expressions,

$$\begin{aligned}
\pi_j^b &= \frac{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} p(\gamma_j = 1, \boldsymbol{\gamma}_{\bar{j}} | \mathbf{y}, \boldsymbol{\tau}, b) p(\boldsymbol{\tau}) d\boldsymbol{\tau}}{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} p(\gamma_j = 0, \boldsymbol{\gamma}_{\bar{j}} || \mathbf{y}, \boldsymbol{\tau}, b) p(\boldsymbol{\tau}) d\boldsymbol{\tau}} \\
&= \int \pi_j \cdot \frac{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} \xi(\gamma_j = 1, \kappa_j, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\tau}_{\bar{j}}) p(\boldsymbol{\tau}_{\bar{j}}) d\boldsymbol{\tau}_{\bar{j}}}{\int \sum_{\boldsymbol{\gamma}_{\bar{j}}} \xi(\gamma_j = 0, \kappa_j, \boldsymbol{\gamma}_{\bar{j}}, \boldsymbol{\tau}_{\bar{j}}) p(\boldsymbol{\tau}_{\bar{j}}) d\boldsymbol{\tau}_{\bar{j}}} \cdot p(\tau_j) d\tau_j \\
&= \int \pi_j \xi_j p(\kappa_j) d\kappa_j,
\end{aligned} \tag{2.41}$$

with ξ_j defined as (2.25). It is easy to show that $\xi_j = 1$ for orthogonal design since $\mathbf{c}_{\boldsymbol{\gamma}_{\bar{j}}} = \mathbf{0}$ for all j.

The proof of the second part for π_j is trivial. Obviously, $\pi_j \to 1$ as $\kappa_j \to 1$ and $\pi_j \to 0$ as $\kappa_j \to 0$. For π_j^b, it is more convenient using $\pi_j^b = \int \pi_j \xi_j p(\tau_j) d\tau_j$, where π_j and ξ_j are measurable functions indexed by b. Both π_j and ξ_j are bounded by some positive number for all b. When $b \to 0$, $\lim_{b\to 0} \pi_j \to 1$ and $\lim_{b\to 0} \xi_j \to 1$, thus according to Lebesgue's Dominated Convergence Theorem (DCT), $\lim_{b\to 0} \pi_j^b = \lim_{b\to 0} \int \pi_j \xi_j p(\tau_j) d\tau_j = \int \lim_{b\to 0} (\pi_j \xi_j) p(\tau_j) d\tau_j = 1$. When $b \to \infty$, $\lim_{b\to\infty} \pi_j \to 0$ and $\lim_{b\to\infty} \xi_j$ equal to some finite number. Again the limit and integral commute by DCT, thus we have $\lim_{b\to\infty} \pi_j^b = \int \lim_{b\to\infty} (\pi_j \xi_j) p(\tau_j) d\tau_j = 0$.

2.7.3 Proof of Theorem 2.5.2

The existence of m_j indicates the marginal prior $p(\beta_j) = \int p(\beta_j|\tau_j)p(\tau_j)d\tau_j$ is bounded for $\beta_j \in \mathbb{R}$, which is true for Cauchy and Laplace prior. Using identity

$$(\beta_j - a_j)p(\mathbf{y}|\beta_j, \gamma_j = 1) = \mathbf{x}_j^T \frac{\partial}{\partial \mathbf{y}} p(\mathbf{y}|\beta_j, \gamma_j = 1),$$

so that

$$\begin{aligned} m_j[E(\beta_j|\mathbf{y}, \gamma_j = 1) - a_j] &= \int (\beta_j - a_j)p(\mathbf{y}|\beta_j, \gamma_j = 1)p(\beta_j)d\beta_j \\ &= \int \mathbf{x}_j^T \frac{\partial}{\partial \mathbf{y}} p(\mathbf{y}|\beta_j, \gamma_j = 1)p(\beta_j)d\beta_j \\ &= \mathbf{x}_j^T \frac{\partial}{\partial \mathbf{y}} \log m_j. \end{aligned} \tag{2.42}$$

Following the lemma given in Pericchi and Smith (1992), the interchange of the derivative and the integral is justified. The second result of Theorem 2.5.2 is straightforward by observing $m_j \propto \exp\left(-\frac{1}{2}\mathbf{y}^T\mathbf{y}\right)\pi_j^b$, thus

$$\mathbf{x}_j^T \frac{\partial}{\partial \mathbf{y}} \log m_j = \mathbf{x}_j^T \left(-\mathbf{y} + \frac{d \log \pi_j^b}{da_j} \frac{da_j}{d\mathbf{y}} \right) = -a_j + \frac{d \log \pi_j^b}{da_j}.$$

For horseshoe prior, $p(\beta_j)$ is not bounded. However, using the technique introduced in Carvalho and Polson (2010) by defining $m_j^* = \int p(\mathbf{y}|\beta_j, \gamma_j = 1)p(\beta_j|\tau_j)p(\tau_j)\tau_j^{-1}d\beta_j d\tau_j$, it can be shown

$$E(\beta_j|\mathbf{y}, \gamma_j = 1) = -\frac{m_j^*}{m_j}\mathbf{x}_j^T \frac{\partial}{\partial \mathbf{y}} \log m_j^*,$$

and similar arguments then follow. For horseshoe prior, it also can be shown

$$\frac{\pi_j^b}{\pi_j^{b*}} \frac{d}{da_j} \log \pi_j^b = \frac{1}{\pi_j^{b*}} \frac{d}{da_j} \left(\int \pi_j(\tau_j^{-1} + 1)p(\tau_j)d\tau_j \right) - E(\kappa_j|\mathbf{y}, \gamma_j = 1)a_j$$

where $\pi_j^{b^*} = \int \pi_j \tau_j^{-1} p(\tau_j) d\tau_j$.

2.7.4 Proof of Theorem 2.5.3

It is more convenient to use following equivalent representation of π_j^b

$$\pi_j^b = \int \exp\left[\frac{a_j^2}{2}(1+\tau_j)^{-1}\right] [\tau_j/(1+\tau_j)]^{\frac{1}{2}} \, p(\tau_j|b) d\tau_j,$$

where $p(\tau_j|b)$ is corresponding prior of τ_j given b such that $\pi_j^b = \int p(\beta_j|b)p(\tau_j)d\tau_j = \int p(\beta_j)p(\tau_j|b)d\tau_j$. Then the similar condition for $p(\tau_j|b)$ can be derived from the condition for $p(\tau_j)$ in Theorem 2.5.3, i.e.,

$$p(\sigma_j^2|b) \sim (\sigma_j^2)^{\alpha-1} \exp\left(-\frac{\lambda}{b^2}\sigma_j^2\right) L^b(\sigma_j^2) d\sigma_j^2, \text{ as } \sigma^2 \to \infty, \tag{2.43}$$

where L^b is the slowly varying function conditioning on parameter b.

Then the marginal odds π_j^b can be expressed as

$$\pi_j^b \propto \exp\left(\frac{a_j^2}{2}\right) m^b(a_j) = \exp\left(\frac{a_j^2}{2}\right) \int \omega_j^{-1} \exp\left(-\frac{a_j^2}{\omega_j^2}\right) p(\omega_j^2|b) d\omega_j^2,$$

where $\omega_j^2 = 1 + \sigma_j^2$, and the integral, $m^b(a_j)$, is a scale mixture of normals. Now the proof is similar to Polson and Scott (2011). If prior $p(\sigma_j^2|b)$ satisfies the conditions defined in (2.43), so does $p(\omega_j^2|b)$ satisfy similar conditions, i.e.,

$$p(\omega_j^2|b) \sim (\omega_j^2)^{\alpha-1} \exp\left(-\frac{\lambda}{b^2}\omega_j^2\right) L^b\left(\omega_j^2\right) \text{ as } \omega_j^2 \to \infty.$$

Then following Theorem 6.1 of Barndorff-Nielsen et al. (1982), as $a_j \to \infty$, $m^b(a_j)$ can be

approximated as

$$m^b(a_j) \sim \begin{cases} |a_j|^{2\alpha-1} L^b\left(a_j^2\right) & if \;\; \lambda = 0 \\ |a_j|^{\alpha-1} \exp\left(-\sqrt{\frac{2\lambda}{b^2}}|a_j|\right) L^b\left(|a_j|\right) & if \;\; \lambda > 0, \end{cases} \tag{2.44}$$

as $a_j \to \infty$. The results in Theorem 2.5.3 follow by taking derivative respect to $|a_j|$ and b respectively.

2.7.5 The Calculation of π_j^b

For orthogonal designs, π_j^b with Laplace prior can be integrated out directly from (2.26)

$$\begin{aligned} \pi_j^b &= \int_0^1 \kappa_j^{\frac{1}{2}} \exp\left[\frac{a_j^2}{2}(1-\kappa_j)\right] \cdot \frac{1}{2b^2}\kappa_j^{-2} \exp\left(-\frac{1-\kappa_j}{2b^2\kappa_j}\right) d\kappa_j \\ &= \frac{1}{2b^2}\left(\frac{2\pi}{\lambda}\right)^{1/2} \exp\left(\frac{1}{2b^2} + \frac{a_j^2}{2} - \frac{\sqrt{a_j^2}}{b}\right) \int_0^1 \left(\frac{\lambda}{2\pi}\right)^{1/2} \kappa_j^{-3/2} \exp\left[-\frac{\lambda(\kappa_j-\mu)^2}{2\mu^2\kappa_j}\right] d\kappa_j, \end{aligned} \tag{2.45}$$

where $\lambda = 1/b^2$ and $\mu = \sqrt{1/(b^2 a_j^2)}$, and the expression in the integral is the CDF of inverse Gaussian distribution. Borrowing the expression of the CDF of the inverse Gaussian, we then integrated out the integral to get Expression (2.31).

π_j^b with horseshoe prior for orthogonal design can also be derived directly from (2.26):

$$\begin{aligned} \pi_j^b &= \int_0^1 \kappa_j^{\frac{1}{2}} \exp\left[\frac{a_j^2}{2}(1-\kappa_j)\right] \cdot \frac{b}{\pi}\kappa_j^{-1/2}(1-\kappa_j)^{-1/2}(1-\kappa_j+b^2\kappa_j)^{-1} d\kappa_j \\ &= \frac{1}{\pi b}\exp\left(\frac{a_j^2}{2}\right) \int_0^1 \kappa_j^{1-1}(1-\kappa_j)^{1/2-1}\left[(1-b^{-2})\kappa_j + b^{-2}\right]^{-1} \exp\left(-\frac{a_j^2}{2}\kappa_j\right) d\kappa_j, \end{aligned} \tag{2.46}$$

where the expression in the integral is the transformation of the hypergeometric inverted-

beta distribution which was shown to be represented by degenerate hypergeometric functions (Gordy, 1998; Polson and Scott, 2010), thus we can follow Polson and Scott (2010) to express it as (2.32).

2.8 Simulation Study

2.8.1 Case One: Comparison of Three Priors

The first simulation study will examine a simple linear regression model with a general form

$$y = \sum_{j \in V^*} \beta_j x_j + \epsilon, \tag{2.47}$$

where $V^* = \{2, 3, 5, 10\}$, $n = 50$ is the sample size, $p = 100$, $x_j \sim N(0, 1), j = 1, ..., p$ and $\epsilon \sim N(0, 1)$. Particularly, we will consider one large signal set and one small signal set: $\{\beta_2, \beta_3, \beta_5, \beta_{10}\} = \{-4, 2, -1, 2.5\}$ and $\{-0.9, 0.7, -0.6, 0.8\}$.

In this simulation, we performed the single-site updating with a total of 6000 iterations for each settings, discarding the first 2000 iterations as burn-in, then calculated the average γ_j's over the remaining $N = 4000$ iterations as the marginal selection probabilities. Figure 2.6 plots the marginal selection probability of all variables against the global shrinkage parameter b. For large signals, as shown in the upper row of Figure 2.6, the horseshoe and Cauchy priors perform similarly and show the robustness of large signals. While the selection probability for the true signals remains near 1 for b large, it quickly drops to 0.5 at smaller values. The horseshoe prior shows even more robustness than Cauchy prior on the large b region. Both priors have a wide window of b in which the true signals are well separated from the noise signals. On the other hand, Laplace prior

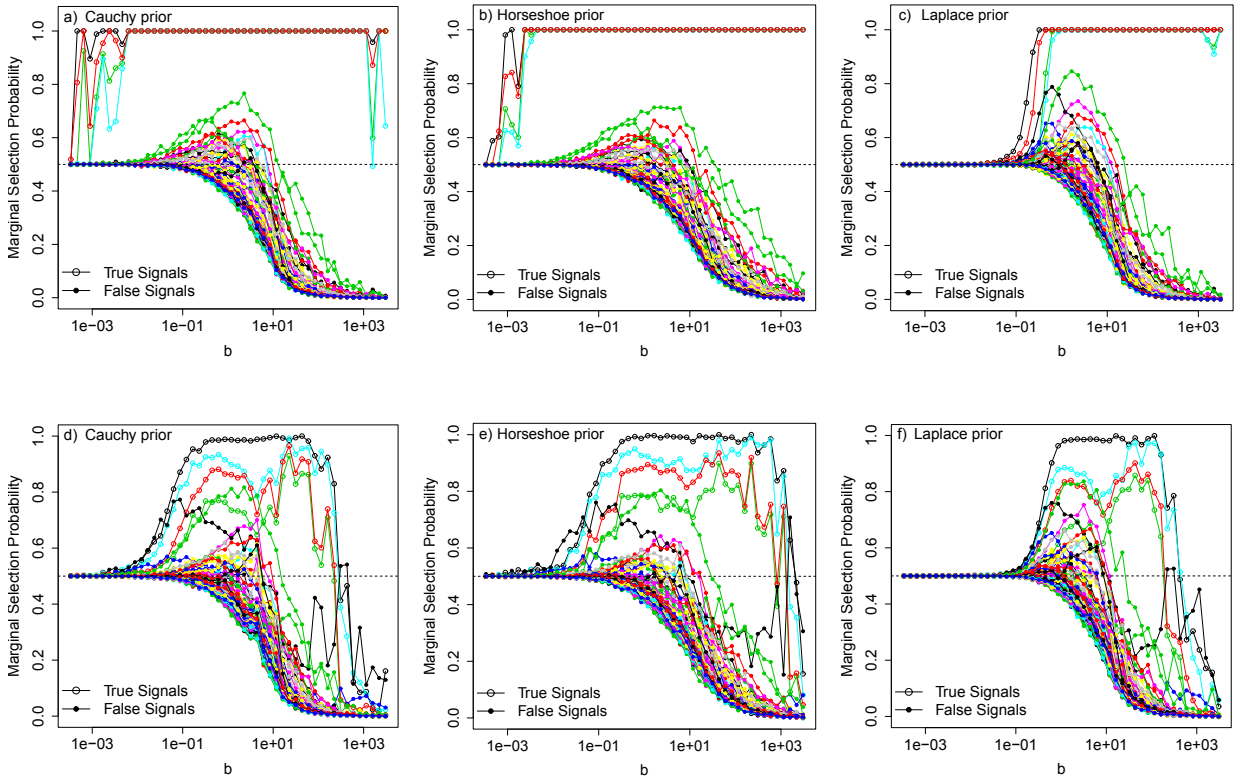

Figure 2.6: The profile curves of the selection probability of the simulation model (2.47) with different priors for large signal setting (a-c), and small signal setting (d-f).

does not exhibit such robustness for large signals: as the shrinkage parameter $b \to 0$, the selection probability for the true signals drops to 0.5 very early. Around $b = 0.1$, all signal probabilities reach the 0.5 line under Laplace prior. In the large b region, the Laplace prior seems perform a little better than the Cauchy prior. We can also see that the conclusions drawn from the simulation about the performance of the three priors agrees with the exact calculation of the marginal selection probabilities from Figure 2.3.

The bottom row of Figure 2.6 displays the simulation results for small signals. In general, the window for true signals maintaining a high selection probability narrows for all three priors. The selection probability of the true signal for all priors starts to drop

to 0.5 around $b = 0.1$, and around $b = 1000$, it drops to 0. However, the drop rate on both ends of b is different for three priors, resulting in different widths for the working widow of b. The horseshoe prior has the widest window, while the Laplace prior has the narrowest. Again, the conclusion we make is exactly the same as from the calculation in Figure 2.3.

We may note some other observations from Figure 2.6, especially for small signals. First we see that around $b = 10$, the selection probability of true signals first drops a little and then increases. This happens right above the peak of the selection probability of the noise, which indicates a potential interaction between the noise and the true signals. This is easy to see since when sample size is small, the correlation between a true signal and noise is large. The parameter ξ_j in (2.23) is therefore large enough to distort the profile curve. The second observation from Figure 2.6 is that, although the overall performance of three priors differs in the width of the working window, we can select the best value of b so that for all priors the noise and signals are well distinguishable. For instance, by choosing b between 10 and 1000, all priors will have a cut-off probability of 0.5 for selecting true signals and separating them from noise.

Based on this first case study, we find that the horseshoe prior has the largest working window. Hence the rest of the dissertation employs only horseshoe prior unless otherwise stated.

2.8.2 Case Two: Three Regions of Global Shrinkage Parameter b

In this simulation we will examine the case where p is large, say $p = 1000$ or $p = 500$. The linear model still has the form (2.47) with $x_j \sim N(0, 1), j = 1, ..., p$ and $\epsilon \sim N(0, 1)$, but we consider the following specific models and settings

Model I: A $p = 1000, n = 200$, $\beta_j = 0.8$ if j is odd; $\beta_j = 1.0$ if j is even.

$V^* = \{31, 91, ..., 931\} \bigcup \{60, 120, ..., 960\}$

B $p = 1000, n = 500$, $\beta_j = 0.8$ if j is odd; $\beta_j = 1.0$ if j is even.

$V^* = \{31, 91, ..., 931\} \bigcup \{60, 120, ..., 960\}$

Model II: A $p = 500, n = 100$, $\beta_j = -0.8$ if j is odd; $\beta_j = 0.8$ if j is even.

$V^* = \{31, 91, ..., 451\} \bigcup \{60, 120, ..., 480\}$

B $p = 500, n = 500$, $\beta_j = -0.8$ if j is odd; $\beta_j = 0.8$ if j is even.

$V^* = \{31, 91, ..., 451\} \bigcup \{60, 120, ..., 480\}$

Thus for Model I and II, the number of true β_j's is the cardinality $|V^*| = 32$ and $|V^*| = 16$ respectively. Under each setting, we performed the single-site updates with a total of 8000 iterations and discarded the first 3000 as burn-in, thus calculating the average γ_j's over $N = 5000$ iterations as the marginal selection probabilities.

Figure 2.7 (a-b) and Figure 2.8 (a-b) plot the marginal selection probability of all variables against b for all settings. These visualizations give us a general idea of the effects of global shrinkage, and they allow us the opportunity to compare suitable working windows of b with changes in across the different models. For example, in Figure 2.7 (a), the working window ranges from $b \approx 0.01$ to $b \approx 2000$, while in Figure 2.7 (b) it is from $b \approx 0.001$ to $b \approx 3000$. Within this window, we can see that for both setting A and B, b can be further divided into three regions I, II and III. In Figure 2.7 (a) Region I represents the high temperature or large shrinkage area, where b is around 0.1 or smaller. Region II is a moderate shrinkage area with b between 1 and 100. The last region, Region III, is around 1000, and it varies from several hundreds to several thousands. The widths of these three regions also change depending on the strength of the signals. We can see that in most cases, the signals are well separated from the noise in Region I and III. On the other hand, in

Region II, if the signal is not large enough or the sample size n is small, some oscillations or strong interactions occur between signal and noise on the profile curves, resulting in a total mix-up of noise and signal. Thus if we were to suggest the appropriate value of b, Region II must be avoided.

Another interesting observation from this simulation is that while in general, both Region I and III both can be used to separate signals from noise, the performance of the MCMC sampling routine may have different properties in these two regions. We have discussed that Region I has the large shrinkage property, but it may not exhibit sparse consistency. Fan and Li (2001) shows that in general, sparse consistency requires $\lambda \to 0$ or small shrinkage ($\lambda \propto b^{-2}$ is the shrinkage parameter in their paper). Region III, representing small shrinkage area, hence may maintain sparse consistency while Region I does not. This property is shown in Figure 2.8 (a) for Model II A, where the best value of b is in Region III with which all signals are distinguishable from the noise. On the other hand the noise and signals mix up when b is in Region I. This can be seen more clearly in Figure 2.8 (c-d) for two specific b values. For $b = 223$, most of the true signals have selection probability 1 and are distinguished from the noise, while for $b = 0.23$, some of the true signals have a smaller selection probability than some noise.

How one determines the best setting for parameter b is an interesting issue. Some authors suggest assigning another prior for b, such as the horseshoe prior in Polson and Scott (2011). Unfortunately, because b is a global parameter, when p is large, the posterior distribution of b will be forced to some value that is not in Region III where we would prefer. Therefore in this dissertation, we will not consider assigning a prior for b, but instead consider it as a tuning parameter. A practical way to select b is to try several b values and choose the one that shows the largest gap in selection probability. It is useful to keep in mind that b is usually between ten to several thousands.

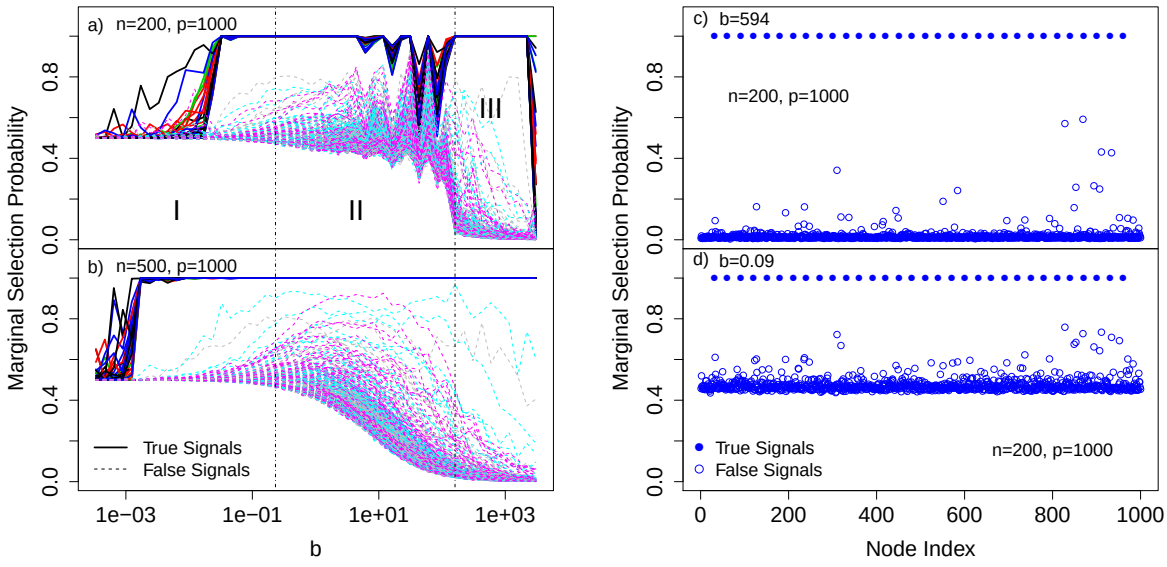

Figure 2.7: The profile curves of the selection probability of Model I A and B (a-b). Selection probability at two b values for Model I A (c-d).

2.8.3 Case Three: Comparison of Cluster and Single-site Algorithm

As discussed in Section 2.6, assigning the scale normal mixture prior for β_j's with shrinkage parameter b is also a tempering algorithm, which means our model has built-in advantage regarding the mixing issue. Hence, there may be little to gain in trying to improve mixing with a cluster algorithm. We will show that the performances of the cluster and single-site algorithms both are b-dependent. In certain regions of b, one algorithm may outperform the other.

To demonstrate this, we consider the simple simulation with the same model as (2.47) with large signals $\{\beta_2, \beta_3, \beta_5, \beta_{10}\} = \{-4, 2, -1, 2.5\}$, $n = 200$ and we vary p from 50 to 1500. We run the simulation with four representative b's, two are large and two are small, so that we may compare the different behavior of the two algorithms across different shrinkage parameters.

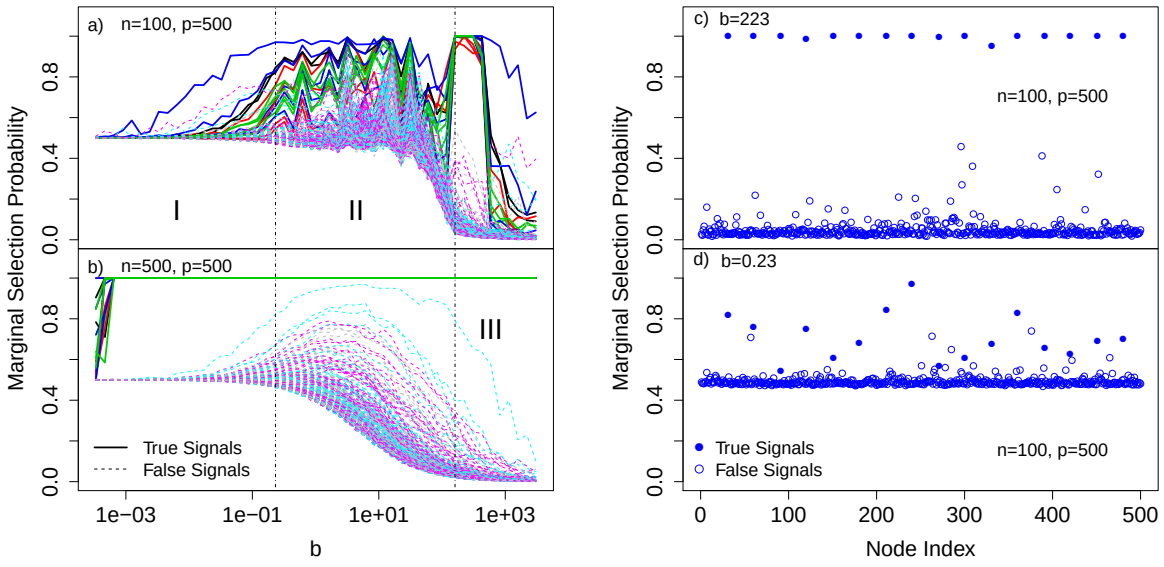

Figure 2.8: The profile curves of the selection probability of Model II A and B (a-b). Selection probability at two b values for Model II A (c-d).

To measure the mixing or correlation time, it is convenient to define the "magnetization", $M^{(i)}$, which represents the average value of the binary random variable γ_j's at ith sweep of the MCMC iteration.

$$M^{(i)} = \frac{1}{p}\sum_{j=1}^{p}\gamma_j^{(i)}.$$

Thus the mixing time of the MCMC sampler can be measured using the time-delayed autocorrelation function (ACF) of the Monte Carlo chain of "magnetization",

$$C(t) = \frac{\sum_{i=1}^{N-t}(M^{(i)} - \bar{M})(M^{(i+t)} - \bar{M})}{\sum_{i=1}^{N}(M^{(i)} - \bar{M})^2},$$

where t is the lag or the iteration time from the origin, measured in Monte Carlo sweeps (MCS), and $\bar{M}$ is the average magnetization over total N iterations. We assume the absolute value of $C(t)$ decays exponentially, i.e., $|C(t)| \approx C_0 \exp(-t/\tau)$, where C_0 is some positive constant, and τ is defined as the exponential correlation time. Therefore, we can

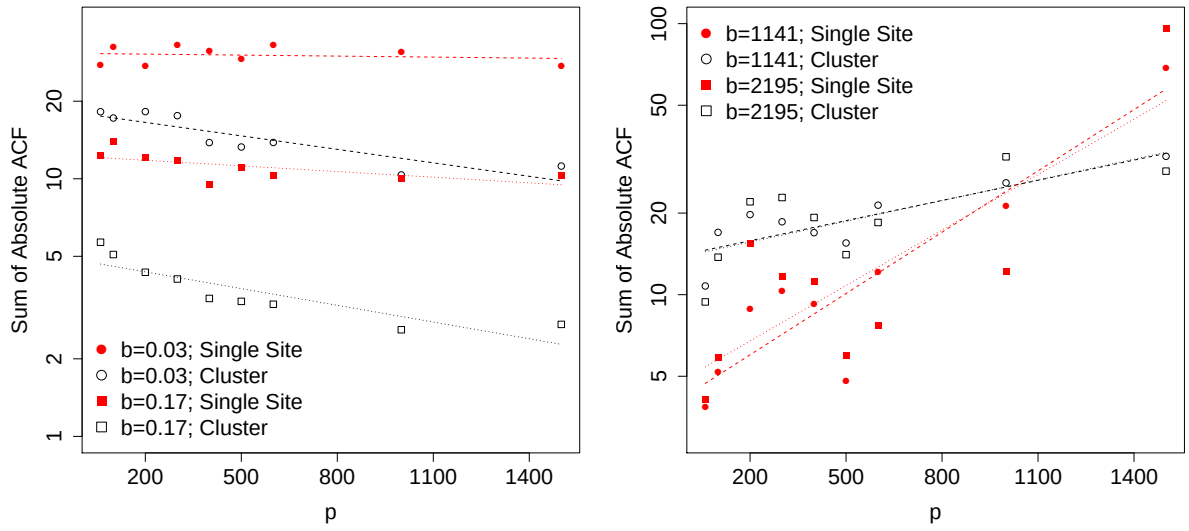

Figure 2.9: The sum of absolute ACF against variable number p for cluster algorithm and single-site algorithm at different b values.

use τ to measure how fast the chain converges or mixes. The smaller the τ, the faster the system mixes. Another way to measure the mixing time is simply using the summation of the autocorrelation time, $\sum_{t=0}^{L} |C(t)|$, where L is the maximum lag calculated.

For each p we performed 15000 iterations or sweeps for each setting and discarded the first 5000. From the remaining $N = 10000$ sweeps we calculated the autocorrelation function $C(t)$ up to $L = 100$ lags. Figure 2.9 shows the summation of absolute ACF time against the nodes size p for $b = 0.03, 0.17, 1141$ and 2195.

From Figure 2.9 we can see the differences in behavior of the cluster and single-site algorithms. In the large shrinkage region, $b = 0.03$ or 0.17, the cluster algorithm has mixing time uniformly smaller than the single-site algorithm. Note that as the node size increases, the mixing time for the entire algorithm first decreases slightly and then stabilizes. It may increases as p goes larger still. This profile is not yet well understood. Nevertheless, in this region, we can conclude that the cluster algorithm uniformly outperforms

the single-site algorithm in terms of fast mixing time, and the mixing time of the cluster algorithm is at least two times shorter.

In the small shrinkage region where $b = 1141$ and 2195, as shown in Figure 2.9 (b), we see different characteristics. First, the measured mixing time is much noisier than in Figure 2.9 (a), but the trend against p is clear. Second, unlike in the large shrinkage area, here we see that for both algorithms the mixing time increases as p increases. Furthermore, when p is small, the single-site algorithm has a shorter mixing time, which slows down very fast as p increases. For example, when $p = 60$, the summation of autocorrelation function is only several MCS, but reaches almost 100 MCS when p is large than 1500, which translates to an extremely slow MCMC process. On the other hand, although the cluster algorithm is about two times slower when p is small, also slowing down with increasing p, the mixing time increases at a smaller rate and reaches no more than 50 when $p = 1500$.

Hence in general, we find that the cluster algorithm outperforms single-site algorithm in terms of mixing time. However, which algorithm should be used depends on the data. The single-site algorithm is much less time consuming since the cluster algorithm spends time in forming the cluster. The overall computational time for the cluster algorithm is high when $p > 1000$. Additionally, in many situations, the mixing time may not be much worse for the single-site algorithm. Thus we prefer using the single-site algorithm to achieve the results quickly.

2.8.4 Case Four: Bayesian Sparse Additive Model

In this section, we demonstrate variable selection in the following Bayesian sparse additive model:

$$y = f_1(x_1) + f_2(x_2) + f_3(x_3) + f_4(x_4) + \epsilon, \tag{2.48}$$

where $x_j = (w_j + tu)/(1+t), j = 1, ..., p$ and $w_1, ...w_p$ and u are iid from Uniform (0,1), and $\epsilon \sim N(0, 1.74)$. Therefore Corr$(\mathbf{x}_i, \mathbf{x}_j) = t^2/(1+t^2)$ for $i \neq j$. We consider $t = 0$ and $t = 1$. The later gives the correlation between two predictors around 0.5. This simulation is similar to Example 1 in Lin and Zhang (2006) but with $p = 10, 80$ and 150. We also consider sample size of $n = 100$. Functions f_j's have following forms.

$$\begin{aligned}
&f_1(x) = x, \\
&f_2(x) = (2x-1)^2, \\
&f_3(x) = \sin(2\pi x)/[2 - \sin(2\pi x)], \\
&f_4(x) = 0.1\sin(2\pi x) + 0.2\cos(2\pi x) + 0.3\sin^2(2\pi x) + 0.4\cos^3(2\pi x) + 0.5\sin^3(2\pi x).
\end{aligned} \tag{2.49}$$

As shown in Appendix A, for each $\mathbf{x}_j$, the LS basis employs two precision parameters $\tau_{ej} = \sigma_{ej}^{-2}$ and $\tau_{dj} = \sigma_{dj}^{-2}$. We treat the whole set of $\{\tau_{ej}, \tau_{dj} : j = 1, ...p\}$ independent parameters. Similarly, we can still assign $G(1/2, 1/2)$, $IG(1, 1/2)$, or $C^+(0, 1)$ as their priors. However, since $\boldsymbol{\beta}_j$ is the $M_j \times 1$ vector, and for each node we have two variance components, the marginal prior for β_j given b is no longer a simple Cauchy, Laplace or horseshoe prior, but will share some properties with its counterpart in the linear parametric model.

In this simulation, we employ the independent $G(1/2, 1/2)$ prior for each τ_{ej} and τ_{dj} only. For the number of knots of the LS basis, we may consider the case where each predictor has a different number of knots, but it turns out that a fixed number for all M_j's, say $M_j = 6$, will give sufficiently good results. Therefore, we fix $M_j = 6, j = 1, ..., p$ in this

simulation. In total, we use 6000 iterations with the single-site algorithm, discarding the first 2000 for all settings.

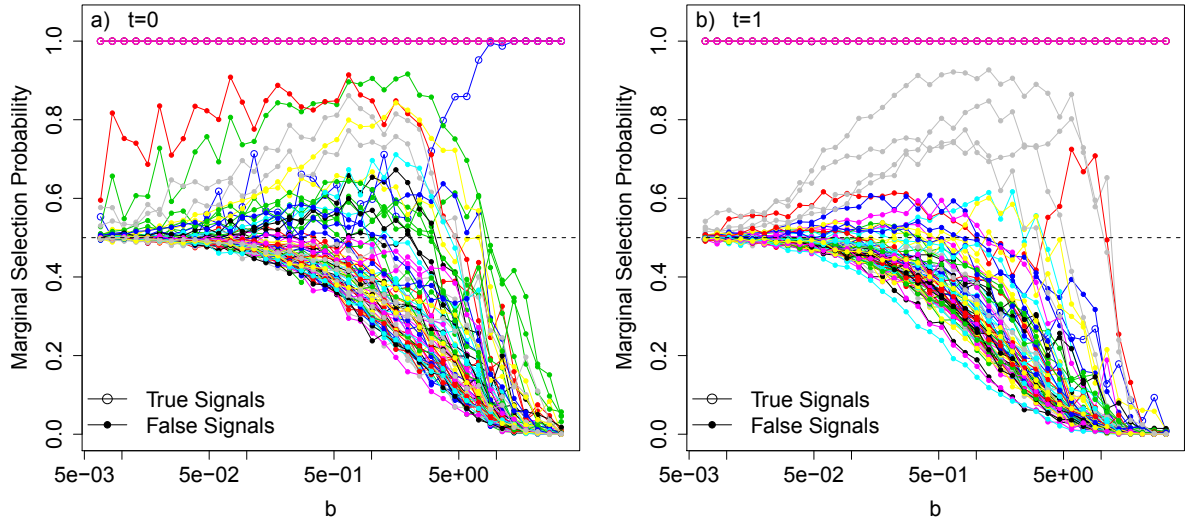

Figure 2.10: The profile curves of the selection probability of the simulation model (2.48) with $p = 80, n = 100$ for independent setting (t=0) (a), and correlated setting (t=1) (b).

Table 2.2: Simulation results of the sparse additive model (2.48) for 500 runs.

	t	p	FP-rate	FN-rate	MS	f_1SE	f_2SE	f_3SE	f_4SE
BIGM	0	10	0.00(0.02)	0.00(0.03)	3.99(0.17)	0.07(0.05)	0.16(0.06)	0.18(0.08)	0.74(0.27)
		80	0.00(0.01)	0.00(0.03)	4.16(0.48)	0.07(0.05)	0.15(0.06)	0.18(0.08)	0.70(0.27)
		150	0.00(0.02)	0.01(0.04)	4.81(5.52)	0.08(0.09)	0.16(0.07)	0.18(0.11)	0.73(0.44)
	1	10	0.00(0.01)	0.09(0.10)	3.56(0.54)	0.08(0.07)	0.18(0.09)	0.16(0.08)	0.79(0.40)
		80	0.00(0.01)	0.09(0.11)	3.68(0.66)	0.09(0.08)	0.18(0.08)	0.16(0.07)	0.77(0.39)
		150	0.01(0.03)	0.11(0.11)	4.48(7.17)	0.10(0.10)	0.18(0.09)	0.18(0.10)	0.80(0.40)
COSSO	0	10	0.00(0.01)	0.00(0.00)	4.00(0.06)	0.07(0.04)	0.05(0.04)	0.11(0.06)	0.32(0.13)
		80	0.07(0.08)	0.18(0.07)	9.85(11.2)	0.17(0.28)	0.79(0.11)	1.55(0.32)	5.28(0.58)
	1	10	0.01(0.03)	0.04(0.08)	3.86(0.48)	0.07(0.07)	0.26(0.10)	0.14(0.10)	2.00(1.00)
		80	0.10(0.09)	0.19(0.10)	12.5(14.0)	0.36(0.40)	0.37(0.16)	1.05(0.44)	4.67(0.54)

Figure 2.10 (a-b) show us how the selection probabilities changes for a range of b with $t = 0$ and $t = 1$ for the simulation setting $p = 80$. Note that in Figure 2.10 (a), one true signal is buried in the noise until $b = 1$, while the same signal is always mixed with noise in Figure 2.10 (b). As shown in Figure 2.10, when $b \approx 26$ all false signals go to 0 and we

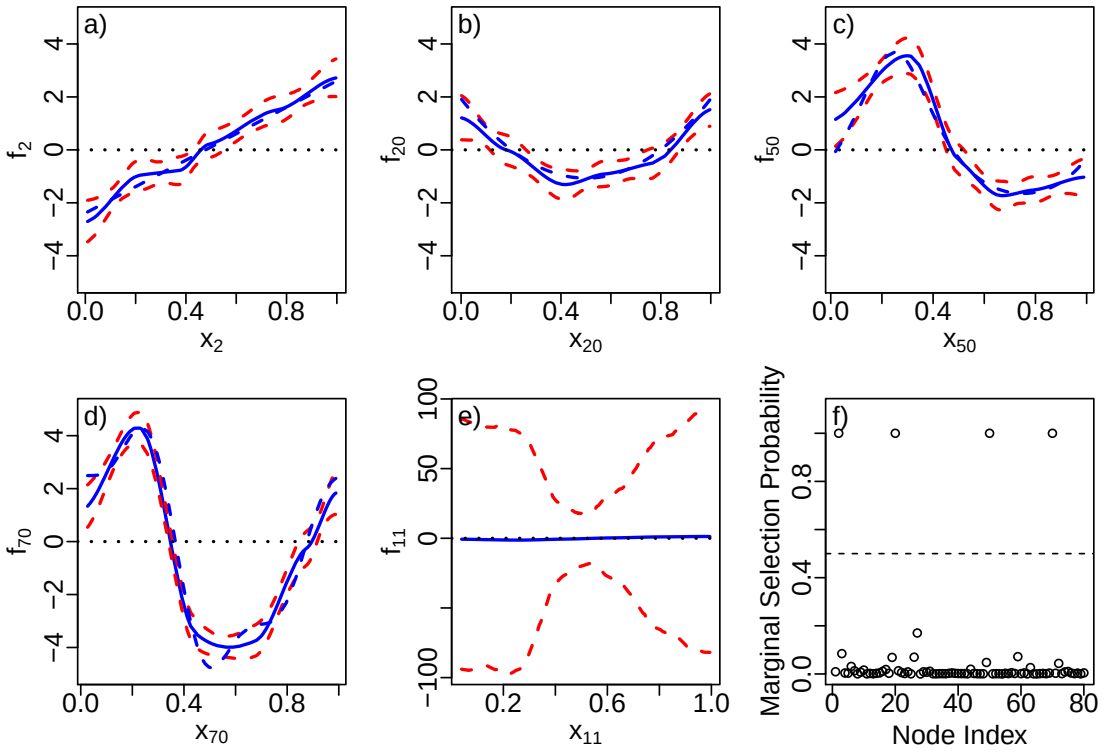

Figure 2.11: True function f_j's (blue dashed lines) and estimated function $\hat{f}_j$'s (blue solid lines) with 95% credible interval (red dashed lines) for the 4 true nodes (a-d) and a noise node (e) of a run of the simulation model (2.48) with independent setting $t = 0$ and $p = 80$. The marginal selection probability at $b = 26$ (f). Note we reordered the first 4 true nodes number to (2, 20, 50, 70) for a better view.

achieve the largest gap between signals and the noise.

One feature of our BIGM is its ability to simultaneously select the important variables and estimate the selected function components. We show the four true functions and one noise function, the corresponding estimated functions, and the selection probability for all nodes in the case where $b = 26$ and $p = 80$ for $t = 0$ and $t = 1$ in Figure 2.11 and Figure 2.12 respectively. Based on the LS basis, the function components estimated are always centered, so we also center the true functions. In this simulation run, the four true nodes are selected exactly, and their estimated functions are calculated by $\hat{f}_j = Z_j E(\boldsymbol{\beta}_j|\gamma_j = 1)$, where the expectation is based on the $N = 4000$ iterations, and the 95% credible intervals

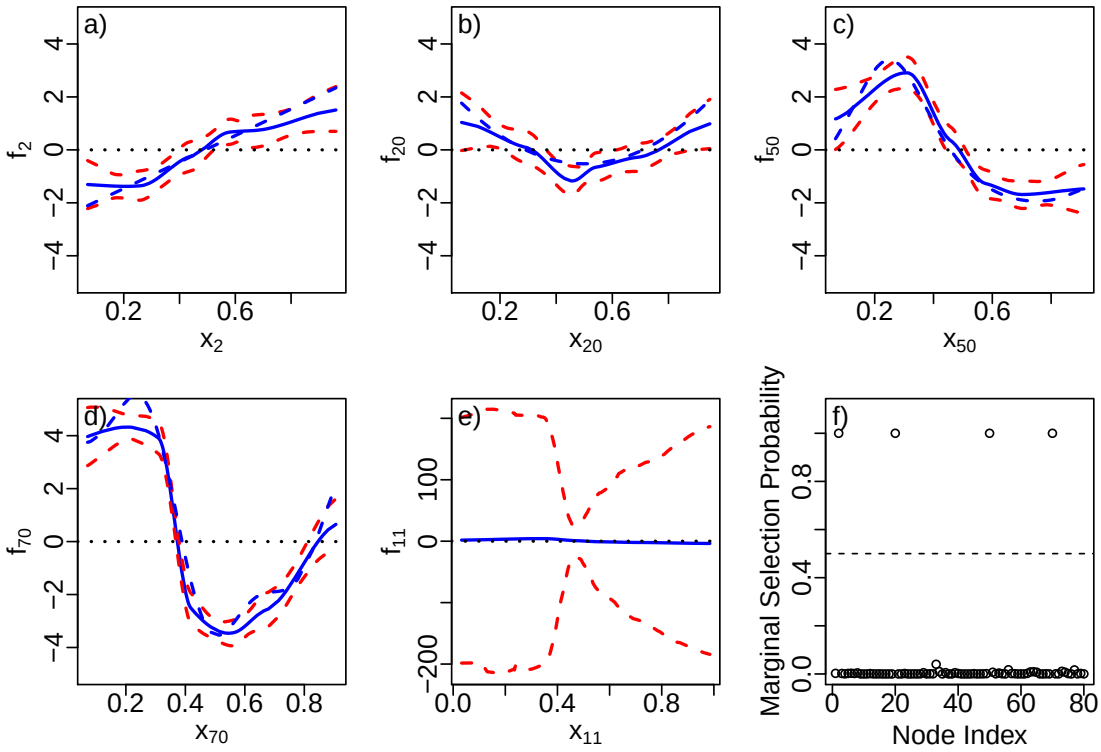

Figure 2.12: True function f_j's (blue dashed lines) and estimated function $\hat{f}_j$'s (blue solid lines) with 95% credible interval (red dashed lines) for the 4 true nodes (a-d) and a noise node (e) of a run of simulation model (2.48) with independent setting $t = 1$ and $p = 80$. The marginal selection probability at $b = 26$ (f). Note we reordered the first 4 true nodes number to (2, 20, 50, 70) for a better view.

are plotted as well.

As shown in Figure 2.11 and 2.12, for both $t = 0$ and $t = 1$ the estimated functions are very close to the true functions. Note that for the noise function, f_{11}, the selection probability is close to zero, and thus the estimated function is calculated by $\hat{f}_j = Z_j E(\boldsymbol{\beta}_j)$, an expectation over all N iterations. This is why we see a very wide credible interval for f_{11}, because when $\gamma_j = 0$ the posterior of $\boldsymbol{\beta}_j$ is a multivariate normal with large variance. Also note that to have a better view of the selection probability, we reordered the nodes such that the 4 true nodes are $(2, 20, 50, 70)$.

To further examine our method's performance in variable selection and its estimation accuracy, 500 simulation runs have been employed for $p = 10, 80$ and 150 respectively. We calculated seven statistics: "False Positive Rate (FP-rate)", "False Negative Rate (FN-rate)", "Model Size (MS)", and "Squared Error (SE)" of the 4 true functions, where FP-rate $= \frac{\#False\,Positive}{\#False\,Positive+\#True\,Negative}$, FN-rate $= \frac{\#False\,Negative}{\#False\,Negative+\#True\,Positive}$, and SE $= \sum_i^n (f_{j,i} - \hat{f}_{j,i})^2/n$. The estimated function is calculated by $\hat{f}_j = Z_j E(\boldsymbol{\beta}_j|\gamma_j = 1), j \in \{\text{true nodes}\}$. Since it can happen that $p(\gamma_j = 1|\mathbf{y}) = 0$ for any true function components, we simply estimate f_j by $\hat{f}_j = 0$ for the 4 true nodes if $p(\gamma_j = 1|\mathbf{y}) = 0$ in each run. The SE Statistics can be used to assess the accuracy of the estimation of the nonlinear function f_j because the smaller the SE, the closer the estimation $\hat{f}_j$ is to the true function f_j. The average and standard deviation of those statistics over 500 runs are reported in Table 2.2 and compared with the Component Selection and Smoothing Operator (COSSO) (Lin and Zhang, 2006).

As shown in Table 2.2, the results for our method is rather robust to p. For each t, all statistics are similar across different p's with the exception of a slight increase in the mean and standard deviation of those statistics. For different t, our method is also pretty robust, except for the increase in the values of FN-rate and SE's. On the other hand, we see that COSSO only performs well for small p. When $p = 80$ (COSSO cannot work for the case of $n < p$, so we get no result for $p = 150$), all of the statistics increase. The SE's are especially large for the four true function components, which means that COSSO cannot estimate those function components correctly. In general, we see that our method works very well for BSAM even for the cases of large p and high correlation in both variable selection and function component estimation.

2.8.5 Case Five: Linear Chain Prior

Again, we consider the same form of Model (2.47) but we set $p = 100, n = 100$, $\beta_j = 0.4$ if j is odd, $\beta_j = 0.8$ if j is even, and $V^* = \{1, 2, ..., 15\}$. This example is special in the sense that its true predictor set S and false signal set $\bar{S}$ both are continuous in their node index. Obviously, the simplest prior network information is a linear chain: any node's two neighbors are most likely to aligned with this node. While this is not true for neighbored nodes 15 and 16, the discontinuity has a small effect on the system as a whole. Keeping this in mind, we consider the linear chain prior for the nodes. $W = \{w\lambda_{ij}\}$, $\lambda_{ij} = 1$ for $|i - j| \leq 1$ and $\lambda_{ij} = 0$ otherwise. In order to have an exchangeable prior, the two boundary nodes 1 and p can be treated as neighbors, i.e., $\lambda_{ij} = 1$ if $|i - j| = p - 1$. To fully use this prior information, we employ the cluster algorithm and use the adjacency matrix $\Lambda = \lambda_{ij}$ to form the cluster.

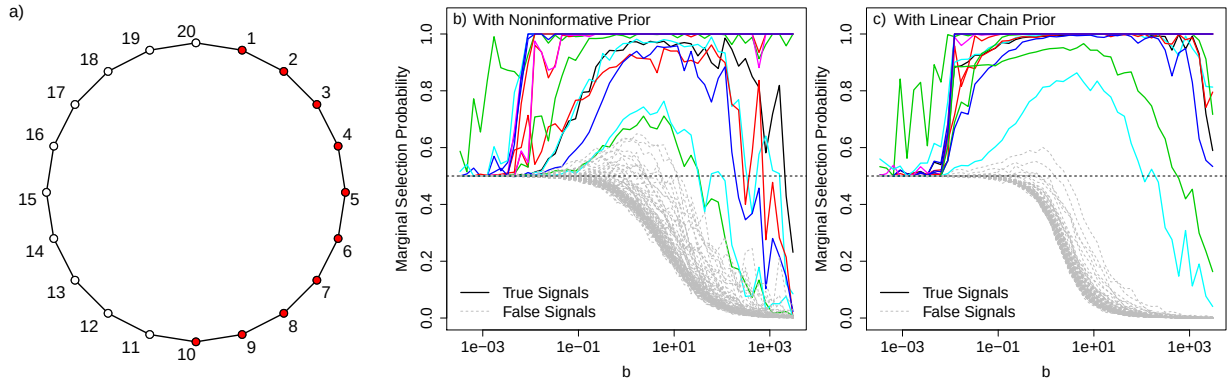

Figure 2.13: The graph of a linear chain prior with 20 nodes with 1 through 10 nodes "in" (a). The profile curves of the selection probability of case four model calculated by the cluster algorithm with noninformative prior (b), and with the linear chain prior for γ (c).

We also compare the results with the noninformative prior (employed by the single-site algorithm). Figure 2.13 (a) shows an example graph of a linear chain with $p = 20$ nodes. Note how the two end nodes are connected to form a loop and ensure exchangeability. Figure 2.13 (b-c) are the selection probability profile plot with noninformative and

linear chain priors. For each b in both plots, the total 6000 iterations were employed with the first 2000 as burn-in. For the linear chain prior we take $w = \Phi(\log(b))$ where Φ is the standard normal CDF such that the interaction strength vanishes for large shrinkage and maintains at 1 for small shrinkage. The difference of two plots is obvious: with the non-informative prior, the two true signals are very close to the noise for the entire range of b, making them hard to distinguish from the false signals, while with the linear chain prior, we see for a large range of b such that the true signals are well distinguishable from the noise.

2.9 Real Data Analysis

2.9.1 Ozone Data

As an illustration of BSAM implemented by the BIGM, we consider the ozone data analyzed by Breiman and Friedman (1985) and Lin and Zhang (2006). The ozone data are available in the R packages `cosso` and `gss`. The Ozone dataset consists of eight explanatory meteorological variables that were recorded in the Los Angeles area for 330 days in 1976. The response variable of interest is the daily maximum hourly average ozone concentration. The eight explanatory variables are Height (Hgt), Wind Speed (WS), Humidity (Hum), Temperature (Temp), Inversion Base Height (InvHt) , Pressure (Press), Inversion Base Temperature (InvTp), and Visibility (Vis). All predictors were standardized and the response was log-transformed to normalize its distribution.

We applied the Bayesian graph model described in Section 2.6 with $M_j = 6$ for all predictors. A total of 20000 iterations were employed using the single-site algorithm, and half of them were discarded as burn in. By quickly examining a small number of b, we

saw the selection probability profile curves are all well defined (not shown here). We conclude that it is appropriate to choose a modest shrinkage, $b = 1.6$, in this case such that all selected predictors reach their highest selection probability. At this value of b, only two predictors have selection probability less than 0.5.

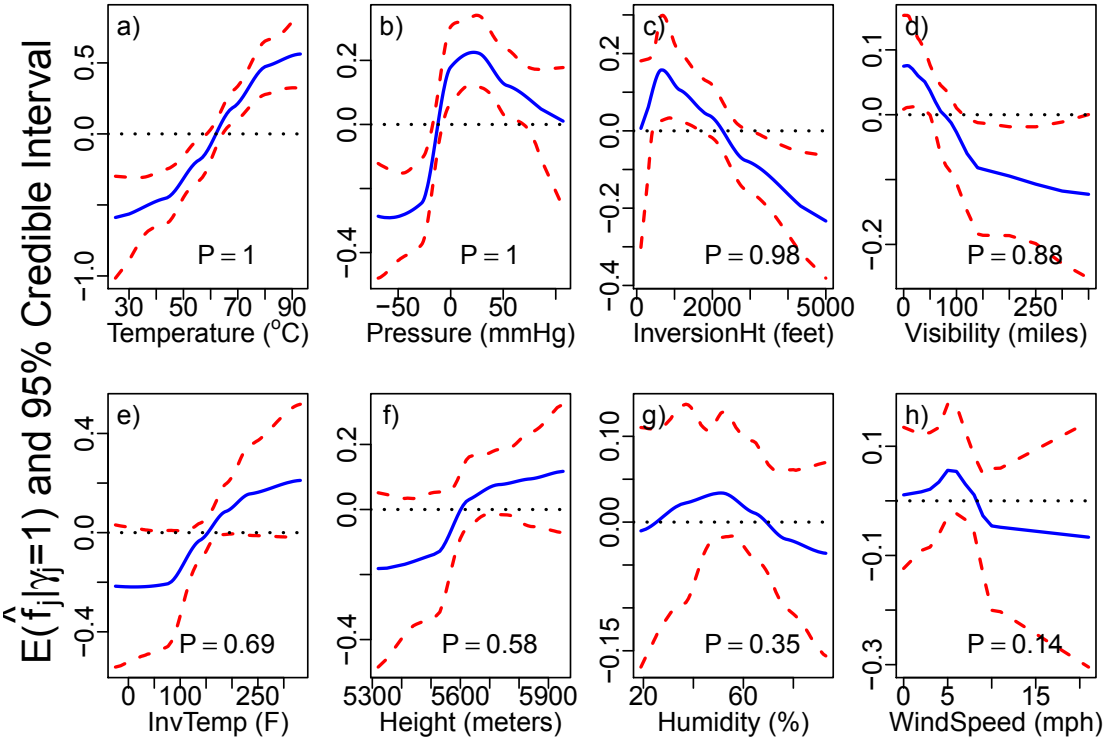

Figure 2.14: Estimated function $\hat{f}_j$ (blue solid lines) with 95% credible interval (red dashed lines) for the 8 predictors of ozone data labeled by the marginal selection probability $P = p(\gamma_j = 1|\mathbf{y})$ at $b = 1.6$.

The estimated results for $\boldsymbol{\beta}_j$'s are summarized in Figure 2.14, where the additive function components, $\hat{f}_j = Z_j E(\beta_j|\gamma_j = 1)$'s, are plotted with an enveloping 95% credible band. The marginal selection probability, $P = p(\gamma_j|\mathbf{y})$, for each variable is labeled in each plot. Because the smallest P is 0.14, we decide that there are enough iterations for all $\gamma_j = 1$ to estimate all $\hat{f}_j$. We can therefore identify three groups of the variables. The first group, which includes Temp, Press, InvHt and Vis, has $P \geq 0.88$. The second group

includes InvTp and Hgt with $P = 0.69$ and 0.58. The last group contains Hum and WS with $P < 0.5$.

From a variable selection standpoint, we would certainly want to choose all variables in the first group, and the last group of variables would be excluded, since their selection probabilities are very close to the baseline. Because of the small number of variables admitted and the near independence of each variable, we can consider the second group as true variables with small signals.

Regarding function components estimation, the selection probability is consistent with signal estimation. As shown in Figure 2.14, the first group of variables has their credible interval bands only cover a small part of the zero line. The third group of variables has its credible interval band almost completely centered on the zero line. While the credible intervals of the second group covers the zero line entirely, the zero line is only on the edge of the credible interval.

We also report the summary statistics for all precision parameters and the intercept in Table 2.3. In this example we include the intercept term μ in Model (2.48) and assign a prior for μ: $[\mu] \sim N(0, \tau_\mu^{-1})$, and a Gamma prior for τ_μ: $[\tau_\mu] \sim G(4, 2)$. The full conditional distributions for μ and τ_μ are easy to derive, but they are not shown here. Note how the posterior means of these parameters adapt to the data. Especially interesting is that the τ_{ej}'s and τ_{dj}'s all start with the same prior, but the analyses give different posterior means. For the first group of variables, the posterior means for τ_{ej}'s and τ_{dj}'s are obviously different from their priors.

2.9.2 Gene Selection in Pathway Data

Mootha et al. (2003) presented an pathway based analysis to test a priori defined path-

Table 2.3: Parameter estimation of ozone data under Bayesian sparse additive model at $b = 1.6$.

Parameter	Prior		Posterior				
	Mean	Std.Dev.	Mean	Std.Dev.	Median	Lower 2.5%	Upper 2.5%
τ_e^{Temp}	1.000	1.414	0.671	1.192	0.150	0.003	4.199
τ_e^{Press}	1.000	1.414	0.787	1.270	0.282	0.008	4.570
τ_e^{InvHt}	1.000	1.414	0.198	0.597	0.021	0.000	1.700
τ_e^{Vis}	1.000	1.414	1.316	1.517	0.805	0.021	5.485
τ_e^{InvTp}	1.000	1.414	1.010	1.365	0.500	0.010	4.884
τ_e^{Hgt}	1.000	1.414	1.062	1.392	0.547	0.005	5.032
τ_e^{Hum}	1.000	1.414	1.050	1.431	0.509	0.001	5.129
τ_e^{WS}	1.000	1.414	1.104	1.505	0.545	0.001	5.334
τ_d^{Temp}	1.000	1.414	0.226	0.724	0.026	0.002	2.212
τ_d^{Press}	1.000	1.414	0.033	0.047	0.021	0.002	0.125
τ_d^{InvHt}	1.000	1.414	0.889	1.352	0.349	0.004	4.731
τ_d^{Vis}	1.000	1.414	0.751	1.235	0.227	0.004	4.226
τ_d^{InvTp}	1.000	1.414	1.122	1.413	0.611	0.005	5.227
τ_d^{Hgt}	1.000	1.414	1.045	1.429	0.521	0.009	5.140
τ_d^{Hum}	1.000	1.414	1.061	1.474	0.513	0.001	5.048
τ_d^{WS}	1.000	1.414	0.949	1.383	0.404	0.000	4.985
ϕ	-	-	6.602	0.542	6.585	5.599	7.681
μ	0.000	$\sqrt{\tau_\mu}$	2.143	0.066	2.145	2.011	2.265
τ_μ	2.000	1.000	1.043	0.495	0.971	0.316	2.205

ways for association with the diabetic disease. A pathway is a predefined set of genes that serve a particular cellular or physiological function. A genetic pathway can be expressed by a graph to represent the gene network within this pathway. Mootha et al. (2003) identified several significant pathways including "Oxidative phosphorylation" and "Alanine-and-aspartate metabolism". However, even with those significant pathways identified, gene selection in microarray data analysis is still difficult because the effects of alterations in gene expression are modest due to the large number of genes, small sample sizes and variability between subjects.

Stingo et al. (2011) provide a Bayesian technique to incorporate biological information into linear models to select genes and pathways. Similar to Stingo et al. (2011), we incorporate the pathway network information into our graph model, and we apply it to gene selection from the diabetes data from Mootha et al. (2003). However, our method uses the gene network information in the pathway as the prior for γ, and it does not select path-

ways. The data contains gene expressions from $n = 35$ subjects, 17 normal and 18 Type II diabetes patients. We merged three pathways of interest, "Oxidative phosphorylation", "Alanine-and-aspartate metabolism" and "Glutamate-metabolism" into one graph with a total of $p = 173$ nodes. Some nodes are different probe sets of the same gene, so the gene names are identical. The graph with 173 nodes is a subgraph of the corresponding merged graph obtained from the KEGG database. The continuous response y is the glucose level.

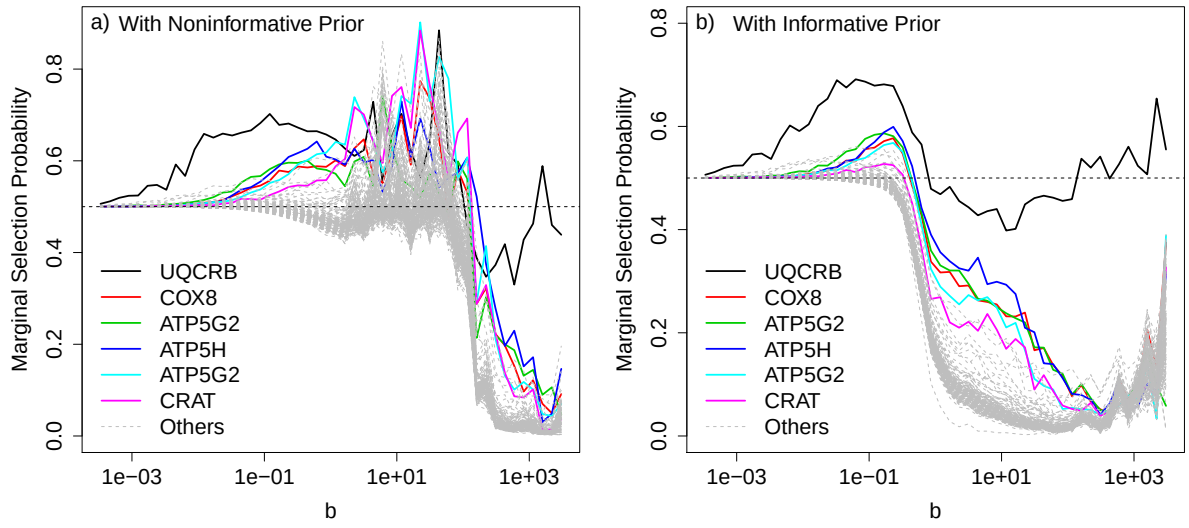

Figure 2.15: Profile curves of the selection probability of genetic pathway data with noninformative prior for γ (a), and with informative prior as (2.50) (b).

The top left plot of Figure 2.16 shows the network of our merged gene set. Note that the prior required for our graph model is undirected graph with only positive interactions. We see that most of the nodes are independent in this dataset, and that there are only three genetic clusters. Because of this, if we apply the cluster algorithm and use the adjacency matrix Λ based on the network information into Expression (2.18), we will end up with a few nodes in the same genetic cluster that could potentially form the clusters

for the algorithm. Therefore, we consider the following interaction matrix $W = \{w_{ij}\lambda_{ij}\}$ for the prior of $\boldsymbol{\gamma}$ with adjacency matrix $\Lambda = \{\lambda_{ij}\}$ as

$$\begin{aligned} &\lambda_{ij} = 1,\ i, j = 1, ..., p \\ &w_{ij} = \begin{cases} w, & i \notin S \text{ or } j \notin S \\ w + \Delta w, & i \in S \text{ and } j \in S, \end{cases} \end{aligned} \tag{2.50}$$

where S represents one of the three genetically networked gene clusters in the pathway network, w and Δw are small positive numbers stand for the strength of the interaction in the prior and the difference of two types of interaction. If $\Delta w = 0$, we can consider (2.50) as a baseline graph prior for $\boldsymbol{\gamma}$, which is a complete graph with positive fixed interaction. Since we also vary b to have an overall view of the selection probability, it is necessary to have $w \to 0$ when $b \to 0$ since with large shrinkage, $J_{ij} \to 0$ and we do not want w_{ij} to dominate the graph interaction. One convenient way to avoid this to express w as $w = w_0\Phi[\log(b)]$ which approaches 0 as $b \to 0$ and reaches the maximum w_0 for large b, where $\Phi(\cdot)$ is the CDF of standard normal. Note that the choice of w is involved in the consideration of the so-called phase transition (Li and Zhang, 2010). If w is too large all the nodes will always be connected which leads to either selection of all nodes or none of the nodes.

Now we consider $\Delta w \neq 0$, say $\Delta w = 5w$, so that we incorporate the genetic network information into the graph prior. Δw can not be too large, otherwise those genes in the genetic clusters will always be aligned which means that they will all have a small selection probability. We therefore choose $w_0 \approx 0.01$ as small as possible to avoid the phase transition phenomenon, but w_0 must be large enough to reduce the size of region II for

b, which is caused by small sample size. With this selection and for large b, we have the prior interaction $w_{ij} \approx 0.01$ for two nodes not in the same genetic cluster, and $w_{ij} \approx 0.06$ for two nodes in the same cluster.

As shown in Figure 2.15, the effect of incorporating prior information for the graph model is obvious. We ran the cluster algorithm for total $N = 40000$ iterations, and discarded the first 10000 as burn-in. Therefore the selection probability is calculated by taking the mean of $\boldsymbol{\gamma}$ over 30000 iterations. In Figure 2.15 (a), with noninformative prior for $\boldsymbol{\gamma}$, we can see that even though we are still able to identify several genes behaving differently from the rest (highlighted by solid lines in the plot), all the curves are mixed for the moderate value of b. On the other hand, in Figure 2.15 (b), with an informative prior for $\boldsymbol{\gamma}$ defined as (2.50), the profile curves are much "cleaner" even with moderate b. Around $b = 10$ we see a bunch of curves are clearly distinguishable from the rest. We highlighted six nodes with the highest selection probability around $b = 10$ in Figure 2.15 (b).

In Figure 2.16 we fixed $b = 8.5$ and ran the cluster algorithm for $N = 60000$ iterations with the first 20000 discarded. With this shrinkage parameter, the prior interaction parameter was $w_{ij} \approx 0.06$ for i and j in the genetic cluster and $w_{ij} \approx 0.01$ otherwise. The selection probability for all nodes are shown in the bottom left of Figure 2.16, where we take a cut-off probability as 0.2 and identify 6 nodes that have relative high selection probabilities. Among those nodes, UQCRB has the largest selection probability for the entire range of b, so it is easy to identify UQCRB as the most significant gene. We also select other five genes, COX8, ATP5G2 (two probe sets), ATP5H and CRAT at $b = 8.5$. All the genes selected except CRAT are from "Oxidative phosphorylation" pathway, which is related to ATP synthesis. It is well known that ATP plays an importance role in Type II diabetic disease. CRAT is from "Alanine-and-aspartate metabolism" pathway.

Since our cluster algorithm forms the Wolff cluster at each iteration, a byproduct of the

MCMC sampler is the frequency of two nodes being aligned or anti-aligned when they are in the cluster. The top right plot in Figure 2.16 is the heatmap matrix of the frequency of two nodes being aligned in the cluster out of 40000 iteration. The bottom left plot is the frequency of two nodes being anti-aligned in the cluster. The color bar of two plots shows the scale of the frequency, the darker the color the lower the frequency. In the top right plot, the dark colored lines are the genes that have a lower chance of being aligned to the others when they form the cluster. In the bottom left plot, the bright colored lines are the same genes but with a high chance of being anti-aligned with others if they form the cluster. Note that those lines are consistent to the genes that have a high selection probability in the bottom right plot. This is because most of the genes have low selection probability around 0.05, and genes with higher (lower) selection probabilities should have a lower (higher) chance of being (anti-)aligned with them. We define individual nodes as always self-aligned, so that the diagonal in the top right plot is 1. Meanwhile, an individual node is never anti-aligned to itself, so the diagonal in the bottom left plot is zero. The prominent color of these genes in the two heatmaps shows that we can also use the cluster information to distinguish genes.

So far, we identify 6 genes (probe sets) with cut-off probability 0.2. If we so choose, we may decrease the cutoff to select more genes. However, the selection probabilities are for the most part low, with the exception of UQCRB at fixed b. This might simply means the signals are weak. Here we selected the genes not only depending on the selection probability at fixed b, but also by examining their overall profile as shown in Figure 2.15. We also demonstrate that the graph model variable selection can easily adopt the prior graph information, and thus we can consider similar approach to that of Stingo et al. (2011) to select networked pathways, which may result in a higher selection probability for pathways at the optimal shrinkage parameter b.

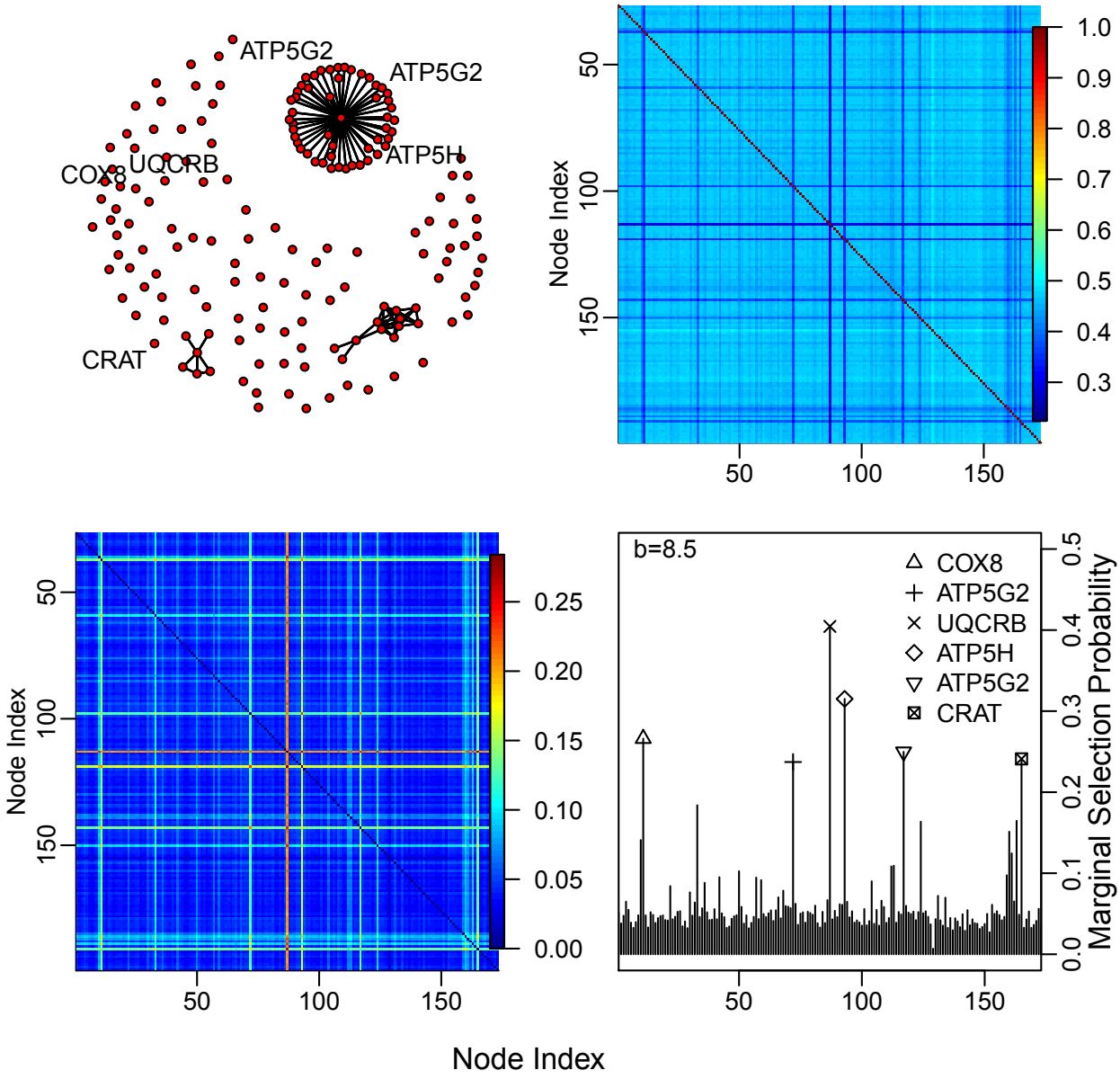

Figure 2.16: Summary of the results for the genetic pathway data. Top left: genetic network structure of the data. Top right: the frequency matrix of two nodes aligned in the cluster over total iterations. Bottom left: the frequency matrix of two nodes anti-aligned in the cluster over total iterations. Bottom right: Selection probability with cluster algorithm at $b = 8.5$ with informative prior (2.50).

2.10 Discussion

The goal of this dissertation is to present the BIGM from two major viewpoints. The first is how to sample the "in" or "out" binary random variable. We pointed out that BVS can be considered as a binary random process on a complete graph given noninformative prior for γ_j's, and we compared the single-site and generalized Wolff cluster updating algorithm. The other viewpoint deals with how to construct the interaction matrix of the complete graph, which is implemented by sampling the linear model coefficient β_j's through the scale mixtures of normal priors.

We also discussed the marginal selection probability profile under different shrinkage parameters and compared three prior settings for β_j, which represent three typical situations of shrinkage proportion. Our BIGM method possesses the advantages of simplicity, easy implementation, and straightforward extension to more complex models. For example, the BIGM is very easy to extend to Bayesian sparse model by representing the nonparametric function components f_j as a linear combination of the basis matrix $f_j = Z_j\boldsymbol{\beta}_j$. We can then employ the group selection of vector $\boldsymbol{\beta}_j$. Another example of an extension of our method is the incorporation of network information for $\boldsymbol{\gamma}$. Although this dissertation does not focus on how to construct the prior network structure information, the simulation and real data analyses show that it is easy to incorporate prior graph information and improve the performance of the BIGM. This dissertation also systematically studies the behaviors of the marginal selection probability against the shrinkage. Both theoretical and simulated results show that, to have the largest gap between the signals and the noise, it is critical for the scale mixture normal prior to maintain substantial proportion on small shrinkage.

However, this dissertation only starts with a different view about BVS, and further

research could include but is not limited to the following questions. (1) We simply choose b where there is the largest gap between our two sets of signals. Further research should show the consistency of selecting the predictors based on the "largest gap" rule. (2) Fixing b limits the performance of our method, we can adopt a remedy similar to exchange Monte Carlo by running parallel MCMC's at two or more b's with some b's small and others large, and exchanging their configuration according to a certain probability that remains detailed balanced. (3) Our BSAM does not include interaction terms, but it should be easily extended to do so similar to Reich et al. (2009). (4) It is difficult to construct a meaningful network for the prior of $\boldsymbol{\gamma}$. This keeps open the question discussed by Li and Zhang (2010) and Monni and Li (2010).

Chapter 3

Sparsity Recovery From Multivariate Nonparametric Models

3.1 Introduction

The variable selection problem is important in many research areas such as genomics, data mining, image analysis, text and speech analysis, and other areas with high dimensional data. In general, the input variables form an interacting network with each other and modeling these interactions is complicated due to high order interaction terms.

A number of methods and programs for variable selection problems have been developed in linear or nonparametric regression models. In the linear regression model, substantial progress for variable selection has been made with both numerical and theoretical understanding of the oracle properties and the consistency conditions for LASSO (Fan and Li, 2001; Meinshausen and Bühlmann, 2006; Wainwight, 2009; Yuan and Lin, 2007; Zhao and Yu, 2006; Zou, 2006). It is well known that LASSO fails if the incoherence

conditions for consistency are not satisfied. This places some limitation on the application of LASSO.

There are a few approaches for modeling high dimensional data using multidimensional nonparametric models via spline-based generalized additive models, smoothing spline and thin plate splines, and functional ANOVA (Green and Silverman, 1994; Hastie and Tibshirani, 1990; Wahba, 1990). Those methods include the Component Selection and Smoothing Operator (COSSO) method (Lin and Zhang, 2006), high dimensional additive modeling (Meier et al., 2009), Sparse Additive Models (SpAMs) (Ravikumar et al., 2009), and an extension of SpAMs, Variable Selection using Adaptive Nonlinear Interaction Structure in High dimensions (VANISH) (Radchenko and James, 2010). These nonparametric variable selection approaches are performed in terms of function components selection, i.e., modeling the function components (including nonlinear interactions) additively and then selecting significant components. However, when the number of input variables is large and their interactions are complicated, modeling each interaction term is extremely expensive so these function components approaches may not be efficient. For example, in the case of a genetic pathway analysis, genes in a pathway serve a particular cellular or physiological function together, so they form a complicated interaction network of high order, unsuitable for analysis through function components.

Other variable selection approaches based on the kernel machine method have achieved great success. Liu et al. (2007) established the connection between the least squares kernel machine (LSKM) and linear mixed models. Zou et al. (2010) employed a nonparametric regression model with a Gaussian process which simultaneously considers all the possible interactions. In these works, the interactions among the multi-dimensional variables are modeled automatically by the kernel. Because of their simplicity and generality, function kernels and associated function spaces are a powerful technique to analyze

multi-dimensional data. Unlike spline based regression models, they do not specify the smoothing conditions of an unknown function, although they share the same fundamental theoretical base.

In this dissertation we also focus on a variable selection approach based on the kernel machine method because the family of kernel functions is extremely rich for multivariate smoothing. Our method has the flexibility to model additive functional ANOVA, spline based models or nonadditive smoothing functions. To demonstrate this, note that under certain conditions Mercer's theorem (Rasmussen and Williams, 2006) states that a unique function space is specified by a kernel function. For example, since any symmetric positive definite matrix is a valid Gram matrix (a symmetric matrix embedding a finite set of observations specified by the kernel function $k(\cdot,\cdot)$), an additive Gram matrix $K = \sum_j^p \xi_j K_j$ can be used to model functional ANOVA $f\left(\boldsymbol{x}^T\right) = \sum_{j=1}^p f_j(x_j)$, where K_j is the Gram matrix for the jth function space f_j and $\boldsymbol{x}^T = (x_1, ..., x_p)$. According to the Representer Theorem (Kimeldorf and Wahba, 1971), a nonparametric function is represented using a kernel function, $f_j(x) = \sum_{l=1}^n \alpha_l k_j(x_l, x)$ (the dual representation) where $k_j(\cdot,\cdot)$ is the kernel function for the jth function component and α_l's are the kernel function coefficients. With penalty on the norm (or pseudonorm) of the jth function component f_j, $\|f_j\|_{\mathcal{H}_{K_j}}$, sparsity of the function components can be recovered. This has already been applied to COSSO and multiple kernel learning (MKL) (Rakotomamonjy et al., 2008). Furthermore, in the MKL $K = \sum_j^p \xi_j K_j$ matrix, each K_j corresponds to a specific function f_j spanned by a particular set of orthogonal basis function $\{\phi_l^j(x)\}_{l=1}^d$. i.e., the function component of jth variable can be expressed as $f_j(x) = \sum_{l=1}^d \omega_l^j \phi_l^j(x)$ (the primal representation). Applying the penalty on $\boldsymbol{\omega}^j$ through $\sqrt{n^{-1}\boldsymbol{\omega}^{j^T}\boldsymbol{\phi}^{j^T}\boldsymbol{\phi}^j\boldsymbol{\omega}^j}$, where $\boldsymbol{\omega}^j$ is vector of ω_l^j's and $\boldsymbol{\phi}^j$ is vector of ϕ_l^j's, one can have the sparse additive models (SpAMs) and function components sparsity might be obtained by shrinking f_j to zero.

However, to the best of our knowledge, no variable selection based on the kernel machine has currently been established in a nonadditive multivariate smoothing function model. In terms of a nonadditive multivariate smoothing function, the kernel function $k(\boldsymbol{x}, \boldsymbol{x}')$ is usually a nonlinear function of multivariate $\boldsymbol{x}$, such as the Gaussian kernel function, $\exp(-\rho\|\boldsymbol{x} - \boldsymbol{x}'\|^2)$, with scale parameter ρ. In a model with such a kernel function, the response can no longer be expressed as an additive function components and no sparse function components are available. Therefore variable selection for recovering the sparsity of $\boldsymbol{x}$ within the nonadditive function becomes challenging. The goal of this dissertation is to propose a new variable selection approach based on the kernel machine that is able to recover sparsity of input variables in a nonadditive multivariate smoothing function.

Our method is motivated by Automatic Relevance Determination (ARD), which was originally formulated in the framework of neural networks in the context of Gaussian process (Neal, 1996; MacKay, 1994) considering kernel functions of the form $k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left\{-\sum_{j=1}^{p} \xi_j (x_j - x_j')^2\right\}$. ARD has been used to estimate scale parameters ξ_j's using features for feature selection and classification, such as neural networks and Support Vector Machines (SVM) (Neal, 1996; Tipping, 2001; Rasmussen and Williams, 2006). Estimation of these hyperparameters $\xi_j, j = 1, ..., p$ reveals the sparsity of the input variables. For those ξ_j's close to zero the response becomes relatively insensitive to the corresponding input variable x_j.

Similarly, we consider modeling the nonadditive multivariate smoothing function with a general kernel function with hyperparameters ξ_j's controlling the importance of the individual predictors. By shrinking these ξ_j's similarly to the nonnegative garrote estimator (Breiman, 1995; Yuan and Lin, 2007), we can select the predictors as ARD does. In this way, we generalize the linear nonnegative garrote model to a nonlinear version

of a nonnegative garrote on a kernel machine. However, because the kernel matrix K can be flexible in catching the features, our approach can be applied to either additive or nonadditive models by choosing different K structures. We further show that the variable selection problem with nonparametric models can be considered as a special case of the problem of learning the kernel function via regularization (Micchelli and Pontil, 2005; Lanckriet et al., 2004). With this point of view, specific kernel functions are determined from a specific set of scaling parameters ξ_j, and by varying the ξ_j's we optimize the objective function through learning the kernel.

For a theoretical understanding of our approach, we develop the incoherence conditions for NGK. Ravikumar et al. (2009) studied the sparsistency properties of SpAMs and Bach (2008) extended the consistency conditions to MKL. There are few theoretical studies on the consistency of linear nonnegative garrote (Yuan and Lin, 2007). Unlike LASSO where the correlation of the predictors determines the incoherence conditions, the incoherence conditions of NGK are determined by the correlation of the vectors of the first derivatives of the smoothing function or the nonlinear components respective to the scale parameters, ξ_j's. This difference offers a new approach to recovering sparsity of predictors when linear LASSO fails. We show that under certain conditions sparsistency can be established on NGK. To recover sparsity of ξ_j's, an efficient coordinate descent/backfitting algorithm has been developed to achieve the regularization path for ξ_j's.

In Section 3.2, we first define the optimization function of our approach using the kernel machine model and discuss the connection of our approach with the linear nonnegative garrote model and the kernel machine learning problem. In Section 3.3 we propose our coordinate descent updating algorithm for the solution path of the scaling parameters. In Section 3.4 we discuss the necessary and sufficient conditions for consistency and sparsistency. We also show the asymptotic properties of our method with the consistent

initial kernel function coefficients. Some theoretical proofs are given in Section 3.5. In Section 3.6, we present several simulated examples of NGK. In Section 3.7, we apply our method to two real datasets: one is from the area of cryptography research, the other contains the genetic pathway expression data. Section 3.8 concludes with remarks.

3.2 Flexible Multivariate Nonparametric Model

3.2.1 Multivariate Nonparametric Model Using Kernel Machine

Consider an n-observation and p-predictor dataset $(\mathbf{y}, X)$, where $X = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_p]$, $\mathbf{x}_j = (x_{j1}, ..., x_{jn})^T$ is an $n \times 1$ vector of the jth predictor, $j = 1, ..., p$. In other words, $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n]^T$ where $\boldsymbol{x}_i^T$ is a $1 \times p$ vector of predictors of the ith observation, $i = 1, ..., n$.

According to the Representer Theorem, the multivariate nonparametric regression model can be expressed as (Kimeldorf and Wahba, 1971)

$$\mathbf{y} = \mathbf{f}(X) + \boldsymbol{\epsilon} = K\boldsymbol{\alpha} + \boldsymbol{\epsilon}, \tag{3.1}$$

where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 I)$ and K is the kernel matrix corresponding to the function space $\mathcal{H}_K$, $f \in \mathcal{H}_K$, and also known as a "Gram matrix" of the kernel function $k(\boldsymbol{x}, \boldsymbol{x}')$. Thus the nonlinear function $f(\boldsymbol{x})$ can be expressed as $\sum_{i=1}^n \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x})$, where $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_n)^T$ is an $n \times 1$ vector of the kernel function coefficients. Note in Model (3.1), $\mathbf{y}$ is centered, i.e., $\sum y_i = 0$. We also standardize X such that $\sum_{l=1}^n x_{jl} = 0$ and $\sum_{l=1}^n x_{jl}^2 = 1, j = 1, ..., p$.

To estimate $\boldsymbol{\alpha}$ in (3.1), least squares kernel machine estimation minimizes the least squares error with a penalized norm of $\|\mathbf{f}\|_{\mathcal{H}_K}^2 = \boldsymbol{\alpha}^T K \boldsymbol{\alpha}$ induced by the kernel of the

function space $\mathcal{H}_K$

$$\frac{1}{2}\|\mathbf{y} - K\boldsymbol{\alpha}\|^2 + \frac{1}{2}\lambda_0 \boldsymbol{\alpha}^T K \boldsymbol{\alpha}, \tag{3.2}$$

and the solution is

$$\hat{\boldsymbol{\alpha}} = (\lambda_0 I + K)^{-1}\mathbf{y}, \tag{3.3}$$

where $\lambda_0 > 0$ is the smoothing parameter that balances the tradeoff between goodness of fit and smoothing of the curve or high dimensional surface.

3.2.2 Nonnegative Garrotte on Kernel (NGK)

The Gram matrix can be viewed as applying a componentwise function on the similarity matrix among observations. The similarity metric between two vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$ can be the negative of the squared Euclidean distance, $-\|\boldsymbol{x} - \boldsymbol{x}'\|^2$, or the angle (dot product) between them, $\boldsymbol{x}^T\boldsymbol{x}'$. Both similarity metrics can be written in an additive form in terms of p predictors, i.e., $-\|\boldsymbol{x}-\boldsymbol{x}'\|^2 = -\{(x_1-x_1')^2+...+(x_p-x_p')^2\}$ and $\boldsymbol{x}^T\boldsymbol{x}' = x_1x_1'+...+x_px_p'$. By this additivity, the kernel matrix can be expressed as a linear or nonlinear function of additive form. For example, the Gram matrix defined by the dot product $\boldsymbol{x}^T\boldsymbol{x}'$ among observations results in the linear kernel

$$K(X) = \rho XX^T = \rho\sum_{j=1}^{p}\mathbf{x}_j\mathbf{x}_j^T = \rho\sum_{j=1}^{p}D^j,$$

where $D^j = \mathbf{x}_j\mathbf{x}_j^T$, with (k,l)th entry $d_{kl}^j = x_{jk}x_{jl}, 1 \le k,l \le n$, and ρ is a scale parameter. Unlike the linear kernel, the Gaussian kernel matrix can be expressed in a form of a nonlinear function, $\exp(\cdot)$, of $-\|\boldsymbol{x} - \boldsymbol{x}'\|^2$

$$K(X) = \exp\left(\rho\sum_{j=1}^{p}D^j\right),$$

where D^j is the matrix with (k,l)th entry $d_{kl}^j = -(x_{jk} - x_{jl})^2$, or the (k,l)th entry of matrix $\sum_{j=1}^p D^j$ is $-\|\boldsymbol{x}_k - \boldsymbol{x}_l\|^2 = -\sum_{j=1}^p (x_{jk} - x_{jl})^2$.

More generally, considering a set of nonnegative scale parameters $\boldsymbol{\xi} = (\xi_1, ..., \xi_p)$ with ξ_j corresponding to each predictor, $\mathbf{x}_j$, both kernels can be expressed as

$$K(\boldsymbol{\xi}, X) = g\left(\sum_{j=1}^{p} \xi_j D^j\right), \tag{3.4}$$

where $\xi_j \geq 0, j = 1, ..., p$, and $g(\cdot)$ is a componentwise function of the matrix entries. That is, for a linear kernel, all $\xi_j = \rho$ and $g(\cdot)$ is the identity function, and for a Gaussian kernel, all $\xi_j = \rho$ and $g(\cdot) = \exp(\cdot)$. Note that we do not need extra constraints on ξ_j's such as $\sum \xi_j = 1$. For the Gaussian kernel, $\exp(\cdot)$ already places a constraint on ξ_j's, and for the linear, the solution paths of ξ_j's with and without constraints differ only by a scalar, while their sparsity properties remain the same. Thus, for computing convenience, we do not apply constraints on $\boldsymbol{\xi}$ for both kernels.

By introducing such nonnegative parameters in the kernel matrix, we develop a variable selection approach for the nonparametric regression model (3.2) similar to the linear nonnegative garrotte method (Breiman, 1995). That is, we apply an extra penalty on $\boldsymbol{\xi}$ such that the optimization problem (3.2) is subject to $\xi_j \geq 0$ and $\sum \xi_j \leq c$ (c is a positive real number), which results in the following optimal problem

$$\frac{1}{2}\|\mathbf{y} - K(\boldsymbol{\xi}, X)\boldsymbol{\alpha}\|^2 + \frac{1}{2}\lambda_0 \boldsymbol{\alpha}^T K(\boldsymbol{\xi}, X)\boldsymbol{\alpha} + n\lambda \sum \xi_j, \tag{3.5}$$

where $\lambda > 0$ is a tuning parameter. We refer to this method as "nonnegative garrote on kernel machines".

3.2.3 Connection with Linear Nonnegative Garotte Estimator

Introduced by Breiman (1995) for variable selection on a linear model $\mathbf{y} = \mathbf{f} + \boldsymbol{\epsilon} = X\boldsymbol{\beta} + \boldsymbol{\epsilon}$, the linear nonnegative garotte estimator for the shrinking factor $\boldsymbol{\xi} = (\xi_1, ..., \xi_p)^T$ is the solution that minimizes

$$\frac{1}{2}\left\|\mathbf{y} - \sum_{j=1}^{p} \xi_j \mathbf{x}_j \tilde{\beta}_j^{OLS}\right\|^2 + n\lambda \sum \xi_j, \text{ subject to } \xi_j \geq 0, \forall j, \tag{3.6}$$

where $\tilde{\beta}_j^{OLS}$ is the initial estimate of β_j from the ordinal least squares estimate. For an orthonormal design $X^T X = I$ (i.e., $\mathbf{x}_i^T \mathbf{x}_j = \delta_{ij}$ such that $\delta_{ij} = 0 \, \forall i \neq j$ and $\delta_{ij} = 1 \, \forall i = j$), the nonnegative garrote solution for β_j is

$$\hat{\beta}_j = \hat{\xi}_j \tilde{\beta}_j^{OLS}, \text{ and } \hat{\xi}_j = \left(1 - \frac{n\lambda}{\left(\tilde{\beta}_j^{OLS}\right)^2}\right)_+, \; j = 1, ...p.$$

where subscript "+" indicates the positive part of the expression. We can show that (3.6) is the special case of (3.5) with a linear kernel. To see this, consider (3.5) without the penalty on $\boldsymbol{\alpha}$, thus $\lambda_0 = 0$, and the kernel matrix is $K = \sum \xi_j \mathbf{x}_j \mathbf{x}_j^T$. Then the least squares kernel machine solution is related to the OLS solution by choosing the initial $\tilde{\boldsymbol{\alpha}} = \mathbf{y}$, and we obtain the initial estimation for the response

$$\tilde{\mathbf{f}} = K\tilde{\boldsymbol{\alpha}} = \sum \xi_j \mathbf{x}_j \mathbf{x}_j^T \mathbf{y} = \sum \xi_j \mathbf{x}_j \tilde{\beta}_j^{OLS} = \sum \xi_j \tilde{f}_j,$$

where $\tilde{f}_j$ represents the initial marginal response and the OLS estimation of the linear model is $\tilde{\beta}_j^{OLS} = \mathbf{e}_j^T (X^T X)^{-1} X^T \mathbf{y} = \mathbf{x}_j^T \mathbf{y}$ because $X^T X = I$ ($\mathbf{e}_j^T$ is the selection vector with 1 in the jth position).

Yuan and Lin (2007) suggested that the linear nonnegative garrote can be used in combination with the initial estimator of $\boldsymbol{\beta}$ other than the least squares estimate. They proposed a more general linear nonnegative garrote model

$$\frac{1}{2}\left\|\mathbf{y}-Z\boldsymbol{\xi}\right\|^2+n\lambda\sum\xi_j,\ \text{subject to}\ \xi_j\geq 0,\forall j, \tag{3.7}$$

where Z are a matrix with columns as the initial estimates of the marginal response. They prove that if the initial estimate is consistent then the nonnegative garrote estimate is also consistent. Based on this idea, Yuan (2007) applied the nonnegative garrote component selection method to functional ANOVA models, where the initial estimates of the function components $\tilde{f}_j$'s were used as the columns of Z ($\tilde{f}_j$ can be an interaction component). So the linear nonnegative garrote method can be applied to additive functional ANOVA models with interaction components. In Section 3.3, we will further derive the approximated linear nonnegative garrote form of our model (3.5) and show that Z is a matrix with columns $\left(\frac{\partial K}{\partial \xi_j}\right)\tilde{\boldsymbol{\alpha}}=D^j\tilde{\boldsymbol{\alpha}}, j=1,...,p$. In this case, a local linear approximation of the kernel may be needed, for example, $K(\boldsymbol{\xi})\approx K(\boldsymbol{\xi}^*)+\sum_{j=1}^{p}(\xi_j-\xi_j^*)K_j'(\boldsymbol{\xi}^*)$. For a general kernel function, $\left(\frac{\partial K}{\partial \xi_j}\right)\tilde{\boldsymbol{\alpha}}$ can be understood as the slope of the change of initial f along ξ_j direction given initial $\tilde{\boldsymbol{\alpha}}$. Note that (3.7) can be derived without an approximation for the linear kernel. On the other hand, (3.7) is only meaningful in theoretical analysis for a Gaussian kernel since in practice we can not use it to derive an efficient algorithm.

3.2.4 Connection with the Kernel Machine Learning

First define function $Q_0(\cdot)$ as

$$Q_0\left(K(\boldsymbol{\xi},X),\boldsymbol{\alpha}\right)=\frac{1}{2}\left\|\mathbf{y}-K(\boldsymbol{\xi},X)\boldsymbol{\alpha}\right\|^2+\frac{\lambda_0}{2}\boldsymbol{\alpha}^TK(\boldsymbol{\xi},X)\boldsymbol{\alpha}. \tag{3.8}$$

The following lemma shows that Q_0 can be expressed as a function of K.

Lemma 3.2.1. *(Micchelli and Pontil, 2005) If K is in the set of all kernels on input set $\mathbb{X}$, and if for a set of distinct points $X \in \mathbb{X}$ and $K(X)$ is positive definite, then*

$$Q_0(K) = \frac{\lambda_0}{2}\mathbf{y}^T \left(\lambda_0 I + K\right)^{-1} \mathbf{y}, \tag{3.9}$$

and $Q_0(K)$ is a non-increasing convex function of K.

The form of (3.9) can be easily derived since the solution for the least squares kernel machine problem is $\hat{\boldsymbol{\alpha}} = (\lambda_0 I + K)^{-1}\mathbf{y}$. Plug this solution back into (3.8) and by simple algebra we have (3.9). A formal proof of Lemma 3.2.1 and the convexity of $Q_0(K)$ can be found in Micchelli and Pontil (2005) and Lanckriet et al. (2004). Then the solution $\hat{\boldsymbol{\xi}}$ of the optimization problem (3.5) is

$$\begin{aligned} &\min_K \; Q_0(K) = \frac{\lambda_0}{2}\mathbf{y}^T \left(\lambda_0 I + K\right)^{-1} \mathbf{y}, \\ &\text{subject to } K \in \mathbb{K}^* = \left\{ K(\boldsymbol{\xi}, X) : \boldsymbol{\xi} \in \mathbb{R}_+^p \text{ and } \sum_{j=1}^{p} \xi_j \le c, j = 1, ..., p \right\}, \end{aligned} \tag{3.10}$$

where $\mathbb{R}_+^p$ is the set of p dimensional nonnegative real numbers.

The objective function (3.10) implies that we have a kernel based learning problem on $\mathbb{K}^*$, a subset of all kernels on input set $\mathbb{X}$ ($X \in \mathbb{X}$). More generally, associated with the function $Q_0(K)$ and the kernel K is the variation problem of Micchelli and Pontil (2005)

$$Q_0(\mathbb{K}) = \inf\{Q_0(K) : K \in \mathbb{K}\}, \tag{3.11}$$

where $\mathbb{K}$ is a convex set of all positive semidefinite kernel functions. Thus our problem can be viewed as a special case of (3.11), as learning the kernel function via regularization

$\frac{\lambda_0}{2}\|\mathbf{f}\|^2_{\mathcal{H}_K} = \frac{\lambda_0}{2}\boldsymbol{\alpha}^T K\boldsymbol{\alpha}$, subject to $K \in \mathbb{K}^*$, where $\mathbb{K}^* \subset \mathbb{K}$. If optimizing Q_0 on $\mathbb{K}$, this is the problem of learning the kernel discussed by Micchelli and Pontil (2005).

Although our problem is learning the kernel K in the set $\mathbb{K}^*$, it is different from the problem of learning the kernel function discussed by Micchelli and Pontil (2005), Lanckriet et al. (2004) and Rakotomamonjy et al. (2008). This is because the set $\mathbb{K}^* \subset \mathbb{K}$ is usually not convex, and the optimization problem is learning the kernel through a nonlinear function $g(\cdot)$ on $\boldsymbol{\xi}$.

We can express (3.5) as a function of $\boldsymbol{\xi}$:

$$Q(\boldsymbol{\xi}) = Q_0(\boldsymbol{\xi}) + \lambda\sum_{j=1}^{p}\xi_j = \frac{\lambda_0}{2}\mathbf{y}^T\left(\lambda_0 I + K(\boldsymbol{\xi})\right)^{-1}\mathbf{y} + \lambda\sum_{j=1}^{p}\xi_j. \tag{3.12}$$

The convexity of $Q_0(\boldsymbol{\xi})$ is interesting since it determines the convexity of $Q(\boldsymbol{\xi})$ and convex objective functions have many convenient properties in the optimization problem. Unfortunately, however, it is not straightforward to determine the convexity of $Q_0(\boldsymbol{\xi})$ as its convexity completely depends on the kernel function $K(\boldsymbol{\xi})$ and X. The following Lemma shows a sufficient condition for $Q(\boldsymbol{\xi})$ to be a convex function of $\boldsymbol{\xi}$.

Lemma 3.2.2. *If the matrix set of $K(\boldsymbol{\xi}) = g(\sum_{j=1}^{p}\xi_j D^j)$ is concave on $\boldsymbol{\xi}$, i.e., $K(\theta\boldsymbol{\xi}+(1-\theta)\boldsymbol{\xi}') \succeq \theta K(\boldsymbol{\xi}) + (1-\theta)K(\boldsymbol{\xi}')$ where $0 \le \theta \le 1$, then the regularization problem (3.12) is convex on $\boldsymbol{\xi} \in \mathbb{R}^p_+$.*

This can be easily shown by the composition theorem (Boyd and Vandenberghe, 2004). That is, for a function $f(x) = h\{g(x)\}$, f is convex if h is convex and non-increasing and g is concave (see Section 3.5.1). An obvious example of a concave kernel K is the linear kernel (which is convex too). So we conclude that the objective function $Q_0(\boldsymbol{\xi})$ is a convex function of $\boldsymbol{\xi}$ for a linear kernel and $Q(\boldsymbol{\xi}) = Q_0(\boldsymbol{\xi}) + \lambda\sum\xi_j$ is a strictly convex function of $\boldsymbol{\xi}$. Thus the regularization problem (3.12) has many of the nice properties of convex

optimization. In particular, if $Q(\boldsymbol{\xi})$ is strictly convex, the solution $\hat{\boldsymbol{\xi}}$ is unique.

However, in many cases, it is not straightforward to derive the concavity or convexity of $K(\boldsymbol{\xi})$. For instance, with the Gaussian kernel, since $K(\boldsymbol{\xi})$ is neither concave nor convex on $\boldsymbol{\xi}$, it is difficult to determine the convexity of $Q(\boldsymbol{\xi})$. The most ideal scenario is, $Q(\boldsymbol{\xi}) = Q_0(\boldsymbol{\xi}) + \lambda \sum \xi_j$ is quasicovex (unimodal) so that $Q(\boldsymbol{\xi})$ has a unique minimum. Constructing a criterion for the global optimum is a difficult mathematical problem, especially when the form of $Q(\boldsymbol{\xi})$ is complicated with arbitrary kernel functions rather than linear kernels. In this case, the Hessian matrix of $Q(\boldsymbol{\xi})$ may not be positive (semi-)definite everywhere. However, one can expect a positive (semi-)definite Hessian matrix in the neighborhood of a minimum, i.e.,

$$H = \left\{ \frac{\partial^2 Q(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}^T \partial \boldsymbol{\xi}} \right\} \succeq 0. \tag{3.13}$$

In practice, we start with the initial $\hat{\boldsymbol{\xi}}$ close to zero when solving the optimization problem. Hence we can assume the initial values and the solution path are always in the neighborhood of a minimum where $H \succeq 0$. This is a reasonable assumption with Gaussian kernels since in the sparsity problem most ξ_j's are zero and, in general, non-zero ξ_j's are all small positive numbers. When ξ_j's are small numbers or are zero, the Gaussian kernel can be well approximated to the linear order of the Taylor expansion as a concave matrix of $\boldsymbol{\xi}$. In the following section we provide the regularity conditions which are usually satisfied in least squares error optimization.

3.2.5 Some Notation and Regularity Conditions

First we offer some notation. Let $\boldsymbol{\xi}^*$ and $\hat{\boldsymbol{\xi}}$ stand for the true $\boldsymbol{\xi}$ and minimum solution to (3.5). Consider a sparse vector $\boldsymbol{\xi}^*$, i.e., some $\xi_j^* = 0$. Without loss of generality, we denote $\boldsymbol{\xi}^* = (\xi_1^*, ..., \xi_p^*)^T = \left(\boldsymbol{\xi}_1^{*T}, \boldsymbol{\xi}_0^{*T} \right)^T$, where $\boldsymbol{\xi}_1^*$ is the vector of the first a nonzero ξ_i^*'s

and $\boldsymbol{\xi}_0^*$ is the vector of rest zero. We also define the nonzero index set of $\boldsymbol{\xi}^*$ as $\mathcal{A} := \{j \in \{1, ..., p\} | \xi_j^* > 0\}$, and define $\hat{\mathcal{A}} := \{j \in \{1, ..., p\} | \hat{\xi}_j > 0\}$ as the nonzero index set of $\hat{\boldsymbol{\xi}}$. Note that $\mathcal{A}$ has relatively small cardinality, $a = |\mathcal{A}|$, the number of true nonzero ξ_j's.

Let the least squares error estimate of $\boldsymbol{\alpha}$ be $\hat{\boldsymbol{\alpha}} = \Delta^{-1}(\hat{\boldsymbol{\xi}})\mathbf{y}$, where $\Delta(\boldsymbol{\xi}) = \lambda_0 I + K(\boldsymbol{\xi})$, and denote the true $\boldsymbol{\alpha}$ vector as $\boldsymbol{\alpha}^*$. We further define matrices Z, Z_1 and Z_0 for a given $\boldsymbol{\alpha}^*$ and $\tilde{Z}$, $\tilde{Z}_1$ and $\tilde{Z}_0$ for any given estimate $\tilde{\boldsymbol{\alpha}}$ as

$$\begin{aligned} Z = [\mathbf{z}_1, ..., \mathbf{z}_p] = [Z_1, Z_0] &= \left[\left\{ K_j'(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* \right\}_{1 \le j \le a}, \left\{ K_j'(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* \right\}_{a+1 \le j \le p} \right], \\ \tilde{Z} = [\tilde{\mathbf{z}}_1, ..., \tilde{\mathbf{z}}_p] = [\tilde{Z}_1, \tilde{Z}_0] &= \left[\left\{ K_j'(\boldsymbol{\xi}^*)\tilde{\boldsymbol{\alpha}} \right\}_{1 \le j \le a}, \left\{ K_j'(\boldsymbol{\xi}^*)\tilde{\boldsymbol{\alpha}} \right\}_{a+1 \le j \le p} \right], \end{aligned} \tag{3.14}$$

where $K_j'(\boldsymbol{\xi}^*) = \left. \frac{\partial K}{\partial \xi_j} \right|_{\boldsymbol{\xi}^*}$ which is obtained by taking the partial derivative of the componentwise entries of K. Note $K_j'\boldsymbol{\alpha}$ is an $n \times 1$ vector and both Z's and $\tilde{Z}$'s are $n \times p$ matrices. Covariance matrices are also defined as

$$\begin{aligned} \Sigma_{11} &= \left(n^{-1} Z_1^T Z_1 \right), \quad \Sigma_{01} = \left(n^{-1} Z_0^T Z_1 \right), \\ \tilde{\Sigma}_{11} &= \left(n^{-1} \tilde{Z}_1^T \tilde{Z}_1 \right), \quad \tilde{\Sigma}_{01} = \left(n^{-1} \tilde{Z}_0^T \tilde{Z}_1 \right), \end{aligned} \tag{3.15}$$

where both Σ_{11} and $\tilde{\Sigma}_{11}$ are assumed to be invertible.

Now, we define the $p \times 1$ vector $\mathbf{v}_n(\boldsymbol{\xi})$ and the $p \times p$ matrix $M_n(\boldsymbol{\xi})$ as the following

$$\mathbf{v}_n(\boldsymbol{\xi}) = \lambda_0^{-1} n^{-1/2} \frac{\partial Q_0(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = - \left\{ \frac{1}{2\sqrt{n}} \mathbf{y}^T \Delta^{-1} \left(\frac{\partial K}{\partial \xi_j} \right) \Delta^{-1} \mathbf{y} \right\}_{1 \le j \le p}^T,$$

$$M_n(\boldsymbol{\xi}) = (\lambda_0 n)^{-1} \frac{\partial^2 Q_0(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}^T \partial \boldsymbol{\xi}} = \left\{ \frac{1}{2n} \mathbf{y}^T \Delta^{-1} \left[\frac{\partial K}{\partial \xi_i} \Delta^{-1} \frac{\partial K}{\partial \xi_j} + \frac{\partial K}{\partial \xi_j} \Delta^{-1} \frac{\partial K}{\partial \xi_i} - \frac{\partial^2 K}{\partial \xi_i \partial \xi_j} \right] \Delta^{-1} \mathbf{y} \right\}_{1 \le i, j \le p}.$$

These vectors and matrices are analogous to the negative score and Hessian matrix of a log likelihood function $-Q_0(\boldsymbol{\xi})$ with the $\log |K|$ term omitted. To see this and the regulari-

ty conditions, first we consider the log likelihood function of our model. From a Bayesian point of view, assume the following distributions,

$$\begin{aligned} \mathbf{y}|\boldsymbol{\alpha},\boldsymbol{\xi} &\sim N\left(K(\boldsymbol{\xi},X)\boldsymbol{\alpha},\sigma^2 I\right), \\ \boldsymbol{\alpha}|\boldsymbol{\xi} &\sim N\left(\mathbf{0},\sigma_\alpha^2 K(\boldsymbol{\xi},X)^{-1}\right), \\ \xi_j &\sim \tilde{\lambda}e^{-\tilde{\lambda}\xi_j}, j=1,...,p. \end{aligned}$$

Then up to an additive constant the negative log likelihood function of $\boldsymbol{\xi}$ is equal to

$$\frac{1}{2\sigma^2}\left\|\mathbf{y}-K(\boldsymbol{\xi},X)\boldsymbol{\alpha}\right\|^2+\frac{1}{2\sigma_\alpha^2}\boldsymbol{\alpha}^T K(\boldsymbol{\xi},X)\boldsymbol{\alpha}-\frac{1}{2}\log|K(\boldsymbol{\xi},X)|+n\tilde{\lambda}\sum\xi_j.$$

By letting $\lambda_0=\frac{\sigma^2}{\sigma_\alpha^2}$ and $\sigma^2\tilde{\lambda}=\lambda$, the above expression is equivalent to

$$\frac{1}{2}\left\|\mathbf{y}-K(\boldsymbol{\xi},X)\boldsymbol{\alpha}\right\|^2+\frac{\lambda_0}{2}\boldsymbol{\alpha}^T K(\boldsymbol{\xi},X)\boldsymbol{\alpha}-\frac{\sigma^2}{2}\log|K(\boldsymbol{\xi},X)|+n\lambda\sum\xi_j. \tag{3.16}$$

Expression (3.16) only differs from (3.5) by a $\log|K|$ term. Thus, strictly speaking, the estimate of $\hat{\boldsymbol{\xi}}$ by (3.5) is no longer the maximum likelihood estimation (MLE). The reason for omitting the $\log|K|$ term is that in the NGK model, values of ξ_j's is usually sparse and small in the region where we estimate $\hat{\boldsymbol{\xi}}$, and the determinant of K is almost constant of $\boldsymbol{\xi}$ for both the Gaussian kernel and linear kernel. In this sense, (3.5) and (3.16) are equivalent. However, our algorithm benefits greatly from omitting that term since taking the derivative of $\log|K|$ results in complicated expressions. Therefore, we assume the minimum of (3.16) is not affected much by $\log|K|$, and we can still consider our objective functions $Q(\boldsymbol{\xi})$ and $Q_0(\boldsymbol{\xi})$ as good approximations of the negative log likelihood function of $(\boldsymbol{\alpha},\boldsymbol{\xi})$ with and without a prior for $\boldsymbol{\xi}$, respectively.

Based on these arguments, considering the convexity and differentiability of $Q_0(\boldsymbol{\xi})$,

we can assume the regularity conditions of log likelihood are also applied to $Q_0(\boldsymbol{\xi})$ such that

$$\begin{aligned} &\lim_{n\to\infty} \mathbf{v}_n(\boldsymbol{\xi}^*) \to \mathbf{v}^*, \text{ and } \|\mathbf{v}^*\|_\infty < \infty, \\ &\lim_{n\to\infty} M_n(\boldsymbol{\xi}^*) \to M^*, \text{ and } \|M^*\|_\infty < \infty. \end{aligned} \tag{3.17}$$

Particularly

$$\lim_{n\to\infty} \mathbf{v}_n(\boldsymbol{\xi}^*) \to -\lim_{n\to\infty} \left\{ \frac{1}{2\sqrt{n}} \boldsymbol{\alpha}^{*T} K_j'(\boldsymbol{\xi}^*) \boldsymbol{\alpha}^* \right\}_{1\le j\le p}^{T} = \mathbf{v}^*. \tag{3.18}$$

Conditions (3.17-3.18) indicate that $\mathbf{v}_n(\boldsymbol{\xi}) = O_p(1)$ at $\boldsymbol{\xi}^*$, and $M_n(\boldsymbol{\xi})$ is finite and positive (semi-)definite at $\boldsymbol{\xi}^*$. These conditions are consistent with the convexity assumption of $Q_0(\boldsymbol{\xi})$ discussed in previous section.

3.3 An Efficient Algorithm

In this section, we provide an efficient algorithm for NGK models.

3.3.1 Backfitting Algorithm to Update ξ_j's

An efficient algorithm to achieve the regularization path of (3.5) for $(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\xi}})$ is still an open problem. A possible algorithm is iteratively updating between $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\xi}}$ until convergence from certain initial $(\hat{\boldsymbol{\alpha}}^{(0)}, \hat{\boldsymbol{\xi}}^{(0)})$. However, this updating approach could be very expensive and may not be able to converge. The one-step update algorithm proposed by Lin and Zhang (2006) for COSSO can be applied to our model, that is, at each fixed λ, solve $\hat{\boldsymbol{\xi}}$ with given $\hat{\boldsymbol{\alpha}}$, then update $\hat{\boldsymbol{\alpha}} = \Delta^{-1}\mathbf{y}$ with the new $\hat{\boldsymbol{\xi}}$ and continue to next step. However, a one-step update algorithm may not be necessary to solve the solution path of $\hat{\boldsymbol{\xi}}$ as long

as we have some consistent initial estimation of $\tilde{\boldsymbol{\alpha}}$ and keep it fixed through the entire solution path.

In Section 3.4 we will show theoretically that given $\tilde{\boldsymbol{\alpha}}$ is consistent, the sparsity of $\boldsymbol{\xi}$ can be recovered as n increases. Thus we propose a nonlinear version of the nonnegative garrote updating algorithm, which is similar to the linear nonnegative garrote algorithm in the sense that $\hat{\boldsymbol{\xi}}$ depends on some initial function components (Yuan and Lin, 2007). Although the initial $\boldsymbol{\alpha}$-fixed algorithm may not results in a consistent estimation of $\boldsymbol{\xi}$, in Section 3.4, we will also show that under certain conditions, the estimation consistency of $\boldsymbol{\xi}$ can be achieved.

With fixed initial consistent $\tilde{\boldsymbol{\alpha}}$, the algorithm to update $\boldsymbol{\xi}$ becomes efficient. In some cases, such as with the linear kernel or other additive multiple kernels, our algorithm is equivalent to the least angle regression selection (LARS) algorithm on the linear nonnegative garrote model proposed by Yuan and Lin (2007). Furthermore, our algorithm can be included into the backfitting framework.

The proposed algorithm is as follows:

- *Step* 1 Initialize $\tilde{\boldsymbol{\alpha}}$ and $\lambda_0 = \sigma^2/\sigma_\alpha^2$ by setting all $\xi_j = \rho$ and fitting the least squares kernel machine by MLE/REML method with the Newton-Raphson algorithm, which estimates parameters $\sigma^2, \sigma_\alpha^2$ and ρ, then $\tilde{\boldsymbol{\alpha}} = (\lambda_0 I + K(\rho))^{-1}\mathbf{y}$.

- *Step* 2 Determine the initial λ with which all $\hat{\xi}_j^{(0)} = 0$:

$$\lambda^{(0)} = \max_j \left\{ n^{-1} \left(\tilde{\mathbf{y}} - K(0)\tilde{\boldsymbol{\alpha}}\right)^T \left(K_j'(0)\tilde{\boldsymbol{\alpha}}\right) \right\},$$

  where $\tilde{\mathbf{y}} = \mathbf{y} - \frac{\lambda_0}{2}\tilde{\boldsymbol{\alpha}}$.

- *Step* 3 Update $\hat{\boldsymbol{\xi}}$ coordinately at $\lambda^{(k+1)}$ with given $\tilde{\boldsymbol{\alpha}}$ by the following equation until

converge:

$$\hat{\xi}_j = \left[\tilde{\xi}_j + \frac{(\tilde{\mathbf{y}} - K\tilde{\boldsymbol{\alpha}})^T K_j' \tilde{\boldsymbol{\alpha}} - n\lambda^{(k+1)}}{\left(K_j'\tilde{\boldsymbol{\alpha}}\right)^T \left(K_j'\tilde{\boldsymbol{\alpha}}\right)} \right]_+, \tag{3.19}$$

where $\tilde{\xi}_j$ denotes previously updated $\hat{\xi}_j$, and K and K_j' are calculated from the previously updated $\tilde{\xi}_j$'s.

• *Step* 4 Decrease λ, repeat step 3.

• *Step* 5 Stop when model selection criterion reaches minimum or turning point, or $\lambda = 0$.

In Section 3.4, we will consider the linear approximation assumption of the kernel to derive the consistency conditions for $\hat{\boldsymbol{\xi}}$. However, in practice this assumption is too strong to develop an efficient algorithm to update $\hat{\boldsymbol{\xi}}$. We derive the updating equation (3.19) for $\hat{\boldsymbol{\xi}}$ by a different approximation of $K(\boldsymbol{\xi})$. For a given λ, assuming the current iteration $\tilde{\boldsymbol{\xi}}$ is close to the minimum solution $\hat{\boldsymbol{\xi}}$, the kernel matrix can be extended in one coordinate direction around $\tilde{\xi}_j$:

$$K(\tilde{\boldsymbol{\xi}}_{-j}, \hat{\xi}_j) = K(\tilde{\boldsymbol{\xi}}) + (\hat{\xi}_j - \tilde{\xi}_j) \left(\frac{\partial K}{\partial \xi_j} \right)_{\tilde{\boldsymbol{\xi}}} + O(\|\hat{\xi}_j - \tilde{\xi}_j\|^2).$$

where the suffix $(-j)$ denotes without ξ_j.

In a simple notation

$$K(\tilde{\boldsymbol{\xi}}_{-j}, \hat{\xi}_j) \approx K + (\hat{\xi}_j - \tilde{\xi}_j) K_j', \tag{3.20}$$

where $K = K(\tilde{\boldsymbol{\xi}})$ and $K_j' = \left(\frac{\partial K}{\partial \xi_j}\right)_{\tilde{\boldsymbol{\xi}}}$. For example, for a Gaussian kernel $K_j' = K \circ D^j$ and for a linear kernel $K_j' = D^j$, where "$\circ$" denotes the Schur product or entrywise product of two matrices.

The updated solution of $\hat{\xi}_j$ given $\tilde{\boldsymbol{\xi}}$ is achieved by plugging (3.20) into (3.5), and solving $\hat{\xi}_j = \arg\min (3.5)$ given $\tilde{\boldsymbol{\alpha}}$ and λ_0.

As mentioned before, we note that Expression (3.19) is similar to the backfitting algorithm (Hastie and Tibshirani, 1990; Ravikumar et al., 2009) in a nonparametric additive model except our algorithm is a version of backfitting on a nonadditive models by performing the ξ_j's updating step as:

• *Step* a Initialize $\hat{\xi}_j = \hat{\xi}_j^{(k)}, j = 1, ..., p$, with $\tilde{\boldsymbol{\alpha}}$ given.

• *Step* b (1) Compute the residual, $\mathbf{r}_j = \tilde{\mathbf{y}} - K\tilde{\boldsymbol{\alpha}} + \tilde{\xi}_j K_j' \tilde{\boldsymbol{\alpha}}$.

(2) Project the residual onto $\mathbf{z}_j = K_j' \tilde{\boldsymbol{\alpha}}$, $P_j = \mathbf{z}_j^T \mathbf{r}_j$.

(3) Update the soft threshold, $\hat{\xi}_j = \left(\frac{P_j - n\lambda}{\|\mathbf{z}_j\|^2} \right)_+$.

• *Step* c Continue Step b until the individual $\hat{\xi}_j$'s do not change.

When we use a linear kernel, $K = \sum \xi_j \mathbf{x}_j \mathbf{x}_j^T$, our problem becomes the additive kernel case, $K = \sum \xi_j K_j$, which has been thoroughly discussed by Bach (2008) in the MKL with LASSO. For the additive kernel case, our method can be applied to the functional ANOVA models. Yuan and Lin (2007) proposed the linear nonnegative garrote component selection method in these models, and introduced the LARS algorithm for linear nonnegative garrote when $p < n$. When $p > n$, the LARS algorithm may not be unique because the correlation matrix of the active variable set may not be invertible. Therefore, LARS may end up with non-unique results by using the general inverse matrix. On the other hand, our algorithm has two main advantages: it works for $p > n$, and it works with nonadditive kernels while in both cases linear nonnegative garrote algorithm fails. In addition, our algorithm is also related to ARD which is mostly discussed in Bayesian approaches (Neal, 1996; Krishnapuram et al., 2004; Zou et al., 2010). To the best of our knowledge, our algorithm is the only non-Bayesian approach for determining those hyper parameters in ARD with penalty on $\boldsymbol{\xi}$ and it is more efficient than Bayesian methods.

3.3.2 Model Selection Criterion

As discussed before, NGK variable selection is an interesting but rather new topic within the kernel machine framework. No similar work provides a perfect criterion to select the penalty parameter λ. We propose using the Bayesian information criterion (BIC) as Liu et al. (2007) suggested for least squares kernel machine mixed modeling.

For given minimum solution $\hat{\boldsymbol{\xi}}$, the estimated function for $\mathbf{f}$ can be expressed as $\hat{\mathbf{f}} = S\mathbf{y}$, where S is the smoothing matrix. For the least squares error kernel machine, $S = K(\hat{\boldsymbol{\xi}}) \left(\lambda_0 I + K(\hat{\boldsymbol{\xi}})\right)^{-1}$, and the degree of freedom of the kernel machine smoother S is defined as $df = \text{Trace}(S)$. The least squares kernel machine BIC is defined as

$$BIC = \log(RSS) + \frac{df\ \log(n)}{n},$$

where the residual sum of squares, $RSS = (\mathbf{y} - \hat{\mathbf{f}})^T(\mathbf{y} - \hat{\mathbf{f}})$. Note if we have an intercept term in (3.1), we may modify the smoothing matrix S and add 1 to df to count the degree of freedom of the intercept parameter, but little will change.

To select λ, a smaller BIC is preferred. However, in practice, most of the popular model selection criteria like BIC, Cp and GCV may face the dilemma that these criteria become flat and difficult to determine an appropriate minimum. One approach is to choose λ at the turning point of the criterion with a simpler model (less number of predictors). Another issue is that the performance of a criterion not only depends on the model but also on the data structure. For instance, in the same dataset a criterion may work well with a small number of predictors but may over fit the model when a larger number of predictors are included.

The theoretical understanding of the performance of the those criteria in kernel ma-

chine variable selection approaches is still an interesting but challenging job. In the data analysis sections, we will not choose the variables depending on a single run. Rather, we propose two different resampling procedures to obtain the variable selection probability. Based on these procedures, we decide on the final set of selected variables by the selection probability. Further theoretical work may be required for these resampling procedures for NGK, but there are some similar theoretical works on LASSO (Chatterjee and Lahiri, 2011; Hall et al., 2009). It turns out that using such resampling procedures is more powerful.

3.4 Some Theoretical Properties

In recent years, tremendous progress has been made to understand the mechanism of variable selection in linear LASSO. Those theoretical works focus on consistency analysis of LASSO estimation. As pointed out by Zhao and Yu (2006), consistency of variable selection includes two aspects: estimation consistency and model selection consistency. Between these two concepts of consistency, one does not necessarily imply the another. The former requires $\|\hat{\boldsymbol{\xi}} - \boldsymbol{\xi}^*\| \to \mathbf{0}$ as $n \to \infty$, and the later requires $\lim_{n\to\infty} P(\hat{\mathcal{A}} = \mathcal{A}) \to 1$. The model consistency is also referred to as the sparsistency, shorthand for "sparsity pattern consistency" (Ravikumar et al., 2009). Similar to the consistency of LASSO, in this section we first show that under certain conditions the NGK estimator is $\sqrt{n}$ consistent. Then we will further discuss under which conditions that the NGK estimators are sparsistent given the initial fixed $\tilde{\boldsymbol{\alpha}}$. This is important because in our NGK algorithm, we assume the initial $\tilde{\boldsymbol{\alpha}}$ is fixed.

3.4.1 Necessary and Sufficient Conditions for the Consistency of the NGK Estimator

First, we establish the $\sqrt{n}$ consistency of the NGK estimator, $\hat{\boldsymbol{\xi}}$.

Theorem 3.4.1. *Under the regularity conditions (3.17), if* $\sqrt{n}\lambda \to 0$, *then there exists a local maximimum* $\hat{\boldsymbol{\xi}}$ *of* $Q(\boldsymbol{\xi})$ *such that* $\|\hat{\boldsymbol{\xi}} - \boldsymbol{\xi}^*\| = O_p(n^{-1/2})$.

The proof of Theorem 3.4.1 is similar to Fan and Li (2001) and Wang and Leng (2007), where both used the regularity conditions of the log likelihood function. We will show the proof of Theorem 3.4.1 in Section 3.5.2. The $\sqrt{n}$ estimation consistency of $\hat{\boldsymbol{\xi}}$ indicates that when n is large enough, the minimum solution of (3.12) is consistent with $\boldsymbol{\xi}^*$. This means when n is sufficiently large, the kernel matrix $K(\hat{\boldsymbol{\xi}})$ is close to $K(\boldsymbol{\xi}^*)$, and estimation $\hat{\boldsymbol{\alpha}} = \Delta^{-1}(\hat{\boldsymbol{\xi}})\mathbf{y}$ is also some δ consistent estimation of $\boldsymbol{\alpha}^*$ with $\delta \to 0$.

Before we derive the conditions for the sparsistency of $\hat{\boldsymbol{\xi}}$, let's first look into the case of linear LASSO regularization, $\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\beta}\|^2 + n\lambda\|\boldsymbol{\beta}\|_1$. It has been shown that $\hat{\boldsymbol{\beta}}$ is $\sqrt{n}$ consistent, but the $\sqrt{n}$ convergence rate of $\hat{\boldsymbol{\beta}}$ does not guarantee the oracle properties (Fan and Li, 2001). In linear LASSO, for $\hat{\boldsymbol{\beta}}$ to possess the consistency properties, the necessary and sufficient conditions are (including the error effect, see Wainwight (2009)):

$$\begin{aligned} \left| X_0^T X_1 (X_1^T X_1)^{-1} \left[\frac{1}{n} X_1^T \boldsymbol{\epsilon} - \lambda \text{sgn}(\boldsymbol{\beta}_1^*) \right] - \frac{1}{n} X_0^T \boldsymbol{\epsilon} \right| &\preceq \lambda \mathbf{1}, \\ \left| \boldsymbol{\beta}_1^* + \left(\frac{1}{n} X_1^T X_1 \right)^{-1} \left[\frac{1}{n} X_1^T \boldsymbol{\epsilon} - \lambda \text{sgn}(\boldsymbol{\beta}_1^*) \right] \right| &\succ \mathbf{0}. \end{aligned} \tag{3.21}$$

Note on both sides of the inequalities are vectors, and "$\preceq$"and "$\succ$" stand for the general inequality for vectors. If ignore the error effect, say $\boldsymbol{\epsilon} = \mathbf{0}$, then the necessary and

sufficient conditions for the linear lasso estimator become

$$\begin{aligned} \left|X_0^T X_1 (X_1^T X_1)^{-1} \text{sgn}(\boldsymbol{\beta}_1^*)\right| &\preceq \mathbf{1}, \\ \left|\boldsymbol{\beta}_1^* - \lambda \left(\frac{1}{n} X_1^T X_1\right)^{-1} \text{sgn}(\boldsymbol{\beta}_1^*)\right| &\succ \mathbf{0}, \end{aligned} \tag{3.22}$$

which were first shown independently by Zou (2006), Yuan and Lin (2007), and Zhao and Yu (2006). In the above equations $\text{sgn}(\boldsymbol{\beta}_1^*)$ is the sign vector of the true nonzero $\boldsymbol{\beta}_1^*$ vector and the $\text{sgn}(\cdot)$ function is defined as

$$\text{sgn}(\beta_j) := \begin{cases} +1, & \text{if } \beta_j > 0 \\ -1, & \text{if } \beta_j < 0 \\ 0, & \text{if } \beta_j = 0. \end{cases}$$

A similar sign function can be applied to $\boldsymbol{\xi}$

$$\text{sgn}(\xi_j) := \begin{cases} +1, & \text{if } \xi_j > 0 \\ 0, & \text{if } \xi_j = 0. \end{cases}$$

Then a similar argument has been shown for additive nonparametric regression and MKL (Ravikumar et al., 2009; Bach, 2008). Wainwight (2009) further explored those conditions for LASSO and provided a sharper threshold for the convergence rate. In our model, we can show similar results. The following lemma states the necessary and sufficient conditions for $\hat{\boldsymbol{\xi}}$ to be consistent.

Lemma 3.4.2. *Given initial $\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha}^*$, the necessary and sufficient conditions for $\hat{\boldsymbol{\xi}}$ to be consistent,*

i.e., $\lim_n P(\hat{\mathcal{A}} = \mathcal{A}) = 1$ *or* $\lim_n P\{sgn(\hat{\boldsymbol{\xi}}) = sgn(\boldsymbol{\xi}^*)\} = 1$, *are*

$$\frac{1}{n} Z_0^T \left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2} \boldsymbol{\alpha}^* \right) - Z_0^T Z_1 (Z_1^T Z_1)^{-1} \left[\frac{1}{n} Z_1^T \left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2} \boldsymbol{\alpha}^* \right) - \lambda \mathbf{1} \right] \preceq \lambda \mathbf{1}, \tag{3.23a}$$

$$\boldsymbol{\xi}_1^* + \left(\frac{1}{n} Z_1^T Z_1 \right)^{-1} \left[\frac{1}{n} Z_1^T \left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2} \boldsymbol{\alpha}^* \right) - \lambda \mathbf{1} \right] \succ \mathbf{0}. \tag{3.23b}$$

Note in the above expressions, $\mathbf{1}$ is a vector with all 1's (size different for (3.23a) and (3.23b)).

To prove Lemma 3.4.2, we need an approximated form of (3.5). According to Theorem 3.4.1, $\hat{\boldsymbol{\xi}}$ is $\sqrt{n}$-consistent, i.e., $\hat{\boldsymbol{\xi}} \to \boldsymbol{\xi}^*$ as $n \to \infty$. The linear approximation of the kernel function holds: $K(\hat{\boldsymbol{\xi}}) = K(\boldsymbol{\xi}^*) + \sum_{j=1}^p (\hat{\xi}_j - \xi_j^*) K_j'(\boldsymbol{\xi}^*) + O_p(\|\hat{\boldsymbol{\xi}} - \boldsymbol{\xi}^*\|^2)$. Given $\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha}^*$, $\mathbf{y} - \mathbf{f} = \mathbf{y} - K(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* = \boldsymbol{\epsilon}$. Plug $\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha}^*$ and the approximated $K(\hat{\boldsymbol{\xi}})$ back into the expression of $Q(\boldsymbol{\xi})$ in (3.5), and the regularization problem is then approximated as

$$\frac{1}{2} \left\| \mathbf{y} - K(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* - \sum_{j=1}^p (\hat{\xi}_j - \xi_j^*) K_j'(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* \right\|^2 + \frac{\lambda_0}{2} \sum_{j=1}^p (\hat{\xi}_j - \xi_j^*) \boldsymbol{\alpha}^{*T} K_j'(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* + n\lambda \sum_{j=1}^p \hat{\xi}_j. \tag{3.24}$$

With the notation $\tilde{\mathbf{y}} = \mathbf{y} - K(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* - \frac{\lambda_0}{2}\boldsymbol{\alpha}^* + \sum_{j=1}^p \xi_j^* K_j'(\boldsymbol{\xi}^*)\boldsymbol{\alpha}^* = \boldsymbol{\epsilon} - \frac{\lambda_0}{2}\boldsymbol{\alpha}^* + Z\boldsymbol{\xi}^*$, rearranging the above expression results in an equivalent expression

$$\frac{1}{2} \left\| \tilde{\mathbf{y}} - Z\hat{\boldsymbol{\xi}} \right\|^2 + n\lambda \sum_{j=1}^p \hat{\xi}_j. \tag{3.25}$$

Expression (3.25) is similar to the linear nonnegative garrotte form (3.7) proposed by Yuan and Lin (2007) except the response $\tilde{\mathbf{y}}$ is modified with a nonlinear term $\frac{\lambda_0}{2}\boldsymbol{\alpha}^*$. Note that the above form is the exact result for the linear kernel model without the linear approximation of the kernel. Note that we use this approximation only for the purpose of theoretical analysis. When we derive the algorithm, we only approximate the kernel in one ξ_j direc-

tion. Thus we can not derive a form similar to (3.25) for Gaussian kernels because the $\tilde{\mathbf{y}}$ vector and the Z matrix are no longer fixed, but are updated by $\hat{\boldsymbol{\xi}}$ each time (See Section 3.3 for the algorithm). For a linear kernel, because the kernel is a linear combination of multiple kernels, we can derive the exact linear negative garrote form as above without the approximation.

When the minimum solution of $\hat{\boldsymbol{\xi}}$ is close to $\boldsymbol{\xi}^*$ as Theorem 3.4.1 states, the solution of (3.25) is consistent to the solution of (3.5). Thus we can start from (3.25) to derive the incoherence conditions as (3.23a-3.23b) (see Section 3.5.3).

3.4.2 Recovery of Sparsity

Note in Lemma 3.4.2, conditions (3.23a-3.23b) are derived with the initial $\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha}^*$. However, in practice we consider a δ-consistent $\tilde{\boldsymbol{\alpha}}$ and $\tilde{Z}$ matrix. A question arises about whether or not we can similarly solve $\hat{\boldsymbol{\kappa}}_0$ and $\hat{\boldsymbol{\xi}}_1$ based on $\tilde{\boldsymbol{\alpha}}$ by

$$\lambda\hat{\boldsymbol{\kappa}}_0 = \frac{1}{n}\tilde{Z}_0^T\left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2}\tilde{\boldsymbol{\alpha}}\right) - \tilde{Z}_0^T\tilde{Z}_1(\tilde{Z}_1^T\tilde{Z}_1)^{-1}\left[\frac{1}{n}\tilde{Z}_1^T\left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2}\tilde{\boldsymbol{\alpha}}\right) - \lambda\mathbf{1}\right], \tag{3.26a}$$

and

$$\hat{\boldsymbol{\xi}}_1 = \boldsymbol{\xi}_1^* + \left(\frac{1}{n}\tilde{Z}_1^T\tilde{Z}_1\right)^{-1}\left[\frac{1}{n}\tilde{Z}_1^T\left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2}\tilde{\boldsymbol{\alpha}}\right) - \lambda\mathbf{1}\right], \tag{3.26b}$$

such that we can use them to recover sparsity of $\boldsymbol{\xi}^*$, where $\hat{\boldsymbol{\kappa}}_0$ is the subgradient of $\|\hat{\boldsymbol{\xi}}\|_1$ corresponding to those $\hat{\xi}_j = 0$ (see Section 3.5.3). To show this with expressions (3.26a-3.26b), we consider some additional conditions required for $\tilde{\boldsymbol{\alpha}}$, such as how fast it converges to $\boldsymbol{\alpha}^*$.

The above argument indicates that, if we have a consistent estimate of $\boldsymbol{\alpha}^*$, we are able to recover sparsity of $\boldsymbol{\xi}$ using (3.26a-3.26b). Thus we do not need to estimate $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\xi}}$

together at each step. This is the fundamental aspect of our algorithm with which we use some $\tilde{\boldsymbol{\alpha}}$ as the initial values and keep $\tilde{\boldsymbol{\alpha}}$ fixed for the whole solution path of $\boldsymbol{\xi}$. A similar situation has been discussed in the linear nonnegative garrote model by Yuan and Lin (2007).

Thus, motivated by the consistency conditions (3.23a-3.23b), we consider the following zero noise incoherence conditions on the Z matrix:

$$\Sigma_{01}\Sigma_{11}^{-1}\mathbf{1} - \frac{\lambda_0}{2n\lambda} Z_0^T P \boldsymbol{\alpha}^* \preceq (1-\gamma)\mathbf{1} \tag{3.27}$$

where $\gamma \in (0, 1]$, and $P = [I - Z_1(Z_1^T Z_1)^{-1} Z_1^T]$ is a projection matrix. Expressions (3.26a-3.26b) and (3.27) are calculated based on the true $\boldsymbol{\alpha}^*$. We can show that as long as $\tilde{\boldsymbol{\alpha}}$ is δ consistent with $\delta \to 0$, the similar condition is satisfied for $\tilde{\boldsymbol{\alpha}}$ based calculation (see the proof of Theorem 3.4.3 in Section 3.5.4)

$$\tilde{\Sigma}_{01}\tilde{\Sigma}_{11}^{-1}\mathbf{1} - \frac{\lambda_0}{2n\lambda} \tilde{Z}_0^T \tilde{P} \tilde{\boldsymbol{\alpha}} \preceq (1-\tilde{\gamma})\mathbf{1}, \tag{3.28a}$$

where $\tilde{\gamma}$ is some positive number $\tilde{\gamma} \in (0, 1]$. Further more, we need the following assumptions,

$$\Lambda_{min}(\tilde{\Sigma}_{11}) \geq \tilde{C}_{min} > 0, \tag{3.28b}$$

$$\tilde{\Sigma}_{01}\tilde{\Sigma}_{11}^{-1} \to \Sigma_{01}\Sigma_{11}^{-1} \text{ with rate no slower than } \delta, \tag{3.28c}$$

where $\Lambda_{min}(\cdot)$ denotes the minimum positive eigenvalue.

There are two interesting observations about the condition (3.27) or (3.28a). First, unlike the incoherence conditions of linear LASSO where Σ_{11} and Σ_{01} are the correlation matrices of predictors, in (3.27), they are the correlation matrices of $\mathbf{z}_j$'s, the vectors of first derivatives of initial $\mathbf{f} = K\boldsymbol{\alpha}^*$ with respect to ξ_j's. Second, (3.28a) can be understood by

nothing that the sparsity in NGK not only depends on how much the irrelevant Z_0 matrix is correlated with the matrix space of Z_1, but also depends on how the irrelevant Z_0 matrix is correlated to the nonlinear component $\boldsymbol{\alpha}^*$ projected onto the true matrix space of Z_1.

Now we can show the following major theorem:

Theorem 3.4.3. *Under the following conditions*

1 The initial estimate $\tilde{\boldsymbol{\alpha}}$ is δ consistent, i.e., $|\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*|_\infty = O_p(\delta)$ for some $\delta \to 0$, and

2 The conditions (3.28a-3.28c),

there exists some λ with $n\lambda^2 \to \infty$ such that for some constant $\eta_1 > 0$ we have the following results with probability $1 - \exp(-\eta_1 n\lambda^2) \to 1$

(a) $\hat{\mathcal{A}} \subseteq \mathcal{A}$ and the upper bound of $\|\hat{\boldsymbol{\xi}}_1 - \boldsymbol{\xi}_1^*\|_\infty$ converges to

$$\rho(\lambda) = \lambda \left[\frac{4\sigma}{\sqrt{\tilde{C}_{min}}} + \|\tilde{\Sigma}_{11}^{-1}\|_\infty \cdot \frac{\lambda_0}{\lambda} \left(n^{-1/2} \|\mathbf{v}^*\|_\infty + O_p(\delta) \right) + \|\tilde{\Sigma}_{11}^{-1}\|_\infty \right]$$

(b) If $\rho(\lambda) < \min_{j \in \mathcal{A}} \xi_j^*$, then we have the sparsistency of $\hat{\boldsymbol{\xi}}$, i.e., $\hat{\mathcal{A}} = \mathcal{A}$.

This result generalizes Theorem 1 of Yuan and Lin (2007) on the consistency of the linear nonnegative garrote to nonadditive models. The proof steps are similar to the ones in Wainwright (2009) and Ravikumar et al. (2009) using the technique of a primal dual witness on model selection consistency. In Theorem 3.4.3 we use the assumption that $Z_1^T Z_1$ is invertible. Without this assumption the solutions to $\hat{\boldsymbol{\xi}}_1$ and $\hat{\boldsymbol{\kappa}}_0$ are not unique. In Theorem 3.4.3, λ is required to be greater than $\sqrt{\frac{\log p}{n}} \cdot C$, where C is some constant determined by δ, σ^2, and γ, so that $\exp(-\eta_1 n\lambda^2) \to 0$ as $n\lambda^2 \to \infty$ and (a) is satisfied.

This places some limitation on λ such that λ can not be artificially small. Nevertheless, according to (a) in Theorem 3.4.3, if we have the addition of $\lambda + \lambda_0\sqrt{a/n} + O_p(\lambda_0\sqrt{a}\delta) + \sqrt{a}\lambda \to 0$, then $\|\hat{\boldsymbol{\xi}} - \boldsymbol{\xi}^*\| \to 0$ which means we can have estimation consistency too.

3.5 Proofs of the Lemmas and Theorems

3.5.1 Proof of Lemma 3.2.2

Proof: This is a result of the composition theorem (Boyd and Vandenberghe 2004 Chp.3). For function $Q_0(\boldsymbol{\xi}) = Q_0\left(K(\boldsymbol{\xi})\right)$ with domain $\mathbf{dom}\, Q_0(\boldsymbol{\xi}) = \{\boldsymbol{\xi} \in \mathbf{dom}\, K(\boldsymbol{\xi}) | K(\boldsymbol{\xi}) \in \mathbf{dom}\, Q_0(K)\}$, if $Q_0(K)$ is convex and non-increasing and $K(\boldsymbol{\xi})$ is concave, then $Q_0(\boldsymbol{\xi})$ is convex. To see this, assuming $\boldsymbol{\xi}, \boldsymbol{\xi}' \in \mathbf{dom}\, Q_0(\boldsymbol{\xi})$, we have $\boldsymbol{\xi}, \boldsymbol{\xi}' \in \mathbf{dom}\, K(\boldsymbol{\xi})$ and $K(\boldsymbol{\xi}), K(\boldsymbol{\xi}') \in \mathbf{dom}\, Q_0(K)$. Since $\mathbf{dom}\, K(\boldsymbol{\xi}) = \mathbb{R}_+^p$ is convex, we have $\theta\boldsymbol{\xi} + (1-\theta)\boldsymbol{\xi}' \in \mathbf{dom}\, K(\boldsymbol{\xi})$, and from the concavity of $K(\boldsymbol{\xi})$, we have

$$K(\theta\boldsymbol{\xi} + (1-\theta)\boldsymbol{\xi}') \succeq \theta K(\boldsymbol{\xi}) + (1-\theta)K(\boldsymbol{\xi}'). \tag{3.29}$$

Since $K(\boldsymbol{\xi}), K(\boldsymbol{\xi}') \in \mathbb{K}^* \subset \mathbf{dom}\, Q_0(K) = \mathbb{K}$, we conclude that $\theta K(\boldsymbol{\xi}) + (1-\theta)K(\boldsymbol{\xi}') \in \mathbf{dom}\, Q_0(K)$. Since $\theta\boldsymbol{\xi} + (1-\theta)\boldsymbol{\xi}' \in \mathbf{dom}\, K(\boldsymbol{\xi})$, we have $K(\theta\boldsymbol{\xi} + (1-\theta)\boldsymbol{\xi}') \in \mathbf{dom}\, Q_0(K)$ too, which means $\theta\boldsymbol{\xi} + (1-\theta)\boldsymbol{\xi}' \in \mathbf{dom}\, Q_0(\boldsymbol{\xi})$. Now using the fact $Q_0(K)$ is nonincreasing and (3.29), we have

$$Q_0\left(K(\theta\boldsymbol{\xi} + (1-\theta)\boldsymbol{\xi}')\right) \leq Q_0\left(\theta K(\boldsymbol{\xi}) + (1-\theta)K(\boldsymbol{\xi}')\right). \tag{3.30}$$

Because of the convexity of $Q_0(K)$, we have

$$Q_0\left(\theta K(\boldsymbol{\xi}) + (1-\theta)K(\boldsymbol{\xi}')\right) \leq \theta Q_0\left(K(\boldsymbol{\xi})\right) + (1-\theta)Q_0\left(K(\boldsymbol{\xi}')\right). \tag{3.31}$$

Combining the above two inequations, we get

$$Q_0\left(\theta\boldsymbol{\xi} + (1-\theta)\boldsymbol{\xi}'\right) \leq \theta Q_0\left(\boldsymbol{\xi}\right) + (1-\theta)Q_0\left(\boldsymbol{\xi}'\right), \tag{3.32}$$

which proves the convexity of $Q_0(\boldsymbol{\xi})$. Since $\|\boldsymbol{\xi}\|_1$ is convex function of $\boldsymbol{\xi}$, we have the convexity of $Q(\boldsymbol{\xi})$.

3.5.2 Proof of Theorem 3.4.1

Proof: We use Expression (3.12) to prove Theorem 3.4.1. Following Fan and Li (2001), to show the existence of an $d_n = n^{-1/2}$-consistent local minimum in the ball $\boldsymbol{\xi}^* + d_n\mathbf{u} : \|\mathbf{u}\| \leq C$, we need to show for any given $\epsilon > 0$, there exists a large enough constant C, such that

$$\liminf_n P\left\{\inf_{\|\mathbf{u}\|=C} Q(\boldsymbol{\xi}^* + d_n\mathbf{u}) > Q(\boldsymbol{\xi}^*)\right\} \geq 1-\epsilon. \tag{3.33}$$

Calculating the following expression:

$$\begin{aligned}
&Q(\boldsymbol{\xi}^* + d_n\mathbf{u}) - Q(\boldsymbol{\xi}^*) \\
&\approx d_n\left(\frac{\partial Q_0}{\partial\boldsymbol{\xi}}\right)^T_{\boldsymbol{\xi}^*}\mathbf{u} + \frac{d_n^2}{2}\mathbf{u}^T\left(\frac{\partial^2 Q_0}{\partial\boldsymbol{\xi}^T\partial\boldsymbol{\xi}}\right)_{\boldsymbol{\xi}^*}\mathbf{u} + nd_n\lambda(\|\boldsymbol{\xi}^* + d_n\mathbf{u}\| - \|\boldsymbol{\xi}^*\|) \\
&\geq \sqrt{n}\lambda_0 d_n\mathbf{v}_n(\boldsymbol{\xi}^*)^T\mathbf{u} + \frac{n\lambda_0 d_n^2}{2}\mathbf{u}^T M_n(\boldsymbol{\xi}^*)\mathbf{u} + nd_n\lambda\sum_{i=1}^{a}\left(|\xi_i + d_n u_i| - |u_i|\right) \\
&\geq \lambda_0\mathbf{v}_n^T(\boldsymbol{\xi}^*)\mathbf{u} + \frac{\lambda_0}{2}\mathbf{u}^T M_n(\boldsymbol{\xi}^*)\mathbf{u} - \sqrt{n}\lambda a\|\mathbf{u}\|.
\end{aligned} \tag{3.34}$$

Using the regularity conditions of (3.17), we note for $\|\mathbf{u}\| = C$, on the right hand side (RHS) of (3.34) $\mathbf{v}_n^T = O_p(1)$. Thus the first term on RHS of (3.34) is uniformly bounded by the second term for C sufficiently large. To see this, we notice at $\|\mathbf{u}\| = C$, $0.5\mathbf{u}^T M_n \mathbf{u}$ is uniformly larger than $0.5\Lambda_{min}(M_n)C^2$, which is a quadratic function of C because M_n is finite positive (semi-)definite, and $\|\mathbf{v}_n^T\mathbf{u}\| \leq \|\mathbf{v}_n^T\|C$ which is a linear function of C since $\|\mathbf{v}_n^T\| = O_p(1)$. For sufficiently large C, the quadratic form of C always dominates the linear form of C. As $n \to \infty$, we assume $\sqrt{n}\lambda \to 0$. Thus the last term on RHS of (3.34) is also bounded by the second term. Hence with a sufficiently large C, (3.33) holds.

3.5.3 Proof of Lemma 3.4.2

Proof: To continue the proof, we notice (3.25) is a convex function of $\boldsymbol{\xi}$. By the Karush-Kuhn-Tucker conditions for the optimality in a convex problem, the point $\hat{\boldsymbol{\xi}} \in \mathbb{R}_+^p$ is optimal if and only if there exists a subgradient $\hat{\boldsymbol{\kappa}} \in \partial(\|\hat{\boldsymbol{\xi}}\|_1)$ such that

$$\left.\frac{\partial \tilde{Q}_0}{\partial \boldsymbol{\xi}}\right|_{\hat{\boldsymbol{\xi}}} + n\lambda\hat{\boldsymbol{\kappa}} = 0 \tag{3.35}$$

where $\tilde{Q}_0$ is the first two terms of (3.24), and the collection of subgradient of $\|\hat{\boldsymbol{\xi}}\|_1$ at point $\hat{\boldsymbol{\xi}}$ is the subdifferential $\partial(\|\hat{\boldsymbol{\xi}}\|_1)$:

$$\partial(\|\hat{\boldsymbol{\xi}}\|_1) = \{\hat{\boldsymbol{\kappa}} \in \mathbb{R}^p : \hat{\kappa}_j = 1 \text{ for } \hat{\xi}_j > 0; \hat{\kappa}_j \leq 1 \text{ for } \hat{\xi}_j = 0\}. \tag{3.36}$$

Plugging (3.25) in (3.35), after simple algebra, we get

$$Z^T Z(\hat{\boldsymbol{\xi}} - \boldsymbol{\xi}^*) - Z^T\boldsymbol{\epsilon} + \frac{\lambda_0}{2}Z^T\boldsymbol{\alpha}^* + n\lambda\hat{\boldsymbol{\kappa}} = 0. \tag{3.37}$$

Suppose $\lim_{n\to\infty} P(\hat{\mathcal{A}} = \mathcal{A}) \to 1$ or $\lim_{n\to\infty} P(\text{sgn}(\hat{\boldsymbol{\xi}}) = \text{sgn}(\boldsymbol{\xi}^*)) \to 1$, then

$$\hat{\boldsymbol{\xi}}_1 \succ \mathbf{0}, \hat{\boldsymbol{\xi}}_0 = \mathbf{0}, \text{ and } \hat{\boldsymbol{\kappa}}_1 = \mathbf{1}, \hat{\boldsymbol{\kappa}}_0 \preceq \mathbf{1}.$$

Substituting those observations into (3.37) and rearranging, we can solve $\hat{\boldsymbol{\kappa}}_0$ and $\hat{\boldsymbol{\xi}}_1$ as

$$\frac{1}{n} Z_0^T \left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2} \boldsymbol{\alpha}^* \right) - Z_0^T Z_1 (Z_1^T Z_1)^{-1} \left[\frac{1}{n} Z_1^T \left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2} \boldsymbol{\alpha}^* \right) - \lambda \mathbf{1} \right] = \lambda \hat{\boldsymbol{\kappa}}_0, \tag{3.38a}$$

$$\boldsymbol{\xi}_1^* + \left(\frac{1}{n} Z_1^T Z_1 \right)^{-1} \left[\frac{1}{n} Z_1^T \left(\boldsymbol{\epsilon} - \frac{\lambda_0}{2} \boldsymbol{\alpha}^* \right) - \lambda \mathbf{1} \right] = \hat{\boldsymbol{\xi}}_1. \tag{3.38b}$$

Considering conditions (3.36) for $\hat{\boldsymbol{\kappa}}_0 \preceq \mathbf{1}$ and $\hat{\boldsymbol{\xi}}_1 \succ \mathbf{0}$, we get the necessary and sufficient conditions of (3.23a) and (3.23b).

3.5.4 Proof of Theorem 3.4.3

Proof: Condition $|\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*|_\infty = O_p(\delta)$ denotes that $|\tilde{\alpha}_k \tilde{\alpha}_l - \alpha_k^* \alpha_l^*| \leq (|\tilde{\alpha}_k| + |\alpha_l^*|)|\tilde{\alpha}_l - \alpha_l^*| = O_p(\delta)$ for $1 \leq k, l \leq n$, thus we have following relationships

$$\begin{aligned} n^{-1} \tilde{\mathbf{z}}_i^T \tilde{\mathbf{z}}_j &= n^{-1} \mathbf{z}_i^T \mathbf{z}_j + O_p(\delta), \\ n^{-1} \tilde{\mathbf{z}}_j^T \tilde{\boldsymbol{\alpha}} &= n^{-1} \mathbf{z}_j^T \boldsymbol{\alpha}^* + O_p(\delta), \end{aligned} \tag{3.39}$$

where $1 \leq i, j \leq p$. Above relationships are derived based on the following two inequali-

ties:

$$\begin{aligned}
\left|\frac{1}{n}\left\{\tilde{\mathbf{z}}_i^T\tilde{\mathbf{z}}_j - \mathbf{z}_i^T\mathbf{z}_j\right\}\right| &= \left|\frac{1}{n}\left\{\tilde{\boldsymbol{\alpha}}^T K_i' K_j' \tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^{*T} K_i' K_j' \boldsymbol{\alpha}^*\right\}\right| \\
&= \left|\frac{1}{n}\left\{\sum_{k,l}(K_i'K_j')_{k,l}(\tilde{\alpha}_k\tilde{\alpha}_l - \alpha_k^*\alpha_l^*)\right\}\right| \\
&\leq \frac{1}{n}\left\{\sum_{k,l}|K_i'K_j'|_{k,l}|\tilde{\alpha}_k\tilde{\alpha}_l - \alpha_k^*\alpha_l^*|\right\} \\
&\leq \frac{O_p(\delta)}{n}\sum_{k,l}|K_i'K_j'|_{k,l} = O_p(\delta)\frac{\mathbf{1}^T|K_i'K_j'|\mathbf{1}}{n} \\
&\leq O_p(\delta)\cdot C = O_p(\delta),
\end{aligned} \tag{3.40}$$

and

$$\begin{aligned}
\left|\frac{1}{n}\left\{\tilde{\mathbf{z}}_j^T\tilde{\boldsymbol{\alpha}} - \mathbf{z}_j^T\boldsymbol{\alpha}^*\right\}\right| &= \left|\frac{1}{n}\left\{\tilde{\boldsymbol{\alpha}}^T K_j' \tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^{*T} K_j' \boldsymbol{\alpha}^*\right\}\right| \\
&= \left|\frac{1}{n}\left\{\sum_{k,l}(K_j')_{k,l}(\tilde{\alpha}_k\tilde{\alpha}_l - \alpha_k^*\alpha_l^*)\right\}\right| \\
&\leq \frac{1}{n}\left\{\sum_{k,l}|K_j'|_{k,l}|\tilde{\alpha}_k\tilde{\alpha}_l - \alpha_k^*\alpha_l^*|\right\} \\
&\leq O_p(\delta)\frac{\mathbf{1}^T|K_j'|\mathbf{1}}{n} \\
&\leq O_p(\delta)\cdot C = O_p(\delta),
\end{aligned} \tag{3.41}$$

where C's are some small positive numbers. These inequalities are true for Gaussian kernels and linear kernels because X is standardized. For example for Gaussian kernels, we have $n^{-1}\mathbf{1}^T|K_j'|\mathbf{1} \leq 2$ where $|\cdot|$ is the componentwise absolute value. To see this, note $\mathbf{1}^T|K_j'|\mathbf{1} = \mathbf{1}^T|K \circ D^j|\mathbf{1} \leq \mathbf{1}^T|D^j|\mathbf{1}$ because all elements of K are positive and smaller than 1, and $\mathbf{1}^T|D^j|\mathbf{1} = \sum_{k,l}(x_{jk} - x_{jl})^2 = \sum_{k,l}(x_{jk}^2 + x_{jl}^2 - 2x_{jk}x_{jl}) = \sum_k(nx_{jk}^2 + 1) = 2n$. For linear kernels, $K_j' = D^j = \mathbf{x}_j\mathbf{x}_j^T$, thus $\mathbf{1}^T|K_j'|\mathbf{1} = \sum_{k,l}|x_{jk}x_{jl}| \leq (\sum_l |x_{jl}|)^2 \leq n\sum_l x_{jl}^2 = n$. In both cases we use $\sum_l x_{jl} = 0$ and $\sum_l x_{jl}^2 = 1$. Similarly, we can show the inequalities

for $n^{-1}\mathbf{1}^T|K_i'K_j'|\mathbf{1}$ which are bounded by some small numbers.

In addition, because of the conditions (3.27), (3.28b-3.28c) and the relationships (3.39), with $\delta \to 0$, the left hand side of (3.27) is only different from $\tilde{\Sigma}_{01}\tilde{\Sigma}_{11}^{-1}\mathbf{1} - \frac{\lambda_0}{2n\lambda}\tilde{Z}_0^T\tilde{P}\tilde{\boldsymbol{\alpha}}$ with $O_p(\delta)$ ($\tilde{P}$ is a projection matrix thus does not change much the norm). Thus for δ sufficiently small, Expression (3.28a),

$$\tilde{\Sigma}_{01}\tilde{\Sigma}_{11}^{-1}\mathbf{1} - \frac{\lambda_0}{2n\lambda}\tilde{Z}_0^T\tilde{P}\tilde{\boldsymbol{\alpha}} \preceq (1-\tilde{\gamma})\mathbf{1},$$

holds for some $\tilde{\gamma} \in (0, 1]$. The converse is true too: if (3.28a) is satisfied, then we can always find a small positive γ when $\delta \to 0$ the condition (3.27) is true. This equivalence allows us to show the sparsistency by using (3.28a).

Starting from (3.5.4), our argument parallels and uses similar technique of a primal dual witness on model selection consistency of LASSO Wainwight (2009), which contains the following steps:

1. Obtain $\hat{\boldsymbol{\xi}}_1$ by solving (3.26b), and set $\hat{\boldsymbol{\xi}}_0 = 0$.
2. Set $\hat{\boldsymbol{\kappa}}_1 = \partial(\|\boldsymbol{\xi}_1^*\|_1)$; for nonnegative garrotte $\hat{\boldsymbol{\kappa}}_1 = \mathbf{1}$.
3. With these setting of $\hat{\boldsymbol{\xi}}_1$ and $\hat{\boldsymbol{\kappa}}_1$, obtain $\hat{\boldsymbol{\kappa}}_0$ through (3.26a), and check whether or not $\hat{\boldsymbol{\kappa}}_0 \in \partial(\|\boldsymbol{\xi}_0^*\|_1)$; for nonnegative garrotte $\hat{\boldsymbol{\kappa}}_0 \prec \mathbf{1}$.
4. Check whether or not $\hat{\boldsymbol{\kappa}}_1 = \mathbf{1}$.

Lemma 2 in Wainwight (2009) states that if dual feasibility established (Step 1-3 succeed), then $\hat{\mathcal{A}} \subseteq \mathcal{A}$. In Step 3 using $\hat{\boldsymbol{\kappa}}_0 \prec \mathbf{1}$ instead of $\hat{\boldsymbol{\kappa}}_0 \preceq \mathbf{1}$ is to ensure uniqueness by strict dual feasibility. Furthermore, if Step 4 succeeds too, then $\hat{\mathcal{A}} = \mathcal{A}$.

Following Wainwight (2009) with the technical issues particular in nonnegative garrotte on kernels, we prove Theorem 3.4.3 in two steps. Given $\tilde{\boldsymbol{\alpha}}$ thus $\tilde{Z}$ defined as before, from (3.26a-3.26b), we define two random variables:

$$A_i := \tilde{\mathbf{z}}_i^T \left\{ \tilde{Z}_1 (\tilde{Z}_1^T \tilde{Z}_1)^{-1} \mathbf{1} - \frac{\lambda_0}{2n\lambda} \tilde{P} \tilde{\boldsymbol{\alpha}} \right\} + \frac{1}{n\lambda} \tilde{\mathbf{z}}_i^T \tilde{P} \boldsymbol{\epsilon}, \ i \in \mathcal{A}^c$$
$$\hat{\xi}_j - \xi_j^* := \mathbf{e}_j^T \tilde{\Sigma}_{11}^{-1} \left(\frac{1}{n} \tilde{Z}_1^T \boldsymbol{\epsilon} \right) - \mathbf{e}_j^T \tilde{\Sigma}_{11}^{-1} \left\{ \frac{\lambda_0}{2n} \tilde{Z}_1^T \tilde{\boldsymbol{\alpha}} + \lambda \mathbf{1} \right\}, \ j \in \mathcal{A}$$

where $\mathbf{e}_j^T$ is the selection vector with 1 in jth position.

- *Dual feasibility*

  Write A_i as $E(A_i) + A_i^*$, where $E(A_i) = \tilde{\mathbf{z}}_i^T \left\{ \tilde{Z}_1 (\tilde{Z}_1^T \tilde{Z}_1)^{-1} \mathbf{1} - \frac{\lambda_0}{2n\lambda} \tilde{P} \tilde{\boldsymbol{\alpha}} \right\}$, and $A_i^* = \frac{1}{n\lambda} \tilde{\mathbf{z}}_i^T \tilde{P} \boldsymbol{\epsilon}$. To have the subgradient vector $\hat{\boldsymbol{\kappa}}_0 \preceq \mathbf{1}$ is equivalent to show

$$\max_i A_i \leq 1.$$

  Using the definition of A_i and condition (3.5.4) we have

$$\max_i A_i \leq (1 - \tilde{\gamma}) + \max_i A_i^*$$

  A_i^* is a zero mean sub-Gaussian random variable, according to Wainwight (2009), the variance of A_i^* is bounded by

$$\begin{aligned} \text{Var}(A_i^*) = \frac{\sigma^2}{\lambda^2 n^2} (\tilde{\mathbf{z}}_i^T P \tilde{\mathbf{z}}_i) \leq \frac{\sigma^2}{\lambda^2 n^2} \|\tilde{\mathbf{z}}_i\|_2^2 &= \frac{\sigma^2}{\lambda^2 n} \left(n^{-1} \|\mathbf{z}_i\|_2^2 + O_p(\delta) \right) \\ &\leq \frac{\sigma^2}{\lambda^2 n} (1 + O_p(\delta)), \end{aligned}$$

  where we use the relationship (3.39), the properties of projection matrix and nor-

malized $\mathbf{z}_i$ vector such that $\|\mathbf{z}_i\|_2^2 \leq n$, and $\delta \to 0$.

By the sub-Gaussian tail bound results combined with the union bound (Wainwright, 2009), we have

$$\begin{aligned} P\left(\max_i A_i^* \geq \frac{\tilde{\gamma}}{2}\right) &\leq (p-a)\exp\left\{-\frac{(\tilde{\gamma}/2)^2}{2\sigma^2\lambda^{-2}n^{-1}[1+O_p(\delta)]}\right\} \\ &= \exp\left\{-\frac{\lambda^2 n\tilde{\gamma}^2}{8\sigma^2}(1+O_p(\delta))^{-1} + \log(p-a)\right\}. \end{aligned}$$

Putting all parts together, we conclude that

$$P\left(\max_i A_i > 1 - \frac{\tilde{\gamma}}{2}\right) \leq \exp\left(-\eta_1\lambda^2 n\right).$$

If we choose some λ such that $\frac{\lambda^2 n\tilde{\gamma}^2}{8\sigma^2(1+O_p(\delta))} > \log(p-a)$, say

$$\lambda > \frac{2}{\tilde{\gamma}}\sqrt{\frac{2\sigma^2\log p}{n}(1+O_p(\delta))}, \tag{3.42}$$

the probability for $\{\max_i A_i > 1 - \tilde{\gamma}/2\}$ vanishes with rate $\exp(-\eta_1\lambda^2 n)$ as $n \to \infty$. Or in other words, with probability $1 - \exp(-\eta_1\lambda^2 n)$ approach one, we have $\hat{\mathcal{A}} \subseteq \mathcal{A}$.

- *Bounding* $\|\hat{\boldsymbol{\xi}}_1 - \boldsymbol{\xi}_1^*\|_\infty$

  The upper bound of $\left\|\hat{\boldsymbol{\xi}}_1 - \boldsymbol{\xi}_1^*\right\|_\infty$ is

$$\|\hat{\boldsymbol{\xi}}_1 - \boldsymbol{\xi}_1^*\|_\infty \leq \underbrace{\left\|\tilde{\Sigma}_{11}^{-1}\left(\frac{1}{n}\tilde{Z}_1^T\boldsymbol{\epsilon}\right)\right\|_\infty}_{I} + \underbrace{\left\|\tilde{\Sigma}_{11}^{-1}\left(\frac{\lambda_0}{2n}\tilde{Z}_1^T\tilde{\boldsymbol{\alpha}}\right)\right\|_\infty}_{II} + \underbrace{\lambda\|\tilde{\Sigma}_{11}^{-1}\|_\infty}_{III}. \tag{3.43}$$

  Note the $\infty-$norm of matrix $\tilde{\Sigma}_{11}^{-1}$ is bounded as

$$\|\tilde{\Sigma}_{11}^{-1}\|_\infty \leq \sqrt{a}\tilde{C}_{min}^{-1}. \tag{3.44}$$

Thus, part III is bounded as

$$III := \lambda \|\tilde{\Sigma}_{11}^{-1}\|_\infty = \sqrt{a}\lambda \tilde{C}_{min}^{-1}.$$

Part II is bounded as

$$\begin{aligned} II :=& \left\| \tilde{\Sigma}_{11}^{-1} \left(\frac{\lambda_0}{2n} \tilde{Z}_1^T \tilde{\boldsymbol{\alpha}} \right) \right\|_\infty \leq \|\tilde{\Sigma}_{11}^{-1}\|_\infty \left\| \frac{\lambda_0}{2n} \tilde{Z}_1^T \tilde{\boldsymbol{\alpha}} \right\|_\infty \\ =& \|\tilde{\Sigma}_{11}^{-1}\|_\infty \left\| \frac{\lambda_0}{\sqrt{n}} \left(\frac{1}{2\sqrt{n}} Z_1^T \boldsymbol{\alpha}^* + O_p(\delta\sqrt{n}) \right) \right\|_\infty \\ \leq& \|\tilde{\Sigma}_{11}^{-1}\|_\infty \cdot \lambda_0 \left(n^{-1/2} \max_j |v_j^*| + O_p(\delta) \right), \end{aligned} \tag{3.45}$$

where we use (3.39) and (3.18) for $\mathbf{v}^*$. Using (3.44) we have

$$II \leq \frac{\sqrt{a}\lambda_0}{\tilde{C}_{min}} \left(n^{-1/2} \max_j |v_j^*| + O_p(\delta) \right). \tag{3.46}$$

Note in $\hat{\xi}_j - \xi_j^*$, the random part $U_j := \mathbf{e}_j^T \tilde{\Sigma}_{11}^{-1} (n^{-1} \tilde{Z}_1^T \boldsymbol{\epsilon})$ with $\boldsymbol{\epsilon} \sim N(0, \sigma^2 I)$, so U_j is zero mean Gaussian, i.e., $E(U_j) = 0$, and

$$\text{Var}(U_j) = \frac{\sigma^2}{n} \mathbf{e}_j^T \tilde{\Sigma}_{11}^{-1} \mathbf{e}_j \leq \frac{\sigma^2}{n} \tilde{C}_{min}^{-1}. \tag{3.47}$$

Again using the sub-Gaussian tail bound (Wainwright, 2009), we get

$$\begin{aligned} P\left(\max_j |U_j| > t \right) &\leq 2 \exp\left(-\frac{t^2 n}{2\sigma^2 \tilde{C}_{min}^{-1}} + \log a \right) \\ &= 2 \exp\left(-\frac{t^2 n}{2\sigma^2} \tilde{C}_{min} + \log a \right). \end{aligned} \tag{3.48}$$

Set $t = 4\sigma\lambda \tilde{C}_{min}^{-1/2}$, and by choosing λ as (3.42), we have $8\lambda^2 n > \log p \geq \log a$ so that $P\left(\max_j |U_j| > 4\sigma\lambda \tilde{C}_{min}^{-1/2} \right) \to 0$ with rate at least $2\exp(-\eta_2 \lambda^2 n)$ where $\eta_2 > 0$. And

we have the upper bound

$$\begin{aligned}\|\hat{\boldsymbol{\xi}}_1 - \boldsymbol{\xi}_1^*\|_\infty &\leq \lambda \left[\frac{4\sigma}{\sqrt{\tilde{C}_{min}}} + \|\tilde{\Sigma}_{11}^{-1}\|_\infty \cdot \frac{\lambda_0}{\lambda} \left(n^{-1/2} \max_j |v_j^*| + O_p(\delta) \right) + \|\tilde{\Sigma}_{11}^{-1}\|_\infty \right] \\ &\leq \lambda \left[\frac{4\sigma}{\sqrt{\tilde{C}_{min}}} + \frac{\lambda_0\sqrt{a}}{\lambda\tilde{C}_{min}} \left(n^{-1/2} \max_j |v_j^*| + O_p(\delta) \right) + \frac{\sqrt{a}}{\tilde{C}_{min}} \right],\end{aligned}$$

with probability $1 - 2\exp(-\eta_2\lambda^2 n)$ approach one.

From the bounding expression, we can see that if we have $\lambda + \lambda_0\sqrt{a/n} + O_p(\lambda_0\sqrt{a}\delta) + \sqrt{a}\lambda \to 0$ and with $\lambda^2 n \to \infty$, $\hat{\boldsymbol{\xi}}_1 \to \boldsymbol{\xi}_1^*$ with probability 1.

Furthermore defining

$$\rho(\lambda) = \lambda \left[\frac{4\sigma}{\sqrt{\tilde{C}_{min}}} + \|\tilde{\Sigma}_{11}^{-1}\|_\infty \cdot \frac{\lambda_0}{\lambda} \left(n^{-1/2}\|\mathbf{v}^*\|_\infty + O_p(\delta) \right) + \|\tilde{\Sigma}_{11}^{-1}\|_\infty \right],$$

we conclude, as $\lambda^2 n \to \infty$, if $\rho(\lambda) < \min_{j\in\mathcal{A}} \xi_j^*$, and we have all $\hat{\xi}_j > 0, j \in \mathcal{A}$. Thus we establish the sign consistency $\hat{\mathcal{A}} = \mathcal{A}$.

3.6 Simulation Results

3.6.1 Comparison with Linear LASSO

In many cases, even though the true model is nonlinear, variable selection using linear LASSO can be easily achieved because efficient algorithms for linear LASSO are already available as long as the incoherence conditions (3.22) is satisfied. Hence one question arises about whether or not NGK can perform better than linear LASSO in variable selection. In this section we show that the sparsity of input variables using NGK can be recovered, while this is not possible with linear LASSO due to the failure of conditions (3.22). We

use the same 3-variable setting by Zhao and Yu (2006), where they use this simulation to demonstrate the incoherence condition (3.22) in LASSO. First we generate iid random variables $\mathbf{x}_1$, $\mathbf{x}_2$, $\boldsymbol{\epsilon}$ and $\mathbf{e}$ from $N(0,1)$ with sample size $n = 100$. The third predictor $\mathbf{x}_3$ is generated by

$$\mathbf{x}_3 = a\mathbf{x}_1 + b\mathbf{x}_2 + c\mathbf{e},$$

where $a = 2/3$, $b = 2/3$ and $c = 1/3$, and the response is generated by

$$\mathbf{y} = \beta_1^*\mathbf{x}_1 + \beta_2^*\mathbf{x}_2 + \boldsymbol{\epsilon},$$

where $\beta_1^* = 2$ and $\beta_2^* = 3$. We denote $X_1 = [\mathbf{x}_1, \mathbf{x}_2]$ and $X_0 = [\mathbf{x}_3]$. Zhao and Yu (2006) showed with this setting,

$$\left(\frac{1}{n}X_0^T X_1\right)\left(\frac{1}{n}X_1^T X_1\right)^{-1} = \left(\frac{2}{3}, \frac{2}{3}\right).$$

Thus the incoherence condition (3.22) for linear LASSO is never satisfied with $\text{sgn}(\beta_1^*) = \text{sgn}(\beta_2^*)$. However, the incoherence condition (3.28a)of NGK, provides a different point of view about the incoherence conditions. To demonstrate this, we consider using a linear kernel. Thus with $\boldsymbol{\xi}_1^* = (\xi_1^*, \xi_2^*)^T$ and $\xi_3^* = 0$, we have $K(\boldsymbol{\xi}^*) = \xi_1^*\mathbf{x}_1\mathbf{x}_1^T + \xi_2^*\mathbf{x}_2\mathbf{x}_2^T$. Using the notations in (3.14-3.15), we get

$$\begin{aligned}\tilde{\Sigma}_{01}\tilde{\Sigma}_{11}^{-1}\mathbf{1} &= \begin{bmatrix} a\tilde{\boldsymbol{\alpha}}^T\mathbf{x}_3\mathbf{x}_1^T\tilde{\boldsymbol{\alpha}} & b\tilde{\boldsymbol{\alpha}}^T\mathbf{x}_3\mathbf{x}_2^T\tilde{\boldsymbol{\alpha}} \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{\alpha}}^T\mathbf{x}_1\mathbf{x}_1^T\tilde{\boldsymbol{\alpha}} & 0 \\ 0 & \tilde{\boldsymbol{\alpha}}^T\mathbf{x}_2\mathbf{x}_2^T\tilde{\boldsymbol{\alpha}} \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ &= a\frac{\tilde{\boldsymbol{\alpha}}^T\mathbf{x}_3}{\tilde{\boldsymbol{\alpha}}^T\mathbf{x}_1} + b\frac{\tilde{\boldsymbol{\alpha}}^T\mathbf{x}_3}{\tilde{\boldsymbol{\alpha}}^T\mathbf{x}_2}\end{aligned} \tag{3.49}$$

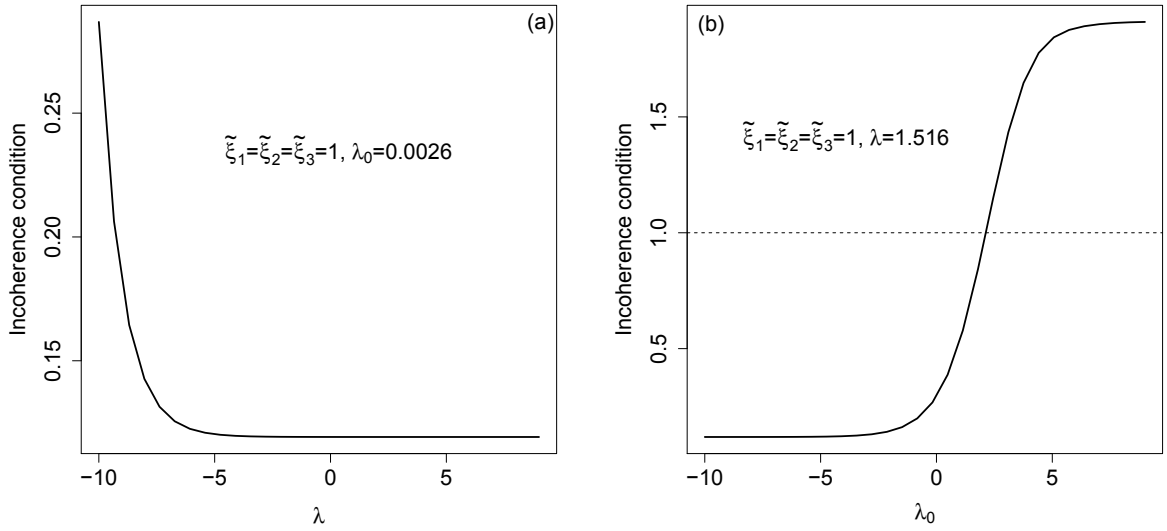

Figure 3.1: Incoherence condition values vs. λ with λ_0 fixed at 0.0026 (a), and vs. λ_0 with λ fixed at 1.516 (b). Both use initial $\tilde{\boldsymbol{\alpha}} = \Delta^{-1}(\tilde{\boldsymbol{\xi}})$ with $\tilde{\boldsymbol{\xi}} = (1,1,1)^T$.

and

$$\begin{aligned}\frac{\lambda_0}{2n\lambda}\tilde{Z}_0^T\tilde{P}\tilde{\boldsymbol{\alpha}} &= \frac{\lambda_0}{2n\lambda}\tilde{Z}_0^T(I-\tilde{Z}_1(\tilde{Z}_1^T\tilde{Z}_1)^{-1}\tilde{Z}_1^T)\tilde{\boldsymbol{\alpha}} \\ &= \frac{\lambda_0}{2n\lambda}\tilde{Z}_0^T\left[I-\left(\frac{1}{n}\mathbf{x}_1\mathbf{x}_1^T+\frac{1}{n}\mathbf{x}_2\mathbf{x}_2^T\right)\right]\tilde{\boldsymbol{\alpha}} \\ &= \frac{\lambda_0}{2n\lambda}\tilde{\boldsymbol{\alpha}}\left(\mathbf{x}_3\mathbf{x}_3^T-a\mathbf{x}_3\mathbf{x}_1^T-b\mathbf{x}_3\mathbf{x}_2^T\right)\tilde{\boldsymbol{\alpha}}.\end{aligned} \tag{3.50}$$

In the above equations (3.49-3.50), we use the fact that for independent random normals, $\frac{1}{n}\mathbf{x}_i^T\mathbf{x}_j = \delta_{ij}, i,j = 1,2$, and $\frac{1}{n}\mathbf{x}_3^T\mathbf{x}_j = a$ or b for $j = 1$ or 2. Given $\tilde{\boldsymbol{\alpha}} = (\lambda_0 I + K(\tilde{\boldsymbol{\xi}}))^{-1}\mathbf{y}$ with $\tilde{\boldsymbol{\xi}} = (1,1,1)^T$, we can calculate the left hand side of (3.28a). For demonstration with one example of simulation, we plot two incoherence condition curves vs λ and λ_0 in Figure 3.1 respectively. For the first curve varying λ, we use the initial $\tilde{\boldsymbol{\alpha}}$ and $\lambda_0 = 0.0026$ estimated by the REML estimation. For the second curve, we fix $\lambda = 1.516$, where we choose the model with the minimum BIC and vary λ_0.

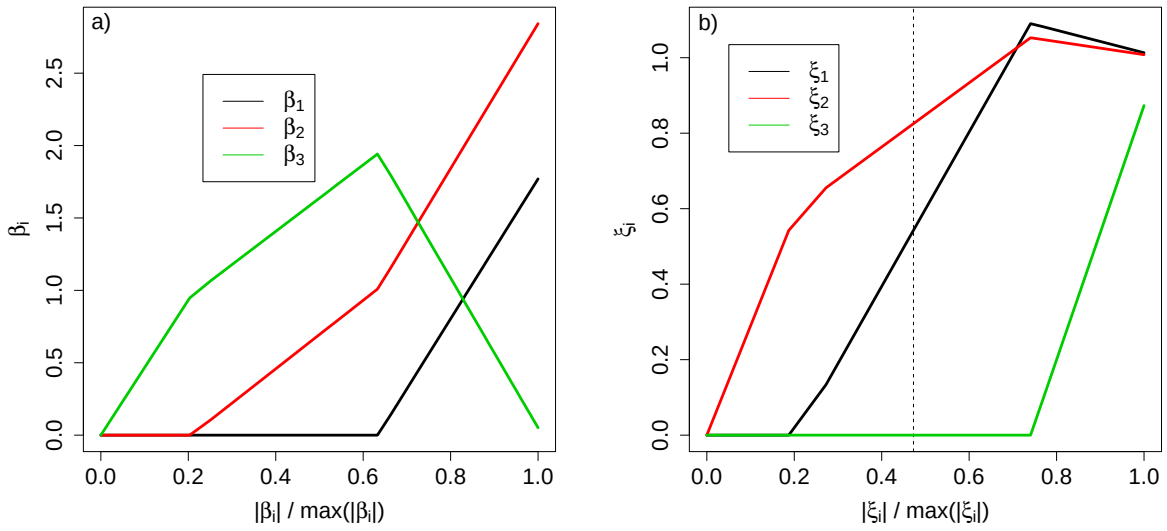

Figure 3.2: Solution paths of β_i's calculated by linear LASSO (a), and ξ_i's calculated by NGK with a linear kernel (b). The solutions paths of NGK are achieved with initial $\tilde{\boldsymbol{\alpha}}$ and $\lambda_0 = 0.0026$ estimated by REML.

Figure 3.1 shows for certain λ and λ_0 values, the incoherence condition values are smaller than one, which satisfies condition (3.28a). This indicates there is a possibility that the variable selection procedure of NGK can recover sparsity of those irrelevant variables.

Figure 3.2 plots the regularization paths calculated by linear LASSO and NGK. We see that for linear LASSO, β_3 is always non-zero on the solution path except when $\lambda = 0$, which means linear LASSO will always choose β_3. However, the regularization paths of NGK show that for some $\lambda > 0$, $\xi_3 = 0$, but both ξ_1 and ξ_2 are greater than zero, which gives the possibility to choose the correct variable set. The dashed line in Figure 3.2 indicates where we achieve the minimum BIC and choose the model.

3.6.2 Simulation Example 1

In this section we test the implementation of NGK on a nonlinear multivariate simulation scenario. We consider a simple situation where the number of predictors p is 11 where the first $a = 5$ predictors being relevant. Three settings with sample sizes $n = 64, 128$ and 256 are compared. For each setting, a total of 200 runs were generated.

Table 3.1: Selection frequency of each predictor in example 1 for 200 runs.

		Predictor Labels										
		1	2	3	4	5	6	7	8	9	10	11
$n = 64$	NGK Gauss	141	137	149	141	143	34	32	31	31	32	21
	NGK Linear	144	152	139	146	147	13	18	27	22	30	14
	COSSO	146	144	151	152	151	29	27	28	26	29	16
$n = 128$	NGK Gauss	132	125	139	132	139	20	17	24	22	16	29
	NGK Linear	135	136	143	149	139	19	10	22	11	13	15
	COSSO	138	140	139	141	141	15	7	16	19	18	16
$n = 256$	NGK Gauss	139	121	132	138	133	11	13	22	25	11	21
	NGK Ploy	141	139	121	132	132	13	11	10	14	10	12
	COSSO	159	146	146	152	151	7	15	5	9	16	9

The nonlinear function $\mathbf{f}$ was generated using a stationary Gaussian process

$$\mathbf{f} \sim N\left(\mathbf{0}, \sigma_\alpha^2 K(\boldsymbol{\xi}^*, X)\right),$$

where $X = [\mathbf{x}_1, ..., \mathbf{x}_p]$, with each column generated by $\mathbf{x}_j \sim U(-2.5, 2.5)$ independently, and $\xi_1^* = ... = \xi_a^* = 2$. The responses then were produced by

$$\mathbf{y} = \mathbf{f} + \boldsymbol{\epsilon}, \tag{3.51}$$

where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 I)$. In this scenario, we chose $\sigma_\alpha^2 = 10$ and $\sigma^2 = 1$ ($\lambda_0 = 1/10$) to pro-

duce our datasets. Note Model (3.51) is equivalent to $\mathbf{y} = K\boldsymbol{\alpha} + \boldsymbol{\epsilon}$ with $\boldsymbol{\alpha} \sim N(\mathbf{0}, \sigma_\alpha^2 K^{-1})$. We also note that in this example, f is not a fixed function anymore, or we can say $\boldsymbol{\alpha}$ is random. This should lead to different incoherence conditions, and because of the randomness of $\boldsymbol{\alpha}$ the probability of recovering the sparsity is expected to be low. However, we only use this example to demonstrate the performance of NGK with other similar variable selection methods, such as COSSO.

Figure 3.3 shows a selected example of the solution paths calculated by NGK with the Gaussian and a linear kernel. The right side of Figure 3.3 also plots the BIC vs. $\log\lambda$ curve and indicates where the model is selected. Table 3.1 reports the frequency of variables selected in the model. Since there is no similar work on variable selection of nonadditive smoothing function, we simply compare the performance of our approach with COSSO.

Five statistics, "False Positive Rate (FP-rate)", "False Negative Rate (FN-rate)", "Model Size (MS)", "Residual Sum of Squares (RSS)" and "Squared Error (SE)", where FP-rate $= \frac{\#False\,Positive}{\#False\,Positive+\#True\,Negative}$, FN-rate $= \frac{\#False\,Negative}{\#False\,Negative+\#True\,Positive}$, RSS $= \sum_i^n (y_i - \hat{f}_i)^2/n$ and SE $= \sum_i^n (f_i - \hat{f}_i)^2/n$, are calculated for each individual run, and the averages and standard deviations of the statistics from the 200 runs are reported in Table 3.2. The estimated functions $\hat{\mathbf{f}}$'s are calculated using the least squared error estimation of a kernel machine with corresponding $\hat{\boldsymbol{\xi}}$, i.e., $\hat{\mathbf{f}} = K(\hat{\boldsymbol{\xi}})\Delta^{-1}(\hat{\boldsymbol{\xi}})\mathbf{y}$. SE can be used to examine the accuracy of the estimation of the nonlinear function f.

The performance of the three methods are similar. COSSO is slightly better in terms of FP rate and FN rate. However, we consider NGK using a linear kernel as the best method in this example not only because the linear NGK has the FP and FN rate close to COSSO but also because it shows the best accuracy of estimating f. In addition, NGK with the Gaussian kernel also has higher accuracy of estimation than COSSO. As expected, we see all methods have a comparablely high FN rate because the function f is not fixed.

Table 3.2: Simulation results of example 1 for 200 runs.

		FP-rate	FN-rate	MS	RSS	SE
$n = 64$	NGK Gauss	0.12(0.11)	0.20(0.15)	4.46(1.77)	1.29(0.71)	0.55(0.63)
	NGK Linear	0.08(0.10)	0.20(0.12)	4.26(1.15)	0.92(0.19)	0.14(0.07)
	COSSO	0.09(0.10)	0.19(0.12)	4.42(1.24)	1.20(0.30)	1.01(0.18)
$n = 128$	NGK Gauss	0.09(0.09)	0.22(0.14)	3.98(1.56)	1.13(0.36)	0.26(0.33)
	NGK Linear	0.06(0.08)	0.21(0.12)	3.96(1.07)	0.96(0.12)	0.07(0.04)
	COSSO	0.05(0.08)	0.21(0.12)	3.87(1.10)	1.04(0.15)	1.00(0.12)
$n = 256$	NGK Gauss	0.07(0.09)	0.22(0.15)	3.83(1.65)	1.10(0.31)	0.16(0.29)
	NGK Linear	0.05(0.08)	0.24(0.10)	3.68(0.98)	0.96(0.08)	0.04(0.02)
	COSSO	0.04(0.07)	0.18(0.13)	4.03(1.07)	1.00(0.10)	1.00(0.09)

3.6.3 Simulation Example 2

In this example, we consider f is fixed and the response y is generated by the following function

$$y = f + \epsilon = 10\cos(x_1) + 3x_2^2 + 5\sin(x_3) + 6\exp(x_4/3)x_4 + 8\cos(x_5) + x_5x_2x_1 + \epsilon, \quad (3.52)$$

where $\epsilon \sim N(0,1)$, and $x_j \sim U(0,1), j = 1, ..., p$. This f is similar to the one used in Liu et al. (2007). In this example, we consider the total number of predictor $p = 10$ where the first $a = 5$ predictors are relevant. Again, three settings with sample size $n = 64$, 128, and 256 and total 200 runs for each setting were generated.

Figure 3.4 shows a selected example of the solution paths for NGK with the Gaussian and the linear kernel and the BIC curves. Table 3.3 and Table 3.4 report the frequency of being selected and five statistics of 200 runs. It can be seen that all three methods have the same zero FN rate. When sample size $n = 64$, the COSSO approach seems to perform better in terms of FP rate, but still has the worst estimation accuracy. When the sample size increases, NGK methods catch up quickly in terms of FP rate. When $n = 256$, the

Table 3.3: Selection frequency of each predictor in example 2 for 200 runs.

		Predictor Labels									
		1	2	3	4	5	6	7	8	9	10
$n = 64$	NGK Gaussian	200	200	200	200	200	27	24	26	17	24
	NGK Linear	200	200	200	200	200	10	19	11	20	8
	COSSO	200	200	200	200	200	13	14	14	12	10
$n = 128$	NGK Gaussian	200	200	200	200	200	3	3	3	7	3
	NGK Linear	200	200	200	200	200	9	3	10	13	10
	COSSO	200	200	200	200	200	2	0	3	5	1
$n = 256$	NGK Gaussian	200	200	200	200	200	1	1	2	2	1
	NGK Linear	200	200	200	200	200	0	0	2	1	3
	COSSO	200	200	200	200	200	3	1	0	2	3

three methods have nearly the same FP rate. It can be seen that as sample size increases, COSSO keeps almost the same estimation accuracy, while the estimations of NGK methods seem become more accurate as the sample size gets larger. In this example, NGK with the Gaussian kernel is considered as the best method because it not only performs as well as others in terms of the FN and FP rate but also has the best estimation accuracy.

According to example 1 and example 2, we observed that although COSSO method is based on an additive model (or with at most a second order interactions), it is also capable to select the input variables in models with higher order interactions. This capability should be available for all additive variable selection methods including SpAMs, VANISH and nonnegative garrote function component selection. Those additive type variable selection methods considering up to the second order interactions are suitable when we want to select the function and interaction components. However, they become too complicated and unnecessary when we only want to select relevant predictors,because these models have to consider all interaction components and when p is large modeling each high order interaction becomes impracticable. When higher order interactions are in the

true model, additive type methods may not perform as well as the kernel based methods in terms of estimation accuracy, such as in example 1, the interaction should be any order since we use a Gaussian process to generate the data. In example 2, the true model contains a third order interaction, in both cases the COSSO method has the worst accuracy. On the other hand, instead of modeling each interaction component, NGK with the Gaussian kernel can model any order interaction as well as select the correct variable set.

Table 3.4: Simulation results of example 2 for 200 runs.

		FP-rate	FN-rate	MS	RSS	SE
$n = 64$	NGK Gaussian	0.09(0.11)	0.00(0.00)	5.59(0.83)	1.02(0.24)	0.34(0.09)
	NGK Linear	0.05(0.09)	0.00(0.00)	5.34(0.56)	1.14(0.20)	0.35(0.08)
	COSSO	0.04(0.08)	0.00(0.00)	5.32(0.61)	0.84(0.20)	0.99(0.18)
$n = 128$	NGK Gaussian	0.02(0.05)	0.00(0.00)	5.01(0.31)	1.15(0.17)	0.27(0.05)
	NGK Linear	0.04(0.07)	0.00(0.00)	5.23(0.46)	1.20(0.15)	0.31(0.05)
	COSSO	0.01(0.04)	0.00(0.00)	5.06(0.27)	0.95(0.14)	1.02(0.13)
$n = 256$	NGK Gaussian	0.01(0.03)	0.00(0.00)	5.04(0.18)	1.12(0.12)	0.20(0.05)
	NGK Linear	0.01(0.03)	0.00(0.00)	5.03(0.17)	1.22(0.11)	0.29(0.03)
	COSSO	0.01(0.03)	0.00(0.00)	5.05(0.21)	0.98(0.09)	1.01(0.09)

3.6.4 Simulation Example 3

As mentioned before, when $p > n$, COSSO and LARS on linear nonnegative garrote both fail. In this example, we consider a special case with $p = 80$ and $n = 64$. To our best knowledge there are no other approaches capable at fitting a nonadditive model and select the predictors for $p > n$ cases, thus we only compare Gaussian kernel NGK and linear kernel NGK using our backfitting algorithm. As example 2, example 3 has the same true function, (3.52). Again, the first five predictors are relevant and a total of 400 runs has

been simulated. Because the procedure is computationally intensive for large p, we only demonstrate the results with $n = 64$.

Table 3.5: Simulation results of example 3 for 400 runs.

		FP-rate	FN-rate	MS	RSS	SE
$n = 64$	NGK Gaussian	0.08(0.04)	0.003(0.022)	11.88(4.21)	1.56(0.30)	1.01(0.20)
	NGK Linear	0.08(0.11)	0.030(0.110)	12.52(14.5)	1.27(1.53)	0.87(1.37)

Figure 3.5 is the selected example solution paths for example 3 calculated by NGK Gaussian and linear kernel. In both cases the variable size selected by BIC is larger than 5. Other criteria will also select a larger model size. Therefore in example 3, variable selection according to a single run is not sufficient to reveal the correct model.

Because of the number of predictors, instead of a table, we report the selection frequency or probability of each variable based on the 400 runs in Figure 3.6. There is little difference between the two NGK kernels in terms of the selection probability. This can be seen from Figure 3.6 where the first five variables have selection probability very close to 1.0 for both kernels. The linear kernel has a slightly higher FN rate than Gaussian kernel since for the Gaussian kernel those five variables have slightly higher selection probability. However, both kernels show the same behavior that the first five variables are clearly separated from the other 75 variables in terms of selecting probability.

From Table 3.5 we can see the FP rate and FN rate for example 3. Compared with example 2, the FP rate of example 3 for Gaussian kernel is comparable the same, which are 0.09 and 0.08. For a linear kernel, the FP rate increases slightly. The major change is that in example 2 the FN-rates are zero for both kernels, but in example 3 the FN-rates are not zero anymore. This is reasonable since including many irrelevant predictors deteriorates the performance of variable selection.

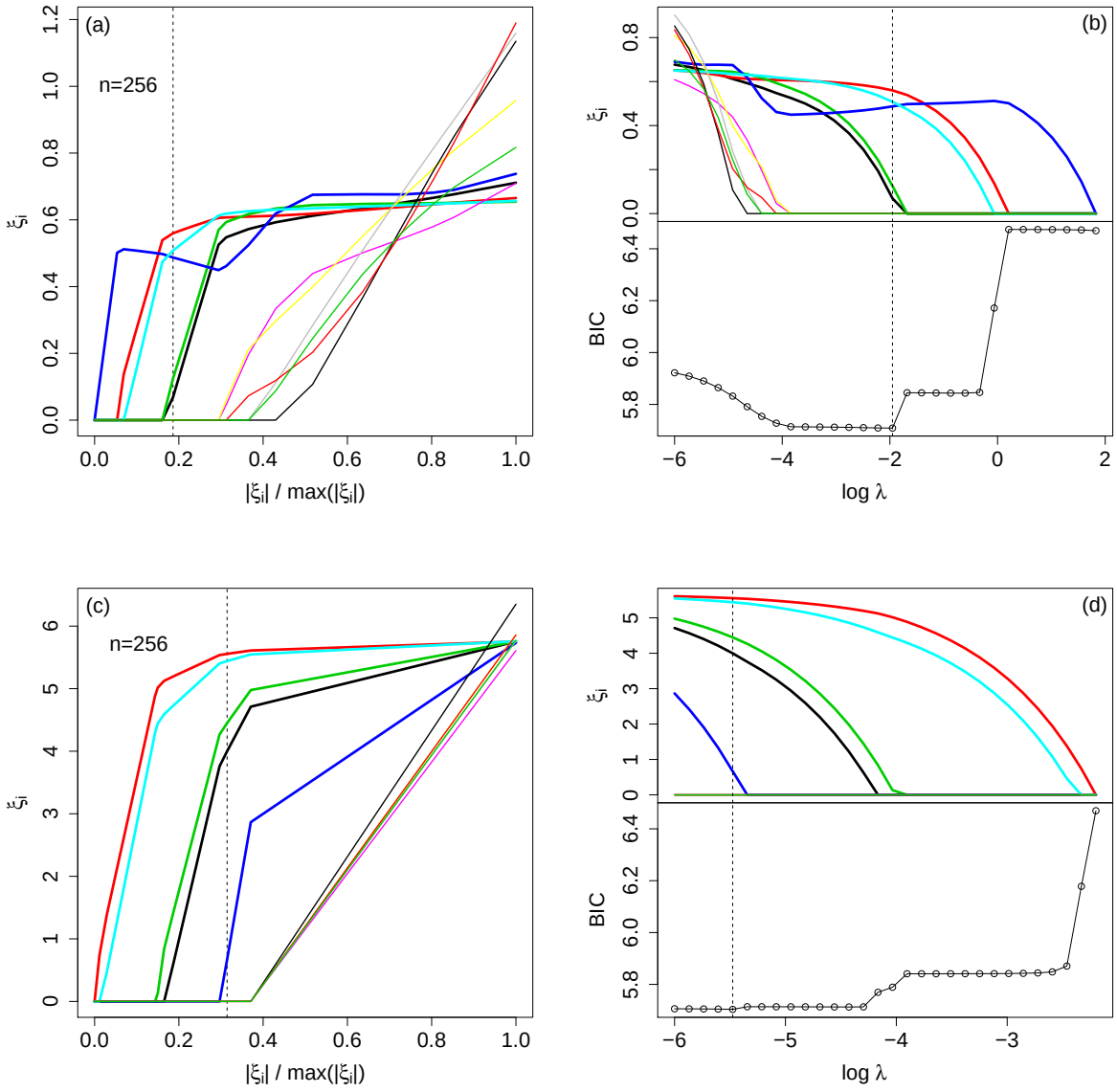

Figure 3.3: Selected example of NGK solution path for example 1 using the Gaussian kernel, (a) and (b), and the linear kernel, (c) and (d). Left side: ξ_j's vs. L_1 norm of ξ_j's, Right side: ξ_j's and BIC vs. $\log \lambda$.

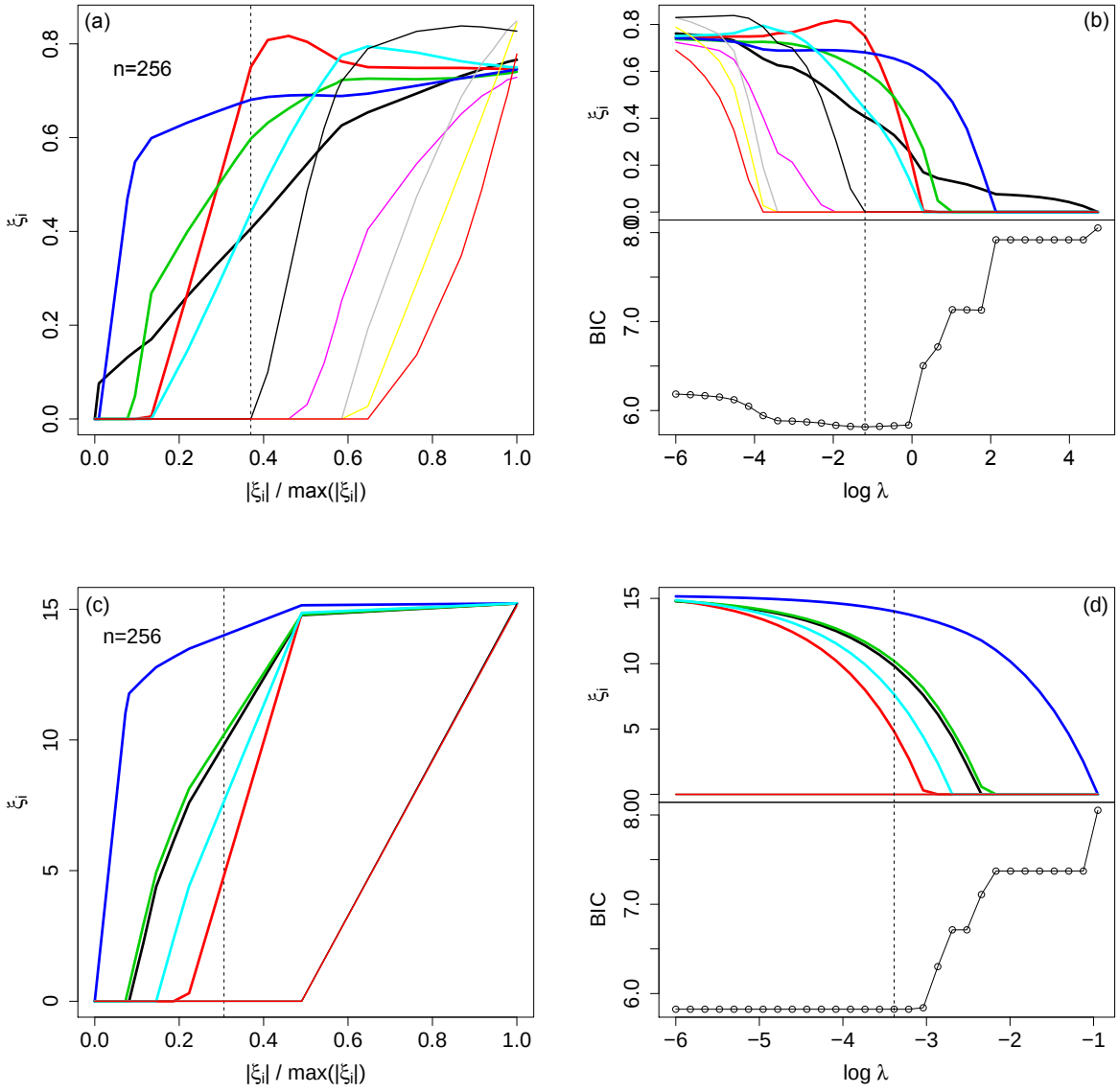

Figure 3.4: Selected example of NGK solution path for example 2 using the Gaussian kernel, (a) and (b), and the linear kernel, (c) and (d). Left side: ξ_j's vs. L_1 norm of ξ_j's, Right side: ξ_j's and BIC vs. $\log \lambda$.

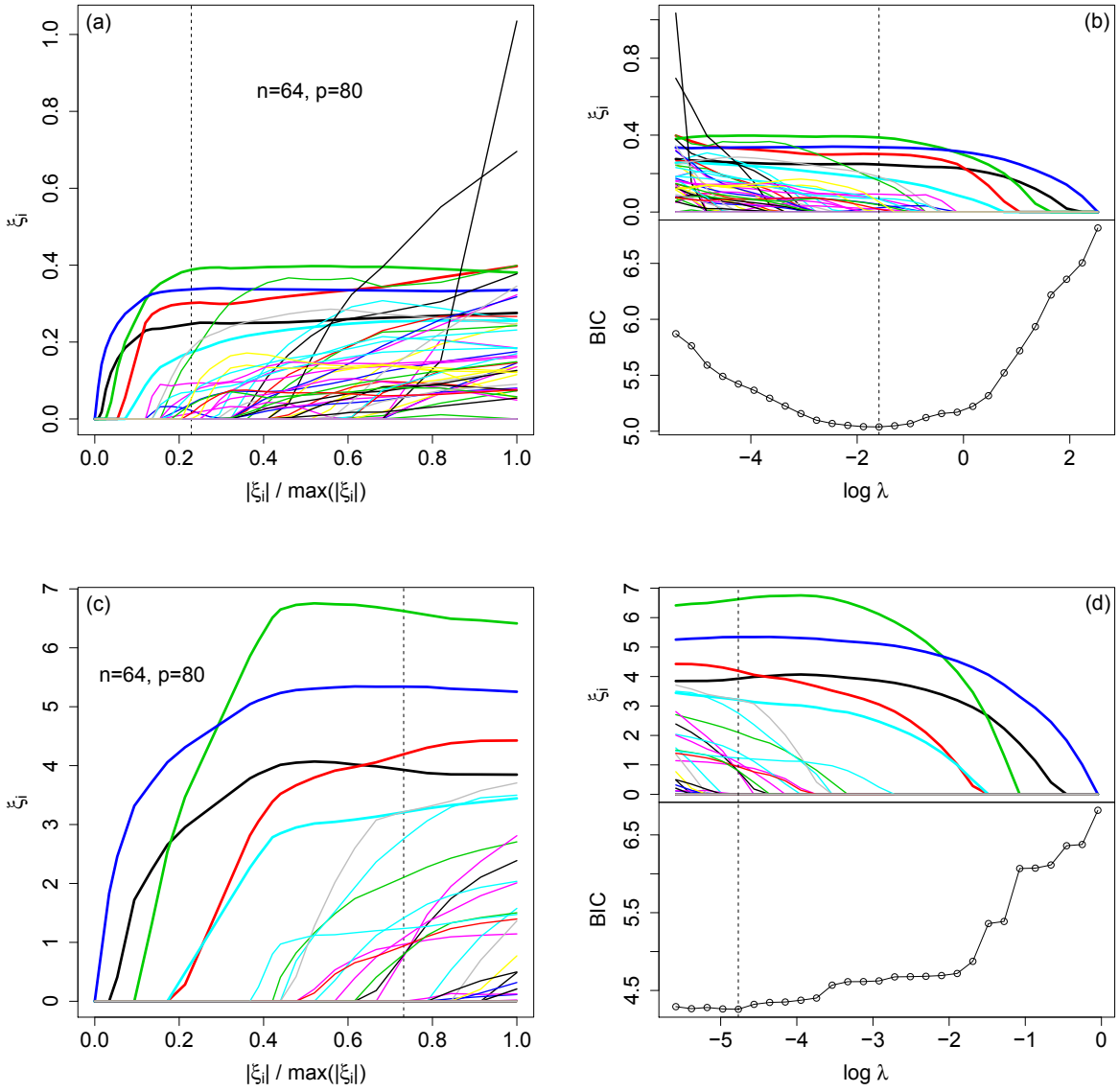

Figure 3.5: Selected example of NGK solution path for example 3 using the Gaussian kernel, (a) and (b), and the linear kernel, (c) and (d). Left side: ξ_j's vs. L_1 norm of ξ_j's, Right side: ξ_j's and BIC vs. $\log \lambda$.

One additional observation from Table 3.5 is that the standard deviation of five statistics of 400 runs is larger for a linear kernel while the average is close for both kernels. This may be because we determine the selection according to the minimum BIC. While the BIC

curve with the Gaussian kernel has a clear minimum, the BIC curve with the linear kernel drops from $\xi_j = 0$ and becomes flat. We choose the model at the turning point of the BIC curve which may introduce some variability for different runs. Another observation from Table 3.5 is the large average model size.

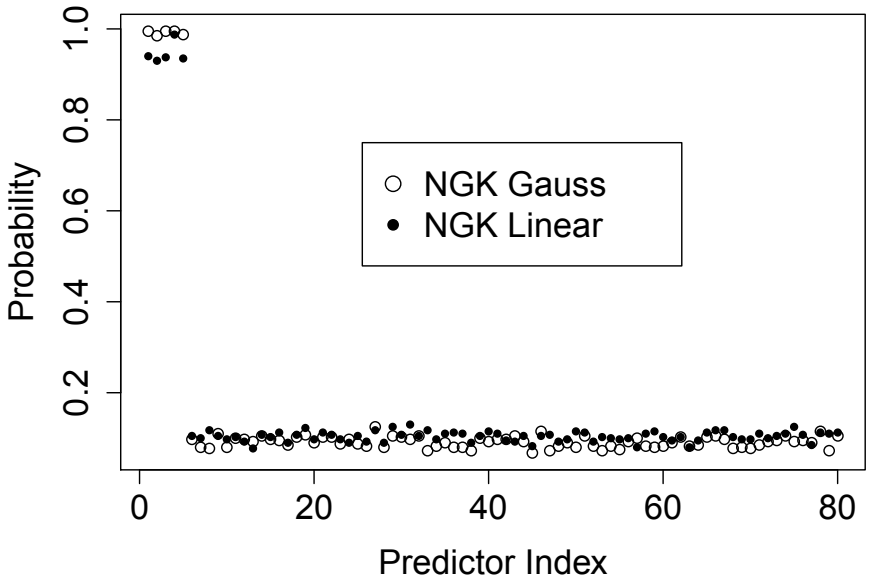

Figure 3.6: Selection probability of each predictor in example 3 for 400 runs.

The above simulation results show that if we choose the model according to BIC or any other criteria based on single run, especially when p is very large, we may include more irrelevant predictors. However, if we choose the model based on the selection frequency or probability, such as Figure 3.6, it is clearer that the five true variables behave differently from the rest. This provides a new perspective about the variable selection that we can rank the selection probability for each predictor and have more power to choose the correct variable set. This can be done using some multiple drawing or resampling approaches (Chatterjee and Lahiri, 2011; Hall et al., 2009; Knight and Fu, 2000). In the following section we apply this idea to two real datasets.

3.7 Applications

In this section, we describe two applications of our method in a practical setting.

3.7.1 Key Selection in Cryptography Data

Our first example is taken from a research of cryptology. Side-channel analysis (SCA) is a technique of the cryptanalysis with which an attacker estimates the secret key based on information gained from the physical implementation of a cryptographic algorithm.

Figure 3.7(a) gives the diagram of attacking on an Advanced Encryption Standard (AES) system. The large box represents an electronic circuit system that implements the AES algorithm. The AES algorithm processes the input data, "in_k", and produces the encrypted output, "out_k", using a secret key with byte $\theta_k, k = 0, ..., 15$. The 16 S boxes each takes an 8-bit key byte, θ_k. The attacker's objective is to determine the value of the secret key byte θ_k's. The output of the AES algorithm is captured in each of the 16 encryption rounds, and the corresponding power consumption of all rounds is recorded as $\mathbf{y}$. By observing the output, one can therefore infer 16 estimates $X_k = [\mathbf{x}_{1,k}, \mathbf{x}_{2,k}, \cdots, \mathbf{x}_{256,k}], k = 0, ..., 15$, corresponding to 16 secret key bytes, and there are $2^8 = 256$ possibilities for each estimate, only one is true for the corresponding key byte. The SCA proceeds by observing n encryptions. The data structure can be expressed in a matrix-form as shown in Figure 3.7(b), where $\mathbf{y}$ is a $n \times 1$ vector, and each X_k is a $n \times 256$ matrix. Note that each $\mathbf{x}_{k,j}, j = 1, ..., 256$, is a function of the output out_k and the jth key guess $\hat{\theta}_{k,j}$ on the kth S box. The SCA problem is to find the set of columns that represents the true θ_k's. The index of the selected column in each X_k returns the value of secret key byte θ_k.

Define $X = [X_0, \ldots, X_{15}]$ to be an $n \times (r \times k)$ matrix, reflecting the internal estimates

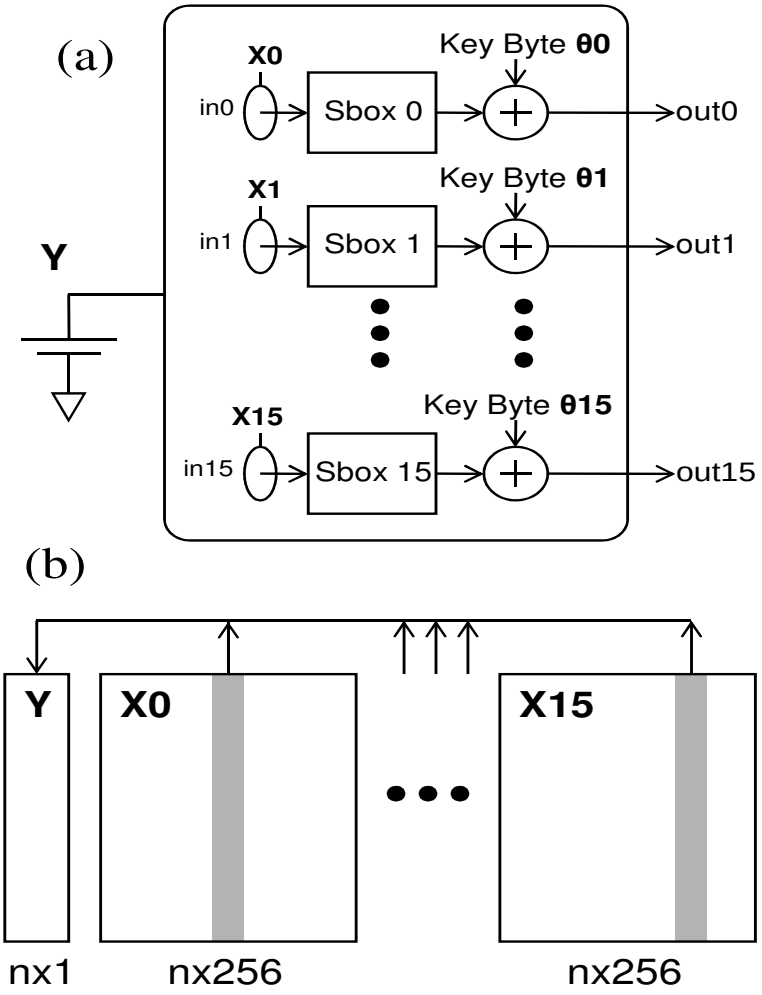

Figure 3.7: (a) Diagram of side-channel attack; (b) Data structure.

for an SCA with n measurements, k key bytes, and r guesses per key byte. In the SCA example, $n = 5120$, $r = 256$ and $k = 16$. The objective is to identify what possible key guesses (or what combination of columns of X) are highly associated with the power consumption trace $\mathbf{y}$. Since there are k key bytes and r possible key guesses for each key byte, there are a total $\binom{(r\times k)}{k}$ possible ways to select k variables. The power consumption trace $\mathbf{y}$ can be expressed in terms of the key estimates X by Model (3.1). It is reasonable to assume that there are no interactions among different keys and key guesses, so that we can use a linear kernel on this data.

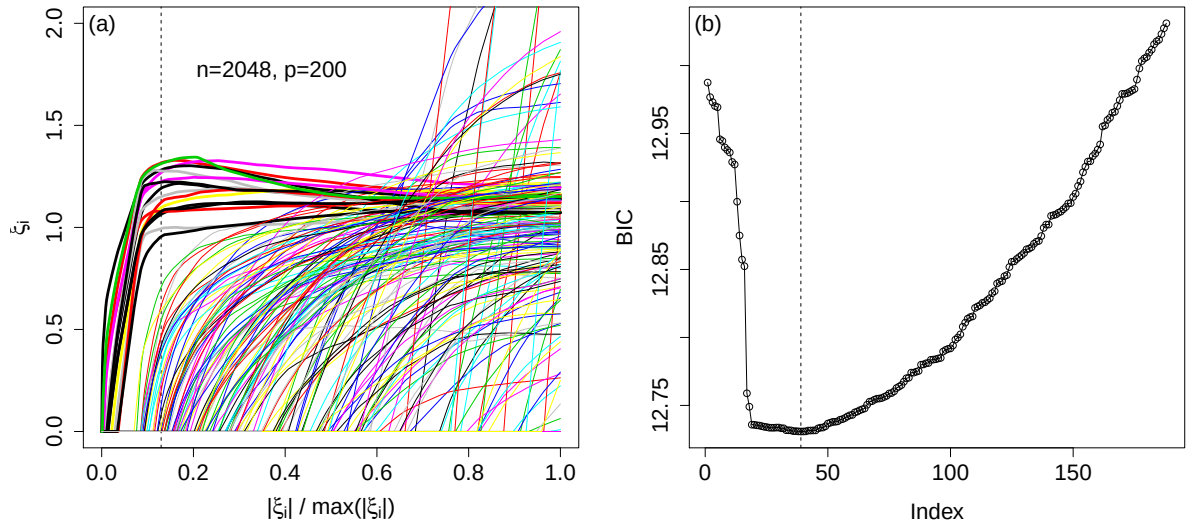

Figure 3.8: NGK solution path for SCA data using linear kernel. (a), ξ_j's vs. L_1 norm of ξ_j's; (b), ξ_j's and BIC vs. $\log \lambda$.

The data set $(\mathbf{y}, X)$ contains 5120 observations and a total of $p = 16 \times 256 = 4096$ predictors. Identifying the 16 key bytes is clearly a problem of variable selection. Due to the high dimensionality of X space, directly application of our NGK approach is not realistic. Fan and Lv (2008) discuss the sure independence screening (SIS) for ultrahigh dimensional feature space, and Fan et al. (2011) extend the correlation learning in a linear models to nonparametric independence screening (NIS) in additive models. They argue that, under certain conditions, the probability that the screened model includes all the relevant predictors approaches one as n increases. Following SIS, a variable selection procedure like LASSO can be applied as NIS-LASSO. We adopt the similar procedure NIS-NGK, i.e., following a NIS procedure we apply the NGK variable selection method.

Another issue about this data set is the observation size. With $n = 5120$, the computing becomes expensive due to the calculation of the kernel K. We take a resampling approach with observation size $m = 2048$ to reduce the computing burden. However, it turns out

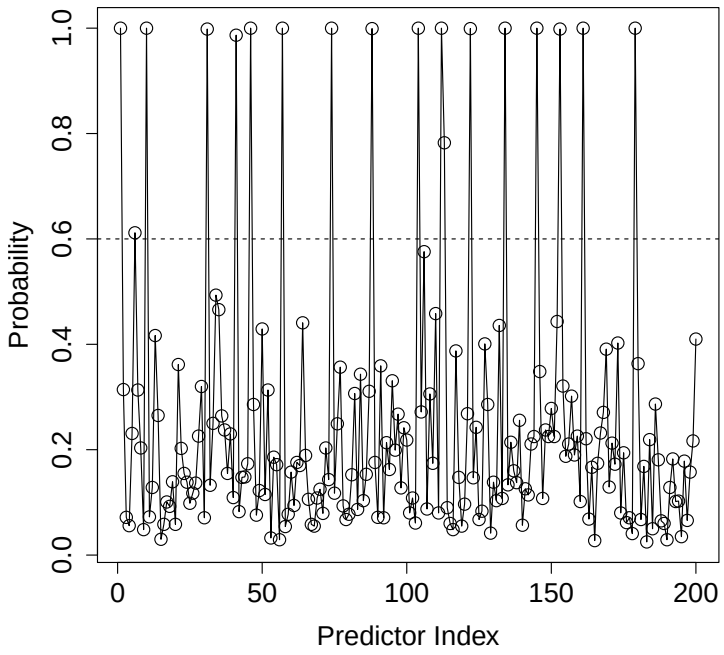

Figure 3.9: Selection probability of each key guess of SCA data using m-out-of-n resampling procedure, $m = 2048, n = 5120$ and total 1200 runs.

that variable selection by resampling a large dateset is more powerful than variable selection by a single run over the entire data set. We have already discussed the advantage of resampling in Section 3.6.3. There is little work discussing the resampling/bootstrap procedure in variable selection by LASSO or nonnegative garrote. Hall et al. (2009) proposed an m-out-of-n bootstrapping method on linear LASSO, which may provide some theoretical basis for our resampling procedure with NGK.

The NIS screening approach we applied is to rank the predictors according to the descent order of the residual sum of squares of the componentwise nonparametric regression (Fan et al., 2011),

$$\mathcal{S} = \{1 \leq j \leq p : r_j \leq C\} \tag{3.53}$$

where $r_j = \min_{\xi_j \boldsymbol{\alpha}} \|\mathbf{y} - \xi_j D^j \boldsymbol{\alpha}\|^2$, C is a predefined threshold value depending on n. To reduce the computing time, we take $\boldsymbol{\alpha} = \mathbf{y}$ and all $\xi_j = 1$. Then the NIS screening is equivalent to SIS which ranks the predictors by the correlation $\mathbf{x}_j \mathbf{y}^T$. This can be seen by plugging $\boldsymbol{\alpha} = \mathbf{y}$ and $\xi_j = 1$ into r_j and $r_j = \mathbf{y}^2 - (\mathbf{x}_j \mathbf{y}^T)^2$. Using this approach we first screen down the predictor size to 200. According to Theorem 1 in Fan et al. (2011), with $n = 5120$ very large and a finite predictor size, the probability of the 200 predictors to include the true 16 predictors is close to one.

Following the NIS step, we apply the NGK variable selection to one resampled data set from the 5120 observations with resampling size $m = 2048$. We note that our backfitting algorithm has the same results as LARS since the linear kernel K is additive. Figure 3.8 is the solution path and BIC curve of an example of the m-out-of-n resampling procedure.

The solution path in Figure 3.8(a) shows that there are 16 predictors (bold lines) behaving differently from the others. Because we have information about the AES key, we know there are 16 bytes corresponding to the 16 true predictors. By checking the AES key, those 16 predictors are exactly the 16 key bytes. In addition, the BIC curve shows those 16 predictors are clearly separated from the rest of the curve (Figure 3.8(b)). Other criteria, like Cp, GCV, all show the same behavior. Obviously those 16 predictors should be selected as the true model. However, if we use the minimum BIC as the criterion to choose the model, we will have a total of 38 predictors selected in this run. Although these 38 predictors include the 16 true predictors, it is too many. Even when we sample more observations or use all 5120 observations, the minimum or turning point of BIC criterion always selects more than 16 irrelevant predictors.

Thus we further resample the dataset up to 1200 runs with replacement. Each run chooses the variables according to the BIC criterion, and counts the frequency of selected variables. As we observed that the BIC minimum usually selects around 50 variables,

in order to reduce the computing, we use selecting windows such that we choose the first 50 predictors for each resampling run. The selection probability/frequency the 200 predictors are plotted in Figure 3.9. Using 60% as the selection threshold, we will choose 18 predictors from Figure 3.9. If we use 80% as the threshold, we can exactly choose the true 16 key bytes. By resampling the dataset from the large dataset, we are able to simulate the probability of being chosen, and this process of variable selection is more powerful than simply relying on one fixed data set using the usual criterion.

3.7.2 Gene Selection in a Pathway Data

We applied our NGK with the Gaussian kernel to a set of diabetes data from Mootha et al. (2003). They utilized the HGC-133a Affymetrix genechip with 22,283 genes to study 17 normal glucose tolerance individuals vs. 18 Type II diabetes mellitus patients. The 22,283 genes make up a total of 251 pathways. The top significant pathways related to the diabetes disease have been identified (Pang et al., 2006; Pang and Zhao, 2008). It is known that genes in a pathway are not independent of each other. They interact with each other without a known structure. Among those pathways, pathway 133, 4 and 140 are three interesting ones with 58, 18 and 22 genes, respectively. In each pathway we label the genes by their appearance index, gene 1, gene 2 ... and so on. However, since the 18 genes in pathway 4 are all included in pathway 140, we use the gene index of pathway 140 to label genes in pathway 4 except for genes 4, 5, 19 and 20 do not appear. Thus in this application, the data set structure is $(\mathbf{y}, X)$ with $n = 35$ observations, and $p = 58, 18$ and 22 predictors respectively. The response is the outcome of glucose levels.

Figure 3.10(a), 3.11(a) and 3.12(a) plot the solution paths of ξ_j's corresponding to the genes of the three pathways, and Figure 3.10(b), 3.11(b) and 3.12(b) show the BIC curves to

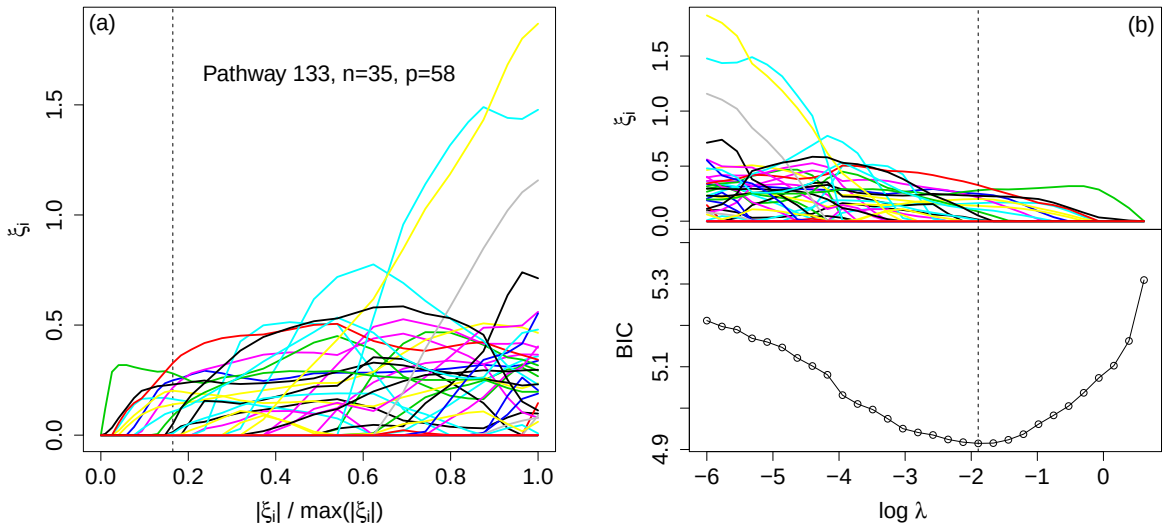

Figure 3.10: NGK solution path for diabetes data pathway 133 using Gaussian kernel. (a), ξ_j's vs. L_1 norm of ξ_j's; (b), ξ_j's and BIC vs. $\log \lambda$.

select the genes where 13, 7 and 9 genes are selected respectively. The index sets for the selected genes of three pathways by Gauss kernel NGK are $\hat{\mathcal{A}}_{133} = \{1, 4, 5, 14, 19, 23, 29, 31, 34, 41, 51, 53, 57\}$, $\hat{\mathcal{A}}_4 = \{8, 10, 11, 12, 13, 14, 21\}$ and $\hat{\mathcal{A}}_{140} = \{5, 8, 10, 11, 12, 13, 14, 18, 21\}$. However, as we observed in simulation example 1 (Section 3.6.4) and the SCA experiment (Section 3.7.1), variable selection depending on a single drawing of data may not be powerful even if the number of observations is large. In the diabetes data, there are only 35 observations. So we need further steps to increase the selection power. In this section we propose following a residual permutation procedure to repeat the variable selection process and thus we can count the total frequency/probability of each predictor.

- *Step 1*: Apply the NGK variable selection with the Gaussian kernel on the original data set using the backfitting algorithm introduced in Section 3.3 and obtain the selected variables $\hat{\boldsymbol{\xi}} = (\hat{\xi}_j)^T_{j \in \hat{\mathcal{A}}}$. Use $\hat{\boldsymbol{\xi}}$ to fit the Gaussian kernel machine again to achieve the new $\hat{\boldsymbol{\alpha}}$ and new λ_0 by REML method, such that, $\hat{\mathbf{y}} = K(\hat{\boldsymbol{\xi}})\hat{\boldsymbol{\alpha}}$. Obtain the

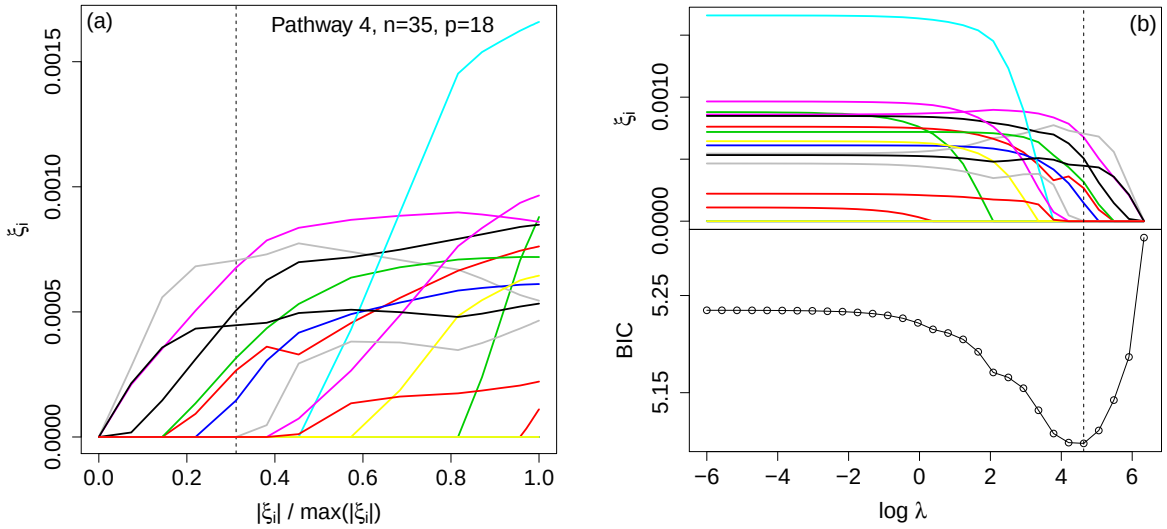

Figure 3.11: NGK solution path for diabetes data pathway 4 using Gaussian kernel. (a), ξ_j's vs. L_1 norm of ξ_j's; (b), ξ_j's and BIC vs. $\log \lambda$.

residual $\hat{\boldsymbol{\epsilon}} = \mathbf{y} - \hat{\mathbf{y}}$. Center $\hat{\boldsymbol{\epsilon}}$ by subtracting their mean.

- *Step 2*: Permute the residual $\hat{\boldsymbol{\epsilon}}$ to get new $\hat{\boldsymbol{\epsilon}}^*$ and simulate outcomes as $\mathbf{y}^* = K(\hat{\boldsymbol{\xi}})\hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\epsilon}}^*$.

- *Step 3*: Based on the new data set $(\mathbf{y}^*, X)$ with fixed initial $\hat{\boldsymbol{\alpha}}$ and fixed λ_0, apply the NGK variable selection again and obtain the selected gene set.

- *Step 4*: Repeat Steps 2-3 many times (e.g. 3000 times).

- *Step 5*: Obtain the empirical probability/frequency of selecting each variable.

The results of permutating NGK steps are summarized in Figure 3.13. Because pathway 4 is part of pathway 140, we plot the results of two pathways in one plot Figure 3.13(b). If 60% is taken as as the threshold, the sets of genes selected are $\mathcal{A}^*_{133} = \{4, 5, 14, 19, 23, 31, 34, 41, 53\}$, $\mathcal{A}^*_4 = \{8, 10, 11, 12, 21\}$ and $\mathcal{A}^*_{140} = \{5, 12, 21\}$ respectively.

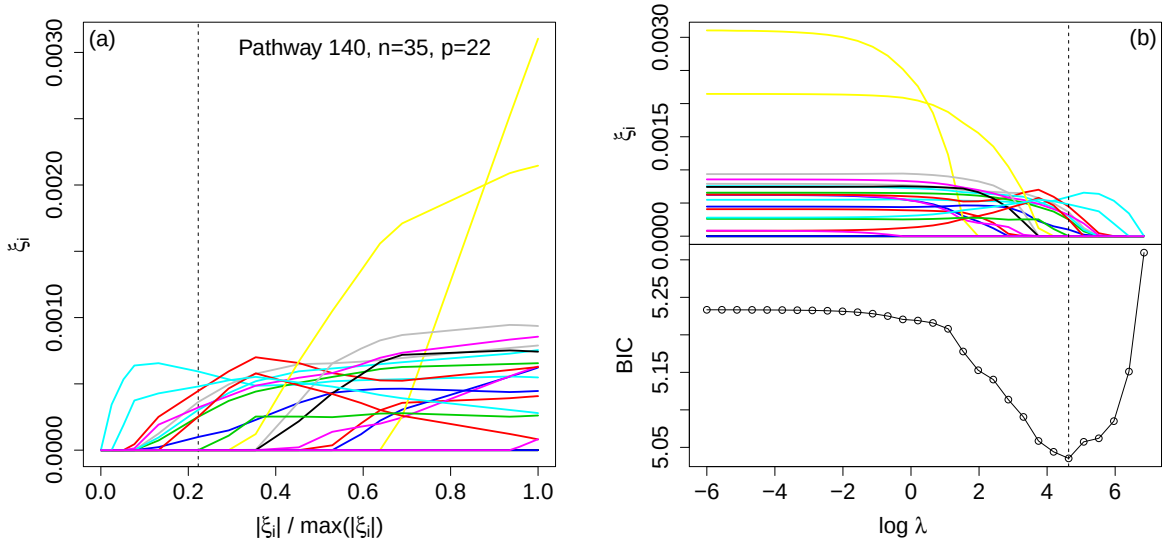

Figure 3.12: NGK solution path for diabetes data pathway 140 using Gaussian kernel. (a), ξ_j's vs. L_1 norm of ξ_j's; (b), ξ_j's and BIC vs. $\log \lambda$.

Comparing these with $\hat{\mathcal{A}}$, we see that $\mathcal{A}^* \subset \hat{\mathcal{A}}$. For example, for pathway 133 four extra genes were selected using the single NGK step, $\{1, 29, 51, 57\}$. Especially for gene 1, by using the permutation approach, the probability of selecting gene 1 is less than 20%, but in one run NGK it is selected. Again there is little theory work on variable selection with residuals permutation. Some references can be found at Knight and Fu (2000) and Chatterjee and Lahiri (2011) for the residuals bootstrapping LASSO estimator.

Interesting observations for pathway 4 and pathway 140 are found in Figure 3.13(b). First, some of the genes are not significantly related to the response, such as gene $\{1, 2, 3, 7, 9, 15, 22\}$. In both pathways, their selection probabilities remain small. Some genes are significantly related to the response and have a high selection probability in both pathways, such as gene 21. Second, some genes seem to interact with each other. For example, genes $\{10, 11, 12, 13, 14\}$ appear to group a gene segment that has similar selection probabilities in both pathways. An interesting gene is gene 5 which does not appear in pathway

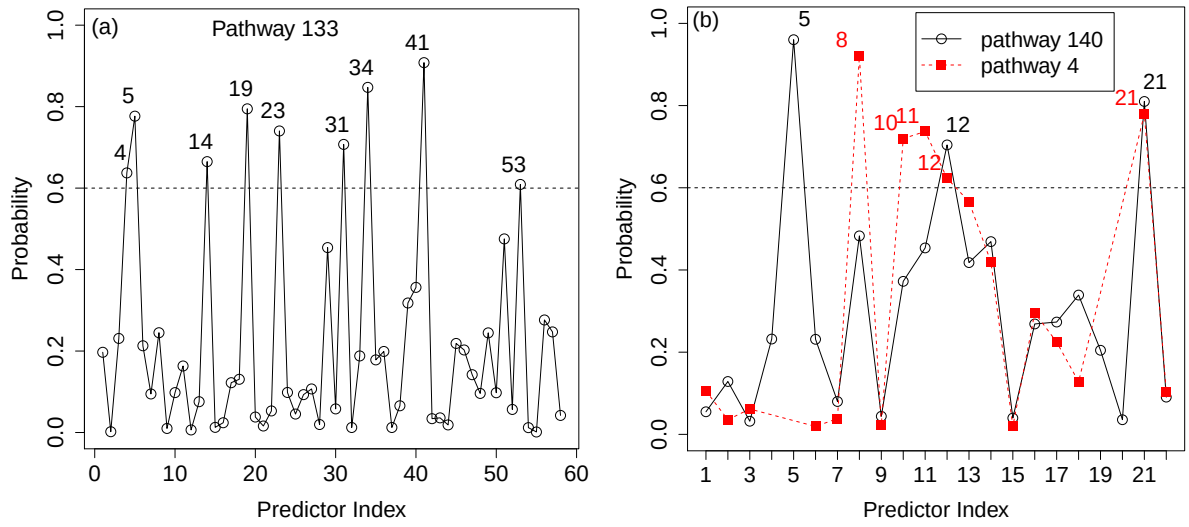

Figure 3.13: Selection probability of each gene using residual permutation method for pathway 133 (a), and pathway 140 and 4 (b), total 3000 runs for each pathway.

4. But gene 5 has the highest selection probability in pathway 140. While gene 5 appears in pathway 140, the selection probability of $\{8, 10, 11, 13\}$ go down compared with pathway 4. This indicates some interaction may happen between gene 5 and gene 8 and the gene segment$\{10, 11, 12, 13, 14\}$.

3.8 Discussion

In this dissertation we have addressed a nonnegative garrote variable selection procedure connecting with kernel machines, so that we can recover sparsity of the multivariate input variable in a nonadditive smoothing function. The method we proposed is attractive because it is applicable not only to nonadditive smoothing function, but also to additive models by choosing different kernels. We also have developed an efficient coordinate descent updating procedure for the scale parameters ξ_j's which inherits the nice properties

of the regular backfitting method and can replace the LARS algorithm in models with an additive multiple kernel.

The results reported in this dissertation show that similar theoretical properties of linear LASSO can be extended to our NGK models. There are additional interesting theoretical properties. Theorem 3.4.3 makes strong approximation assumption on the kernel matrix. The convergence rate of recovering sparsity should take into consideration the possible effects of this assumption. For example, it may result in a slow convergence rate. Another issues related to the theoretical properties could be the convergence properties of the coordinate descent algorithm we proposed.

Further interesting theoretical works include studying the performance of the model selection criterion like BIC in the least squares error kernel machine models. In this dissertation, we did not directly use the criterion to choose the final selected variables, instead we proposed resampling procedures in two cases : when n is large and n is small. Thus the consistency and convergence properties of resampling/bootstrap on NGK can be a interesting topic in the future.

Possible extensions of our method include applying the NGK to generalized linear models (GLM). Logistic kernel machine regressions and multiple categorical classifications are very popular in many applications. Selecting input variables with NGK applied to GLMs is also a challenging work because the link function of GLM is nonlinear too.

Another interesting extension of our method can be considering kernels with more complicated structures. To illustrate this, we can consider a data set with q multivariate variables such as q genetic pathways, each one containing multidimensional genetic expressions and some pathways may share genes. Thus the kernel in this model can be $K = \rho_1 K_1(\boldsymbol{\xi}_1, X_1) + \rho_2 K_2(\boldsymbol{\xi}_2, X_2) + ..., \rho_q K_q(\boldsymbol{\xi}_q, X_q)$. By applying penalties on $\boldsymbol{\rho} = (\rho_1, ..., \rho_q)^T$

and on $\boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_q$, NGK may be able to recover sparsity of X_j's as well as the additive functional components of $f_j = K_j\boldsymbol{\alpha}$'s. This extension might be considered as a group NGK. This extension is interesting because by applying the group NGK we may be able to select interesting pathways and interactions from the pathway pool.

Chapter 4

Semiparametric Mixed Model for Evaluating Pathway Environment Interaction

4.1 Introduction

Gene-related diseases are complex processes associated not only with specific gene or gene sets but also with gene-gene and gene-environment interaction. For decades, statistical methods have focused on analyzing microarray data based on single genes or single-nucleotide polymorphisms (SNPs) analysis (Chatterjee et al., 2006; Hahn et al., 2003; Maity et al., 2009; Moore et al., 2010; Ritchie et al., 2001). However, single-gene based methods have many limitations. For instance, the effect of one gene on a disease is difficult to interpret and current methods are unable to model gene dependencies so that they may not detect genes with moderate changes that give more insight into biological

processes but pick up single gene with dramatic changes (Mootha et al., 2003). For these reasons, gene-set or pathway-based approaches have attracted increasing attention in recent years (Goeman et al., 2004, 2005; Liu et al., 2007; Wang, et al., 2007; Pang et al., 2006, 2011; Kim et al., 2011). It is recognized that a joint study of the association between the outcome and a group of genes within the same pathway could complement genes/SNPs analysis for providing insight in understanding complex diseases (Wang, et al., 2007).

A genetic pathway is the interactions of genes that depend on each other's individual functions and act accordingly to create the aggregate function related to a cellular process (Goeman et al., 2004). There are several special characteristics of pathways, such as various dimensionality (a pathway can contain several genes or over a thousand ones), and interaction network (genes within the a pathway are expected to function together and hence interact with each other). Thus traditional statistical analyses face difficulties in handling these situations. For instance, linear parametric models usually either fail due to the "curse of dimensionality", or end up with computational explosion in the number of possible interactions among genes within a pathway. To deal with these difficulties, many innovative statistical methods have merged in recent years. Goeman et al. (2004) proposed a global test derived from a random effects model to determine the significance of the global expression pattern of a group of genes. A random forests approach was proposed by Pang et al. (2006). Liu et al. (2007) proposed a semiparametric model for covariate and genetic pathway effects on continuous outcomes, where the covariate effects and the pathway effect are modeled parametrically and nonparametrically, respectively. They established the connection between the least squares kernel machine (LSKM) and linear mixed models, which simplifies specification of a nonparametric model with multidimensional data. Pang et al. (2011) considered more complicated situations with two or more pathway effects presented in the linear mixed model, which allows the researcher

to study how multiple pathways relate to the phenotype of interest. A semiparametric Bayesian approach has also been proposed for evaluating pathway effects on clinical outcomes Kim et al. (2011). However, despite the success of analyzing pathways instead of a single gene, all existing methods ignore the environment exposure covariates, and still fewer focus on the interaction between environmental variables and the genetic pathways.

It has been recognized that genetic factors alone cannot account for many cases of gene related disease (Adami, et al., 2008; Chakravarti and Little, 2003). The gene-environment (G-E) or pathway-environment (P-E) interactions are critical in understanding the dynamic process of disease since ignoring them may mask the detection of a genetic effect and may lead to inconsistent association results (Manolio et al., 2006). Furthermore, understanding the G-E interactions can be important for risk prediction and evaluating the benefit of changes in modifiable environmental exposures or environmental regulations. For these reasons, the number of studies utilizing gene-environment interactions has increased dramatically. These range from semiparametric linear or logistic regression models with linear combinations of genes/SNPs as the predictor (Chatterjee et al., 2006; Maity et al., 2009; Park and Hastie, 2008) to the multifactor dimensionality reduction (MDR) as a data mining technique for identifying genetic and environmental effects associated with either dichotomous or continuous phenotypes (Ritchie et al., 2001; Hahn et al., 2003; Moore et al., 2010). Unfortunately, these studies are all genes/SNPs based methods, and they possess problems in dealing with the pathway analysis. For example, representing the pathway effects with linear combinations of genes has limitations in detecting non-linear patterns of interacting genes. Furthermore, the number of genes in a pathway can be in the hundreds or thousands, which makes modeling the gene-gene or gene-environment interaction very consuming.

To capture high order interactions within the high dimensional genes regressor space as well as the G-E interactions, Zou et al. (2010) employed a nonparametric regression model with a Gaussian process. With their model the gene and environmental variables are modeled non-parametrically, and all of the possible interactions effects are considered simultaneously. However, using one Gaussian process to describe both gene and environmental variable function spaces results in all the interaction effect being indistinguishable. Thus it is almost impossible to apply a suitable test for interesting effects such as G-E interaction.

In this dissertation, we propose a semiparametric mixed effects model to include environmental variables, genetic pathway effect, and their interaction. By extending Liu et al. (2007)'s linear mixed model to our model, we evaluate the interaction between an environmental variable and pathway as well as allow nonlinear relationships between the environmental variable and a continuous outcome. Assuming that both the pathway and interaction effects have multivariate normal distributions with a zero mean and covariance structure with specific kernels, we model them within the framework of Gaussian processes. Thus in our model both pathway and interaction effects are indeed modeled as random effects. Instead of modeling the environmental variable as a parametric fixed effect, we model it non-parametrically via natural cubic spline. By modeling environmental variables and pathways in this way, we can construct the kernel for the P-E interaction based on the analysis-of-variance-like (ANOVA-like) decompositions of functions (Wahba, 1990; Gu and Wahba, 1993) for a multivariate function. The feature of our method is to model the interaction between environmental and pathway covariates separately from the interactions among genes within the pathway, which are automatically modeled by the Gaussian process for pathway effect. Our model also extends the additive and interaction smoothing splines for univariate functions to multivariate functions with arbitrary

kernel.

In a mixed model, the smoothing parameters of the spline and the Gaussian kernels can be considered as the variance components of the random effects, and thus are simultaneously estimated by maximizing the restricted maximum likelihood (REML). By additively modeling the multivariate functions, this model is suitable for analyzing genetic pathway data in which the P-E interaction attracts particular interests. Furthermore, the covariance structure of our model makes the test of the "overall" pathway effect or P-E interaction effect possible. By "overall" we mean either the main effect of a pathway, the interaction effect associated with the pathway, or both. The restricted likelihood ratio test (RLRT) of two zero variance components under non-standard conditions is employed to test the overall pathway effect, while the RLRT of one zero-variance component and score test are applied to test the P-E interaction.

We first define our model in Section 4.2, and discuss two REML methods to estimate the model parameters in Section 4.3. Then in Section 4.4, we introduce PLRT statistics for testing two or one zero-variance components and the score test for testing the P-E interaction effect. In Section 4.5, we present a set of simulation studies concerning nonparametric function estimates and variance component tests for various settings. In Section 4.6 we apply our method to the genetic pathway data for Type II diabetes. Finally, in the last section, we conclude our work and discuss potential extensions of our model.

4.2 Construction of Semiparametric Linear Mixed Effects Models

4.2.1 Model Description and the Kernel of the Interaction Function Space

Let us consider that we have a total of n subjects and the ith subject has a continuous disease-related outcome $y_i, i = 1, 2, ...n$. We are interested in relating this response $\mathbf{y} = (y_1, y_2, ..., y_n)^T$ to one particular pathway gene expression data $Z = (\boldsymbol{z}_1, \boldsymbol{z}_2, ..., \boldsymbol{z}_n)^T$ and k environmental variables. In a general form, we can write this nonlinear relationship as

$$\mathbf{y} = \mathbf{f} + \boldsymbol{\epsilon}, \tag{4.1}$$

where $\boldsymbol{\epsilon}$ and $\mathbf{f}$ are $n \times 1$ dimensional vectors with a specific relationship with $\mathbf{y}$ for the ith entry as $y_i = f(\boldsymbol{x}_i^T, \boldsymbol{z}_i^T) + \epsilon_i$, in which $\boldsymbol{x}_i^T = (x_{i1}, x_{i2}, ..., x_{ik})$ is $1 \times k$ vector of environmental variables and $\boldsymbol{z}_i^T = (z_{i1}, z_{i2}, ..., z_{ip})$ is the $1 \times p$ vector of gene expression within a pathway and p is the gene number. In this dissertation, we only consider the case with one environmental variable, i.e., $k = 1$ so that the input $\mathbf{x}^T$ is reduced to univariate x. We assume that the errors $\boldsymbol{\epsilon} \sim N(0, \sigma^2 I)$ are $n \times 1$ iid random variables vector. $f(\cdot)$ denotes the unknown non-linear smooth functions for x_i, $\boldsymbol{z}_i^T$, and their interaction. In this dissertation, we assume function f has the following form:

$$f(x, \boldsymbol{z}^T) = \beta_0 + f_x(x) + f_z(\boldsymbol{z}^T) + f_{xz}(x, \boldsymbol{z}^T), \tag{4.2}$$

where β_0 is the intercept term, and $f_\alpha, \alpha \in \{x, z, xz\}$, represents the nonlinear effect of the environmental variable, the pathway or the interaction respectively. The above equa-

tion is similar to the additive model with two univariate variables and their interaction, except $\boldsymbol{z}^T$ is a multivariate variable. By writing the general model (4.1) in this way, we can estimate f_x, f_z and their interaction f_{xz} separately according to the characteristics of the pathway and the environmental variable. We model $f_x(x)$ using the nonparametric function such as a cubic smoothing spline (Wahba, 1990; Lin and Zhang, 1999; Zhang and Lin, 2003). To handle the high dimensional pathway covariates, $\boldsymbol{z}^T$, we may consider a Gaussian process to express $f_z(\boldsymbol{z}^T)$ since the least squares kernel machine method with the Gaussian kernel has achieved success in a genetic pathway data analysis (Liu et al., 2007).

Before we derive the specific representation for the interaction function, we need examine the function space of f_x and f_z respectively. For the smoothing spline $x \in \mathcal{T} = [0, 1]$, f_x is spanned on the function space $\mathcal{H}_x = \mathcal{H}_x^0 \bigoplus \mathcal{H}_x^1$, where "$\bigoplus$", $\mathcal{H}_x^0$ and $\mathcal{H}_x^1$ represent the direct sum operator of two subspaces, the null function space and the penalized function space respectively (Wahba, 1990). Assuming n distinct observed values of x_i such that $0 < x_1^0 < \cdots < x_n^0 < 1$, the mth order smoothing spline estimator $f_x(x)$ can be expressed as (Wahba, 1990; Zhang and Lin, 2003),

$$f_x(x) = \sum_{j=1}^{m} b_j \phi_j(x) + \sum_{i=1}^{n} c_i k_x(x, x_i^0),$$

where $\phi_j(x)$ is the polynomial basis that span the null space $\mathcal{H}_x^0$ with $\phi_j(x) = x^{j-1}/(j-1)!, j = 1, 2, \ldots, m$, and $k_x(x, x_i^0) = [(m-1)!]^{-2} \int (x-u)_+^{m-1} (x_i^0 - u)_+^{m-1} du$ is the kernel which uniquely determines the space $\mathcal{H}_x^1$. For $m = 2$, the natural cubic spline that we shall apply in our model, the kernel of $\mathcal{H}_x^1$ can be calculated as (Hastie et al., 2009; Rasmussen and

Williams, 2006)

$$k_x(x, x') = \int_0^1 (x-u)_+(x'-u)_+ du = \frac{\min(x,x')^3}{3} + \frac{\min(x,x')^2|x-x'|}{2}, \tag{4.3}$$

where subscript "+" indicates the positive part of the expression. For the null space $\mathcal{H}_x^0$, the kernel is calculated as $k_x^0(x, x') = \sum_{j=1}^2 \phi_j(x)\phi_j(x') = 1 + xx'$.

With the orthonormal polynomial basis, $\mathcal{H}_x^0 = \{1\} \bigoplus \{x\}$, where $\{1\}$ and $\{x\}$ stand for the linear function spaces spanned by the constant 1 and the linear basis x which is centered (Guo, 2002). Since the kernel of the function space of the direct sum of two subspaces is expressed by the direct sum of the kernel of the subspaces (Aronszajn, 1950; Wahba, 1990), we can derive the kernel of the function space without the the constant term for the cubic smoothing spline, $\{x\} \bigoplus \mathcal{H}_x^1$, as $[xx' + k_x(x, x')]$.

For the function space of f_z, we consider a similar argument by MacKay (1998) that starting from a parametric model, we can span the function of f_z by a radial basis

$$f_z(\boldsymbol{z}^T) = \sum_{h=1}^{H} c_h \phi_h(\boldsymbol{z}^T), \tag{4.4}$$

where $\phi_h(\boldsymbol{z}^T) = \exp\left[\frac{-\|\boldsymbol{z}-\boldsymbol{z}_h\|^2}{2\rho}\right]$ is the radial basis functions centered at fixed points $\{\boldsymbol{z}_h\}_{h=1}^H$. Assuming $\mathbf{c} = (c_1, ..., c_h)^T \sim N(0, \tau_z I)$, the entry of the covariance matrix of $\mathbf{f}_z$ is expressed as

$$R = \tau_z \sum_{h=1}^{H} \phi_h(\boldsymbol{z})\phi_h(\boldsymbol{z}').$$

Taking as an example a one-dimensional case, MacKay (1998) shows that in the above expression the sum over h becomes an integral when taking the limit $H \to \infty$ such that $R = \tau_z \exp[-\|z - z'\|^2/\rho]$. Generalizing from this particular case, we can define the Gaus-

sian kernel of the function space $\mathcal{H}_z^1$ on $\boldsymbol{z}$

$$k_z\left(\boldsymbol{z}^T, \boldsymbol{z}'^T\right) = \exp\left(-\|\boldsymbol{z} - \boldsymbol{z}'\|^2/\rho\right), \tag{4.5}$$

and we assume that f_z is generated from a zero mean Gaussian process with the kernel matrix produced by k_z.

Since the tensor product of the kernels of two function spaces determines a new function space (Aronszajn, 1950), we use the tensor product of the kernels of $\{x\} \bigoplus \mathcal{H}_x^1$ and $\mathcal{H}_z^1$ to construct a new function space, $\mathcal{H}_{xz}^1$, which contains any order interaction f_{xz} between x and $\boldsymbol{z}^T$. Now we can express the kernel of the interaction function space as

$$k_{xz}\left(x, \boldsymbol{z}^T; x', \boldsymbol{z}'^T\right) = [xx' + k_x(x, x')] \cdot k_z\left(\boldsymbol{z}^T, \boldsymbol{z}'^T\right). \tag{4.6}$$

Therefore, we are able to represent the nonparametric interaction function using a zero mean Gaussian process with the kernel matrix produced by this kernel function.

In the rest of this dissertation, we use K_x, K_z and K_{xz} to stand for the Gram or kernel matrices produced by k_x, k_z and k_{xz} respectively. In a specific problem, the environmental variable x must be scaled into $\mathcal{T} = [0, 1]$ to construct the interaction kernel. Notice the model Expression (4.2) is not the analysis of variance (ANOVA) decomposition of the smoothing function f since $\mathcal{H}_z^1$ and $\mathcal{H}_{xz}^1$ are not orthogonal to each other. This may cause the identifiability problem between f_z and f_{xz}. However, in practice, this problem only happens to our model in extreme situations such as when the entries of matrix $xx' + k_x(x, x')$ are close to each other. In general, f_z and f_{xz} can be identified well as shown in the simulation and application study.

4.2.2 Linear Mixed Model Representation

Now we are prepared to pose the optimization problem. Based on the above argument, the corresponding function spaces that are penalized are $\mathcal{H}_x^1$, $\mathcal{H}_z^1$ and $\mathcal{H}_{xz}^1$. Analogous to the additive models (Hastie and Tibshirani, 1990), the estimation problem for Model (4.1) becomes: for a given set of predictors $(x_i, \mathbf{z}_i^T), i = 1, 2, ..., n$, find f to maximize

$$-\frac{1}{2}(\mathbf{y}-\mathbf{f})^T(\mathbf{y}-\mathbf{f}) - \frac{1}{2}\sum_{\alpha}\lambda_\alpha \|f_\alpha\|^2_{\mathcal{H}_\alpha^1}, \tag{4.7}$$

where $\|f_\alpha\|_{\mathcal{H}_\alpha^1}$'s are the norms induced by K_α of $\mathcal{H}_\alpha^1, \alpha \in \{x, z, xz\}$, and λ_α's are the penalty parameters that balance the tradeoff between goodness-of-fit and smoothing of the curve or high dimensional surface. The solutions to Expression (4.7) are called the least square kernel machine estimation, and Liu et al. (2007) showed the equivalence of the least square kernel machine to the linear mixed model without interaction effects.

Model (4.2) can be represented in terms of a mixed model as follows. According to the Representer Theorem (Kimeldorf and Wahba, 1971), the nonparametric function can be expressed by the kernel, $f_z(\cdot) = \sum_{i=1}^n a_i k_z(\cdot, \mathbf{z}_i^T)$ and $f_{xz}(\cdot) = \sum_{i=1}^n b_i k_{xz}(\cdot; x_i, \boldsymbol{z}_i^T)$. So the vectors of these functions are

$$\begin{aligned}\mathbf{f}_z &= K_z\mathbf{a},\\ \mathbf{f}_{xz} &= K_{xz}\mathbf{b},\end{aligned}$$

where $a_i \in \mathbb{R}, b_i \in \mathbb{R}$. Based on the properties of reproducing kernels, the squared norms

of $\mathcal{H}_z^1$ and $\mathcal{H}_{xz}^1$ can be expressed as

$$\begin{aligned} \|f_z\|^2_{\mathcal{H}_z^1} &= \mathbf{a}^T K_z \mathbf{a} = \mathbf{f}_z^T K_z^{-1} \mathbf{f}_z, \\ \|f_{xz}\|^2_{\mathcal{H}_{xz}^1} &= \mathbf{b}^T K_{xz} \mathbf{b} = \mathbf{f}_{xz}^T K_{xz}^{-1} \mathbf{f}_{xz}. \end{aligned}$$

To represent the remaining part of Model (4.2), $\beta_0 + f_x(\cdot)$, we follow Lin and Zhang (1999); Zhang et al. (1998); Green (1987); Green and Silverman (1994)'s procedure. The vector of f_x, $\mathbf{f}_x$ (note here the constant β_0 is absorbed into f_x), can be expressed in terms of $\boldsymbol{\beta} = (\beta_0, \beta_1)^T$ and $(n-2) \times 1$ random vector $\mathbf{r}_x$ as

$$\mathbf{f}_x = X\boldsymbol{\beta} + B\mathbf{r}_x \tag{4.8}$$

for n distinct input x values, where $\mathbf{r}_x \sim N(0, \tau_x I)$ and X is the design matrix of the null space $\mathcal{H}_x^0$ spanned by the orthogonal polynomial basis, i.e., $X = (\mathbf{1}, \mathbf{x})$ and $\mathbf{x}$ is the $n \times 1$ vector of centered x. B is a matrix defined as $B = L(L^T L)^{-1}$, where L is $n \times (n-2)$ full rank matrix with $M = LL^T$. M is a penalty matrix defined by Green and Silverman (1994) such that the squared norm of $\mathcal{H}_x^1$,

$$\|f_x\|^2_{\mathcal{H}_x^1} = \int_0^1 \left[f_x''(t)\right]^2 dt = \mathbf{f}_x^T M \mathbf{f}_x = \mathbf{r}_x^T \mathbf{r}_x.$$

More details to define B and M can be found in Green and Silverman (1994), Zhang et al. (1998) and Appendix B.

Plugging those representations of square norms and $\mathbf{f}_\alpha$'s back into (4.7), we have

$$-\frac{1}{2}(\mathbf{y}-\mathbf{f})^T(\mathbf{y}-\mathbf{f}) - \frac{1}{2}\left(\lambda_x \mathbf{r}_x^T \mathbf{r}_x + \lambda_z \mathbf{f}_z^T K_z^{-1} \mathbf{f}_z + \lambda_{xz} \mathbf{f}_{xz}^T K_{xz}^{-1} \mathbf{f}_{xz}\right).$$

If we define $\lambda_x = \sigma^2/\tau_x, \lambda_z = \sigma^2/\tau_z$ and $\lambda_{xz} = \sigma^2/\tau_{xz}$, and have random vectors $\mathbf{r}_z = \mathbf{f}_z, \mathbf{r}_z \sim N(0, \tau_z K_z)$ and $\mathbf{r}_{xz} = \mathbf{f}_{xz}, \mathbf{r}_{xz} \sim N(0, \tau_{xz} K_{xz})$, then the above equation is equivalent to

$$-\frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{f})^T(\mathbf{y}-\mathbf{f}) - \frac{1}{2\tau_x}\mathbf{r}_x^T\mathbf{r}_x - \frac{1}{2\tau_z}\mathbf{r}_z^T K_z^{-1}\mathbf{r}_z - \frac{1}{2\tau_{xz}}\mathbf{r}_{xz}^T K_{xz}^{-1}\mathbf{r}_{xz}, \tag{4.9}$$

which is the triple penalized log likelihood function of the linear mixed model

$$\mathbf{y} = \mathbf{f} + \boldsymbol{\epsilon} = X\boldsymbol{\beta} + B\mathbf{r}_x + \mathbf{r}_z + \mathbf{r}_{xz} + \boldsymbol{\epsilon}. \tag{4.10}$$

From the Bayesian point-of-view, $\mathbf{f}$ is interpreted as the sum of four zero-mean stationary Gaussian processes, each with a prior covariance function $\tau_\alpha K_\alpha$ ($\boldsymbol{\beta}$ can be viewed with infinite variance). The vectors $\mathbf{r}_z$ and $\mathbf{r}_{xz}$ have more specific meanings as the pathway main effect and the P-E interaction effect. Although $\mathbf{r}_x$ does not have such a meaning, it can be interpreted as the nonlinear contribution of the relationship of the response and the environmental variable.

Differentiating Expression (4.10) with respect to $\boldsymbol{\beta}$ and $\mathbf{r}_\alpha$'s, it is easy to show that the best linear unbiased prediction (BLUP) estimate of the random effects, given σ^2 and τ_α's as fixed, is obtained from solving

$$\begin{bmatrix} X^TX & X^TB & X^T & X^T \\ B^TX & B^TB+\lambda_x I & B^T & B^T \\ X & B & I+\lambda_z[K_z]^{-1} & I \\ X & B & I & I+\lambda_{xz}[K_{xz}]^{-1} \end{bmatrix} \times \begin{bmatrix} \boldsymbol{\beta} \\ \mathbf{r}_x \\ \mathbf{r}_z \\ \mathbf{r}_{xz} \end{bmatrix} = \begin{bmatrix} X^T\mathbf{y} \\ B^T\mathbf{y} \\ \mathbf{y} \\ \mathbf{y} \end{bmatrix}. \tag{4.11}$$

Equation (4.11) shows that the BLUP estimate of $\boldsymbol{\beta}$ and $\mathbf{r}_\alpha$'s are unique if X^TX is full rank which is usually satisfied.

4.2.3 Estimate Pathway and Interaction Effects

Given the fixed parameters σ^2 and τ_α's, the covariance of $\mathbf{y}$ is obtained as follows using Model (4.10),

$$\Sigma = \text{Cov}(\mathbf{y}) = \sigma^2 I + \tau_x BB^T + \tau_z K_z + \tau_{xz} K_{xz}. \tag{4.12}$$

Instead of solving Expression (4.11) directly, we perform recursive steps to simultaneously achieve the approximate expressions of $\boldsymbol{\beta}$ and $\mathbf{r}_\alpha$'s, $\alpha \in \{x, z, xz\}$,

$$\begin{aligned}
\hat{\boldsymbol{\beta}} &= \left(X^T\Sigma^{-1}X\right)^{-1} X^T\Sigma^{-1}\mathbf{y}, \\
\hat{\mathbf{r}}_x &= \left(B^T\Delta_1^{-1}B + \tau_x^{-1}I\right)^{-1} B^T\Delta_1^{-1}(\mathbf{y} - X\hat{\boldsymbol{\beta}}), \\
\hat{\mathbf{r}}_z &= \left(\Delta_2^{-1} + \tau_z^{-1}K_z^{-1}\right)^{-1} \Delta_2^{-1}(\mathbf{y} - X\hat{\boldsymbol{\beta}} - B\hat{\mathbf{r}}_z), \\
\hat{\mathbf{r}}_{xz} &= \left(\Delta_3^{-1} + \tau_{xz}^{-1}K_{xz}^{-1}\right)^{-1} \Delta_3^{-1}(\mathbf{y} - X\hat{\boldsymbol{\beta}} - B\hat{\mathbf{r}}_x - \hat{\mathbf{r}}_z),
\end{aligned} \tag{4.13}$$

where I is the $(n-2)\times(n-2)$ identity matrix, and $\Delta_j,\ j = 1, 2, 3$, are covariances for the following distributions,

$$\begin{aligned}
&\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon}_0, && \boldsymbol{\epsilon}_0 \sim N(0, \Delta_0 = \Sigma), \\
&\mathbf{y} - X\hat{\boldsymbol{\beta}} = B\mathbf{r}_x + \boldsymbol{\epsilon}_1, && \boldsymbol{\epsilon}_1 \sim N(0, \Delta_1 = \sigma^2 I + \tau_z K_z + \tau_{xz} K_{xz}), \\
&\mathbf{y} - X\hat{\boldsymbol{\beta}} - B\hat{\mathbf{r}}_x = \mathbf{r}_z + \boldsymbol{\epsilon}_2, && \boldsymbol{\epsilon}_2 \sim N(0, \Delta_2 = \sigma^2 I + \tau_{xz} K_{xz}), \\
&\mathbf{y} - X\hat{\boldsymbol{\beta}} - B\hat{\mathbf{r}}_x - \hat{\mathbf{r}}_z = \mathbf{r}_{xz} + \boldsymbol{\epsilon}, && \boldsymbol{\epsilon} \sim N(0, \Delta_3 = \sigma^2 I).
\end{aligned} \tag{4.14}$$

The above expressions for $\hat{\boldsymbol{\beta}}$ and $\hat{\mathbf{r}}_\alpha$'s are all linear transformations of $\mathbf{y}$; thus, their covariances are easily determined using identity $\text{Cov}(A\mathbf{y}) = A\text{Cov}(\mathbf{y})A^T = A\Sigma A^T$, where A is the transformation matrix in expressions (4.13).

4.3 REML Estimation of the Variance Components

4.3.1 REML Approach for Estimating Variance Components

In the previous section, when solving Equation (4.11) we assume that the regularization parameters, τ_x, τ_z and τ_{xz}, the scale parameter ρ for Gaussian processes, and the error variance σ^2 are already known. In this linear mixed model framework, we can estimate the parameter $\boldsymbol{\theta} = (\sigma^2, \tau_x, \tau_z, \tau_{xz}, \rho)^T$ simultaneously using restricted maximum likelihood (REML) estimation. REML is superior to the maximum likelihood (ML) method in terms of adjusting the small sample bias (Zhang and Lin, 2003). The REML of our model is derived routinely (Harville, 1977) up to the usual additive constant

$$l_R = -\frac{1}{2}\log|\Sigma| - \frac{1}{2}|X^T\Sigma^{-1}X| - \frac{1}{2}(\mathbf{y} - X\hat{\boldsymbol{\beta}})^T\Sigma^{-1}(\mathbf{y} - X\hat{\boldsymbol{\beta}}) + c, \tag{4.15}$$

where c is constant. Another advantage of using REML is that it accounts for the degrees-of-freedom adjustment of replacing $\boldsymbol{\beta}$ with $\hat{\boldsymbol{\beta}}$ in Expression (4.15) (Breslow and Clayton, 1993). Taking the derivatives of (4.15) with respect to $\boldsymbol{\theta}$, the estimates of $\boldsymbol{\theta}$ are obtained by solving

$$\begin{aligned}
&\frac{\partial l_R}{\partial \sigma^2} = -\frac{1}{2}\text{Tr}(P) + \frac{1}{2}(\mathbf{y} - X\hat{\boldsymbol{\beta}})^T\Sigma^{-1}\Sigma^{-1}(\mathbf{y} - X\hat{\boldsymbol{\beta}}) = 0,\\
&\frac{\partial l_R}{\partial \tau_\alpha} = -\frac{1}{2}\text{Tr}\left(\frac{\partial \Sigma}{\partial \tau_\alpha}P\right) + \frac{1}{2}(\mathbf{y} - X\hat{\boldsymbol{\beta}})^T\Sigma^{-1}\frac{\partial \Sigma}{\partial \tau_\alpha}\Sigma^{-1}(\mathbf{y} - X\hat{\boldsymbol{\beta}}) = 0, \quad \alpha \in \{x, z, xz\},\\
&\frac{\partial l_R}{\partial \rho} = -\frac{1}{2}\text{Tr}\left(\frac{\partial \Sigma}{\partial \rho}P\right) + \frac{1}{2}(\mathbf{y} - X\hat{\boldsymbol{\beta}})^T\Sigma^{-1}\frac{\partial \Sigma}{\partial \rho}\Sigma^{-1}(\mathbf{y} - X\hat{\boldsymbol{\beta}}) = 0,
\end{aligned} \tag{4.16}$$

where $P = \Sigma^{-1} - \Sigma^{-1}X(X^T\Sigma^{-1}X)^{-1}X^T\Sigma^{-1}$, and $\frac{\partial\Sigma}{\partial\rho} = \tau_z\frac{\partial K_z}{\partial\rho} + \tau_{xz}\frac{\partial K_{xz}}{\partial\rho}$. The 5×5 information matrix $\mathcal{I}(\boldsymbol{\theta})$ has the i, jth entry as

$$\mathcal{I}(\boldsymbol{\theta})_{ij} = \frac{1}{2}\text{Tr}\left(P\frac{\partial\Sigma}{\partial\theta_i}P\frac{\partial\Sigma}{\partial\theta_j}\right), \tag{4.17}$$

and the variance of $\hat{\boldsymbol{\theta}}$ can be estimated through the expression of the information matrix. Equation (4.16) can be solved using an iteration method such as Fisher's scoring method. In practice, the sample size n may be small, for instance the Type II diabetes data contains only 35 observations, while Model (4.10) includes two fixed-effect parameters and three smoothing parameters. We may have problems with overparameterization, and it may cause a negative estimate of the variance components based on REML. In such case, the step-halving method can be adopted (Jennrich and Schluchter, 1986), but still the corresponding variance component can be estimated as very close to zero.

4.3.2 Profile REML Approach for Estimating Variance Components

In this section, we suggest a modification to the REML estimation of the variance components so that the estimate of the error components always remains in the parameter space. This new approach makes the use of the profile restricted maximum likelihood (p-REML). The covariance of $\mathbf{y}$ in Expression (4.12) can be written as $\Sigma = \sigma^2\Sigma_\lambda$, where $\Sigma_\lambda = (I + \lambda_x^{-1}BB^T + \lambda_z^{-1}K_z + \lambda_{xz}^{-1}K_{xz})$. Defining the matrix $P_\lambda = \Sigma_\lambda^{-1} - \Sigma_\lambda^{-1}X(X^T\Sigma_\lambda^{-1}X)^{-1}X\Sigma_\lambda^{-1}$, and $P = P_\lambda/\sigma^2$, the restricted log likelihood function (4.15) can be rewritten as

$$l_R = -\frac{1}{2}(n-q)\log(\sigma^2) - \frac{1}{2}|\Sigma_\lambda| - \frac{1}{2}\log|X^T\Sigma_\lambda^{-1}X| - \frac{1}{2}\frac{\mathbf{y}^T P_\lambda \mathbf{y}}{\sigma^2} + c, \tag{4.18}$$

where $q = 2$ is the rank of X. Assuming that $\lambda_\alpha, \alpha \in \{x, z, xz\}$ are known, by solving the derivative of (4.18) with respect to σ^2 set equal to zero, the p-REML estimator of σ^2 is

$$\hat{\sigma}^2 = \frac{\mathbf{y}^T P_\lambda \mathbf{y}}{n-q}. \tag{4.19}$$

Since $P_\lambda \Sigma_\lambda$ is idempotent, $\frac{\mathbf{y}^T P_\lambda \mathbf{y}}{\sigma^2} \sim \chi^2_{\mathrm{r}(P_\lambda)}$, where $\mathrm{r}(P_\lambda) = \mathrm{Tr}(P_\lambda)$ is the rank of P_λ, the variance of $\mathrm{Var}(\hat{\sigma}^2) \approx 2\hat{\sigma}^4 \mathrm{Tr}(P_\lambda)/(n-q)^2$. Plug $\hat{\sigma}^2$ back into Expression (4.18) and we have the log profile restricted likelihood (PRL) function

$$l_{PR} = -\frac{1}{2}\log|\Sigma_\lambda| - \frac{1}{2}|X^T \Sigma_\lambda^{-1} X| - \frac{n-q}{2}\log(\mathbf{y}^T P_\lambda \mathbf{y}) + c. \tag{4.20}$$

Now we can use the similar scoring algorithm to estimate $\boldsymbol{\theta}^* = (\lambda_x^{-1}, \lambda_z^{-1}, \lambda_{xz}^{-1}, \rho)$. By simple algebra the score of the p-REML likelihood is

$$\frac{\partial l_{PR}}{\partial \theta_j^*} = -\frac{1}{2}\mathrm{Tr}\left(\frac{\partial \Sigma_\lambda}{\partial \theta_j^*} P_\lambda\right) + \frac{1}{2\hat{\sigma}^2}\mathbf{y}^T P_\lambda \frac{\partial \Sigma_\lambda}{\partial \theta_j^*} P_\lambda \mathbf{y}, j = 1, 2, 3, 4, \tag{4.21}$$

and the i, jth entry of the information matrix $\mathcal{I}^*(\boldsymbol{\theta}^*)$ for the PRL can be approximated as

$$\mathcal{I}^*(\boldsymbol{\theta}^*)_{ij} = \frac{1}{2(n-q)}\left\{(n-q-2)\mathrm{Tr}\left(\frac{\partial \Sigma_\lambda}{\partial \theta_i^*} P_\lambda \frac{\partial \Sigma_\lambda}{\partial \theta_j^*} P_\lambda\right) - \mathrm{Tr}\left(\frac{\partial \Sigma_\lambda}{\partial \theta_i^*} P_\lambda\right)\mathrm{Tr}\left(\frac{\partial \Sigma_\lambda}{\partial \theta_j^*} P_\lambda\right)\right\}. \tag{4.22}$$

Note that $\mathcal{I}^*(\boldsymbol{\theta}^*)$ is positive definite when n is large enough. Claeskens (2004) also showed the convergence of $\mathcal{I}^*(\boldsymbol{\theta}^*)$ under regular conditions so that we can apply the restricted likelihood ratio test (RLRT, see Section 4.4). Since PRL is not a true likelihood, we only use PRL for statistical test purposes, and use p-REML to obtain a better estimate of the variance components. The variances of $\boldsymbol{\theta}$ is found by plugging the p-REML estimates into (4.17).

4.4 Test for Pathway Effects

4.4.1 Test for Two Zero Variance Components

One of the primary problems in the study of pathway based analysis is testing the "overall" pathway effects. Recall that the meaning of "overall" refers to either the main effect of a pathway, the interaction effect associated with the pathway, or both. In Model (4.10), two random effects are involved with the overall pathway effects. Thus, the hypothesis for testing the overall pathway effect is

$$H_0 : \tau_z = \tau_{xz} = 0 \text{ vs. } H_a : \tau_z > 0 \text{ or } \tau_{xz} > 0, \tag{4.23}$$

which is equivalent to the following test

$$H_0 : \lambda_z^{-1} = \lambda_{xz}^{-1} = 0 \text{ vs. } H_a : \lambda_z^{-1} > 0 \text{ or } \lambda_{xz}^{-1} > 0. \tag{4.24}$$

For this type of test problem, a likelihood ratio test (LRT) is most commonly used. Note that parameter space for $\boldsymbol{\theta} = (\lambda_x^{-1}, \lambda_z^{-1}, \lambda_{xz}^{-1}, \rho)^T$ equals $[0, \infty)^3 \times (0, \infty)$ (to avoid abuse of notation, in this section, $\boldsymbol{\theta}$ and $\mathcal{I}$ stand for counterparts of PRL). The true parameters $\boldsymbol{\theta}_0$ are either in the interior or on the boundary of the parameter space, so the LRT is nonstandard. Vu and Zhou (1997) generalized the hypothesis test for both interior and boundary problems within a setting of mixed regression fitting, so it allows the nonidentically distributed response variable y_i's to depend on the covariates and allows the random effects to induce dependence between the response values. (Claeskens, 2004) further extended the non-standard LRT test to the profile restricted likelihood ratio test (RLRT), focusing on nonparametric mixed models with spline fitting.

Following (Claeskens, 2004), we apply RLRT to test hypothesis (4.24). Under this hypothesis, the RLRT test statistics, D, is the deviance of two times the log PRL, $-2l_{PR}(\boldsymbol{\theta})$, i.e. $D = 2l_{PR}(\boldsymbol{\theta}) - 2l_{PR}(\boldsymbol{\theta}_0)$. Note that D is the same using either l_R or l_{PR}. Assuming that the corresponding regular conditions in Vu and Zhou (1997) are satisfied for the PRL function model, D converges to

$$D \to \inf_{\boldsymbol{\theta}\in\tilde{C}_0} \|U - \boldsymbol{\theta}\|^2 - \inf_{\boldsymbol{\theta}\in\tilde{C}} \|U - \boldsymbol{\theta}\|^2, \tag{4.25}$$

where $\tilde{C} = \{\tilde{\boldsymbol{\theta}} : \tilde{\boldsymbol{\theta}} = \mathcal{I}(\boldsymbol{\theta}_0)^{T/2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0), \boldsymbol{\theta} \in C_\Omega\}$ is the orthonormal transformation of the cone approximation, C_Ω, of the parameter space Ω with $\boldsymbol{\theta}_0$ as the vertex, and $\tilde{C}_0 = \{\tilde{\boldsymbol{\theta}} : \tilde{\boldsymbol{\theta}} = \mathcal{I}(\boldsymbol{\theta}_0)^{T/2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0), \boldsymbol{\theta} \in C_{\Omega_0}\}$ is the orthonormal transformed cone approximation of the parameter space Ω_0 under the null hypothesis. U is a random vector from $N(0, I)$, and $\mathcal{I}(\boldsymbol{\theta}_0)^{T/2}$ is the right Cholesky square root of p-REML information matrix, i.e. $\mathcal{I}(\boldsymbol{\theta}_0) = [\mathcal{I}(\boldsymbol{\theta}_0)]^{1/2}\,[\mathcal{I}(\boldsymbol{\theta}_0)]^{T/2}$.

Note that under the null hypothesis, $\boldsymbol{\theta}_0 = (\lambda_x^{-1}, 0, 0, \rho)^T$, ρ is inestimable. We suggest estimating the parameters with ρ fixed at the average of $\|\mathbf{z} - \mathbf{z}'\|^2$ (average on pairwise observations) to not only reduce the parameter space dimensions but also achieve a better fit. Let $\boldsymbol{\theta} = (\lambda_x^{-1}, \lambda_z^{-1}, \lambda_{xz}^{-1})^T = (\theta_1, \theta_2, \theta_3)^T$. Now the cone parameter spaces are reduced to $C_\Omega = [0, \infty)^3$ and $C_{\Omega_0} = [0, \infty) \times \{0\} \times \{0\}$. However, in this problem, all three parameters can be on the boundaries and the orthonormal transformation for the nuisance parameter θ_1 is not invariant, which leads to a transformation for 3 dimensional space. The calculation of (4.25) in a 3 dimensional space becomes considerably more difficult when the information matrix is not diagonal. To simplify the calculation, we consider the special case that $\theta_1 \approx 0$, which is a reasonable consideration for the Type II diabetes data in a later section, where the p-REML estimates of θ_1's are very close to zero for most pathways.

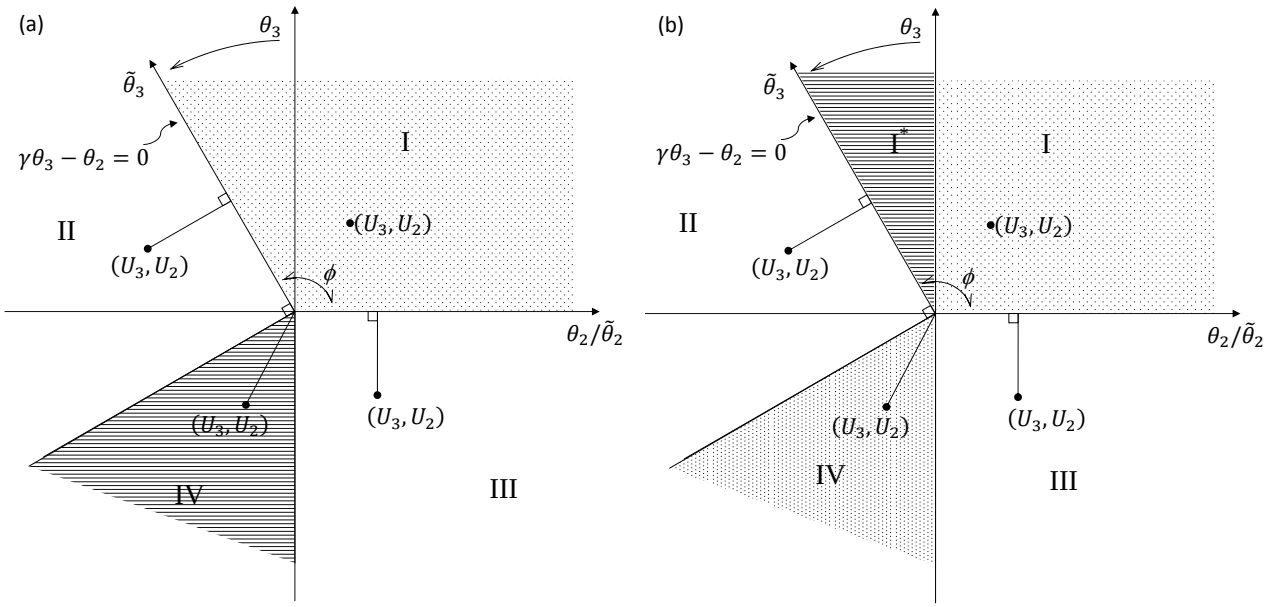

Figure 4.1: Diagram of the parameter space of RLRT for testing two zero variance components (a), and testing the P-E interaction effect (b).

Now the parameter space is reduced to 2 dimensions. Under the orthonormal transformation, the cone spaces become to $\tilde{C} = \{\boldsymbol{\theta} : \gamma\theta_3 - \theta_2 \geq 0, \theta_3 \geq 0\}$, and $\tilde{C}_0 = \{\boldsymbol{\theta} : \theta_3 = \theta_2 = 0\}$, where $\gamma = \tilde{\mathcal{I}}_{23} \cdot |\tilde{\mathcal{I}}(\boldsymbol{\theta}_0)|^{-1/2}$ is the slope of the axis θ_3 after transformation as shown in Figure 4.1(a). To account for the fact that θ_1 is estimated, $\tilde{\mathcal{I}}(\boldsymbol{\theta}_0)$ is defined from the 3×3 information matrix $\mathcal{I}(\boldsymbol{\theta}_0)$ as

$$\tilde{\mathcal{I}}(\boldsymbol{\theta}_0) = \begin{bmatrix} \tilde{\mathcal{I}}_{22} & \tilde{\mathcal{I}}_{23} \\ \tilde{\mathcal{I}}_{32} & \tilde{\mathcal{I}}_{33} \end{bmatrix} = \begin{bmatrix} \mathcal{I}_{22} & \mathcal{I}_{23} \\ \mathcal{I}_{32} & \mathcal{I}_{33} \end{bmatrix} - \begin{bmatrix} \mathcal{I}_{21} \\ \mathcal{I}_{31} \end{bmatrix} \mathcal{I}_{11}^{-1} \left[\mathcal{I}_{12}, \mathcal{I}_{13}\right].$$

From the graphic point of view, the representation of the test statistics (4.25) is determined by the minimum distance of the independent normal vector $U = (U_2, U_3)^T$ to $\boldsymbol{\theta}$. Under the alternative hypothesis, the minimum distance, $\inf_{\boldsymbol{\theta} \in \tilde{C}} \|U - \boldsymbol{\theta}\|^2$, can be understood as the projection of U on the cone space $\tilde{C}$ when U is outside of the cone. As shown in Figure 4.1(a), the representations of $\inf_{\boldsymbol{\theta} \in \tilde{C}} \|U - \boldsymbol{\theta}\|^2$ are different in the four regions of the plane

with coordinates (θ_2, θ_3)

$$\inf_{\boldsymbol{\theta}\in\tilde{C}} \|U-\boldsymbol{\theta}\|^2 = \begin{cases} 0 & \theta_3 \geq 0,\ \ \gamma\theta_3 - \theta_2 \geq 0, & I \\ U_2^2 + U_3^2 - (\gamma U_2 + U_3)^2/(1+\gamma^2) & \theta_3 + \gamma\theta_2 \geq 0,\ \ \gamma\theta_3 - \theta_2 < 0, & II \\ U_3^2 & \theta_3 < 0,\ \theta_2 \geq 0, & III \\ U_2^2 + U_3^2 & \theta_3 + \gamma\theta_2 < 0,\ \ \theta_2 < 0, & IV. \end{cases} \tag{4.26}$$

The area proportions, $(\phi, 1/4, 1/4, 1/2-\phi)$ as in the aforementioned order, of these four regions determine the probabilities that the vector U lies in which region, where $\phi = \cos^{-1}\left(\gamma\cdot(1+\gamma^2)^{-1/2}\right) = \tilde{\mathcal{I}}_{23}\cdot(\tilde{\mathcal{I}}_{22}\tilde{\mathcal{I}}_{33})^{-1/2}$.

Under the null hypothesis, the parameters space is reduced to the origin of the plane, thus according to Vu and Zhou (1997)

$$\inf_{\boldsymbol{\theta}\in\tilde{C}_0} \|U-\boldsymbol{\theta}\|^2 = U_2^2 + U_3^2.$$

Then the asymptotic distribution of D is the difference of the above two representations

$$D \to \begin{cases} U_2^2 + U_3^2 & \text{with probability } \phi, & I \\ (\gamma U_2 + U_3)^2/(1+\gamma^2) & \text{with probability } 1/4, & II \\ U_2^2 & \text{with probability } 1/4, & III \\ 0 & \text{with probability } 1/2-\phi, & IV. \end{cases} \tag{4.27}$$

Note that because U_2 and U_3 are independent, thus $(\gamma U_2 + U_3)/\sqrt{1+\gamma^2} \sim N(0,1)$, and the

final approximate asymptotic distribution of D is

$$D \sim \phi\chi_2^2 + 0.5\chi_1^2 + (0.5 - \phi)\chi_0^2. \tag{4.28}$$

In this dissertation, we suppose $\lim_{n\to\infty} |\gamma| < \infty$. If $\lim_{n\to\infty} |\gamma| \to \infty$, the representation of $\inf_{\boldsymbol{\theta}\in\tilde{C}} \|U - \boldsymbol{\theta}\|^2$ is in different form (Vu and Zhou, 1997) and the asymptotic distribution of D may be different. An additional approximation is that we obtain γ with a finite sample size under the null hypothesis, so we assume that n is large enough that the finite γ is close to the converged value.

4.4.2 Test for the P-E Interaction Effect

The RLRT for two variance components introduced above allows us to test the overall pathway effect. Furthermore, we may be attracted to testing single variance components, such as testing the P-E effect, given that the overall the pathway effect test is significant. The hypothesis of this problem is

$$H_0 : \lambda_{xz}^{-1} = 0 \ \text{ vs. } \ H_a : \lambda_{xz}^{-1} > 0, \tag{4.29}$$

which is equivalent to testing $H_0 : \tau_{xz} = 0$ vs. $H_a : \tau_{xz} > 0$. The RLRT test statistics $d = 2l_{PR}(\boldsymbol{\theta}) - 2l_{PR}(\boldsymbol{\theta}_0)$ for one variance component in semiparametric model with PRL was also suggested by Claeskens (2004), and an exact RLRT algorithm was proposed by Crainiceanu et al. (2005). Unfortunately, this exact RLRT method cannot apply to test (4.29) for Model (4.10). In their work, there are no random effects in the model under the null hypothesis, thus d can be represented exactly as the form of a mixture of chi-square distribution. On the contrary, our model (4.10) under the null hypothesis of (4.29)

contains two random effects $\mathbf{r}_x$ and $\mathbf{r}_z$, which makes it impossible to represent d exactly.

The second choice is to use the method described in the previous section using an asymptotic distribution. However, we need the same approximations; that is, we fix ρ and assume that the relationship between the response and the environmental variable is almost linear, i.e. $\tau_x \approx 0$. Then similarly, the parameters cone space is reduced to 2 dimensions. One interesting parameter $\theta_3 = \lambda_{xz}^{-1}$, and one nuisance parameter $\theta_2 = \lambda_z^{-1}$, both have the true values on the boundary. Thus, $C_\Omega = [0, \infty) \times [0, \infty)$ and $C_{\Omega_0} = [0, \infty) \times \{0\}$.

Under the approximations described above, the asymptotic representation of 2 times the log PRL function under the null hypothesis is

$$\inf_{\boldsymbol{\theta} \in \tilde{C}_0} \|U - \boldsymbol{\theta}\|^2 = 0 \cdot I(U_2 > 0) + U_2^2 I(U_2 \leq 0) + U_3^2, \tag{4.30}$$

where $I(\cdot)$ is the indicator function. The representation under the alternative hypothesis is the same as in (4.26), but because the cone under the null hypothesis is no longer the origin of the (θ_2, θ_3) plane, $\inf_{\boldsymbol{\theta} \in \tilde{C}_0} \|U - \boldsymbol{\theta}\|^2$ has two regions as shown by (4.30). Now we must divide the plane with coordinates (θ_2, θ_3) into five regions and set the approximated asymptotic representation of d as (see Figure 4.1(b))

$$d \to \begin{cases} U_3^2 & \text{with probability } 1/4, & I \\ U_2^2 + U_3^2 & \text{with probability } \phi - 1/4, & I^* \\ (\gamma U_2 + U_3)^2/(1+\gamma^2) & \text{with probability } 1/4, & II \\ 0 & \text{with probability } 1/4, & III \\ 0 & \text{with probability } 1/2 - \phi, & IV. \end{cases} \tag{4.31}$$

Thus, we have the asymptotic distribution of d for testing $\theta_3 = \lambda_{xz}^{-1} = 0$ or $\tau_{xz} = 0$

$$d \sim (\phi - 0.25)\chi_2^2 + 0.5\chi_1^2 + (0.75 - \phi)\chi_0^2, \tag{4.32}$$

where ϕ is calculated through γ under hypothesis (4.29).

In many cases, the relationship between the response and the environmental variable is not linear, i.e. τ_x is significant and not equal to 0, then we are in the 3 dimension space to derive the asymptotic distribution of the d, which becomes arduous. In this situation, we adopt a score test approach based on the REML function (4.15) which was proposed by Lin (1997) in a mixed model. The asymptotic distribution of the REML score may not converge to a standard normal distribution, Zhang and Lin (2003) suggested using the scaled chi-square approximation of the test statistics. More generally, the REML score for covariance component $\tau_\alpha, \alpha \in \{x, z, xz\}$ of (4.16) can also be written as

$$\frac{\partial l_R}{\partial \tau_\alpha} = \frac{1}{2}(P\mathbf{y})^T \frac{\partial \Sigma}{\partial \tau_\alpha} P\mathbf{y} - \frac{1}{2}\text{Tr}\left(P\frac{\partial \Sigma}{\partial \tau_\alpha}\right),$$

where we used identity $(\mathbf{y} - X\hat{\boldsymbol{\beta}})^T\Sigma^{-1} = (P\mathbf{y})^T$. P can be expressed as $P = \Gamma(\Gamma^T\Sigma\Gamma)^{-1}\Gamma^T$ (Searle et al., 1992), where Γ^T is $(n-q) \times n$ matrix with full row rank $n - q$ ($q = 2$ is the rank of X). The matrix Γ^T satisfies $\Gamma^T X = 0$ and $\Gamma^T\mathbf{y} \sim N(0, \Gamma^T\Sigma\Gamma)$. Thus the REML version score test statistics can be written as

$$U_{\tau_\alpha} = \frac{1}{2}(P\mathbf{y})^T \frac{\partial \Sigma}{\partial \tau_\alpha} P\mathbf{y} = \tilde{\mathbf{y}}^T M \tilde{\mathbf{y}}, \tag{4.33}$$

where $\tilde{\mathbf{y}} = (\Gamma^T\Sigma\Gamma)^{-\frac{1}{2}}\Gamma^T\mathbf{y}$ with $\tilde{\mathbf{y}} \sim N(0, I_{n-q})$, and $M = \frac{1}{2}(\Gamma^T\Sigma\Gamma)^{-\frac{1}{2}}\Gamma^T\frac{\partial \Sigma}{\partial \tau_\alpha}\Gamma(\Gamma^T\Sigma\Gamma)^{-\frac{1}{2}}$. U_{τ_α} is the quadratic form of $\mathbf{y}$ with mean $E(U_{\tau_\alpha}) = \frac{1}{2}\text{Tr}\left(P\frac{\partial \Sigma}{\partial \tau_\alpha}\right)$ and variance $\text{Var}(U_{\tau_\alpha}) = \mathcal{I}_{jj}$, where $\mathcal{I}_{jj}$ is the corresponding entry of the information matrix (4.17) for the interesting

variance component of $\tau_\alpha \in \{\tau_x, \tau_z, \tau_{xz}\}$.

Let r denote the number of non-zero eigenvalues of M, then M can be further decomposed using the spectral decomposition as $M = H\Xi H^T = \sum_{i=1}^{r} \xi_i h_i h_i^T$, where $H = (h_1, ..., h_r)$ is $n \times r$ orthogonal normal matrix, i.e. $h_i^T h_j = \delta_{ij}$, and $\Xi = \langle \xi_i \rangle$ is $r \times r$ diagonal matrix. It follows that

$$U_{\tau_\alpha} = \tilde{\mathbf{y}}^T H \Xi H^T \tilde{\mathbf{y}} = \sum_i^r \xi_i \tilde{\mathbf{y}}^T h_i h_i^T \tilde{\mathbf{y}} \sim \sum_i^r \xi_i \chi_1^2.$$

Therefore, under H_0, the distribution of U_{τ_α} can be represented as a weighted mixture of chi-square distribution. This is because $\tilde{\mathbf{y}}^T h_i h_i^T \tilde{\mathbf{y}} \sim \chi_1^2$ since $h_i h_i^T$ is an idempotent matrix with rank 1. Because the calculation for ξ_i's is intensive, we follow Zhang and Lin (2003) in using the Satterthwaite method to approximate the distribution of U_{τ_α} by a scaled chi-square distribution $\kappa\chi_\nu^2$, where $\kappa = \mathcal{I}_{jj}/2E(U_{\tau_\alpha})$, and $\nu = 2E(U_{\tau_\alpha})^2/\mathcal{I}_{jj}$. Zhang and Lin (2003) also suggested to further account for the fact that $\boldsymbol{\theta} = (\sigma^2, \tau_x, \tau_z, \tau_{xz}, \rho)^T$ is estimated, so that κ and ν are calculated by replacing $\mathcal{I}_{jj}$ with the efficient information $\tilde{\mathcal{I}}jj = \mathcal{I}_{jj} - \mathcal{I}_{j\vartheta}\mathcal{I}_{\vartheta\vartheta}^{-1}\mathcal{I}_{j\vartheta}^T$, where $\mathcal{I}_{j\vartheta}$ and $\mathcal{I}_{\vartheta\vartheta}$ are the corresponding vector and matrix if we rearrange the 5×5 information matrix $\mathcal{I}(\boldsymbol{\theta})$ as

$$\mathcal{I}(\boldsymbol{\theta}) = \begin{bmatrix} \mathcal{I}_{jj} & \mathcal{I}_{j\vartheta} \\ \mathcal{I}_{j\vartheta}^T & \mathcal{I}_{\vartheta\vartheta} \end{bmatrix}.$$

In this dissertation, we are particularly interested in testing the P-E interaction effect, i.e., τ_{xz}.

Table 4.1: Assessments of estimating f_x, f_z and f_{xz} simulated by (4.34) using REML and p-REML procedures with ρ estimated from initial value 2 or fixed at 2. Total runs number 200 for each scenario, and the average values are reported.

		fitted p		$\hat{\rho}$	$f_x \sim \hat{f}_x$			$f_z \sim \hat{f}_z$			$f_{xz} \sim \hat{f}_{xz}$		
	n	(true p)	$\hat{\sigma}^2$	(initial ρ)	Int	Slope	R^2	Int	Slope	R^2	Int	Slope	R^2
		30(30)	0.34	2130(2)	-0.38	1.00	0.97	-0.01	10.51	0.90	-0.14	5.19	0.46
	100	40(30)	0.29	1824(2)	-0.55	1.06	0.96	0.01	11.65	0.89	-0.11	4.17	0.50
REML		50(30)	0.32	1929(2)	-1.53	1.26	0.96	-0.02	16.07	0.87	-0.13	5.28	0.48
ρ		30(30)	0.26	1604(2)	-1.15	1.17	0.98	0.09	5.87	0.93	-0.17	3.70	0.54
estimated	150	40(30)	0.29	1814(2)	-0.68	1.18	0.97	-0.09	8.65	0.91	-0.15	3.95	0.48
		50(30)	0.35	2054(2)	-1.24	1.18	0.97	0.06	12.32	0.90	-0.15	4.79	0.45
		30(30)	6.9e-10	2	0.10	0.99	0.99	0.01	0.85	0.99	0.01	1.44	0.90
	100	40(30)	8.6e-10	2	0.13	0.98	0.98	0.02	0.86	0.98	0.00	1.40	0.90
REML		50(30)	8.5e-10	2	0.16	0.98	0.96	0.01	0.86	0.98	0.01	1.41	0.88
ρ		30(30)	8.5e-10	2	0.05	0.99	0.99	0.01	0.84	0.99	0.01	1.41	0.93
fixed	150	40(30)	8.7e-10	2	-0.00	1.00	0.99	0.00	0.84	0.99	-0.01	1.40	0.92
		50(30)	7.1e-10	2	0.10	0.99	0.99	0.02	0.85	0.99	0.00	1.38	0.91
		30(30)	0.04	3.96(2)	-0.24	1.04	1.00	0.01	0.85	0.99	-0.04	1.38	0.90
	100	40(30)	0.07	3.36(2)	-0.19	1.03	1.00	-0.01	0.87	0.99	-0.05	1.46	0.89
p-REML		50(30)	0.09	4.72(2)	-0.31	1.04	1.00	0.06	0.90	0.98	-0.04	1.44	0.88
ρ		30(30)	0.02	3.00(2)	-0.28	1.04	1.00	0.01	0.85	0.99	-0.05	1.29	0.92
estimated	150	40(30)	0.02	3.63(2)	-0.29	1.04	1.00	0.01	0.86	0.99	-0.04	1.29	0.91
		50(30)	0.04	3.19(2)	-0.13	1.02	1.00	0.01	0.85	0.99	-0.02	1.37	0.91
		30(30)	0.04	2	-0.08	1.01	1.00	0.02	0.85	0.99	-0.01	1.64	0.91
	100	40(30)	0.11	2	-0.17	1.03	0.99	-0.00	0.88	0.98	-0.03	1.52	0.91
p-REML		50(30)	0.11	2	-0.12	1.02	0.99	-0.00	0.90	0.98	-0.01	1.38	0.91
ρ		30(30)	0.02	2	-0.08	1.01	1.00	0.02	0.86	0.99	-0.01	1.34	0.93
fixed	150	40(30)	0.03	2	-0.11	1.02	1.00	-0.01	0.85	0.99	-0.05	1.37	0.92
		50(30)	0.04	2	-0.09	1.01	1.00	0.02	0.86	0.99	-0.02	1.44	0.92

4.5 Simulation Study

4.5.1 Parameters Estimation

We carried out the simulation study to evaluate the accuracies of the estimators; 200 runs were performed for each of the simulation scenarios. Let p denote the number of genes in the pathway and n denote the number of observations. We considered a setup that mimics the real diabetes pathway data with a total of 50 genes within a pathway. The

true model of the ith observations is

$$y_i = f_x(x_i) + f_z(\boldsymbol{z}_i^T) + f_{xz}(x_i, \boldsymbol{z}_i^T) + \epsilon_i,\ \epsilon_i \sim N(0, \sigma^2)$$

with nonparametric functions

$$\begin{aligned} f_x(x_i) &= 5.6 + 0.1x_i + \cos(x_i\pi/18), \\ f_z(\boldsymbol{z}_i^T) &= a \cdot \boldsymbol{z}_i^{(30)} \exp\left(-0.2|\bar{\boldsymbol{z}}|_i^{(30)}\right)/5, \\ f_{xz}(x_i, \boldsymbol{z}_i^T) &= b \cdot e^{x_i/10} \sin\left(\bar{\boldsymbol{z}}_i^{(30)}\right) \cos\left(\bar{\boldsymbol{z}}_i^{(30)}\right)/8, \end{aligned} \tag{4.34}$$

where $\boldsymbol{z}_i^{(30)}, |\bar{\boldsymbol{z}}|_i^{(30)}$ and $\bar{\boldsymbol{z}}_i^{(30)}$ stand for $\sum_{j=1}^{30} z_{ij}, \sum_{j=1}^{30} |z_{ij}|/30$ and $\sum_{j=1}^{30} z_{ij}/30$. We sample x_i and $z_{ij}(j = 1, ..., 50)$ from $\mathrm{Uniform}[18, 36]$ and $N(0, 1)$, respectively. Furthermore, a and b are parameters to control the magnitude of the nonparametric functions respectively. In this section they are fixed at $a = 1.5$ and $b = 2$. In the true model (4.34), a total of 30 genes, $z_{i1}, ..., z_{i30}$, are involved. However in a real situation, we may fit the model with extra genes that are not involved in the true model. Thus we consider the following settings for Model (4.34):

Setting 1: $n = 100/150$, true $p = 30$, fitted $p = 30$, $\sigma^2 = 0.2^2$,

Setting 2: $n = 100/150$, true $p = 30$, fitted $p = 40$, $\sigma^2 = 0.2^2$,

Setting 3: $n = 100/150$, true $p = 30$, fitted $p = 50$, $\sigma^2 = 0.2^2$.

For each setting, two sample sizes $n = 100$ and 150 were considered. In Section 4.3 we introduced two methods to estimate the variance components using REML and p-REML. We are particularly interested in comparing the performance of these two methods. One of the difficulties of solving Equation (4.16) or (4.21) using a scoring method is finding the initial values for $\boldsymbol{\theta}$ or $\boldsymbol{\theta}^*$, since there are no analytic expressions to roughly obtain those initial values. Breslow and Clayton (1993) suggested starting the variance parameters

Table 4.2: Type I error and power of RLRT of overall pathway effect with ρ fixed at different values and estimated. Simulated samples size $n = 100$, and both used and true gene number equal to $p = 30$.

	ρ	$b = 0$	0.2	0.35	0.5	1
$a = 0$	2	0.03	0.34	0.91	1.00	1.00
	5	0.02	0.34	0.89	0.99	1.00
	10	0.02	0.30	0.88	0.99	1.00
	estimated	0.03	0.33	0.87	0.99	1.00
		$a = 0$	0.05	0.1	0.2	0.5
$b = 0$	2	0.03	0.07	0.37	0.96	1.00
	5	0.02	0.07	0.37	0.95	1.00
	10	0.02	0.06	0.34	0.91	1.00
	estimated	0.03	0.06	0.34	0.93	1.00

from small positive values within a complex situation. We started the variance components with $(\sigma^2, \tau_x, \tau_z, \tau_{xz})^T = (0.001, 0.001, 0.001, 0.001)^T$, which is equivalent to starting with $(\sigma^2, \lambda_x^{-1}, \lambda_z^{-1}, \lambda_{xz}^{-1})^T = (0.001, 1, 1, 1)$ for p-REML. For scale parameter ρ, we can either fix or estimate it. In this simulation study, we choose the initial value $\rho = 2$ which is the average of $\|\boldsymbol{z} - \boldsymbol{z}'\|^2$ on all pairwise observations if it is estimated. We also compare the results with ρ fixed at 2. Note that if ρ is estimated, we consider two possible ways. One way is to perform a two-step procedure where we first fix ρ at 2 and evaluate $(\sigma^2, \tau_x, \tau_z, \tau_{xz})$ until convergence and then use the results with $\rho = 2$ as the initial values to evaluate $(\sigma^2, \tau_x, \tau_z, \tau_{xz}, \rho)$ until convergence. The other way is to evaluate $(\sigma^2, \tau_x, \tau_z, \tau_{xz}, \rho)$ together from an initial value $(0.001, 0.001, 0.001, 0.001, 2)^T$. The simulation results show that the former method is more stable, so only these results are shown. Similarly, a two-step procedure was used for p-REML when ρ is estimated.

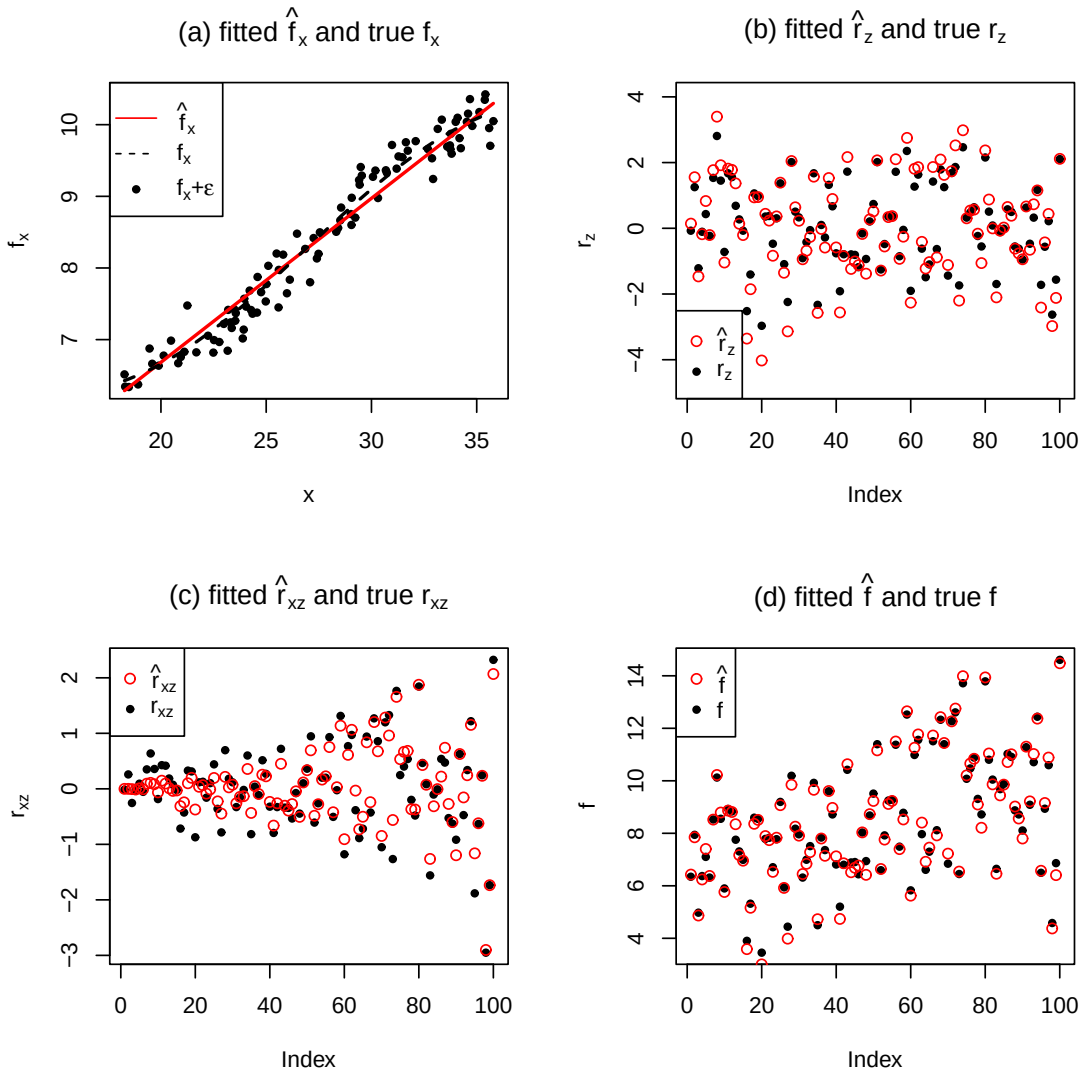

Figure 4.2: Selected example of fitting results of setting 1. Because of the high dimensionality, $\mathbf{r}_z, \mathbf{r}_{xz}$ and $\mathbf{f}$ are plotted vs. the observation index only.

To demonstrate the fitting results, Figure 4.2 shows one selected example of setting 1 comparing estimated $\mathbf{f}, \mathbf{f}_x, \mathbf{r}_z$ and $\mathbf{r}_{xz}$ with the true ones. The overall response $\mathbf{f}$ is fitting very well as shown in Figure 4.2(d). As shown in Figure 4.2(b) and (c), there is not much identifiability issue since both the fitted pathway effect $\hat{\mathbf{r}}_z$ and fitted the interaction effect $\hat{\mathbf{r}}_{xz}$ capture the patterns of the true ones very well.

To have a overall evaluation of the goodness-of-fit of the nonparametric function f_x, f_z and f_{xz}, we followed the techniques used by Liu et al. (2007), who suggested regressing the true nonparametric functions on the fitted ones. By reporting the average intercepts, slopes and R^2's from these regressions, the goodness-of-fit of the fitted nonparametric functions can be assessed empirically. The closer to 0 and 1 of the intercepts and slopes are and the closer to 1 of R^2 is, the better the performance of the estimation is. In Table 4.1 we summarized the goodness-of-fit of $f_\alpha, \alpha \in \{x, z, xz\}$ for 200 hundred runs. The scenarios of three settings were used in four procedures: I) REML with ρ estimated, II) REML with ρ fixed at 2, III) p-REML with ρ estimated, and IV) p-REML with ρ fixed at 2. It can be seen that the performance of using procedure I) is not so good; ρ goes to an extremely large value and f_α's deviate from $\hat{f}_\alpha$'s. This may be because the REML likelihood function dose not have a maximum and the likelihood increases or becomes flat with ρ. In such a case, the entries of K_z becomes a matrix of ones. One solution when the REML function becomes flat with ρ is to fix ρ at the turning point of the REML function. In procedure II) we fixed ρ at 2. The average of $\|\boldsymbol{z} - \boldsymbol{z}'\|^2$ on all pairwise observations is

Table 4.3: Type I error and power of RLRT of overall pathway effect with fitted genes number p equal or larger than true one $p = 30$. Simulated samples size $n = 60$ and $n = 35$. The parameter ρ is fixed at 2.

	n	used p	$b = 0$	0.2	0.35	0.5	1
$a = 0$	60	30	0.03	0.18	0.57	0.88	1.00
		50	0.03	0.15	0.48	0.76	0.99
	35	30	0.04	0.10	0.27	0.46	0.85
		50	0.03	0.08	0.23	0.38	0.78
			$a = 0$	0.1	0.2	0.5	1.5
$b = 0$	60	30	0.03	0.15	0.51	0.72	0.72
		50	0.03	0.13	0.41	0.72	0.76
	35	30	0.04	0.09	0.25	0.56	0.63
		50	0.03	0.05	0.18	0.43	0.55

very close to 2 and using this ρ allows us to avoid having extreme values for the entries of K_z. The performance of this procedure is improved significantly; all the R^2 values are over 90% and close to 1, and the intercepts and slopes of the regressions are close to 0 and 1. However, $\hat{\sigma}^2$ values are all close to zero. The zero error component happens in REML estimation (Searle et al., 1992), especially with high dimensional parameter spaces.

Table 4.1 shows that the performance is much better for the two p-REML procedures. Not only is the fitting of nonparametric functions very good, but the estimate of error variance component $\hat{\sigma}^2$ is close to the true value. As expected, fitting with extra genes introduces more error, which results in the increase of $\hat{\sigma}^2$. This is because fitting irrelevant genes is equivalent to introducing more noise into the model. However, the results show little difference in fitting f_α's for differently used gene numbers. Increasing the number of observations is expected to improve the fitting performance. Although overall there is no much difference between $n = 100$ and 150, there is slight improvement in fitting the P-E interaction effect. This can be seen from the fact that R^2 increases and the slope of regressing f_{xz} on $\hat{f}_{xz}$ is closer to 1 for $n = 150$. The overall goodness-of-fit using p-RMEL is

Table 4.4: Type I error and power of RLRT and score test of P-E interaction with ρ fixed at different values. Fitted and used gene numbers are equal to $p = 5$, and $n = 100$.

	ρ	$b = 0$	0.1	0.2	0.35	0.5	0.8	1
	2	0.04	0.24	0.58	0.95	1.00	1.00	1.00
RLRT	5	0.04	0.24	0.64	0.98	1.00	1.00	1.00
	10	0.03	0.24	0.67	0.97	1.00	1.00	1.00
score	2	0.08	0.31	0.68	0.98	1.00	1.00	1.00
test	5	0.06	0.30	0.72	0.97	1.00	1.00	1.00
	10	0.06	0.26	0.72	0.98	1.00	1.00	1.00

very good, except there are small biases: the regression slope of f_z on $\hat{f}_z$ is slightly smaller than 1 and the the regression slope of f_{xz} on $\hat{f}_{xz}$ is slightly larger than one. This means that f_z is overestimated and f_{xz} is slightly underestimated. However, for each f_z and f_{xz}, the

fitted results can explain most of the variations as all the R^2 values are very close to 1. We also realized that the fitting of $f_z + f_{xz}$ is much better than individual ones (the regression parameters of $f_z + f_{xz}$ on $\widehat{f_z + f_{xz}}$ are not shown), which is easy to be understood if we can treat $\mathbf{r} = \mathbf{r}_z + \mathbf{r}_{xz}$ as one random effect with covariance $\tau_z K_z + \tau_{xz} K_{xz}$. This indicates that there is no bias in fitting $f_z + f_{xz}$, but the weight between f_z and f_{xz} might be biased. The reason for this can be understood from the interaction kernel expression (4.6). It can be seen that if the entries of matrix $xx' + k_x(x, x')$ are close to each other, then $\tau_z K_z + \tau_{xz} K_{xz}$ is nothing more than a scalar times K_z, and we will have overestimation of f_z. However, this bias is not too significant, because the good fit of $f_z + f_{xz}$ and the high R^2 values of fitting f_{xz} indicate that it has little influence on testing either the overall pathway or the P-E interaction effect.

4.5.2 Test Study

To obtain better convergence, for the rest of this dissertation we adopt the Marquardt procedure as a scoring method. With the Marquardt method we have flexible iteration steps, this is

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} + \left[\mathcal{I}(\boldsymbol{\theta}^{(k)}) + \delta^{(k)} I\right]^{-1} \left.\frac{\partial l_R}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}^{(k)}},$$

where l_R, $\mathcal{I}$, and $\boldsymbol{\theta}$ are replaced by the counterparts of the p-REML procedure when it is required. The scalar $\delta^{(k)}$ partially determines the step size and I is the identity matrix. If $\delta^{(k)}$ is small, the procedure approximates a scoring method. If $\delta^{(k)}$ is large, a small step is taken in approximately the direction of the scoring method. We modify $\delta^{(k)}$ accordingly to achieve increasing likelihood. In this dissertation, we start from $\delta^{(k)} = (1.0 \times 10^{-5}) \times \frac{\text{Tr}(\mathcal{I}(\boldsymbol{\theta}))}{\text{number of } \theta_i' \text{s}}$ to make the initial step size as large as possible.

We first studied the performance of RLRT of testing two zero variance components

under hypothesis (4.24). In this simulation study we are particularly interested in two issues: how RLRT performs at different fixed ρ values since we prefer to estimate the parameters with ρ fixed and how the performance degrades with irrelevant genes included in the model. The true model used and the data generating method are the same as described for (4.34) in Section 4.5.1. For both issues, we first set $a = 0$ and vary b, and then set $b = 0$ and vary a. It turns out the test is very powerful when both a and b are not equal to zero, so no simulation on this situation is shown here. For all cases, the total number of simulation runs is 1000 times. In addition, the function $f_x(\cdot)$ in (4.34) has a trivial nonlinear component, so we can apply RLRT in this simulation.

For the first issue, we consider the case where the sample size is $n = 100$, and both the true and used gene numbers are $p = 30$. Table 4.2 presents the Type I errors and powers of testing hypothesis (4.24) for 2 sets of $\{a, b\}$ values at 4 different ρ values (one is estimated). In general, the power curve of RLRT does not depend on ρ very much. Liu et al. (2007) revealed the same phenomena for the score test of a single variance component within a model with only one random effect. This is because moderate differences of ρ do not change the structure of the covariance matrix very much, except for extreme values such as $\rho \to 0$ or $\rho \to \infty$, with which the covariance matrix turns to an identity matrix or a matrix of ones. Note that the empirical Type II errors of all situations are around 0.03, smaller than the nominal one. The reason could be the approximation of (4.28) due to the assumption, $\theta_1 = \lambda_x^{-1} \approx 0$. To test two zero variance components with extra genes, we consider simulations with the sample sizes $n = 60$ and $n = 35$. The latter mimics the Type II diabetes data where the total subjects under study are $n = 35$. Fitting with the equal true and used gene numbers is compared to fitting with an extra 20 irrelevant genes. The results in Table 4.3 show that, when fitting with extra genes, the power decreases as expected but not dramatically, which means that the model we proposed can be applied to

Table 4.5: Estimated parameters of top 20 pathways obtained from p-REML and ranked by p-values of testing RLRT D. The numbers in the round brackets are the standard errors.

pathway ID	gene#	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\sigma}^2$	$\hat{\tau}_x$	$\hat{\tau}_z$	$\hat{\tau}_{xz}$	fixed ρ	RLRT D	RLRT p-value
73	11	5.09(1.51)	-0.01(0.21)	0.08(0.39)	1.0e-11(0.02)	6.09(3.12)	17.7(11.8)	0.457	12.2	0.001
274	16	7.25(1.35)	0.20(0.16)	0.66(0.71)	2.1e-09(0.02)	4.74(3.09)	9.74(8.90)	0.581	7.68	0.006
230	121	5.69(1.39)	0.15(0.14)	0.10(1.03)	7.3e-11(0.02)	5.75(3.42)	6.17(6.99)	0.330	7.81	0.006
229	133	5.82(1.13)	0.15(0.12)	1.29(1.28)	1.7e-03(0.02)	3.25(2.99)	3.96(6.31)	0.289	6.65	0.012
152	11	6.13(1.12)	0.21(0.15)	2.16(0.91)	8.6e-09(0.02)	1.57(2.21)	7.48(8.69)	1.266	6.20	0.014
16	49	5.76(1.00)	0.14(0.13)	1.98(1.24)	1.5e-08(0.02)	1.89(2.55)	4.57(6.82)	0.308	5.93	0.017
173	11	6.06(1.07)	0.19(0.15)	2.14(0.92)	2.1e-09(0.01)	1.57(2.22)	7.10(7.93)	0.756	5.77	0.017
236	22	6.27(1.06)	0.23(0.15)	2.10(1.06)	1.4e-08(0.02)	1.41(2.24)	7.24(8.08)	0.862	5.63	0.019
144	7	5.43(1.21)	0.15(0.20)	2.35(0.85)	1.6e-03(0.02)	1.16(2.26)	11.5(11.7)	0.411	5.35	0.019
151	20	6.08(1.04)	0.22(0.14)	2.15(1.06)	7.5e-09(0.02)	1.52(2.24)	6.21(7.51)	0.937	5.62	0.019
14	49	6.09(1.20)	0.16(0.14)	1.57(1.27)	1.3e-09(0.02)	2.76(2.91)	5.72(7.42)	0.706	5.30	0.024
228	43	6.16(0.77)	0.20(0.14)	2.88(1.18)	7.4e-11(0.02)	0.03(1.73)	5.91(6.86)	0.374	4.95	0.028
103	37	6.09(0.90)	0.20(0.14)	2.58(1.20)	9.8e-09(0.02)	0.74(2.08)	5.76(7.42)	0.751	4.82	0.030
271	37	6.20(0.92)	0.22(0.14)	2.45(1.23)	7.5e-12(0.02)	0.94(2.19)	5.73(7.19)	0.702	4.83	0.030
150	21	5.98(0.94)	0.19(0.14)	2.54(1.12)	7.5e-11(0.02)	0.97(2.10)	5.75(7.65)	1.161	4.66	0.033
172	8	5.85(0.92)	0.15(0.18)	2.75(0.99)	2.6e-03(0.02)	3.5e-10(1.61)	10.1(9.8)	0.812	4.22	0.039
133	58	6.01(0.83)	0.18(0.14)	2.71(1.29)	1.8e-03(0.02)	0.32(2.04)	6.28(7.18)	0.339	4.15	0.044
8	27	5.87(0.78)	0.18(0.15)	2.92(1.15)	1.6e-02(0.04)	3.0e-09(1.72)	5.96(7.21)	0.527	4.08	0.045
101	13	6.08(0.90)	0.19(0.16)	3.01(1.01)	5.7e-10(0.02)	0.23(1.59)	6.81(8.79)	0.458	3.88	0.045
158	8	5.79(1.00)	0.15(0.14)	2.55(0.98)	1.3e-09(0.02)	1.55(2.24)	5.39(7.72)	0.621	3.53	0.056

pathway data for which only some of the genes are related to the responses. In addition, comparing Table 4.2 and 4.3 shows that the power does decrease with the sample size n.

The simulation study for testing P-E interaction using RLRT and the score test is carried out using a new setup for the data generation. We continue using the same nonparametric Expression (4.34) except with true gene number $p = 5$; that is, simply replacing $f_z(\cdot)$ and $f_{xz}(\cdot)$ as $f_z(\boldsymbol{z}_i^T) = a \cdot \boldsymbol{z}_i^{(5)} \exp\left(-0.2|\bar{\boldsymbol{z}}|_i^{(5)}\right)/5$ and $f_{xz}(x_i, \boldsymbol{z}_i^T) = b \cdot e^{x_i/10} \sin\left(\bar{\boldsymbol{z}}_i^{(5)}\right) \cos\left(\bar{\boldsymbol{z}}_i^{(5)}\right)/8$, where $\boldsymbol{z}_i^{(5)} = \sum_{j=1}^5 z_{ij}$, $|\bar{\boldsymbol{z}}|_i^{(5)} = \sum_{j=1}^5 |z_{ij}|/5$ and $\bar{\boldsymbol{z}}_i^{(5)} = \sum_{j=1}^5 z_{ij}/5$. x_i, z_{ij} and ϵ_i are generated the same way as before. Note the function form changes when the gene number is different in (4.34). We use this setup to compare two test procedures for testing (4.29). For the score test, we first estimate the parameters using p-REML and then calculate the statistics using expressions (4.17) and (4.33). The results are listed in Table 4.4. Again, we see that the test's power does not depend on ρ. The results indicate

that the RLRT are slightly lower in power and that the type I errors of the two test methods are all closer to the nominal 5% from different directions. These results indicate we can apply both test methods under suitable conditions.

4.6 Application to Type II Diabetes Data

We applied our mixed model (4.10) to a set of diabetes data from Mootha et al. (2003). They utilized the HGC-133a Affymetrix genechip with 22,283 genes to study 17 normal glucose tolerance individuals vs. 18 Type II diabetes mellitus patients. The 22,283 genes make up a total of 251 pathways. The goal of this study is to identify pathways with the highest significant overall pathway effect when an environmental variable, body mass index, is present in the model, and from them identify pathways with significant P-E interaction effect. Therefore, there are a total of 251 sets of data, each having $n = 35$ observations. Corresponding to each individual pathway, the data set contains $(\mathbf{y}, X, Z)$, where $\mathbf{y}$ is the outcomes of glucose level, X has the same meaning as before with the first column of 1's and the second column as the body mass index data of 35 subjects, and $Z(n \times p)$ is the gene expression levels of each pathway, which contains the number of genes ranging from $p = 3$ to $p = 543$.

The fitting results of the top 20 pathways are listed in Table 4.5 ranked ascendingly in the p-value of testing the overall pathway effect using RLRT D. It has almost an identical order of the magnitude as the D. It can be seen that 19 out of the 251 pathways are significant. For each pathway, the variance components are estimated using p-REML methods and the standard error of those parameters including $\hat{\sigma}^2$ are calculated using information matrix (4.17) with the p-REML estimates plugged in. Again, the initial values for the variance parameters are $(\sigma^2, \lambda_x^{-1}, \lambda_z^{-1}, \lambda_{xz}^{-1})^T = (0.001, 1, 1, 1)^T$ and ρ is fixed at the

average of $\|\mathbf{z} - \mathbf{z}'\|^2$ of different pairwise observations, which ranges from 0.1 to 1.8 for different pathways. To show an overall view of the fitting results for 251 pathways, Figure

Table 4.6: P-values of different tests for top 20 pathway significant in the overall pathway effect. Columns 2 and 3 are labels indicating appearance in the top 50 list of other methods or not. Missing values in column 6 is because the information matrix is not positive definite.

pathway ID	Global Score Test	Forest Tree	RLRT test for D	permutation test for D	RLRT test for d	permutation test for d	score test for $U_{\tau_{xz}}$
73	Yes	Yes	0.001	0.001	0.002	0.001	0.005
274	Yes	No	0.006	0.011	0.025	0.013	0.016
230	Yes	Yes	0.006	0.010	-	0.025	0.007
229	Yes	Yes	0.012	0.020	-	0.138	0.062
152	No	No	0.014	0.015	0.179	0.303	0.163
16	Yes	Yes	0.017	0.027	0.126	0.147	0.058
173	Yes	Yes	0.017	0.020	0.017	0.018	0.002
236	No	No	0.019	0.021	0.133	0.119	0.104
144	Yes	Yes	0.019	0.020	0.076	0.072	0.106
151	No	No	0.019	0.023	0.205	0.262	0.146
14	Yes	No	0.024	0.031	0.113	0.054	0.046
228	Yes	Yes	0.028	0.035	0.032	0.024	0.006
103	No	Yes	0.030	0.039	0.121	0.106	0.086
271	No	No	0.030	0.037	0.148	0.142	0.110
150	No	No	0.033	0.034	0.080	0.062	0.044
172	No	No	0.039	0.044	0.016	0.015	0.009
133	No	No	0.044	0.057	0.053	0.043	0.018
8	Yes	Yes	0.045	0.052	0.051	0.038	0.032
101	No	No	0.045	0.044	0.068	0.049	0.056
158	Yes	No	0.056	0.054	-	0.343	0.560

4.3 plots the four estimated variance components in the same order of the p-value of RLRT D. The straight dashed line divides the significant and insignificant pathways of RLRT. The error components, $\hat{\sigma}^2$'s, are around the constant 3.0 except for those top significant pathways. This is consistent with the test results indicating that for those pathways with

genes relevant to the responses, the error is reduced since part of the variation of the responses is explained by pathway main effect or P-E interaction effect. The variations of $\hat{\tau}_x$ and $\hat{\tau}_z$ seems to compensate for each other. For the top 50 pathways, $\hat{\tau}_x$'s are close to zero and $\hat{\tau}_z$ values are large. On the other side, for those pathways which are ranked as lower than 50, $\hat{\tau}_z$ values are very small and $\hat{\tau}_x$ values increase. This indicates that for those pathways not relevant enough to the response, part of the variation of response is explained by the nonlinear relationship of the responses and the environmental variable. The variation of $\hat{\tau}_{xz}$ seems less dramatic than other random effects. It does not decrease to zero for those non significant pathways, and stabilizes after the top 100 pathways. However, using the test of RLRT d, we show that the lower ranked pathways, ranked as [50, ..., 251], are not significant in the interaction effect. These results suggest that the body mass index is important in explaining the relationship between the glucose level and the genetic pathway since many pathways that are significant in the overall pathway effect are either significant in the interaction effect or not.

Because the distribution for D is asymptotic, the p-value calculated based on 35 observations may not be as accuracte as expected. Hence, we carried out a permutation test process to obtain the exact distribution of D as follows:

- *Step 1*: We fit the observed data with the full model (4.10) and reduced model under hypothesis (4.24) using the p-REML approach. In both models, we set $\tau_x = 0$ since we assume that τ_x is insignificant when deriving (4.28). Then we obtained test statistics D, and calculated the residual $\hat{\boldsymbol{\epsilon}}_0 = \hat{\mathbf{r}}_z + \hat{\mathbf{r}}_{xz} + \hat{\boldsymbol{\epsilon}}$ using the fitted results of the full model from $\mathbf{y} = X\boldsymbol{\beta} + \mathbf{r}_z + \mathbf{r}_{xz} + \boldsymbol{\epsilon}$.

- *Step 2*: We permuted the residual $\hat{\boldsymbol{\epsilon}}_0$ to get new $\hat{\boldsymbol{\epsilon}}_0^*$ and simulate outcomes as $\mathbf{y}^* = X\hat{\boldsymbol{\beta}} + \hat{\boldsymbol{\epsilon}}_0^*$.

- *Step 3*: Based on $\mathbf{y}^*$, X and Z, we fit the full model (4.10) and reduced model under hypothesis (4.23) again using the p-REML approach and then calculated the test statistics D^*.

- *Step 4*: We repeated Steps 2-3 for a large number of times (e.g. 10,000 times).

- *Step 5*: We obtained the empirical p-value of the RLRT by formula p-value = (number of D^*'s greater than D) $\div$ (total number of D^*'s).

The p-value of the permutation test of D as well as the RLRT D are listed in Table 4.6 in the same order of Table 4.5 for the top 20 pathways. Note that for RLRT if the sample size is too small such that the information matrix (4.22) is non positive definite, ϕ in (4.28) cannot be calculated, so we are not able to get the asymptotic distribution of D. However the information matrices of the 251 pathways under hypothesis (4.24) are all positive definite (not true under hypothesis (4.29)), so we are able to test the overall pathway effect for all using RLRT D. The results of both tests are similar to each other with respect to the general rank of the significance, specifically both tests have the same top 3 pathways, which are pathways 73, 274, and 230. In addition, most of the p-values of the permutation tests are slightly larger than those of RLRT, as expected, since the permutation test is usually more conservative. Table 4.6 also labels those significant pathways ranked in the top 50 list according to the global score test (Goeman et al., 2004) and the forest tree method Pang et al. (2006); Pang and Zhao (2008), which do not take into account the environmental variable in their models. Our approach identified pathways that have either significant main pathway effect, the interaction effect, or both, while other methods determined many as having a significant main pathway effect only. Through following one zero variance component test, we also discovered that some pathways have a significant P-E interaction effect although they may not have a significant main pathway effect.

Furthermore, the p-values of RLRT d are also listed in Table 4.6. There are pathways for

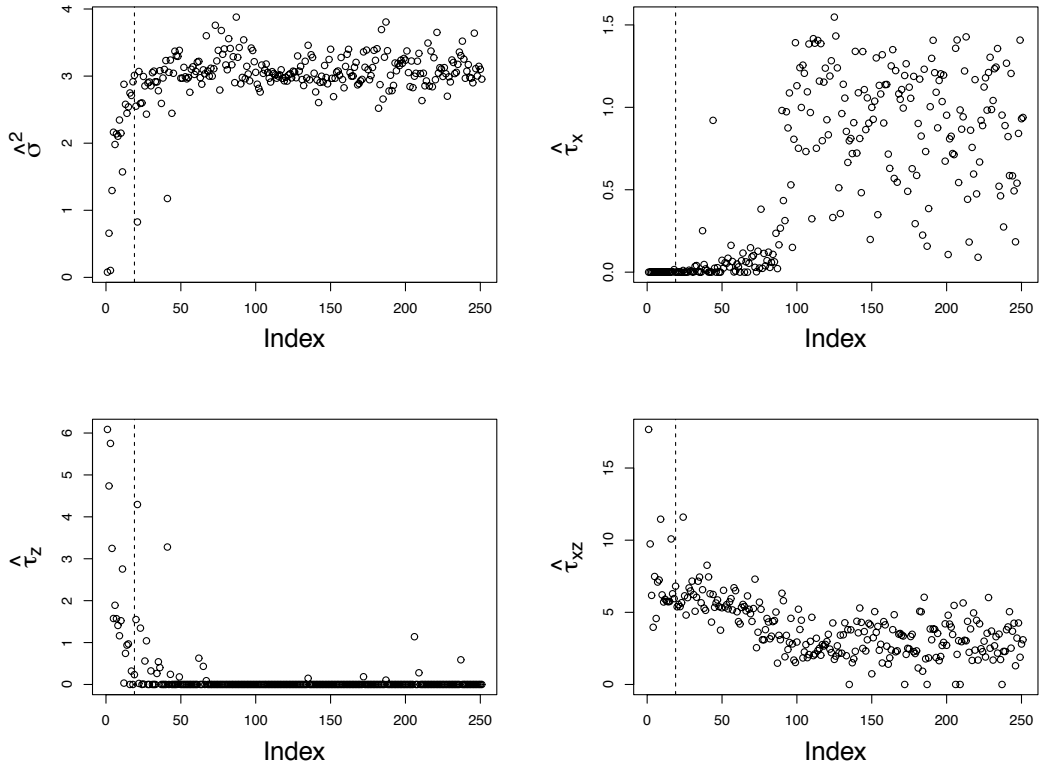

Figure 4.3: The estimated variance components of $\hat{\sigma}^2, \hat{\tau}_x, \hat{\tau}_z, \hat{\tau}_{xz}$ for 251 pathways ordered by p-values of testing the overall pathway effect. The dash lines separate the significant and insignificant pathways at 5% level.

which we are unable to calculate d because the information matrix is not positive definite. In Figure 4.4 the p-values of RLRT D and RLRT d of all pathways are plotted for comparison. Among the top 50 that are significant in overall pathway effect, only part of them are significant in the interaction effect, but for the remaining 151 pathways, none are significant in either interaction effect or overall pathway effect. Similar to RLRT D, a permutation test process for the exact distribution of RLRT d is introduced here:

- *Step 1*: We fit the observed data with the full model (4.10) and reduced model under

hypothesis (4.29) using the p-REML approach. Again in both models we assume that τ_x is negligible. Then we obtained d, and calculated the residual $\hat{\boldsymbol{\epsilon}}_0 = \hat{\mathbf{r}}_{xz} + \hat{\boldsymbol{\epsilon}}$ using the fitted results of the full model from $\mathbf{y} = X\boldsymbol{\beta} + \mathbf{r}_z + \mathbf{r}_{xz} + \boldsymbol{\epsilon}$.

- *Step 2*: We permuted the residual $\hat{\boldsymbol{\epsilon}}_0$ to get new $\hat{\boldsymbol{\epsilon}}_0^*$ and simulated outcomes as $\mathbf{y}^* = X\hat{\boldsymbol{\beta}} + \hat{\mathbf{r}}_z + \hat{\boldsymbol{\epsilon}}_0^*$.

- *Step 3*: Based on $\mathbf{y}^*$, X and Z, we fit the full model and reduced model under hypothesis (4.29) again using the p-REML approach and then calculated the test statistics d^*.

- *Step 4*: We repeated Steps 2-3 a large number of times (e.g. 10,000 times).

- *Step 5*: We obtained the empirical p-value of the RLRT by formula p-value = (number of d^*'s greater than d) $\div$ (total number of d^*'s).

The permutation test results of RLRT d are close to those of RLRT d in the 20 pathways, but it is difficult to tell which one is more conservative.

We also calculated the p-values of testing H_0 (4.29) using the score test approach for the top 20 pathways. Compared with the RLRT d and RLRT d permutation tests, the p-values of the score test is similar in sense of determining the significant pathways at the 5% level. Among these top 20 pathways with significant overall pathway effect, the pathways with insignificant interaction effect are $\{229, 152, 16, 236, 144, 151, 103, 271, 101, 158\}$ according to the score test, and $\{229, 152, 16, 236, 144, 151, 14, 103, 271, 150, 158\}$ according to the RLRT d permutation test. Note that the difference of the two sets, $\{14, 101, 150\}$, all have marginal p-values for the two tests at the 5% level. If they are removed from the two sets, both tests have identical pathways which have insignificant P-E environment interaction effects. Based on the three tests procedures, we identified the pathways

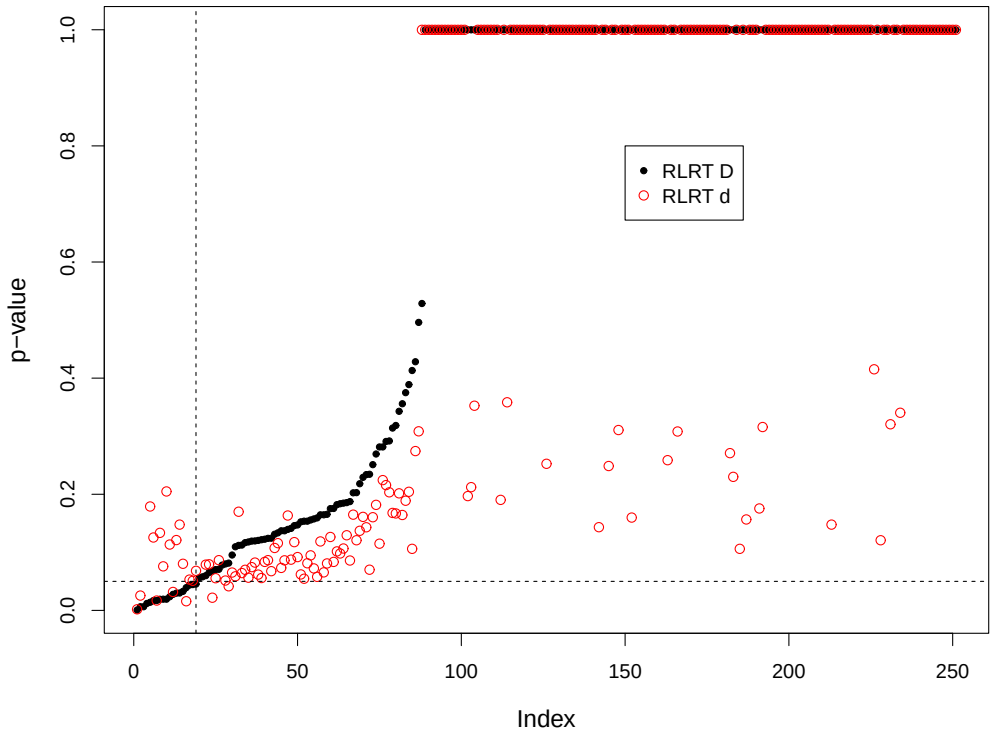

Figure 4.4: The p-values of testing overall pathway effect (RLRT D) and P-E interaction effect (RLRT d) for 251 pathways. The vertical dash line divides the significant and insignificant pathways of overall pathway effect test, and the horizontal dash line indicates 5% significant level. Some p-values of RLRT d are missing because the information matrix is not positive definite.

with a significant P-E environment interaction effect for all tests among the top 20 pathways. They are $\{73, 274, 230, 173, 228, 172\}$ pathways at the 5% level. These pathways are known to be related to Type II diabetes. Pathway 73 is a *Cysteine metabolism* pathway. It is known that taurine (a semi-essential sulphur amino acid) derived from cysteine metabolism can prevent diabetes mellitus and/or insulin resistance (Franconi et al., 2006). Pathway 274 is involved in the *Urea cycle and metabolism of amino groups*, which has also been reported to be related to Type II diabetes (Czyzyk et al., 1989). Pathway 230 is *OXPHOS_HG-U133A_probes* pathway. It has been reported that genes involved in

oxidative phosphorylation are coordinately upregulated with fasting hyperglycaemia in the livers of patients with Type II diabetes (Misu et al., 2007). The transcription levels of a class of genes involved in oxidative phosphorylation mechanisms are consistently lower in diabetics than in controls (Mootha et al., 2003; Misu et al., 2007). Pathway 173 is $MAP00531\_Glycosaminoglycan\_degradation$ pathway. It is known that Type II diabetes mellitus also induces an increased urinary excretion of total glycosaminoglycans (Juretić et al., 2002). Pathway 228 is involved in $Oxidative\,phosphorylation$. It is known to be related to diabetes (Misu et al., 2007; Mootha et al., 2003, 2004). This pathway is a process of cellular respiration in humans (or in general eukaryotes) and contains coregulated genes across different tissues and is related to insulin/glucose disposal. It is associated with ATP synthesis, a pathway involved in energy transfer. Pathway 172 is $MAP00530\_Aminosugars\_metabolism$ pathway. Aminosugars (= glucosamine) have no effect on fasting blood glucose levels, glucose metabolism, or insulin sensitivity at any oral dose level in healthy subjects, individuals with diabetes, or those with impaired glucose tolerance (Simon et al., 2011).

4.7 Discussion

The development of a pathway-based mixed model to relate the response with genetic pathways is motivated by the fact that genes always interact with the environmental variables. Modeling the P-E interaction effect can help in further understanding the biological mechanisms underlying diseases and facilitate the discovery of potential biomarkers. However, no existing approaches are able to jointly analyze pathways with the environmental variables when P-E interaction exists.

In this dissertation, we have addressed a mixed effects model connecting with kernel

machine methods and smoothing spline, so that we can analyze the genetic pathway data with a continuous clinical outcome when the P-E interaction effect is present in the model. We demonstrated the application of our method to a pathway data of Type II diabetes. Our approach allows us to evaluate the pathway effect and its interaction with the environmental variables by estimating the corresponding variance components and testing the significance of those parameters. Because of the high dimensional parameters space, there are usually some difficulties in solving the REML equations, such as non-positive error estimated. We reduced the parameter space dimension in solving REML equations by introducing the p-REML approach to estimate the variance components so that the error component is always in the parameter space. The p-REML approach not only allows us to solve the REML equations efficiently, but also provides an efficient choice in testing one or two zero variance components besides the global score test, i.e. the profile restricted likelihood ratio test for testing the overall pathway effect or P-E interaction.

Modeling the linear mixed model with a kernel machine has other advantages. It allows us to choose appropriate kernels to construct the variance matrix of the random effect as well as the interaction random effect in accordance with the data structure. In this dissertation, we focused on the Gaussian kernel, but when the sample size is large so that the computation becomes expensive, some less computational intensive alternatives to Gaussian kernel are available, such as rational quadratic kernel: $k(\boldsymbol{z}^T, \boldsymbol{z}'^T) = 1 - \|\boldsymbol{z} - \boldsymbol{z}'\|^2/(\|\boldsymbol{z} - \boldsymbol{z}'\|^2 + c)$. Other kernels, such as a polynomial kernel, an exponential kernel, an inverse multiquadric kernel, etc., have also been examined and can replace the Gaussian kernel in appropriate situations. Note that these kernels are similar to the Gaussian kernel in terms of reducing the dimension of the covariates through measuring the similarity of $\boldsymbol{z}$ and $\boldsymbol{z}'$. To some extent, this may be a disadvantage of the kernel method since there may be some information lost beyond the similarity of the two attributes.

Possible extensions of our method include applying the interaction kernel machine to generalized linear models. Logistic kernel machine regression with a Gaussian kernel has been developed by Liu et al (2008), but no interaction between the genetic pathway effect and environmental variable has been considered. By adding the interaction kernel machine to a generalized linear model, our method can be applied in more general genomewide association studies, especially in the case-control studies of G/P-E interaction. The second potential extension of our method is to consider a higher dimension of environmental variables $\boldsymbol{x}_i^T$, such as bivariate $\boldsymbol{x}_i^T = (x_{i1}, x_{i2})$, longitude and latitude data, and the nonparametric function $f_x(\boldsymbol{x}_i^T)$ can be fitted using thin plate splines (Gu and Wahba, 1993). With the kernel of the thin plate splines, we can construct the interaction function space kernel similarly. This extension may have wider applications such as in spatial data where the interaction between location and other high dimensional covariates are particularly interesting.

We note that we evaluate the interaction between each pathway and environmental variable. It is known that pathways are not independent of each other because of shared genes and interactions among pathways as well as their interaction with environmental variables, making it difficult to adjust the p-value due to the complex dependency structure. Because existing multiple comparison methods based on false discovery rates (Benjamini and Hochberg, 1995; Storey, 2002) were developed only for single gene based analysis that did not take into account the interaction between genes and environmental variables, they are not applicable in such a complicated situation as our problem. Developing a multiple comparison method will be an interesting and challenging problem because of the complex dependence structure among pathways and environmental variables.

Bibliography

Adami, H. O., Hunter, D., and Trichopoulos, D. (2008). *Textbook of Cancer Epidemiology*. New York: Oxford University Press.

Aronszajn, N. (1950). Theory of Reproducing Kernels, *Transactions of the American Mathmatical Society*, **68**, 337-404.

Bach, F. (2008). Consistency of the Group Lasso and Multiple Kernel Learning. *Journal of Machine Learning Research*, **9**, 1179-1225.

Barndorff-Nielsen, O., Kent, J., and Sørensen, M. (1982). Normal Variance-Mean Mixtures and z Distribution. *Internatiional Statistical Review*, **50**, 145-159.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the False Discovery Rate: a Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society, Series B*, **57**, 289-300.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. New York: Cambridge University Press.

Breiman, L. (1995). Better Subset Regression Using the Nonnegative Garrote. *Technometrics*, **37**, 373-384.

Breiman, L. and Friedman, J. H. (1985). Estimating Optimal Transformations for Multiple Regression and Correlation. *Journal of the American Statistical Association*, **80**, 580-598.

Breslow, N. E. and Clayton, D. G. (1993). Approximate Inference in Generalized Linear Mixed Models. *Journal of the American Statistical Association*, **88**, 9-25.

Carvalho, C. and Polson, N. G. (2010). The Horseshoe Estimator for Sparse Signals. *Biometrika*, **97**, 465-480.

Chakravarti, A. and Little, P. (2003). Nature, Nurture, and Human Disease. *Nature*, **421**, 412-414.

Chatterjee, A. and Lahiri, S. N. (2011). Bootstrapping Lasso Estimators. *Journal of the American Statistical Association*, **106**, 608-625.

Chatterjee, N., Kalaylioglu, Z., Moslehi, R., Peters, U., and Wacholder, S. (2006). Powerful Multilocus Tests of Genetic Association in the Presence of Gene-gene and Gene-environment Interactions. *The American Journal of Human Genetics*, **79**, 1002-1016.

Chib, S. and Greenberg, E. (2010). Additive Cubic Spline Regression with Dirichlet Process Mixture Errors. *Journal of Econometrics*, **156**, 322-336.

Claeskens, G. (2004). Restricted Likelihood Ratio Lack-of-fit Tests Using Mixed Spline Models. *Journal of the Royal Statistical Society, Series B*, **66**, 909-926.

Crainiceanu, C., Ruppert, D., Claeskens, G., and Wand, M. P. (2005). Exact Likelihood Ratio Tests for Penalized Splines. *Biometrika*, **92**, 91-103.

Czyzyk, A., Lao, B., Orowska, K., Szczepanik, Z., and Bartosiewicz, W. (1989). Effect of Antidiabetics on Post-exercise Alaninemia in Patients with Non-insulin-dependent Diabetes Mellitus (Type 2). *Polskie Archiwum Medycyny Wewntrznej*, **81**, 193-206.

Efron, B., Johnstone, I., Hastie, T., and Tibshirani, R. (2004). Least Angle Regression. *The Annals of Statistics*, **32**, 407-499.

Fan, J., Feng, Y., and Song, R. (2011). Nonparametric Independence Screening in Sparse Ultra-High-Dimensional Additive Models. *Journal of the American Statistical Association*, **106**, 544-557.

Fan, J. and Li, R. (2001). Varibale Selection via Nonconcave Penalized Likelihood and its Oracle Properities. *Journal of the American Statistical Association*, **96**, 1348-1360.

Fan, J. and Lv, J. (2008). Sure Independence Screening for Ultrahigh Dimensional Feature Sapce. *Journal of the Royal Statistical Society, Series B*, **70**, 849-911.

Franconi, F., Loizzo, A., Ghirlanda, G., and Seghieri, G. (2006). Taurine Supplementation and Diabetes Mellitus. *Current Opinion in Clinical Nutrition & Metabolic Care*, **9**, 32-36.

Gelman, A. (2006). Prior Distribution for Variance Parameters in Hierarchical Models. *Bayesian Analysis*, **1**, 515-533.

Geyer, C. J. and Thompson, E. A. (1995). Annealing Markov Chain Monte Carlo with Application to Ancestral Inference. *Journal of the American Statistical Association*, **90**, 909-920.

Goeman, J. J., Oosting, J., Cleton-Jansen, A., Anninga, J. K., and van Houwelingen, H. C. (2005). Testing Association of a Pathway with Survival Using Gene Expression data. *Bioinformatics*, **21**, 1950-1957.

Goeman, J. J., van de Geer, S. A., de Kort F., and van Houwelingen, H. C. (2004). A Global Test for Groups of Genes: Testing Association with a Clinical Outcome. *Bioinformatics*, **20**, 93-99.

Gordy, M. B. (1998). A generalization of Generalized Beta Distribution. In *Finance and Economics Discussion Series*. Board of Gobernors of the Federal Reserve System.

Green, P. J. (1987). Penalized Likelihood for General Semi-parametric Regression Models. *International Statistical Review*, **55**, 245-259.

Green, P. J. and Silverman, B. W. (1994). *Nonparametric Regression and Generalized Linear Models*. London: Chapman and Hall.

Gu, C. and Wahba, G. (1993). Semiparametric Analysis of Variance with Tensor Product Thin Plate. *Journal of the Royal Statistical Society, Series B*, **55**, 353-368.

Guo, W. (2002). Inference in Smoothing Spline Analysis of Variance. *Journal of the Royal Statistical Society, Series B*, **64**, 887-898.

Hahn, L. W., Ritchie, M. D., and Moore, J. H. (2003). Multifactor Dimensionality Reduction Software for Detecting Gene-gene and Gene-environment Interaction. *Bioinformatics*, **19**, 376-382.

Hall, P., Lee, E. R., and Park, B. U. (2009). Bootstrap-Based Penalty Choice for the Lasso, Achieving Oracle Performance. *Statistica Sinica*, **19**, 449-471.

Harville, D. A. (1977). Maximum Likelihood Approaches to Variance Component Estimation and to Related Problems. *Journal of the American Statistical Association*, **72**, 320-338.

Hastie, T. and Tibshirani, R. (1990). *Generalized Additive Models*. London; New York: Chapman and Hall.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer.

Heaton, M. and Scott, J. (2010). Bayesian Computation and the Linear Model. In *Frontiers of Statistical Decision Making and Bayesian Analysis*, eds. M. H. Chen, D. K. Dey, P. Mueller, D. Sun, and K. Ye. New York: Springer.

Higdon, D. M. (1998). Auxiliary Variable Methods for Markov Chain Monte Carlo with Applications. *Journal of the American Statistical Association*, **93**, 585-595.

Iba, Y. (2001). Extended Ensemble Monte Carlo. *International Journal of Modern Physics C*, **12**, 623-656.

Jennrich, R. J. and Schluchter, M. D. (1986). Unbalanced Repeated Measures Models with Structured Covariance Matrices. *Biometrics*, **42**, 805-820.

Juretić, D., Krajnović, V., and Lukac-Bajalo, J. (2002). Altered Distribution of Urinary Glycosaminoglycans in Diabetic Subjects. *Acta Diabetologica*, **39**, 123-8.

Kim, I., Pang, H., and Zhao, H. (2011). Semiparametric Methods for Evaluating Pathway Effects on Clinical Outcomes Using Gene Expression Data. *Technical Report*.

Kimeldorf, G. and Wahba, G. (1971). Some Results on Tchebycheffian Spline Functions. *Journal of Mathematical Analysis and Applications*, **33**, 82-95.

Knight, K. and Fu, W. (2000). Asymptotics for Lasso-Type Estimators. *The Annals of Statistics*, **28**, 1356-1378.

Krishnapuram, B., Hartemink, A. J., and Carin L. (2004). A Bayesian Approach to Joint Feature Selection and Classifier Design. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**, 1105-1111.

Kuo, L. and Mallick, B. (1998). Variable Selection for Regression Models. *Sankhyā: The Indian Journal of Statistics B*, **60**, 65-81.

Lancaster, P. and Šalkauskas, K. (1986). *Curve and Surface Fitting: an Introduction*. San Diego: Academic Press.

Lanckriet, G., Cristianini, N., Bartlett, P., Ghaoui, L. E., and Jordan, M. I. (2004). Learning the Kernel Matrix with Semi-Definite Programming. *Journal of Machine Learning Research*, **5**, 27-72.

Li, F. and Zhang, N. R. (2010). Bayesian Variable Selection in Structured High-Dimensional Covariates Spaces with Applications in Genomics. *Journal of the American Statistical Association*, **105**, 1202-1214.

Lin, X. (1997). Variance Component Testing in Generalized Linear Models with Random Effects. *Biometrika*, **84**, 309-326.

Lin, X. and Zhang, D. (1999). Inference in Generalized Additive Mixed Models by Using Smoothing Splines. *Journal of the Royal Statistical Society, Series B*, **61**, 381-400.

Lin, Y. and Zhang, H. H. (2006). Component Selection and Smoothing in Multivariate Nonparametric Regression. *The Annals of Statistics*, **34**, 2272-2297.

Liu, D., Ghosh, D., and Lin, X. (2008). Estimation and Testing for the Effect of a Genetic Pathway on a Disease Outcome Using Logistic Kernel Machine Regression via Logistic Mixed Models. *BMC Bioinformatics*, **9**, 292.

Liu, D., Lin, X., and Ghosh, D. (2007). Semiparametric Regression of Multi-Dimensional Genetic Pathway Data: Least Squares Kernel Machines and Linear Mixed Models. *Biometrics*, **63**, 1079-1088.

Lykou, A. and Ntzoufras, I. (2012). On Bayesian Lasso Variable Selection and the Spefification of the Shrinkage Parameter. *Statistics and Computing*, DOI: 10.1007/s11222-012-9316-x, 2012.

Lyubartsev, A. P., Martsinovski, A. A., Shevkunov, S. V., and Vorontsov-Velyaminov, P. N. (1992). New Approach to Monte Carlo Calculation of the Free Engery: Method of Expanded Ensembles. *Journal of Chemical Physics*, **96**, 1776-1783.

MacKay, D. J. C. (1994). Bayesian Methods for Backprop Networks. In Domany, E., van Hemmen, J. L. and Schulten, K., eidtors, *Models of Neural Networks, III*, Chapter **6**, 211-254. Springer.

MacKay, D. J. C. (1998). Introducing to Gaussian Process. In Bishop, C. M., editor, *Neural Networks and Machine Learning*. New York: Springer-Verlag.

Maity, A., Carroll, R. J., Mammen, E., and Chatterjee, N. (2009). Testing in Semiparametric Models with Interaction, with Applications to Gene-environment Interactions. *Journal of the Royal Statistical Society, Series B*, **71**, 75-96.

Manolio, T. A., Bailey-Wilson, J. E., and Collins, F. S. (2006). Genes, Environment and the Value of Prospective Cohort Studies. *Nature Review Genetics*, **7**, 812-820.

Meier, L., van de Geer, S., and Bühlmann P. (2009). High-dimenional additive modeling. *The Annals of Statistics*, **37**, 3779-3821.

Meinshausen, N. and Bühlmann P. (2006). High-Dimenional Graphs and Variable Selection with the Lasso. *The Annals of Statistics*, **34**, 1436-1462.

Micchelli, C. A. and Pontil, M. (2005). Learning the Kernle Fucntion via Regularization. *Journal of Machine Learning Research*, **6**, 1099-1125.

Misu, H., Takamura, T., Matsuzawa, N., Shimizu, A., Ota, T., Sakurai, M., Ando, H., Arai, K., Yamashita, T., Honda, M., Yamashita, T., and Kaneko, S. (2007). Genes Involved in Oxidative Phosphorylation are Coordinately Upregulated with Fasting Hyperglycaemia in Livers of Patients with Type 2 Diabetes. *Diabetologia*, **50**, 268-277.

Monni, S. and Li H. (2010). Bayesian Methods for Network-Structured Genomics Data. *UPenn Biostatistics Working Papers*, Working Paper 34.

Moore, J. H., Asselbergs, F. W., and Williams, S. M. (2010). Bioinformatics Challenges for Genome-wide Association Studies. *Bioinformatics*, **26**, 445-455.

Mootha, V. K., Handschin, C., Arlow, D., Xie, X., Pierre, J. S., Sihag, S., Yang, W., Altshuler, D., Puigserver, P., Patterson, N., Willy, P. J., Schulman, I. G., Heyman, R. A., Lander, E. S., and Spiegelman, B. M. (2004). Errα and Gabpa/b Specify PGC-1α-dependent Oxidative Phosphorylation Gene Expression that is Altered in Diabetic Muscle. *Proceedings of the National Academy of Sciences*, **101**, 6570-6575.

Mootha, V. K., Lindgren, C. M., Eriksson, K., Subramanian, A., Sihag, S., Lehar, J., Puigserver, P., Carlsson, E., Ridderstrale, M., Laurila, E., Houstis, N., Daly, M. J., Patterson, N., Mesirov, J. P., Golub, T. R., Tamayo, P., Spiegelman, B., Lander, E. S., Hirschhorn, J. N., Altshuler, D., and Groop, L. C. (2003). PGC-l alpha-Responsive Genes Involved in Oxidative Phosphorylation are Coordinately Downregulated in Human Diabetes. *Nature Genetics*, **34**, 267-273.

Neal, R. M. (1996) *Bayesian Learning for Neural Networks, Lecture Notes in Statistics No. 118*, New York: Springer-Verlag.

Newman, M. E. J. and Barkema, G. T. (1999). *Monta Carlo Methods in Statistical Physics*. New York: Oxford University Press.

Nott, D. J. and Green, P. J. (2004). Bayesian Vairable Selection and the Swenden-Wang Algorithm. *Journal of Computational and Graphical Statistics* , **13**, 141-157.

Pang, H., Kim, I., and Zhao, H. (2011). Random Effect Model for Multiple Pathway Analysis with Applications to Type II Diabetes Microarray Data. *Technical Report*.

Pang, H., Lin, A., Holford, M., Enerson, B., Lu, B., Lawton, M. P., Floyd, E., and Zhao, H. (2006). Pathway Analysis Using Random Forests Classification and Regression. *Bioinformatics*, **22**, 2028-2036.

Pang, H. and Zhao, H. (2008). Building Pathway Clusters form Random Forest Classification Using Class Votes. *BMC Bioinformatics*, **9**, 87.

Park, M. Y. and Hastie, T. (2008). Penalized Logistic Regression for Detecting Gene Interactions. *Biostatistics*, **9**, 30-50.

Pericchi, L. R. and Smith, A. (1992). Exact and Approximate Posterioe Moments for a Normal Location Parameter. *Journal of the Royal Statistical Society, Series B*, **54**, 793-804.

Polson, N. G. and Scott, J. G. (2010). On the Half-Cauchy Prior for a Global Scale Parameter. *Technical report*, University of Texas at Austin.

Polson, N. G. and Scott, J. G. (2011). Shrink Globally, Act Locally: Sparse Bayesian Regularization and Prediction. In *Proceedings of the 9th Valencia World Meeting on Bayesian Statistics*. Oxford University Press, 501-538.

Polson, N. G. and Scott, J. G. (2012). Local Shrinkage Rules, Lévy Processes and Rgularization Regression. *Journal of the Royal Statistical Society, Series B*, **74**, 287-311.

Radchenko, P. and James, G. M. (2010). Variable Selection Using Adaptive Nonlinear Interaction Strcutures in High Dimensions. *Journal of the American Statistical Association*, **105**, 1541-1553.

Rakotomamonjy, A., Bach, F., Canu, S., and Grandvalet, Y. (2008). SimpleMKL. *Journal of Machine Learning Research*, **9**, 2491-2521.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Process for Machine Learning*. Cambridge: MIT Press.

Ravikumar, P., Lafferty, J., Liu, H., and Wasserman, L. (2009). Sparse Additive Models. *Journal of the Royal Statistical Society, Series B*, **71**, 1009-1030.

Reich, B. J., Storlie, C. B., and Bondell, H. D. (2009). Variable Selection in Bayesian Smoothing Spline ANOVA Models: Application to Deterministic Computer Codes. *Journal of Econometrics*, **51**, 110-119.

Ritchie, M. D., Hahn, L. W., Roodi, N., Bailey, L. R., Dupont, W. D., Parl, F. F., and Moore, J. H. (2001). Multifactor-dimensionality Reduction Reveals High-order Interactions Among Estrogen-metabolism Genes in Sporadic Breast Cancer. *The American Journal of Human Genetics*, **69**, 138-147.

Scheipl, F. (2011). spikeSlabGAM: Bayesian Variable Selection, Model Choice and Regularization for Generalized Additive Mixed Models in R. *Journal of Statistical Software*, **43**, 1-24.

Searle, S. R., Casella, G., and McCulloch, C. E. (1992). *Variance Components*. New York: Wiley.

Simon, R., Marks, V., Leeds, A., and Anderson, J. (2011). A Comprehensive Review of Oral Glucosamine Use and Effects on Glucose Metabolism in Normal and Diabetic Individuals. *Diabetes Metabolism Research and Reviews*, **27**, 14-27.

Smith, M. and Kohn, R. (1996). Nonparametric Regression Using Bayesian Variable Selection. *Journal of Econometrics*, **75**, 317-343.

Stingo, F. C., Chen, Y. A., Tadesse, M. G., and Vannucci, M. (2011). Incorporating Biological Information into Linear Models: a Bayesian Approach to the Selection of Pathways and Genes. *The Annals of Applied Statistics*, **5**, 1978-2002.

Storey, J. D. (2002). A Direct Approach to False Discovery Rates. *Journal of the Royal Statistical Society, Series B*, **64**, 479-498.

Swendsen, R. H. and Wang, J. S. (1987). Nonuniversal critical dynamics in Monte Carlo simulations. *Physical Review Letters*, **58**, 86-88.

Tai, F., Pan, W., and Shen, X. (2010). Bayesian Variable Selection in Regression with Networked Predictors. In *High-Dimensional Data Analysis*, eds. T. Cai and X. Shen. Singapore: World Scientific, 147-165.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, **58**, 267-288.

Tipping, M. E. (2001). Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, **1**, 211-244.

Vu, H. T. V. and Zhou, S. (1997). Generalization of Likelihood Ration Tests under Nonstandard Conditions. *The Annals of Statistics*, **25**, 897-916.

Wahba, G. (1990). *Spline Models for Observational Data*. Philadelphia: Society for Industrial and Applied Mathematics.

Wainwight, M. (2009). Sharp Thresholds for High-Dimensional and Noisy Sparsity Recovrery Using l_1-Constrained Quadratic Programming (Lasso). *IEEE Transctions on Information Theory*, **55**, 2183-2202.

Wang, H. and Leng, C. (2007). Unified LASSO Estimation by Least Squares Approximation. *Journal of the American Statistical Association*, **102**, 1039-1048.

Wang, K., Li M., and Bucan, M. (2007). Pathway-based Approaches for Analysis of Genomewide Association Studies. *The American Journal of Human Genetics*, **81**, 1278-1283.

West, M. (1987). On Scale Mixtures of Normal Distributions. *Biometrika*, **74**, 646-648.

Wolff, U. (1989). Collective Monte Carlo Updating for Spin Systems. *Phisical Review Letters*, **62**, 361-364.

Zhang, D. and Lin, X. (2003). Hypothesis Testing in Semiparametric Additive Mixed Models. *Biostatistics*, **4,** 57-74.

Zhang, D., Lin, X., Raz, J., and Sowers, M. (1998). Semiparametric Stochastic Mixed Models for Longitudinal data. *Journal of the American Statistical Association*, **93**, 710-719.

Yuan, M. (2007). Nonnegative Garrote Component Selection in Functional ANOVA Models. *Proceedings of AI and Statistics, AISTATS*, 660-666.

Yuan, M. and Lin, Y. (2007). On the Nonnegative Garrote Estimator. *Journal of the Royal Statistical Society, Series B*, **69**, 143-161.

Zhao, P. and Yu, B. (2006). On Model Selection Consistency of Lasso. *Journal of Machine Learning Research*, **7**, 2541-2563.

Zou, F., Huang, H., Lee, S., and Hoeschele, I. (2010). Nonparametric Bayesian Variable Selection with Applications to Multiple Quantitative Trait Loci Mapping with Epistasis and Gene-Environment Interaction. *Genetics*, **186**, 385-394.

Zou, H. (2006). The Adaptive Lasso and Its Oracle Proptertis. *Journal of the American Statistical Association*, **101**, 1418-1429.

Appendix A

Lancaster and Šalkauskas Basis for Natural Cubic Spline

In this dissertation, we follow Chib and Greenberg (2010) to employ the cubic spline LS basis described by Lancaster and Šalkauskas (1986). Consider the jth function $f_j(x)$, and let $\boldsymbol{\nu}_j = (\nu_{1j}, ..., \nu_{K_j j})$ be the set of $100 \times \frac{k-1}{K_j - 1}\%, k = 1, ..., K_j$ quantile of $x_{ij}, i = 1, ..., n$. Thus $\nu_{1j} = \min_i(x_{ij})$ and $\nu_{K_j j} = \max_i(x_{ij})$. K_j denotes the number of knots for the spline functions. Then the cubic spline expansion of $f_j(x)$ is expressed as

$$\begin{aligned} f_j(x_{ij}) &= \sum_{k=1}^{K_j} [\Phi_{kj}(x_{ij}) g_{kj} + \Psi_{kj}(x_{ij}) s_{kj}] \\ &= \Phi_j(x_{ij})^T \mathbf{g}_j + \Psi_j(x_{ij})^T \mathbf{s}_j, \end{aligned} \tag{A.1}$$

where $\mathbf{g}_j = (g_{1j}, ..., g_{K_j j})^T$ and $\mathbf{s}_j = (s_{1j}, ..., s_{K_j j})^T$ are the coefficients of this expression, $\Phi_j(x_{ij}) = [\Phi_{1j}(x_{ij}), ...\Phi_{K_j j}(x_{ij})]^T$ and $\Psi_j(x_{ij}) = [\Psi_{1j}(x_{ij}), ...\Psi_{K_j j}(x_{ij})]^T$ are two basis vec-

tors, and the basis functions $\{\Phi_{kj}(x)\}_{k=1}^{K_j}$ and $\{\Psi_{kj}(x)\}_{k=1}^{K_j}$ are defined as

$$\Phi_{kj}(x) \propto \begin{cases} 0, & x < \nu_{k-1,j} \\ -(2/h_{kj}^3)(x-\nu_{k-1,j})^2(x-\nu_{kj}-0.5h_{kj}), & \nu_{k-1,j} \le x < \nu_{kj} \\ (2/h_{k+1,j}^3)(x-\nu_{k+1,j})^2(x-\nu_{kj}+0.5h_{k+1,j}), & \nu_{kj} \le x < \nu_{k+1,j} \\ 0 & x \ge \nu_{k+1,j}, \end{cases}$$

$$\Psi_{kj}(x) \propto \begin{cases} 0, & x < \nu_{k-1,j} \\ (1/h_{kj}^2)(x-\nu_{k-1,j})^2(x-\nu_{kj}), & \nu_{k-1,j} \le x < \nu_{kj} \\ (1/h_{k+1,j}^2)(x-\nu_{k+1,j})^2(x-\nu_{kj}), & \nu_{kj} \le x < \nu_{k+1,j} \\ 0 & x \ge \nu_{k+1,j}, \end{cases} \tag{A.2}$$

where $h_{kj} = \nu_{kj} - \nu_{k-1,j}$. Note that Φ_{1j}, Ψ_{1j} and Φ_{K_jj}, Ψ_{K_jj} are defined by last two lines and first two lines of above expressions respectively. $\mathbf{g}_j$ and $\mathbf{s}_j$ are interpreted the ordinate and slope of $f_j(x)$. Since $f_j(x)$ is a natural cubic splines with the second derivative equal to zero at two end points, and continuous derivative at knot points, both $\mathbf{g}_j$ and $\mathbf{s}_j$ are constrained (Lancaster and Šalkauskas, 1986) by $\mathbf{s}_j = A_j^{-1}C_j\mathbf{g}_j$, where

$$A_j = \begin{pmatrix} 2 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \omega_{2j} & 2 & \mu_{2j} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \omega_{2j} & 2 & \mu_{2j} & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \cdots & \ddots & \ddots & \ddots & \cdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & \omega_{K_j-1,j} & 2 & \mu_{K_j-1,j} \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 2 \end{pmatrix},$$

and

$$C_j = \begin{pmatrix} -\frac{1}{h_{2j}} & \frac{1}{h_{2j}} & 0 & 0 & \cdots & 0 & 0 & 0 \\ -\frac{\omega_{2j}}{h_{2j}} & \frac{\omega_{2j}}{h_{2j}} - \frac{\mu_{2j}}{h_{3j}} & \frac{\mu_{2j}}{h_{3j}} & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\frac{\omega_{3j}}{h_{3j}} & \frac{\omega_{3j}}{h_{3j}} - \frac{\mu_{3j}}{h_{4j}} & \frac{\mu_{3j}}{h_{4j}} & \cdots & 0 & 0 & 0 \\ \vdots & \cdots & \ddots & \ddots & \cdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\frac{\omega_{K_j-1,j}}{h_{K_j-1}} & \frac{\omega_{K_j-1,j}}{h_{K_j-1}} - \frac{\mu_{K_j-1,j}}{h_{K_j}} & \frac{\mu_{K_j-1,j}}{h_{K_j}} \\ 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{1}{h_{K_jj}} & \frac{1}{h_{K_ij}} \end{pmatrix},$$

where $\omega_{kj} = h_{kj}/(h_{kj} + h_{k+1,j})$ and $\mu_{kj} = 1 - \omega_{kj}$ for $k = 2, ..., K_j$. With this constraints, $\mathbf{s}_j$ can be replaced from the function Expression (A.1),

$$\begin{aligned} f_j(x_{ij}) &= [\Phi_j(x_{ij})^T + \Psi_j(x_{ij})^T A_j^{-1} C_j]\mathbf{g}_j \\ &= \mathbf{t}_j^T(x_{ij})\mathbf{g}_j, \end{aligned} \tag{A.3}$$

where $\mathbf{t}_j^T(x_{ij}) = (t_{1j}(x_{ij}), ..., t_{K_jj}(x_{ij})) = \Phi_j(x_{ij})^T + \Psi_j(x_{ij})^T A_j^{-1} C_j$. Furthermore, consider the identifying constraints, $\sum_k g_{kj} = 0$, we can express $g_{1j} = -(g_{2j} + \cdots + g_{K_jj})$, thus

$$f_j(x_{ij}) = \mathbf{t}_j^T(x_{ij})\mathbf{g}_j = [t_{2j}(x_{ij}) - t_{1j}(x_{ij})]g_{2j} + \cdots + [t_{K_jj}(x_{ij}) - t_{1j}(x_{ij})]g_{K_jj} = \mathbf{z}_j^{*T}(x_{ij})\boldsymbol{\beta}_j,$$

where $\boldsymbol{\beta}_j = (g_{2j}, ..., g_{K_jj})^T$ and we define matrix

$$Z_j^* = \begin{pmatrix} \mathbf{z}_j^{*T}(x_{1j}) \\ \vdots \\ \mathbf{z}_j^{*T}(x_{nj}) \end{pmatrix}.$$

Now the jth nonparametric function expressed by the natural cubic spline basis is $f_j(\mathbf{x}_j) = Z_j^* \boldsymbol{\beta}_j$. In order to incorporate the assumption of a priori smoothness, Chib and Greenberg (2010) consider a prior distribution on $\boldsymbol{\beta}_j$'s as,

$$[\boldsymbol{\beta}_j | \sigma_{ej}^2, \sigma_{dj}^2] \sim N\left[\mathbf{0}, \Delta_j^{-1} T_j (\Delta_j^{-1})^T\right], \tag{A.4}$$

where N is the $K_j - 1$ dimensional multivariate normal distribution, and

$$T_j = \begin{pmatrix} \sigma_{ej}^2 & 0 & 0 \\ 0 & \sigma_{dj}^2 I_{K_j - 3} & 0 \\ 0 & 0 & \sigma_{ej}^2 \end{pmatrix},$$

where two variance components σ_{ej}^2 and σ_{dj}^2 are selected here because of the different normal assumptions for the differences of the ordinates and the differences of slopes. Δ_j is given by

$$\Delta_j = \begin{pmatrix} \frac{2}{h_{2j}} & \frac{1}{h_{2j}} & \frac{1}{h_{2j}} & \frac{1}{h_{2j}} & \frac{1}{h_{2j}} & \cdots & \frac{1}{h_{2j}} & \frac{1}{h_{2j}} \\ \frac{1}{h_{2j}} & -\left(\frac{1}{h_{2j}} + \frac{1}{h_{3j}}\right) & \frac{1}{h_{3j}} & 0 & 0 & \cdots & 0 & 0 \\ 0 & \frac{1}{h_{3j}} & -\left(\frac{1}{h_{3j}} + \frac{1}{h_{4j}}\right) & \frac{1}{h_{4j}} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & -\frac{1}{h_{K_j j}} & \frac{1}{h_{M_j j}} \end{pmatrix}.$$

So far the construction of function $f_j(\mathbf{x}_j)$ is exactly the same as Chib and Greenberg (2010). Note that $f_j(\mathbf{x}_j) = Z_j^* \boldsymbol{\beta}_j$ with the prior of $\boldsymbol{\beta}_j$ given by (A.4) is equivalent to have $f_j(\mathbf{x}_j) = Z_j^* \Delta_j^{-1} \boldsymbol{\beta}_j$ with $[\boldsymbol{\beta}_j] \sim N(\mathbf{0}, T_j)$. Henceforth, we define the final $n \times M_j$ basis matrix $Z_j = Z_j^* \Delta_j^{-1}$ such that $f_j(\mathbf{x}_j) = Z_j \boldsymbol{\beta}_j$, where $M_j = K_j - 1$. Define $\tau_{ej} = \sigma_{ej}^{-2}$ and

$\tau_{bj} = \sigma_{bj}^{-2}$, and modify the one variance component prior algorithm in Section 2.4.2, then we can easily employ the LS basis into BSAM.

Appendix B

The Representation of the Natural Cubic Spline

Following Green and Silverman (1994), the representation of the natural cubic spline (4.8) in Section 4.2.2 is called the value-second derivative representation. Details for defining matrices B and M are shown as the following.

Suppose f_x is the natural cubic spline with n distinct $x_1^0 <, ..., < x_n^0$. Define

$$f_{x,i} = f_x(x_i^0) \text{ and } \gamma_i = f_x''(x_i^0) \text{ for } i = 1, ..., n.$$

By the definition of the natural cubic spline, $\gamma_1 = \gamma_n = 0$. Let $\mathbf{f}_x$ stands for the vector $(f_{x,1}, ..., f_{x,n})^T$ and let $\boldsymbol{\gamma} = (\gamma_2, ..., \gamma_{n-1})^T$ where $\boldsymbol{\gamma}$ is a $(n-2) \times 1$ vector with the element index starting at $i = 2$. Now define two matrices, Q and R. Let $h_i = x_{i+1}^0 - x_i^0$ for $i = 1, ..., n-1$. Let Q be the $n \times (n-2)$ matrix with entries q_{ij}, for $i = 1, ..., n-1$ and

$j = 2, ..., n-1$, given by

$$q_{j-1,j} = h_{j-1}^{-1},\ q_{jj} = -h_{j-1}^{-1} - h_j^{-1},\ \text{and}\ q_{j+1,j} = h_j^{-1}, \tag{B.1}$$

for $j = 2, ..., n-1$ and $q_{ij} = 0$ for $|i-j| \geq 2$. The columns of Q are indexed in the same way as the elements of $\boldsymbol{\gamma}$ starting at $j = 2$, so that the first element of Q is q_{12}.

R is a $(n-2) \times (n-2)$ symmetric matrix with elements r_{ij}, for i and j running from 2 to $n-1$, given by

$$\begin{aligned} r_{ii} &= \frac{1}{3}(h_{i-1} + h_i) \text{ for } i = 2, ..., n-1, \\ r_{i,i+1} = r_{i+1,i} &= \frac{1}{6}h_i \text{ for } i = 2, ..., n-2, \end{aligned} \tag{B.2}$$

and $r_{ij} = 0$ for $|i-j| \geq 2$.

The matrix R is strictly diagonal dominant and strictly positive definite. Using the Cholesky factorization that avoids taking the square roots in Chapter 2.6.1 of Green and Silverman (1994), we can factorize R as

$$R = U\Lambda U^T,$$

where Λ is a diagonal matrix and U is a lower triangular band matrix with diagonal elements all equal to 1. Since R are strictly positive definite, all diagonal elements of Λ are positive, $R^{-1} = (\Lambda^{1/2}U^T)^{-1}(U\Lambda^{1/2})^{-1}$. The penalty matrix M can be expressed as

$$M = QR^{-1}Q^T = Q(\Lambda^{1/2}U^T)^{-1}(U\Lambda^{1/2})^{-1}Q^T = LL^T, \tag{B.3}$$

where $L = Q(\Lambda^{1/2}U^T)^{-1}$. The B matrix thus is calculated by

$$\begin{aligned} B = L(L^TL)^{-1} &= Q(\Lambda^{1/2}U^T)^{-1}\left\{[(\Lambda^{1/2}U^T)^{-1}]^TQ^TQ(\Lambda^{1/2}U^T)^{-1}\right\}^{-1} \\ &= Q(\Lambda^{1/2}U^T)^{-1}(\Lambda^{1/2}U^T)(Q^TQ)^{-1}(\Lambda^{1/2}U^T)^T \\ &= Q(Q^TQ)^{-1}U\Lambda^{1/2}. \end{aligned}$$

Theorem 2.1 in Green and Silverman (1994) states that the vectors $\mathbf{f}_x$ and $\boldsymbol{\gamma}$ specify a natural cubic spline f_x if and only if the condition $Q^T\mathbf{f}_x = R\boldsymbol{\gamma}$ is satisfied. If this condition is satisfied then the roughness penalty will satisfy

$$\begin{aligned} \int_0^1 \{f_x''(x)\}^2dx &= \sum_{j=1}^{n-1}\frac{\gamma_{j+1}-\gamma_j}{h_j}(f_{x,j}-f_{x,j+1}) = \boldsymbol{\gamma}^TQ^T\mathbf{f}_x \\ &= \boldsymbol{\gamma}^TR\boldsymbol{\gamma} = \mathbf{f}_x^TQR^{-1}Q^T\mathbf{f}_x = \mathbf{f}_xM\mathbf{f}_x. \end{aligned}$$

In the above derivation we assume that $x_i^0, i = 1, ..., n$, were distinct and ordered, so the rank of the penalty matrix M is $n-2$ and B is a $n \times (n-2)$ matrix. In our model, we shall have r distinct and ordered $x_i^0, i = 1, ..., r$, from the observed data $x_i, i = 1, ..., n$, where $r \leq n$ and x_i's may not be ordered. Based the r x_i^0's, B is a $r \times (r-2)$ matrix. Thus we will use a $n \times r$ incidence matrix N defined in a way similar to that given by Green and Silverman (1994), Chapter 4.3.1, such that $B = NB$, where the left B is what we shall use in the model, and the right B is calculated based on r distinct x_i^0's.