Semiparametric and Nonparametric Methods for Complex Data

Byung-Jun Kim

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, Chair
Xinwei Deng
Pang Du
George Terrell

June 1, 2020
Blacksburg, Virginia

Keywords: Bayesian Fused Lasso, High-Dimensional Regression, Kernel Machine Learning-Based Regression, Matched Case-Control Study, Semiparametric Regression

Semiparametric and Nonparametric Methods for Complex Data

Byung-Jun Kim

(ABSTRACT)

A variety of complex data has broadened in many research fields such as epidemiology, genomics, and analytical chemistry with the development of science, technologies, and design scheme over the past few decades. For example, in epidemiology, the matched case-crossover study design is used to investigate the relationship between the clustered binary outcomes of disease and a measurement error in covariate within a certain period by stratifying subjects' conditions. In genomics, high-correlated and high-dimensional(HCHD) data are required to identify important genes and their interaction effect over diseases. In analytical chemistry, multiple time series data are generated to recognize the complex patterns among multiple classes. Due to the great diversity, we encounter three problems in analyzing those complex data in this dissertation. We then provided several contributions to semiparametric and nonparametric methods for dealing with the following problems: the first is to propose a method for identifying the significance of a functional relationship under the matched study; the second is to develop a method to simultaneously identify important variables and build a network in HDHC data; the third is to propose a multi-class dynamic model for recognizing a pattern in time-trend analysis.

For the first topic, we propose a semiparametric omnibus test for identifying the significance of a functional relationship between the clustered binary outcomes and covariates with measurement error by taking into account the effect modification of matching covariates. We develop a flexible omnibus test for testing purposes without a specific alternative form of a hypothesis. The advantages of our omnibus test are demonstrated through simulation studies and 1-4 bidirectional matched data analyses from an epidemiology study.

For the second topic, we propose a joint semiparametric kernel machine network approach to provide a connection between variable selection and network estimation. Our approach is a unified and integrated method which can simultaneously identify important variables and build a network among them. We develop our approach under a semiparametric kernel machine regression framework, which can allow for the possibility that each variable might be nonlinear and is likely to interact with each other in a complicated way. We demonstrate our approach using simulation studies and real application on genetic pathway analysis.

Lastly, for the third project, we propose a Bayesian focal-area detection method for multi-class dynamic model under a Bayesian hierarchical framework. Two-step Bayesian sequential procedures are developed to estimate patterns and detect focal intervals, which can be used for gas chromatography. We show the performance of our proposed approach using a simulation study and real application on gas chromatography on Fast Odor Chromatographic Sniffer (FOX) system.

Semiparametric and Nonparametric Methods for Complex Data

Byung-Jun Kim

(GENERAL AUDIENCE ABSTRACT)

A variety of complex data has broadened in many research fields such as epidemiology, genomics, and analytical chemistry with the development of science, technologies, and design scheme over the past few decades. For example, in epidemiology, researchers conduct an experimental design to investigate the relationship between the clustered binary outcomes of disease, e.g., control and case are denoted as 0 and 1, and a measurement error in a covariate within a certain period by stratifying subjects' conditions. Here, the experimental design is called a matched case-crossover design. In genomics, high-correlated and high-dimensional (HCHD) data required to identify essential genes and their interaction effect over diseases. "High-dimensional and High-correlated" defines the number of predictor variables are much larger than the sample size and the predictors are correlated with each other. A typical example is numerous genes in a certain organism where another gene or several other genes could change one gene's effect. In analytical chemistry, multiple time series data are generated to recognize the complex patterns among multiple classes. Due to the great diversity, we encounter three problems in analyzing the following three types of data: (1) matched case-crossover data, (2) HCHD data, and (3) Time-series data. We contribute to the development of statistical methods to deal with such complex data.

The goal of this dissertation is to develop the following three statistical methodologies to deal with the aforementioned complex data: (1) statistical testing for matched case-crossover design, (2) selection of important variables and estimation of their network (i.e. interaction) for HCHD data, and (3) functional estimation for multilevel dynamic trend data. The proposed methods enable applied researchers to make flexible inferences and comprehensive interpretation in solving different challenging problems of each data. The proposed statistical methods have a broad range of applications to answer scientific questions. We demonstrate advantages of our approaches using simulation studies and real applications on several data analyses from epidemiology, genomics, and analytical chemistry.

To my mother, for her unconditional love and everlasting support.

Acknowledgments

I want to express my deepest gratitude to my advisor, Dr. Inyoung Kim, for her guidance and encouragement throughout the period of my dissertation research. Thanks to her assistance and patience, I could endure my Ph.D. life of hardship and toil and finish the dissertation. She is one of my role models in my academic life. She has always devoted herself to offering great courses and taken great care of her students.

I am thankful to my committee members, Dr. Xinwei Deng, Dr. Pang Du, and Dr. George Terrell, for their time and precious feedback. I greatly appreciate Dr. Scottland Leman, who was instrumental in improving my writing skills and settling my teaching philosophy. I express my warmest gratitude to Dr. Van Mullekom, for offering me to work with SAIG and have experience of collaboration with non-statisticians. I would also like to thank you to the faculty and staff from the Department of Statistics for all the support.

I would like to thank all my friends and colleagues in and outside the department. They gave me a haven from the grind of my dissertation work and relieved my research anxieties.

Last but not least, my deep and sincere gratitude to my family for their everlasting and unconditional love, encouragement, and support. A special thank you to my grandparents, aunt, and uncle, for their spiritual support and endless patience throughout my educational life. I owe my mother, Kyung-Ae Kim, the biggest debt of gratitude for infinitely soothing and selflessly encouraging me to seek my happiness. None of this would have been possible without the love and support of you all.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

1.1.1 Semiparametric Prospective and Retrospective Model in 1-M Matched Case Crossover Studies

A matched case-crossover study is a blend of the matched case-control study and the crossover design. The typical case-crossover design uses a sample from a study based on a population of individuals that have all experienced the outcome of interest. When the measurements are taken on each subject in exposed and unexposed settings, each subject acts as his/her own control. Hence, this study can be viewed as a stratified data analysis of retrospective, self-matched follow-up studies, each with a sample size of one. The stratifying variable is the individual patient or subject.

Let us consider 1-M matched case-crossover studies. Let V be a matching covariate and k be a stratum level, $k = 1, \ldots, K$. We let $(Y_{1k}, Y_{2k}, \ldots, Y_{M+1k})$ be clustered binary outcomes of the the kth stratum for one case-M control status, $(\mathbf{Z}_{1k}, \ldots, \mathbf{Z}_{M+1k})$ be covariates with $q \times 1$ vector $\mathbf{Z}_{jk}$, and $(X_{1k}, X_{2k}, ..., X_{M+1k})$ be a covariate with a scalar X_{jk}, $j = 1, \ldots, M+1$.

The matched case-crossover studies are based on the classical prospective logistic regression and model

$$Pr(Y_{jk} = 1|\mathbf{Z}_{jk}, X_{jk}, k, V) = H\{\mathbf{Z}_{jk}^T\boldsymbol{\beta}_0 + \beta_1 X_{jk} + q(k, V)\}, \tag{1.1}$$

where $H(\bullet)$ is the logistic distribution function and $q(\bullet)$ is an arbitrary effect of the kth stratum including the intercept and unknown effects of the strata. The classical matched study begins with the model (1.1), but by conditioning on the fixed number of cases and controls in the stratum, any stratum effect is removed, i.e., $q(\bullet)$ disappears [21]. The retrospective logistic regression model for the kth stratum is then expressed as

$$Pr(Y_{1k} = 1|\mathbf{Z}_k, \mathbf{X}_k, \sum_{j=2}^{M+1} Y_{jk} = 1) = \frac{1}{1 + \sum_{j=2}^{M+1} \exp\{(\mathbf{Z}_{jk} - \mathbf{Z}_{1k})^T\boldsymbol{\beta}_0 + (X_{jk} - X_{j1})\beta_1\}} = l_k(\boldsymbol{\beta}_0, \beta_1), \tag{1.2}$$

where $\boldsymbol{Z}_k = (\mathbf{Z}_{1k}, \mathbf{Z}_{2k}, \ldots, \mathbf{Z}_{M+1k})$ and $\boldsymbol{X}_k = (X_{1k}, X_{2k}, \ldots, X_{M+1k})$. Since $q(\bullet)$ disappears, the unknown effects on V cannot be detected from the model (1.2). In practice, some variables may have measurement errors that also share each stratum. We often have distributional assumptions of a measurement error. To avoid this strong assumption, we consider the semiparametric framework, which can evaluate the matching covariate V as well as measurement error in covariate without the distributional assumption of the measurement error. We will discuss the semiparametric framework with measurement error in Section 2.2 of Chapter 2.

1.1.2 Gaussian Graphical Model

A probabilistic graphical model is one of popular probabilistic models which visualizes the conditional dependence structure between random variables. In the probabilistic graphical model, a graph (or network) consists of nodes and edges, where a random variable refers to a node, and an edge of the graph defines one of the connections between the nodes of the

graph. For a type of edges, they typically can be undirected, directed, and bidirected. The type of edge is determined by whether there is theoretical evidence supporting cause-and-effect among the nodes. Unless the evidence exists, undirected edges are employed in the representation of the graph.

A Gaussian graphical model defines a graphical model under the assumption that the random variables follow multivariate Gaussian distribution. A Gaussian graphical model has a property of Gaussian Markov Random Field (GMRF). That is, given an undirected graph with nodes and edges, it can determine the conditional independence or dependence between any two non-adjacent nodes (variables) given all other variables. Based on the pairwise GMRF, the main goal is to investigate the conditional dependence structure between variables by estimating sparse precision matrices. The sparsity of a precision matrix depends on the existence of edges; if the ith and jth variables are conditionally independent, the (i, j)th entry of the precision matrix is 0. Otherwise, the entry is non-zero. If some of the off-diagonal entries in a precision matrix have zeros, it is a sparse precision matrix.

Many methods of Gaussian graphical models have been developed in order to estimate precision matrix and impose the sparsity on the estimation. Friedman et al. [17] developed the graphical lasso (Glasso) algorithm based on a clockwise coordinate descent approach. Guo et al. [19] proposed jointly estimate precision matrices for different classes by re-parameterizing off-diagonal elements of precision matrices to be a multiplication of a common factor across classes and a unique factor for each class. Their method could be solved by the iterative weighted Glasso [17]. Cai et al. [8] proposed a constrained l_1 minimization approach to sparse precision matrix estimation. They presented the convergence rate between the estimator and the true matrix in the spectral norm and Frobenius norm. However, there is a limitation that the estimate has to be symmetrized. Liu and Wang [34] introduced a tuning-insensitive approach for optimally estimating Gaussian graphical models (TIGER). They applied column-by-column regression to estimate the precision matrix. They showed

the computational efficiency of the method by solving a simple SQRT-Lasso problem. In addition, Danaher et al. [11] used a generalized fused lasso or group lasso as the penalty and employed the alternating directions method of multipliers (ADMM) algorithm to solve the optimization problem. Cheng et al. [9] developed a multi-level graphical modeling approach.

1.2 Overview

The rest of the dissertation is organized as follows. In Chapter 2, we propose a semiparametric omnibus test for identifying the significance of a functional association between covariates with measurement error and the clustered binary outcomes by taking into account the effect modification of matching covariates. We introduce a flexible omnibus test for testing purposes using an efficient score based on estimating equations of null hypothesis. The advantages of our omnibus test are demonstrated through simulation studies and application with 1-4 bidirectional matched data from an epidemiology study. In Chapter 3, we propose a joint semiparametric kernel machine network approach. Our proposed approach is a unified and integrated method which can simultaneously identify important variables and build a network among them. The performance of our approach is evaluated in two aspects of simulation studies: variable selection and network estimation. We also demonstrate the advantage of our approach using a type II diabetes gene pathway data. In Chapter 4, we propose a nonparametric focal-area detection method for multi-class dynamic model using a fully Bayesian hierarchical modeling. Our goal is to simultaneously estimate unknown functional trends of multi-level compounds in mixture as well as to detect specific focal areas so as to have computational efficiency on the estimation of new data. We describe gas chromatographic data on Fast Odor Chromatographic Sniffer (FOX) system to demonstrate the advantage of our proposed approach. In Chapter 5, we summarize the contributions of this dissertation and discuss directions for future research.

Chapter 2

Flexible Omnibus Test in 1-M Matched Case-Crossover Study with Measurement Error in Covariate

2.1 Introduction

In matched case-crossover studies, conditional logistic regression is commonly used for analysis because any stratum effect can be removed [21]. Kim et al. [24] generalized the traditional conditional logistic regression to semiparametric regression using regression splines. On the other hand, as an alternative approach, the generalized linear mixed model has been used by treating the stratum effect as a random variable [12, 41]. The parameters from generalized linear mixed models can be estimated with a normal distribution assumption [7, 44]. This approach, however, relies on a strong distribution assumption of the stratum random effect, e.g., normal distribution. Furthermore, estimating nuisance parameters, the stratum effect, may cause unknown effects on the estimation of the parameter of interest.

Although it is generally accepted that covariates for which cases and controls are matched

cannot exert a confounding effect on independent covariates included in the analyses, effect modification by matching covariates may occur. The methods for assessing and characterizing potential effect modifications by matching covariates are quite limited. Kim et al. [23] proposed a graph-based representation method for effect heterogeneity by a matching covariate. This method was based on a parametric conditional logistic model with varying coefficients for potential effect modification. It detected a parametric effect modification from matching covariates on the relative risk of disease. However, the model cannot be applied to mixed parametric and nonparametric relationships between predictors and the relative risk. Also, Kim et al. [25] proposed semiparametric and nonparametric models with varying coefficients for an ordered categorical matching stratum. Kim et al. [26] further developed penalized and Bayesian semiparametric varying-coefficient models. Ortega-Villa et al. [42] investigated the prediction accuracy when using the time-varying effect of covariate by employing a nonparametric model with a Bayesian approach. However, the major limitation of these approaches is their inability to evaluate measurement error in covariates. Often, covariates in such studies are measured with error. Not accounting for this error can lead to incorrect inferences for all covariates in the model [51]. The challenging problem in a measurement error model is how to specify the distribution of measurement error because this distribution is unknown. A misspecified distribution also causes incorrect inference.

The methods for simultaneously evaluating effect modification by matching effect as well as assessing and characterizing error-in-covariates in matched case-control studies are quite limited. Most of the existing methods in the matched study are based on estimation, without testing the functional relationship between clustered binary outcomes and covariates and ignoring possible measurement error in covariates. Tests to identify functional associations between covariates and the clustered binary outcomes have not yet been studied. Our goal is to propose a flexible testing procedure that considers measurement error but does not require any distributional assumptions, while the existing method is based on the parametric

distributional assumption of the measurement error. Thus, we developed a "flexible omnibus test" for testing purposes with much weaker requirements. It was developed under a semiparametric framework in which a likelihood function is no longer necessary.

Our flexible omnibus test involves a redesigned efficient score without the requirement for an explicit likelihood. The idea for this type of test [20, 35] was used for a goodness-of-fit test. Hart [20] proposed an omnibus goodness-of-fit test that is a hybrid of Bayesian and frequentest ideas, which was developed by using a Laplace approximation. Ma et al. [35] adapted it to the general measurement error framework and proposed both local and omnibus tests under the assumption of independent outcome variables. However, our matched case-crossover studies have clustered binary outcomes and contain the possibility of effect modification by matching the covariates. Hence, in this chapter of the dissertation, we propose a semiparametric omnibus test for the purpose of testing the significance of the functional relationship between the clustered binary outcomes and covariates with the general measurement error framework by taking into account the effect modification of matching covariates.

Our flexible omnibus test has the following features: (a) it is applicable for clustered binary outcomes; (b) it can be useful for testing whether there is effect modification by matching covariates; (c) it does not require specific alternative hypotheses so that flexible inference is available; (d) it does not require the estimation of the parameters for the alternative model, which can reduce the burden of the computation; (e) it does not require the distributional assumption of the measurement error. To the best of our knowledge, no current test statistics have all of these features.

This dissertation is motivated by an example dataset on incidence of aseptic meningitis against exposure to water turbidity in children from South Korea. The dataset contains information about hospital admissions due to aseptic meningitis and drinking water turbidity in 669 patients who are younger than 15 years old in urban communities in South Korea. Note

that aseptic meningitis is an inflammation of the layers lining the brain without any bacterial infection. Children have the high risk of aseptic meningitis. Based on the etiologic research, enteroviruses are the primary viruses that cause aseptic meningitis [28]. It is reported that incomplete and undefined etiological waterborne outbreaks may be highly related to the enteroviruses [27].

We used the 1-4 bidirectional matched case-crossover design, which controls the characteristic within a subject [22]. Thus, one case was matched to four controls; the two control measurements are collected 35 days before and 42 days after the case measurement. By considering the information that the virus that causes the aseptic meningitis can survive for about 31 days, the control periods of the experimental design were set to 35 days before and 42 days after the case measurement because they do not overlap with the latent period of 31 days. We collected 31 datasets corresponding to 31 lag days. Denote these 31 datasets as "Data 0", "Data 1",..., "Data 30" and represent d as the lag day within the latent period, which should not be confused with calendar date: "Data 0" contains measurements collected on disease onset, while "Data 1" includes the measurements for dates that are one day ahead of the dates of "Data 0" for the controls and case. For example, for any subject in "Data 0," the date of case measurement is 4/15, and $Y = (0, 0, 1, 0, 0)$ at dates $(3/4, 3/11, 4/15, 5/20, 5/27)$; here, $d = 1$. In "Data 1," the case date is considered to be one day before onset, and $Y = (0, 0, 1, 0, 0)$ at $(3/3, 3/10, 4/14, 5/19, 5/26)$; here, $d = 2$. In a similar way, we defined the remaining datasets until "Data 30," in which the case date is thirty days before the onset of "Data 0" ($d = 31$), collected at dates $(2/2, 2/9, 3/16, 4/20, 4/27)$. Note that the dates might not be matched between subjects, but the lagged time information is the same. Figure 2.1 contains a visual representation of the datasets.

One can investigate the association between aseptic meningitis and drinking water turbidity by using a traditional conditional logistic regression. However, this analysis cannot reflect the characteristics of these 31 latent period data structures due to the limitations

of conditional logistic regression. To investigate the time lag effect, which is the average delay between the time of the exposure onset and natural death of the virus, the study data consists of 31 daily measurements of the patients at disease onset. Our main questions of interest are (1) to identify the significance of the functional relationship between the incidence of meningitis and exposure to water turbidity based on each lag by considering the possibility of measurement error in the covariate and time-varying effect; (2) to test whether there is effect modification by matching covariates, time, and (3) any existence of interaction effects between two covariates. There are no existing methods to test the significance of the functional relationship between the covariates with the measurement error and the response in a matched case-control study.

The Chapter 2 of this dissertation is organized as follows. In Section 2.2, we first explain the limitations of conditional logistic regression and then introduce a semiparametric framework in a 1-M bidirectional matched case-crossover study with general measurement error framework. In Section 2.3, we propose flexible tests in matched case-crossover studies. In Section 2.4, we describe the score-type test. In Section 2.5, we describe how to construct a local test and the flexible omnibus test. In Section 2.6, we perform the simulation studies with three settings to test: (1) nonlinear functional association between a clustered binary outcome and a covariate with the measurement error, (2) a time-varying coefficient with the mis-measured covariate, and (3) the interaction effect between a mis-measured covariate and a fixed covariate. In Section 2.7, we apply our flexible omnibus test to our motivating example. Lastly, our concluding remarks are presented in Section 2.8.

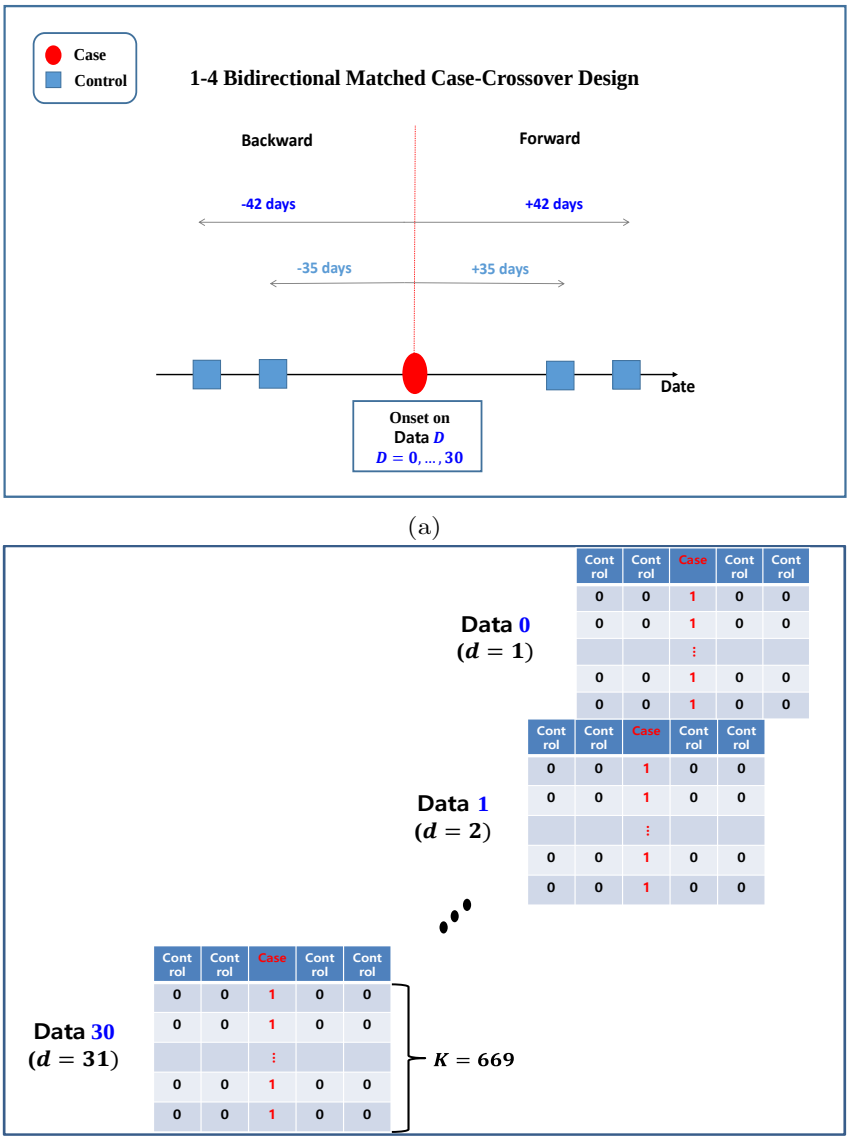


Figure 2.1: Motivating 1-4 bidirectional matched case-crossover example dataset on incidence of aseptic meningitis against exposure to water turbidity in 669 children; (a) One case is matched to four controls; the two control measurements are collected 35 days before and 42 days after the case measurement; Onset is varied by 31 datasets corresponding to 31 lag days; (b) These 31 datasets are denoted as "Data 0", "Data 1",..., "Data 30" and also d denote as the lag day; "Data 0" contains measurements collected on disease onset; "Data 1" includes the measurements which dates are one day ahead of the dates of "Data 0" for controls and case; the remaining datasets until "Data 30" in which the case date is thirty days before onset of "Data 0".

2.2 Semiparametric Framework in 1-M Matched Case-Crossover Studies with Measurement Error

For unknown effects on V, we can consider a varying coefficient that can be modeled nonparametrically. We assume that $\boldsymbol{Z}$ does not have measurement error and X has a measurement error. We consider the classical measurement error model. The covariate X is not observable; instead, we employ a surrogate for $\boldsymbol{X}_k$, denoted by $\boldsymbol{W}_k$. Unlike parametric assumption on a distribution of the classical measurement error, we do not assume it. Suppose we have $\boldsymbol{W}_k = (W_{jk1}, ..., W_{jkR}; j = 1, \ldots, M+1; k = 1, \ldots, K; r = 1, \ldots, R)$, where $\boldsymbol{W}_{jk}$ is the jth covariate in the kth stratum and R is the number of the replicates, that is, $W_{jkr} = X_{jk} + U_{jkr}$, where U_{jkr} is the measurement error. We assume that $\boldsymbol{U}_k$ is independent of $(\boldsymbol{X}_k, \boldsymbol{Y}_k)$, where $\boldsymbol{Y}_k = (Y_{1k}, Y_{2k}, ..., Y_{M+1k})$.

For an unknown association between outcomes and X, an unknown effect on stratum effect V, and an unknown interaction between two covariates, we consider the following semiparametric frameworks, respectively,

$$
\begin{aligned}
Pr(Y_{jk} = 1 | \mathbf{Z}_k, \mathbf{X}_k, V, \sum_{j=1}^{M+1} Y_{jk} = 1) &= \frac{1}{1 + \sum_{j=2}^{M+1} \exp[(\boldsymbol{Z}^*_{jk})^T \boldsymbol{\beta}_0 + X^*_{jk}\beta_1 + \{\theta^*_1(X_{jk})\}]}; \\
Pr(Y_{jk} = 1 | \mathbf{Z}_k, \boldsymbol{X}_k, V, \sum_{j=1}^{M+1} Y_{jk} = 1) &= \frac{1}{1 + \sum_{j=2}^{M+1} \exp\{(\boldsymbol{Z}^*_{jk})^T \boldsymbol{\beta}_0 + X^*_{jk}\beta(V)\}}; \\
Pr(Y_{jk} = 1 | \mathbf{Z}_k, \boldsymbol{X}_k, V, \sum_{j=1}^{M+1} Y_{jk} = 1) &= \frac{1}{1 + \sum_{j=2}^{M+1} \exp[(\boldsymbol{Z}^*_{jk})^T \boldsymbol{\beta}_0 + X^*_{jk}\beta_1 + \{\theta^*_3(\boldsymbol{Z}_{jk}, X_{jk})\}]},
\end{aligned}
$$

where $\boldsymbol{Z}^*_{jk} = \boldsymbol{Z}_{jk} - \boldsymbol{Z}_{1k}$, $X^*_{jk} = X_{jk} - X_{1k}$, $\theta^*_1(X_{jk}) = \theta_1(X_{jk}) - \theta_1(X_{1k})$, and $\theta^*_3(\boldsymbol{Z}_{jk}, X_{jk}) = \theta_3(\boldsymbol{Z}_{jk}, X_{jk}) - \theta(\boldsymbol{Z}_{1k}, X_{1k})$. $\theta_1(\bullet)$, $\beta(\bullet)$, and $\theta_3(\bullet, \bullet)$ are unknown functions.

Our goal is to test an unknown association between outcomes and X, an unknown effect on stratum effect V, and an unknown interaction between two covariates without estimating $\theta_1(\bullet)$, $\beta(\bullet)$, and $\theta_3(\bullet, \bullet)$, under the semiparametric framework, by proposing our flexible

omnibus test. Explicit likelihoods may be neither available nor necessary because the actual forms of $\theta_1(\bullet)$, $\beta(\bullet)$, and $\theta_3(\bullet,\bullet)$ are unknown functions. Thus, we may be unable to derive an explicit likelihood.

2.3 Flexible Omnibus Tests in Matched Case-crossover Studies

First, let us illustrate the features of our flexible omnibus test.

2.3.1 Features of Omnibus Test

Our main questions of interest are to test whether there is an alternative model having unknown-nonlinear direction of the departure from null model,

$$Pr(Y_{jk}=1|\mathbf{Z}_{jk},X_{jk},V)=H\{\mathbf{Z}_{jk}^T\boldsymbol{\beta}_0+\beta_1X_{jk}+q(k,V)\}.$$

We consider two tests: one is called a local test and the other is an omnibus test. A local test is for an alternative allowing for quadratic departures of X from linearity might be

$$Pr(Y_{jk}=1|\mathbf{Z}_{jk},X_{jk},V)=H\{\mathbf{Z}_{jk}^T\boldsymbol{\beta}_0+\beta_1X_{jk}+\gamma X_{jk}^2+q(k,V)\}.$$

In general, a local test can be considered as an alternative allowing for any parametric known form of X from the null model.

A omnibus test is for an alternative allowing for unknown-nonlinear departures of X from linearity might be

$$Pr(Y_{jk}=1|\mathbf{Z}_{jk},X_{jk},V)=H\{\mathbf{Z}_{jk}^T\boldsymbol{\beta}_0+\beta_1X_{jk}+\theta_1(X_{jk})+q(k,V)\}, \tag{2.1}$$

where $\theta_1(\bullet)$ is an unspecified function that is orthogonal to X.

Omnibus test is for an alternative allowing for nonlinear varying coefficient departures from a constant coefficient might be

$$Pr(Y_{jk} = 1|\mathbf{Z}_{jk}, X_{jk}, V) = H\{\mathbf{Z}_{jk}^T\boldsymbol{\beta}_0 + \{\beta_1 + \theta_2(V)\}X_{jk} + q(k, V)\}, \tag{2.2}$$

where $\theta_2(\bullet)$ is an unspecified function that is orthogonal to V. Here $\beta(V) \equiv \beta_1 + \theta(V)$.

Omnibus test is for an alternative allowing for the interaction effect departures from the main effects might be

$$Pr(Y_{jk} = 1|\mathbf{Z}_{jk}, X_{jk}, V) = H[\mathbf{Z}_{jk}^T\boldsymbol{\beta}_0 + \beta_1 X_{jk} + \theta_3(\mathbf{Z}_{jk}, X_{jk}) + q(k, V)], \tag{2.3}$$

where $\theta_3(\bullet, \bullet)$ is an unspecified bivariate function that is orthogonal to X and Z.

Under the models (2.1), (2.2), and (2.3), we can obtain the retrospective models and write the conditional likelihoods, respectively, as follows,

$$\begin{aligned}
&\prod_{k=1}^{K} \frac{1}{1 + \sum_{j=2}^{M+1} \exp[(\boldsymbol{Z}_{jk}^*)^T\boldsymbol{\beta}_0 + X_{jk}^*\beta_1 + \{\theta_1^*(X_{jk})\}]}; \\
&\prod_{k=1}^{K} \frac{1}{1 + \sum_{j=2}^{M+1} \exp\{(\boldsymbol{Z}_{jk}^*)^T\boldsymbol{\beta}_0 + X_{jk}^*\beta(V)\}}; \\
&\prod_{k=1}^{K} \frac{1}{1 + \sum_{j=2}^{M+1} \exp[(\boldsymbol{Z}_{jk}^*)^T\boldsymbol{\beta}_0 + X_{jk}^*\beta_1 + \{\theta_3^*(\boldsymbol{Z}_{jk}, X_{jk})\}]}.
\end{aligned}$$

The null model is equivalent to $\theta_1(\bullet) = 0$ or $\theta_3(\bullet, \bullet) = 0$. We can also show that the likelihood starting from prospective model is approximately equivalent to that from retrospective model in Appendix A.1.

2.3.2 Flexible Omnibus Tests

In general, specific departures with direction from the null hypothesis is unknown. Using a linear combination of sufficiently many basis functions, any smooth function can be approximated to an unknown function. By using this feature, we consider I basis functions, $h_1(Z,X),\ldots,h_I(Z,X)$, which are arranged from lowest to highest frequency. We then consider I different local tests with various directions of departure in the alternative hypotheses using our flexible omnibus test statistic. By considering only one omnibus test statistic, we can avoid the multiple testing.

Let $p_{Y|Z,X,\sum Y=1}$ denote the retrospective model and express the retrospective model (1.2) as $p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k,(\boldsymbol{Z}_{jk}-\boldsymbol{Z}_{1k})^T\boldsymbol{\beta}_0+(X_{jk}-X_{1k})\beta_1\}$. The null hypothesis ($H_0$) can be written as follows. The three alternative hypotheses are (1) the ith alternative hypothesis (H_{i1}) for unknown-nonlinear departures from the null model, (2) the ith alternative hypothesis (H_{i2}) for unknown-nonlinear departures of effect modification by V from the null model, and (3) the ith alternative hypothesis (H_{i3}) for unknown-nonlinear departures of interaction from the null model, $i=1\ldots I$, which are also expressed as follows,

$$H_0: Pr(Y_{jk}=1|\boldsymbol{Z}_k,X_{jk},\sum_{j=1}^{M+1}Y_{jk}=1)=p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k,(\boldsymbol{Z}_{jk}-\boldsymbol{Z}_{1k})\boldsymbol{\beta}_0+(X_{jk}-X_{1k})\beta_1\};$$

$$H_{i1}: Pr(Y_{jk}=1|\boldsymbol{Z}_k,X_{jk},\sum_{j=1}^{M+1}Y_{jk}=1)=p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k,(\boldsymbol{Z}_{jk}-\boldsymbol{Z}_{1k})\boldsymbol{\beta}_0+(X_{jk}-X_{1k})\beta_1+\gamma_{i1}\{h_i(X_{kj})-h_i(X_{k1})\}\};$$

$$H_{i2}: Pr(Y_{jk}=1|\boldsymbol{Z}_k,X_{jk},\sum_{j=1}^{M+1}Y_{jk}=1)=p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k,(\boldsymbol{Z}_{jk}-\boldsymbol{Z}_{1k})\boldsymbol{\beta}_0+(X_{jk}-X_{1k})\beta_1+\gamma_{i2}h_i(V)\};$$

$$\begin{aligned}H_{i3}: Pr(Y_{jk}&=1\ |\boldsymbol{Z}_k,X_{jk},\sum_{j=1}^{M+1}Y_{jk}=1)\\&=\ p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k,(\boldsymbol{Z}_{jk}-\boldsymbol{Z}_{1k})\boldsymbol{\beta}_0+(X_{jk}-X_{1k})\beta_1+\gamma_{i3}(h_i(\boldsymbol{Z}_{kj},X_{kj})-h_i(\boldsymbol{Z}_{k1},X_{k1}))\}.\end{aligned}$$

Our tests on $\theta_1(\bullet)=0$, $\theta_2(\bullet)=0$, or $\theta_3(\bullet,\bullet)=0$ under models (2.1)-(2.3) are equivalent to H_0: $\gamma_i=0$ vs H_{ia}: $\gamma_{ia}\neq 0$, where $a=1,2,3$.

We propose a score-type test based on an estimating equation that does not require the full model. The main advantage of this test is that we can construct the test statistic only under the null model.

2.4 Score-type Test in Matched Case-crossover Study

In order to obtain a score-type test, we first estimated the distribution of measurement error using kernel density estimation and then calculated the marginal likelihood under the null hypothesis by taking integration of the joint likelihood with respect to X. We then calculated an estimating equation. These procedures are described in Sections 2.4.1- 2.4.3.

2.4.1 Measurement Error and Marginal Likelihood under The Null

Using the multiple measurements that we described in Section 2.2, we can obtain pseudo-covariate and pseudo-measurement error using kernel density estimation, which we explain in Appendix A.2. We denote this estimated density of the measurement error as $\hat{p}_U$.

Before calculating the estimating equation, we need to calculate the marginal likelihood L_k under the null hypothesis. Define $\boldsymbol{\beta} = (\boldsymbol{\beta}_0^T \ \beta_1)^T$. This calculation requires integration of the complete likelihood with respect to X. If we have the distribution of $X|Z$, $p_{X|Z}(\bullet)$, we can obtain marginal likelihood as follows,

$$L_k(\boldsymbol{\beta}, \gamma = 0) = \int l_k(\boldsymbol{\beta}, \gamma = 0 | \boldsymbol{Z}_k, \boldsymbol{X}_k, \sum_{j=2}^{M+1} Y_{jk} = 1) p_{X_k|Z_k}(\boldsymbol{X}_k) \hat{p}_U(\boldsymbol{W}_k - \boldsymbol{X}_k) \quad d\mathbf{X}_k,$$

where $\boldsymbol{\beta} = (\boldsymbol{\beta}_0^T \ \beta_1)^T$. The Metropolis-Hastings algorithm is employed. The candidate observation of X can be generated by using the Markov Chain Monte Carlo (MCMC) method. Details about calculating the marginal likelihood under the null are explained in Appendix A.3.

2.4.2 Estimating Equation

Let $\phi_{\boldsymbol{\beta},k}(\bullet)$ and $\psi_{\gamma,k}(\bullet)$ denote estimating functions of $\boldsymbol{\beta}$ and γ, respectively, and have same dimensions as $\boldsymbol{\beta}$ and γ. The estimating functions can then be obtained from

$$\phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)=\frac{\partial L_k(\boldsymbol{\beta},\gamma)}{\partial\boldsymbol{\beta}}\Big|_{\gamma=0} \quad \& \quad \psi_{\gamma,k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)=\frac{\partial L_k(\boldsymbol{\beta},\gamma)}{\partial\gamma}\Big|_{\gamma=0}.$$

We then finally obtain the estimating equations which satisfy

$$\sum_{k=1}^{K}\phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)=0 \quad \& \quad \sum_{k=1}^{K}\psi_{\gamma,k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)=0.$$

Under the null hypothesis, $h(\cdot)$s disappear. Therefore, the score-type function can be derived from the estimating equation under the null model but has similar properties to the original score function. The estimating equation can be simplified to

$$\sum_{k=1}^{K}\phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},0,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)=0.$$

By replacing $\boldsymbol{\beta}$ with its root $\hat{\boldsymbol{\beta}}$, the estimated score becomes

$$\widehat{U}=\frac{1}{\sqrt{K}}\sum_{k=1}^{K}\psi_{\boldsymbol{\beta},k}(\hat{\boldsymbol{\beta}},0,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U).$$

Detailed procedures for deriving the estimating equations for our flexible omnibus test statistic on alternative hypotheses H_{i1}–H_{i3} are provided in Appendix A.4.

2.4.3 Score-type Test Statistic Based on Estimating Equation

We first define some formulas for all the following expectations under the null hypothesis:

$$
\begin{aligned}
A_1 &= E\left\{\frac{\partial \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},0,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)}{\partial \boldsymbol{\beta}^T}\right\},\quad A_2 = E\left\{\frac{\partial \psi_{\gamma,k}(\boldsymbol{\beta},0,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)}{\partial \boldsymbol{\beta}^T}\right\},\\
A_3 &= E\left\{\frac{\partial \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)}{\partial \gamma}\right\},\quad A_4 = E\left\{\frac{\partial \psi_{\gamma,k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)}{\partial \gamma}\right\},\\
B_{11} &= E\{\phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},0,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)\phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},0,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)^T\},\\
B_{22} &= \mathrm{cov}\{\psi_{\gamma,k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)\},\\
B_{12} &= \mathrm{cov}\{\phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta},0,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U),\psi_{\gamma,k}(\boldsymbol{\beta},\gamma,\boldsymbol{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U)\},\\
V_\beta &= A_1^{-1}B_{11}(A_1^{-1})^T,\quad \Sigma_0 = \mathrm{cov}\{\psi_\gamma(\cdot,\boldsymbol{\beta},0,\hat{p}_u) - A_2A_1^{-1}\phi_{\boldsymbol{\beta}}(\cdot,\boldsymbol{\beta},0,\hat{p}_u)\}.
\end{aligned}
$$

If we denote the sample covariance matrix of Σ_0 as $\widehat{\Sigma}_0$, then our score-type test statistic can be written as $\widehat{T} = \widehat{U}^T\widehat{\Sigma}_0^{-1}\widehat{U}$. The proposed score-type test statistic $\widehat{T}$ asymptotically follows χ^2 with p_T degrees of freedom. The proof of this argument is derived in Appendix A.5. Hence, we develop our flexible omnibus test on the basis of the score-type test.

2.5 Flexible Omnibus Test Statistic

Without loss of generality, we let $h_i(\cdot)$ be the basis function for the alternative model. For each $h_i(\cdot)$, the score-type test statistic can then be written as $\widehat{T}_i^2 = \widehat{u}_i^2/\widehat{\sigma}_i^2$, where $\widehat{u}_i$ and $\widehat{\sigma}_i^2$ are the one-dimensional versions of $\widehat{U}_i$ and $\widehat{\Sigma}_{0i}$, respectively. For the (m,n)th element of Σ, we further define A_{m2} and B_{m12} of A_2 and B_{12} similarly for the one-dimensional version of A_{n2} and B_{n12} of A_2 and B_{12} described in Section 2.4.3.

Let $\widehat{\boldsymbol{T}} = (\widehat{T}_1,...,\widehat{T}_I)^T$. We can show that $\sqrt{K}\,\widehat{\boldsymbol{T}} \sim N(\boldsymbol{0},\Sigma)$ asymptotically under the null

hypothesis, where the (m,n)th element of Σ can be estimated as

$$E\Big(\frac{\hat{u}_m}{\hat{\sigma}_m}\frac{\hat{u}_n}{\hat{\sigma}_n}\Big) = \frac{1}{\sigma_m\sigma_n}\Big[A_{m2}A_1^{-1}B_{11}A_1^{-1T}A_{n2}^T - A_{m2}A_1^{-1}B_{n12} - A_{n2}A_1^{-1}B_{m12} + E\{\psi_{\gamma_m,k}(\cdot;\hat{\boldsymbol{\beta}},0,\hat{p}_U)\psi_{\gamma_n,k}(\cdot;\hat{\boldsymbol{\beta}},0,\hat{p}_U)\}\Big]$$

because $E(\widehat{u}_m/\widehat{\sigma}_m)=0$ for any m, where $1\le m,n\le I$. This also means that the marginal limit distribution is $\widehat{u}_m/\widehat{\sigma}_m \to N(0,1)$ in distribution. Our score-type function is then

$$\widehat{U} = \frac{1}{\sqrt{K}}\sum_{k=1}^{K}\psi_{\boldsymbol{\beta},k}(\widehat{\boldsymbol{\beta}},0,\mathbf{Z}_k,\boldsymbol{W}_k,\boldsymbol{Y}_k,\hat{p}_U).$$

We then propose a flexible omnibus test using I local tests, where T_i^2 is the ith local test statistic, $i=1,\ldots,I$. We have combined the multiple score-type test statistics to build an omnibus test, even though it is difficult to obtain a closed-form distribution or asymptotic distribution. By following the approach taken to obtain a value of χ^2 from the sum of the square of normal distributions, we designed our flexible omnibus test statistic, which can be expressed as $\widehat{\mathcal{T}} = \sum_{i=1}^{I}\omega_i\exp(\widehat{T}_i^2/2)$, where ω_i is a weight. The purpose of this weight term is to assign larger weights to bases with lower frequency than higher frequency. The basis functions are arranged from lowest to highest frequency with I basis functions. By ordering the basis functions with the weight, the test can have better power than using arbitrary order of the basis functions.

The omnibus test statistic generally weighs basis function reflecting simple and global features. If there is prior information that some alternatives are more feasible for capturing the true function than others, then the test can give high weight to them and reorder the basis functions to obtain more powerful test. For the number of the basis functions used in the proposed test, the power of the test becomes better as the number of the basis functions increases. However, the performance of the omnibus test is not sensitive to the

number of basis functions when it is beyond some minimum, and rather becomes worse when the test uses an excessively large number of the basis functions due to the occurrence of negative definite covariance matrices. We describe the performance of our flexible omnibus test related to the number of basis functions using simulation study in Section 2.6. Assuming that each of the indices of i is proportional to the local test statistic associated with the corresponding basis, it is reasonable to specify the prior of the corresponding local test statistic as $\pi_i = (1+i^c)^{-1},\ i = 1, \ldots, I,$ where $c > 1$ for the local test statistics. Our weight ω_i is $\pi_i/(1-\pi_i)$. With this weight, our flexible omnibus test statistic can be rewritten as $\widehat{\mathcal{T}} = \sum_{i=1}^{I}\{\pi_i/(1-\pi_i)\}\exp(\widehat{T}_i^2/2)$.

With the selection of $c = 2$, we have $\omega_i = 1/i^2$. Our flexible omnibus test statistic finally becomes

$$\widehat{\mathcal{T}} = \sum_{i=1}^{I} \frac{1}{i^2} \exp\left(\widehat{T}_i^2/2\right). \tag{2.4}$$

The p-value of our omnibus test can be calculated from the result that $\widehat{\mathcal{T}}$ has an asymptotically multivariate normal distribution. With the semiparametric framework under matched case-crossover study, our exible omnibus test algorithm proceeds. The algorithm for calculating the empirical p-value comprises the following steps:

Step T1: Using the estimating equations based on the multiple basis functions $\boldsymbol{h}(\cdot) = \{h_1(\cdot), ..., h_I(\cdot)\}$, obtain the multiple score-type test statistics $\widehat{\boldsymbol{T}}_{\text{obs}} = (\widehat{T}_1, ..., \widehat{T}_I)^T$, where $\widehat{T}_i^2 = \widehat{u}_i^2/\widehat{\sigma}_i^2$ with the ith local test;

Step T2: Define $\widehat{\Sigma}$ as the estimated covariance matrix for $\widehat{\boldsymbol{T}}_{\text{obs}}$. The (m, n) element of $\widehat{\Sigma}$ can be derived from Equation where $1 \le m, n \le I$. Compute $\widehat{\mathcal{T}}_{\text{obs}} = \sum_{i=1}^{I} i^{-2}\exp(\widehat{T}_i^2/2)$ such as Equation (2.4);

Step T3: Generate a set of the multiple local test statistics $\boldsymbol{T}_b$ under H_0. That is, we generate the vector $\boldsymbol{T}_b = (T_{1,b}, ..., T_{I,b})$ from the multivariate $N(0, \widehat{\Sigma})$ distribution, and then

calculate the generated flexible omnibus test statistic such that

$$\mathcal{T}_b = \sum_{i=1}^{I} \frac{1}{i^2} \exp(T_{i,b}^2/2).$$

In the same way, generate the omnibus test statistic and save them many times $(b = 1, ..., B)$.

Step T4: The empirical p-value can be estimated as p-value$\approx B^{-1} \sum_{b=1}^{B} I\{\mathcal{T}_b > \widehat{\mathcal{T}}_{\text{obs}}\}$.

These procedures are summarized in Algorithm 1.

Algorithm 1 Algorithm for Flexible Omnibus Test in the matched study

for $i = 1$ *to* I **do**
  Calculate $\widehat{T}_i^2 = \widehat{u}_i^2/\widehat{\sigma}_i^2$ based on basis function $h_i(\cdot)$;
  $\widehat{\boldsymbol{T}}_{obs}$=$\widehat{\boldsymbol{T}}_{obs} \cup \widehat{T}_i^2$;
end
Compute $\widehat{\Sigma}$ based on $\widehat{\boldsymbol{T}}_{obs}$;
Compute $\widehat{\mathcal{T}}_{obs} = \sum_{i=1}^{I} i^{-2} \exp(\widehat{T}_i^2/2)$;
for $b = 1$ *to* B **do**
  Generate $\boldsymbol{T}_b = (T_{1,b}, \ldots, T_{I,b})$ from multivariate $N(\mathbf{0}, \widehat{\Sigma})$; Calculate the generated flexible omnibus test statistic such that $\mathcal{T}_b = \sum_{i=1}^{NT} i^{-2} \exp(T_{i,b}^2/2)$;
end
Estimate empirical p-value by p-value$\approx B^{-1} \sum_{b=1}^{B} I\{\mathcal{T}_b > \widehat{\mathcal{T}}_{\text{obs}}\}$;

The proposed omnibus test for matched case-crossover study can provide the flexibility to make inference on various settings of hypotheses on $H_{i1} - H_{i3}$ because the forms of omnibus test statistic on the alternative hypotheses are all same but they are built with different estimating equations on each hypothesis.

2.6 Simulation

We conducted simulation studies under four cases to reflect the flexibility of our testing: Case 1 is the scenario when the distribution of the measurement error is correctly specified, but the distribution of the unobservable covariate is incorrectly selected; Case 2 is the scenario when the distributions of both the covariate and the measurement error are misspecified; Case 3 is the setting when effect modification by V exists; and Case 4 is the setting when there is an interaction between two covariates, Z and X. We will explain how to generate each case in detail shortly.

By motivating from our example of aseptic meningitis incidence, we consider 1-4 bidirectional matched case-control studies. Each stratum consists of one case and four controls. That is, $\boldsymbol{y}_k = (0, 0, 1, 0, 0)$, where $k = 1, \ldots, K$ and $K = (100, 200, \ldots, 1,000)$. We consider a classical measurement error model with the surrogate, $w_{kjr} = x_{kj} + u_{kjr}$, where $j = 1, \ldots, 5$ and $r = 1, \ldots, 6$ $(R = 6)$.

2.6.1 Simulation Setting

(1) Case 1: Misspecification of the distribution of the covariate X

In this case, the null and alternative models are H_0: $Pr(Y_{jk} = 1|X_{jk}, k, V) = H\{\beta_1 X_{jk} + q(k, V)\}$ vs H_1: $Pr(Y_{jk} = 1|X_{jk}, k, V) = H\{\beta_1 X_{jk} + \beta_2 X_{jk}^2 + q(k, V)\}$, where $H(\bullet)$ is the logistic distribution function, $q(\bullet) \sim N(0, 1)$, $U \sim N(0, 0.2^2)$, true parameters $\beta_1 = 1$ and $\beta_2 = 5/\sqrt{K}$ in consideration with sensitivity of the power test for each size of the strata K, and the latent covariate X are generated from the following two cases:

- Case 1 (a): True latent covariate $X \sim N(-0.5, 1)$ but we use uniform distribution based on the minimum and maximum of the sample X; that is $p_X = N(-0.5, 1)$ and $p_X^* = \text{Unif}[\min(\text{x}), \max(\text{x})]$, where p_X and p_X^* represent the true latent and

misspecified distributions of X, respectively.

- Case 1 (b): True latent covariate $X \sim \text{Unif}[-4, 3]$, but we use the misspecified $N(\overline{W}, S_w)$, where $\overline{W}$ and S_w are the sample mean and variance of W; that is, $p_X = \text{Unif}[-4, 3]$ and $p_X^* = N(\overline{W}, S_w)$.

(2) Case 2: Misspecification of distributions of both the covariate X and measurement error U

This case is the same setting as Case 1 (b) except that the measurement error U is generated from uniform distribution, $\text{Unif}[-0.7, 0.7]$, to satisfy the assumption about symmetry of the distribution p_U. Let p_U^* be the misspecified distribution of U. The true parameters are set as $\beta_1 = 0.5$ and $\beta_2 = 1/\sqrt{K}$ to easily generate $\boldsymbol{y}_k = (0, 0, 1, 0, 0)$. We use normal distributions for both misspecified distributions, that is, $p_X = \text{Unif}[-4, 3]$ and $p_U = \text{Unif}[-0.7, 0.7]$, but $p_X^* = N(\overline{W}, S_w)$ and p_U^* is based on kernel density estimation described in Appendix A.2.

(3) Case 3: Existence of varying coefficient

The null model is the same setting as the model in Case 1 (b), but the alternative model is H_a: $Pr(Y_{jk} = 1|X_{jk}, k, V) = H\{\beta(V)X_{jk} + q(k, V)\}$, where $\beta(v) = \cos(\pi v)$ with $v = 1, 2, \ldots, 25$ and $K(v) = (40, 400)$. $p_X = \text{Unif}[-4, 3]$ and $p_X^* = N(\overline{W}, S_w)$ for the misspecification case.

(4) Case 4: Existence of interaction between two covariates

In this case, the null and alternative models $Pr(Y_{jk} = 1|Z_{jk}, X_{jk}, k, V) = H\{\beta_1 Z_{jk} + \beta_2 X_{jk} + q(k, V)\}$ and $Pr(Y_{jk} = 1|Z_{jk}, X_{jk}, k, V) = H\{\beta_1 Z_{jk} + \beta_2 X_{jk} + \beta_3 (X_{jk})^2 \sin(Z_{jk}) + q(k, V)\}$, where $Z \sim N(0.5, 0.5^2)$, $X \sim N(-0.5, 1)$, $U \sim N(0, 0.2^2)$, and $q(\bullet) \sim N(0, 0.1^2)$. True parameters are set as $\beta_1 = 1$, and $\beta_2 = 1$, and $\beta_3 = 10/\sqrt{K}$ to generate $\boldsymbol{y}_k = (0, 0, 1, 0, 0)$.

For each combination of case and K value, we simulate 200 datasets with 1-4 matched study with K strata. The distribution of U is estimated using kernel density estimation. The type-I error and power are estimated to investigate the performance of our flexible omnibus test. A set of the trigonometric basis functions, $\boldsymbol{h}(x) = \{\cos(x), \sin(x), \cos(2x), \sin(2x), \cos(3x), \sin(3x)\}$ for Cases 1-2, $\boldsymbol{h}(v) = \{\cos(v), \sin(v), \cos(2v), \sin(2v), \cos(3v), \sin(3v)\}$ for Case 3, and $\boldsymbol{h}(x+z) = \{\cos(x+z), \sin(x+z), \cos\{2(x+z)\}, \sin\{2(x+z)\}, \cos\{3(x+z)\}, \sin\{3(x+z)\}\}$ for Case 4 are employed, respectively. Since the trigonometric functions belong to Fourier series, our testing procedures can capture the feature of interaction effect.

2.6.2 Sensitivity Analysis on The Number of Basis Functions

We also conducted simulation to investigate the performance of the flexible omnibus test based on the number of the basis functions. We vary the number of basis functions from 2 to 80. Under the condition that the distribution p_X is correctly chosen, the testing procedure is performed in the same setting as Case 1 in Section 2.6.1. We consider the number of strata K as 100 and 600.

2.6.3 Simulation Results

For Case 1, where the distribution of the covariate, p_X is misspecified, we considered two misspecified distributions of X. For Case 1 (a), we used the misspecified distribution of X using uniform distribution. We considered nominal levels 0.01, 0.05, and 0.1. The average type-I error and power are summarized in Table 2.1. The type-I error is close to the nominal value, and its power is almost 1 based on 200 simulated datasets regardless of the number of strata. For Case 1 (b), we employ the misspecified distribution of X using normal distribution based on sample mean and sample variance of the surrogate. The average values of type-I error rate and power are summarized in Table A.1 in Appendix A.8, which also suggests

out-performances of our approach except when the number of strata is from 100 to 300 at the nominal level 0.01. The overall performances of our approach under Case 1 are consistent with the result when the correct model is specified.

Table 2.1: The average value of empirical type I errors and powers of the flexible omnibus test under Case 1 (a); The two rows from the top and two rows from the bottom of the table represent the average values of type I error and power based on each nominal level (α) and the number of strata (K), respectively; p_X is the correct model and p_X^* is the misspecified uniform distribution of the latent variable X; For each combination of α and K value, we simulated 200 datasets with 1-4 matched case-crossover study with K strata.

		The values of K									
		100	**200**	**300**	**400**	**500**	**600**	**700**	**800**	**900**	**1000**
Type I error	Normal p_X										
	α =**0.01**	0.010	0.010	0.025	0.005	0.010	0.005	0.005	0.015	0.010	0.010
	0.05	0.040	0.075	0.060	0.050	0.045	0.040	0.060	0.040	0.050	0.055
	0.1	0.105	0.070	0.100	0.140	0.085	0.140	0.115	0.115	0.100	0.100
	Uniform p_X^*										
	α =**0.01**	0.015	0.010	0.020	0.005	0.005	0.000	0.010	0.010	0.015	0.010
	0.05	0.050	0.075	0.055	0.050	0.035	0.040	0.045	0.035	0.050	0.045
	0.1	0.095	0.075	0.100	0.145	0.090	0.145	0.115	0.125	0.100	0.100
Power	Normal p_X										
	α =**0.01**	0.960	0.985	0.990	0.990	1.000	1.000	1.000	1.000	1.000	1.000
	0.05	0.990	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	0.1	0.990	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	Uniform p_X^*										
	α =**0.01**	0.960	0.990	0.985	0.995	1.000	1.000	1.000	1.000	1.000	1.000
	0.05	0.990	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	0.1	0.995	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

For Case 2, we also investigated the performance on the proposed flexible omnibus test when the distributions of both the latent variable X and the measurement error U are not correctly specified. From Table A.2 in Appendix A.8, the performance of our testing procedure is similar to the previous case, although the performance is less than when using the correct distribution. When the number of strata is between 100 and 400, the values of power in the misspecified normal model of Case 1 (b) are largely higher than those in Case 2. Except for the corresponding situation, however, the type-I error and power between Case 1

and Case 2 are similar to each other. Thus, this supports how our approach can be robust and efficient, regardless of distributions of X and U.

For Case 3, the case of the existence of the varying coefficient, the empirical type-I error rate, and power are calculated. The number of strata based on each varying point are 100, 200, 300, and 400. Since we considered the varying coefficients using 25 values ($V = 25$), the total strata are 2,500, 5,000, 7,500, and 10,000 ($V \times K$) for each setting. The average values of type-I error and power results of each strata are summarized in Table 2.2, which shows that type-I errors are close to zero and powers are close to one, regardless of whether or not the distribution of X is misspecified. By comparing it with Case 1 (b), the type-I error and power becomes smaller and larger than those of Case 1 (b).

Table 2.2: The average value of empirical type I errors and powers of the flexible omnibus test under Case 3 for varying coefficient; The two rows from the top and two rows from the bottom of the table represent the average values of type I error and power based on each nominal level (α) and the number of strata (K); p_X is the correct model and p_X^* is the misspecified model for the latent variable X; For each combination of α and K value, we simulated 200 datasets with 1-4 matched case-crossover study with K strata.

			The values of K			
			100	**200**	**300**	**400**
Type I error		Normal p_X				
	$\boldsymbol{\alpha}$	**0.01**	0.000	0.000	0.000	0.000
		0.05	0.000	0.005	0.005	0.005
		0.1	0.005	0.005	0.005	0.005
		Normal p_X^*				
	$\boldsymbol{\alpha}$	**0.01**	0.005	0.004	0.001	0.002
		0.05	0.006	0.005	0.002	0.004
		0.1	0.01	0.007	0.008	0.006
Power		Normal p_X				
	$\boldsymbol{\alpha}$	**0.01**	0.960	1.000	1.000	0.990
		0.05	0.990	1.000	1.000	1.000
		0.1	0.990	1.000	1.000	1.000
		Normal p_X^*				
	$\boldsymbol{\alpha}$	**0.01**	0.906	1.000	1.000	1.000
		0.05	0.953	1.000	1.000	1.000
		0.1	0.986	1.000	1.000	1.000

For Case 4, the case of existence of the interaction effect between the covariate with measurement error and a fixed covariate, we consider $K = 400$. The average type-I error and power results are summarized in Table 2.3. Under the null model, the empirical type-I error is 0.055, which means that we have high acceptance rate of H_0. On the contrary, under the alternative model, the empirical power is 0.95 and the acceptance rate of H_0 is close to 0.

Table 2.3: The average values of the empirical type I errors and powers obtained from our flexible omnibus test under Case 4; H_0 is the null model $Pr(\boldsymbol{Y}_k = 1|\boldsymbol{Z}_k, \boldsymbol{X}_k, \sum_{j=1}^{M+1} Y_{jk} = 1) = p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k, (\boldsymbol{Z}_{jk} - \boldsymbol{Z}_{1k})\boldsymbol{\beta}_0 + (X_{jk} - X_{1k})\beta_1\}$ and H_{i3} is the alternative model $Pr(\boldsymbol{Y}_k = 1|\boldsymbol{Z}_k, \boldsymbol{X}_k, \sum_{j=1}^{M+1} Y_{jk} = 1) = p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k, (\boldsymbol{Z}_{jk} - \boldsymbol{Z}_{1k})\boldsymbol{\beta}_0 + (X_{jk} - X_{1k})\beta_1 + \gamma_{i3}(h_{i3}(\boldsymbol{Z}_{kj}, X_{kj}) - h_{i3}(\boldsymbol{Z}_{k1}, X_{k1}))\}$; The proportions of acceptance and rejection are computed using 200 simulated datasets under true generating models, respectively; The number of strata (K) of each dataset is 400.

True model	Type I error Proportion of rejection $H_0 : \gamma_i = 0$	Proportion of acceptance $H_0 : \gamma_i = 0$
$H\{\beta_1 Z_{jk} + \beta_2 X_{jk} + q(k, V)\}$	0.055	0.945
$H\{\beta_1 Z_{jk} + \beta_2 X_{jk} + \beta_3 (X_{jk})^2 \sin(Z_{jk})$ $+q(k, V)\}$	0.05 $H_{i3} : \gamma_i \neq 0$ Proportion of rejection	0.950 $H_{i3} : \gamma_i \neq 0$ Proportion of acceptance Power

Finally, the sensitivity analysis for the number of basis functions was conducted as well. For basis functions, we employed Fourier basis functions, $\sin(\bullet)$ and $\cos(\bullet)$. If the number of basis functions is denoted by I, a set of basis functions can be expressed as $\boldsymbol{h}(x) = \{\cos(x), \sin(x), \cos(2x), \sin(2x), \ldots, \cos(\eta x), \sin(\eta x)\}$ where $\eta = \lfloor I/2 \rfloor$. Type-I errors and powers were estimated when I was set to be from 2 to 80 given the number of strata K is 100 and 600. The average type-I error and power are summarized in Tables A.3 and A.4 in Appendix A.8. The performances of our omnibus test are consistent, regardless of the number of basis functions. We also determined that as the number of basis functions increases, the covariance matrices in the corresponding test statistics are often negative definite, especially when the number of strata is small. To overcome this numerical problem, we adjusted the

covariance matrix with diagonal matrix with small control value. The detailed procedures are described in Appendix A.6.

Therefore, our simulation results suggest that our proposed omnibus test has the flexibility to make inferences on various settings of hypotheses. Our flexible omnibus test performs well in terms of type-I error and power.

2.7 1-4 Bidirectional Matched Case-crossover Epidemiology Study

We applied our flexible omnibus test to the example from the incidence of aseptic meningitis against exposure of water turbidity in children under age of 15 from South Korea. We have 1-4 matched case-crossover clustered binary outcomes Y indicating the status of aseptic meningitis.

As described in Section 2.1, our main interests of the aseptic meningitis data are to identify the significance of functional associations between the risk of the meningitis and exposure to water turbidity based on each lag and to determine the potential effect modification of the water turbidity by lag. Furthermore, because meteorological factors can be influential factors to the onset of meningitis [1], we also investigated the existence of interaction between the water turbidity and the air temperature at the measurement over the risk of the meningitis. We considered the exposure to water turbidity as X and the air temperature as Z in degrees Celsius, respectively. In consideration with the measurement error in covariate X, we treated multiple datasets collected in consecutive days as the replicates. Thus, there were total 29 lag days with three replicates.

Based on the prior analysis using the conditional logistic regression, there was significant evidence of a linear relationship between the exposure to the water turbidity and the risk of

meningitis [22]. We performed our flexible omnibus test through the baseline model including the linear effect of the water turbidity on the incidence of the meningitis.

First, we conducted the flexible omnibus test for identifying the functional relationship between the exposure to water turbidity and the risk of the meningitis. By denoting d as the order of the day when the datasets are collected, we displayed the empirical p-values of testing functional association on the corresponding d lag days in Figure 2.2, $d = 1, \ldots, 29$. The first lag of 4 days had insignificant nonlinear relationship between the water turbidity and the meningitis. However, there was significant nonlinear relationship between the covariate and the risk of meningitis in the second lag of 12 days (from $d = 5$ to $d = 16$). For last lag of 13 days between days 17 to 29, the functional association became linear again. This result suggests that there is significant evidence that the critical change of the pattern in the risk of meningitis occurs as the water turbidity increases at $d = 5$ and $d = 17$.

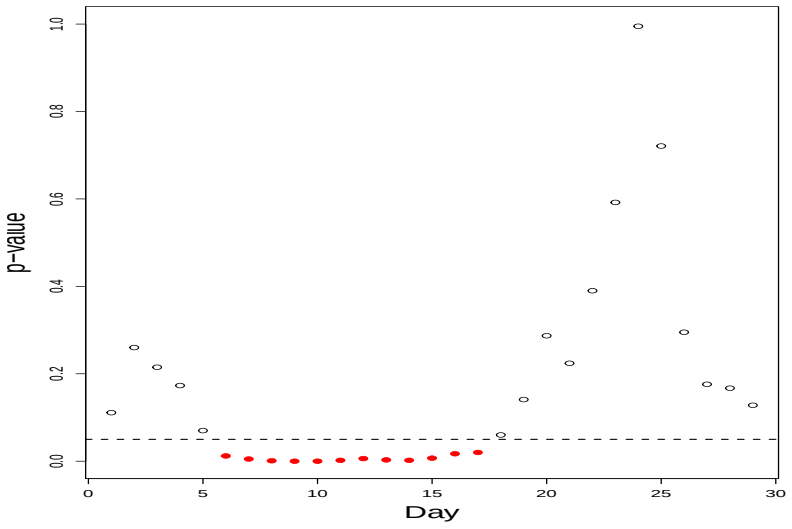


Figure 2.2: Scatter plot of lag day (d) and the empirical p-values which are obtained from the flexible omnibus test whether there is a nonlinear relationship between the water turbidity and the risk of meningitis; Total 29 days are considered by combining 3 consecutive days as the replicates for the measurement error; The solid circle represents the significance nonlinear relationship between the water turbidity and the meningitis; The empty circle represents the non-significance one; The dashed line represents the horizontal line at 0.05.

As an alternative approach to further evaluate whether the testing results are consistent with nonparametric estimation, we estimated the nonparametric function $m(x)$ using regression splines. Detailed estimating procedure is described in Appendix A.7.1. In addition to the estimation, the first derivative of the estimated function except the linear effect is also computed based on each day to identify the significance of nonlinearity. Figure 2.3 describes the estimated functions and its first derivative for each $d = 2$ and 10. Figure A.1 in Appendix A.8 shows the estimated functions and the first derivatives for $d = 17$ and 24. We also display 95% bootstrap confidence bands along with the estimated first derivative function of $m(x)$. The estimated functions on $d = 2$ and 24 seem to be linear, which is consistent with our test results. The corresponding confidence bands also contain zero against whole range of the water turbidity. That is, the nonlinear relationship between the covariate and outcome was not significant on the corresponding day. On the contrary, the estimated functions on $d = 10$ and 17 have nonlinearity. Their confidence bands do not include zero, so there are significant nonlinear relationships on these days as well. This result indicates that the risk of meningitis is slowly decreasing as water turbidity increases through $d = 4$, and then rapidly and nonparametrically changing at $d = 5$. Finally, the risk also slowly increase from $d = 17$. Hence, the testing results are consistent with nonparametric estimation.

For testing the existence of the effect modification of the water turbidity based on lag day, we obtained the empirical p-value of our omnibus test. There is a strong, significant evidence of the existence of p-value as 0, suggesting that the effect of water turbidity over the incidence of aseptic meningitis changes over lag day. Hence, there is a statistical evidence that the lag day is an effect modifier. In addition, $\beta(d)$ is estimated using regression splines. Detailed procedure is explained in Appendix A.7.2. Figure 2.4 displays the estimated varying-coefficient $\hat{\beta}(d)$ and surface plots for the estimated mean function $\hat{m}(x, d)$. The 95% credible interval for $\hat{\beta}(d)$ is also provided in Figure 2.4. The estimated coefficient follows a linear pattern as the day increases, suggesting the consistent result of the omnibus

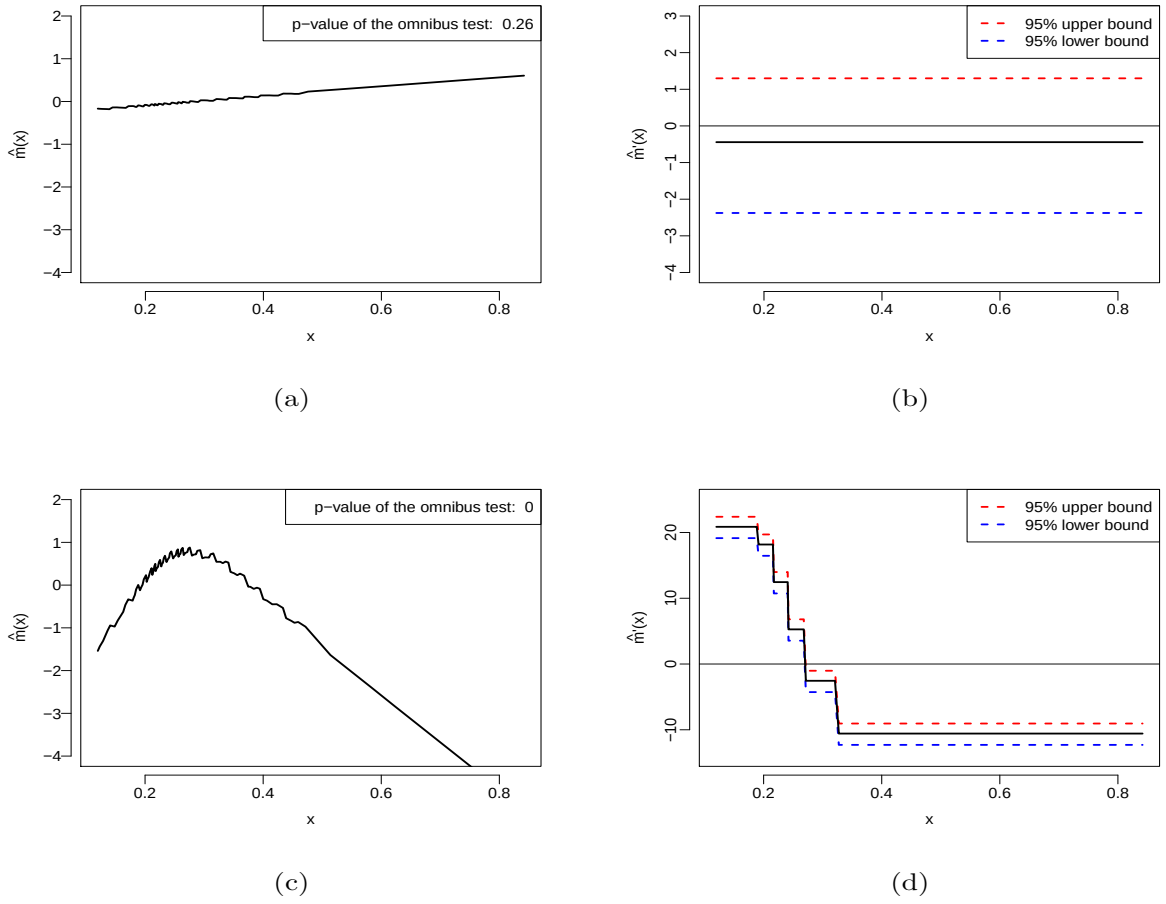
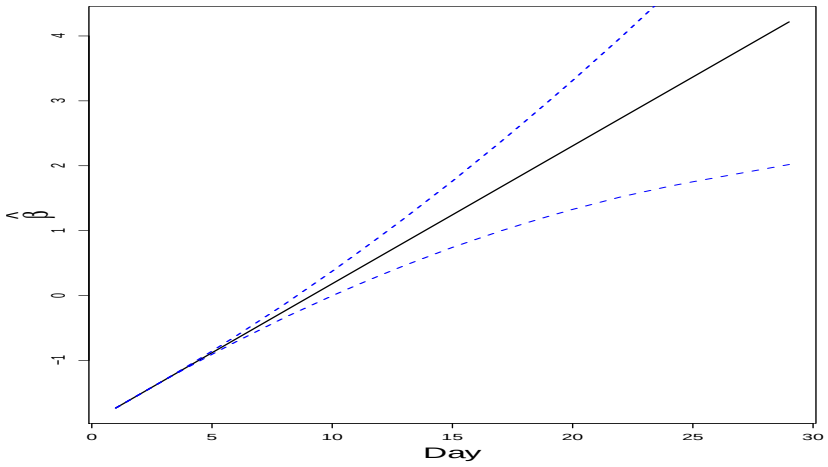


Figure 2.3: Plots of nonparametric estimation for true mean functions $m(x)$ on the 2nd and 10th days; (a) Plots of the estimated mean function $\hat{m}(x)$ with p-value of the flexible omnibus test for nonlinear association on the 2nd day. (b) Plots of the first derivative function of $\hat{m}(x)$ with 95% bootstrap confidence bands (dashed lines) on the 2nd day. (c) Plots of the estimated mean function $\hat{m}(x)$ with p-value of the flexible omnibus test for nonlinear association on the 10th day. (d) Plots of the first derivative function of $\hat{m}(x)$ with 95% bootstrap confidence bands (dashed lines) on the 10th day.

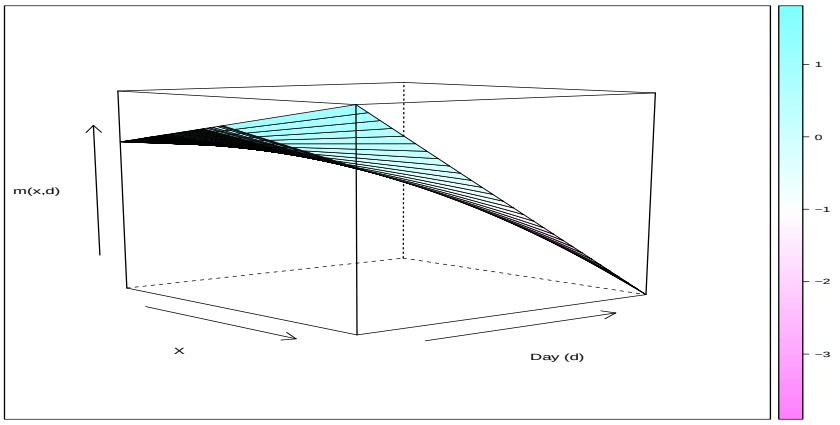
test.

Finally, we also performed a flexible omnibus test to examine the significance of the interaction effect between the water turbidity and the temperature on the risk of the meningitis. The summary of p-values is displayed in Figure 2.5, which suggests that the interaction effect is significant on most d. The incidence of the aseptic meningitis changes the effect of water

turbidity coupled with change of temperature. This means that the temperature can be an effect modifier as well. Hence, it suggests the potential for predicting meningitis cases up to a temperature change to aide decision makers.



(a)



(b)

Figure 2.4: Plots of the estimated time-varying coefficient $\beta(d)$; (a) Plots of the estimated varying-coefficient $\hat{\beta}(d)$ (solid line) with 95% credible intervals (dashed lines); (b) Surface plots of $(x, d, \hat{m}(x, d))$, where x is water turbidity, d is lag day, and $\hat{m}(x, d)$ is inversely proportional to the risk of the meningitis

Therefore, our omnibus test provided several important and unique features in the clustered 1-4 matched study on the incidence of the aseptic meningitis: (1) lag day within the latent period is an important factor for investigating the significant nonlinear relationship between the incidence of the meningitis and exposure to water turbidity; consequently, the lag day was identified as the significant effect modifier; in addition, (3) because of the significance of the interaction effect between the exposure to water turbidity and temperature, the temperature was also identified as the effect modifier. These features indicate that lag day, water turbidity, and temperature are needed for scrutinizing the risk of the meningitis. These features cannot be identified from the existing testing procedure.

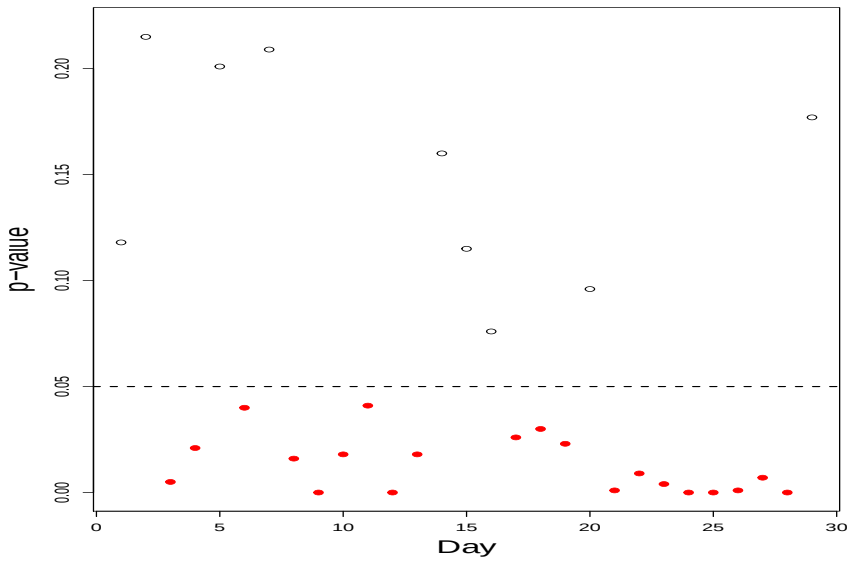


Figure 2.5: Scatter plot between lag day and the empirical p-values which are obtained from the flexible omnibus test whether there is an interaction effect between the water turbidity and the temperature over the aseptic meningitis; The solid circle represents the significance nonlinear relationship between the water turbidity and the meningitis; The empty circle represents the non-significance one; The dashed line represents the horizontal line at 0.05.

2.8 Discussion

In this chapter, we proposed a semiparametric and flexible omnibus test for the purpose of testing of the significance of functional relationship between the clustered binary outcomes and covariates with the general measurement error framework by taking into account effect modification of matching covariates. Our approach was developed using a semiparametric framework in which an explicit likelihood function is not available. We proposed an efficient score, which serves as a local test statistic associated with estimating equations, to avoid likelihood derivations, when unavailable. Our proposed omnibus test has the following features: (a) it is applicable for clustered binary outcomes; (b) it can be useful for testing whether or not there is effect modification by matching covariates; (c) it does not require specific alternative hypotheses so that flexible inference is available; (d) it does not require the estimation of the parameters on the alternative model so that it can reduce the burden of the computation; (e) it does not require the distributional assumption of the measurement error. To the best of our knowledge, no current test statistics have all of these features.

Our simulation results suggest that our proposed omnibus test has flexibility for making inferences on various settings of hypotheses. Our approach is both robust and efficient, regardless of distributions of measurement error and the size of stratum. Our flexible omnibus test performed well in terms of type-I error and power. We also have demonstrated the advantage of our approach using our 1-4 matched case-crossover study, which provides significant evidences of the departure of nonlinearity on lag day, existence of effect modification of lag day, and irregular interaction between the water turbidity and the temperature on the risk of the meningitis.

Further research is still needed to examine the theoretical properties in detail. Deriving theoretical distributions of our omnibus test statistic will be useful for reducing further computation burden. Although we found that our method can detect well the existence of

nonlinearity or small difference between two functions, it is difficult to derive the bound of this difference in a general case. It will also be important to develop multiple comparison methods using our approach, which may be a challenging problem because of unknown dependence structures among data.

Chapter 3

Joint Semiparametric Kernel Machine Network Regression

3.1 Introduction

For the past few decades, numerous statistical methods have been developed for analyzing high-dimensional data. Variable selection and graphical modeling play critical roles in detecting important components among many variables, including noises and estimating connections of the components. For example, in the application using variable selection and graphical modeling, they are commonly used in genomics to figure out important gene-gene interactions in high-dimensional data.

However, most of the variable selection approaches in high-dimensional data are a quite limited because their approaches were developed under the additive models and componentwise selections. For instance, sparse additive models (SpAM) proposed by Ravikumar et al. [47] imposes a sparsity constraint on the additive and nonparametric setting. Lin and Zhang [31] have proposed Component selection and smoothing operator (COSSO), which enables componentwise variable section with regularization on the component functions belonging to

a reproducing kernel Hilbert space (RKHS). Radchenko and James [45] presented variable selection using an adaptive nonlinear interaction structure in high dimensions (VANISH) to take account for specific forms of interactions and nonlinearities. However, when the number of input variables is very large and their interactions are complicated, modeling each interaction term is extremely expensive and these component function approaches may not be efficient. Thus, variable selection in non-additive and nonparametric regression with high dimensional variables has been challenging.

To overcome the problem regarding the unknown functional association and complicated interactions, the application of some kernel functions was considered in the regression setting. A kernel function can take account of unknown functional relationship in the data without explicitly searching for a high-dimensional space by making linearly inseparable data separable by mapping the data points onto the high-dimensional feature space through inner products. Maity and Lin [36] developed a flexible testing under the powerful kernel machine framework to test the joint effects of individual genes on a continuous outcome. Wang et al. [53] extended their idea for multiple genes and multiple environmental interactions. However, these approaches were still challenging in dealing with the high-dimensional data because they conducted modeling on element-wise interaction of variables and environmental elements. To solve the complication in modeling interactions in high-dimensional case, Fang et al. [14] developed a flexible variables selection methods for non-additive nonparametric models. They modeled the smoothing function by a Least Squares Kernel Machine (LSKM) and constructed the nonnegative garrote objective function. However, Fang et al. [14]'s approach can not fully incorporate the dependence structure among variables into the variable selections. Thus, there are quite limited methods to simultaneously take account for the variable selection with unknown functional form and modeling unknown dependence structure among high correlated high-dimensional variables.

On the other hand, Gaussian graphical model is an alternative approach to study such

dependence among the components of our interest. The method is able to investigate the conditional dependence structure between variables by estimating sparse precision matrices. For a given class of interest such as disease status, the estimated precision matrices can be mapped into networks for visualization. Friedman et al. [17] developed the graphical lasso (Glasso) algorithm based on a clockwise coordinate descent approach. Guo et al. [19] proposed jointly estimate precision matrices for different classes by re-parameterizing off-diagonal elements of precision matrices to be a multiplication of a common factor across classes and a unique factor for each class. Their method could be solved by the iterative weighted Glasso [17]. On the other hand, Danaher et al. [11] used generalized fused lasso or group lasso as the penalty, and employed the alternating directions method of multipliers (ADMM) algorithm to solve the optimization problem. Nevertheless, both Guo et al. [19] and Danaher et al. [11] are still limited to investigate a single level conditional dependence when there is multi-level data structure. Cheng et al. [9] developed a multi-level graphical modeling approach. However, the main limitation of these existing Gaussian graphical models is that it is only applicable for discretized response variables in the case $p\log(p) \ll n$, where p is the number of variables and n is the sample size. Thus, when the number of variables is huge including noise variables, it is needed to develop a joint method between variable selection and graphical model without loss of the dependence among the variables. To the best of our knowledge, the methods for simultaneously conducting variable selection as well as estimating networks among variables under the semiparametric regression settings are quite limited.

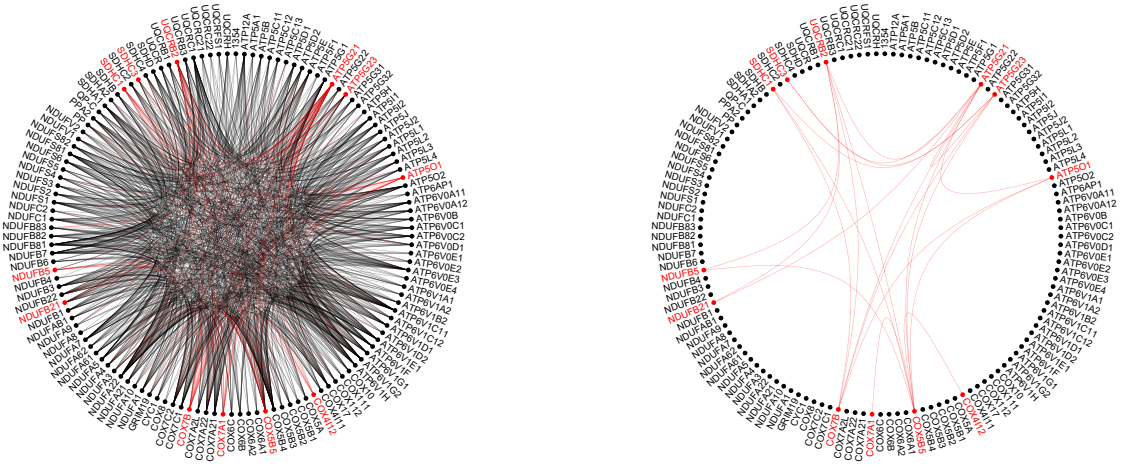
In this chapter, we develop a joint semiparametric kernel machine network approach, which enables the simultaneous run of the variable selection and the network estimation by providing a connection between them. Our approach is a unified and integrated method which can simultaneously identify important variables and build a network among the variables in the high-dimensional case. We develop our approach under a semiparametric kernel

machine regression framework, which can allow for the possibility that each variable might be nonlinear and possibly interact with each other in intricate form. The advantages of our proposed method are that it can (1) simultaneously perform variable selection and build a network among high-correlated and high-dimensional variables under a regression setting; (2) automatically do modeling unknown and complicated interactions among the variables and estimate networks among these variables; (3) have the flexibility for any semiparametric model including non-additive and nonparametric model; (4) provide interpretable network by selecting or combining the important variables.

We illustrate the performance of our proposed method with a toy example using a gene pathway based on the type-II diabetes gene-pathway data for our application. The pathway consists of 133 genes and some of the genes are strongly dependent on each other. We apply the graphical lasso and our method to the gene pathway in order to estimate the network structure. Figure 3.1 shows the results of the graphical model estimates based on graphical lasso and our proposed method. We see that our method results in much sparse network based on a variable selection, compared to the graphical lasso's. In addition, our method can incorporate the estimated network into the joint estimation of a continuous clinical outcome. Hence, we expect that our method will provide high interpretability on the network. We will discuss the details through simulation studies and application.

The rest of this chapter is organized as follows. In Section 3.2, we introduce our joint semiparametric kernel network regression model and describe how we jointly model semiparametric regression and Gaussian graphical model altogether. In Section 3.3, we develop the joint estimation method to achieve both variable selection and network estimation. In Section 3.4, we propose a joint algorithm using two second-order approximated algorithms for the variable selection and a block-coordinate descent algorithm for the network estimation. In Section 3.5, we illustrate the performance of our approach through several simulation studies. In Section 3.6, we apply the proposed method to real datasets on genetic pathway

analysis. Lastly, our concluding remarks are presented in Section 3.7.



(a) Global gene network of Pathway 229 (b) Selected gene network of Pathway 229

Figure 3.1: Comparison of gene networks (Pathway 229) based on Graphical Lasso (left) and Joint Semiparametric Kernel Machine Network (right)

3.2 Joint Semiparametric Kernel Machine Network Regression

In this section, we propose a joint semiparametric regression model for simultaneously conducting variable selection as well as estimating networks among variables. We first explain how we jointly connect semiparametric regression model and Gaussian graphical model in Section 3.2.1 and introduce a joint kernel machine network in Section 3.2.2.

3.2.1 Joint Modeling

We first define some notations. Let $\mathbf{y} = (y_1, ..., y_n) \in \mathbb{R}^n$ denote a vector of continuous clinical outcomes. Define $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]^T$ by a $n \times m$ matrix of clinical covariates, e.g., age and gender, which consists of m-vectors, x_i's with $i = 1, \ldots, n$. Let $\mathbf{Z}$ be a $n \times p$ matrix of predictors such as gene expression variables where $\mathbf{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_n]^T$ and $\boldsymbol{z}_i = [z_{i1}, \ldots, z_{ip}]^T$ is a $p \times 1$ predictor vector for the ith observation. Let p^* denote the number of important predictors, where $p^* \leq p$. We consider that $\mathbf{y}$ depends on $\boldsymbol{X}$ parametrically and $\boldsymbol{Z}$ nonparametrically via an unknown smooth function $\boldsymbol{h}(\boldsymbol{Z})$, which lies in the Reproducing Kernel Hilbert Spaces (RKHS). We also further consider $\mathbf{z}_i$ follows the multivariate normal distribution for estimating networks among the variables.

Given these notations and settings, we consider two models. One is a semiparametric regression model,

$$\boldsymbol{y}|\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{h}(\boldsymbol{Z}) + \boldsymbol{\epsilon}, \tag{3.1}$$

where $\boldsymbol{\epsilon} \sim (\mathbf{0}, \sigma_\epsilon^2 I)$, and the other is a Gaussian graphical model,

$$\boldsymbol{z}_i \overset{i.i.d}{\sim} MVN(\mathbf{0}, \Omega^{-1}) \in \mathbb{R}^p, \quad i = 1, ..., n \tag{3.2}$$

where $\boldsymbol{\beta}$ is a $m \times 1$ vector of regression coefficients, $\boldsymbol{\epsilon}$ is a random error with mean zero and variance σ_ϵ^2, and Ω is a $p \times p$ precision matrix of $\boldsymbol{z}$.

Here, $h(\boldsymbol{z})$ can be modeled by connecting kernel machine learning with the mixed model due to the high-dimensional space of $\boldsymbol{Z}$ and complicated interaction among $\boldsymbol{z}$s. According to Mercer's theorem [10], a kernel function $K(\cdot, \cdot)$ is spanned by a particular set of orthogonal basis functions, $\{\phi_b(\boldsymbol{z})\}_{b=1}^B$. Since the unknown function $h(\cdot)$ can be expressed using a set of the basis functions, a kernel function $K(\cdot, \cdot) \in \mathcal{H}_K$ can also be used to represent the nonparametric function $h(\boldsymbol{z})$ where $h(\boldsymbol{z}) = \sum_{i=1}^n \alpha_i K(\boldsymbol{z}_i, \boldsymbol{z})$, where α_i is a coefficient of

kernel function, by the Representer Theorem. Under the mixed model framework, $h(\boldsymbol{Z})$ can be considered as the following Gaussian process,

$$\boldsymbol{h}(\boldsymbol{Z}) = \begin{pmatrix} h_1(\boldsymbol{Z}) \\ h_2(\boldsymbol{Z}) \\ \vdots \\ h_n(\boldsymbol{Z}) \end{pmatrix} \overset{i.i.d}{\sim} MVN\big[\mathbf{0}, \tau K_{n\times n}(\boldsymbol{Z}|\Omega)\big], \tag{3.3}$$

where τ is an unknown positive parameter. Based on the representation of $\boldsymbol{h}(\boldsymbol{z})$ using the kernel function, we can rewrite semiparametric regression model (3.1) as

$$\boldsymbol{y}|\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{\beta} + K(\boldsymbol{Z}|\Omega)\boldsymbol{\alpha} + \boldsymbol{\epsilon}, \qquad K(\cdot) \in \mathcal{H}_K, \tag{3.4}$$

where $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_n)^T$ is a vector of coefficients of the kernel function which can be estimated by solving the least squares kernel machine.

Finally, by jointly modeling 3.1-3.3 all together, our joint semiparametric model can be expressed as

$$\begin{aligned} \boldsymbol{y}|\boldsymbol{Z} &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{h}(\boldsymbol{Z}) + \boldsymbol{\epsilon}, \\ \boldsymbol{h}(\boldsymbol{Z}) &\sim MVN\big[\mathbf{0}, \tau K_{n\times n}(\boldsymbol{Z}|\Omega)\big], \\ \boldsymbol{z}_i &\overset{i.i.d}{\sim} MVN(\mathbf{0}, \Omega^{-1}) \in \mathbb{R}^p, \quad i = 1, ..., n \end{aligned} \tag{3.5}$$

where $\boldsymbol{\epsilon} \sim (\mathbf{0}, \sigma_\epsilon^2 I)$. We refer this joint model as "joint semiparametric kernel machine network regression $(JSKNR)$".

3.2.2 Joint Semiparametric Kernel Machine Network

In the joint model (3.5), the coefficient, $\boldsymbol{\alpha}$ can be estimated by minimizing the least squares error with regularized norm $||\boldsymbol{h}||^2_{\mathcal{H}_K} = \boldsymbol{\alpha}^T K \boldsymbol{\alpha}$ such that

$$\frac{1}{2}||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - K\boldsymbol{\alpha}||^2 + \frac{1}{2}\lambda_0 \boldsymbol{\alpha}^T K \boldsymbol{\alpha}, \qquad K(\cdot) \in \mathcal{H}_K. \tag{3.6}$$

The closed forms of estimates are

$$\widehat{\boldsymbol{\alpha}} = (\lambda_0 I + K)^{-1}(\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}) \tag{3.7}$$

and

$$\widehat{\boldsymbol{\beta}} = \left\{ \boldsymbol{X}^T (I + \lambda_0^{-1} K)^{-1} \boldsymbol{X} \right\}^{-1} \boldsymbol{X}^T (I + \lambda_0^{-1} K)^{-1} \boldsymbol{y},$$

where λ_0 is a positive smoothing parameter that controls the degrees of a smoothing curve or surface in a high-dimensional setting.

For estimating the precision matrix Ω, we first explain the kernel function $K(\cdot, \cdot)$, and then describe how we model Ω within the kernel matrix K. K is a kernel matrix corresponding to the functional space $\mathcal{H}_K$, generated by symmetric and positive semi-definite kernel function, $k(\cdot, \cdot)$. K is also known as a "Gram matrix" of the kernel function. Two popular kernels are linear polynomial kernel and Gaussian kernel (denoted by PK and GK, respectively), which are further explained as follows.

- Polynomial kernel: $K(\boldsymbol{Z}|\rho) = \sum_{j=1}^{p} \rho S^j$, where ρ is a nonnegative scale parameter and S^j denotes a similarity matrix, which is calculated using the dot product. In polynomial kernel, the (k, l)th element of S^j is $s^j_{kl} = z_{lj} z_{kj}$, $1 \leq k, l \leq n$; the larger the product, the larger similarity between the two observations (kth and lth observations).

The polynomial kernel is one of the representative kernels for a nonlinear and additive form of p predictors.

- Gaussian kernel: $K(\boldsymbol{Z}|\rho) = \exp\left(\frac{-\sum_{j=1}^{p} S^j}{\rho}\right)$, where ρ is a nonnegative scale parameter and S^j denotes a similarity matrix, which is obtained from the Euclidean distance. In Gaussian kernel, the (k,l)th element of S^j is $s^j_{kl} = (z_{kj} - z_{lj})^2$, $1 \leq k, l \leq n$; the closer the distance comes to zero, the larger similarity between the two observations (kth and lth observations). The Gaussian kernel is a representative kernel for a nonlinear and non-additive form of p predictors.

By introducing nonnegative garrote scale parameter, $\boldsymbol{\xi} = (\xi_1, ..., \xi_p)$, into the kernel machine framework, the kernel function can be represented as

$$K(\boldsymbol{Z}|\boldsymbol{\xi}) = g\Big\{\sum_{j=1}^{p} \xi_j(\rho) S^j\Big\}, \tag{3.8}$$

where $g(\cdot)$ is an element-wise function of matrix entry and $\xi_j(\cdot) \geq 0$ with $j = 1, ..., p$.

For instance, in the linear polynomial kernel, $\xi_j(\rho) = \rho$ and $g(u) = u$ (the identity function). In the Gaussian kernel, $\xi_j(\rho) = (\rho c)^{-1}$ and $g(u) = \exp(u)$, where c is a positive constant that scales each similarity distance. Based on the average Euclidean distance between two observations into the kernel, $E[||\boldsymbol{z}_k - \boldsymbol{z}_l||^2]$, we set c as sample mean of all distance entries where $\bar{S} = \frac{2}{c} \times \frac{2}{n(n+1)} \sum_{1 \leq l,k, \leq n} \sum_{j=1}^{p} -||z_{kj} - z_{lj}||^2$. Here, c can be viewed as a smoothing tuning parameter. If c is not fixed, variable selection of ξ possibly contains the information of c. Hence, we may not distinguish whether variables are selected via the variable selection mechanism or they are selected by changing of c; thus, it is appropriate to conduct variable selection after fixing c.

In expressing a $p \times p$ matrix, $\Omega(\boldsymbol{\xi})$ with the diagonal terms $(\xi_1, ..., \xi_p)$ and the off-diagonal

terms $\mathbf{0}$ such as

$$\Omega(\boldsymbol{\xi}) = \begin{pmatrix} \xi_1 & 0 & \cdots & 0 \\ 0 & \xi_2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \xi_p \end{pmatrix},$$

the equation (3.8) can be rewritten as

$$K(\boldsymbol{Z})_{k,l} = \begin{cases} \sum_{j=1}^{p} \xi_j z_{jk} z_{jl} \equiv \boldsymbol{z}_k \Omega(\boldsymbol{\xi}) \boldsymbol{z}_l^T, \qquad 1 \leq k, l \leq n, & if\ Polynomial\ kernel(PK) \\ \exp\left\{ -\sum_{j=1}^{p} \xi_j \frac{(z_{kj} - z_{lj})^2}{c} \right\} \equiv \exp\left\{ -\boldsymbol{z}_{k,l}^* \Omega(\boldsymbol{\xi}) \boldsymbol{z}_{k,l}^{*T} \right\} & if\ Gaussian\ kernel(GK) \end{cases}$$

where $\boldsymbol{z}_{k,l}^* = [(z_{k1} - z_{l1}), ..., (z_{kp} - z_{lp})]^T$ and $\Omega(\boldsymbol{\xi})$ is a $p \times p$ diagonal matrix for the distance vector $\boldsymbol{z}^*$ with $\boldsymbol{\xi} = (\xi_1, ..., \xi_p)$. Thus, we can view that under GK, each $\mathbf{z}$ follows a multivariate normal distribution with mean $\mathbf{0}$ and precision matrix $\Omega(\boldsymbol{\xi})$. However, in this form of $\Omega(\boldsymbol{\xi})$, the possible dependence among $\mathbf{z}$s is ignored.

Hence, we incorporate possible but unknown dependence structure into the precision matrix $\Omega(\boldsymbol{\xi})$ such as

$$\Omega(\boldsymbol{\xi}) = \begin{pmatrix} \xi_1 & \xi_{12} & \cdots & \xi_{1p} \\ \xi_{21} & \xi_2 & \xi_{jj'} & \vdots \\ \vdots & \xi_{j'j} & \ddots & \xi_{p-1,p} \\ \xi_{p1} & \cdots & \xi_{p,p-1} & \xi_p \end{pmatrix}, \qquad 1 \leq j \neq j' \leq p. \tag{3.9}$$

Using this precision matrix, we can simultaneously conduct variable selection using the diagonal nonnegative garrote, given fixed off-diagonal values and estimate network among selected variables using the off-diagonal entries. Thus, the precision matrix plays an important role in selecting significant variables and building a statistical network among the predictors.

Based on the proposed precision matrix (3.9), the kernel matrix for the linear PK can be rewritten as

$$K(\boldsymbol{Z})_{k,l} = \sum_{j=1}^{p} \xi_j z_{jk} z_{jl} + 2 \sum_{j'>j} \xi_{jj'} z_{jk} z_{j'l} \equiv \boldsymbol{z}_k \Omega(\boldsymbol{\xi}) \boldsymbol{z}_l^T, \qquad 1 \leq k, l \leq n$$

where the (k, l)th entry of the kernel matrix. In the case of GK, the (k, l)th element of Gaussian kernel matrix is then

$$K(\boldsymbol{Z})_{k,l} = \exp\left[-\frac{1}{c} \left\{ \sum_{j=1}^{p} \xi_j (z_{kj} - z_{lj})^2 + 2 \sum_{j'>j} \xi_{jj'} (z_{kj} - z_{lj'})^2 \right\} \right] \equiv \exp\left\{ -\boldsymbol{z}_{k,l}^* \Omega(\boldsymbol{\xi}) \boldsymbol{z}_{k,l}^{*T} \right\}, \quad 1 \leq k, l \leq n.$$

In general, we can express a kernel matrix with the componentwise function $g(\cdot)$ as

$$K(\boldsymbol{\xi}, \boldsymbol{Z}) = g\left(\sum_{j=1}^{p} \xi_j(\rho) S^j + 2 \sum_{j'>j} \xi_{jj'}(\rho) S^{jj'} \right). \tag{3.10}$$

By incorporating a kernel matrix (3.10) into our joint semiparametric kernel regression (3.5), we develop our joint method to achieve the dual tasks, variable selection and network estimation. Through the implementation of the dual tasks by our method, it can give flexibility and interpretability to take account of the significance of the network over the outcomes as well as estimation on the dependence structure among the predictors.

3.3 Joint Estimation

Under the model (3.5), we consider the joint pseudo-likelihood function

$$L\{\Omega(\boldsymbol{\xi}), \boldsymbol{\beta} | \boldsymbol{y}, Z\} = f_{y|Z}(\boldsymbol{y}|Z) \cdot f_Z(Z),$$

where $f_{y|Z}(\cdot)$ and $f_Z(\cdot)$ represent the probability density of y given Z and the probability density of Z. Considering that there are a large number of variables (p), we impose an

l_1-constraint on $\Omega(\boldsymbol{\xi})$, that is, $||\Omega(\boldsymbol{\xi})||_1 < \boldsymbol{t}$. It means that we have two constraints such that $\sum_{j=1}^{p} \xi_j < t_1$ and $\sum_{j \neq j'} |\xi_{jj'}|_1 < t_2$ for the diagonals and the off-diagonals, respectively. Hence, we can identify a smaller subset of the variables and a sparse network among them.

Let $\boldsymbol{\xi}_D = (\xi_1, ..., \xi_p)$ and $\boldsymbol{\xi}_{OD} = (\xi_{12}, ..., \xi_{jj'}, ..., \xi_{p-1,p})$ denote a vector of the diagonals and the off-diagonals of $\Omega(\boldsymbol{\xi})$, respectively, that is, $\boldsymbol{\xi} = (\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD})$. Using the joint pseudo-likelihood function and the l_1-constraints of $\Omega(\boldsymbol{\xi})$, we estimate $\boldsymbol{\xi}$ by solving the following optimization problem,

$$\begin{aligned}(\hat{\boldsymbol{\xi}}, \widehat{\Omega}) &= \underset{||\Omega(\boldsymbol{\xi})||_1 < \boldsymbol{t}}{\operatorname{argmin}} \Big[-\log f_{\boldsymbol{y}|\boldsymbol{Z}}\{\boldsymbol{y}|\boldsymbol{Z}, \Omega(\boldsymbol{\xi})\} - \log f_{\boldsymbol{Z}}\{\boldsymbol{Z}|\Omega(\boldsymbol{\xi})\} \Big] \\ &= \underset{\sum_{j=1}^{p} |\xi_j|_1 < t_1, \sum_{j \neq j'} |\xi_{jj'}|_1 < t_2}{\operatorname{argmin}} \Big[-\log f_{\boldsymbol{y}|\boldsymbol{Z}}(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) - \log f_{\boldsymbol{Z}}(\boldsymbol{Z}|\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) \Big],\end{aligned} \tag{3.11}$$

with $1 \leq j \neq j' \leq p$.

Define an objective function $Q(\bullet)$ as

$$\begin{aligned}Q\{\Omega(\boldsymbol{\xi})\} = Q(\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) = &- \log f_{\boldsymbol{y}|\boldsymbol{Z}}(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) - \log f_{\boldsymbol{Z}}(\boldsymbol{Z}|\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) \\ &+ \lambda_1 \sum_{j=1}^{p} \xi_j + \lambda_2 \sum_{j \neq j'} |\xi_{jj'}|,\end{aligned} \tag{3.12}$$

which is equivalent to the optimization problem (3.11). The two constraints of the optimization problem possibly lead to a singularity problem in estimating the precision matrix because of zero diagonal entries corresponding with the noise variables from the variable selection. Since our goal is to estimate the precision matrix to be spare and also solve this singularity issue at the same time, we incorporate the idea of the backward elimination to estimate our precision matrix. Then, we have the following optimization problem and the

objective function:

$$
\begin{aligned}
(\hat{\boldsymbol{\xi}}, \widehat{\Omega}) &= \underset{||\Omega(\boldsymbol{\xi})||_1 < \boldsymbol{t}}{\operatorname{argmin}} \Big[-\log f_{\boldsymbol{y}|\boldsymbol{Z}}\{\boldsymbol{y}|\boldsymbol{Z}, \Omega(\boldsymbol{\xi})\} - \log f_{\boldsymbol{Z}}\{\boldsymbol{Z}|\Omega(\boldsymbol{\xi})\} \Big] \\
&= \underset{\sum_{j=1}^{p}|\xi_j|_1 < t_1, \sum_{(j,j')\in\mathcal{J}}|\xi_{jj'}|_1 < t_2}{\operatorname{argmin}} \Big[-\log f_{\boldsymbol{y}|\boldsymbol{Z}}(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) - \log f_{\boldsymbol{Z}}(\boldsymbol{Z}|\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) \Big],
\end{aligned} \tag{3.13}
$$

and

$$
\begin{aligned}
Q\{\Omega(\boldsymbol{\xi})\} = Q(\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) = &-\log f_{\boldsymbol{y}|\boldsymbol{Z}}(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) - \log f_{\boldsymbol{Z}}(\boldsymbol{Z}|\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD}) \\
&+ \lambda_1 \sum_{j=1}^{p} \xi_j + \lambda_2 \sum_{(j,j')\in\mathcal{J}} |\xi_{jj'}|,
\end{aligned} \tag{3.14}
$$

where $\mathcal{J} = \{(j, j') | 1 \leq j \neq j' \leq p, \text{ given } I(|\hat{\xi}_j| < t_1) \neq 0\}$. The solutions of $(\boldsymbol{\xi}_D, \boldsymbol{\xi}_{OD})$ can be obtained by minimizing the objective function (3.14). Note that λ_1 and λ_2 are tuning parameters to impose the sparsity for selecting important variables and building network structure, respectively.

Connecting the objective function (3.14) to the kernel machine least square function (3.6), the objective function $Q(\bullet)$ can be expressed as

$$
\begin{aligned}
Q\{K(\boldsymbol{Z}|\Omega(\boldsymbol{\xi})), \boldsymbol{\alpha}\} = &\Big[\frac{1}{2}||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - K(\boldsymbol{Z}|\Omega(\boldsymbol{\xi}))\boldsymbol{\alpha}||^2 + \frac{\lambda_0}{2}\boldsymbol{\alpha}^T K(\boldsymbol{Z}|\Omega(\boldsymbol{\xi}))\boldsymbol{\alpha}\Big] \\
&+ \Big[-\log\det\{\Omega(\boldsymbol{\xi})\} + \operatorname{tr}(S\Omega(\boldsymbol{\xi}))\Big] + \lambda_1||\boldsymbol{\xi}_D||_1 + \lambda_2||\boldsymbol{\xi}^*_{OD}||_1,
\end{aligned} \tag{3.15}
$$

where $\boldsymbol{\xi}^*_{OD} \in \{\xi_{j,j'} | (j, j') \in \mathcal{J}\}$, S is a sample covariance matrix of $\boldsymbol{Z}$, and tr denotes trace of the corresponding matrix. Based on the kernel machine framework, the solution

$\hat{\boldsymbol{\xi}} = (\hat{\boldsymbol{\xi}}_D, \hat{\boldsymbol{\xi}}_{OD})$ of the optimization problem (3.15) is considered as

$$\begin{aligned} &\underset{K}{\operatorname{argmin}}\, Q\{K(\boldsymbol{Z}|\Omega(\boldsymbol{\xi}))\}, \\ &\text{subject to} \quad K \in \mathbb{K}^* = \left\{ K(\boldsymbol{Z}|\Omega(\boldsymbol{\xi})) : \boldsymbol{\xi} \in \mathbb{R}_+^p, \sum_{j=1}^{p} |\xi_j|_1 \leq t_1, \text{ and } \sum_{(j,j')\in\mathcal{J}} |\xi_{jj'}|_1 \leq t_2 \right\}, \end{aligned} \tag{3.16}$$

where $\mathbb{R}_+^p$ is a p-dimensional nonnegative space of real numbers.

3.4 Joint Method

It is challenging to jointly solve the objective function (3.15) for diagonal and off-diagonal parameters in one step. In this section, we develop a joint method to simultaneously conduct variable selection via the approximated coordinate algorithm for diagonals and estimate sparse network via a block-coordinate decent algorithm for off-diagonals.

3.4.1 Approximated Coordinate Algorithm to Update $\boldsymbol{\xi}_D$

First, we develop an approximated coordinate algorithm to estimate the diagonal nonnegative garrote $\boldsymbol{\xi}_D$ given initial off-diagonals. Let $\boldsymbol{\xi}_{OD}^{(0)}$ be the initial off-diagonals which can be obtained using a block-coordinate decent algorithm for the off-diagonals in Graphical lasso [17] using $\boldsymbol{Z}$. According to the consistency of the initial estimates [14, 54], we also set an initial estimate of the coefficient as $\tilde{\boldsymbol{\alpha}} = (\lambda_0 I + K(\hat{\rho}, \boldsymbol{\xi}_{OD}^{(0)}))^{-1}(\boldsymbol{y} - \boldsymbol{X}\tilde{\boldsymbol{\beta}})$, where $\hat{\rho}$ and $\tilde{\boldsymbol{\beta}}$ are obtained by restricted maximum likelihood estimator (REML) and the best linear unbiased predictor (BLUP).

Given the fixed initial $\tilde{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\xi}}_{OD}$, the solution of the diagonals, $\hat{\boldsymbol{\xi}}_D = (\xi_1, ..., \xi_p)$ can be

obtained by minimizing the following objective function:

$$Q_1 := \frac{1}{2}||\boldsymbol{y} - \boldsymbol{X}\tilde{\boldsymbol{\beta}} - K(Z, \boldsymbol{\xi}_D|\hat{\boldsymbol{\xi}}_{OD})\tilde{\boldsymbol{\alpha}}||^2 + \frac{\lambda_0}{2}\tilde{\boldsymbol{\alpha}}^T K(Z, \boldsymbol{\xi}_D|\hat{\boldsymbol{\xi}}_{OD})\tilde{\boldsymbol{\alpha}} + \lambda_1||\boldsymbol{\xi}_D||_1, \tag{3.17}$$

which is the first term of the bracket in the joint objective function (3.15) with regularization on the diagonals. The above solution is equivalent to minimize the following penalized log-likelihood function:

$$\hat{\boldsymbol{\xi}}_D = \underset{\boldsymbol{\xi}_D}{\operatorname{argmin}} \Big[-\log f_{\boldsymbol{y}|\boldsymbol{Z}}(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{\xi}_D, \hat{\boldsymbol{\xi}}_{OD}) + \lambda_1||\boldsymbol{\xi}_D||_1 \Big],$$

where λ_1 is the tuning parameter.

We approximately solve this objective function (3.17) using two approaches: The first approach is to employ a Taylor approximation of K function and the other is based on a Taylor approximation of a joint pseudo-likelihood function.

First, since the kernel function K is possibly a complicated nonlinear function, we take the second-order Taylor expansion of the kernel function K such that

$$K(\tilde{\boldsymbol{\xi}}_{-j}, \hat{\xi}_j) \approx K(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j) + K'(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j) + \frac{1}{2}K''(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j)^2 + O_p(|(\hat{\xi}_j - \tilde{\xi}_j)|^3). \tag{3.18}$$

By plugging this equation (3.18) into the function (3.17), we estimate $\hat{\xi}_j$ by solving the gradient equation of the objective function given $\tilde{\alpha}$ and $\hat{\boldsymbol{\xi}}_{OD}$. We refer this approach as $JSKNR2$. The detailed steps of the approximated coordinate algorithm are described in Appendix B.1.1.

The second approach is developed by taking the 2nd approximation of the pseudo-likelihood function at $\xi_j = \tilde{\xi}_j$, where $j = 1, \ldots, p$,

$$L(\tilde{\boldsymbol{\xi}}_{-j}, \hat{\xi}_j) \approx L(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j) + L^{(1)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j) + \frac{1}{2}L^{(2)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j)^2. \tag{3.19}$$

Then, the non-negative garrote parameter is updated coordinatewisely using the following way,

$$\hat{\xi}_j = \tilde{\xi}_j - [L^{(2)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)]^{-1} L^{(1)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j). \tag{3.20}$$

We refer this approach as $JSKNR_{NR}$. The detailed steps of the approximated coordinate algorithm are described in Appendix B.1.2.

3.4.2 Blockwise coordinate decent Algorithm to Update $\boldsymbol{\xi}_{OD}$

Denoting the following cone

$$S_+^p = \{A \in R^{p\times p} | A = A^T, A \succeq 0\},$$

which is formed by all symmetric positive semi-definite matrices in p dimensions, we assume that the covariance matrix Σ^* and precision matrix $\Omega^*(\hat{\xi}_D, \xi_{OD})$ of the random vector $\mathbf{Z}$ are positive semi-definite, and so lie in the interior $S_{++}^p = \{A \in R^{p\times p} | A = A^T, A \succ 0\}$ of the cone S_+^p.

Given fixed $\hat{\xi}_D$ and sample covariance $S_{JSKN}^{iter+1/2}$ which is the current sample covariance at the $iter$th $JSKNR$, the solution of the off-diagonals, ξ_{OD}, can be obtained by minimizing the following objective function

$$Q_2 := \underset{\xi_{OD}\in S_{++}^p}{\operatorname{argmin}} \left\{ \operatorname{tr}\left(\Omega(\hat{\xi}_D, \xi_{OD}) S_{JSKN}^{iter+1/2}\right) - \log det\left(\Omega^*(\hat{\xi}_D, \xi_{OD})\right) + \lambda_2 ||\xi_{OD}||_1 \right\}, \tag{3.21}$$

where $det(A)$ is the determinant of the matrix A. Here, we define the off-diagonal l_1 regularizer as

$$||\xi_{OD}||_1 = \sum_{j\neq j'} |\xi_{j,j'}|,$$

where the sum ranges over all $j, j' = 1, \ldots, p$ with $j \neq j'$. The problem of estimating the entries of the precision matrix $\Omega^*(\hat{\xi}_D, \xi_{OD})$ corresponds to parameter estimation, while the problem of determining which off-diagonal entries ξ_{OD} are non-zero, that is, the set

$$E(\Omega^*(\hat{\xi}_D, \xi_{OD})) = \{(k,l) \in V | k \neq l, \Omega^*(\hat{\xi}_D, \xi_{OD})^*_{kl} \neq 0\},$$

corresponds to the problem of Gaussian graphical model selection.

Friedman et al. [17] solved the optimization problem (3.21). By the blockwise coordinate descent method, the subgradient of the objective function was solved. This procedure was implemented in glasso algorithm. We adapted this procedure to estimate $\boldsymbol{\xi}_{OD}$ given $\hat{\boldsymbol{\xi}}_D$.

We note that given some regularization constant $\lambda_2 > 0$, we consider estimating $\Omega^*(\hat{\xi}_D, \xi_{OD})$ by solving the l_1 *regularized log-determinant program* (3.21), which returns a symmetric positive definite matrix $\hat{\Omega}(\hat{\xi}_D, \xi_{OD})$. As shown in Ravikumar et al. [46, 47], for any $\lambda_2 > 0$ and sample covariance matrix S_{JSKN} with positive diagonal entries, this convex optimization problem has a unique optimum, so there is no ambiguity in equation (3.21). When the data is actually drawn from a multivariate Gaussian distribution, the problem of (3.21) is simply l_1 regularized maximum likelihood. The equality $\xi_{jj'} = 0$ indicates the absence of an edge between nodes j and j' for the corresponding Gaussian graphical model, so the penalty $||\xi_{OD}||_1$ encourages a sparse graphical model.

3.4.3 Joint Model Selection

Both variable selection and network estimation among important predictors require to select two penalty parameters λ_1 and λ_2. The first λ_1 for variable selection is selected using the least square kernel BIC [32],

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

where $RSS = (\boldsymbol{y} - \hat{\boldsymbol{y}})^T(\boldsymbol{y} - \hat{\boldsymbol{y}})$ and df denotes the degrees of freedom of the smoothing matrix A of the kernel machine. The smoother matrix is given by $A = (I+\lambda_0^{-1}K)^{-1}[\lambda_0^{-1}K + X\{\boldsymbol{X}^T(I+\lambda_0^{-1}K)^{-1}\boldsymbol{X}\}^{-1}\boldsymbol{X}^T(I+\lambda_0^{-1}K)^{-1}]$ and the predicted response, $\hat{\boldsymbol{y}} = A\boldsymbol{y}$.

The other penalty parameter λ_2 for the off-diagonals using Gaussian graphical model is selected using an extended BIC [13] denoted by $EBIC$ such that

$$EBIC = -2\log f_Z(Z|\widehat{\Omega}(\hat{\boldsymbol{\xi}})) + |E|\log(n) + 4|E|\gamma\log(p),$$

where $|E|$ is the number of non-zero edges in E and γ is a regularizer of the edge set for the consistency of tuning search even when p grows. In general, γ is set to be 0.5 based on Drton and Foygel [13].

Using these model selection criteria, the algorithm for our joint semiparametric kernel machine regression iterative searches for an optimal joint model.

3.5 Simulation

We conduct simulation studies to investigate the performance of our JSKNR methods (denoted as $JSKNR2$ and $JSKNR_{NR}$ in Section 3.4 in terms of accuracy of variable selection and network estimation. The detailed simulation settings and simulation evaluation are described in Sections 3.5.1 and 3.5.2, respectively. The simulation results on the variable selection and the network estimation are summarized in Sections 3.5.3 and 3.5.4.

For the performance of the variable selection, we compare our approaches with NGK with scale parameter c (denoted as NGK_s) and an iterative COSSO (denoted as $iCOSSO$) approach for the performance of the variable selection which is applicable for variable selection under semiparametric regression. Originally, NGK [14] was developed without scale c. NGK without scale c (denoted as NGK_{wos}) often provided unstable and numerical problems

because of a lack of ability to distinguish the information of the variable selection with the degrees of the dissimilarity between the points. The performance of NGK with scale c has much improved over the NGK without c. For $iCOSSO$, since COSSO [31] was developed under nonparametric model, we further extended it for the semiparametric regression setting by incorporating an iterative method into COSSO. The detailed algorithm of $iCOSSO$ is provided in Appendix B.2. For the estimation of network, we also compare our approaches with GLASSO [17].

We study the performance of our approaches when $p < n$, $p \approx n$, and $p > n$. We consider two cases: correctly specified and mis-specified kernels using polynomial kernel (PK) and Gaussian kernel (GK). For the setting of correctly specified kernel, a true generating kernel function is matched to the fitting kernel such that both are PKs or GKs. For the case of mis-specified kernel, the kernels in the true model and the fitted model can be arbitrarily different. For example, the true model is generated using PK, but the specified kernel in fitting the model is GK, or vice versa. In this simulation study, we focus on the setting when the true generating model is based on PK but the fitted model uses GK. We consider three settings of precision matrices in our simulation study: (1) precision matrix with diagonal matrix $I_{p\times p}$, (2) precision matrix with AR(1) structure, and (3) precision matrix with AR(2) structure.

For the tuning parameters of $JSKNR$ and NGK method, we choose the optimal values for the tuning parameters by using the two BICs described in Section 3.4.3 while the tuning parameter of $iCOSSO$ is selected using the cross-validation.

3.5.1 Simulation Setting

We generate X, Z , and y from our joint semiparametric model (3.5). We first set true parameters for $(\beta, \tau, \sigma^2_\epsilon)$ as (1,10,1). The kernel function $K(\cdot)$ is considered as either PK or

GK. We have five combinations of n and p: $(n,p) = (64,20), (30,20), (30,40), (30,70)$, and $(30,120)$, which correspond to $n > p$, $n \geq p$, $n \leq p$, and $n < p$. The number of true signal variables (p^*) is set as (5,10,15,20) and the number of noise variables is $p - p^*$. These (n,p) settings were motivated by our Type II diabetes genomics application. We make sure that $n \geq c_0 \times d^2 \log(p^*)$ where c_0 is any real number and d is the maximum number of non-zeros in any row of the precision matrix.

Three setting of precision matrix are (1) $\Omega = I_{p\times p}$, (2) $\Omega = AR(1)$, where diagonal element $\Omega_{jj} = 1$ and off-diagonal element $\Omega_{j,j-1} = \Omega_{j-1,j} = \rho^*$ with $\rho^* = (0.25, 0.50)$, and (3) $\Omega = AR(2)$, where $\Omega_{jj} = 1$, $\Omega_{j,j-1} = \Omega_{j-1,j} = \rho^*$, and $\Omega_{j,j-2} = \Omega_{j-2,j} = \rho^{*2}$, $j = 1, \ldots, p$.

We then generate X from Unif$[-6,6] \in \mathbb{R}^n$. The true signal variables ($\boldsymbol{z}_{true,j}$) are generated from $\boldsymbol{z}_{true,j} \overset{i.i.d}{\sim} MVN\big(\mathbf{0}, \Omega^{-1}\big) \in \mathbb{R}^n$. The noise variables ($\boldsymbol{z}_{noise,j}$) are generated from Unif$(0,1) \in \mathbb{R}^n$. Finally, y is generated from a joint semiparametric model (3.5).

3.5.2 Simulation Evaluation

We employ 200 simulated datasets for each combination and evaluate the performance of methods. For the evaluation of the performance on the variable selection, we use the following six measures: accuracy (ACC), false positive rate (FPR), false negative rate(FNR), true positive rate(TPR), residual sum of squares(RSS), and mean squared error (MSE), defined as follows:

$$ACC = \frac{TP}{TP+TN}, \quad FPR = \frac{FP}{TN+FP}, \quad FNR = \frac{FN}{FN+TP}, \quad TPR = \frac{TP}{FN+TP},$$

$$RSS = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}, \quad MSE = \frac{\sum_{i=1}^{n}(h_i - \hat{h}_i)^2}{n},$$

where TP and FP stand for the number of true positives (correctly identified important components) and the number of false positives (incorrectly identified important components), respectively. The average and standard deviation of the statistics from 200 runs are calculated. $\hat{\boldsymbol{y}}$ is equal to $X\hat{\boldsymbol{\beta}} + \hat{\boldsymbol{h}}$, where $\hat{\boldsymbol{h}}$ is estimated using the least squared estimation with the kernel machine corresponding to the estimated precision matrix $\widehat{\Omega}$ from our $JSKNR$ method. That is, $\hat{\boldsymbol{h}} = K(\widehat{\Omega})\{I + \lambda_0^{-1}K(\widehat{\Omega})\}^{-1}(\boldsymbol{y} - X\widehat{\boldsymbol{\beta}})$, where $\widehat{\boldsymbol{\beta}} = \left[X^T\{I + \lambda_0^{-1}K(\widehat{\Omega})\}^{-1}X\right]^{-1}X^T\{I + \lambda_0^{-1}K(\widehat{\Omega})\}^{-1}\boldsymbol{y}$. MSE is used to evaluate the estimation accuracy of the semiparametric function.

On the other hand, for the evaluation of the performance on the estimation of the network structure, we use the following three matrix norms: (1) Frobenius norm, (2) element-wise l_1 norm, and (3) matrix l_1 norm, defined as follows:

$$||\Omega||_F = \sqrt{\sum_{i,j} \xi_{ij}^2}, \quad ||\Omega||_{1,e} = \sum_{j=1}^{p}\sum_{i=1}^{p} |\xi_{ij}|, \quad ||\Omega||_{1,m} = \max_{1\le j\le p} \sum_{i=1}^{p} |\xi_{ij}|.$$

The average values of bias and standard error of the statistics are computed for each matrix norm.

3.5.3 Simulation Result for the Variable Selection

The performance of the variable selection using our $JSKNR$ methods are summarized in terms of correctly-specified kernels and misspecified kernels with the following three setting of precision matrix:

- Correctly-specified kernels with identity, AR(1), and AR(2) precision matrices

  We explore the performances of each variable selection approach when both kernels used in the true model and the fitted model are the two types of kernels and the three various structures of precision matrix. In this dissertation, we summarize the

performances of each variable selection method when the both models use GKs and the precision matrix has the AR(1) structure with the magnitude of the correlation as $\rho^* = 0.25$ in Table 3.1. For the performances based on the other cases, they are provided in Tables B.1 – B.9 of Appendix B.3.

- Mis-specified kernels with identity, AR(1), and AR(2) precision matrices

  When the true model is based on PK but the fitted model uses GK with the identity precision matrix, the performances of each variable selection method are summarized in Table 3.2. In same case as the previous setting but using the AR(1) precision matrix, the performances of variable selection approaches are described in Table 3.3. For the same setting as the previous one but AR(2), the performances of the variable selection are summarized in Tables B.11 and B.12 of Appendix B.3.

From Table 3.1, when using Gaussian kernel in both the true model and the fitted model, $JSKNR2$ and $JSKNR_{NR}$ have better performances than NGK method in the all aspects of the measures. Table B.6 – B.9 of Appendix B.3, $JSKNR2$ and $JSKNR_{NR}$ outperform NGK method, especially when both kernels used in the true model and the fitted model are GKs when the magnitude of the correlation and p gets higher. This result suggests that our second approximation approaches of the kernel function or pseudo joint likelihood function are more likely to reduce the bias of the functional estimation and the variable selection, compared to the NGK method using the first approximation. When using the linear polynomial kernel in the correctly specified case, the performances of $JSKNR$ method and NGK method are comparable to each other because the second approximation based on the polynomial kernel becomes same as the first approximation in the NGK method. However, when the correlation among the predictors gets larger, the performance of the NGK becomes unstable because it does not take account of the correlation in the regression setting. The results based on the polynomial kernel are provided in Tables B.1 – B.5 of Appendix B.3. For the performance of

iCOSSO, *iCOSSO* is not applicable, when $(n, p) = (30, 20)$ as well as when $n < p$ because of the complications of modeling the large amount of high-order interactions componentwise. Even when $n > p$ such that $(n, p) = (64, 20)$, *iCOSSO* has poor performance on the variable selection based on the all models. Thus, thses simulation results suggest that our proposed methods outperform the existing methods.

Table 3.1: Simulation results of four methods when true kernel is Gaussian and the precision matrix has AR(1) structure with $\rho^* = 0.25$ based on 200 simulation runs; four methods $JSKNR2$, $JSKNR_{NR}$, NGK_s, and *iCOSSO* are compared in tems of six evaluation measures; Gaussian kernel is used for fitting a model

Method	True / Fitted	(n, p)	ACC	FPR	FNR	TPR	RSS	MSE
$JSKNR2$	Gaussian / Gaussian	(64,20)	0.952(0.101)	0.032(0.092)	0.063(0.192)	0.937(0.192)	1.377(1.315)	1.117(1.121)
		(30,20)	0.965(0.062)	0.009(0.043)	0.060(0.118)	0.940(0.118)	1.269(1.149)	0.996(0.754)
		(30,40)	0.966(0.054)	0.016(0.051)	0.088(0.122)	0.912(0.122)	1.129(0.546)	0.917(0.254)
		(30,70)	0.970(0.051)	0.015(0.039)	0.114(0.153)	0.886(0.153)	1.042(0.317)	0.879(0.251)
		(30,120)	0.941(0.061)	0.039(0.048)	0.280(0.248)	0.720(0.248)	1.056(0.590)	0.939(0.305)
$JSKNR_{NR}$		(64,20)	0.878(0.123)	0.153(0.221)	0.09(0.197)	0.910(0.197)	1.227(0.980)	1.023(0.737)
		(30,20)	0.931(0.076)	0.046(0.085)	0.092(0.124)	0.908(0.124)	1.066(0.459)	0.915(0.305)
		(30,40)	0.958(0.054)	0.023(0.051)	0.099(0.120)	0.901(0.120)	0.993(0.163)	0.931(0.249)
		(30,70)	0.904(0.090)	0.067(0.082)	0.273(0.189)	0.727(0.189)	0.905(0.222)	0.878(0.232)
		(30,120)	0.934(0.068)	0.044(0.056)	0.308(0.243)	0.692(0.243)	0.914(0.426)	0.919(0.251)
NGK_s		(64,20)	0.871(0.162)	0.080(0.200)	0.179(0.257)	0.821(0.257)	3.201(3.310)	2.673(2.745)
		(30,20)	0.883(0.124)	0.189(0.242)	0.045(0.127)	0.955(0.127)	1.220(1.085)	1.026(0.820)
		(30,40)	0.880(0.134)	0.114(0.148)	0.139(0.132)	0.861(0.132)	1.014(0.581)	0.958(0.465)
		(30,70)	0.841(0.110)	0.138(0.099)	0.286(0.200)	0.714(0.200)	0.935(0.360)	0.930(0.310)
		(30,120)	0.859(0.06)	0.115(0.049)	0.429(0.215)	0.571(0.215)	0.844(0.430)	1.001(0.314)
iCOSSO		(64,20)	0.578(0.056)	0.025(0.063)	0.819(0.142)	0.181(0.142)	1.108(0.809)	1.842(0.713)
		(30,20)	NA	NA	NA	NA	NA	NA
		(30,40)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA
		(30,120)	NA	NA	NA	NA	NA	NA

For the case of the misspecified kernel, as shown in Tables 3.2–3.3, and Tables B.10 – B.12 of Appendix B.3, the performances of both $JSKNR$ and NGK methods deteriorate in terms of RSS and MSE, compared to the correctly specified case. This may be because Gaussian kernel, which is a family of flexible functions, may lead to over-fitting so that it can give incorrect selection of the components and large RSS and MSE. However, although FNR increases due to the misspecification, the two $JSKNR$ methods maintain reasonable accuracy and TPR in selecting the true signals. Furthermore, the $JSKNR$ methods have less RSS and MSE than those of NGK method.

Table 3.2: Simulation results of three methods when the true kernel is polynomial and the precision matrix is diagonal based on 200 simulation runs; three methods $JSKNR2$, $JSKNR_{NR}$, and NGK_s are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model

Method	True / Fitted	(n,p)	ACC	FPR	FNR	TPR	RSS	MSE
$JSKNR2$	Polynomial / Gaussian	(30,20)	0.961(0.048)	0.002(0.015)	0.075(0.097)	0.925(0.097)	6.708(18.201)	5.527(15.055)
		(30,40)	0.976(0.027)	0.003(0.011)	0.086(0.100)	0.914(0.100)	8.531(23.353)	6.978(19.207)
		(30,70)	0.984(0.019)	0.003(0.013)	0.096(0.109)	0.904(0.109)	5.059(14.053)	2.249(5.115)
		(30,120)	0.981(0.018)	0.003(0.014)	0.198(0.130)	0.802(0.130)	20.717(33.115)	15.176(24.875)
$JSKNR_{NR}$		(30,20)	0.922(0.073)	0.022(0.077)	0.134(0.127)	0.866(0.127)	5.440(14.323)	4.487(11.696)
		(30,40)	0.945(0.058)	0.020(0.060)	0.158(0.138)	0.842(0.138)	7.390(18.795)	6.238(16.034)
		(30,70)	0.962(0.034)	0.012(0.033)	0.195(0.148)	0.805(0.148)	8.503(21.761)	7.271(19.270)
		(30,120)	0.968(0.029)	0.010(0.028)	0.275(0.144)	0.725(0.144)	15.215(29.176)	12.843(25.117)
NGK_s		(30,20)	0.712(0.129)	0.420(0.242)	0.156(0.114)	0.844(0.114)	6.219(25.161)	5.710(22.497)
		(30,40)	0.861(0.105)	0.111(0.140)	0.225(0.130)	0.775(0.130)	27.991(38.668)	24.217(35.280)
		(30,70)	0.903(0.069)	0.071(0.082)	0.247(0.140)	0.753(0.140)	31.480(45.086)	27.627(40.866)
		(30,120)	0.897(0.059)	0.093(0.063)	0.219(0.114)	0.781(0.114)	18.47(30.626)	15.628(26.056)

Table 3.3: Simulation results of three methods when the true kernel is polynomial and the precision matrix has AR(1) structure with $\rho^* = 0.25$ based on 200 runs: four methods $JSKNR2$, $JSKNR_{NR}$, and NGK_s are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model

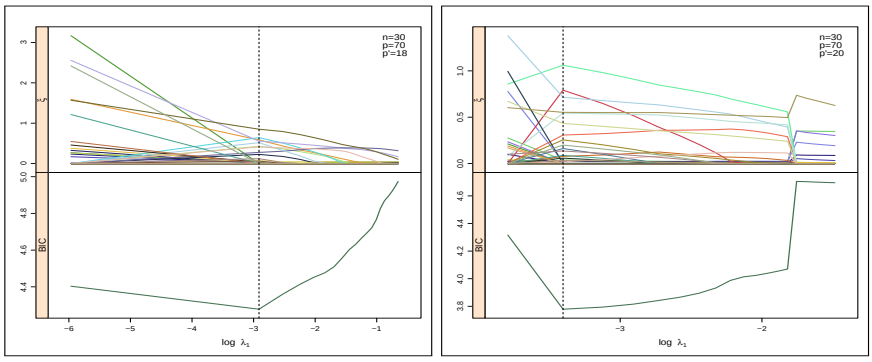
Method	True / Fitted	(n,p)	ACC	FPR	FNR	TPR	RSS	MSE
$JSKNR2$	Polynomial / Gaussian	(30,20)	0.923(0.077)	0.003(0.017)	0.150(0.154)	0.850(0.154)	12.843(23.700)	8.285(28.142)
		(30,40)	0.973(0.031)	0.003(0.012)	0.099(0.114)	0.901(0.114)	9.072(24.161)	7.320(19.689)
		(30,70)	0.983(0.019)	0.002(0.007)	0.109(0.130)	0.891(0.130)	8.654(22.587)	6.649(18.144)
		(30,120)	0.982(0.015)	0.003(0.009)	0.182(0.146)	0.818(0.146)	20.18(31.066)	15.957(25.265)
$JSKNR_{NR}$		(30,20)	0.898(0.081)	0.018(0.079)	0.186(0.145)	0.814(0.145)	10.121(22.677)	8.856(28.783)
		(30,40)	0.940(0.053)	0.023(0.056)	0.172(0.153)	0.828(0.153)	7.904(20.332)	6.713(17.447)
		(30,70)	0.958(0.036)	0.015(0.034)	0.203(0.162)	0.797(0.162)	9.032(21.214)	7.401(17.853)
		(30,120)	0.975(0.024)	0.007(0.02)	0.228(0.162)	0.772(0.162)	16.399(30.464)	13.897(26.381)
NGK_s		(30,20)	0.754(0.176)	0.348(0.305)	0.145(0.126)	0.855(0.126)	8.274(28.387)	7.523(25.389)
		(30,40)	0.693(0.118)	0.353(0.140)	0.168(0.113)	0.832(0.113)	24.242(39.778)	20.325(36.218)
		(30,70)	0.877(0.088)	0.096(0.107)	0.285(0.137)	0.715(0.137)	37.971(59.004)	33.784(56.064)
		(30,120)	0.910(0.051)	0.076(0.053)	0.242(0.128)	0.758(0.128)	19.016(31.095)	15.984(26.502)

In addition, Figure 3.2 provides the variable selection procedure to find the solution path for the 10 important components among the total 70 components. Figures 3.2 (a) and (b) illustrate the solution paths based on NGK method and Figures 3.2 (c) – (h) show the soultion paths based on $JSKNR$ method. As shown in the figures, NGK method tends to select more components than the number of the important components while our $JSKNR$ approach is able to closely select the important components through the several iterations.

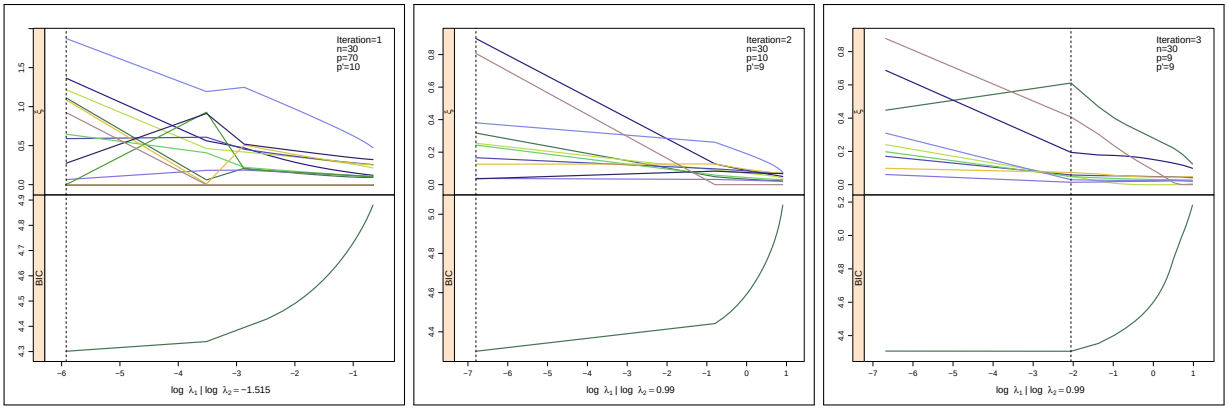
3.5.4 Simulation Result for Network Estimation

We also explore the performance of our $JSKNR$ method regarding the estimation of the network structure. Glasso is employed as the existing method for the comparison. We use the three norm metrics defined in Section 6.2 for the comparison of the performance. We then calculate the bias of the total of entries between the true and the estimated precision matrix. Based on the fact that $JSKNR2$ and $JSKNR_{NR}$ have similar performances to each other, we apply $JSKNR2$ method to the estimation as representative. Table 3.4 describes the average of biases and standard errors from the estimation of the AR(1) precision matrix using the Gaussian kernel. For the other settings, the results are summarized in Tables B.13 – B.15 of Appendix B.3.

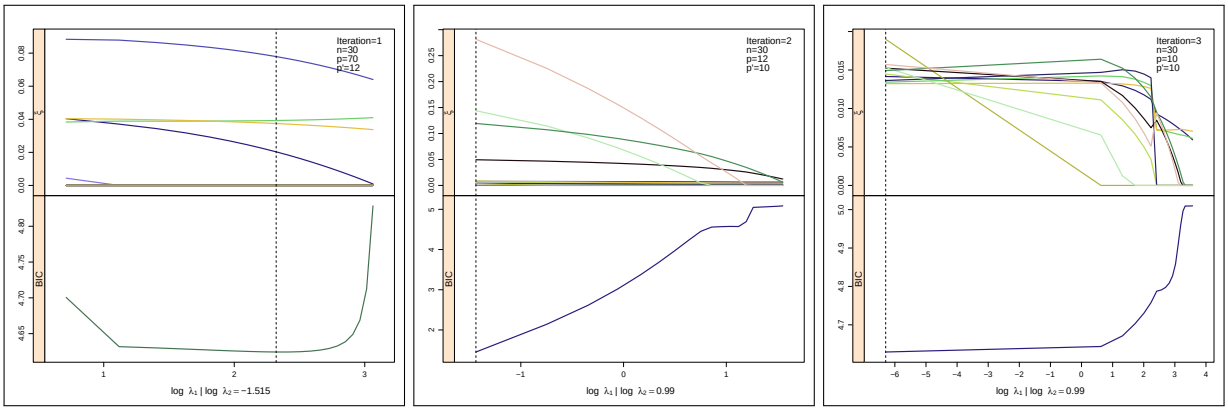
Based on the results, our $JSKNR$ method uniformly outperforms Glasso in terms of all of the norms. As the dimension of the precision matrix or the magnitude of the correlation increase, the performance of $JSKNR$ gets remarkably better than that of Glasso because $JSKNR$ method is able to conduct the dimension reduction as well as the estimation of the network structure.



(a) Example 1 with NGK (b) Example 2 with NGK



(c) Example 1 with JSKNR2; 1st iteration (d) Example 1 with JSKNR2; 2nd iteration (e) Example 1 with JSKNR2; 3rd iteration



(f) Example 2 with JSKNR2; 1st iteration (g) Example 2 with JSKNR2; 2nd iteration (h) Example 2 with JSKNR2; 3rd iteration

Figure 3.2: Two selected examples of NGK's solution paths displayed in Figures (a) – (b) and JSKNR2's solution paths displayed in Figures (c) – (h) using Gaussian kernel when $(n, p) = (30, 70)$

Table 3.4: Summary results of the average bias (s.e) of two methods when three precision matrices are considered based on 200 runs; two methods $JSKNR$ and $Glasso$ are compared in terms of three matrix norms. Three precision matrices have diagonal ($\rho^* = 0$), AR(1) with $\rho^* = 0.25$, and AR(1) with $\rho^* = 0.5$ structures; the estimated precision matrix is using Gaussian kernel;

Kernel	(n, p)	ρ^*		$\widehat{\Omega}_{JSKNR2}$	$\widehat{\Omega}_{Glasso}$
Gaussian	(30, 40)	0.00	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.157 (0.197)	1.095 (0.541)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.002 (0.081)	0.080 (0.095)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	0.551 (0.666)	9.224 (3.366)
		0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.056 (1.176)	8.442 (5.589)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.088 (0.674)	1.269 (1.158)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	1.736 (4.212)	51.634 (33.565)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.288 (1.410)	8.854 (4.594)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.037 (0.776)	1.071 (0.962)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	2.979 (5.066)	57.432 (28.486)
	(30, 70)	0.00	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.143 (0.242)	1.743 (0.825)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.005 (0.094)	0.074 (0.106)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	0.475 (0.864)	18.481 (6.834)
		0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.873 (1.438)	13.742 (7.309)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.665 (0.766)	1.801 (1.093)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	2.558 (5.045)	114.962 (59.467)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.052 (1.509)	14.319 (6.827)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.790 (0.811)	1.795 (1.028)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	3.611 (5.465)	122.631 (56.951)
	(30, 120)	0.00	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.501 (0.249)	2.950 (0.006)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.096 (0.030)	0.109 (0.001)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	2.893 (1.469)	38.225 (0.089)
		0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	5.619 (3.018)	23.809 (0.153)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	1.842 (0.643)	2.390 (0.078)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	21.406 (13.015)	262.159 (1.675)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	2.166 (1.997)	23.686 (1.758)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	1.277 (0.684)	2.276 (0.211)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	7.382 (6.535)	266.774 (19.385)

In addition, we explore the estimated network structure based on the estimated precision matrices using heatmaps. In this dissertation, the results of estimation on AR(1) network structures using GK with $(n, p) = \{(30, 40), (30, 70), (30, 120)\}$ are shown in Figures 3.3–3.5, respectively. For the other simulation settings, the results are illustrated in Figures B.1 – B.7 of Appendix B.3. The figures show the heatmaps of the estimated precision matrices based on $JSKNR$ method and Glasso with various combinations of (n, p). As shown in the figures, $JSKNR$ method takes advantage of both the dimension reduction and the estimation of the network structure. As the dimension of the precision matrix increases, the benefit of our method will remarkably grow. In terms of the estimation, when the degrees of the correlation is somewhat small such as $\rho^* = 0.25$, the pattern of the network is hard to be estimated correctly in both $JSKNR$ methods and Glasso. In contrast, as the magnitude of the correlation is high, it can give a more accurate estimation but lead to an incorrect result of the variable selection due to multicollinearity. The results suggest that there would be a trade-off between the variable selection and the estimation on the network.

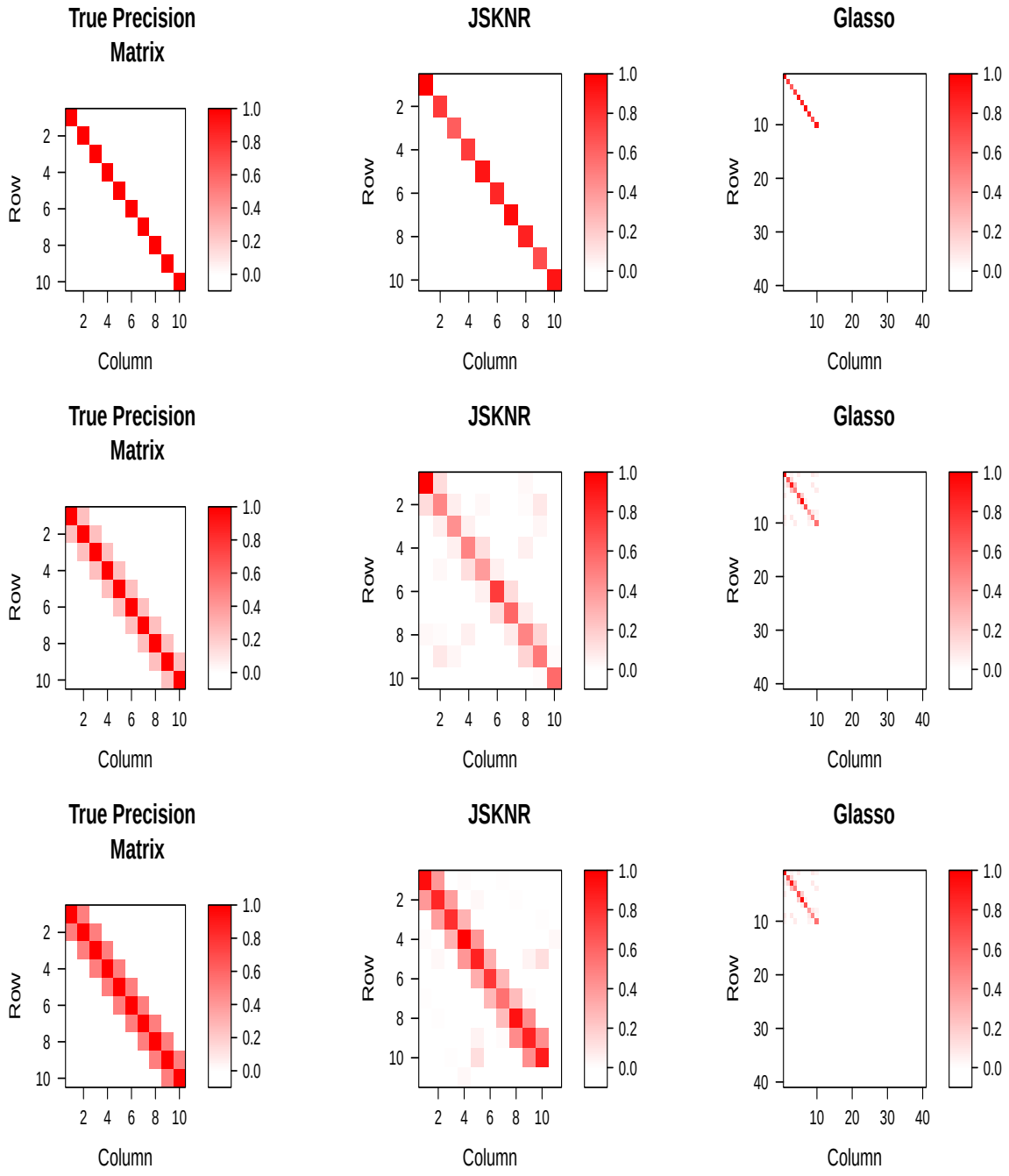


Figure 3.3: Heatmaps of the estimated precision matrices based on $JSKNR$ and $Glasso$ when the precision matrix has a AR(1) structure and Gaussian kernel is used when $(n, p) = (30, 40)$; The figures represent the estimated precision matrices of AR(1) structure with $\rho^* = 0.00$ (top three), 0.25 (middle three), and 0.50 (bottom three), respectively.

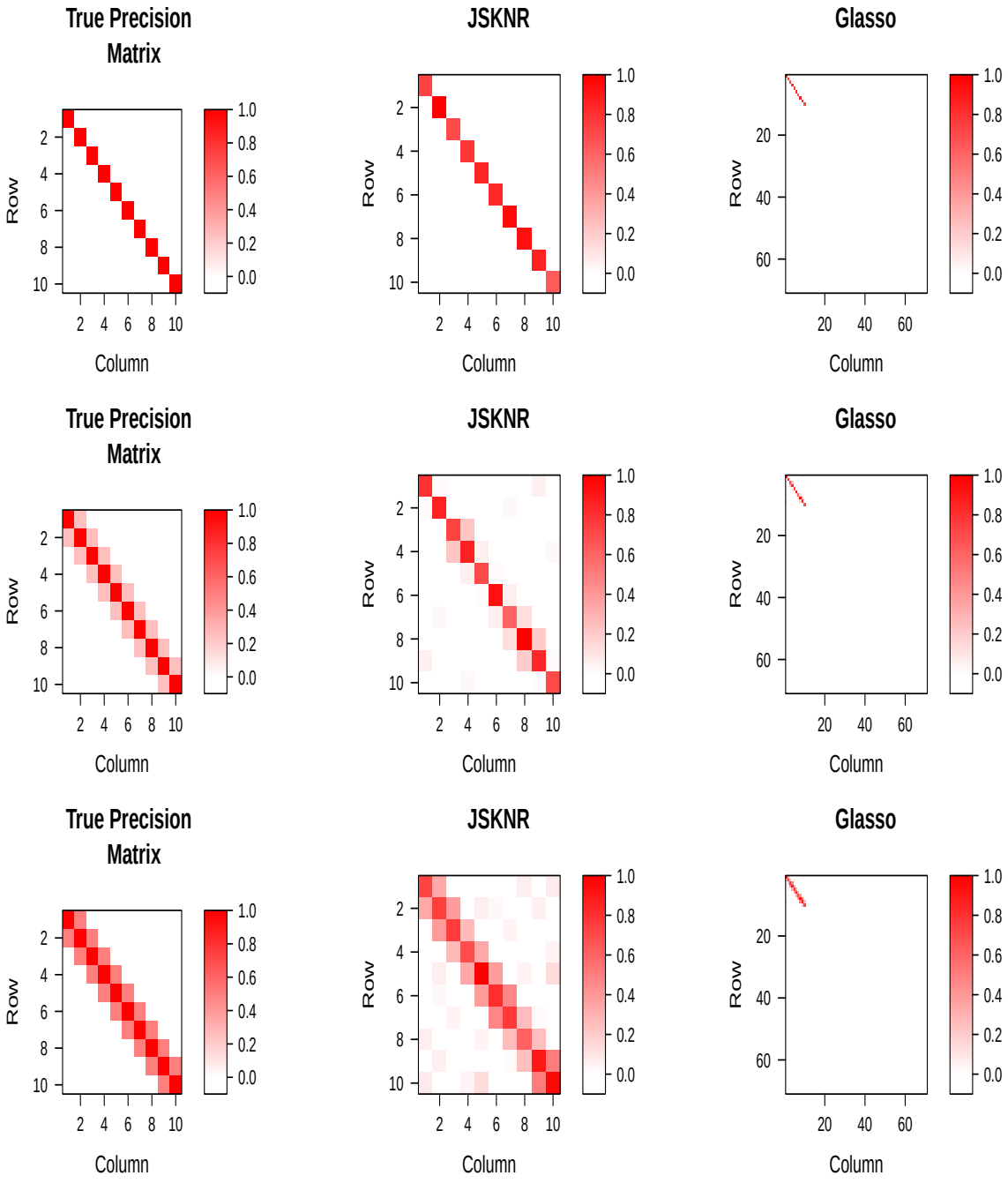


Figure 3.4: Heatmaps of the estimated precision matrices based on $JSKNR$ and $Glasso$ when the precision matrix has a AR(1) structure and Gaussian kernel is ised when $(n, p) = (30, 70)$; The figures represent the estimated precision matrices of AR(1) structure with $\rho^* = 0.00$ (top), 0.25 (middle), and 0.50 (bottom), respectively.

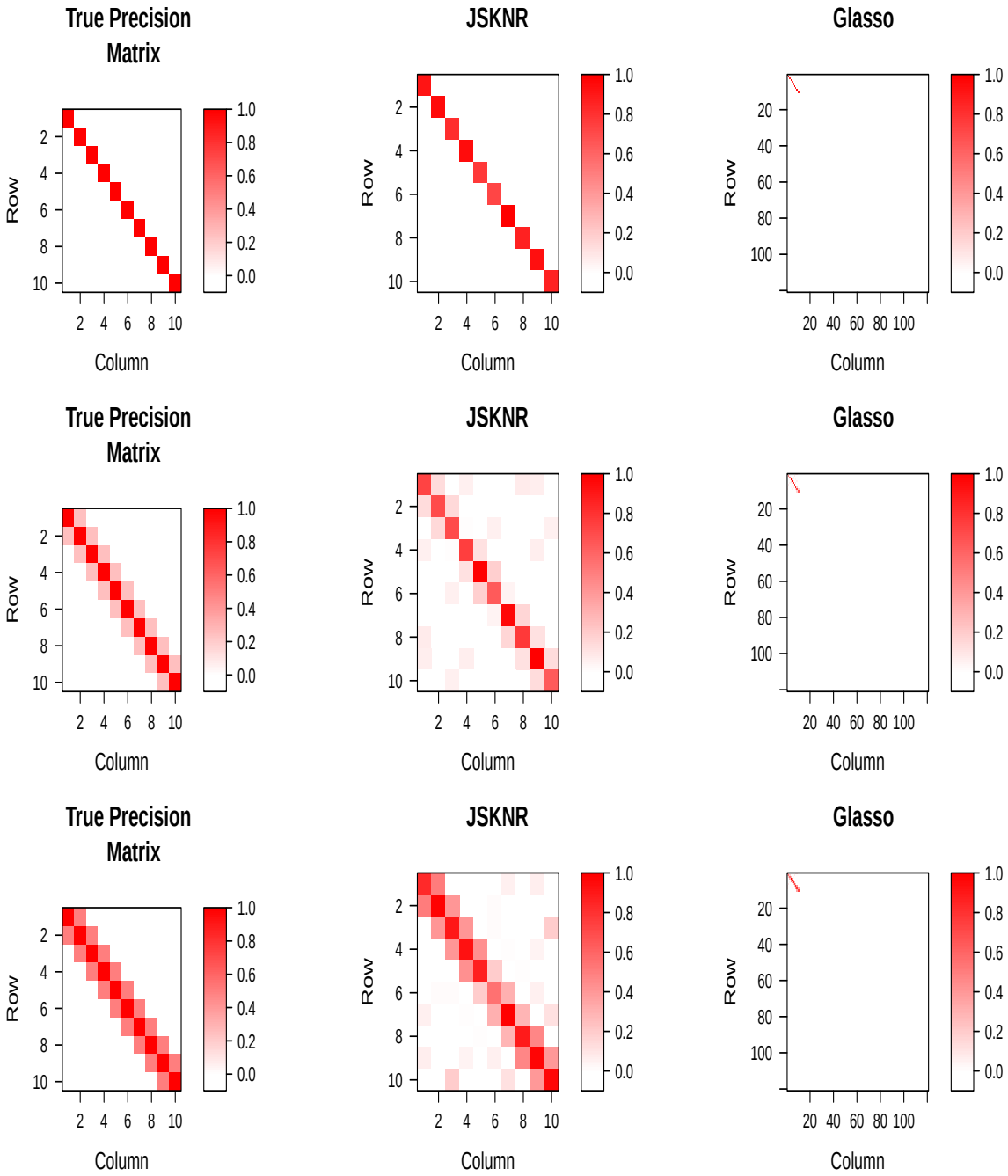


Figure 3.5: Heatmaps of the estimated precision matrices based on $JSKNR2$ and Glasso when the precision matrix has a AR(1) structure and linear polynomial kernel is used with $(n, p) = (30, 120)$; The figures represent the estimated precision matrices of AR(1) structure with $\rho^* = 0.00$ (top three), 0.25 (middle three), and 0.50 (bottom three), respectively.

3.6 Application

In this section, we applied our $JSKNR$ methods to gene expression microarray data of Type II diabetes mellitus (DM2) experimented by Mootha et al. [39]. Data contains the information for over 22,000 genes in skeletal muscle biopsy samples from males at the age of 43, where there are 17 with normal glucose level and 18 with DM2. Mootha et al. [39] suggested that a single gene-based analysis with a large number of genes and high variability among the subjects merely detects the subtle change in gene expression. They proposed a pathway-based analysis to increase the power of detecting subtle but coordinated changes in gene expression. Here, a pathway is defined as a set of genes that mutually serve a particular cellular physiological function. It is acknowledged that the multiple members of the pathways are correlated to each other with unknown structures of interaction. Mootha et al. [39] reported that the pathways related to oxidative phosphorylation (OXPHOS) are significantly correlated with insulin resistance and aerobic capacity.

Our goals are to apply our approaches to the top significant pathways related to OXPHOS. Our interest is to identify important genes within a pathway and to discover unknown and correlated network structure among the important genes. We focus on five pathways: pathway 133 ("MAP00190-Oxidative phosphorylation"), pathway 4 ("Alanine and aspartate metabolism"), pathway 140 ("MAP00252-Alanine and aspartate metabolism"), pathway 16 ("ATP synthesis"), and pathway 229 ("Oxidative phosphorylation"). The pathways consist of 18, 22, 58, 49, and 133 genes. The first three pathways were also used by Fang et al. [14] for NGK approach. We note that the first two pathways contain a smaller number of genes than the sample size, while the last three pathways contain a larger number than the sample size. Fang et al. [14] suggested the possible existence of the interaction between pathway 4 and pathway 140. However, this NGK method is not able to provide information of the correlations among the important genes within the pathways, while our JSKNR methods can

do because our approaches are developed by incorporating the dependence structure among genes within the pathways into the variable selections.

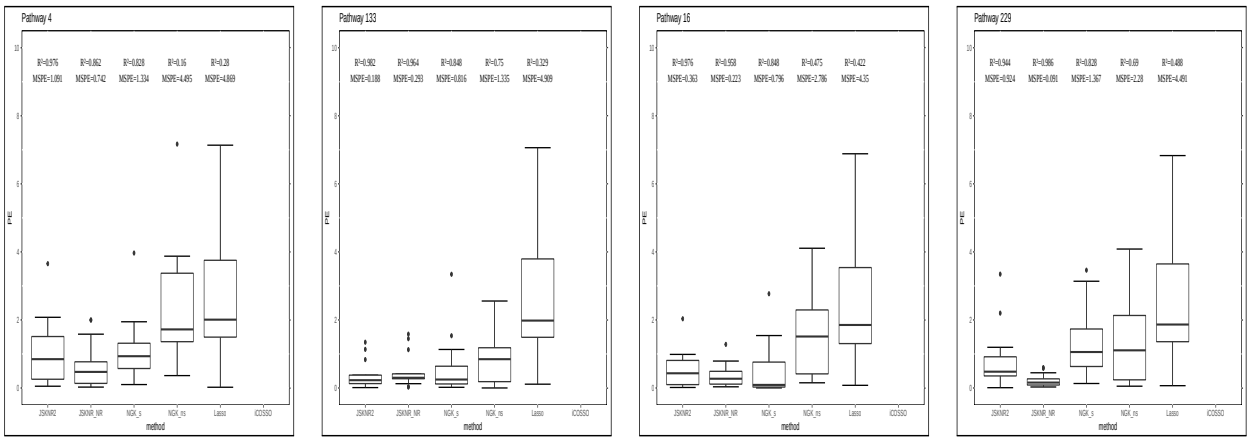
In the application, the response ($\boldsymbol{y}_{35\times 1}$) represents the continuous degrees of glucose tolerance in blood capillary. Microarray-derived gene expression levels($\boldsymbol{Z}_{35\times p}$) varies from different pathways, where $p = 18, 22, 58, 49$, and 133 genes are corresponding to each pathway. We compare our $JSKNR2$ and $JSKNR_{NR}$ approaches with other four approaches: lasso, NGK with and without the scale parameter c (denoted them as NGK_s and NGK_{wos}, respectively), and $iCOSSO$. We conduct the comparison of the performances in terms of selected genes, R^2 between the actual and predicted response value, and the mean squared prediction error (MSPE) such that

$$MSPE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_{i,-i})^2,$$

where $\hat{y}_{i,-i}$ is the ith predicted response value obtained from the fitted model excluding the ith observation.

For results of the important gene selection, our $JSKNR$ methods selected $\mathcal{A}_4 = \{$ADSL1, CRAT1, ASPA, ABAT, GAD2$\}$ and $\mathcal{A}_{140} = \{$DDX3X2, ADSL1, GAD2$\}$ as the important genes in pathway 4 and pathway 140, respectively. Note that $\mathcal{A}_{n_{path}}$ defines a set of the selected important genes in the n_{path}th pathway. We found that ADSL1 and GAD2 overlap between pathway 4 and 140. Compared to the selected genes based on our method, the number of genes chosen by lasso is less than the $JSKNR$'s selected genes, while NGK selected more important genes than our method. It is no surprise because Lasso's l_1 penalty tends to make estimated coefficients more sparse. For the NGK method, Fang et al. [14] suggested using a permutation of NGK because the variable selection depending on a single draw is likely to select many genes so that it might not be powerful. An interesting finding is that the selected genes based on $JSKNR$ method for each pathway are consistent with

the genes chosen by the NGK permutation in the previous work by Fang et al. [14]. This implies that our algorithm can give more efficient variable selection without the permutation. Thus, $JSKNR$ is a less expensive and faster method that can produce a powerful result, compared to the NGK method. There are also similar patterns of the results in selecting the important genes for each method in pathway 133, 49, and 229, which correspond to the case when $p > n$. In terms of R^2 and MSPE, Figure 3.6 shows that $JSKNR$ methods provide high R^2 and small MSPE, compared to the other method in all pathways. Note that $iCOSSO$ was not able to conduct the variable selection because of numerical instability. Hence we marked NA for $iCOSSO$'s result in the results of the tables.



(a) Boxplot for Pathway 4 (b) Boxplot for Pathway 133 (c) Boxplot for Pathway 16 (d) Boxplot for Pathway 229

Figure 3.6: Boxplots of prediction errors from the estimated functions based on $JSKNR2$, $JRSNR_{NR}$, NGK_s(NGK with the scale parameter c), NGK_{ns} (NGK without scale parameter c), Lasso, and $iCOSSO$ for pathways 4, 133, 16 and 229; No boxplot for $iCOSSO$ due to numerical issue in $iCOSSO$

For the result of network among important genes, we construct separate graphs for the genetic network among the important genes selected by $JSKNR$ method within each pathway. Figure 3.7–3.8 provide the network structures before and after considering only important genes within each pathway. The graphs show that all of the important genes are connected so that all of the genes have at least one conditional dependence. Based on the result from

Figure 3.6 and their conditional dependence, it suggests that the correlated structure of the covariance matrix can improve the estimation on the model. Furthermore, our method is able to make a feasible and efficient interpretation of the relationship between the important genes and glucose tolerance in blood capillary with no loss of information from discretizing the response.

3.7 Discussion

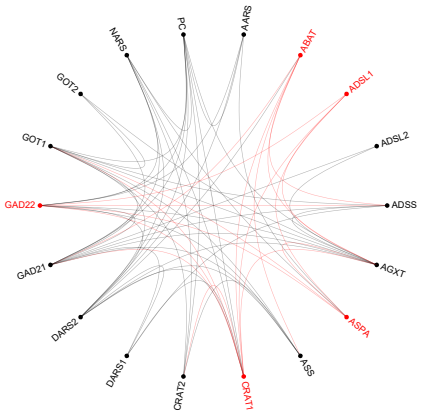
In this dissertation, we develop the joint semiparametric kernel machine network approach to solve the limitations of the variable selection approach and graphical model. Gaussian graphical model is that it is only applicable for discretized response variable and for the case $p\log(p) \ll n$, where p is the number of variables and n is the sample size. It is needed to develop a joint method between variable selection and graphical model. Our approach is a unified and integrated method that can simultaneously identify important variables and build a network among them. We develop our approach under a semiparametric kernel machine regression framework, which can allow for the possibility that each variable might be nonlinear and interact with each other in a complicated way. To the best of our knowledge, the methods for simultaneously conducting variable selection as well as estimating networks among variables under the semiparametric regression settings are quite limited. The advantages of our approach are that (1) it can perform simultaneously variable selection and build network among high-correlated and high-dimensional variables under regression setting; (2) automatically model unknown and complicated interactions among variables and also estimate networks among these variables; (3) have the flexibility for any semiparametric model including non-additive and nonparametric model; and (4) provide interpretable network by selecting or combining the variables.

We compared our approach with the existing approaches. Our approaches can work well

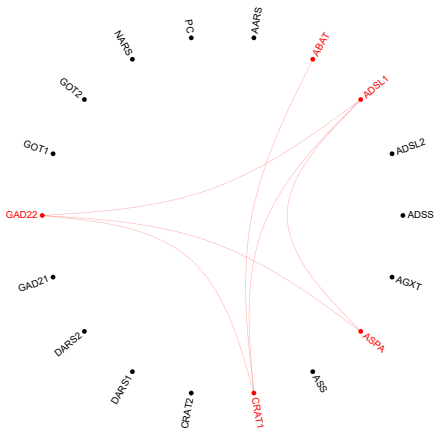
for variable selections in terms of the six measures as well network estimation in terms of the three measures aforementioned in Section 2.6.

We note that there would be a trade-off between the variable selection and the estimation on the network. When the degrees of the correlation is somewhat small such as ρ^*, the pattern of the network is hard to be estimated correctly in both JSKNR method and Glasso. However, as the magnitude of the correlation is high, it can give more accurate estimation but lead to an incorrect result of the variable selection due to multicollinearity.

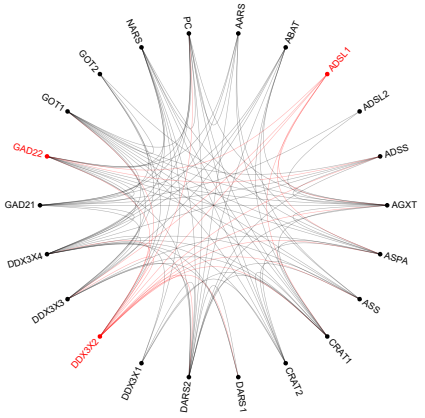
Our approach can be extended to the generalized linear kernel model [33] as well as survival kernel model proposed by Zhang and Kim [55]. However, the algorithm may be slow due to no closed-form solution of parameters. It will be worthwhile to develop a fast algorithm to avoid slower convergence.



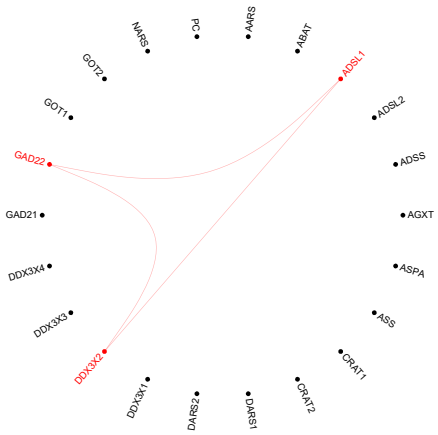
(a) Global gene network of Pathway 4



(b) Selected gene network of Pathway 4

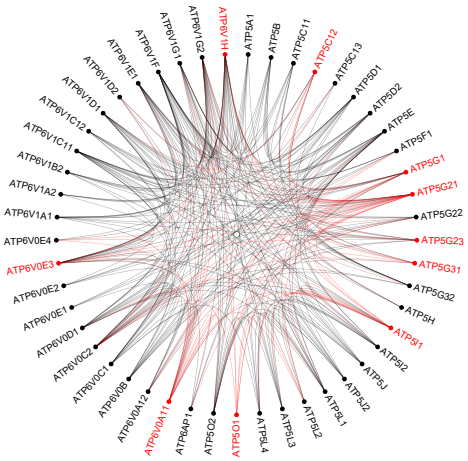


(c) Global gene network of Pathway 140

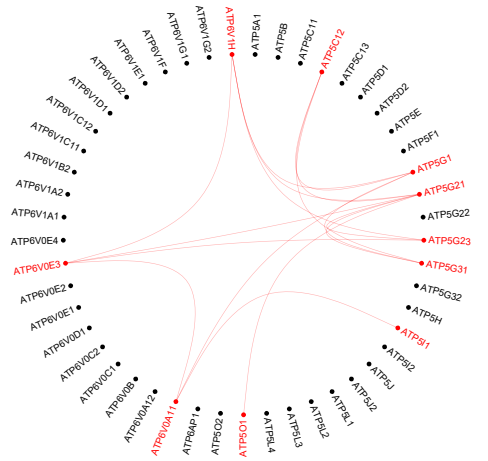


(d) Selected gene network of Pathway 140

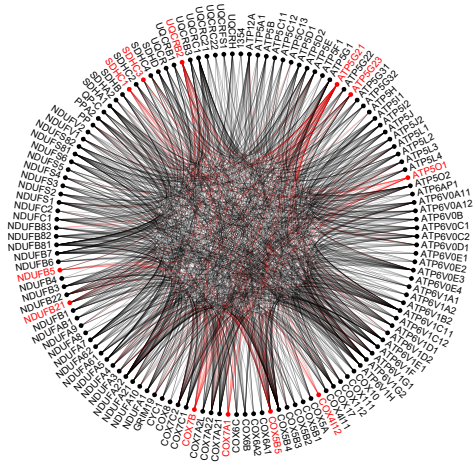
Figure 3.7: Estimated gene network structures of Pathway 4 and 140; The figures (a,c) on the left side provide the estimated network of whole genes based on *Glasso* and the figures (b,d)on the right side represent the estimated network of selected genes based on $JSKNR2$.



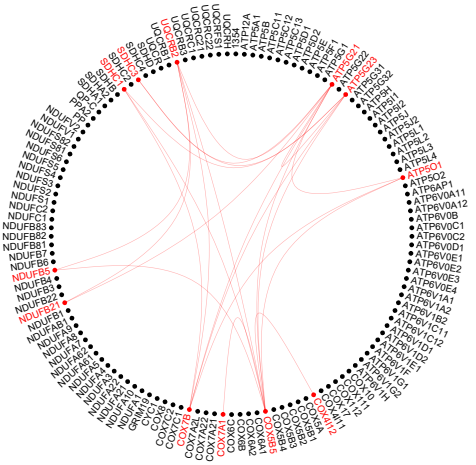
(a) Global gene network of Pathway 16



(b) Selected gene network of Pathway 16



(c) Global gene network of Pathway 229



(d) Selected gene network of Pathway 229

Figure 3.8: Estimated gene network structures of Pathway 16 and 229; The figures (a,c) on the left provide the estimated network of whole genes based on Glasso and the figures (b,d) on the right represent the estimated network of selected genes based on $JSKNR2$.

Chapter 4

Bayesian Focal-area Detection for Multi-class Dynamic Model with Application to Gas Chromatography

4.1 Introduction

Discriminating and quantifying chemical compounds is one of the important applications in analytical chemistry [5]. It is a challenging task because the chemicals often consist of hundreds of compounds as a mixture [16]. For example, gasoline is a complicated mixture consisting of paraffinic, olefinic and aromatic hydrocarbons, and other components containing oxygen and sulfur. The composition of the gasoline is affected by the nature of crude oil, the refining process, the presence of additives, and so on [15].

Gas chromatography (GC) is a very prevalent analytical tool for analyzing the volatile components by separating the chemicals in a complex sample [18]. GC systems monitor the intensity of the gas stream of the chemical compounds over the different time points (called as retention time) given a specific amount of VOCs injected into a narrow tube, known as the

column. Since different chemical constituents in the mixture pass at different rates depending on their various chemical or physical properties, it can obtain the information of individual compounds through the pattern of the intensity against the retention time, which is called a chromatogram [37]. Furthermore, by the introduction of microelectromechanical systems (MEMS) based GC systems, the "electric nose" has been developed for the identification of the chemical. It utilizes an array of chemical sensors and pattern recognition system incorporating the classical GC and odor assessment [49].

Although the performance of the class discrimination has improved by collecting more information, there are still challenging problems with an accurate estimation of gas chromatograms and proper identification of the compounds when there are multiple but unknown classes of the compounds. The mixture of the compounds might cause wrong interpretation by a device and incorrect discrimination. In addition, it has difficulty in recognizing the correct patterns of the compounds due to the unknown and tangled effects of the complex samples. Utilizing an advanced device with high chromatographic resolution might be considered as one of the solutions for proper identification. However, the good separation of the chemical compounds demands a long analysis time so that the identification process is computationally expensive. Thus, instead of using high resolution, it is inevitable to improve the processing software of the device, which is based on statistical analysis to interpret the chromatograms. Hence, it is important to provide a fast and efficient detection tool for the complicated patterns of the gas chromatograms. By extracting its own relevant features and patterns from the data, we can analyze the gas chromatograms.

In this chapter, we build a focal-area detection tool using the Bayesian hierarchical model with shrinkage priors of fused lasso and group fused lasso [4, 29] to elucidate the characteristics of data that has its own vibrational and sequential patterns. Under this Bayesian fused lasso framework, we are able to simultaneously estimate the patterns and identify important time intervals for distinguishing the multiple classes. In addition, we can extract the

common and unique chromatographic patterns for each class by adapting model reformation based on the idea of Bleakey and Vert [4]. Finally, we propose a two-step sequential Bayesian algorithm to estimate patterns as well as detect focal area to characterize each class. Our goal is to statistically detect specific focal areas that play important roles on classifying the different classes. By identifying the focal areas based on historical training sets, our method can have a computationally efficient prediction tool.

Chapter 4 of this dissertation is organized as follows. In Section 4.2, we describe a motivating example from real application on gas chromatography of multi-class gasoline from Fast Odor Chromatographic Sniffer (FOX) system. In Section 4.3, we explain how we reformulate the model with a fused lasso to quantity the common and individual pattern for each class. In Section 4.4, we describe our Bayesian hierarchical fused model framework and then explain how to detect the focal area under the Bayesian framework. In Section 4.5, we then explain how to estimate common global function and unique functions. In Section 4.6, we summarize the simulation results. In Section 4.7, we apply our approach to gas chromatography of multi-class gasoline on the Fast Odor Chromatographic Sniffer (FOX) system. Lastly, our concluding remarks are presented in Section 4.8.

4.2 Gas Chromatographic Data on FOX System

In this section, we describe gas chromatographic data about various types of gasoline generated from the Fast Odor Chromatographic Sniffer (FOX) system. FOX system incorporates gas chromatography (GC) and electric nose (EN). We also delineate the challenging problem in the application. Our goal is to estimate unknown functional forms of gas chromatograms and detect significant areas, enabling the identification of various types of gasoline to be more efficient and accurate based on the estimated trends.

Degree of gasoline adulteration varies depending on the number of low-level solvents

or fuels such as kerosene and diesel in pure fuel. To discriminate the levels of gasoline adulteration in samples, Akbar et al. [2] designed a device of gas chromatographic EN, which has five columns of separation so that it can create five unique chromatograms from a single test injection. In detail, after the injection, the injected sample passes through a flow splitter and is divided into five separate columns. In the experiment, there are two stationary phases chosen for this study by room-temperature ionic liquids, phase A and phase B. These two different phases enable the device to be capable of generating five unique chromatograms from a single test injection [48]. A full chromatographic separation provides a significant amount of information for evaluating the chemical components. However, the time to generate chromatograms with high-resolved peaks requires from 5 minutes to several hours per sample [38]. Thus, detecting the partial chromatograms which have important information for the classification is crucial.

In the experimental study, the data contains the chromatograms from the combination of column and phase. Each chromatogram has 2,000-time points during 2 minute-probation until the end of the separation. For types of gasoline tested in the experimentation, eleven types of gasoline from the combination of two columns (4 and 5) and two phases are considered. The data also includes ten concentrations of kerosene for gasoline adulteration ranges from 0% to 10%, that is, $c \times 100\%$ kerosene in a sample indicates that the sample consists of $c \times 100\%$ kerosene and $(1 - c) \times 100\%$ gasoline with $c \in [0, 10]$. Each type of gasoline has ten replicates.

Motivated by the gas chromatographic data mentioned above, our goal is to estimate unknown common and unique functional trends of multi-class gasoline, considering the hierarchical model framework. Based on the estimation, we focus on detecting significant time intervals that provide useful information for efficient discrimination of gasoline type of the newly tested samples. In practice, classes of the samples are unknown and the properties of the patterns might be quite similar although they have different levels of gasoline. For

example, there are several differences between the patterns among the 11 types of gasoline in the specific time intervals, as shown in Figure 4.1. It has difficulty in distinguishing the different concentrations of gasoline without high-resolved and full chromatogram, but it is highly time-consuming. To avoid misinterpretation and wrong classification, we develop a fast and practical approach for functional estimation and area detection considering accuracy, efficiency, and statistical interpretability. Our proposed method can identify the differences and subtle changes in the sequential trends by using the fused lasso regularization and Bayesian framework. By identifying areas where significant differences among the classes exist, our method can give suggestions to search for the specific areas to conduct the classification of newly tested samples efficiently. Furthermore, Using a Bayesian approach, our proposed method has good interpretability on the estimated functions and parameters. We will discuss it in Section 4.3.

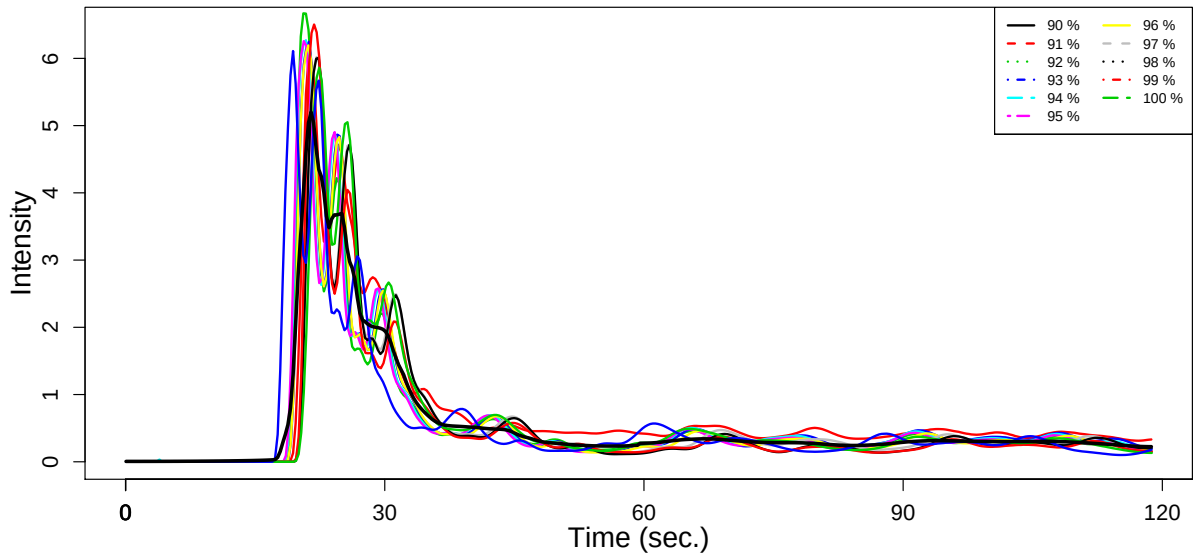


Figure 4.1: Sketch of the gas chromatograms based on 11 classes of gasoline obtained from the column 5 and phase B during the retention time (2 minutes)

4.3 Sequential Model with Multiple Classes

Let y_{ij} be the measurement of intensity from the jth gas stream (profile) at the ith time point where $i = 1, \ldots, n$ within the retention time with the jth class where $j = 1, \ldots, M$. Let β_i be the global(dominant) mean function of the gasoline intensity at the ith time point and γ_{ij} be the individual functional effect with the jth class of gasoline at the ith time point. Then, we consider the following model:

$$y_{ij} = \beta_i + \gamma_{ij} + \epsilon_{ij}, \quad i = 1, ..., n, \ j = 1, ..., M, \tag{4.1}$$

where $\epsilon_{ij} \sim N(0, \sigma_\epsilon^2)$. Here β_i and γ_{ij} are unknown sequential functions, which can be estimated using a piecewise-constant approximation.

In matrix form, equation (4.1) is rewritten as

$$\boldsymbol{Y} = \boldsymbol{B} + \boldsymbol{\Gamma} + \boldsymbol{E}, \tag{4.2}$$

where

$$\boldsymbol{Y} = \begin{pmatrix} \boldsymbol{Y}_{\bullet,1} & \boldsymbol{Y}_{\bullet,2} & \cdots & \boldsymbol{Y}_{\bullet,M} \end{pmatrix} \in \mathbb{R}^{n \times M}, \quad \boldsymbol{B} = \begin{pmatrix} \boldsymbol{\beta} & \boldsymbol{\beta} & \cdots & \boldsymbol{\beta} \end{pmatrix} \in \mathbb{R}^{n \times M},$$
$$\boldsymbol{\Gamma} = \begin{pmatrix} \boldsymbol{\Gamma}_{\bullet,1} & \boldsymbol{\Gamma}_{\bullet,2} & \cdots & \boldsymbol{\Gamma}_{\bullet,M} \end{pmatrix} \in \mathbb{R}^{n \times M}$$

with column vectors $\boldsymbol{Y}_{\bullet,j}$ and $\boldsymbol{\Gamma}_{\bullet,j}$ for $\boldsymbol{Y}$ and $\boldsymbol{\Gamma}$, respectively. For the matrix notations, $A_{i,\bullet}$ denotes the ith row vector of A and $A_{\bullet,j}$ denotes the jth column vector of A for any matrix $A \in \mathbb{R}^{p \times q}$. The random error matrix $\boldsymbol{E} \sim \mathcal{N}(\boldsymbol{0}, \sigma_\epsilon^2 I_{n \times n} \bigotimes I_{M \times M})$. Our primary goal is to investigate the dominant dynamic pattern of gas chromatograms between the classes and detect the important areas where the unique function for each class are able to be distinguished. Based on the property of the gas chromatogram, the pattern of the intensities from

gasoline have many fluctuations with noise, which possibly interrupt the estimation on the functional trend and the discrimination of the class effects.

We estimate $\boldsymbol{B}$ and $\boldsymbol{F}$ using the piecewise-constant approximation. Due to the large amounts of fluctuations and time points (n), we estimate them by imposing a fused lasso prior and a group fused lasso prior, respectively, in order to have the following constraints $\sum_{i=1}^{n-1} |\beta_{i+1} - \beta_i| < t_1$ and $\sum_{i=1}^{n} \sqrt{\sum_{j=1}^{M-1} (\gamma_{i,j+1} - \gamma_{i,j})^2} < t_2$.

For the efficient computation of the solution, we reformulate the parameters by following Bleakey and Vert [4]. That is, we change parameters from $(\boldsymbol{\beta}, \boldsymbol{\Gamma})$ to $(\boldsymbol{U}, \boldsymbol{V})$, given by

$$\begin{aligned} U_0 &= \beta_1 \\ U_i &= \beta_{i+1} - \beta_i, \quad i = 1, \ldots, n-1 \end{aligned}$$

and

$$\begin{aligned} \boldsymbol{V}_0 &= \Gamma_{\bullet,1} \\ \boldsymbol{V}_{\bullet,j} &= \Gamma_{\bullet,j+1} - \Gamma_{\bullet,j}, \quad j = 1, \ldots, M-1, \end{aligned}$$

where $\boldsymbol{V}_0 = (V_{01}, \ldots, V_{0n})^T \in \mathbb{R}^{n \times 1}$.

Then, we can express $\boldsymbol{\beta}$ and $\boldsymbol{\Gamma}$ as a function of $\boldsymbol{U}$ and $\boldsymbol{V}$ as follows:

$$\begin{aligned} \beta_1 &= U_0 \\ \beta_i &= U_0 + \sum_{k=1}^{i} U_k, \quad i = 1, \ldots, n-1. \end{aligned}$$

In vector form, $\boldsymbol{\beta}$ can be expressed as

$$\boldsymbol{\beta} = \mathbf{1}_{n,1} U_0 + X_U U,$$

where $\mathbf{1}_{n,1}$ is a column vector having all 1s and X_U is the $n \times (n-1)$ matrix with entries $X_{U,(i,j)} = 1$ for $i > j$, and 0 otherwise. In the same way, we have

$$\mathbf{\Gamma}^T = \mathbf{1}_{M,1}\boldsymbol{V}_0^T + X_V\boldsymbol{V}^T,$$

where X_V is the $M \times (M-1)$ matrix with entries $X_{V,(i,j)} = 1$ for $i > j$, and 0 otherwise. Let $\boldsymbol{Q}_0$ be $\mathbf{1}_{n,1}U_0 + \boldsymbol{V}_0$, which is a nuisance parameter in the estimation.

Hence, under this reformulation, the fused lasso and group fused lasso prior become to control $\sum_{i=1}^{n-1}|U_i|$ and $\sum_{i=1}^{n}||\boldsymbol{V}_{i,\bullet}||$, respectively. We estimate the differences of the class effects $\boldsymbol{V} \in \mathbb{R}^{n\times(M-1)}$ instead of $\mathbf{\Gamma} \in \mathbb{R}^{n\times M}$ with large n. We will discuss the details of the Bayesian hierarchical model with the reformulation in Section 4.4.

4.4 Bayesian Hierarchical Model

In this section, we explain prior specification for the parameters and the corresponding full conditional distributions.

4.4.1 Prior Specification

We impose the fused lasso prior and the group fused lasso prior for β and γ, respectively. Assuming that the increment parameters, β and γ across the classes are independent and have identical Laplace priors, we can interpret the estimates as the Bayes posterior modes for $\beta_{i+1} - \beta_i$ and $\gamma_{ij+1} - \gamma_{ij}$s [30, 52].

Let $\mathbf{\Gamma}$ be a matrix of the M-class functional effects, where $\mathbf{\Gamma} = (\boldsymbol{\gamma}_1^T, ..., \boldsymbol{\gamma}_M^T)$ and $\boldsymbol{\gamma}_j$, $j = 1, ..., M$ is an $n \times 1$ vector of the jth class's unique effect across the n time points. By following priors settings from Kyung et al. (2010), we specify the following priors:

- The conditional prior for the mean function $\boldsymbol{\beta}|\sigma_\epsilon^2$ is expressed as

$$\pi(\boldsymbol{\beta}|\sigma_\epsilon^2) \propto \exp\left(-\frac{\lambda_1}{\sigma_\epsilon}\sum_{i=1}^{n-1}|\beta_{i+1}-\beta_i|\right). \tag{4.3}$$

- The conditional prior for the functional random effect $\boldsymbol{\Gamma}|\sigma_b^2$ is as follows:

$$\pi(\boldsymbol{\Gamma}|\sigma_b^2) \propto \exp\left(-\frac{\lambda_2}{\sigma_b}\sum_{i=1}^{n}\sqrt{\sum_{j=1}^{M-1}(\gamma_{i,j+1}-\gamma_{i,j})^2}\right). \tag{4.4}$$

Under the reformulation, the corresponding conditional Laplace priors for $\boldsymbol{\beta}$ and $\boldsymbol{\Gamma}$ become

- The conditional priors for U

$$\pi(\boldsymbol{U}|\sigma_\epsilon^2) \propto \exp\left(-\frac{\lambda_1}{\sigma_\epsilon}\sum_{i=1}^{n-1}|U_i|\right) \tag{4.5}$$

- The conditional priors for V

$$\pi(\boldsymbol{V}|\sigma_b^2) \propto \exp\left(-\frac{\lambda_2}{\sigma_b}\sum_{i=1}^{n}||V_{i,\bullet}||\right) \tag{4.6}$$

Based on the property of the Laplace prior, the conditional priors (4.5) and (4.6) are rewritten as follows:

-

$$\boldsymbol{U}|\sigma_\epsilon^2,\tau_1^2,...,\tau_{n-1}^2 \sim \mathcal{N}_{n-1}\Big(\mathbf{0}_{n-1,1},\sigma_\epsilon^2\boldsymbol{D}_\tau\Big), \quad \boldsymbol{D}_\tau = \text{diag}(\tau_1^2,...,\tau_{n-1}^2) \in \mathbb{R}^{(n-1)\times(n-1)},$$
$$\tau_i^2|\lambda_1 \sim \text{Gamma}\left(1,\frac{\lambda_1^2}{2}\right), \quad i=1,\ldots,n-1,$$

•

$$\boldsymbol{V}_{i,\bullet}|\sigma_b^2,\omega_i^2 \sim \mathcal{N}_{M-1}\Big(\mathbf{0}_{M-1,1},\sigma_b^2\boldsymbol{D}_{\omega_i}\Big), \quad \boldsymbol{D}_{\omega_i}=\text{diag}(\omega_i^2,...,\omega_i^2)\in\mathbb{R}^{(M-1)\times(M-1)},$$
$$\omega_i^2|\lambda_2 \sim \text{Gamma}\left(\frac{M}{2},\frac{\lambda_2^2}{2}\right),$$

Then, the Bayesian hierarchical model can be written as the follows:

$$\begin{aligned}
\boldsymbol{y}|Q_0,X_U,X_V,U,V,\sigma_\epsilon^2 &\sim \mathcal{N}_{nM}\left(\boldsymbol{\mu},\sigma_\epsilon^2\boldsymbol{I}_{nM}\right),\\
\boldsymbol{U}|\sigma_\epsilon^2,\tau_1^2,...,\tau_{n-1}^2 &\sim \mathcal{N}_{n-1}\Big(\mathbf{0}_{n-1,1},\sigma_\epsilon^2\boldsymbol{D}_\tau\Big), \quad \boldsymbol{D}_\tau=\text{diag}(\tau_1^2,...,\tau_{n-1}^2)\in\mathbb{R}^{(n-1)\times(n-1)},\\
\tau_i^2|\lambda_1 &\sim \text{Gamma}\left(1,\frac{\lambda_1^2}{2}\right), \quad i=1,\ldots,n-1,\\
\boldsymbol{V}_{i,\bullet}|\sigma_b^2,\omega_i^2 &\sim \mathcal{N}_{M-1}\Big(\mathbf{0}_{M-1,1},\sigma_b^2\boldsymbol{D}_{\omega_i}\Big), \quad \boldsymbol{D}_{\omega_i}=\text{diag}(\omega_i^2,...,\omega_i^2)\in\mathbb{R}^{(M-1)\times(M-1)},\\
\omega_i^2|\lambda_2 &\sim \text{Gamma}\left(\frac{M}{2},\frac{\lambda_2^2}{2}\right),\\
\sigma_\epsilon^2 &\sim \pi(\sigma_\epsilon^2), \quad \sigma_b^2 \sim \pi(\sigma_b^2),\\
\lambda_1 &\sim \text{Gamma}\left(\rho_1,\delta_1\right), \quad \lambda_2 \sim \text{Gamma}\left(\rho_2,\delta_2\right),
\end{aligned}$$

where (ρ_1,δ_1) and (ρ_2,δ_2) are the sets of hyper-parameters for λ_1 and λ_2, respectively. We can use non-informative priors or inverse gamma priors for $\pi(\sigma_\epsilon^2)$ and $\pi(\sigma_b^2)$ such as $\pi(\sigma^2)=1/\sigma^2$ or $\pi(\sigma^2)\sim \text{Inv.Gamma}(a_{\sigma^2},b_{\sigma^2})$, where $(a_{\sigma^2},b_{\sigma^2})$ is a set of hyper-parameters. With the above specification of prior distributions, we can jointly estimate the parameters using the Gibbs sampler. The tuning parameters, λ_1 and λ_2, can also be updated by using the estimates through marginal likelihood with a hybrid of EM and Gibbs algorithm [29, 43]. Let $\lambda^{(k)}$ be the estimate from iteration k. Then, the estimates of λ_1 and λ_2 at the k-th iteration are

given by

$$\lambda_1^{(k)} = \sqrt{\frac{2(n-1)}{\sum_{i=1}^{n-1} E_{\lambda_1^{(k-1)}}[\tau_i^2|\boldsymbol{y}_c]}}, \quad \lambda_2^{(k)} = \sqrt{\frac{2n}{\sum_{i=1}^{n} E_{\lambda_2^{(k-1)}}[\omega_i^2|\boldsymbol{y}_c]}},$$

respectively at the M-step in the EM algorithm. Here, $\boldsymbol{y}_c$ is a centered response variable such that $\boldsymbol{y}_c = \boldsymbol{y} - \bar{\boldsymbol{y}}$.

4.4.2 Full Conditional Distributions

Because the normal distribution based likelihood function and the conjugate priors setting, the full conditional distributions for all parameters have the closed forms as follows:

- The full conditional distribution of $\boldsymbol{U}$ follows $\mathcal{N}_{n-1}(\boldsymbol{\mu}_U, \Sigma_U)$, where

$$\boldsymbol{\mu}_U = \left(\boldsymbol{D}_\tau^{-1} + X_U^T X_U\right)^{-1} \sum_{j=1}^{M} X_U^T \left(\boldsymbol{Y}_{\bullet,j} - \boldsymbol{\mu}_{\bullet,j(-U)}\right),$$
$$\Sigma_U = \sigma_\epsilon^2 \left(\boldsymbol{D}_\tau^{-1} + X_U^T X_U\right)^{-1},$$

  $\boldsymbol{\mu}_{(-a)}$ means the overall average, excluding a parameter a.

- The full conditional posterior of $\boldsymbol{V}_{i,\bullet}$ is $\mathcal{N}_{M-1}\left(\boldsymbol{\mu}_{V_{i,\bullet}}, \Sigma_V\right)$, where

$$\boldsymbol{\mu}_{V_{i,\bullet}} = \left(\frac{\boldsymbol{D}_\omega^{-1}}{\sigma_b^2} + \frac{X_V^T X_V}{\sigma_\epsilon^2}\right)^{-1} \sum_{l=1}^{n} X_V^T \left(\boldsymbol{Y}_{l,\bullet} - \boldsymbol{\mu}_{l,\bullet(-V)}\right), \quad i = 1, ..., n-1,$$
$$\Sigma_{V_{i,\bullet}} = \left(\frac{\boldsymbol{D}_\omega^{-1}}{\sigma_b^2} + \frac{X_V^T X_V}{\sigma_\epsilon^2}\right)^{-1}.$$

- The full conditional distributions for τ_i^2 and ω_j^2 are inverse Gaussian distributions as

follows:

$$[\tau_i^{-2}|-] \sim \text{Inv.Gaussian}\left(\sqrt{\frac{\lambda_1^2\sigma_\epsilon^2}{U_i^2}}, \lambda_1^2\right)$$

and

$$[\omega_i^{-2}|-] \sim \text{Inv.Gaussian}\left(\sqrt{\frac{\lambda_2^2\sigma_b^2}{||V_{i,\bullet}||^2}}, \lambda_2^2\right),$$

where "$-$" denotes all other parameters conditioned in the full conditional distribution.

- The full conditional distributions for σ_ϵ^2 and σ_b^2 are

$$[\sigma_\epsilon^2|-] \sim \text{Inv.Gamma}\left(\nu_{\sigma_\epsilon^2}, \eta_{\sigma_\epsilon^2}\right),$$

where a scale parameter $\nu_{\sigma_\epsilon^2}$ and a shape parameter $\eta_{\sigma_\epsilon^2}$, where

$$\nu_{\sigma_\epsilon^2} = \frac{nM+n-1}{2} + a_{\sigma_\epsilon^2},$$
$$\eta_{\sigma_\epsilon^2} = \left(\frac{1}{2}\sum_{j=1}^{M}\left(\boldsymbol{Y}_{\bullet,j} - X_U U - \boldsymbol{\mu}_{\bullet,j(-U)}\right)^T\left(\boldsymbol{Y}_{\bullet,j} - X_U U - \boldsymbol{\mu}_{\bullet,j(-U)}\right) + \frac{1}{2}U^T\boldsymbol{D}_\tau^{-1}U + b_{\sigma_\epsilon^2}\right)$$

and

$$[\sigma_b^2|-] \sim \text{Inv.Gamma}\left(\nu_{\sigma_b^2}, \eta_{\sigma_b^2}\right),$$

where

$$\nu_{\sigma_b^2} = \frac{M-1}{2} + a_{\sigma_b^2},$$
$$\eta_{\sigma_b^2} = \frac{1}{2}\sum_{i=1}^{n}\left(\boldsymbol{V}_{i,\bullet}^T\boldsymbol{D}_{\omega_i}^{-1}\boldsymbol{V}_{i,\bullet}\right) + b_{\sigma_b^2}.$$

If $(a_{\sigma_\epsilon^2}, b_{\sigma_\epsilon^2}) = (a_{\sigma_b^2}, b_{\sigma_b^2}) = (0, 0.1)$, the priors for σ_ϵ^2 and σ_b^2 are weakly informative priors. For the tuning parameters, λ_1 and λ_2, we get Gamma distributions based on the conjugacy as follows:

$$[\lambda_1^2|-] \sim \text{Gamma}\left(n + \rho_1 - 1, \frac{1}{2}\sum_{i=1}^{n-1} \tau_i^2 + \delta_1\right)$$

and

$$[\lambda_2^2|-] \sim \text{Gamma}\left(\frac{n(M-1)}{2} + \rho_2, \frac{1}{2}\sum_{i=1}^{n} \omega_i^2 + \delta_2\right).$$

The hyperparameters for λ_1 and λ_2 can be chosen by using prior knowledge and analyzing historical data. If it is unknown, we follow the suggestion of Kyung et al. [29] such as using shape parameter ρ as 1 and scale parameter δ as 0.1, which give more weights on the data and lead to estimate near the MLE with high posterior probability. Since the full conditional distributions have closed forms and hyperparameters are set using the closed form as well, we can use an efficient Gibbs sampling procedure. The derivation of the full conditional distributions are described in Appendix C.1.

4.4.3 Bayesian Focal-Area Detection

Using the Gibbs sampler from the full conditional distributions, we can identify focal area detection to determine which intervals are significantly distinguishable between classes by calculating Bayesian credible intervals.

We construct the sequential credible interval for γ_{ij}, which is the jth unique class effect. Because the unique class effect $\boldsymbol{\Gamma}$ is the linear transformation of $\boldsymbol{V}$, the maximum a posteriori probability (MAP) estimate of $\boldsymbol{V}$ from the Gibbs sampler satisfies the invariance under the reparameterization corresponding with the condition. Thus, we can obtain the MAP estimate

of $\mathbf{\Gamma}$ and construct the credible intervals for each class effect. Then, the significant intervals can be determined based on the credible intervals.

The detailed procedure of the focal area detection is as follows:

Step 1: Compute the maximum a posteriori probability (MAP) estimate of $\boldsymbol{V}$, denoted by $\widehat{\boldsymbol{V}}_{MAP}$ based on the posterior samples from the Gibbs sampler.

Step 2: Transform $\widehat{\boldsymbol{V}}_{MAP}$ to $\widehat{\mathbf{\Gamma}}$ such that $\widehat{\mathbf{\Gamma}}_{MAP} = \widehat{\boldsymbol{V}}_{MAP} X_V^T$.

Step 3: Draw sufficiently many samples from the posterior distribution of $\mathbf{\Gamma} \sim \mathcal{N}_{n\times M}(\boldsymbol{\mu}_V X_V^T, X_V \Sigma_V X_V^T)$.

Step 4: Construct the $(1-\alpha)100\%$ credible intervals for each $\mathbf{\Gamma}_{\bullet,j}$ based on the $\alpha/2$th and $1-\alpha/2$th percentiles from the posterior samples with $j = 1, ..., M$.

Step 5: Determine statistically significant points or intervals where at least one credible interval does not overlap with the others such that

$$\mathcal{A} = \left\{ i | \prod_{j \neq j'}^{M} I(\gamma_{L,j} < \hat{\gamma}_{i,j'} < \gamma_{U,j}) = 0, \quad \forall j = 1, \ldots, M \right\}, \quad i = 1, \ldots, n,$$

where $\gamma_{L,j}$ and $\gamma_{U,j}$ are the lower and upper bounds of the credible interval for the functional effect of the jth class and $I(\bullet)$ is an indicator function.

Based on the set of the significant points $\mathcal{A}$, we can conclude that the observations within the focal area reflect the unique property of the corresponding class and can facilitate the efficient classification in the follow-up experiment. We refer this procedure to *Bayesian Focal-Area Detection* (BFAD).

4.5 Implementation

Although the Gibbs sampler is fast and efficient in drawing posterior samples using the closed forms of the full conditional distributions, there are some numerical issues in the sampling due to τ_i^2, ω_i^2, and two hyperparameters λ_1 and λ_2. Roualdes [50] addressed that the shrinkage parameters τ_i^2 and ω_i^2 can cause computational issues in the Bayesian framework because of the inversion of the parameters sampled from the inverse-Gaussian distribution. There is a possibility that any posterior sample from the inverse-Gaussian distribution have numerical instability when the estimate of the corresponding penalty term is close to zero. Furthermore, the updates of the two parameters with the hyperparameters could have computationally slower convergence of the parameters than that on a model having only one shrinkage prior.

To ease such numerical issues, we propose two-step Bayesian sequential updating procedures. In the first step, we estimate the global mean function across multiple classes using the fused lasso priors, given unique class effects. In the second step, given the common global function, we estimate the unique class effect with the group fused lasso priors. This two-step functional estimation can lead to relatively stable convergence of the parameters and relieve the numerical issues.

4.5.1 Bayesian Estimation on Global Mean Function

We estimate a global mean function using the Bayesian hierarchical model with the fused lasso prior, given the class effects $\mathbf{\Gamma}$. Given the fixed initial or estimated $\tilde{\mathbf{\Gamma}}$, we reduce the model. We proceed with the following procedure to estimate global mean function:

Step1: Standardize the data $\boldsymbol{y}$ and initialize the hyperparameters, $\rho_1, \delta_1, a_\epsilon^{(0)}$, and $b_\epsilon^{(0)}$ as follows:

a. Standardize $\boldsymbol{y}_j$ and denote it as $\boldsymbol{y}_{std,j}$, $\quad j = 1 \dots M$.

b. Initialize $\widehat{\boldsymbol{\gamma}}^{(itr=0)}$ as $\mathbf{0} \in \mathbb{R}^{n\times M}$. Otherwise, calculate $\boldsymbol{y}_{std} - \widehat{\boldsymbol{\gamma}}^{(itr>0)}$, denoted by $\boldsymbol{y}^{*}_{std}$.

Step 2: Set the initial values for the parameters $\boldsymbol{U}^{(0)}, \sigma_{\epsilon}^{2(0)}$, and $\tau^{(0)}$ as follows:

a. Set $\boldsymbol{X}_c\ (I_{n\times n} - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\boldsymbol{X} \in \mathbb{R}^{n\times(n-1)}$.

b. Use the least-squares estimates for $\boldsymbol{U}^{(0)}$ such that $\widehat{\boldsymbol{U}}_j^{(0)}$ using $(\boldsymbol{X}_c^T\boldsymbol{X}_c)^{-1}\boldsymbol{X}_c^T\boldsymbol{y}^{*}_{std,j} \in \mathbb{R}^{(n-1)\times 1}, \quad j = 1, ..., M$.

c. Update residuals $\boldsymbol{r}_j$ by using $\boldsymbol{y}^{*}_{std,j} - \boldsymbol{X}_c\widehat{\boldsymbol{U}}_j \in \mathbb{R}^{n\times 1}$.

d. Set $\sigma_{\epsilon}^{2(0)}$ using $\frac{1}{nM}\sum_{j=1}^{M}\boldsymbol{r}_j^T\boldsymbol{r}_j$.

e. Set $\tau_{ij}^{-2(0)}$ using $\boldsymbol{U}_{ij}^{-2(0)}, \qquad i = 1, ..., n-1$.

Step 3: For $k = 1$ to K, do the Gibbs sampling as follows:

a. Sample $\boldsymbol{U}_j^{(k)}$ from $N\left(\mu_{U_j}^{(k)}, \Sigma_{U_j}^{(k)}\right)$ with the update of $\mu_{U_j}^{(k)}$ and $\Sigma_{U_j}^{(k)}$.

b. Sample $\sigma_{\epsilon}^{2(k)}$ from Inv.Gamma$\left(\nu_{\sigma_{\epsilon}^2}^{(k)}, \eta_{\sigma_{\epsilon}^2}^{(k)}\right)$ with the update of $\nu_{\sigma_{\epsilon}^2}^{(k)} = \frac{nM+n-1}{2} + a_{\sigma_{\epsilon}^2}^{(0)}$ and $\eta_{\sigma_{\epsilon}^2}^{(k)} = \frac{1}{2}\sum_{j=1}^{M}\left(\boldsymbol{Y}_{\bullet,j} - X_U U^{(k)} - \boldsymbol{\mu}_{\bullet,j(-U)}\right)^T\left(\boldsymbol{Y}_{\bullet,j} - X_U U^{(k)} - \boldsymbol{\mu}_{\bullet,j(-U)}\right) + \frac{1}{2}U^{(k)T}\boldsymbol{D}^{-1}_{\tau^{(k-1)}}U^{(k)} + b_{\sigma_{\epsilon}^2}^{(0)}$.

c. Sample ${\tau_j}^{-2(k)}$ from Inv.Gauss$\left(\mu_{\tau_j}^{(k)}, \Sigma_{\tau_j}^{(k)}\right)$. with the update $\mu_{\tau_j}^{(k)}$ and $\Sigma_{\tau_j}^{(k)}$.

d. Sample $\lambda_{1,j}^{(k)}$ from Gamma $\left(\rho_1 + n - 1, \delta_1 + \frac{1}{2}\sum_{i=1}^{n-1}\tau_{ij}^{-2(k)}\right)$.

Step 4: Use the samples for the parameters after the burn-in period and obtain the MAP estimates of the parameters from the samples.

Step 5: Update $\widehat{\boldsymbol{\beta}}_{MAP}$ using $\boldsymbol{X}_c\widehat{\boldsymbol{U}}_{MAP}$.

Step 6: Obtain $\bar{\boldsymbol{\beta}}^{(itr)} = \frac{1}{M} \times \sum_{j=1}^{M}\left(\widehat{\boldsymbol{\beta}}_{MAP}\right) \in \mathbb{R}^{n\times 1}$.

4.5.2 Bayesian Estimation on Functional Class Effects

In the second step, we estimate unique class effects using the group fused lasso prior. Given the estimated global mean function $\widehat{\boldsymbol{\beta}}$, the solution of $\boldsymbol{\Gamma}$ can be obtained by minimizing the following procedure:

Step1: Standardize the data $\boldsymbol{y}$ and initialize the hyperparameters, $\rho_2, \delta_2, a_b^{(0)}$, and $b_b^{(0)}$ as follows:

a. Standardize $\boldsymbol{y}_j$ and denote it as $\boldsymbol{y}_{std,j}, \quad j = 1 \dots M$.

b. Update $\boldsymbol{y}_{std,c}$ using $\boldsymbol{y}_{std} - \bar{\boldsymbol{\beta}}^{(itr)} \mathbf{1}_M^T \in \mathbb{R}^{n \times M}$ and vectorize $\boldsymbol{y}_{std,c}^T$, denoted as $\boldsymbol{y}_c \in \mathbb{R}^{nM \times 1}$.

Step 2: Set the initial values for the parameters $\boldsymbol{V}^{(0)}, \sigma_b^{2(0)}$, and $\omega^{(0)}$ as follows:

a. Compute the centered matrix, $\boldsymbol{X}_{\gamma,c} \in \mathbb{R}^{nM \times n(M-1)}$.

b. Use the least squares estimates for $\boldsymbol{V}^{(0)}$ such that $\widehat{\boldsymbol{V}}^{(0)}$ using $(\boldsymbol{X}_{\gamma,c}^T \boldsymbol{X}_{\gamma,c})^{-1} \boldsymbol{X}_{\gamma,c}^T \boldsymbol{y}_c \in \mathbb{R}^{n(M-1) \times 1}$.

c. Compute residuals $\boldsymbol{r}_\gamma$ by $\boldsymbol{y}_c - \boldsymbol{X}_{\gamma,c} \widehat{\boldsymbol{V}} \in \mathbb{R}^{nM \times 1}$.

d. Set $\sigma_b^{2(0)}$ using $\frac{1}{nM} \boldsymbol{r}_\gamma^T \boldsymbol{r}_\gamma$.

e. Set $\omega_i^{-2(0)}$ using $\boldsymbol{\gamma}_i^{-2(0)}, \qquad i = 1, ..., n-1$.

Step 3: For $k = 1$ to K, do the Gibbs sampling as follows:

a. Sample $\boldsymbol{V}_j^{(k)}$ from $N\left(\mu_V^{(k)}, \Sigma_V^{(k)}\right)$ with the update of $\mu_V^{(k)}$ and $\Sigma_V^{(k)}$.

b. Sample $\sigma_b^{2(k)}$ from Inv.Gamma$\left(\nu_{\sigma_b^2}^{(k)}, \eta_{\sigma_b^2}^{(k)}\right)$ with the update of $\nu_{\sigma_b^2}^{(k)} = \frac{M-1}{2} + a_{\sigma_b^2}^{(0)}$ and $\eta_{\sigma_b^2}^{(k)} = \sum_{i=1}^n \left(\boldsymbol{V}_{i,\bullet}^{(k)T} \boldsymbol{D}_{\omega_i^{(k-1)}}^{-1} \boldsymbol{V}_{i,\bullet}^{(k)} \right) + b_{\sigma_b^2}^{(0)}$.

c. Sample $\omega^{-2(k)}$ from Inv.Gauss$\left(\mu_\omega^{(k)}, \Sigma_\omega^{(k)}\right)$. with the update $\mu_\omega^{(k)}$ and $\Sigma_\omega^{(k)}$.

d. Sample $\lambda_2^{(k)}$ from Gamma $\left(\frac{n(M-1)}{2}+\rho_2, \delta_2+\frac{1}{2}\sum_{i=1}^{n(M-1)}\omega_i^{-2(k)}\right)$.

Step 4: Use the samples for the parameters after the burn-in period and obtain the MAP estimates of the parameters from the samples.

Step 5: Update $\widehat{\boldsymbol{\gamma}}_{MAP}$ using $\boldsymbol{X}_{\gamma,c}\widehat{\boldsymbol{V}}_{MAP}$.

Step 6: Transform the vector to the matrix; $\widehat{\boldsymbol{\gamma}}_{MAP} \in \mathbb{R}^{nM\times 1} \rightarrow \widehat{\boldsymbol{\Gamma}}^{(itr)} \in \mathbb{R}^{n\times M}$.

We iterate the two-step sequential procedures until the parameters converge. If we let $\boldsymbol{\psi} = (\bar{\boldsymbol{\beta}}, \boldsymbol{\Gamma}, \sigma_\epsilon^2, \sigma_b^2)$, the convergence is determined by computing the mean squared differences (MSDs) of $\boldsymbol{\psi}$ from the previous and current iterations as below. Suppose ϕ be a $n\times 1$ vector. Then, we have

$$MSD(\phi) = \frac{1}{n}\sum_{i=1}^{n}\left(\phi^{(itr)} - \phi^{(itr-1)}\right)^2.$$

The estimates of the parameters are obtained when the MSD is less than a specified value of tolerance. After the two-step sequential procedures, the posterior distribution of $\boldsymbol{\Gamma}$ can be derived based on the posterior distribution of $\boldsymbol{V}$ and then, the credible intervals for each class effect can be computed from the full conditional distribution. Finally, using the credible intervals, we identify the significant sequential points, where the unique class effects are statistically identifiable. The details of constructing the credible intervals are described in Algorithm 2. Figure 4.2 illustrates the summary of our proposed two-step sequential algorithm.

4.6 Simulation Study

We conduct simulations to investigate the performance of the proposed Bayesian focal-area detection using a Bayesian hierarchical model with fused lasso and group fused lasso. Mo-

Algorithm 2 Algorithm of two-step Bayesian focal-area detection

Require: $\boldsymbol{y}$, δ_1, δ_2, ρ_1, ρ_2

Initialize $\mathcal{A} \leftarrow \emptyset$.

Standardize $\boldsymbol{y}_j$ to get $\boldsymbol{y}_{std,j}, \quad j = 1 \dots M$

for $itr = 1$ to $max.Itr$ **do**

  if $itr = 1$ **then**

    $\widehat{\boldsymbol{\gamma}}^{(itr)} \leftarrow \mathbf{0} \in \mathbb{R}^{n \times M}$

  end

  Estimate a global mean function, $\bar{\boldsymbol{\beta}}^{(itr)}$ from Phase I (ref).

  if $itr = 1$ **then**

  Estimate functional random cluster effects, $\widehat{\boldsymbol{\Gamma}}^{(itr)}$ from Phase II (ref).

  Compute mean squared differences (MSDs) of the parameters $\boldsymbol{\psi} = (\bar{\boldsymbol{\beta}}, \boldsymbol{\Gamma}, \sigma_\epsilon^2, \sigma_b^2)$ from the previous and current iterations as below. Suppose ϕ be a $n \times 1$ vector. Then, we have

$$MSD(\phi) = \frac{1}{n} \sum_{i=1}^{n} \left(\phi^{(itr)} - \phi^{(itr-1)} \right)^2 .$$

$max.MSD \leftarrow \max\left(MSD(\boldsymbol{\psi})\right)$

  if $max.MSD < tol.$ **then**

    Compute $(1-\alpha)100\%$ credible intervals for $\boldsymbol{\Gamma}_{\bullet,j}, \quad j = 1, \dots, M.$

  end

end

Update the set of significant points and intervals, $\mathcal{A}$ where at least one credible interval does not overlap with the others.

$$\mathcal{A} \leftarrow \mathcal{A} \cup \left\{ i | \prod_{j \neq j'}^{M} I(\gamma_{L,j} < \hat{\gamma}_{i,j'} < \gamma_{U,j}) = 0, \quad \forall j = 1, \dots, M \right\}, \quad i = 1, \dots, n,$$

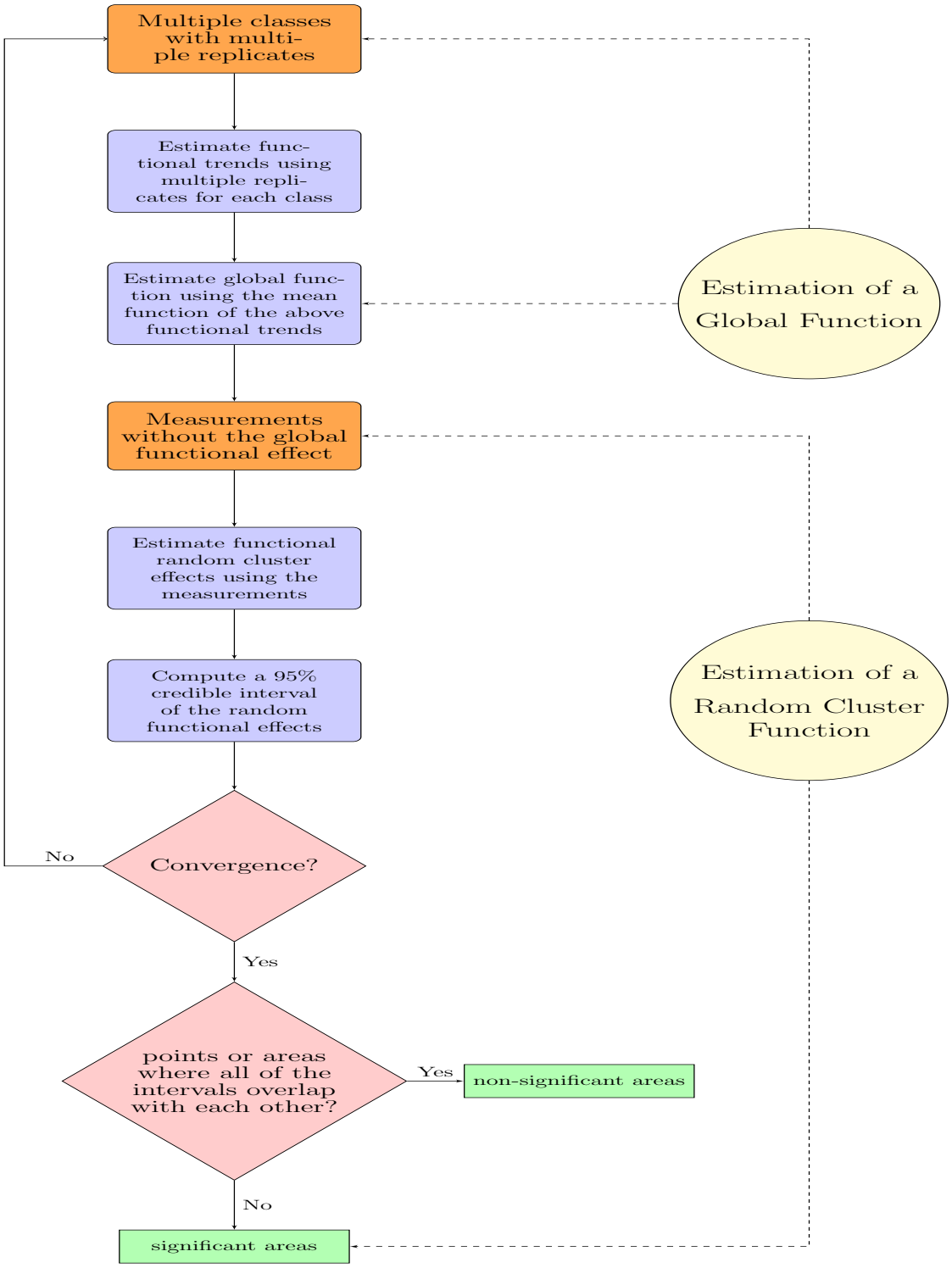


Figure 4.2: Flowchart of Bayesian focal-area detection

tivated by our gas chromatographic data, we generate a toy data including the functional trend for each class with the number of classes $M = 11$, the number of replicates for each class $R = 20$, and the total number of time points $n = 100, 200, 400, 1,000$, and $2,000$. In generating data, we consider the following function:

$$y_{ijr} = f_{ijr} + \epsilon_{ijr} = \beta_{ij} + \gamma_{ijr} + \epsilon_{ijr},$$

where $\epsilon_{ijr} \sim N(0, \sigma_\epsilon^2)$ with $\sigma_\epsilon^2 = 0.25$ $i = 1, ..., n$, $j = 1, ..., M$, and $r = 1, ..., R$. The function f consists of global mean function β and individual class effect γ, which are generated from our real data. We assume that the time points are equally spaced and the class effects are independent with each other. We conduct Bayesian estimation of the functional trends with the Laplace priors of fused lasso and group fused lasso regularization using a combination of two classes among the multiple classes. That is, the Bayesian estimation is performed multiple times for all combinations of two classes.

Because our primary goal is to detect significant time intervals for an efficient discrimination of multi-class samples, we examine if the time points or intervals identified by the BFAD method could improve the performance of the classification. We employ partial least squares discriminant analysis (PLS-DA), which is one of the powerful methods in the classification. The PLS-DA amends the projection direction of LDA by using the information of PLS, making the result closer to the optimal direction [6]. The PLS-DA scores have been used as the best classifiers, which discriminate two classes [40]. We compare the performance of the classification between two cases that overall time points and the significant time points are used in the PLS-DA method.

For each simulated data set, we run the two-step Bayesian sequential procedures described in Section 4.5. For each setting with the various number of time points, we simulate 200 data sets for each class. We run 2,000 MCMC iterations using the Gibbs sampler for each

simulated data set and set a burn-in time at 200 runs. For the prior of σ_ϵ^2 and σ_b^2, the noninformative prior is used. For the prior of the tuning parameters, we use a gamma distribution with the hyperparameters for shape and scale parameters as $(a_\lambda, b_\lambda)=(1, 0.5)$, respectively. After drawing the posterior samples from the Gibbs sampler, we construct a 95% Bayesian credible interval for a difference between the two functional class effects and identify the significant time points based on the proposed method.

To evaluate the performance of the BFAD method in comparison, we calculate computation time, memory usage, and predictive accuracy based on PLS-DA. The computation time and memory profiling is performed by *Rprof* function in statistical software *R*. The run time and memory allocation are measured in seconds and megabytes (MB), respectively. In terms of predictive accuracy, we create a training set and a test set and calculate the predictive accuracy using 10-fold cross-validation with the PLS-DA method. The predictive accuracy is calculated given by

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN},$$

where TP and FP denote true positive and false positive, respectively. TN and FN denote true negative and false negative, respectively. We also calculate the relative efficiency for each measure in order to see the improvement of the classification by focusing on only the significant time points.

Figure 4.3 displays the MCMC trace plots of the Gibbs samples for each parameter. We see that all chains produce a stable convergence of the parameters. Figure 4.4 – 4.6 show the results of the functional estimation of global mean function, functional class effects, and the 95% Bayesian credible intervals for each difference between the two classes. We consider the comparison using cluster 1 vs. 2, cluster 1 vs. 8, and cluster 4 vs. 6, for example. From the results, we see that the group fused lasso prior shrinks the slight differences between the

two class effects to zero. The significant time points are determined by the credible intervals where the interval estimates do not cover zero like the time intervals between the red dashed lines in Figure 4.4c, 4.5c, and 4.6c. Let $\mathcal{A}_{g,h}$ be a set of the significant time points using the gth cluster and hth cluster. The BFAD method identified $\mathcal{A}_{1,2} = \{i|145 \leq i \leq 178\}$, $\mathcal{A}_{1,8} = \{i|147 \leq i \leq 162\}$, and $\mathcal{A}_{4,6} = \{i|i = 181, 182, 183\}$ as the sets of the statistically significant time points. We also investigate the consistency of detecting the significant time points by simulating a selection probability based on relative frequency using 200 simulated data sets for each setting. Figure 4.7 shows the selection probabilities for each comparison. We see that the time intervals are selected with high probabilities over 75%, which is predefined.

After identifying the time points based on the BFAD method, we compare the performance of PLS-DA using only the significant time points with that using the total time points. Table 4.1 summarizes the computation times, the amounts of memory use, and predictive accuracy. It shows that our BFAD method enables PLS-DA to outperform the classification using the total time points while it maintains comparable predictive accuracy. Figure 4.8 describes the relative efficiency of the performances between the two classifications. When the total of the time points is 2,000, the run time of the classification based on the BFAD method is two time faster and save about 60% of the memory, compared to the PLS-DA utilizing the total time points. Therefore, our Bayesian focal-area detection method can facilitate the efficient classification in terms of high speed of the computation and efficiency of memory.

Table 4.1: Summary of computation time, memory usage, and prediction accuracy based on PLS-DA utilizing the total time points(n) and the significant time points obtained from BFAD method; The computation time and memory allocation are measured in a second and a megabyte.

n	Time (total, sec.)	Time (BFAD, sec.)	Memory (total, MB)	Memory (BFAD, MB)	Acc (total, %)	Acc (BFAD, %)
100	4.67	4.43	1299.6	1081.7	97.5	100
200	6.68	5.96	1510.2	1114.2	100	100
400	6.22	5.28	1943.9	1129.4	100	100
1000	8.66	6.68	3318.9	1451.9	100	100
2000	13.12	6.06	5220.3	1460.8	100	100

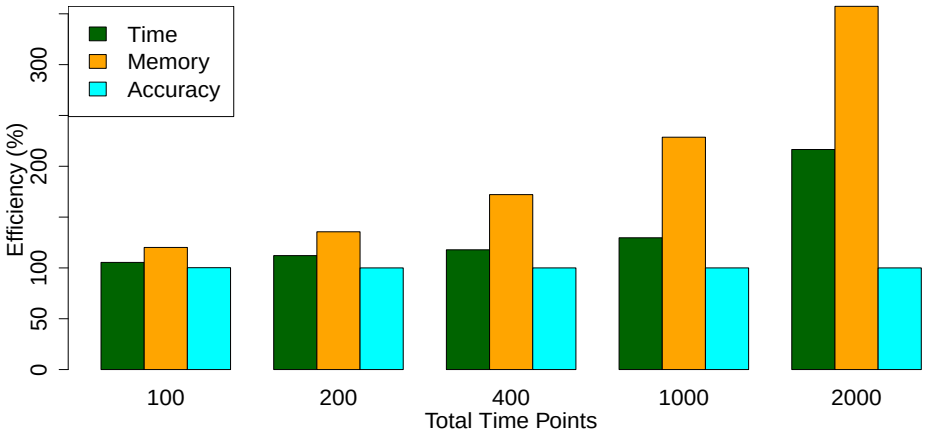


Figure 4.8: Bargraph of the relative efficiency of PLSDA using between total time points and significant time points based on the BFAD method; The efficiency includes computation time (dark green), memory usage (orange), and accuracy (cyan) in percentage with the total time points

4.7 Application

We apply our Bayesian focal-area detection method to the data collected from the chromatographic analyses, which are described in Section 4.2. For the brief description of the data, the data contains the total of 11 profiles (classes) of the gas chromatogram. Each profile contains different concentrations of gasoline and kerosene in the chemical compound: The concentration of gasoline adulteration in the compound ranges from 90% gasoline and

10% kerosene to 100% pure gasoline. Each chromatogram has 2,000-time points during two minute-probation until the end of the separation. The measurements on every chromatogram are collected at equally spaced time points. The data contains 10 replicates for each type of mixture of gasoline and kerosene. That is, $M = 11$ and $R = 10$.

As described in Section 4.2, the device using the FOX system creates five unique chromatograms from a single test injection by utilizing the flow splitter with five separate separation columns. There are two stationary phases of measuring the gas chromatogram, phase A and phase B that provide more information regarding the classification of the mixture adulterants. In our data analysis, we focus on the pattern of gas intensities obtained from the fifth channel of the device in Phase B, denoted by 5B because the trend of gas intensities from 5B has frequent fluctuations, which are more likely to interfere with the identification of the adulterants. Based on the gas chromatograms with 5B, we conduct the proposed BFAD method in order to estimate unknown functional trends of multi-class gasoline by considering that there is a common functional pattern among classes and unique patterns of each class. Based on the functional pattern of the class effects across the time points, we identify significant time intervals that provide useful information for efficient discrimination of gasoline type of the newly tested samples.

We consider pairs of gas chromatograms among the 11 levels of the gasoline, e.g. 90% vs. 91% of gasoline, and estimate their mean function and individual class effects by using the two-step Bayesian estimation as discussed in Section 4.5. Because there are the ten replicates for each class, we estimate the global fixed patterns of the intensities based on each replicate and take an average of them to estimate the global mean function. After obtaining the estimates of the global mean function and functional class effects for each class, we construct the 95% credible intervals for the class effects based on each class in order to find the time interval where the credible intervals do not overlap with each other. That is, the class effects are statistically distinguishable within the corresponding area.

We conduct the Bayesian focal-area detection for identifying the significant time intervals for the classification of adulterants. We consider a combination of two classes of gasoline among the 11 levels of gasoline in the Bayesian estimation with the piece-wise constant approximation. Figure 4.7 and 4.9 illustrate the estimated global mean function and the differences of the two class effects between 90% and 91% of gasoline and between 91% and 92% in the compound, respectively. For the result from the BFAD using 90% and 91% of gasoline, the method detects a time interval between 15.4 and 21.5 seconds as the significant interval for the classification of the two classes. The result suggests that the detected time interval would enable us to conduct the efficient classification without searching for the whole measurements within the retention time. However, the proposed method could not identify any important time points when the two levels of gasoline have a very similar pattern of intensities like 91% and 92% of gasoline, as shown in Figure 4.11. The 95% credible interval for the difference of the class effects in Figure 4.11b includes zero against the whole retention time because the slight differences shrink to zero by the shrinkage priors for the total variation of the class effects. It suggests that we might consider shrinkage priors in the hierarchical model other than Laplace prior or adjust hyperparameters in the priors for the tuning parameters in order to make a delicate detection of the subtle differences between the class effects.

After the significant areas are obtained from the BFAD method using the pairwise gas chromatograms, we validate whether the significant time intervals obtained from the BFAD can increase the efficiency of the classification of the mixture of gasoline and kerosene. We compare the performance of the classification based on the only significant time intervals with that based on the total measurement time. For the classification of two different types of adulterants, we implement partial least squares discriminant analysis (PLS-DA) as described in Section 4.6. Since we had multiple proportions of gasoline, we run the PLS-DA method to discriminate a combination of two classes (two different gasoline types) for a given 5B.

We examine the performance of the implementations on a personal laptop, which contains 8-core 2.60 GHz Intel i7-6700HQ processors and 8 GB of memory. We compute the prediction accuracy, computation time, and memory allocation for the comparison of the performance. The computation time and the amount of memory use are measured in seconds and megabytes (MB), respectively. In terms of prediction accuracy, we build the PLS-DA classifier of two classes using a training set and calculate the prediction accuracy using a 5-fold cross-validation procedure. Since each type of gasoline has ten replicates, we select two replicates including each one class as test sets and the remaining replicates as training sets. Then, the prediction accuracy is computed by the ratio between the number of correct predictions and the total cases. We see that the PLS-DA using the significant time interval takes 1.1 seconds and 273 MB of memory in the computation, while using the total time takes 2.4 seconds and 788 MB of memory on average. Hence, the BFAD method enables us to save the computation time and memory usage for the discrimination of the classes in the follow-up experimentation. Figure 4.10 shows two heatmaps of the prediction accuracy when the PLS-DA method uses the total time points and only significant time points for the classification. The results suggest that the prediction accuracy using the time interval detected by the BFAD method is comparable with the one using the entire time.

Therefore, our Bayesian focal-area detection method provides several benefits in identifying different classes of samples in terms of (1) feasible statistical inference based on a Bayesian credible interval, (2) reduction of computation time and memory requirements, and (3) competitive prediction accuracy with the one having the overall exploration of measurements within whole time interval.

4.8 Discussion

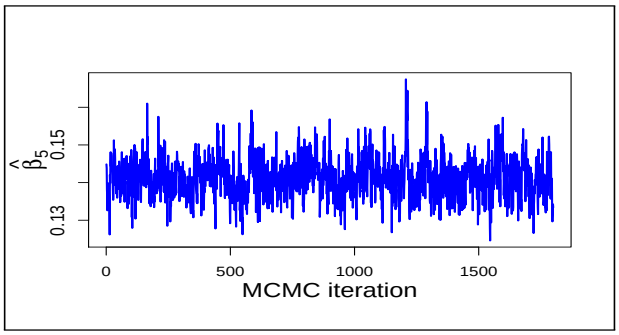
We have proposed a Bayesian focal-area detection method under a nonparametric sequential model for the multi-class dynamic model. Our approach could simultaneously estimate unknown functional trends of multilevel compounds in mixture and statistically detect specific focal areas for distinguishing the unique class effects in the mixture utilizing Bayesian framework. The proposed method is able to suppress noise from noise-mixed functional trends by using the fused lasso prior and group fused lasso prior which represents total variation of the increments of the intensities across the measurement time and the classes. In addition, our proposed method could facilitate comprehensive statistical inference and uncertainty quantification quantification by using the Bayesian hierarchical model. The method estimates all of the parameters including prior knowledge of the tuning parameters in MCMC algorithms so that it can avoid re-fitting the model for the fixed tuning parameters with the cross-validation or the grid-search of them for every bootstrapped sample. Besides, the Laplace priors of the parameters enable the implementation of the simple Gibbs sampler in the estimation with the closed form of the full conditionals. Based on the simulation study and the application, we see that the BFAD method could improve the performance of the classification in terms of computational efficiency. Therefore, the two-step Bayesian sequential method is able to provide important information for the efficient classification of newly tested samples from follow-up experimentation.

The proposed two-step Bayesian sequential algorithm is a useful strategy to reduce the numerical issues due to the matrix inversion of the penalty parameters in the Bayesian framework. By estimating the global mean function and the unique class effects in each step, the method could have the computational stability and produce a fast convergence rate with the Gibbs sampler. Additionally, the reformulation of the parameters could lower the computational cost of the Bayesian estimation by reducing dimensions of the covariance

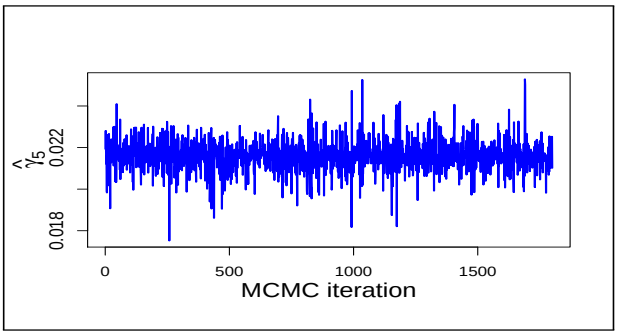
matrices, which changes the computation complexity of the inverse matrix from $\mathcal{O}((nM)^3)$ to $\mathcal{O}((n(M-1))^3)$. Thus, it is a useful technique, even when the amount of time points n is huge.

We note that the group fused lasso prior is applicable to the case that the class is an ordinal. If there is no evidence of the ordinality in the classes, we can consider a horseshoe prior for the class effects. Additionally, because the horseshoe prior has high adaptive properties in the functional estimation, it would expect to perform better than the Laplace prior for the functions including many break points and spikes. We will explore the performances of the BFAD method when using the horseshoe prior for our future research.

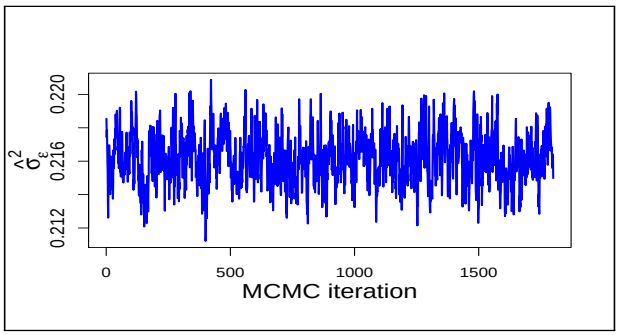
The results in this dissertation bring several interesting future works. For our proposed method, we only addressed the scenario when the time points of the measurements are equally spaced. However, the methods can be extended to a setting with unequally spaced time points. Bleakey and Vert [4] proposed the position-dependent weights for a case of non-uniform spacing. We would consider incorporating the proposed weight into our Bayesian focal-area detection method. In addition, we compared the performance of the BFAD method based on only the PLS-DA method in the simulation study. We will explore the applicability of our proposed method by using various machine learning classifiers, including k-nearest neighborhood (KNN), artificial neural network (ANN), random forest, etc. We expect that these future research will provide more guaranteed underpinnings of the proposed approach in perspective of the generality and practicality.



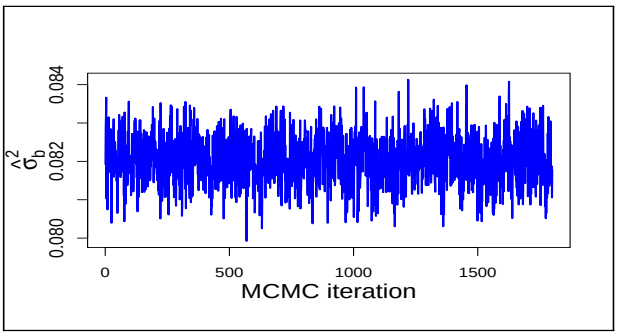
(a) MCMC for β_5



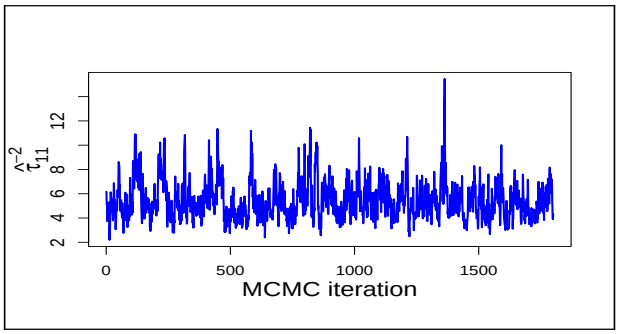
(b) MCMC for γ_5



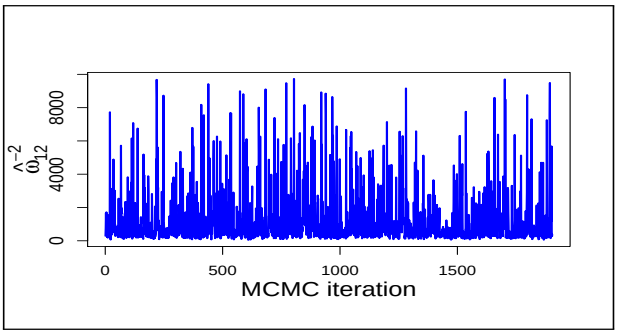
(c) MCMC for σ^2_ϵ



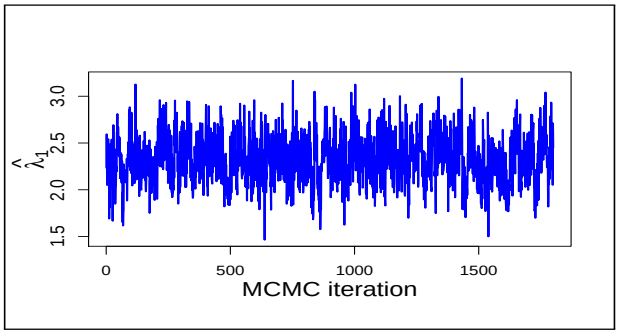
(d) MCMC for σ^2_b



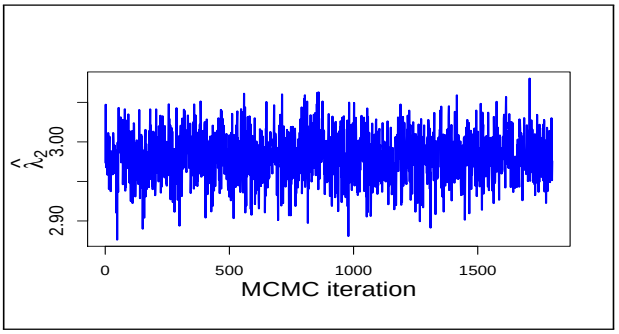
(e) MCMC for τ_{11}^{-2}



(f) MCMC for ω_{12}^{-2}

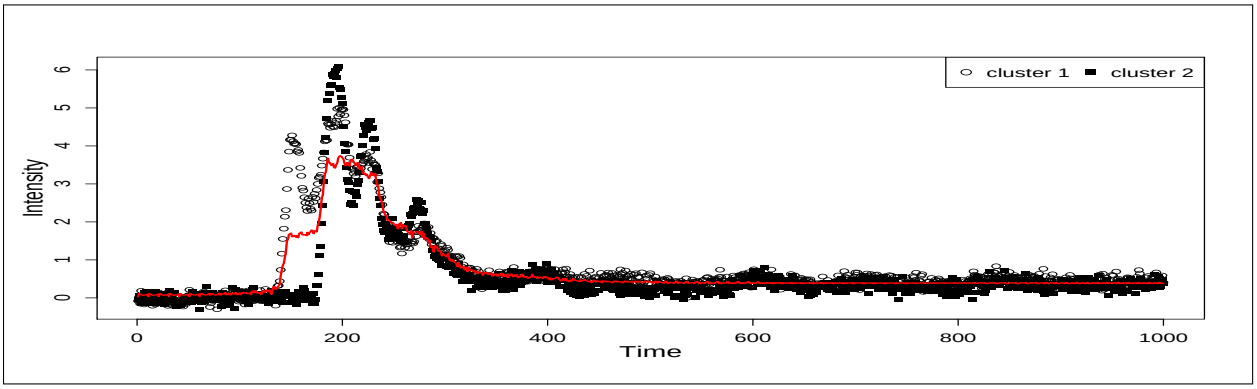


(g) MCMC for λ_1

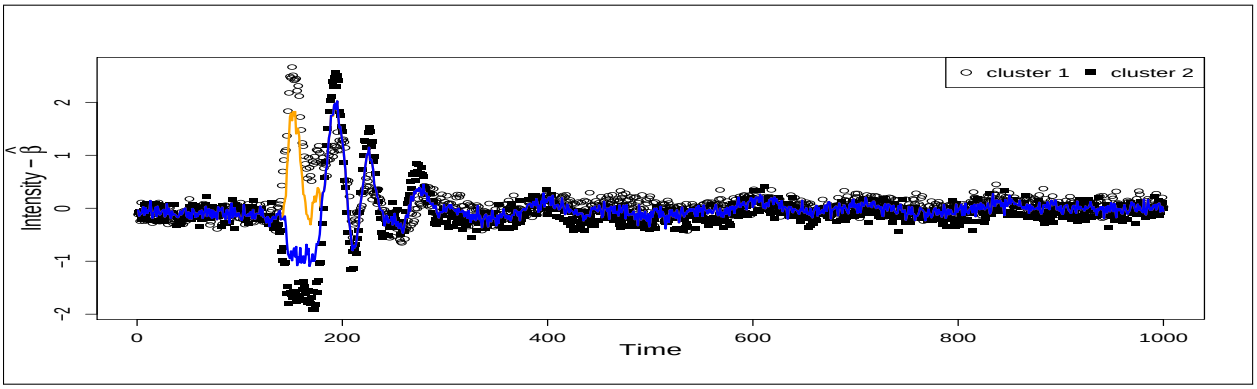


(h) MCMC for λ_2

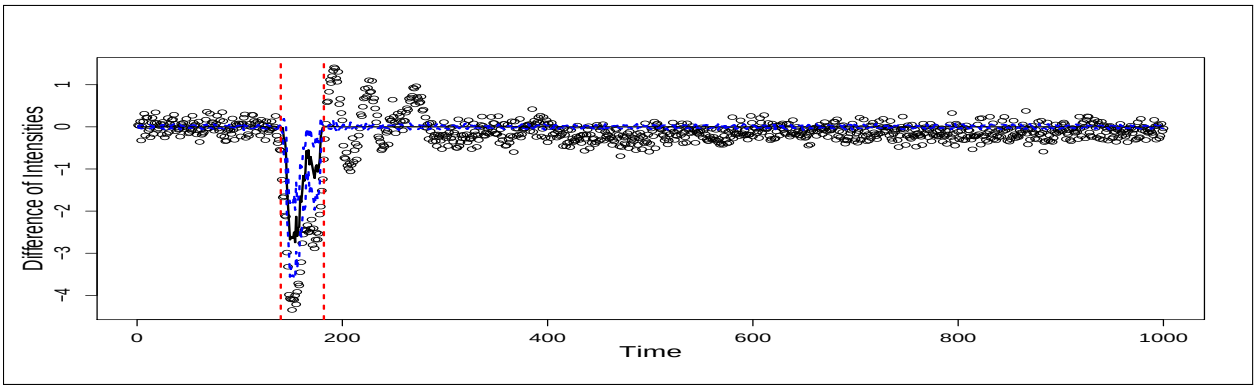
Figure 4.3: Trace plots of the parameters from the Gibbs sampling for 2,000 MCMC iterations



(a) The estimated global mean function between cluster 1 and 2

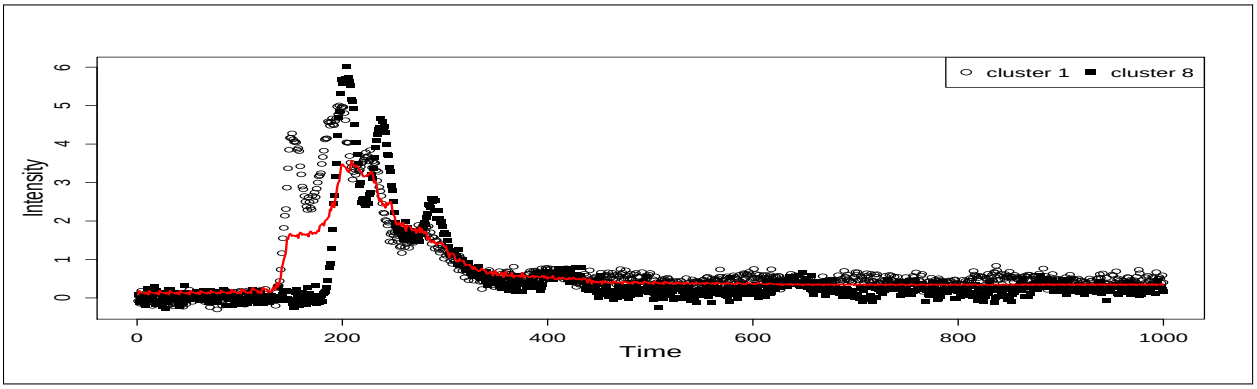


(b) The estimated functional random trends for each cluster 1 and 2

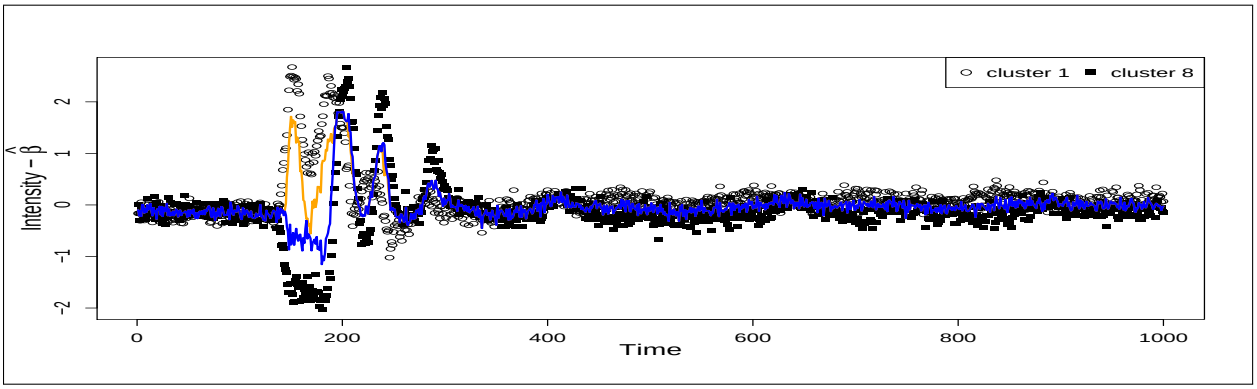


(c) The 95% credible interval for the difference between the effects of cluster 1 and 2

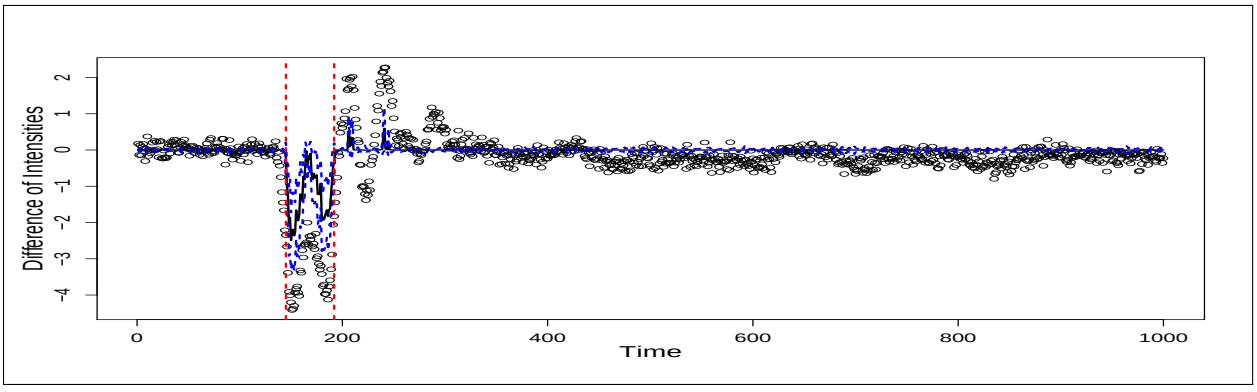
Figure 4.4: The estimated global mean function and functional class effects of cluster 1 and 2 using the BFAD method with 1,000 time points. The 95% credible interval for the difference between the effects of cluster 1 and cluster 2; The interval between 145 and 179 is determined as the significant time points based on the BFAD method.



(a) The estimated global mean function between cluster 1 and 8

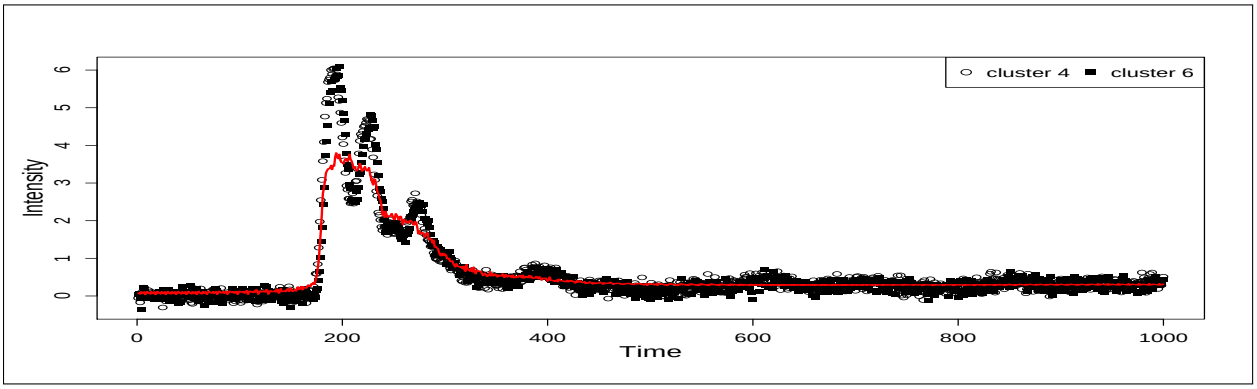


(b) The estimated functional random trends for each cluster 1 and 8

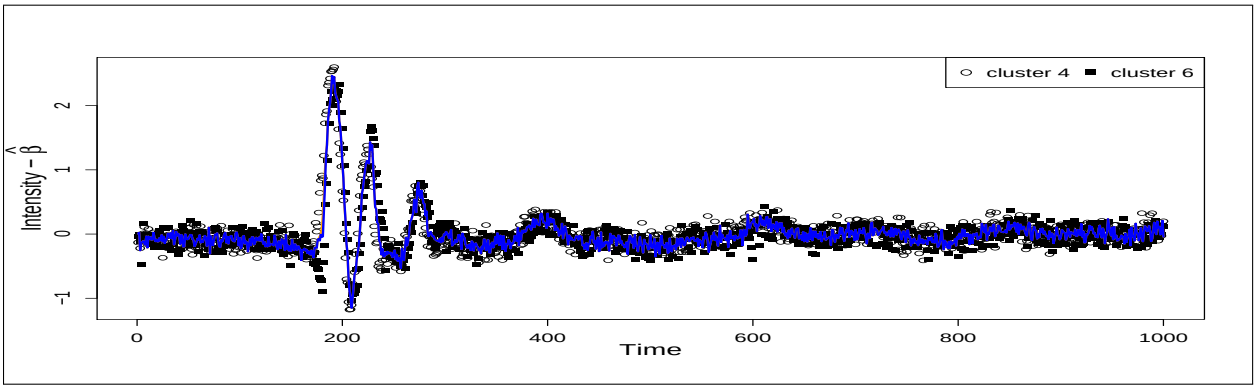


(c) The 95% credible interval for the difference between the effects of cluster 1 and 8

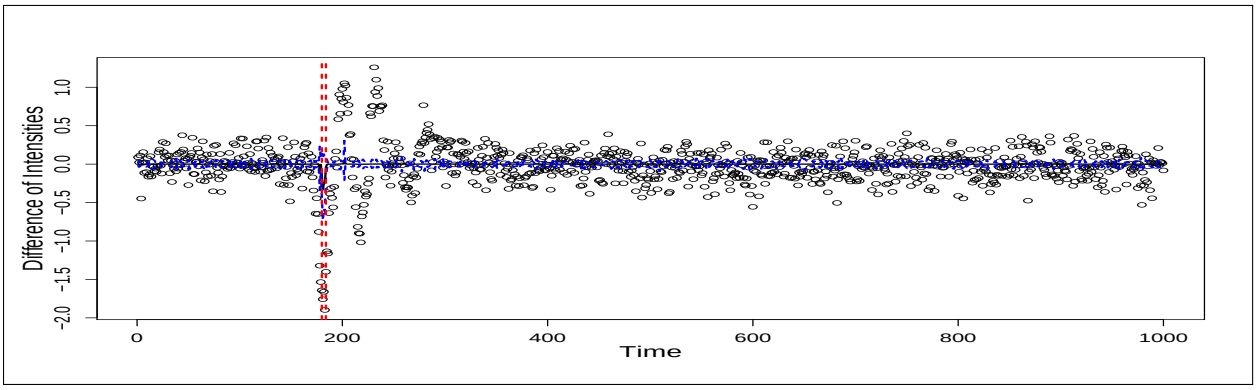
Figure 4.5: The estimated global mean function and functional class effects of cluster 1 and 8 using the BFAD method with 1,000 time points. The 95% credible interval for the difference between the effects of cluster 1 and cluster 8; The interval between 147 and 189 is determined as the significant time points based on the BFAD method.



(a) The estimated global mean function between cluster 4 and 6

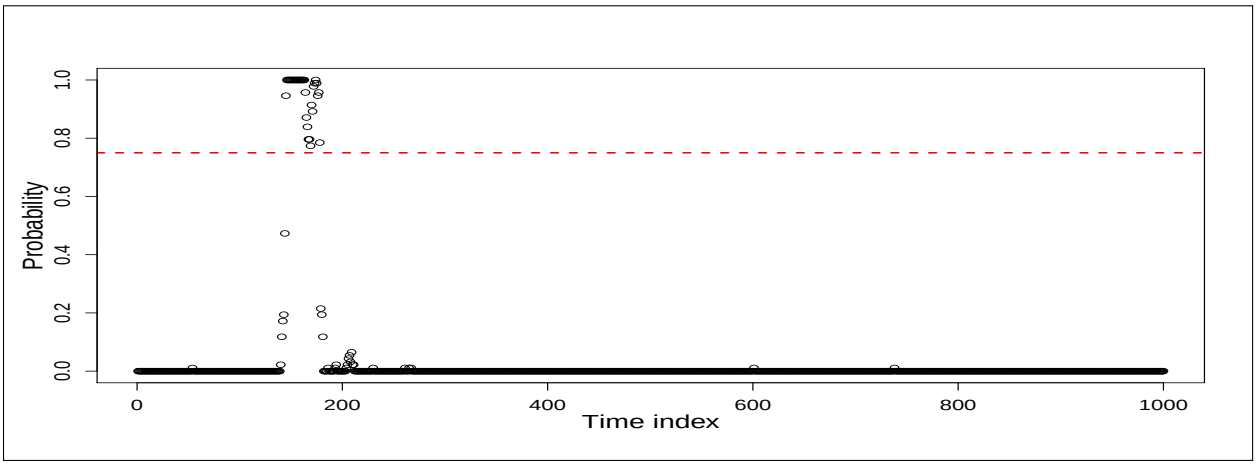


(b) The estimated functional random trends for each cluster 4 and 6

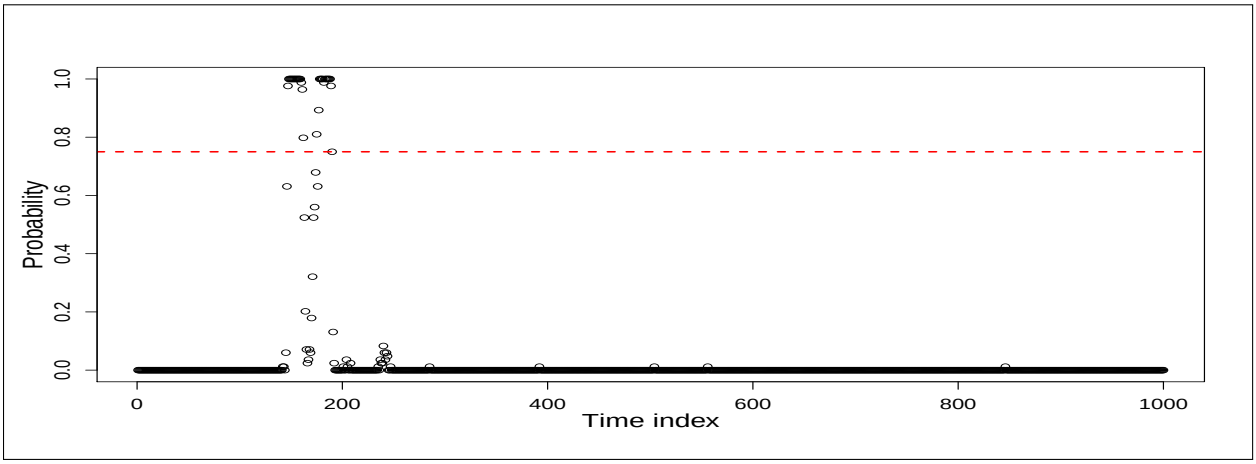


(c) The 95% credible interval for the difference between the effects of cluster 4 and 6

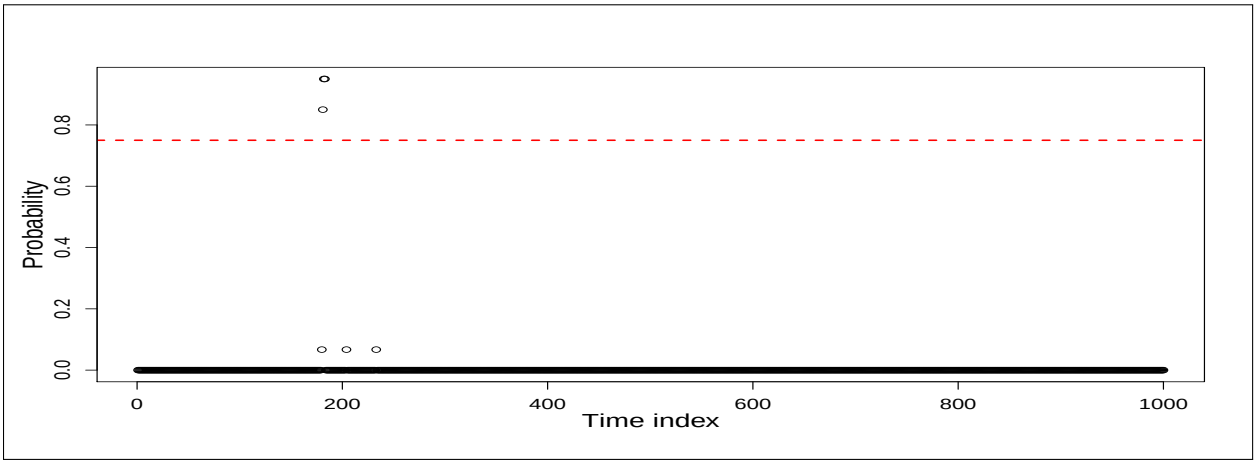
Figure 4.6: The estimated global mean function and functional class effects of cluster 4 and 6 using the BFAD method with 1,000 time points. The 95% credible interval for the difference between the effects of cluster 4 and cluster 6; The interval between 181 and 183 is determined as the significant time points based on the BFAD method.



(a) Probability of significance at each time points based on BFAD between cluster 1 and 2 with 200 runs

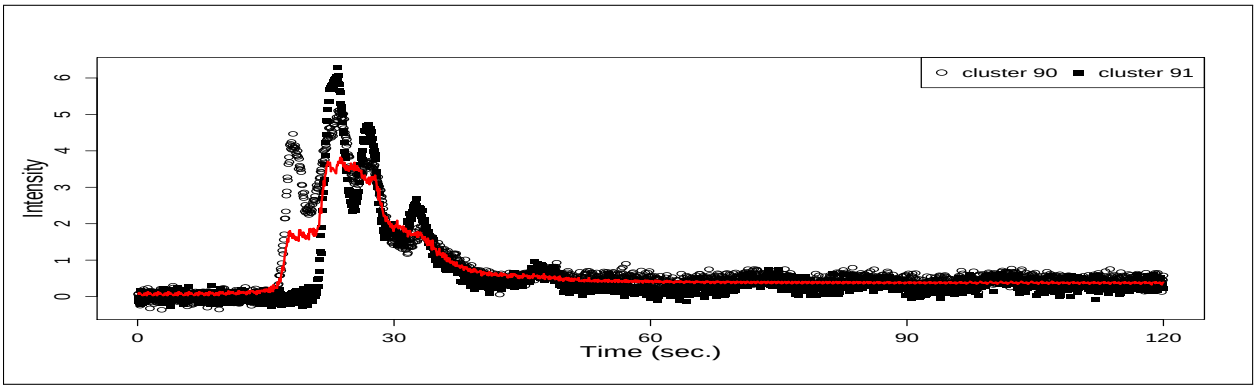


(b) Probability of significance at each time points based on BFAD between cluster 1 and 8 with 200 runs

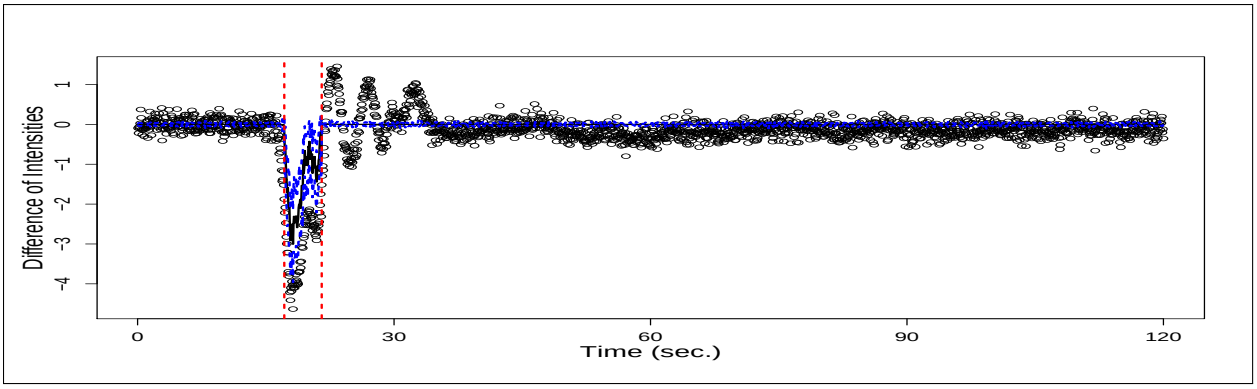


(c) Probability of significance at each time points based on BFAD between cluster 4 and 6 with 200 runs

Figure 4.7: Probability of significance at each time points based on BFAD with 200 runs

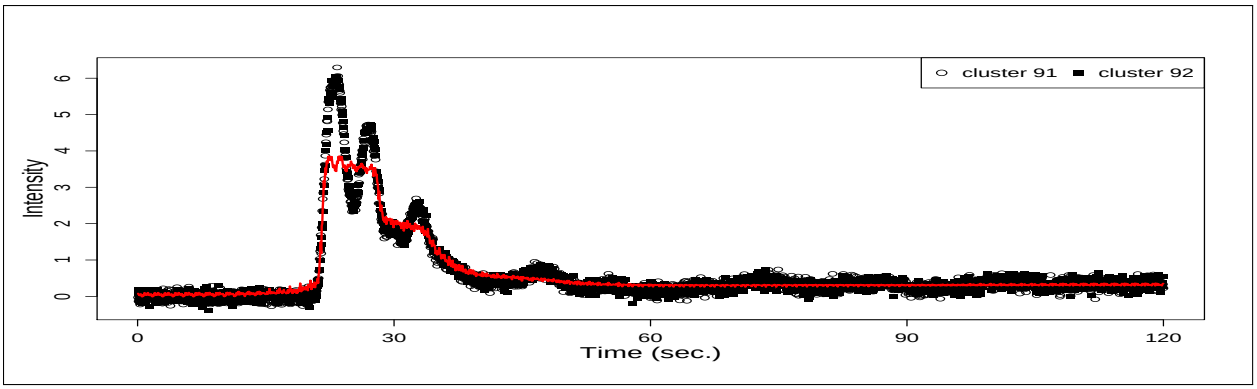


(a) The estimated global mean function between 90% and 91% of gasoline

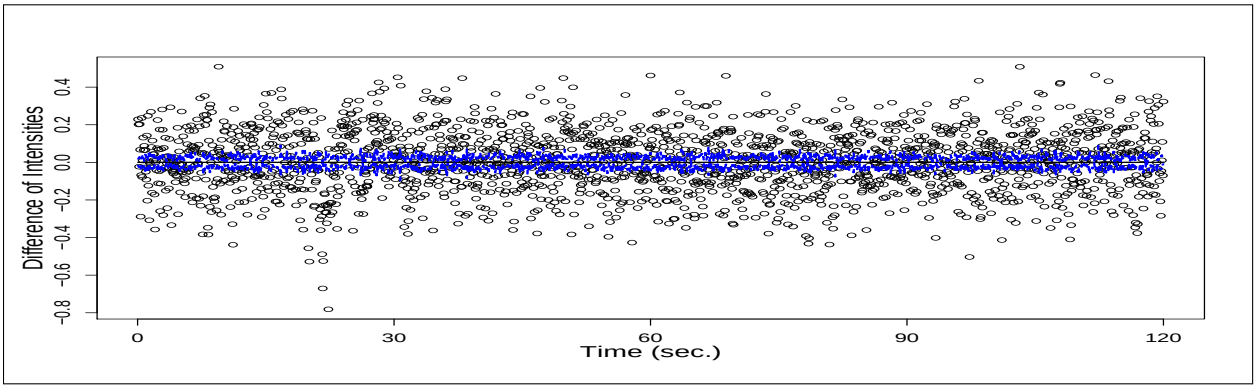


(b) The estimated functional random trends for each 90% and 91% of gasoline

Figure 4.9: The estimated global mean function and class effects of 90% and 91% of gasoline using the BFAD method with 2,000 time points within 2 minutes of the retention time. The 95% credible interval for the difference between the effects of 90% and 91% of gasoline; The time interval between 15.4 and 21.5 secs is significant for the classification based on the BFAD method.

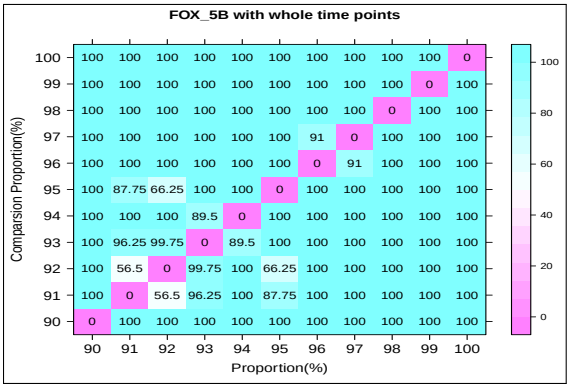


(a) The estimated global mean function between 91% and 92% of gasoline

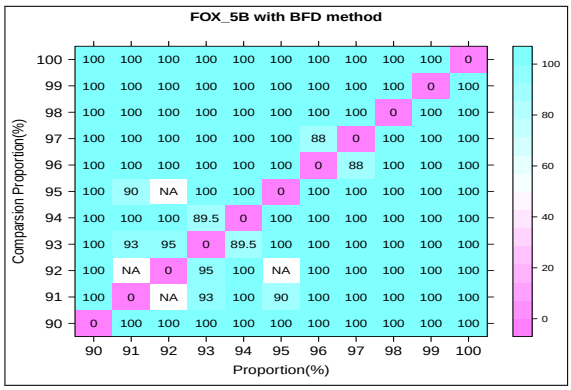


(b) The estimated functional random trends for each 91% and 92% of gasoline

Figure 4.10: The estimated global mean function and class effects of 91% and 92% of gasoline using the BFAD method with 2,000 time points within 2 minutes of the retention time. The 95% credible interval for the difference between the effects of 91% and 92% of gasoline; No significant time interval is detected based on the BFAD method.



(a) Heatmap of the prediction accuracy using the total time points



(b) Heatmap of the prediction accuracy using the significant time intervals based on the BFAD method

Figure 4.11: Heatmaps of the prediction accuracy from PLS-DA using the total 2,000 time points (left) and the significant time intervals (right) obtained from the BFAD method based on 5-fold cross-validation with a combination of the two levels of gasoline

Chapter 5

Conclusions

In this dissertation, we proposed semiparametric and nonparametric methods to deal with the following three types of data: (i) matched case-crossover data, (ii) high-correlated high-dimensional data, and (iii) multi-class dynamic trend data. Our contribution is to develop the following three statistical methodologies for complex data: (i) Flexible omnibus test for matched case-crossover design with measurement error in covariate, (ii) Joint semiparametric kernel machine network regression for high-correlated and high-dimensional data, and (iii) Bayesian functional estimation and focal-area detection for multilevel dynamic trend data.

In Chapter 2, we showed that our flexible omnibus test is able to (i) identify the significance of the functional relationship between the incidence of meningitis and exposure to water turbidity, (ii) test whether the main effect is time-varying, and (iii) assess any existence of interaction effects between water turbidity and air temperature. We developed a flexible omnibus test using a semiparametric framework. We proposed an efficient score from estimating equations on the null model so that the proposed test does not require a specific alternative model. Based on the simulation studies, we figured out that the proposed method can maintain statistical power, even when the distribution of the latent variable is misspecified via simulation studies. The proposed flexible omnibus test enables us to make

flexible inferences on various hypothesis settings under the weak distributional assumptions on the latent variable and the measurement error.

In Chapter 3, we proposed a joint semiparametric kernel machine network regression for high-correlated and high-dimensional data. We developed a unified and integrated method which can simultaneously identify important variables and build a genetic interaction network between them. Based on the simulation studies, we showed that our proposed method allows for modeling nonlinear and non-additive functional effects and complicated interactions. We demonstrated the advantages of The proposed method with the application to the type II diabetes gene pathway data. The proposed method could detect important genes and estimate their network connected with the continuous glucose level. In conclusion, the proposed method has the flexibility for any semiparametric model including non-additive and nonparametric model and provides an interpretable network with the response variable. This work facilitates a comprehensive interpretation of the relationship between the important genes with the dependence structure and the clinical outcome.

In Chapter 4, we proposed Bayesian focal-area detection method for multi-class dynamic model. We showed that our approach could estimate unknown functional trends of multilevel compounds in mixture and statistically detect specific focal areas for the efficient identification of the unknown types of the compounds. Under the Bayesian framework, the proposed method could facilitate comprehensive statistical inference and uncertainty. In addition, the method could elucidate chromatographic trends for each class in the mixture by adapting the two shrinkage priors, fused lasso and group fused lasso. We expect that the proposed method enables applied researchers to focus on a subset of time-domain for efficient identification and reduce the burden of the computation.

Following the research for this dissertation, we have several possible extensions of our methods for future works. First, our flexible omnibus test could be extended to be applicable to testing anomalies in temporal patterns of data in real-time data streaming by

incorporating kernel functions and Bayesian focal-area detection. Second, we could apply our joint semiparametric kernel machine network approach generalized linear models (GLM) to handle the multiple categorical or binary data. The extensions above will provide guidance for deep understanding and intelligent application of knowledge in a wide range of applications, including epidemiology, environmental health, and chemical engineering, with various and complex data.

Bibliography

[1] A. F. Abdussalam, A. J. Monaghan, V. M. Dukic, M. H. Hayden, T. M. Hobson, G. C. Leckebusch, and J. E. Thornes. Climate influences on meningitis incidence in northwest nigeria. *American Meteorological Society*, 6:62–76, 2014.

[2] M. Akbar, M. Restaino, and M. Agah. Chip-scale gas chromatography: From injection through detection. *Microsystems & Nanoengineering*, 1(15039), 2015.

[3] S. M. Berry, J. R. Carroll, and D. Ruppert. Bayesian smoothing and regression splines for measurement error problems. *Journal of the American Statistical Association*, 97: 160–169, 2002.

[4] K. Bleakey and J. P. Vert. The group fused lasso for multiple change-point detection, 2010. arXiv:1106.4199.

[5] K. S. Booksh and B. R. Kowalski. Theory of analytical chemistry. *Analytical Chemistry*, 66(15):782–791, 1994.

[6] R. G. Brereton and G. R. Lloyd. Partial least squares discriminant analysis: taking the magic away. *Journal of Chemometrics*, 28(4):213–225, 2014.

[7] P. R. Burton, L. J. Palmer, K. Jacobs, K. J. Keen, J. M. Olson, and R. C. Elston. Ascertainment adjustment: Where does it take us? *American Journal of Human Genetics*, 76:1505–1514, 2000.

[8] T. Cai, W. Lie, and X. Luo. A constrained l_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106:594–607, 2011.

[9] L. Cheng, L. Shan, and I. Kim. Multilevel gaussian graphical model for multilevel networks. *Journal of Statistical Planning & Inference*, 190:1–14, 2017.

[10] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge University Press, Cambridge, 2000.

[11] P. Danaher, P. Wang, and D. M. Witten. The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society, Series B*, 76:373–397, 2014.

[12] P. Diggle, K. Y. Liang, and S. L. Zeger. *Analysis of Longitudinal Data*. Oxford University Press, New York, 1994.

[13] M. Drton and R. Foygel. Extended bayesian information criteria for gaussian graphical models. *Advances in Neural Information Processing Systems*, 23:2020–2028, 2010.

[14] Z. Fang, I. Kim, and Schaumont P. Flexible variable selection for recovering sparsity in nonadditive nonparametric models. *Biometrics*, 72:1155–1163, 2016.

[15] M. Ferreiro-González, J. Ayuso, J. A. Álvarez, M. G. Palma, and C. Barroso. New headspacemass spectrometry method for the discrimination of commercial gasoline samples with different research octane numbers. *Energy Fuels*, 28(10):6249–6254, 2014.

[16] D. L. Flumignan, N. Boralle, and J. E. De Oliveira. Screening brazilian commercial gasoline quality by hydrogen nuclear magnetic resonance spectroscopic fingerprintings and pattern-recognition multivariate chemometric analysis. *Talanta*, 82(1):99–105, 2010.

[17] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9:432–441, 2008.

[18] R. L. Grob and E. F. Barry. *Modern practice of gas chromatography*. John Wiley & Sons, New York, 2004.

[19] J. Guo, E. Levina, G. Michailidis, and J. Zhu. Joint estimation of multiple graphical models. *Biometrika*, 98:1–15, 2011.

[20] J. D. Hart. Frequentist-bayes lack-of-fit tests based on laplace approximations. *Journal of Statistcal Theory and Practice*, 3:681–704, 2009.

[21] D. W. Hosmer and S. Lemeshow. *Applied Logistic Regression*. Wiley, New York, 1989.

[22] H. Kim, H. K. Cheong, S. K. Park, and G. R. Bae. Drinking water trubidity and aseptic meningitis in children in an urban community in korea. *Epdemiology*, 14:S110, 2003.

[23] I. Kim, R. J. Carroll, and N. Cohen. Effect heterogeneity by a matching covariate in matched case-control studies: a method for graphs-based representation. *American Journal of Epidemiology*, 156:463–470, 2002.

[24] I. Kim, N. Cohen, and R. J. Carroll. Semiparametric regression splines in matched case-control studies. *Biometrics*, 59:1158–1169, 2003.

[25] I. Kim, N. Cohen, and R. J. Carroll. Semiparametric and nonparametric modeling for matched studies. *Computational Statistics & Data Analysis*, 46:631–643, 2004.

[26] I. Kim, H.K. Chenong, and H. Kim. Semiparametric regression models for detecting effect modification in matched case-crossover studies. *Statistics in Medicine*, 30:1837–1851, 2009.

[27] D. M. Kristina, P. G. Charles, N.H. Charles, and B. R. Joan. Risk assessment of waterborne coxsackievirus. *American Water Works Association*, 95:122–131, 2003.

[28] R. Kumar. Aseptic meningitis: Diagnosis and management. *Indian Journal of Pediatrics*, 72:57–63, 2005.

[29] M. Kyung, J. Gill, M. Ghosh, and G. Casella. Penalized regression, standard errors, and bayesian lassos. *Bayesian Analysis*, 5(2):369–412, 2010.

[30] J. Li, Z. Wang, R. Li, and R. Wu. Bayesian group lasso for nonparametric varying-coefficient models with application to functional genome-wide association studies. *Annals of Applied Statistics*, 9(2):640–664, 2015.

[31] Y. Lin and H. H. Zhang. Component selection and smoothing in multivariate nonparametric regression. *The Annals of Statistics*, 34:2272–2297, 2006.

[32] D. Liu, X. Lin, and D. Ghosh. Semiparametric regression of multidimensional genetic pathway data: least-squares kernel machines and linear mixed models. *Biometrics*, 63: 1079–1088, 2007.

[33] D. Liu, X. Lin, and D. Ghosh. Estimation and testing for the effect of a genetic pathway on a disease outcome using logistic kernel machine regressino via logistic mixed models. *BMC Bioinformatics*, 9:1–11, 2007.

[34] H. Liu and L. Wang. Tiger: A tuning-insensitive approach for optimally estimating gaussian graphical models. *Electronic Journal of Statistics*, 11:241–294, 2017.

[35] Y. Ma, J. D. Hart, R. Janicki, and R. J. Carroll. Local and omnibus goodness-of-fit tests in classical measurement error models. *Journal of the Royal Statistical Society, Series B*, 73:81–98, 2011.

[36] A. Maity and X. Lin. Powerful tests for detecting a gene effect in the presence of possible gene-gene interactions using garrote kernel machines. *Biometrics*, 67:1271–1284, 2011.

[37] J. McMurry. *Organic chemistry: with biological applications*. Cengage Learning, Belmont, 2011.

[38] H. M. McNair and E. J. Bonelli. *Basic gas chromatography*. John Wiley & Sons, New Jersey, 2011.

[39] V. K. Mootha, C. M. Lindgren, K. F. Eriksson, Subramanian A., S. Sihag, J. Lehar, P. Puigserver, E. Carlsson, Ridderstråle M., E. Laurila, N. Houstis, M. J. Daly, N. Patterson, J.P. Mesirov, T.R. Golub, P. Tamayo, B. Spiegelman, E. S. Lander, J. N. Hirschhorn, D. Altshuler, and L. C. Groop. Pgc-1alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature Genetics*, 34:267–273, 2003.

[40] M. G. Nespeca, J. F. V. L. Munhoz, D. L. Flumignan, and J. E. de Oliveira. Rapid and sensitive method for detecting adulterants in gasoline using ultra-fast gas chromatography and partial least square discriminant analysis. *Fuel*, 215:204–211, 2018.

[41] J. M. Neuhausel, W. Hauck, and J. D. Kalbfleisch. The effects of mixture distribution misspecification when fitting mixed effects logistic models. *Biometrika*, 79:755–762, 1992.

[42] A. Ortega-Villa, I. Kim, and H. Kim. Semiparametric time varying model for matched case-crossover studies. *Statistics in Medicine*, 36:998–1031, 2017.

[43] T. Park and G. Casella. The bayesian lasso. *Journal of the American Statistical Association*, 103:681–686, 2008.

[44] R. Pfeiffer and Gail M. Inference for covariates that accounts for ascertainment and random genetic effects in family studies. *Biometrika*, 88:933–948, 2001.

[45] P. Radchenko and G. M. James. Variable selection using adaptive nonlinear interaction structures in high dimensions. *Journal of the American Statistical Association*, 105: 1541–1553, 2010.

[46] P. Ravikumar, J. Lafferty, H. Liu, and L. Wasserman. Sparse additive models. *Journal of the Royal Statistical Society, Series B*, 71:1009–1030, 2009.

[47] P. Ravikumar, M. J. Wainwright, G. Raskutti, and B. Yu. High-dimensional covariance estimation by minimizing l1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980, 2011.

[48] B. P. Regmi, R. Chan, and M. Agah. Ionic liquid functionalization of semi-packed columns for high-performance gas chromatographic separations. *Journal of Chromatography A*, 1510:66–72, 2017.

[49] F. Rock, N. Barsan, and U. Weimar. Electronic nose: Current status and futre trend. *Chemical Reviews*, 108(2):705–725, 2008.

[50] E. A. Roualdes. Bayesian trend filtering. 2015. arXiv:1505.07710.

[51] D. Ruppert, M. P. Wand, and R. J. Carroll. *Semiparametric Regression*. Cambridge University Press, New York, 2003.

[52] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.

[53] T. Wang, K. Birsoy, N. W. Hughes, K. M. Krupczak, Y. Post, J. J. Wei, E. S. Lander, and D. M. Sabatini. Identification and characterization of essential genes in the human genome. *Science*, 350:1096–1101, 2015.

[54] M. Yuan and Y. Lin. Model selection and estimation in the gaussian graphical model. *Biometrika*, 94:19–35, 2007.

[55] L. Zhang and I. Kim. Semiparametric bayesian kernel survival model for evaluating pathway effects. *Biometrika*, page https://doi.org/10.1177/0962280218797360, 2018.

Appendices

Appendix A

Appendices for Chapter 2

A.1 The Relationship between Prospective and Retrospective Models

In this section, we show the likelihood from prospective model is approximately equivalent to that from retrospective model. Let $\mathbf{\Theta}$ be the collection of all parameters. Using the prospective model, the probability of y_{kj} being one is

$$\begin{aligned}
& Pr(y_{k1}=0,\ldots,y_{kj}=1,\ldots,y_{k5}=0|\mathbf{\Theta}) \\
= & \int\int Pr(y_{k1}=1|x_{k1},w)\,Pr(y_{k2}=0|x_{k2},w)\,Pr(y_{k5}=0|x_{k5},w)\,p(\mathbf{x}_k)p(\mathbf{w}_k-\mathbf{x}_k) \quad d\mathbf{x}_k d\mathbf{u}_k \\
= & \int\int \frac{\exp\{\beta x_{k1}+q_k(\cdot)\}}{1+\exp\{\beta x_{k1}+q_k(\cdot)\}}\cdots\frac{1}{1+\exp\{\beta x_{k5}+q_k(\cdot)\}}\,p(\mathbf{x}_k)p(\mathbf{w}_k-\mathbf{x}_k) \quad d\mathbf{x}_k d\mathbf{u}_k \\
\equiv & \int\int g_j \quad d\mathbf{x}_k d\mathbf{u}_k,
\end{aligned}$$

where

$$g_j = \frac{\exp\{\beta x_{k1}+q_k(\cdot)\}}{1+\exp\{\beta x_{k1}+q_k(\cdot)\}}\cdots\frac{1}{1+\exp\{\beta x_{k5}+q_k(\cdot)\}}\,p(\mathbf{x}_k)p(\mathbf{w}_k-\mathbf{x}_k).$$

Then the conditional probability can be

$$
\begin{aligned}
& Pr(y_{k1}=1, y_{k2}=0, \ldots, y_{k5}=0 | \sum_{j=1}^{5} y_{kj}=1, \boldsymbol{\Theta}) \\
= & \frac{Pr(y_{k1}=1, y_{k2}=0, \ldots, y_{k5}=0|\boldsymbol{\Theta})}{Pr(y_{k1}=1, y_{k2}=0, \ldots, y_{k5}=0|\boldsymbol{\Theta}) + Pr(y_{k1}=0, y_{k2}=1, \ldots, y_{k5}=0|\boldsymbol{\Theta}) + \cdots + Pr(y_{k1}=0, y_{k2}=0, y_{k5}=1|\boldsymbol{\Theta})}, \\
= & \frac{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k}{\sum_{j=1}^{5} \int\int g_j d\mathbf{x}_k d\mathbf{u}_k}, \\
= & \frac{1}{1+\frac{\sum_{j=2}^{5} \int\int g_j d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k}}, \\
\approx & \frac{1}{1+\frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k}},
\end{aligned}
$$

where $f = \sum_{j=2}^{5} g_j$. Using Taylor expansion, we can have

$$
\frac{1}{1+\frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g d\mathbf{x}_k d\mathbf{u}_k}} \cong 1 - \frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g d\mathbf{x}_k d\mathbf{u}_k}
$$

and also have

$$
\int\int \frac{g_1}{g_1+f} d\mathbf{x}_k d\mathbf{u}_k = \int\int \frac{1}{1+\frac{f}{g_1}} d\mathbf{x}_k d\mathbf{u}_k \cong 1 - \int\int \frac{f}{g_1} d\mathbf{x}_k d\mathbf{u}_k.
$$

Under the following approximation,

$$
\int\int \frac{f}{g_1} d\mathbf{x}_k d\mathbf{u}_k \cong \frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k}, \tag{A.1}
$$

we can show that

$$
\begin{aligned}
\frac{1}{1+\frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k}} & \approx \int\int \frac{g_1}{g_1+f} d\mathbf{x}_k d\mathbf{u}_k \\
& = \int\int \frac{1}{1+\sum_{j=2}^{5} \exp\{-\beta(x_{k1}-x_{kj})\}} p(\mathbf{x}_k) p(\mathbf{w}_k - \mathbf{x}_k) \quad d\mathbf{x}_k d\mathbf{u}_k.
\end{aligned}
$$

Therefore, the conditional probability obtained from the prospective model can be approxi-

mately equivalent to

$$\begin{aligned} & Pr(y_{k1}=1, y_{k2}=0, \ldots, y_{k5}=0 | \sum_{j=1}^{5} y_{kj}=1, \boldsymbol{\Theta}) \\ \approx & \int\int \frac{1}{1+\sum_{j=2}^{5}\exp\{-\beta(x_{k1}-x_{kj})\}} p(\mathbf{x}_k)p(\mathbf{w}_k-\mathbf{x}_k) \quad d\mathbf{x}_k d\mathbf{u}_k, \end{aligned}$$

which is the likelihood from the retrospective model.

This approximation (A.1) can be shown under the condition, $g_j \approx g_{j'}$, $f \approx Mg_2$, $\int\int \sqrt{f}^2 d\mathbf{x}_k d\mathbf{u}_k \gtrsim C_0 \int\int f d\mathbf{x}_k d\mathbf{u}_k$ for some constant $C_0 > 0$, and be justified using an idea of important sampling. These conditions are satisfied in 1-M matched case-control studies.

We first show the same order of left and right side of the equation (A.1). The following sided inequality

$$\frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k} \leq \int\int \frac{f}{g_1} d\mathbf{x}_k d\mathbf{u}_k$$

is true in general under a condition $\int\int \sqrt{f}^2 \gtrsim C_0 \int\int f$ for some constant $C_0 > 0$.

The other sided inequality

$$\frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k} \geq \int\int \frac{f}{g_1} d\mathbf{x}_k d\mathbf{u}_k$$

is also true in special case, $g_j \approx g_{j'}$ and $f \approx Mg_2$, which is true in matched studies. This argument can be shown under the condition,

$$g_j \approx g_{j'}, \ j \neq j'.$$

This means that $g_1 \approx \sum_{j=2}^{M} g_j$ and $f \approx M g_2$. Hence we have

$$g_1 \approx g_2 \quad \text{and} \quad \int\int g_1 d\mathbf{x}_k d\mathbf{u}_k \approx \int\int g_2 d\mathbf{x}_k d\mathbf{u}_k.$$

This means that

$$\frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{M\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k} \approx \frac{\int\int g_2 d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k} \approx \frac{1}{M}\int\int \frac{(M)g_2}{g_1} d\mathbf{x}_k d\mathbf{u}_k \approx \frac{1}{M}\int\int \frac{f}{g_1} d\mathbf{x}_k d\mathbf{u}_k.$$

Therefore, we can show that

$$\frac{\int\int f d\mathbf{x}_k d\mathbf{u}_k}{\int\int g_1 d\mathbf{x}_k d\mathbf{u}_k} \approx \int\int \frac{f}{g_1} d\mathbf{x}_k d\mathbf{u}_k,$$

even if $M \to \infty$.

Next, we will show equality approximately. Using important sampling density $\tilde{g}$, where $\tilde{g}_1 = g_1 / \int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k$, we can show that

$$\begin{aligned}
\frac{\int\int f d\boldsymbol{x}_k d\boldsymbol{u}_k}{\int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k} &= \int\int \frac{f}{g_1} \frac{g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k}{\int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k} \\
&= \int\int \frac{f}{g_1} \tilde{g}_1 d\boldsymbol{x}_k d\boldsymbol{u}_k \\
&= \frac{1}{M_{s,s'}} \sum_s \sum_{s'} \frac{f(x_{k,s}, u_{k,s'})}{g_1(x_{k,s}, u_{k,s'})},
\end{aligned}$$

where $(x_{k,s}, u_{k,s'}) \sim \tilde{g}_1(\cdot,\cdot)$ and $M_{s,s'} = \sum_s \sum_{s'} 1$. On the other hand, using these samplings from $\tilde{g}$, we also have

$$\begin{aligned}
\int\int \frac{f}{g_1} d\boldsymbol{x}_k d\boldsymbol{u}_k &= \left(\int\int \frac{f}{g_1^2} \frac{g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k}{\int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k} \right) \int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k \\
&= \left\{ \frac{1}{M_{s,s'}} \sum_s \sum_{s'} \frac{f(x_{k,s}, u_{k,s'})}{g_1^2(x_{k,s}, u_{k,s'})} \right\} \int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k \\
&= \frac{1}{M_{s,s'}} \sum_s \sum_{s'} \left[\frac{f(x_{k,s}, u_{k,s'})}{g_1(x_{k,s}, u_{k,s'})} \left\{ \frac{1}{g_1(x_{k,s}, u_{k,s'})} \int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k \right\} \right]
\end{aligned}$$

$$= \frac{1}{M_{s,s'}} \sum_{s,s'} \frac{f(x_{k,s}, u_{k,s'})}{g_1(x_{k,s}, u_{k,s'})}$$

under the condition, $(x_{k,s}, u_{k,s'}) \sim (\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k)$, where $(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k) = \int\int (\boldsymbol{x}_k, \boldsymbol{u}_k) d\boldsymbol{x}_k d\boldsymbol{u}_k$, because we have

$$\int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k \cong g_1(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k)$$

and

$$\frac{1}{g_1(x_{k,s}, u_{k,s'})} \int\int g_1 d\boldsymbol{x}_k d\boldsymbol{u}_k \cong \frac{1}{g_1(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k)} g_1(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k).$$

Therefore, the approximation (A.1) is satisfied.

A.2 Kernel Density Estimation for Measurement Error Distribution

We employ a surrogate for $\boldsymbol{X}$, denoted by $\boldsymbol{W}$. If we can obtain multiple observations such that $\boldsymbol{W}_{jk} = (W_{jk1}, ..., W_{jkR})$ for the jth covariate in the kth stratum, where R is the number of the replicates, we can avoid the parametric assumption on the density of the measurement error. On the basis of this idea, if we let U be the measurement error, the replicate of surrogate, $\mathbf{W}_{jk}$ can be expressed as

$$W_{jkr} = X_{jk} + U_{jkr},$$

where $r = 1, ..., R$ and $\boldsymbol{U}$ is independent of $(\boldsymbol{X}, \boldsymbol{Y})$.

Using the multiple measurements, we can obtain pseudo-covariate and pseudo-measurement

error as follows,

$$\begin{aligned}\hat{X}_{jk} &= \sum_{r=1}^{r_m} \frac{W_{jkr}}{r_m} + \sum_{r=r_m+1}^{R} \frac{W_{jkr}}{2R-2r_m};\\ \hat{U}_{jkr} &= \Upsilon_{jkr} = W_{jkr} - \overline{W}_{jk\cdot},\end{aligned}$$

where $\overline{W}_{jk\cdot} = (1/R)\sum_{r=1}^{R} W_{jkr}$ and $r_m = \lfloor \frac{R}{2} \rfloor$.

We can then estimate the density of the measurement error, $p_U(\bullet)$, nonparametrically by using $\boldsymbol{\Upsilon} = (\Upsilon_{jk1}, ..., \Upsilon_{jkR})$. Assume that $\boldsymbol{\Upsilon}$ has the same distribution of $\boldsymbol{U}$ which is unknown and symmetric probability density function, $p_U(\bullet)$. By adapting the idea of Hall and Ma (2007), we can nonparametrically estimate the density of the measurement error as follows,

$$\hat{p}_U(u) = \frac{1}{N}\sum_{k=1}^{K}\sum_{j=1}^{M+1}\sum_{r=1}^{R} K_r(\Upsilon_{jkr} - u),$$

where $K_r(\cdot)$ is a kernel function.

A.3 Marginal Likelihood Calculation

Because the true covariate X is the latent variable due to the measurement error, the complete data log-likelihood in the kth stratum can be expressed as

$$\begin{aligned}L_k(\boldsymbol{\beta}, \gamma = 0 | \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k) &= \int p_{Y_k|Z_k,X_k}(\boldsymbol{y}_k|\boldsymbol{z}_k, \boldsymbol{x}_k, \boldsymbol{\beta}) p_{X_k|Z_k}(\boldsymbol{x}_k|\boldsymbol{z}_k) p_U(\boldsymbol{w}_k - \boldsymbol{x}_k)\ d\boldsymbol{x}_k\\ &= \int l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_k, \boldsymbol{y}_k) p_{X_k|Z_k}(\boldsymbol{x}_k|\boldsymbol{z}_k) p_U(\boldsymbol{w}_k - \boldsymbol{x}_k)\ d\boldsymbol{x}_k,\end{aligned}$$

where $p_{Y|Z,X}(\boldsymbol{Y}|\boldsymbol{Z}, \boldsymbol{X})$ is the data-generating model, $p_{X|Z}(\boldsymbol{X}|\boldsymbol{Z})$ is the model of the latent variable, $p_U(\boldsymbol{W} - \boldsymbol{X})$ is density of the measurement error transformed from the surrogate density $p_{W|Z,X}(\boldsymbol{W}|\boldsymbol{Z}, \boldsymbol{X})$, and $l_k(\boldsymbol{\beta}, \gamma = 0|\boldsymbol{z}_k, \boldsymbol{x}_k, \boldsymbol{y}_k)$ is the retrospective conditional logistic

model with the clustered binary outcome in $\boldsymbol{y}_k$ and case-control surrogate $\boldsymbol{w}_k$ in the kth stratum.

Using Metropolis-Hastings algorithm, the candidate observation of $\boldsymbol{X}$ can be generated by using Markov Chain Monte Carlo (MCMC) method [3]. If we consider the prior $p_{X|Z}(\bullet)$ as a normal distribution with mean μ_x and standard deviation σ_x^2, we can obtain a marginal likelihood

$$\begin{aligned} L_k(\boldsymbol{\beta}, 0|\boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k) &= \int p_{Y_k|Z_k,X_k}(\boldsymbol{y}_k|\boldsymbol{z}_k, \boldsymbol{x}_k, \boldsymbol{\beta}) p_{X_k|Z_k}(\boldsymbol{x}_k|\boldsymbol{z}_k, \mu_x, \sigma_x^2) p_U(\boldsymbol{w}_k - \boldsymbol{x}_k)\ d\boldsymbol{x}_k \\ &= \int l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_k, \boldsymbol{y}_k) p_{X_k|Z_k}(\boldsymbol{x}_k|\boldsymbol{z}_k, \mu_x, \sigma_x^2) p_U(\boldsymbol{w}_k - \boldsymbol{x}_k)\ d\boldsymbol{x}_k. \end{aligned}$$

The estimation of the probability density of the measurement error, $p_U(\bullet)$ is based on Appendix A.2. Using MCMC approximation, we then finally calculate the marginal likelihood as follows,

$$L_k(\boldsymbol{\beta}, 0) \approx \frac{1}{M_c} \sum_{m_c=1}^{M_c} l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_k, \boldsymbol{y}_k) p_{X_k|Z_k}(\boldsymbol{x}_k|\boldsymbol{z}_k, \mu_x, \sigma_x^2) \hat{p}_U(\boldsymbol{w}_k - \boldsymbol{x}_k),$$

where M_c is the number of MCMC samples after burn-in time.

A.4 Estimating Equation

Before explaining estimating equation for our flexible omnibus test statistic, we first define some notations and definitions in terms of tensor notation as follows,

- $A \equiv (A_i^j), \quad i = 1, \ldots, M+1, \quad j = 1, ..., q+1$, where i and j indicate the row index and the column index of the matrix A, respectively;

- $(AX)_i \equiv \sum_{k=1}^{q+1} A_i^k X_k, \quad k = 1, \ldots, q+1$, where X is a $(q+1) \times 1$ vector;

- $(AB)_i^j \equiv A_i^k B_k^j$, where B is a matrix with $(q+1)$ rows;
- $X^T Y \equiv (X^T)^i Y_i \equiv \sum_i^{q+1} x_i y_i = X \cdot Y$, where X and Y are $(q+1) \times 1$ vectors;
- $(ABC)_i^j \equiv A_i^k B_k^l C_l^j$;
- $(X \circ Y)_k \equiv X_k Y_k$, where $\circ$ is Hadamard product for elementwise product of either vectors or matrices;
- $(X \circ A)_k^j \equiv X_k A_k^j$;
- We define $(A \diamond B)_i^j \equiv A_i^j B_i^j$ which represent an array $(M+1) \times \{(q+1) \times (q+1)\}$, where A and B are $(M+1) \times (q+1)$ matrices.

We then derive the estimating equations for testing H_{i1} in Section A.4.1, H_{i2} in Section A.4.2, and for testing H_{i3} in Section A.4.3, respectively.

A.4.1 Estimating Equations for Testing H_{i1}

After sampling the latent variable $\boldsymbol{x}$, we need to estimate $\boldsymbol{\beta}$ under the null model and to construct the score-type test statistic. Newton-Raphson (NR) method is used to estimate the coefficient of the covariate. Define $\boldsymbol{\beta} = (\boldsymbol{\beta}_0^T, \beta_1)^T$ and $\boldsymbol{\mathcal{X}}^* = (\boldsymbol{Z}^*, \boldsymbol{X}^*)$, where $\boldsymbol{Z}^* = \boldsymbol{Z}_j - \boldsymbol{Z}_1$ and $\boldsymbol{X}^* = \boldsymbol{X}_j - \boldsymbol{X}_1$, $j = 1, \ldots, M+1$. Then, the log-likelihood function of the null model for the kth stratum can be expressed as

$$l_k(\boldsymbol{\beta}, \gamma = 0) \;=\; \log\left(\frac{1}{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})}\right) = -\log[1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})],$$

where $\mathbf{1}$ is a $(M+1) \times 1$ vector consisting of all 1s. Using the log-likelihood function, the score-function $S(\boldsymbol{\beta})$ is obtained as

$$S(\boldsymbol{\beta}) \;=\; \sum_{k=1}^{K} \frac{\partial l_k(\boldsymbol{\beta}, 0)}{\boldsymbol{\beta}^T} = \sum_{k=1}^{K} \frac{-\mathbf{1}^T \{\exp(\mathcal{X}_k^* \boldsymbol{\beta}) \circ \mathcal{X}_k^*\}}{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})},$$

which is a $1 \times (q+1)$ vector. The Hessian matrix $H(\boldsymbol{\beta})$ can be derived as

$$\begin{aligned} H(\boldsymbol{\beta}) &= \sum_{k=1}^{K} \frac{\partial^2 l_k(\boldsymbol{\beta}, 0)}{\boldsymbol{\beta}\boldsymbol{\beta}^T} \\ &= \sum_{k=1}^{K} \left(\frac{-\mathbf{1}^T[\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \mathcal{X}_k^*\} \diamond \mathcal{X}_k^*]\{1 + \mathbf{1}^T \exp(\mathcal{X}_k^*\boldsymbol{\beta})\}}{\{1 + \mathbf{1}^T \exp(\mathcal{X}_k^*\boldsymbol{\beta})\}^2} + \frac{[\mathbf{1}^T\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \mathcal{X}_k^*\}]^T[\mathbf{1}^T\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \mathcal{X}_k^*\}]}{\{1 + \mathbf{1}^T \exp(\mathcal{X}_k^*\boldsymbol{\beta})\}^2} \right), \end{aligned}$$

which is a $(q+1) \times (q+1)$ matrix. The estimated coefficient of the $(t+1)$th NR iteration can be obtained from

$$\widehat{\boldsymbol{\beta}}^{(t+1)} = \widehat{\boldsymbol{\beta}}^{(t)} - H(\widehat{\boldsymbol{\beta}}^{(t)})^{-1} S(\widehat{\boldsymbol{\beta}}^{(t)})$$

until $||\widehat{\boldsymbol{\beta}}^{(t+1)} - \widehat{\boldsymbol{\beta}}^{(t)}||_2 < 10^{-6}$ using L_2 norm.

Based on the estimated coefficient from the null model, we can construct the estimating equations, for the alternative model by using some basis function $h(\cdot)$, notation $\boldsymbol{h}^*(\cdot)$ which represents a $(M+1) \times 1$ vector consisting of $h(x_{km_cj}) - h(x_{km_c1})$, where $\boldsymbol{x}^*_{km_c}$ is sample at the m_cth MCMC for the latent variable $\boldsymbol{X}_k^*$, $j = 1, \ldots, M+1$, and $\boldsymbol{\mathcal{X}}^*_{km_c} = (\boldsymbol{z}_k^*, \boldsymbol{x}^*_{km_c})$.

The estimating equations $\phi_{\beta,k}$ under the null and $\psi_{\gamma,k}$ under the alternative model such as $[1 + \mathbf{1}^T \exp\{\mathcal{X}^*_{km_c}\boldsymbol{\beta} + \boldsymbol{h}^*(\boldsymbol{x}_{km_c})\gamma\}]^{-1}$ for the kth stratum are then

$$\begin{aligned} \phi_{\boldsymbol{\beta},k} &= \frac{\sum_{m_c=1}^{M_c} C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}]}{\sum_{m_c=1}^{M_c} D_{km_c}}; \\ \psi_{\gamma,k} &= \frac{\sum_{m_c=1}^{M_c} C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \boldsymbol{h}^*(\boldsymbol{x}_{km_c})\}]}{\sum_{m_c=1}^{M_c} D_{km_c}}, \end{aligned}$$

where

$$\begin{aligned} C_{km_c} &= \{l(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^2 p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2); \\ D_{km_c} &= l(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k) p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2). \end{aligned}$$

Using estimating equations, we then obtain A_1 and A_2 as follows,

$$A_1 = \mathrm{E}\left[\frac{\partial \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}\right] = \frac{1}{K}\sum_{k=1}^{K} \frac{\partial \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}$$

$$= \frac{1}{K}\sum_{k=1}^{K}\left\{\frac{\sum_{m_c=1}^{M_c} \text{Ⓐ} \sum_{m_c=1}^{M_c} D_{km_c}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2} - \frac{\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}])\}^T\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}])\}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2}\right\};$$

$$A_2 = \mathrm{E}\left[\frac{\partial \psi_{\gamma,k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}\right] = \frac{1}{K}\sum_{i=1}^{K} \frac{\partial \psi_{\gamma,k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T},$$

where

$$\text{Ⓐ} \equiv \Big[\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^2 \mathbf{1}^T[\{\exp(\mathcal{X}^*_k\boldsymbol{\beta}) \circ \mathcal{X}^*_k\} \diamond \mathcal{X}^*_k]$$
$$- 2\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^3[\mathbf{1}^T\{\exp(\mathcal{X}^*_k\boldsymbol{\beta}) \circ \mathcal{X}^*_k\}]^T[\mathbf{1}^T\{\exp(\mathcal{X}^*_k\boldsymbol{\beta}) \circ \mathcal{X}^*_k\}]$$
$$\times p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2).$$

Using all these notations, our statistic can be obtained as follows,

$$\widehat{U} = \frac{1}{\sqrt{K}}\sum_{k=1}^{K} \psi_{\gamma,k}(\widehat{\boldsymbol{\beta}}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)$$

and

$$T = \widehat{U}^T\widehat{\Sigma}_0^{-1}\widehat{U} \sim \chi^2,$$

where

$$\Sigma_0 = \mathrm{cov}\{\psi_{\gamma,k}(\cdot, \boldsymbol{\beta}, 0, \hat{p}_u) - A_2A_1^{-1}\phi_{\boldsymbol{\beta},k}(\cdot, \boldsymbol{\beta}, 0, \hat{p}_u)\}.$$

A.4.2 Estimating Equations for Testing H_{i2}

For testing the effect modification by the matching covariate V, we let $\boldsymbol{\beta}(V) = (\boldsymbol{\beta}_0, \beta(V))$ where $\beta(V) = \beta_1 + \theta(V)$ and $\theta(\bullet)$ is an unspecified function which is orthogonal to V. For the null hypothesis, $\theta(V) = 0$ so that $\boldsymbol{\beta}(V) = (\boldsymbol{\beta}_0, \beta_1) = \boldsymbol{\beta}$. The covariates are denoted by $\boldsymbol{\mathcal{X}}^* = (\boldsymbol{Z}^*, \boldsymbol{X}^*)$ like Section A.4.1. Then, the log-likelihood function based on the null model for the kth stratum is represented as

$$l_k(\boldsymbol{\beta}, \theta(V) = 0) \;=\; \log\left(\frac{1}{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})}\right) = -\log[1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})].$$

Based on the above log-likelihood, we can obtain the score function, $S(\boldsymbol{\beta})$ and the hessian matrix, $H(\boldsymbol{\beta})$ as the following equations:

$$S(\boldsymbol{\beta}) \;=\; \sum_{v \in \mathcal{V}} \sum_{k=1}^{K} \frac{\partial l_k(\boldsymbol{\beta}, \theta(v) = 0)}{\boldsymbol{\beta}^T} = \sum_{v \in \mathcal{V}} \sum_{k=1}^{K} \frac{-\mathbf{1}^T \{\exp(\mathcal{X}_k^* \boldsymbol{\beta}) \circ \mathcal{X}_k^*\}}{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})}\Bigg|_{\theta(v)=0},$$

where $\mathcal{V}$ is a support of the matching covariate V which is assumed to be known. The hessian matrix is

$$\begin{aligned} H(\boldsymbol{\beta}) \;&=\; \sum_{v \in \mathcal{V}} \sum_{k=1}^{K} \frac{\partial^2 l_k(\boldsymbol{\beta}, \theta(v) = 0)}{\boldsymbol{\beta}\boldsymbol{\beta}^T} \\ &=\; \sum_{v \in \mathcal{V}} \sum_{k=1}^{K} \left(\frac{-\mathbf{1}^T [\{\exp(\mathcal{X}_k^* \boldsymbol{\beta}) \circ \mathcal{X}_k^*\} \diamond \mathcal{X}_k^*] \{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})\}}{\{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})\}^2} \right. \\ &+\; \left. \frac{[\mathbf{1}^T \{\exp(\mathcal{X}_k^* \boldsymbol{\beta}) \circ \mathcal{X}_k^*\}]^T [\mathbf{1}^T \{\exp(\mathcal{X}_k^* \boldsymbol{\beta}) \circ \mathcal{X}_k^*\}]}{\{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})\}^2} \right) \Bigg|_{\theta(v)=0}. \end{aligned}$$

Using the score function and the Hessian, the estimates under the null can be obtained from the NR iteration.

After obtaining the estimated coefficient, we employ some basis function $h(\cdot)$ for the alternative model such as $p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k, \mathcal{X}_k^* \boldsymbol{\beta} + \gamma_{i2} h_i(V)\}$, $i = 1, \ldots, I$. Then, the estimating equations of the effect modification under null model and alternative model, [1 +

$\mathbf{1}^T \exp\{\mathcal{X}^*_{km_c}\boldsymbol{\beta} + \gamma h(V)\}]^{-1}$ for the kth stratum, can be derived as follows,

$$\phi_{\boldsymbol{\beta},(k,v)} = \frac{\sum_{m_c=1}^{M_c} C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}]}{\sum_{m_c=1}^{M_c} D_{km_c}};$$
$$\psi_{\gamma,(k,v)} = \frac{\sum_{m_c=1}^{M_c} C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ (h(V)\mathcal{X}^*_{km_c})\}]}{\sum_{m_c=1}^{M_c} D_{km_c}},$$

where

$$C_{km_c} = \{l(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k, V)\}^2 p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2);$$
$$D_{km_c} = l(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k, V) p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2),$$

and $h(V)$ is the basis function of the matching covariate V.

We then obtain A_1 and A_2 as follows,

$$A_1 = \mathrm{E}\left[\frac{\partial \phi_{\boldsymbol{\beta},(k,v)}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}\right] = \frac{1}{KN_V}\sum_{v\in\mathcal{V}}\sum_{k=1}^{K}\frac{\partial \phi_{\boldsymbol{\beta},(k,v)}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}$$
$$= \frac{1}{KN_V}\sum_{v\in\mathcal{V}}\sum_{k=1}^{K}\left\{\frac{\sum_{m_c=1}^{M_c} \mathbb{B} \sum_{m_c=1}^{M_c} D_{km_c}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2}\right.$$
$$\left. - \frac{\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}])\}^T\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}])\}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2}\right\};$$
$$A_2 = \mathrm{E}\left[\frac{\partial \psi_{\gamma,(k,v)}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}\right] = \frac{1}{KN_V}\sum_{v\in\mathcal{V}}\sum_{i=1}^{K}\frac{\partial \psi_{\gamma,(k,v)}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T},$$

where N_v is the number of the observations in matching covariate V and

$$\mathbb{B} \equiv \Big[\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k, V)\}^2 \mathbf{1}^T[\{\exp(\mathcal{X}^*_k\boldsymbol{\beta}) \circ \mathcal{X}^*_k\} \diamond \mathcal{X}^*_k]$$
$$- 2\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k, V)\}^3[\mathbf{1}^T\{\exp(\mathcal{X}^*_k\boldsymbol{\beta}) \circ \mathcal{X}^*_k\}]^T[\mathbf{1}^T\{\exp(\mathcal{X}^*_k\boldsymbol{\beta}) \circ \mathcal{X}^*_k\}]$$
$$\times p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2).$$

Hence, the score-type test statistic in testing the effect modification can be constructed as

follows,

$$\widehat{U} = \frac{1}{\sqrt{KN_V}} \sum_{v \in \mathcal{V}} \sum_{k=1}^{K} \psi_{\gamma,(k,v)}(\widehat{\boldsymbol{\beta}}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k).$$

T and Σ_0 are defined as the same to Section A.4.1 except for using $\psi_{\gamma,(k,v)}$.

A.4.3 Estimating Equations for Testing H_{i3}

In the testing of the interaction effect between the fixed covariate $\boldsymbol{Z}$ and the latent variable $\boldsymbol{X}$, we define $\boldsymbol{\beta} = (\boldsymbol{\beta}_0, \beta_1)$, $\mathcal{X}^* = (\boldsymbol{Z}^*, \boldsymbol{X}^*)$ and $\theta(\boldsymbol{Z}, \boldsymbol{X})$ is an unspecified interaction effect between $\boldsymbol{Z}$ and $\boldsymbol{X}$. The log-likelihood function for the kth stratum under the null model is expressed as

$$l_k(\boldsymbol{\beta}, \theta(\boldsymbol{Z}, \boldsymbol{X}) = 0) \;=\; \log\left(\frac{1}{1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})}\right) = -\log[1 + \mathbf{1}^T \exp(\mathcal{X}_k^* \boldsymbol{\beta})].$$

Under the null hypothesis that $\theta(\boldsymbol{Z}, \boldsymbol{X}) = 0$, the score function, $S(\boldsymbol{\beta})$ and the hessian matrix, $H(\boldsymbol{\beta})$ are same as those in Section A.4.1. From the NR method, the null coefficient is estimated.

In the case of testing the interaction effect, we use some basis function $h(\boldsymbol{Z}, \boldsymbol{X})$ for the alternative model such as $p_{Y|Z,X,\sum Y=1}\{\boldsymbol{Y}_k, \mathcal{X}_k^* \boldsymbol{\beta} + \gamma_{i3} h_i^*(\boldsymbol{Z}_k, \boldsymbol{X}_k)\}$ where $h_i^*(\boldsymbol{Z}_k, \boldsymbol{X}_k) = h_i(\boldsymbol{Z}_{kj}, X_{kj}) - h_i(\boldsymbol{Z}_{k1}, X_{k1})$, $i = 1, \ldots, I$ and $j = 1, \ldots, M+1$. Using the basis function $h(\cdot)$ and the estimated coefficient under the null model and alternative model $[1 + \mathbf{1}^T \exp\{\mathcal{X}_{km_c}^* \boldsymbol{\beta} + \boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})\gamma\}]^{-1}$, we can derive the estimating equations as follows,

$$\phi_{\boldsymbol{\beta},k} = \frac{\sum_{m_c=1}^{M_c} C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}_{km_c}^* \widehat{\boldsymbol{\beta}}) \circ \mathcal{X}_{km_c}^*\}]}{\sum_{m_c=1}^{M_c} D_{km_c}};$$
$$\psi_{\gamma,k} = \frac{\sum_{m_c=1}^{M_c} C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}_{km_c}^* \widehat{\boldsymbol{\beta}}) \circ \boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})\}]}{\sum_{m_c=1}^{M_c} D_{km_c}},$$

where

$$C_{km_c} = \{l(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^2 p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2);$$
$$D_{km_c} = l(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k) p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2),$$

and $\boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})$ is a $(M+1)\times 1$ vector for the alternative model, $[1 + \mathbf{1}^T \exp\{\mathcal{X}^*_{km_c}\boldsymbol{\beta} + \boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})\gamma\}]^{-1}$.

Using $\phi_{\boldsymbol{\beta},k}$ and $\psi_{\gamma,k}$, A_1 and A_2 can be obtained as follows,

$$A_1 = \mathrm{E}\left[\frac{\partial \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}\right] = \frac{1}{K}\sum_{k=1}^{K} \frac{\partial \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}$$
$$= \frac{1}{K}\sum_{k=1}^{K}\left[\frac{\sum_{m_c=1}^{M_c} \text{Ⓐ} \sum_{m_c=1}^{M_c} D_{km_c}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2}\right.$$
$$\left. - \frac{\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}])\}^T\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \mathcal{X}^*_{km_c}\}])\}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2}\right];$$
$$A_2 = \mathrm{E}\left[\frac{\partial \psi_{\gamma,k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}\right] = \frac{1}{K}\sum_{i=1}^{K} \frac{\partial \psi_{\gamma,k}(\boldsymbol{\beta}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k)}{\partial \boldsymbol{\beta}^T}$$
$$= \frac{1}{K}\sum_{k=1}^{K}\left[\frac{\sum_{m_c=1}^{M_c} \text{Ⓒ} \sum_{m_c=1}^{M_c} D_{km_c}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2}\right.$$
$$\left. - \frac{\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})\}])\}^T\{\sum_{m_c=1}^{M_c}(C_{km_c}[\mathbf{1}^T\{\exp(\mathcal{X}^*_{km_c}\widehat{\boldsymbol{\beta}}) \circ \boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})\}])\}}{(\sum_{m_c=1}^{M_c} D_{km_c})^2}\right.$$

where

$$\begin{aligned}
\text{Ⓐ} \equiv & \Big[\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^2 \mathbf{1}^T[\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \mathcal{X}_k^*\} \diamond \mathcal{X}_k^*] \\
& - 2\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^3[\mathbf{1}^T\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \mathcal{X}_k^*\}]^T[\mathbf{1}^T\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \mathcal{X}_k^*\}] \\
& \times p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2); \\
\text{Ⓒ} \equiv & \Big[\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^2 \mathbf{1}^T[\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})\} \diamond \mathcal{X}_k^*] \\
& - 2\{l_k(\boldsymbol{\beta}, 0|\boldsymbol{z}_k, \boldsymbol{x}_{km_c}, \boldsymbol{y}_k)\}^3[\mathbf{1}^T\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \boldsymbol{h}^*(\boldsymbol{z}_k, \boldsymbol{x}_{km_c})\}]^T[\mathbf{1}^T\{\exp(\mathcal{X}_k^*\boldsymbol{\beta}) \circ \mathcal{X}_k^*\}] \\
& \times p_x(\boldsymbol{x}_{km_c}|\mu_x, \sigma_x^2)\hat{p}_u(\boldsymbol{w}_k - \boldsymbol{x}_{km_c}|0, \sigma_u^2).
\end{aligned}$$

The score-type test statistic for identifying the interaction effect can then be derived as follows,

$$\widehat{U} = \frac{1}{\sqrt{K}} \sum_{k=1}^{K} \psi_{\gamma,k}(\widehat{\boldsymbol{\beta}}, 0, \boldsymbol{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k).$$

T and Σ_0 are defined as the same to Section A.4.1 except for using $\psi_{\gamma,k}$.

A.5 Asymptotic Theory

Let p be the fixed number of parameter, where $p < K$. Consider that $K \to \infty$ for this argument to work.

Since $\widehat{\boldsymbol{\beta}}$ is based on the null model, it solves $0 = \sum_{k=1}^{K} \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \mathbf{Z}_k, \boldsymbol{W}_k, Y_k, \hat{p}_U)$ for any $\boldsymbol{\beta}^*$, not even subject to the constraint, so its asymptotic expansion is

$$0 = K^{-1/2}\textstyle\sum_{k=1}^{K} \phi_{\boldsymbol{\beta},k}(\widehat{\boldsymbol{\beta}}, 0, \mathbf{Z}_k, \boldsymbol{W}_k, \mathbf{Y}_k, \hat{p}_U) = K^{-1/2}\sum_{k=1}^{K} \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \mathbf{Z}_k, \boldsymbol{W}_k, \mathbf{Y}_k, \hat{p}_U) + A_1 K^{1/2}(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})$$

since when $\gamma = 0$ so that

$$K^{1/2}(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}) = -A_1^{-1} K^{-1/2} \textstyle\sum_{k=1}^{K} \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \mathbf{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k, \hat{p}_U). \tag{A.2}$$

Thus, there is a $\boldsymbol{\beta}^*$ such that

$$\begin{aligned} \widehat{U} &= K^{-1/2} \textstyle\sum_{k=1}^{K} \phi_{\boldsymbol{\beta},k}(\widehat{\boldsymbol{\beta}}, 0, \mathbf{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k, \hat{p}_U) \\ &= K^{-1/2} \textstyle\sum_{k=1}^{K} \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \mathbf{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k, \hat{p}_U) \\ &\quad + K^{-1/2} \textstyle\sum_{k=1}^{K} \dfrac{\partial \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}^*, 0, \mathbf{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k, \hat{p}_U)}{\partial \boldsymbol{\beta}^{\mathrm{T}}} K^{1/2}(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*). \end{aligned}$$

From this,

$$K^{1/2}(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*) = -K^{1/2} A_1^{-1} K^{-1} \sum_{k=1}^{K} \phi_{\boldsymbol{\beta},k}(\boldsymbol{\beta}, 0, \mathbf{Z}_k, \boldsymbol{W}_k, \boldsymbol{Y}_k, \hat{p}_U) \sim N(0, V_{\boldsymbol{\beta}})$$

and $\widehat{U} \to N(0, \Sigma)$. Hence $\widehat{T} = \widehat{U}^T \widehat{\Sigma}^{-1} \widehat{U}$ is asymptotically χ^2 with p_T degrees of freedom.

A.6 Adjusting Covariance Matrix for Non-negative Definite

As described in Section 6.3, the number of missing values due to the negative-definite covariance matrices is drastically increasing as the number of basis functions increases given by the small number of strata. To avoid this numerical problem, we adjust the covariance matrix with diagonal matrix with small control value δ, which is

$$\Sigma^* = \Sigma + \boldsymbol{I}\delta,$$

where Σ is the negative-definite covariance matrix.

For the control value δ, it is set as very small value such as 10^{-8}, in general. Σ^* is the adjusted covariance matrix of Σ, which satisfies

$$\boldsymbol{v}\Sigma^*\boldsymbol{v}^T > 0 \qquad \forall \boldsymbol{v} \neq 0.$$

A.7 The Algorithm for The Nonparametric Estimation in The Matched Study

A.7.1 The Nonparametric Estimation for Nonlinear Functional Association

Step E1: Generate candidate samples for $\boldsymbol{X}$ through Markov chain Monte Carlo (MCMC) algorithm described in Appendix A.3.

Step E2: Based on the generated samples for $\boldsymbol{X}$, define a pth order regression spline with κ knots as $\{x^*, ..., x^{*p}, (x-\xi_1)_+^{*p}, ..., (x-\xi_\kappa)_+^{*p}\}$, where p is any integer for $p \geq 2$, ξs are the knots, $\xi_1 < \cdots < \xi_\kappa$, $(s)_+^p = s^p I(s \geq 0)$, $x^* = x_j - x_1$, and $(x-\xi)_+^{*p} = (x_j - \xi)_+^p - (x_1 - \xi)_+^p$ with $j = 1, \ldots, M+1$

Step E3: Estimate the parameters nonparametrically by using the penalized log-likelihood obtained from the retrospective model as follows,

$$l_p(\boldsymbol{\beta}_P, \boldsymbol{\beta}_N) = -\sum_{k=1}^{K} \log\left[1 + \sum_{j=2}^{M+1} \exp\left\{\boldsymbol{Z}_{jk}^*\boldsymbol{\beta}_0 + \boldsymbol{X}_{jk}^{*T}(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2)\right\}\right] + \frac{\lambda}{2}\boldsymbol{\beta}_{\mathbf{2}}^T\boldsymbol{\beta}_{\mathbf{2}},$$

where

$$\boldsymbol{X}_{jk}^{*T} = [X_{jk}^{*}, ..., X_{jk}^{*p}, (X_{jk} - \xi_1)_{+}^{*p}, ..., (X_{jk} - \xi_\kappa)_{+}^{*p}],$$

where $\boldsymbol{\beta}_1 = (\beta_1, \ldots, \beta_p)^T$, $\boldsymbol{\beta}_2 = (\beta_{p+1}, ..., \beta_{p+KN})^T$, $\boldsymbol{\beta}_P = (\boldsymbol{\beta}_0, \boldsymbol{\beta}_1)$ and $\boldsymbol{\beta}_N = (\boldsymbol{\beta}_2)$, and λ is the smoothing parameter.

The NR method is employed to estimate $\boldsymbol{\beta}_P$ and $\boldsymbol{\beta}_N$. For estimating the smoothing parameter λ, we used a grid search within a pre-specified range and use the Bayesian information criterion (BIC),

$$BIC(\lambda) = (q + p + \kappa)\log(K) - 2\widehat{l}_p(\boldsymbol{\beta}_P, \boldsymbol{\beta}_N),$$

where the dimensions of $\boldsymbol{\beta}_P$ and $\boldsymbol{\beta}_N$ are $q + p$ and κ, respectively.

These procedures are summarized in Algorithm 3.

Algorithm 3 Algorithm for nonparametric estimation for nonlinear functional association

1: Generate candidate samples for $\boldsymbol{X}$ through MCMC algorithm mentioned in Appendix A.3;

2: Define a pth order regression spline with κ knots as $\{x^*, ..., x^{*p}, (x-\xi_1)_+^{*p}, ..., (x-\xi_\kappa)_+^{*p}\}$, where p is any integer for $p \geq 2$, ξs are the knots, $\xi_1 < \cdots < \xi_\kappa$, $(s)_+^p = s^p I(s \geq 0)$, $x^* = x_j - x_1$, and $(x-\xi)_+^{*p} = (x_j - \xi)_+^p - (x_1 - \xi)_+^p$ with $j = 1, \ldots, M+1$.

3: Let $\boldsymbol{\beta}_P = (\boldsymbol{\beta}_0, \boldsymbol{\beta}_1)$ and $\boldsymbol{\beta}_N = (\boldsymbol{\beta}_2)$ be a vector of the parametric and nonparametric coefficients, respectively. **for** $iter = 1$ *to* $Iter$ **do**

4:
end

Set the grid points for the smoothing parameter λ such as $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ where the length of interval between points is $10^{-(iter-1)}$; **for** $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ **do**

5:
end

Estimate $\boldsymbol{\beta}_P$ and $\boldsymbol{\beta}_N$ with NR method using the penalized loglikelihood obtained from the retrospective model, $l_p(\boldsymbol{\beta}_P, \boldsymbol{\beta}_N) = -\sum_{k=1}^{K} \log \left[1 + \sum_{j=2}^{M+1} \exp\left\{\boldsymbol{Z}_{jk}^* \boldsymbol{\beta}_0 + \boldsymbol{X}_{jk}^{*T}(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2)\right\}\right] + \frac{\lambda}{2}\boldsymbol{\beta}_{\boldsymbol{2}}^T \boldsymbol{\beta}_{\boldsymbol{2}}$ where $\boldsymbol{X}_{jk}^{*T} = [X_{jk}^*, ..., X_{jk}^{*p}, (X_{jk} - \xi_1)_+^{*p}, ..., (X_{jk} - \xi_\kappa)_+^{*p}]$;

6: Estimate λ minimizing the Bayesian information criterion (BIC), where $BIC(\lambda) = (q + p + \kappa)\log(K) - 2\hat{l}_p(\boldsymbol{\beta}_P, \boldsymbol{\beta}_N)$;

7: Obtain the final estimates $\widehat{\boldsymbol{\beta}}_P$ and $\widehat{\boldsymbol{\beta}}_N$ with the optimal smoothing parameter $\hat{\lambda}$.

A.7.2 The Nonparametric Estimation with Varying Coefficient:

Step E1: Generate candidate samples for $\boldsymbol{X}$ through Markov chain Monte Carlo (MCMC) algorithm mentioned in Appendix C.

Step E2: Unlike G.1, estimating parameters is very sensitive to the choice of the smoothing parameter. Hence by employing the mixed model framework, we estimate the smoothing parameter. That is, using the pth polynomial regression spline basis with κ knots such as $\{1, v, ..., v^p, (v-\xi_1)_+^p, ..., (v-\xi_\kappa)_+^p\}$, the varying coefficient $\beta(V)$ can be expressed as

$$\beta(v) = \alpha_0 + \alpha_1 v + ... + \alpha_p v^p + \sum_{kn=1}^{\kappa} \alpha_{p+kn}(v-\xi_{kn})_+$$

and $\boldsymbol{\alpha}_P = (\alpha_0, ..., \alpha_p)^T$ are the fixed parameters but $\boldsymbol{\alpha}_N = (\alpha_{p+1}, ..., \alpha_{p+\kappa})$ treat as the random parameters $\boldsymbol{\alpha}_N \sim N(\mathbf{0}, \phi_2^{-1}\boldsymbol{I}_\kappa)$, where ϕ_2 is smoothing parameter.

Step E3: Set the initial value for $\boldsymbol{\alpha}^{(0)}$ as all zeros. Initialize $\phi_2^{(0)}$ by generating sample from Gamma(a, b). By using the NR method, obtain the estimated coefficients for $\boldsymbol{\alpha}_P$, $\widehat{\boldsymbol{\alpha}}_P^{(1)}$.

Step E4: Specify the prior distribution of the remaining parameters as follows,

$$\boldsymbol{\alpha}_N \sim N(\mathbf{0}, \phi_2^{-1}\boldsymbol{I}_\kappa), \quad \phi_2 \sim \text{Gamma}(a, b),$$

where $\boldsymbol{I}_\kappa$ is the κ-dimensional identity matrix, and (a, b) are hyper-parameters,

and calculate the joint likelihood function

$$\mathcal{L}(\boldsymbol{\alpha}_N, \phi_2) = \left[\prod_{k=1}^{K} \frac{1}{1+\sum_{j\neq 3}^{5} \exp\left\{(\boldsymbol{Z}_{jk}-\boldsymbol{Z}_{1k})^T\boldsymbol{\beta}_0 + \beta(v)(X_{jk}-X_{1k})\right\}}\right] \times\left[\frac{\phi_2^2}{2\pi}\exp\Big(-\frac{\phi_2\boldsymbol{\alpha}_N^T\boldsymbol{\alpha}_N}{2}\Big)\right] \times \left[\frac{b^a}{\Gamma(a)}\phi_2^{a-1}\exp\Big(-\phi_2 b\Big)\right].$$

and also obtain the complete conditional likelihoods for $\boldsymbol{\alpha}_N$ and ϕ_2 are as follows

$$p(\boldsymbol{\alpha}_N|\boldsymbol{\alpha}_P, \phi_2) \propto \left[\prod_{k=1}^{K} \frac{1}{1+\sum_{j\neq 3}^{5} \exp\left\{(\boldsymbol{Z}_{jk}-\boldsymbol{Z}_{1k})^T\boldsymbol{\beta}_0 + \beta(v)(X_{jk}-X_{1k})\right\}}\right] \times \left[\exp\Big(-\frac{\phi_2\boldsymbol{\alpha}_N^T\boldsymbol{\alpha}_N}{2}\Big)\right];$$

$$\phi_2|\boldsymbol{\alpha}_P, \boldsymbol{\alpha_N} \sim \text{Gamma}\left\{a+\kappa, \Big(\sum_{kn=1}^{\kappa}\frac{\alpha_{kn+1}^2}{2}+b\Big)\right\}.$$

Given $\widehat{\boldsymbol{\alpha}}_P^{(1)}$ and $\phi_2^{(0)}$ in Step E3, estimate $\widehat{\boldsymbol{\alpha}}_N^{(1)}$ and $\hat{\phi}_2^{(1)}$ by using adaptive rejection metropolis sampling (ARMS) and Gibbs sampler, respectively.

Step E5: Set $\boldsymbol{\alpha} = (\widehat{\boldsymbol{\alpha}}_P^{(1)T}, \widehat{\boldsymbol{\alpha}}_N^{(1)T})^T$ and define $\Psi = (\boldsymbol{\alpha}, \phi_2)$. If $||\boldsymbol{\Psi}^{(1)} - \boldsymbol{\Psi}^{(0)}||_2 < tol.$, where e.g., $tol. = 10^{-6}$ in our case., stop the iteration. Otherwise, start over the algorithm from Step E3.

These procedures are summarized in Algorithm 4.

Algorithm 4 Algorithm for nonparametric estimation for varying coefficient

1: Generate candidate samples for $\boldsymbol{X}$ through MCMC algorithm mentioned in Appendix C;
2: Initialize $\boldsymbol{\alpha}^{(0)}$ and $\phi_2^{(0)} \sim \text{Gamma}(a, b)$;
3: t=1;
4: Update $\boldsymbol{\alpha}_P^{(t)}$ using the Newton-Raphson method given $\boldsymbol{\alpha}_2^{(t-1)}$;
5: Given $\boldsymbol{\alpha}_P^{(t)}$ and $\phi_2^{(t-1)}$, update $\boldsymbol{\alpha}_N^{(t)}$ by using ARMS algorithm;
6: Given $\boldsymbol{\alpha}_N^{(t)}$, update $\phi_2^{(t)}$ using Gibbs sampler from the full conditional distribution;
7: Define $\Psi^{(t)} = (\widehat{\boldsymbol{\alpha}}^{(t)}, \hat{\phi}_2^{(t)})$. $t \leftarrow t+1$; $||\boldsymbol{\Psi}^{(t)} - \boldsymbol{\Psi}^{(t-1)}||_2 < 10^{-6}$
8: Obtain the final estimates $\widehat{\boldsymbol{\alpha}} = (\widehat{\boldsymbol{\alpha}}_P, \widehat{\boldsymbol{\alpha}}_N)$.

A.8 Tables & Figures for Chapter 2

Table A.1: The average value of empirical type I errors and powers of the flexible omnibus test under Case 1(b); The two rows from the top and two rows from the bottom of the table represent the average values of type I error and power based on each nominal level (α) and the number of strata (K), respectively; p^*_X is the misspecified normal model for the latent variable X; For each combination of α and K value, we simulated 200 datasets with 1-4 matched case-crossover study with K strata.

		The values of K									
		100	**200**	**300**	**400**	**500**	**600**	**700**	**800**	**900**	**1000**
	Normal p^*_X										
	α =**0.01**	0.025	0.010	0.010	0.010	0.005	0.010	0.005	0.015	0.010	0.010
Type I error	**0.05**	0.060	0.045	0.060	0.080	0.040	0.065	0.040	0.045	0.065	0.070
	0.1	0.135	0.105	0.125	0.120	0.125	0.120	0.065	0.110	0.085	0.130
	Normal p^*_X										
	α =**0.01**	0.610	0.825	0.850	0.900	0.875	0.915	0.925	0.965	0.970	0.980
Power	**0.05**	0.820	0.940	0.955	0.980	0.955	0.980	0.980	0.985	0.990	1.000
	0.1	0.910	0.965	0.965	0.995	0.965	0.995	0.990	0.985	1.000	1.000

Table A.2: The average value of the empirical type I errors and powers of the flexible omnibus test under Case 2; The two rows from the top and two rows from the bottom of the table represent the average values of type I error and power based on each nominal level (α) and the number of strata (K), respectively; p^*_X is the misspecified normal model for the latent variable X; For each combination of α and K value, we simulated 200 datasets with 1-4 matched case-crossover study with K strata.

		The values of K									
		100	**200**	**300**	**400**	**500**	**600**	**700**	**800**	**900**	**1000**
	Normal p^*_X										
	α =**0.01**	0.020	0.005	0.010	0.030	0.000	0.020	0.010	0.020	0.015	0.010
Type I error	**0.05**	0.055	0.060	0.050	0.065	0.045	0.050	0.040	0.050	0.055	0.055
	0.1	0.130	0.115	0.090	0.125	0.075	0.095	0.100	0.095	0.105	0.105
	Normal p^*_X										
	α =**0.01**	0.625	0.705	0.815	0.845	0.885	0.905	0.945	0.925	0.960	0.970
Power	**0.05**	0.880	0.895	0.920	0.915	0.960	0.960	0.970	0.985	0.985	1.000
	0.1	0.920	0.980	0.950	0.955	0.970	0.975	0.985	0.995	0.995	1.000

Table A.3: The average values of the empirical type I errors and powers of the flexible omnibus test based on the number of basis; The average values of type I error and power based on each nominal level (α) and the number of basis (TN); For each combination of α and the number of basis functions, we simulated 200 datasets with 1-4 matched case-crossover study with K strata; NA indicates the number of missing values due to covariance matrices which are negative definite; The number of strata(K) is 100.

			The number of basis							
			2	**4**	**6**	**10**	**20**	**40**	**60**	**80**
		Normal p_X								
		0.01	0.010	0.000	0.005	0.005	0.005	0.000	0.000	NA
Type I error	$\boldsymbol{\alpha}$	**0.05**	0.045	0.0.035	0.045	0.040	0.046	0.069	0.000	NA
		0.1	0.095	0.100	0.095	0.110	0.107	0.115	0.000	NA
	NA		0	0	0	0	3	113	196	200
		Normal p_X								
		0.01	0.905	0.955	0.940	0.935	0.930	0.929	1.000	NA
Power	$\boldsymbol{\alpha}$	**0.05**	0.990	0.990	0.990	0.990	0.990	0.986	1.000	NA
		0.1	0.995	0.995	0.995	0.995	0.995	0.993	1.000	NA
	NA		0	0	0	0	0	59	180	200

Table A.4: The average values of the empirical type I errors and powers of the flexible omnibus test based on the number of basis; The average values of type I error and power based on each nominal level (α) and the number of basis (TN); For each combination of α and the number of basis functions, we simulated 200 datasets with 1-4 matched case-crossover study with K strata; NA indicates the number of missing values due to covariance matrices which are negative definite; The number of strata(K) is 600.

			The number of basis							
			2	**4**	**6**	**10**	**20**	**40**	**60**	**80**
		Normal p_X								
		0.01	0.010	0.010	0.010	0.015	0.010	0.010	0.010	0.016
Type I error	$\boldsymbol{\alpha}$	**0.05**	0.045	0.040	0.040	0.035	0.025	0.045	0.040	0.027
		0.1	0.095	0.105	0.095	0.095	0.080	0.080	0.086	0.086
	NA		0	0	0	0	0	0	2	15
		Normal p_X								
		0.01	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Power	$\boldsymbol{\alpha}$	**0.05**	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
		0.1	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	NA		0	0	0	0	0	0	0	1

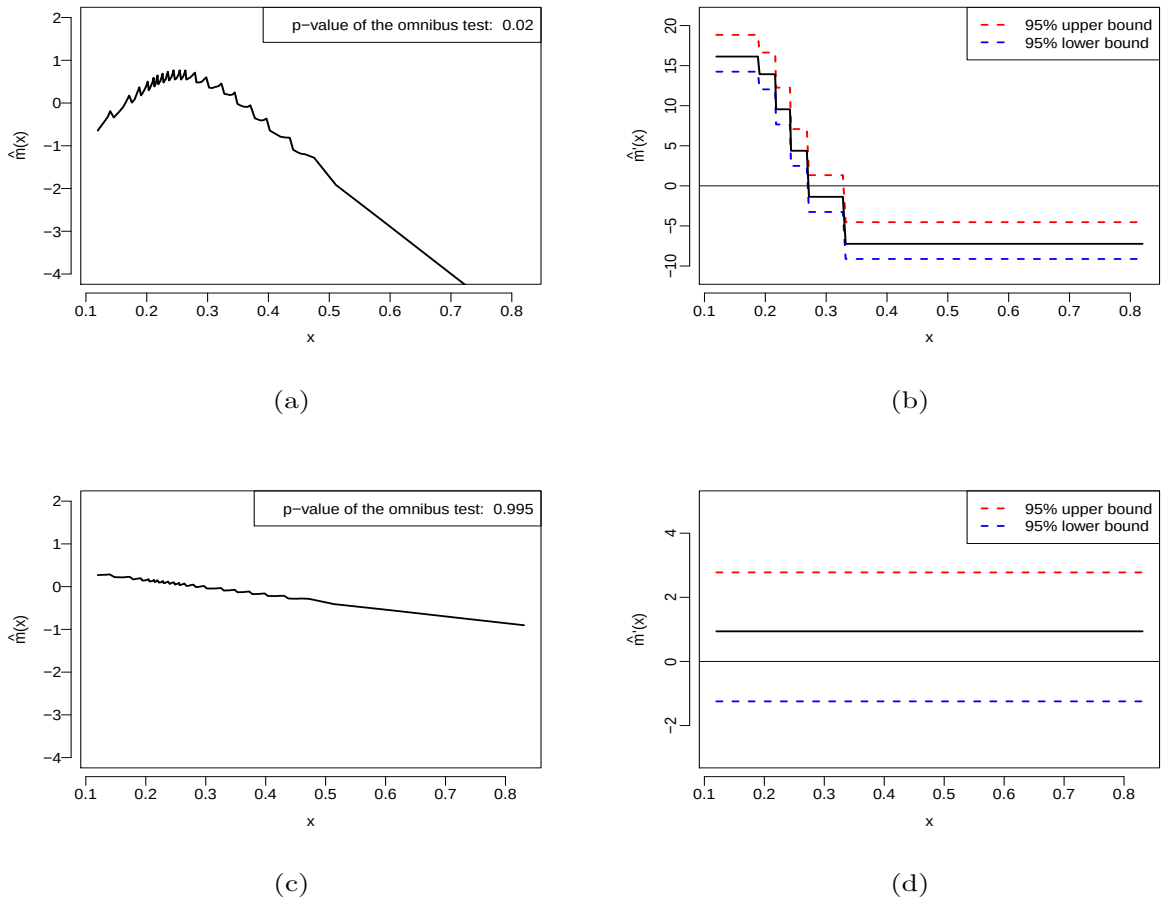


Figure A.1: Plots of nonparametric estimation for true mean function $m(x)$ on the 17th and 24th days; (a) Plots of the estimated mean function $\hat{m}(x)$ with p-value of the flexible omnibus test for nonlinear association on the 17th day. (b) Plots of the first derivative function of $\hat{m}(x)$ with 95% bootstrap confidence bands (dashed lines) on the 17th day. (c) Plots of the estimated mean function $\hat{m}(x)$ with p-value of the flexible omnibus test for nonlinear association on the 24th day. (d) Plots of the first derivative function of $\hat{m}(x)$ with 95% bootstrap confidence bands (dashed lines) on the 24th day.

Appendix B

Appendices for Chapter 3

B.1 The detailed procedure of joint methods

B.1.1 The second approximation of kernel K function ($JSKNR2$)

Step 1. Compute $\tilde{\boldsymbol{\alpha}} = (\lambda_0 I + K(Z, \boldsymbol{\xi}_D|\hat{\boldsymbol{\xi}}_{OD}))^{-1}(\boldsymbol{y} - \boldsymbol{X}\tilde{\boldsymbol{\beta}})$ where $\lambda_0 = \hat{\sigma}_\epsilon^2/\hat{\tau}$ with $\xi_j^{(0)} = \hat{\xi}_j(\hat{\rho})$ for all $j = 1, ..., p$. $\widehat{\boldsymbol{\psi}} = (\hat{\rho}, \hat{\tau}, \hat{\sigma}_\epsilon^2)$ is obtained by REML using the least square kernel machine. $\xi(\rho)$ depends on the corresponding kernel function. For example, $\xi(\rho) = \rho$ based on the linear polynomial kernel and $\xi(\rho) = 1/(c\rho)$ based on Gaussian kernel.

Step 2. Set the maximum of grid search range for λ_1 with $\xi_j = 0$ for all $j = 1, ..., p$ such that

$$\lambda_1^{(0)} = \underset{j}{\text{argmax}} \left\{ \frac{1}{n} \big(\tilde{\boldsymbol{y}} - \boldsymbol{X}\tilde{\boldsymbol{\beta}} - K(\boldsymbol{\xi} = \boldsymbol{0})\tilde{\boldsymbol{\alpha}}\big)^T (K_j'(\boldsymbol{\xi} = \boldsymbol{0})) \right\},$$

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

Step 3. Given the initial nonnegative garrote, $\xi_j^{(0)} = \hat{\xi}_j(\hat{\rho})$ for all $j = 1, ..., p$, update $\hat{\boldsymbol{\xi}}$ coordinate-wise by minimizing the objective funciton (3.17) based on the kernel

function with the 2nd approximation, (3.18) such that

$$\begin{aligned}
\hat{\xi}_j^{(t+1)} &= \underset{\xi_j}{\operatorname{argmin}}\, \tilde{Q}_1 \\
&= \underset{\xi_j}{\operatorname{argmin}}\, \frac{1}{2}||\tilde{\boldsymbol{y}} - \boldsymbol{X}\tilde{\boldsymbol{\beta}} - \tilde{K}(Z, \tilde{\boldsymbol{\xi}}_{-j}, \xi_j|\hat{\boldsymbol{\xi}}_{OD})\tilde{\boldsymbol{\alpha}}||^2 + \lambda_1^{(t+1)}||\boldsymbol{\xi}_D||_1 \\
&= \underset{\xi_j}{\operatorname{argmin}}\, \frac{1}{2}||\tilde{\boldsymbol{y}} - \boldsymbol{X}\tilde{\boldsymbol{\beta}} - \{K(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j) + K'(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j) + \frac{1}{2}K''(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j)^2\}\tilde{\boldsymbol{\alpha}}||^2 \\
&\quad +\lambda_1^{(t+1)}||\boldsymbol{\xi}_D||_1,
\end{aligned}$$

at the $t+1$ iteration. Note that $\tilde{\boldsymbol{\xi}}$ is the updated nonnegative garrote at the previous step. When $||\hat{\boldsymbol{\xi}}^{(t+1)} - \hat{\boldsymbol{\xi}}^{(t)}||_2 < tol.$, stop the iteration for the update. Otherwise, repeat Step 3 until the convergence.

Step 4. Decrease λ_1 gradually and obtain the updated $\boldsymbol{\xi}$s at each λ_1. Among the candidates, final estimate, $\hat{\boldsymbol{\xi}}$ is determined by minimizing the model selection criterion.

The detailed algorithm of $JSKNR2$ is described as follows.

Algorithm 5 Algorithm of $JSKNR2$

Input: Z, X, $\boldsymbol{y}$, $\boldsymbol{\xi}^{(0)}$ and $\Omega^{(0)}(\boldsymbol{\xi})$

Output: $\hat{\boldsymbol{\xi}}$ and $\widehat{\Omega}(\hat{\boldsymbol{\xi}})$

for *iter* $\leftarrow 1$ **to** *maximum iterations* **do**

REML for $\psi = (\tau, \rho, \sigma_\epsilon^2, \beta)$;

$\xi_j^{(0)} = \xi_j(\hat{\rho}), \quad j = 1, ..., p;$

$t \leftarrow 0$;

while $||\hat{\boldsymbol{\xi}}^{(t)} - \hat{\boldsymbol{\xi}}^{(t-1)}||_2 < tol_1$. **do**

for $j \leftarrow 1$ **to** p **do**

$$\begin{aligned}
\hat{\xi}_j^{(t)} &= \underset{\xi_j}{\operatorname{argmin}}\, \tilde{Q}_1 \\
&= \underset{\xi_j}{\operatorname{argmin}}\, \frac{1}{2}||\tilde{\boldsymbol{y}} - \boldsymbol{X}\tilde{\boldsymbol{\beta}} - \tilde{K}(Z, \tilde{\boldsymbol{\xi}}_{-j}, \xi_j|\hat{\boldsymbol{\xi}}_{OD})\tilde{\boldsymbol{\alpha}}||^2 + \lambda_1^{(t+1)}||\boldsymbol{\xi}_D||_1 \\
&= \underset{\xi_j}{\operatorname{argmin}}\, \frac{1}{2}||\tilde{\boldsymbol{y}} - \boldsymbol{X}\tilde{\boldsymbol{\beta}} - \{K(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j) + K'(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j) + \frac{1}{2}K''(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)(\hat{\xi}_j - \tilde{\xi}_j)^2\}\tilde{\boldsymbol{\alpha}}||^2 + \lambda_1^{(t+1)}||\boldsymbol{\xi}_D||_1
\end{aligned}$$

end

$t \leftarrow t + 1$;

end

Select non-zero components among the diagonals, $\hat{\boldsymbol{\xi}}^{(iter)} \in \mathbb{R}^p$ and

define $\boldsymbol{\xi}^{*(iter)} \in \mathbb{R}^{\hat{p}}$ as the selected components, $\hat{p} \leq p$;

Save the predictors corresponding to the selected components, $\boldsymbol{\xi}^{*(iter)}$ in $Z^* \in \mathbb{R}^{n \times \hat{p}}$;

Given $\boldsymbol{\xi}^{*(iter)}$ and Z^*, update the off-diagonals, $\hat{\xi}_{jj'}^{(iter)}$;

if $||\widehat{\Omega}^{(iter)} - \widehat{\Omega}^{(iter-1)}||_1 < tol.$ *and* $\hat{p}^{(iter)} = \hat{p}^{(iter-1)}$ **then**
Stop;

end

$Z \leftarrow Z^*$ and $p \leftarrow \hat{p}^{(iter)}$;

$iter \leftarrow iter + 1$;

end

B.1.2 The second approximation of a joint pseudo-likelihood function

The procedure of the algorithm using the 2nd approximation of the pseudo-likelihood function is similar to the algorithm 5, but the update for the nonnegative garrote is different.

Step 1. Same as the algorithm 5

Step 2. Same as the algorithm 5

Step 3. Given the initial nonnegative garrote, $\xi_j^{(0)} = \hat{\xi}_j(\hat{\rho})$ for all $j = 1, ..., p$, update $\hat{\boldsymbol{\xi}}$ coordinate-wise by minimizing the negative pseudo-likelihood function such that

$$\hat{\xi}_j^{(t+1)} = \underset{\xi_j}{\operatorname{argmin}} \Big[-\log f_{\boldsymbol{y}|\boldsymbol{Z}}(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{\xi}_D, \hat{\boldsymbol{\xi}}_{OD}) + \lambda_1^{(t+1)} ||\boldsymbol{\xi}_D||_1 \Big].$$

It is equivalent to update

$$\hat{\xi}_j^{(t+1)} = \tilde{\xi}_j - \left[L^{(2)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j) \right]^{-1} L^{(1)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j)$$

coordinate-wise at the $t+1$ iteration. $\tilde{\boldsymbol{\xi}}$ denotes the previously updated nonnegative garrote. When $||\hat{\boldsymbol{\xi}}^{(t+1)} - \hat{\boldsymbol{\xi}}^{(t)}||_2 < tol.$, stop the iteration for the update. Otherwise, repeat Step 3 until the convergence.

Step 4. Decrease λ_1 gradually and obtain the updated $\boldsymbol{\xi}$s at each λ_1. Among the candidates, final estimate, $\hat{\boldsymbol{\xi}}$ is determined by minimizing the model selection criterion.

The detailed algorithm of $JSKNR_{NR}$ is described as follows.

Algorithm 6 Algorithm of $JSKNR_{NR}$

Input: Z, X, $\boldsymbol{y}$, $\boldsymbol{\xi}^{(0)}$ and $\Omega^{(0)}(\boldsymbol{\xi})$

Output: $\hat{\boldsymbol{\xi}}$ and $\widehat{\Omega}(\hat{\boldsymbol{\xi}})$

for *iter* $\leftarrow 1$ **to** *maximum iterations* **do**

REML for $\psi = (\tau, \rho, \sigma_\epsilon^2, \beta)$;

$\xi_j^{(0)} = \xi_j(\hat{\rho}), \quad j = 1, ..., p$;

$t \leftarrow 0$;

while $||\hat{\boldsymbol{\xi}}^{(t)} - \hat{\boldsymbol{\xi}}^{(t-1)}||_2 < tol_1$. **do**

for $j \leftarrow 1$ **to** p **do**

$$\begin{aligned} \hat{\xi}_j^{(t+1)} &= \underset{\xi_j}{\operatorname{argmin}} \Big[-\log f_{\boldsymbol{y}|\boldsymbol{Z}}(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{\xi}_D, \hat{\boldsymbol{\xi}}_{OD}) + \lambda_1^{(t+1)} ||\boldsymbol{\xi}_D||_1 \Big] \\ &\Leftrightarrow \quad \hat{\xi}_j^{(t+1)} = \tilde{\xi}_j - \big[L^{(2)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j) \big]^{-1} L^{(1)}(\tilde{\boldsymbol{\xi}}_{-j}, \tilde{\xi}_j); \end{aligned}$$

end

$t \leftarrow t + 1$;

end

Select non-zero components among the diagonals, $\hat{\boldsymbol{\xi}}^{(iter)} \in \mathbb{R}^p$ and

define $\boldsymbol{\xi}^{*(iter)} \in \mathbb{R}^{\hat{p}}$ as the selected components, $\hat{p} \leq p$;

Save the predictors corresponding to the selected components, $\boldsymbol{\xi}^{*(iter)}$ in $Z^* \in \mathbb{R}^{n \times \hat{p}}$;

Given $\boldsymbol{\xi}^{*(iter)}$ and Z^*, update the off-diagonals, $\hat{\xi}_{jj'}^{(iter)}$;

if $||\widehat{\Omega}^{(iter)} - \widehat{\Omega}^{(iter-1)}||_1 < tol.$ *and* $\hat{p}^{(iter)} = \hat{p}^{(iter-1)}$ **then**

Stop;

end

$Z \leftarrow Z^*$ and $p \leftarrow \hat{p}^{(iter)}$;

$iter \leftarrow iter + 1$;

end

B.2 Iterative COSSO

Here we summarize the iterative COSSO procedure.

Algorithm 7 Algorithm for iterative COSSO

1: Initialize $\beta^{(0)}$ and $\boldsymbol{h}^{(0)}(\boldsymbol{z})$.
2: $t = 1$
3: **while** $||\widehat{\psi}^{(t)} - \widehat{\psi}^{(t-1)}||_2 < tol.$ **do**

1. Given $\boldsymbol{h}^{(t-1)}$, estimate $\hat{\beta}^{(t)}$ such that $\hat{\beta}^{(t)} = (X^T X)^{-1} X^T (\boldsymbol{y} - \hat{\boldsymbol{h}}^{(t-1)})$;
2. Solve for h by solving the Cosso problem of minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\{y_i - x_i\hat{\beta}^{(t)} - h(z_i)\}^2 + \lambda_n J(h) \quad \text{with} \quad J(h) = \sum_{\alpha=1}^{A^*} ||P^\alpha h||,$$

   where $P^\alpha h$ the orthogonal projection of h onto $\mathcal{F}^\alpha$;
3. Calculate RSS such that $RSS = \frac{\sum_{i=1}^{n}(y_i - x_i\hat{\beta} - \hat{h}_i)^2}{n} = \hat{\sigma}^{2(t)}$ and let $\psi^{(t)} = (\hat{\beta}^{(t)}, \sigma^{2(t)})$;
4. $\hat{\beta} \leftarrow \hat{\beta}^{(t)}$ and $\hat{\sigma}^2 \leftarrow \hat{\sigma}^{2(t)}$; $t \leftarrow t + 1$;

end

4: Obtain the final estimates, $\hat{\beta}$ and $\hat{\boldsymbol{h}}(\boldsymbol{z})$.

B.3 Tables & Figures for Chapter 3

Table B.1: Simulation results of three methods when the true kernel is polynomial and the precision matrix has an identity precision matrix based on 200 simulation runs; three methods $JSKNR2$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; linear polynomial kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	True / Fitted	(n,p)	ACC	FPR	FNR	TPR	RSS	MSE
$JSKNR2$	Polynomial / Polynomial	(64,20)	0.885(0.065)	0.135(0.106)	0.096(0.096)	0.904(0.096)	1.047(0.293)	0.373(0.207)
		(30,20)	0.853(0.075)	0.173(0.123)	0.121(0.103)	0.879(0.103)	1.030(0.490)	0.614(0.313)
		(30,40)	0.908(0.042)	0.075(0.044)	0.144(0.112)	0.856(0.112)	0.940(0.449)	0.716(0.356)
		(30,70)	0.951(0.025)	0.024(0.020)	0.196(0.170)	0.804(0.170)	1.959(3.128)	1.483(2.663)
		(30,120)	0.969(0.013)	0.015(0.012)	0.207(0.152)	0.793(0.152)	2.043(3.076)	1.498(2.407)
NGK_s		(64,20)	0.884(0.072)	0.126(0.112)	0.105(0.102)	0.895(0.102)	1.196(0.775)	0.482(0.609)
		(30,20)	0.836(0.079)	0.185(0.122)	0.144(0.118)	0.856(0.118)	1.136(0.564)	0.724(0.445)
		(30,40)	0.905(0.044)	0.075(0.047)	0.155(0.120)	0.845(0.120)	1.866(7.503)	1.431(6.107)
		(30,70)	0.948(0.026)	0.033(0.024)	0.167(0.126)	0.833(0.126)	2.591(6.275)	2.022(5.314)
		(30,120)	0.969(0.014)	0.015(0.013)	0.208(0.136)	0.792(0.136)	10.32(16.667)	8.974(14.99)
$iCOSSO$		(64,20)	0.577(0.061)	0.020(0.062)	0.827(0.152)	0.173(0.152)	1.068(1.387)	1.713(1.339)
		(30,20)	NA	NA	NA	NA	NA	NA
		(30,40)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA
		(30,120)	NA	NA	NA	NA	NA	NA

Table B.2: Simulation results of three methods when the true kernel is linear polynomial kernel and the precision matrix has a AR(1) structure with $\rho^* = 0.25$ based on 200 simulation runs; three methods $JSKNR2$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; linear polynomial kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	True / Fitted	(n,p)	ACC	FPR	FNR	TPR	RSS	MSE
$JSKNR2$	Polynomial / Polynomial	(64,20)	0.874(0.073)	0.182(0.116)	0.071(0.087)	0.929(0.087)	0.974(0.256)	0.325(0.142)
		(30,20)	0.828(0.078)	0.239(0.142)	0.104(0.098)	0.896(0.098)	0.969(0.551)	0.626(0.332)
		(30,40)	0.911(0.038)	0.069(0.040)	0.150(0.108)	0.850(0.108)	1.007(0.606)	0.744(0.450)
		(30,70)	0.943(0.025)	0.042(0.025)	0.146(0.116)	0.854(0.116)	0.958(0.715)	0.764(0.431)
		(30,120)	0.961(0.015)	0.025(0.016)	0.191(0.154)	0.809(0.154)	1.708(2.991)	1.420(2.309)
NGK_s		(64,20)	0.867(0.076)	0.186(0.126)	0.080(0.091)	0.920(0.091)	1.000(0.276)	0.339(0.151)
		(30,20)	0.822(0.078)	0.259(0.154)	0.097(0.096)	0.903(0.096)	1.023(0.674)	0.654(0.442)
		(30,40)	0.906(0.041)	0.073(0.045)	0.158(0.113)	0.842(0.113)	1.064(0.517)	0.761(0.440)
		(30,70)	0.935(0.025)	0.052(0.027)	0.143(0.123)	0.857(0.123)	1.234(2.953)	1.048(2.617)
		(30,120)	0.956(0.018)	0.031(0.022)	0.184(0.160)	0.816(0.160)	7.187(14.241)	6.271(12.44)
$iCOSSO$		(64,20)	0.585(0.092)	0.019(0.041)	0.810(0.171)	0.190(0.149)	0.918(1.209)	2.355(6.889)
		(30,20)	NA	NA	NA	NA	NA	NA
		(30,40)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA
		(30,120)	NA	NA	NA	NA	NA	NA

Table B.3: Simulation results of three methods when the true kernel is linear polynomial kernel and the precision matrix has AR(1) structure with $\rho^* = 0.50$ based on 200 simulation runs: three methods $JSKNR2$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; linear polynomial kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Polynomial / Polynomial	(64,20)	0.877(0.070)	0.174(0.112)	0.073(0.089)	0.927(0.089)	1.001(0.347)	0.345(0.231)
		(30,20)	0.836(0.083)	0.171(0.132)	0.158(0.112)	0.842(0.112)	1.534(0.585)	1.004(0.234)
		(30,40)	0.901(0.036)	0.027(0.033)	0.316(0.145)	0.684(0.145)	5.722(7.181)	4.334(5.905)
		(30,70)	0.946(0.024)	0.016(0.017)	0.284(0.140)	0.716(0.140)	3.998(3.951)	2.820(2.977)
		(30,120)	0.964(0.165)	0.013(0.013)	0.284(0.158)	0.716(0.194)	4.132(5.957)	3.065(4.843)
NGK_s		(64,20)	0.868(0.069)	0.190(0.115)	0.073(0.087)	0.927(0.087)	1.005(0.344)	0.348(0.227)
		(30,20)	0.808(0.076)	0.091(0.115)	0.292(0.153)	0.708(0.153)	5.223(8.240)	3.842(6.740)
		(30,40)	0.902(0.037)	0.029(0.037)	0.303(0.141)	0.697(0.141)	7.542(11.697)	5.873(10.001)
		(30,70)	0.943(0.026)	0.020(0.022)	0.282(0.143)	0.718(0.143)	5.667(8.909)	4.220(7.445)
		(30,120)	0.961(0.018)	0.019(0.018)	0.260(0.181)	0.740(0.181)	10.509(19.161)	8.921(16.783)
$iCOSSO$		(64,20)	0.604(0.087)	0.032(0.078)	0.760(0.229)	0.240(0.229)	0.953(2.147)	1.633(2.238)
		(30,20)	NA	NA	NA	NA	NA	NA
		(30,40)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA
		(30,120)	NA	NA	NA	NA	NA	NA

Table B.4: Simulation results of three methods when the true kernel is polynomial and the precision matrix has a AR(2) structure with $\rho^* = 0.25$ based on 200 simulation runs; three methods $JSKNR2$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; linear polynomial kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Polynomial / Polynomial	(30,20)	0.820(0.077)	0.277(0.136)	0.084(0.087)	0.916(0.087)	0.883(0.504)	0.597(0.296)
		(30,70)	0.933(0.027)	0.056(0.027)	0.134(0.119)	0.866(0.119)	0.815(0.892)	0.805(0.523)
NGK_s		(30,20)	0.801(0.086)	0.308(0.175)	0.089(0.099)	0.911(0.099)	0.959(0.634)	0.634(0.400)
		(30,70)	0.833(0.108)	0.146(0.098)	0.290(0.197)	0.710(0.197)	0.908(0.363)	0.922(0.287)
$iCOSSO$		(30,20)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA

Table B.5: Simulation results of three methods when true kernel is linear polynomial kernel and precision matrix has a AR(2) structure with $\rho^* = 0.50$ based on 200 simulation runs; three methods $JSKNR2$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; linear polynomial kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Polynomial / Polynomial	(30,20)	0.818(0.081)	0.270(0.135)	0.094(0.093)	0.906(0.093)	0.901(0.503)	0.600(0.297)
		(30,70)	0.935(0.027)	0.051(0.026)	0.149(0.123)	0.851(0.123)	1.274(2.720)	1.131(2.054)
NGK_s		(30,20)	0.797(0.085)	0.31(0.168)	0.095(0.099)	0.905(0.099)	0.978(0.605)	0.636(0.343)
		(30,70)	0.859(0.121)	0.118(0.109)	0.278(0.217)	0.722(0.217)	0.936(0.452)	0.924(0.325)
$iCOSSO$		(30,20)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA

Table B.6: Simulation results of the four methods when true kernel is Gaussian and precision matrix has a diagonal structure based on 200 simulation runs; four methods $JSKNR2$, $JRKNR_{NK}$, NGK_s, and $iCOSSO$ are compared in terms of six measures; Gaussian kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Gaussian / Gaussian	(64,20)	0.984(0.053)	0.007(0.026)	0.025(0.102)	0.975(0.102)	1.098(1.008)	0.994(0.943)
		(30,20)	0.954(0.050)	0.012(0.038)	0.080(0.089)	0.920(0.089)	0.990(0.093)	0.878(0.233)
		(30,40)	0.972(0.036)	0.011(0.034)	0.079(0.079)	0.921(0.079)	0.993(0.087)	0.913(0.245)
		(30,70)	0.974(0.039)	0.014(0.032)	0.100(0.115)	0.900(0.115)	0.991(0.073)	0.866(0.220)
		(30,120)	0.940(0.065)	0.040(0.052)	0.289(0.254)	0.711(0.254)	0.917(0.208)	0.911(0.242)
JSKNR_{NR}		(64,20)	0.849(0.132)	0.290(0.270)	0.012(0.050)	0.988(0.050)	0.981(0.044)	0.881(0.186)
		(30,20)	0.925(0.074)	0.073(0.111)	0.077(0.115)	0.923(0.115)	1.135(0.629)	0.879(0.241)
		(30,40)	0.923(0.083)	0.049(0.083)	0.163(0.131)	0.837(0.131)	0.991(0.470)	0.932(0.252)
		(30,70)	0.908(0.092)	0.065(0.083)	0.255(0.190)	0.745(0.190)	0.887(0.168)	0.873(0.228)
		(30,120)	0.906(0.067)	0.067(0.056)	0.395(0.230)	0.605(0.230)	0.785(0.198)	0.892(0.247)
NGK_s		(64,20)	0.898(0.150)	0.067(0.188)	0.138(0.243)	0.862(0.243)	2.885(3.309)	2.443(2.718)
		(30,20)	0.758(0.115)	0.231(0.264)	0.253(0.304)	0.747(0.304)	4.544(3.883)	3.254(2.959)
		(30,40)	0.880(0.141)	0.112(0.152)	0.144(0.149)	0.856(0.149)	1.047(0.860)	0.985(0.677)
		(30,70)	0.861(0.098)	0.119(0.092)	0.259(0.195)	0.741(0.195)	1.062(0.751)	1.009(0.526)
		(30,120)	0.851(0.048)	0.123(0.039)	0.433(0.190)	0.567(0.190)	0.805(0.424)	1.008(0.331)
$iCOSSO$		(64,20)	0.568(0.036)	0.300(0.074)	0.834(0.111)	0.166(0.111)	1.306(0.851)	1.984(0.748)
		(30,20)	NA	NA	NA	NA	NA	NA
		(30,40)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA
		(30,120)	NA	NA	NA	NA	NA	NA

Table B.7: Simulation results of four methods when true kernel is Gaussian and the precision matrix has a AR(1) structure with $\rho^* = 0.50$ based on 200 simulation runs; four methods $JSKNR2$, $JSKNR_{NR}$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Gaussian / Gaussian	(64,20)	0.919(0.136)	0.018(0.065)	0.143(0.273)	0.857(0.273)	1.732(1.754)	1.334(1.267)
		(30,20)	0.949(0.103)	0.005(0.034)	0.097(0.206)	0.903(0.206)	1.293(0.873)	0.961(0.537)
		(30,40)	0.968(0.042)	0.034(0.054)	0.028(0.085)	0.972(0.085)	1.081(0.489)	0.932(0.491)
		(30,70)	0.974(0.034)	0.020(0.031)	0.062(0.147)	0.938(0.147)	1.022(0.260)	0.861(0.274)
		(30,120)	0.974(0.02)	0.015(0.018)	0.146(0.208)	0.854(0.208)	1.226(0.809)	0.945(0.454)
JSKNR_{NR}		(64,20)	0.872(0.125)	0.222(0.246)	0.034(0.132)	0.966(0.132)	1.151(0.993)	0.981(0.776)
		(30,20)	0.923(0.090)	0.060(0.122)	0.094(0.167)	0.906(0.167)	1.050(0.409)	0.884(0.273)
		(30,40)	0.965(0.049)	0.016(0.035)	0.093(0.158)	0.907(0.158)	1.055(0.419)	0.906(0.278)
		(30,70)	0.960(0.100)	0.009(0.038)	0.222(0.261)	0.778(0.269)	1.216(0.897)	0.944(0.650)
		(30,120)	0.976(0.028)	0.011(0.021)	0.177(0.209)	0.823(0.209)	1.132(0.892)	0.942(0.511)
NGK_s		(64,20)	0.803(0.188)	0.049(0.124)	0.345(0.325)	0.655(0.325)	4.334(3.743)	3.628(3.058)
		(30,20)	0.892(0.120)	0.161(0.205)	0.056(0.161)	0.944(0.161)	1.437(1.493)	1.123(0.948)
		(30,40)	0.803(0.197)	0.156(0.194)	0.321(0.259)	0.679(0.259)	1.418(1.505)	1.227(1.113)
		(30,70)	0.802(0.128)	0.156(0.111)	0.448(0.266)	0.552(0.266)	1.173(0.821)	1.054(0.524)
		(30,120)	0.857(0.070)	0.107(0.057)	0.542(0.229)	0.458(0.229)	0.977(0.738)	1.095(0.495)
$iCOSSO$		(64,20)	0.599(0.085)	0.022(0.061)	0.779(0.208)	0.221(0.208)	0.883(0.683)	1.511(0.619)
		(30,20)	NA	NA	NA	NA	NA	NA
		(30,40)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA
		(30,120)	NA	NA	NA	NA	NA	NA

Table B.8: Simulation results of four methods when the true kernel is Gaussian and the precision matrix has a AR(2) structure with $\rho^* = 0.25$ based on 200 simulation runs; four methods $JSKNR2$, $JSKNR_{NR}$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Gaussian / Gaussian	(30,20)	0.908(0.087)	0.143(0.181)	0.041(0.074)	0.959(0.074)	1.041(0.246)	0.881(0.236)
		(30,70)	0.960(0.053)	0.022(0.040)	0.153(0.164)	0.847(0.164)	1.128(0.760)	0.920(0.707)
JSKNR_{NR}		(30,20)	0.928(0.073)	0.06(0.113)	0.085(0.098)	0.915(0.098)	1.042(0.364)	0.902(0.25)
		(30,70)	0.943(0.076)	0.038(0.064)	0.173(0.194)	0.827(0.194)	1.008(0.466)	0.877(0.219)
NGK_s		(30,20)	0.918(0.110)	0.066(0.170)	0.098(0.139)	0.902(0.139)	1.110(0.669)	0.972(0.653)
		(30,70)	0.833(0.108)	0.146(0.098)	0.290(0.197)	0.710(0.197)	0.908(0.363)	0.922(0.287)
$iCOSSO$		(30,20)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA

Table B.9: Simulation results of four methods when the true kernel is Gaussian and precision matrix has a AR(2) structure with $\rho^* = 0.50$ based on 200 simulation runs; four methods $JSKNR2$, $JSKNR_{NR}$, NGK_s, and $iCOSSO$ are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model; NA=no results due to numerical instability of iCOSSO

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
JSKNR2	Gaussian / Gaussian	(30,20)	0.902(0.106)	0.135(0.192)	0.061(0.152)	0.939(0.152)	1.131(0.748)	0.953(0.507)
		(30,70)	0.966(0.047)	0.019(0.037)	0.127(0.155)	0.873(0.155)	1.121(0.503)	0.899(0.256)
JSKNR_{NR}		(30,20)	0.927(0.088)	0.060(0.122)	0.085(0.147)	0.915(0.147)	1.238(0.937)	0.975(0.729)
		(30,70)	0.962(0.062)	0.022(0.050)	0.138(0.184)	0.862(0.184)	1.132(0.773)	0.922(0.417)
NGK_s		(30,20)	0.899(0.139)	0.090(0.202)	0.113(0.143)	0.887(0.143)	1.273(1.125)	1.046(0.839)
		(30,70)	0.859(0.121)	0.118(0.109)	0.278(0.217)	0.722(0.217)	0.936(0.452)	0.924(0.325)
$iCOSSO$		(30,20)	NA	NA	NA	NA	NA	NA
		(30,70)	NA	NA	NA	NA	NA	NA

Table B.10: Simulation results of three methods when true kernel is linear polynomial and precision matrix has a AR(1) with $\rho^* = 0.50$ based on 200 simulation runs; three methods $JSKNR2$, JSKNR_{NR}, and NGK_s are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model

Method	**True / Fitted**	(n,p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Polynomial / Gaussian	(30,20)	0.821(0.111)	0.002(0.013)	0.357(0.221)	0.643(0.221)	34.095(33.980)	25.888(28.030)
		(30,40)	0.889(0.055)	0.001(0.010)	0.271(0.218)	0.729(0.218)	28.322(34.131)	25.575(30.986)
		(30,70)	0.976(0.030)	0.002(0.017)	0.156(0.147)	0.844(0.147)	8.336(21.974)	5.270(14.293)
		(30,120)	0.963(0.020)	0.001(0.008)	0.430(0.226)	0.570(0.226)	26.880(25.416)	18.504(19.784)
JSKNR_{NR}		(30,20)	0.842(0.111)	0.002(0.023)	0.315(0.221)	0.685(0.221)	27.930(39.105)	23.039(34.131)
		(30,40)	0.915(0.060)	0.004(0.034)	0.327(0.218)	0.673(0.218)	24.165(34.715)	19.301(29.840)
		(30,70)	0.959(0.035)	0.004(0.017)	0.267(0.195)	0.733(0.195)	6.101(21.188)	4.941(18.833)
		(30,120)	0.973(0.024)	0.002(0.013)	0.297(0.232)	0.703(0.232)	21.185(34.723)	17.279(29.156)
NGK_s		(30,20)	0.778(0.174)	0.159(0.265)	0.286(0.257)	0.714(0.257)	27.171(44.760)	23.885(41.346)
		(30,40)	0.753(0.192)	0.210(0.214)	0.358(0.240)	0.642(0.240)	22.511(41.768)	19.871(37.569)
		(30,70)	0.908(0.058)	0.070(0.071)	0.225(0.264)	0.775(0.264)	28.719(58.866)	26.469(56.667)
		(30,120)	0.871(0.061)	0.095(0.053)	0.502(0.191)	0.498(0.191)	22.251(33.841)	17.812(28.279)

Table B.11: Simulation results of three methods when true kernel is linear polynomial and precision matrix has AR(2) structure with $\rho^* = 0.25$ based on 200 simulation runs; three methods $JSKNR2$, JSKNR$_{NR}$, and NGK_s are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model

Method	**True / Fitted**	(n, p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Polynomial / Gaussian	(30,20)	0.921(0.075)	0.004(0.022)	0.155(0.150)	0.845(0.150)	21.958(32.959)	17.397(27.534)
		(30,70)	0.984(0.017)	0.002(0.006)	0.104(0.114)	0.896(0.114)	6.895(18.785)	5.307(14.864)
JSKNR$_{NR}$		(30,20)	0.892(0.083)	0.023(0.081)	0.192(0.152)	0.808(0.152)	19.917(32.125)	16.614(28.220)
		(30,70)	0.961(0.033)	0.011(0.031)	0.201(0.151)	0.799(0.151)	12.557(27.009)	10.517(23.625)
NGK_s		(30,20)	0.786(0.146)	0.228(0.278)	0.201(0.138)	0.799(0.138)	22.739(33.353)	19.158(29.366)
		(30,70)	0.811(0.071)	0.184(0.076)	0.220(0.120)	0.780(0.120)	23.028(18.103)	22.587(15.792)

Table B.12: Simulation results of three methods when true kernel is linear polynomial and precision matrix has AR(2) structure with $\rho^* = 0.50$ based on 200 simulation runs; three methods $JSKNR2$, JSKNR$_{NR}$, and NGK_s are compared in terms of six evaluation measures; Gaussian kernel is used for fitting a model

Method	**True / Fitted**	(n, p)	**ACC**	**FPR**	**FNR**	**TPR**	**RSS**	**MSE**
$JSKNR2$	Polynomial / Gaussian	(30,20)	0.887(0.093)	0.001(0.008)	0.225(0.186)	0.775(0.186)	25.919(33.808)	20.782(28.588)
		(30,70)	0.979(0.021)	0.001(0.008)	0.144(0.136)	0.856(0.136)	13.562(28.223)	10.422(22.704)
JSKNR$_{NR}$		(30,20)	0.869(0.094)	0.020(0.082)	0.242(0.170)	0.758(0.170)	21.990(33.193)	18.096(28.715)
		(30,70)	0.962(0.029)	0.005(0.018)	0.239(0.164)	0.761(0.164)	16.044(28.764)	13.132(25.099)
NGK_s		(30,20)	0.794(0.154)	0.154(0.251)	0.258(0.164)	0.742(0.164)	26.064(34.187)	21.728(30.022)
		(30,70)	0.814(0.082)	0.170(0.079)	0.283(0.145)	0.717(0.145)	23.517(18.660)	22.666(16.003)

Table B.13: Summary table of the average bias (s.e.) of two methods when three precision matrices are considered based on 200 simulation runs; two methods $JSKNR2$ and Glasso are compared in terms of three matrix norms; three precision matrices are diagonal ($\rho^* = 0$), AR(1) with $\rho^* = 0.25$, and AR(1) with $\rho^* = 0.5$ structures; the estimated precision matrix is obtained from fitting a model with polynomial kernel

Kernel	(n, p)	ρ^*		$\widehat{\Omega}_{JSKNR2}$	$\widehat{\Omega}_{Glasso}$
Polynomial	(30, 40)	0.00	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.463 (1.352)	8.427 (3.427)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	1.015 (0.680)	1.212 (0.655)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	0.635 (4.369)	59.398 (20.523)
		0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	2.464 (2.775)	27.298 (9.809)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	2.169 (1.550)	5.227 (2.226)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	2.639 (8.657)	176.999 (62.983)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.213 (2.438)	30.223 (11.788)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.830 (1.726)	5.812 (2.832)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	7.666 (6.949)	203.058 (79.020)
	(30, 70)	0.00	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.474 (0.577)	3.791 (1.761)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.194 (0.250)	0.097 (0.229)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	4.904 (2.126)	47.320 (14.520)
		0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	5.108 (4.393)	38.003 (13.969)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	3.867 (2.641)	5.010 (2.224)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	11.834 (14.260)	333.004 (117.892)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.296 (4.077)	23.946 (23.513)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.979 (3.494)	2.457 (3.783)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	8.217 (11.198)	219.551 (202.203)
	(30, 120)	0.00	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.487 (0.623)	5.892 (2.790)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.207 (0.276)	0.121 (0.271)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	4.930 (2.230)	88.335 (30.298)
		0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.437 (2.541)	26.815 (11.559)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	1.135 (1.240)	1.717 (1.235)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	0.175 (8.404)	309.525 (124.223)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.204 (2.381)	18.998 (16.430)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.174 (1.514)	0.433 (1.754)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	10.670 (7.517)	228.917 (178.609)

Table B.14: Summary table of the average bias (s.e.) of two methods when three precision matrices are considered based on 200 simulation runs; two methods $JSKNR2$ and Glasso are compared in terms of three matrix norms; three precision matrices are diagonal ($\rho^* = 0$), AR(2) with $\rho^* = 0.25$, and AR(2) with $\rho^* = 0.5$ structures; the estimated precision matrix is obtained from fitting a model with polynomial kernel

Kernel	(n,p)	ρ^*		$\widehat{\Omega}_{JSKNR}$	$\widehat{\Omega}_{Glasso}$
Polynomial	(30,20)	0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	4.956 (2.834)	13.675 (1.578)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	3.245 (1.302)	4.312 (0.622)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	10.536 (9.762)	54.001 (6.756)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	4.519 (2.844)	12.662 (3.102)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	2.735 (1.354)	3.340 (1.130)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	4.976 (9.743)	47.362 (13.330)
	(30,70)	0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	5.058 (3.790)	35.817 (11.631)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	3.014 (1.745)	4.419 (1.806)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	10.66 (12.902)	308.255 (96.216)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	3.435 (4.328)	30.483 (16.490)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	1.828 (2.270)	2.820 (2.557)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	0.525 (14.516)	266.569 (138.844)

Table B.15: Summary table of the average bias (s.e.) of two methods when three precision matrices are considered based on 200 simulation runs; two methods $JSKNR2$ and Glasso are compared in terms of three matrix norms; two precision matrices are AR(2) with $\rho^* = 0.25$, and AR(2) with $\rho^* = 0.5$ structures; the estimated precision matrix is obtained from fitting a model with Gaussian kernel

Kernel	(n,p)	ρ^*		$\widehat{\Omega}_{JSKNR}$	$\widehat{\Omega}_{Glasso}$
Gaussian	(30,20)	0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.483 (1.695)	5.416 (2.896)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	1.007 (0.936)	1.730 (0.992)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	4.329 (5.570)	21.329 (11.679)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.187 (1.743)	5.378 (2.766)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.735 (0.918	1.566 (0.955)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	2.909 (6.073)	20.785 (11.542)
	(30,70)	0.25	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	1.160 (1.750)	13.724 (7.327)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.767 (0.889)	1.766 (1.096)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	2.976 (5.575)	114.480 (59.563)
		0.50	$\|\|\widehat{\Omega}\|\|_F - \|\|\Omega_{true}\|\|_F$	0.791 (1.654)	13.165 (7.625)
			$\|\|\widehat{\Omega}\|\|_{l_1,m} - \|\|\Omega_{true}\|\|_{l_1,m}$	0.541 (0.881)	1.497 (1.146)
			$\|\|\hat{\Omega}\|\|_{l_1,e} - \|\|\Omega_{true}\|\|_{l_1,e}$	1.227 (5.576)	110.91 (62.870)

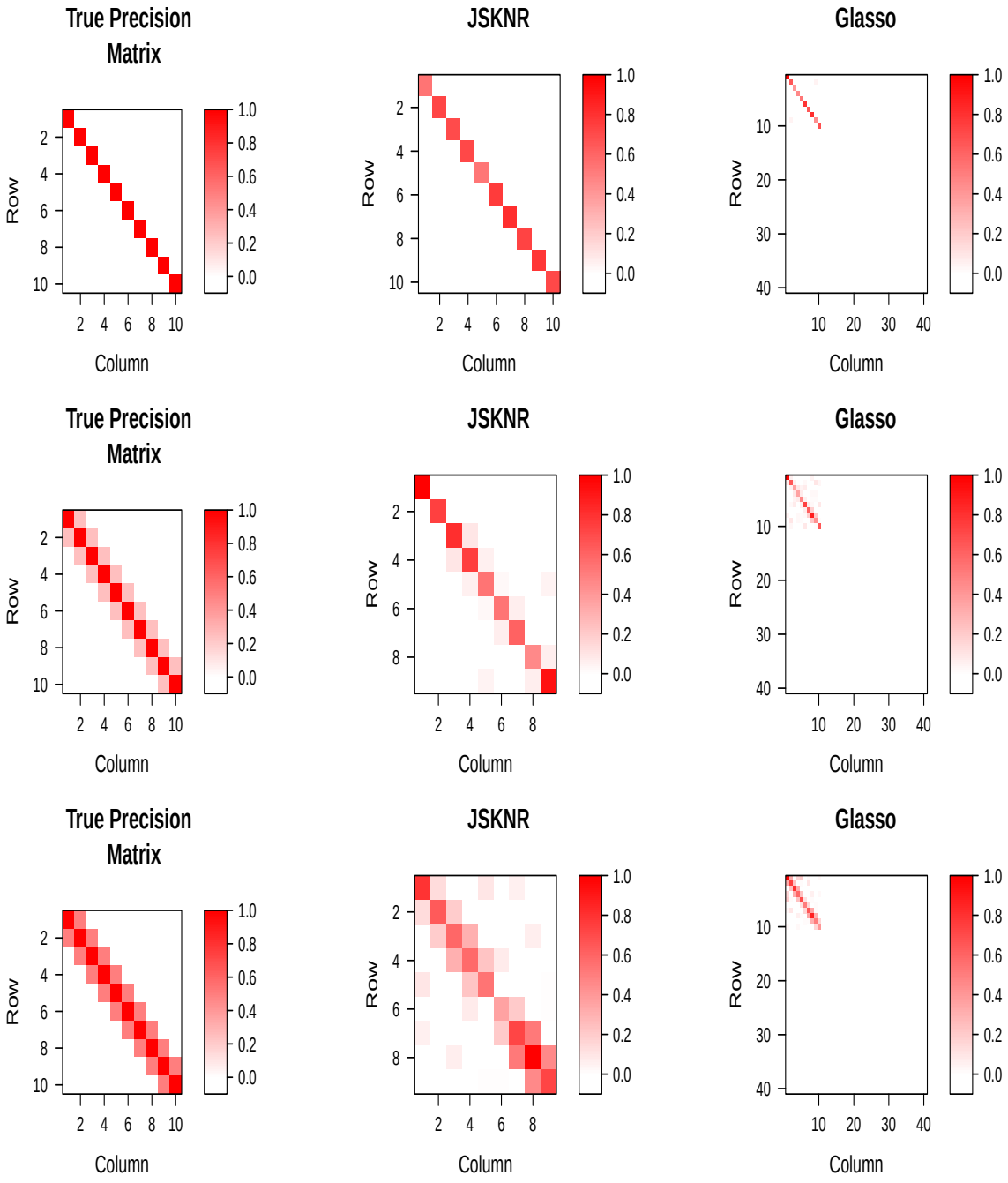


Figure B.1: Heatmaps of the estimated precision matrices based on $JSKNR$ and Glasso when the precision matrix has a AR(1) structure and linear polynomial kernel is used when $(n, p) = (30, 40)$; The figures represent the estimated precision matrices of AR(1) structure with $\rho^* = 0.00$ (top three), 0.25 (middle three), and 0.50 (bottom three), respectively.

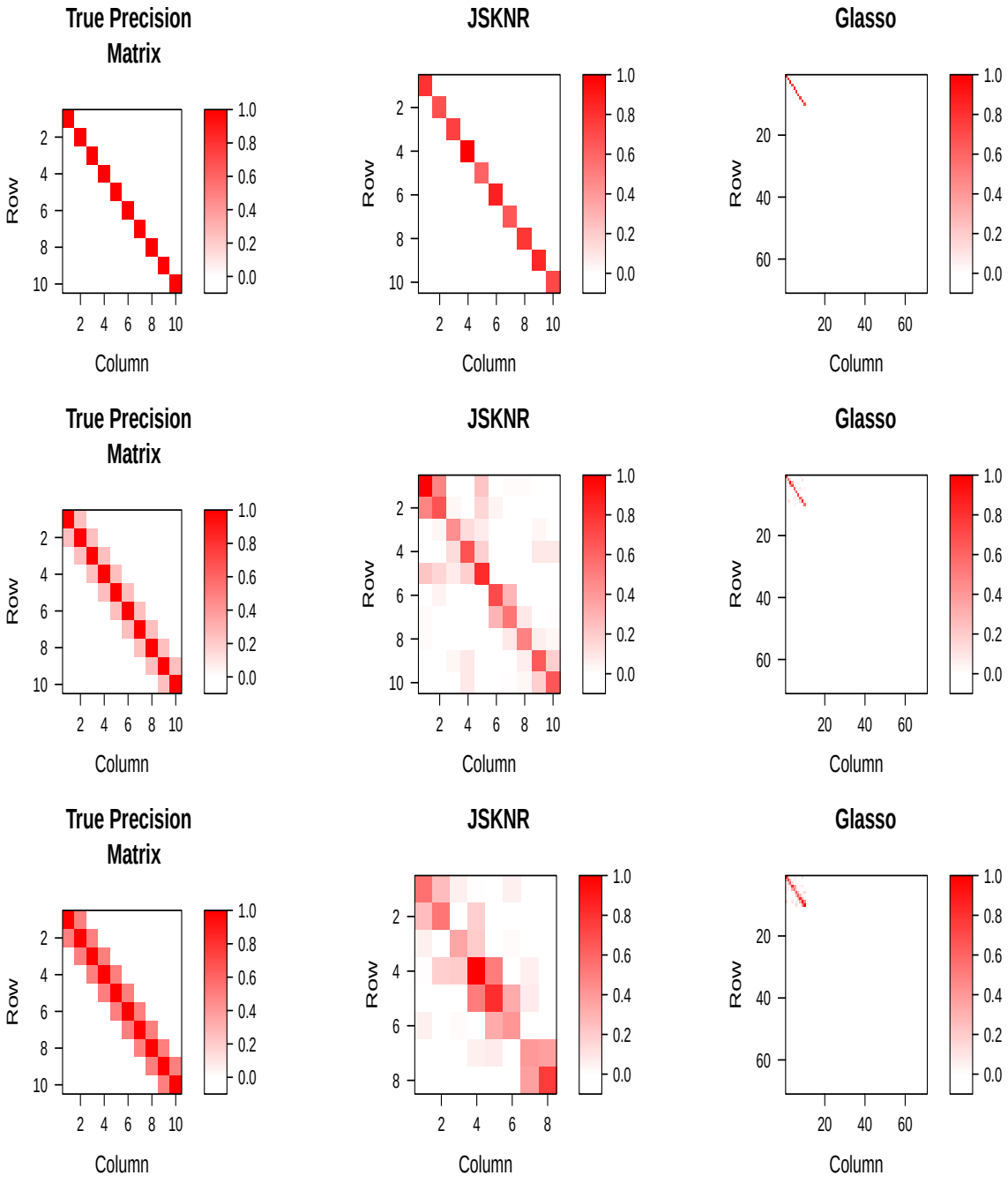


Figure B.2: Heatmaps of the estimated precision matrices based on $JSKNR$ and Glasso when the precision matrix has a AR(1) structure and linear polynomial kernel is used when $(n, p) = (30, 70)$; The figures represent the estimated precision matrices of AR(1) structure with $\rho^* = 0.00$ (top three), 0.25 (middle three), and 0.50 (bottom three), respectively.

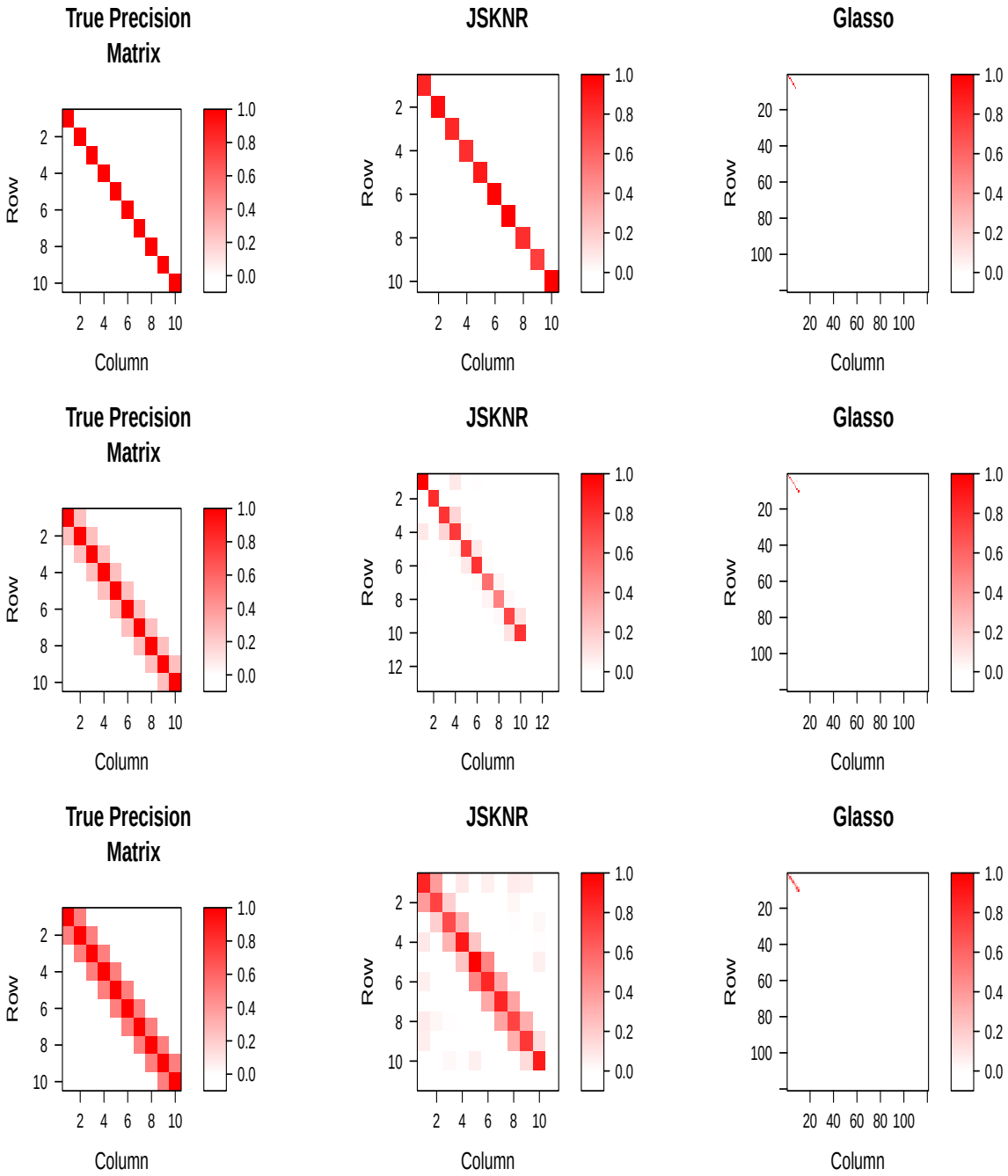


Figure B.3: Heatmaps of the estimated precision matrices based on $JSKNR$ and Glasso when the precision matrix has a AR(1) structure and linear polynomial kernel is used when $(n, p) = (30, 120)$; The figures represent the estimated precision matrices of AR(1) structure with $\rho^* = 0.00$ (top three), 0.25 (middle three), and 0.50 (bottom three), respectively.

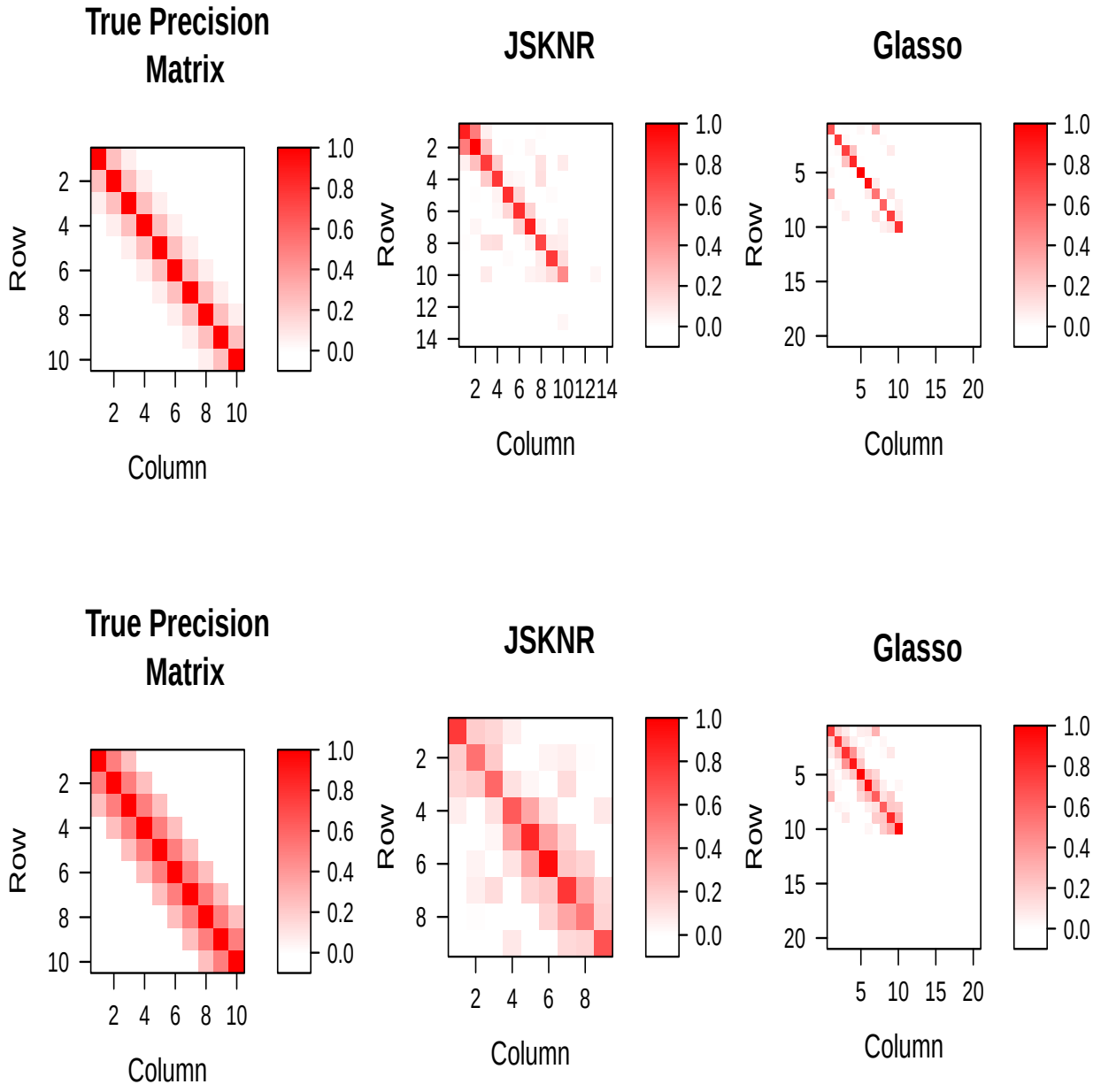


Figure B.4: Heatmaps of the estimated precision matrices based on $JSKNR$ and Glasso when the precision matrix has a AR(2) structure and linear polynomial kernel is used with $(n, p) = (30, 20)$; The figures represent the estimated precision matrices of AR(2) structure with $\rho^* = 0.25$ (top three) and 0.50 (bottom three), respectively.

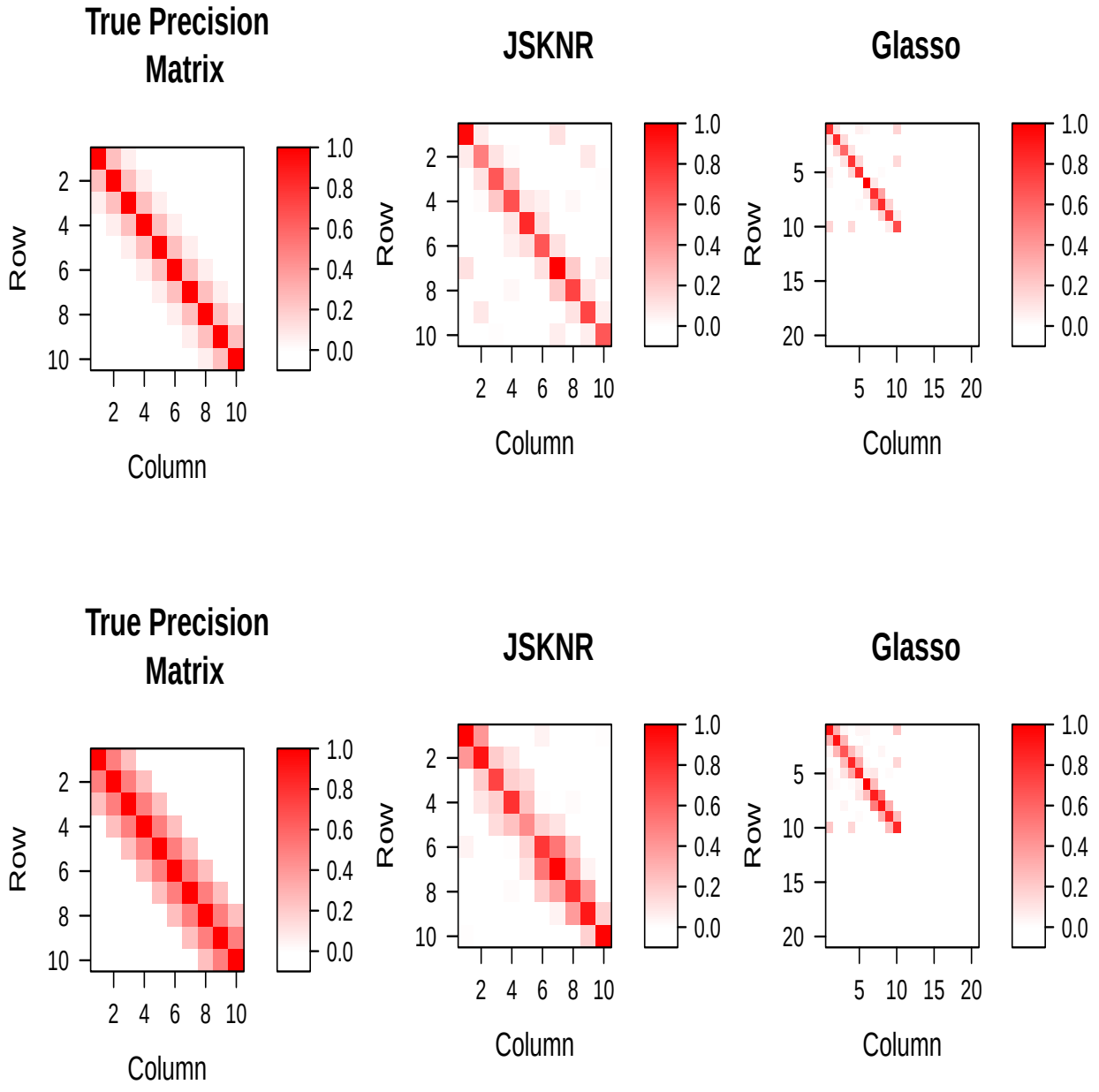


Figure B.5: Heatmaps of the estimated precision matrices based on $JSKNR$ and Glasso when the precision matrix has a AR(2) structure and Gaussian kernel is used with $(n, p) = (30, 20)$; The figures represent the estimated precision matrices of AR(2) structure with $\rho^* = 0.25$ (top three) and 0.50 (bottom three), respectively.

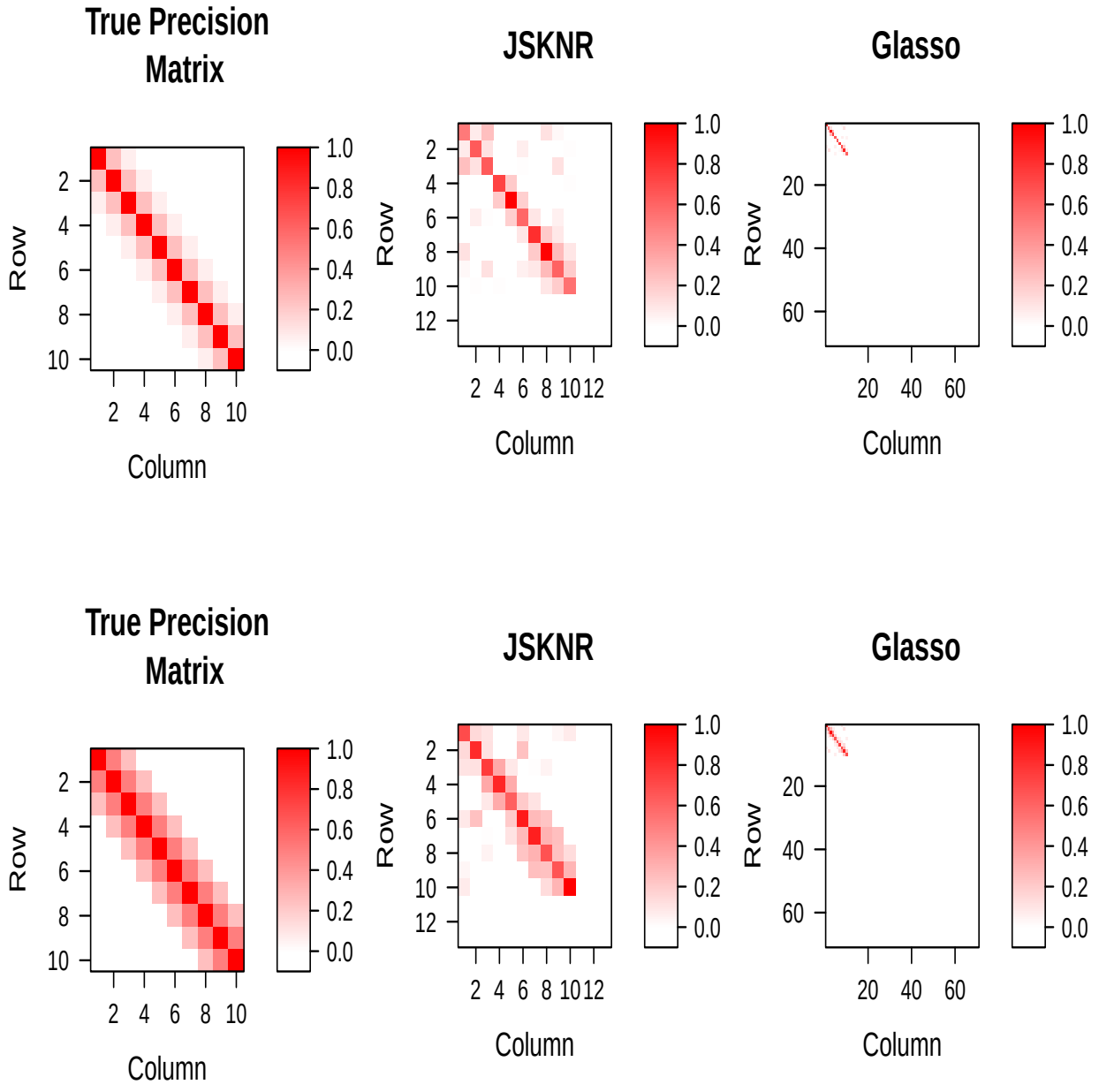


Figure B.6: Heatmaps of the estimated precision matrices based on $JSKNR$ and Glasso when the precision matrix has a AR(2) structure and linear polynomial kernel is used with $(n, p) = (30, 70)$; The figures represent the estimated precision matrices of AR(2) structure with $\rho^* = 0.25$ (top three) and 0.50 (bottom three), respectively.

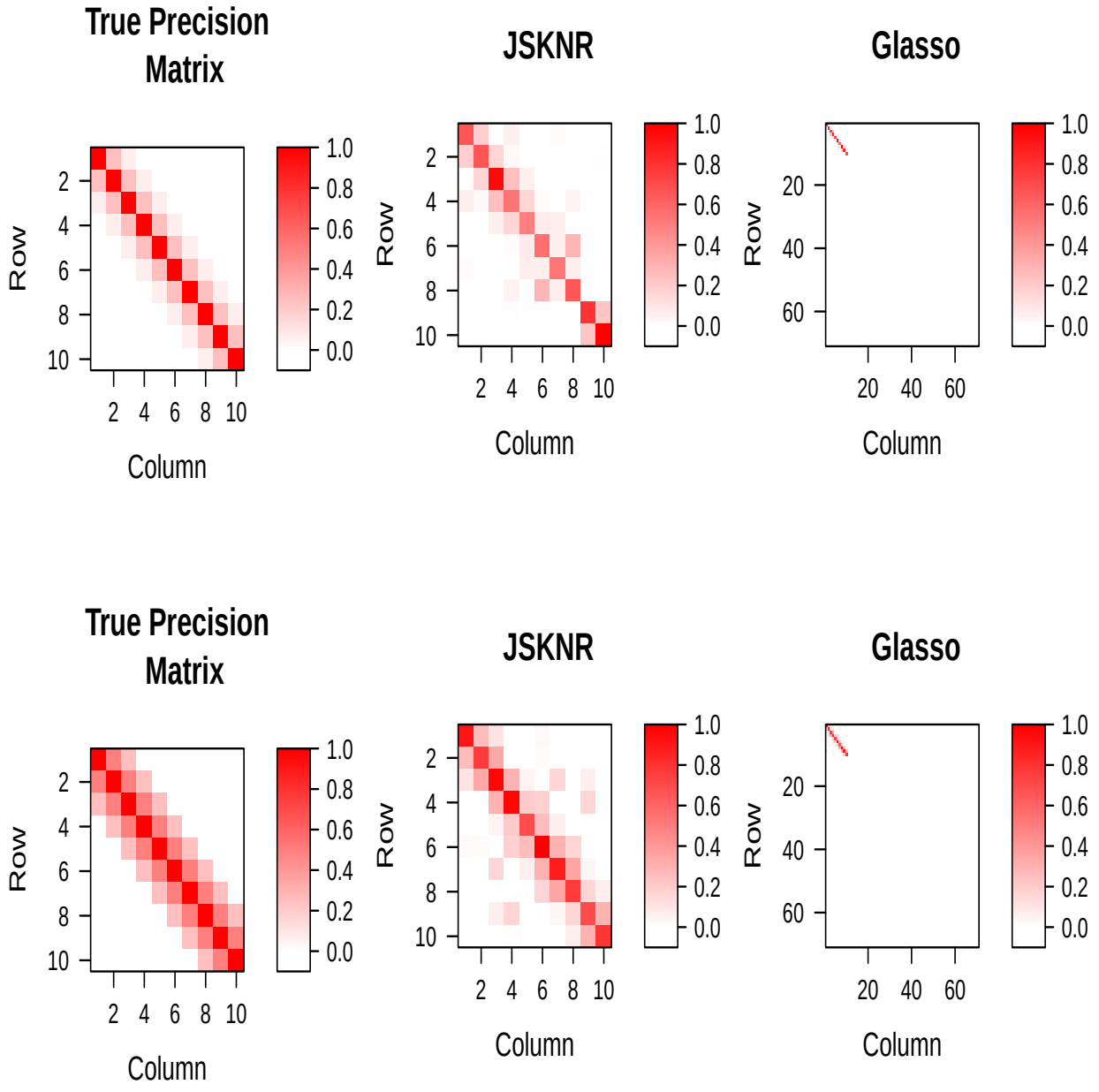


Figure B.7: Heatmaps of the estimated precision matrices based on $JSKNR$ and Glasso in Case2, AR(2) structure using Gaussian kernel with $(n, p) = (30, 70)$; The figures represent the estimated precision matrices of AR(2) structure with $\rho^* = 0.25$ (top) and 0.50 (bottom), respectively.

Appendix C

Appendices for Chapter 4

C.1 Derivation of Full Conditional Distributions

We derive the posterior distributions of the parameters based on the prior specification discussed in 4.4.1. The full conditional distributions for each parameter can be obtained by using the likelihood function from data and the conjugate priors. The full conditionals have closed forms because of the conjugate priors and facilitate the estimation of the parameters by drawing the posterior samples through Gibbs sampler, which is one of the simple and efficient sampling methods among Markov Chain Monte Carlo (MCMC) algorithms.

Based on the prior settings of the hierarchical model corresponding with the reformulation of the fused lasso problem, the full conditional distributions of the hierarchical model can

be derived. First, the conditional distribution of $\boldsymbol{U}$ as

$$\begin{aligned}\pi(\boldsymbol{U}|-) &\propto \pi(\boldsymbol{U}|\sigma_\epsilon^2)\pi(\boldsymbol{y}|-)\\ &\propto \exp\left(-\frac{1}{2\sigma_\epsilon^2}\boldsymbol{U}^T\boldsymbol{D}_\tau^{-1}\boldsymbol{U}\right)\times\exp\left\{-\frac{1}{2\sigma_\epsilon^2}\sum_{j=1}^{M}\left(\boldsymbol{Y}_{\bullet,j}-X_U U-\boldsymbol{\mu}_{\bullet,j(-U)}\right)^T\left(\boldsymbol{Y}_{\bullet,j}-X_U U-\boldsymbol{\mu}_{\bullet,j(-U)}\right)\right\}\\ &\propto \exp\left\{\boldsymbol{U}^T\left(\frac{1}{\sigma_\epsilon^2}\boldsymbol{D}_\tau^{-1}+\frac{1}{\sigma_\epsilon^2}X_U^TX_U\right)\boldsymbol{U}-\frac{2}{\sigma_\epsilon^2}\sum_{j=1}^{M}\left(\boldsymbol{Y}_{\bullet,j}-\boldsymbol{\mu}_{\bullet,j(-U)}\right)^T X_U\boldsymbol{U}\right\},\end{aligned}$$

where "$-$" denotes all other parameters conditioned in a full conditional distribution and $A_{\bullet,j}$ is the jth column of a matrix A. $\boldsymbol{\mu}_{(-a)}$ means the overall average, excluding a parameter a. Hence, the full conditional distribution of $\boldsymbol{U}$ follows $\mathcal{N}_{n-1}(\boldsymbol{\mu}_U,\Sigma_U)$, where

$$\begin{aligned}\boldsymbol{\mu}_U &= \left(\boldsymbol{D}_\tau^{-1}+X_U^TX_U\right)^{-1}\sum_{j=1}^{M}X_U^T\left(\boldsymbol{Y}_{\bullet,j}-\boldsymbol{\mu}_{\bullet,j(-U)}\right),\\ \Sigma_U &= \sigma_\epsilon^2\left(\boldsymbol{D}_\tau^{-1}+X_U^TX_U\right)^{-1}.\end{aligned}$$

Similarly, the full conditional distribution of $\boldsymbol{V}_{i,\bullet}$ is $\mathcal{N}_{M-1}\left(\boldsymbol{\mu}_{V_{i,\bullet}},\Sigma_V\right)$, where

$$\begin{aligned}\boldsymbol{\mu}_{V_{i,\bullet}} &= \left(\frac{\boldsymbol{D}_\omega^{-1}}{\sigma_b^2}+\frac{X_V^TX_V}{\sigma_\epsilon^2}\right)^{-1}\sum_{l=1}^{n}X_V^T\left(\boldsymbol{Y}_{l,\bullet}-\boldsymbol{\mu}_{l,\bullet(-V)}\right),\quad i=1,...,n-1,\\ \Sigma_{V_{i,\bullet}} &= \left(\frac{\boldsymbol{D}_\omega^{-1}}{\sigma_b^2}+\frac{X_V^TX_V}{\sigma_\epsilon^2}\right)^{-1}.\end{aligned}$$

The full conditional distributions for τ_i^2 and ω_j^2 are derived as

$$\begin{aligned}\pi(\tau_i^2|-) &\propto \pi(\tau_i^2|\lambda_1)\pi(\boldsymbol{U}|\tau_i^2,\sigma_\epsilon^2)\\ &\propto \exp\left(-\frac{\lambda_1^2}{2}\tau_i^2\right)\times(\tau_i^2)^{-1/2}\exp\left\{-\frac{1}{2\sigma_\epsilon^2}\left(\boldsymbol{U}^T\boldsymbol{D}_\tau^{-1}\boldsymbol{U}\right)_{i,i}\right\}\\ &\propto (\tau_i^2)^{-1/2}\exp\left\{-\tau_i^2\frac{\lambda_1^2}{2}-\frac{1}{2\sigma_\epsilon^2\tau_i^2}U_i^2\right\},\end{aligned}$$

and

$$\begin{aligned}\pi(\omega_i^2|-) &\propto \pi(\omega_i^2|\lambda_2)\pi(\boldsymbol{V}_{i,\bullet}|\omega_i^2,\sigma_b^2)\\ &\propto (\omega_i^2)^{M/2-1}\exp\left(-\frac{\lambda_2^2}{2}\omega_i^2\right)\times(\omega_i^2)^{-(M-1)/2}\exp\left\{-\frac{1}{2\sigma_b^2}\left(\boldsymbol{V}_{i,\bullet}^T\boldsymbol{D}_{\omega_i}^{-1}\boldsymbol{V}_{i,\bullet}\right)\right\}\\ &\propto (\omega_i^2)^{-1/2}\exp\left\{-\omega_i^2\frac{\lambda_2^2}{2}-\frac{1}{2\sigma_b^2\omega_i^2}||V_{i,\bullet}||^2\right\}.\end{aligned}$$

Then, the full conditional distributions for both τ_i^{-2} and ω_i^{-2} follow inverse Gaussian distributions as below:

$$[\tau_i^{-2}|-]\sim \text{Inv.Gaussian}\left(\sqrt{\frac{\lambda_1^2\sigma_\epsilon^2}{U_i^2}},\lambda_1^2\right)$$

and

$$[\omega_i^{-2}|-]\sim \text{Inv.Gaussian}\left(\sqrt{\frac{\lambda_2^2\sigma_b^2}{||V_{i,\bullet}||^2}},\lambda_2^2\right).$$

The full conditional distributions for σ_ϵ^2 and σ_b^2 are derived as follows:

$$\begin{aligned}\pi(\sigma_\epsilon^2|-) &\propto \pi(\sigma_\epsilon^2)\pi(\boldsymbol{U}|\sigma_\epsilon^2)\pi(\boldsymbol{y}|-)\\ &\propto \sigma_\epsilon^{2(-a_{\sigma_\epsilon^2}-1)}\exp\left(-\frac{b_{\sigma_\epsilon^2}}{\sigma_\epsilon^2}\right)\times\sigma_\epsilon^{2(-\frac{n-1}{2})}\exp\left(-\frac{1}{2\sigma_\epsilon^2}U^T\boldsymbol{D}_\tau^{-1}U\right)\\ &\quad\times\sigma_\epsilon^{2(-\frac{nM}{2})}\exp\left\{-\frac{1}{2\sigma_\epsilon^2}\sum_{j=1}^{M}\left(\boldsymbol{Y}_{\bullet,j}-X_UU-\boldsymbol{\mu}_{\bullet,j(-U)}\right)^T\left(\boldsymbol{Y}_{\bullet,j}-X_UU-\boldsymbol{\mu}_{\bullet,j(-U)}\right)\right\}\\ &\propto\sigma_\epsilon^{2\left(-\frac{nM}{2}-\frac{n-1}{2}-a_{\sigma_\epsilon^2}-1\right)}\exp\left\{-\frac{1}{\sigma_\epsilon^2}\left(\frac{1}{2}\sum_{j=1}^{M}\left(\boldsymbol{Y}_{\bullet,j}-X_UU-\boldsymbol{\mu}_{\bullet,j(-U)}\right)^T\left(\boldsymbol{Y}_{\bullet,j}-X_UU-\boldsymbol{\mu}_{\bullet,j(-U)}\right)+\frac{1}{2}U^T\boldsymbol{D}\right.\right.\end{aligned}$$

Hence, the full conditional distribution of σ_ϵ^2 follows the inverse-gamma distribution with a

scale parameter $\nu_{\sigma_\epsilon^2}$ and a shape parameter $\eta_{\sigma_\epsilon^2}$, where

$$\begin{aligned}
\nu_{\sigma_\epsilon^2} &= \frac{nM+n-1}{2} + a_{\sigma_\epsilon^2}, \\
\eta_{\sigma_\epsilon^2} &= \left(\frac{1}{2} \sum_{j=1}^{M} \left(\boldsymbol{Y}_{\bullet,j} - X_U U - \boldsymbol{\mu}_{\bullet,j(-U)} \right)^T \left(\boldsymbol{Y}_{\bullet,j} - X_U U - \boldsymbol{\mu}_{\bullet,j(-U)} \right) + \frac{1}{2} U^T \boldsymbol{D}_\tau^{-1} U + b_{\sigma_\epsilon^2} \right).
\end{aligned}$$

Similarly, we can obtain the full conditional distribution of σ_b^2 as follows:

$$[\sigma_b^2|-] \sim \text{Inv.Gamma}\left(\nu_{\sigma_b^2}, \eta_{\sigma_b^2} \right),$$

where

$$\begin{aligned}
\nu_{\sigma_b^2} &= \frac{M-1}{2} + a_{\sigma_b^2}, \\
\eta_{\sigma_b^2} &= \frac{1}{2} \sum_{i=1}^{n} \left(\boldsymbol{V}_{i,\bullet}^T \boldsymbol{D}_{\omega_i}^{-1} \boldsymbol{V}_{i,\bullet} \right) + b_{\sigma_b^2}.
\end{aligned}$$

For the full conditionals of the tuning parameters λ_1 and λ_2, we get Gamma distributions based on the conjugacy as follows: The full conditional distribution of λ_1 is derived as follows:

$$\begin{aligned}
\pi(\lambda_1^2|-) &\propto \pi(\lambda_1^2) \prod_{i=1}^{n-1} \pi(\tau_i^2|\lambda_1) \\
&\propto \lambda_1^{2(\rho_1-1)} \exp(-\delta_1 \lambda_1^2) \times \left(\frac{\lambda_1^2}{2} \right)^{n-1} \exp\left(-\frac{\lambda_1^2}{2} \sum_{i=1}^{n-1} \tau_i^2 \right) \\
&\propto \lambda_1^{2(n+\rho_1-2)} \exp\left\{ -\lambda_1^2 \left(\frac{1}{2} \sum_{i=1}^{n-1} \tau_i^2 + \delta_1 \right) \right\}.
\end{aligned}$$

Thus,

$$[\lambda_1^2|-] \sim \text{Gamma}\left(n + \rho_1 - 1, \frac{1}{2} \sum_{i=1}^{n-1} \tau_i^2 + \delta_1 \right).$$

Similarly, we have the full conditional distribution of λ_2 as follows:

$$[\lambda_2^2|-] \sim \text{Gamma}\left(\frac{n(M-1)}{2} + \rho_2, \frac{1}{2}\sum_{i=1}^{n}\omega_i^2 + \delta_2\right).$$