FLEXIBLE SUPERVISED LEARNING TECHNIQUES WITH APPLICATIONS IN NEUROSCIENCE

Guan Yu

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistics and Operations Research.

Chapel Hill
2016

Approved by:

Yufeng Liu

Shankar Bhamidi

Steve Marron

Dinggang Shen

Kai Zhang

©2016
Guan Yu
ALL RIGHTS RESERVED

ABSTRACT

Guan Yu: Flexible Supervised Learning Techniques
with Applications in Neuroscience
(Under the direction of Yufeng Liu)

Supervised learning techniques have been widely used in diverse scientific disciplines such as biology and neuroscience. Among the existing supervised learning techniques, penalized regression is a very popular one, partly due to its simple formulation and good performance in practice. Despite the success of this technique, many challenges remain. The first challenge is how to develop new methods that could incorporate the structure/correlation information among predictors efficiently. Moreover, in many practical applications such as computational neuroscience, we need to predict multiple correlated responses (e.g., class label and clinical scores). It is very important to study new techniques to predict those correlated responses jointly, using not only the correlation information among responses but also the structure/correlation information among predictors. Furthermore, in modern scientific research, many data sets are collected from different modalities (sources or types). Since the observations of a certain modality can be missing completely, block-missing multi-modality data are very common. Flexible and efficient statistical methods applicable to block-missing multi-modality data require careful study. In this dissertation, we propose several new supervised learning techniques to overcome the challenges mentioned above. Both numerical and theoretical studies are presented to demonstrate the effectiveness of our proposed methods. Practical applications of these methods using the Alzheimer's Disease Neuroimaging Initiative (ADNI) data set are provided as well.

ACKNOWLEDGEMENTS

This dissertation would not have been completed without the great support of people who stood by me during my years at UNC. I would like to thank all of them.

Firstly, I would like to express my appreciation and deep gratitude to my advisor Professor Yufeng Liu for being a great advisor and a wonderful person to work with. I have learned many good things from him both on the academic and personal font. He has been there to share ideas and at the same time encourage me to pursue my own. He has been extremely enthusiastic and supportive throughout my PhD years. It is a great pleasure to have him as my advisor. I would also like to thank Professor Dinggang Shen for providing me the opportunity to study in the IDEA lab and giving me many valuable suggestions on my projects. I would like to convey my sincere thanks to my dissertation committee members: Professor Steve Marron, Professor Shankar Bhamidi, and Professor Kai Zhang for their time, support, guidance, and great suggestions on my dissertation. Last but not least, I am very grateful to my friends, my parents and my fiancee for their unending encouragement and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND SYMBOLS

AD	Alzheimer's Disease
MCI	Mild cognitive impairment
NC	Normal control
ADNI	Alzheimer's Disease Neuroimaging Initiative
MRI	Structural magnetic resonance imaging
PET	Fluorodeoxyglucose positron emission tomography
CSF	Cerebrospinal fluid
MMSE	Mini mental state examination score
ADAS-Cog	Alzheimer's disease assessment scale-cognitive subscale score
SRIG	Sparse regression incorporating graphical structure among predictors
$\mathbf{\Sigma}$	Population covariance matrix
$\mathbf{\Omega}$	Population precision matrix
$\mathbf{G}$	Undirected predictor graph
$\mathcal{N}_i$	The neighborhood of predictor i
$\|A\|_2$	The ℓ_2 norm of the vector A
$\|\mathbf{A}\|_\infty$	$\max_{1\leq i\leq k}\sum_{j=1}^{m}\|A_{ij}\|$ if $\mathbf{A}$ is a $k\times m$ matrix
$\|\mathbf{A}\|_F$	$\sqrt{\sum_{i=1}^{k}\sum_{j=1}^{m}A_{ij}^2}$ if $\mathbf{A}$ is a $k\times m$ matrix

CHAPTER 1: INTRODUCTION

1.1 Background

Supervised learning techniques play an important role in statistics. Among the existing supervised learning techniques, penalized regression is a very popular one, partly due to its simple formulation and good performance in practice. The basic idea of penalized regression is to perform penalized least squares incorporating some additional constraints on the regression coefficients. In this section, we first briefly review some fundamental penalized regression techniques. In Section 1.1.1, some popular penalized univariate linear regression methods in the literature are reviewed. In Section 1.1.2, we discuss the extension of penalized regression methods from univariate regression to multivariate regression. In Section 1.1.3, we discuss how to use an undirected graph to represent the structure information among predictors.

1.1.1 Penalized Linear Regression

Linear regression is a typical supervised learning task and it is commonly used in practice. The model is

$$Y = \mathbf{X}\beta^0 + \epsilon, \tag{1.1}$$

where $\mathbf{X} \in \mathbb{R}^{n\times p}$ is the predictor (design) matrix, $Y \in \mathbb{R}^n$ is the response vector, n is the number of observations, p is the number of predictors, $\beta^0 = (\beta_1^0, \beta_2^0, \ldots, \beta_p^0)^T$ is a vector of unknown coefficients, and ϵ is a vector of independently and identically distributed (i.i.d.) random variables with mean 0 and finite variance σ^2.

Under the standard setting with the sample size n larger than the dimension p, the commonly used ordinary least squares (OLS) estimator for the p-dimensional regression coefficient vector β^0 often works well. On the other hand, it is also well known that OLS

often leads to complicate models with low prediction accuracy when the predictors are highly correlated. Furthermore, for the high dimensional data ($p \gg n$), OLS is not applicable due to the rank deficiency of the design matrix. In order to improve OLS, many penalized methods using regularization in model fitting have been proposed in the literature. The general form of penalized regression is shown as follows:

$$\hat{\beta} = \arg\min_{\beta} \|Y - \mathbf{X}\beta\|_2^2 + \lambda P(\beta),$$

where λ is a tuning parameter and $P(\beta)$ is a penalty term that can be used to incorporate all kinds of constraints on the regression coefficients.

Different choices of the penalty term $P(\beta)$ lead to different penalized regression methods. For example, classical ridge regression ((Hoerl and Kennard, 1970)) uses the ridge penalty $\sum_{i=1}^{p} |\beta_i^0|^2$ to possibly achieve better prediction performance through a bias-variance trade-off. The popular Lasso method ((Tibshirani, 1996)) uses the l_1 penalty $\sum_{i=1}^{p} |\beta_i^0|$ to perform continuous shrinkage and automatic variable selection simultaneously. It is known from the literature that Lasso has many good theoretical properties such as model selection consistency ((Zhao and Yu, 2006)), estimation consistency ((Knight and Fu, 2000)), and persistence property ((Greenshtein, 2006)). However, Lasso also has some limitations. For example, the shrinkage introduced by Lasso results in significant bias towards 0 for large regression coefficients ((Fan and Li, 2001)). In the presence of some highly correlated variables, Lasso tends to select only one of those variables ((Zou and Hastie, 2005)).

Besides the Lasso method, a lot of other penalized regression methods have been proposed for simultaneous variable selection and estimation. Some methods are very useful to reduce the bias of estimation. For example, (Fan and Li, 2001) introduced the smoothly clipped absolute deviation (SCAD) method using a non-convex penalty. (Zou, 2006) proposed the adaptive Lasso estimator where adaptive weights are used to penalize different coefficients. (Zhang, 2010) studied the minimax concave penalty (MCP) which is a nearly unbiased method for penalized variable selection. In addition, there are also some methods proposed to encourage the strongly correlated predictors to be in or out of the model together. For example, (Zou and Hastie, 2005) proposed the Elastic net method which uses a

convex combination of the l_1 and ridge penalty. In the literature, there are also some other important penalized regression methods. For example, (Wang et al., 2007) utilized the least absolute deviation Lasso for robust regression. (Witten and Tibshirani, 2009) proposed the Scout method which includes many penalized methods as special cases.

Although the penalized regression methods introduced above are designed for the univariate regression problem, the corresponding regularization ideas are very general and can be also used for multivariate regression. In the next section, we will introduce some penalized regression methods for multivariate regression.

1.1.2 Penalized Multivariate Regression

In Section 1.1.1, we have introduced some penalized linear regression methods. In this section, we focus on penalized multivariate regression, which is also called multi-task learning in machine learning if we use linear models to predict multiple correlated continuous response variables. The multivariate regression model is

$$\mathbf{Y} = \mathbf{XB} + \mathbf{e}, \text{ with } \mathbf{e} = [\mathbf{e}_1, \mathbf{e}_1, \ldots, \mathbf{e}_n]^T, \tag{1.2}$$

where $\mathbf{Y} \in \mathbb{R}^{n \times q}$ is the response matrix, $\mathbf{B} \in \mathbb{R}^{p \times q}$ is the coefficient matrix, and $\mathbf{e}_i = (e_{i1}, e_{i2}, \ldots, e_{iq})^T; i = 1, 2, \ldots, n$, are i.i.d. q-dimensional random vectors following a multivariate distribution with mean $0_{q \times 1}$ and covariance matrix $\mathbf{\Sigma}_Y$.

For multivariate regression, the simplest method is to regress each response variable separately on the same set of predictors. All the univariate regression methods including the above penalized linear regression methods can be applied to each response. However, this method may not be optimal since it does not incorporate the correlation information among different response variables. To build an effective model predicting multiple responses jointly, (Breiman and Friedman, 1997) proposed a method, namely the curd and whey, which predicts multiple responses by some optimal linear combinations of the ordinary least squares predictions. Although the curd and whey method could achieve better prediction performance than the separate univariate regression, it did not address the problem of variable selection.

Besides the curd and whey method, a lot of further developments have been made in the literature. One popular way to capture the relatedness among multiple response variables is to constrain all regression models to share a common set of predictors (i.e., elements in each row of $\mathbf{B}$ are constrained to be zero or nonzero simultaneously). To that end, many existing methods use mixed-norm penalties. Some well known examples of such methods are the l_1/l_2 norm ((Obozinski et al., 2010)) and the l_1/l_∞ norm (Turlach et al., 2005; Zhang et al., 2008). These methods could have good prediction performance and also deliver sparse models for variable selection. The statistical properties of these methods are discussed in (Obozinski et al., 2011b).

Another way to use the correlation information among response variables is to constrain the coefficient matrix $\mathbf{B}$ to have a low-rank structure. However, we can not use the rank function as the penalty term directly to constrain the rank of $\mathbf{B}$ since the corresponding optimization problem is non-deterministic polynomial-time hard (NP-hard). To solve this issue, (Yuan et al., 2007) uses a new penalty based on the trace norm (also called nuclear norm) of the coefficient matrix $\mathbf{B}$. This penalty encourages the sparsity among singular values and therefore reduces the rank of the estimated coefficient matrix. Moreover, the reduced-rank regression methods (Reinsel and Velu, 1998; Chen and Huang, 2012) can be also used to achieve a low-rank estimation of $\mathbf{B}$. Generally, these methods constrain $\text{rank}(\mathbf{B}) = r$ for some $r \leq \min\{p, q\}$. However, as mentioned in (Yuan et al., 2007), since the parameter r is often chosen in a separate hypothesis testing or cross validation step, the reduced-rank regression methods can be unstable. Furthermore, although methods encouraging a low-rank structure of $\mathbf{B}$ incorporate the correlation information among responses, most of them do not address the problem of variable selection. In the literature, besides methods using mixed-norm penalties and methods encouraging a low-rank structure of the coefficient matrix, there are also some methods proposed to estimate the coefficient matrix $\mathbf{B}$ and the covariance (or precision) matrix of $\mathbf{Y}$ jointly. See for example (Rothman et al., 2010), (Sohn and Kim, 2012), and (Lee and Liu, 2012).

1.1.3 Graphical Structure among Predictors

Despite the vast literature on penalized methods shown above for univariate regression or multivariate regression, few methods directly incorporate the structure/correlation information among predictors efficiently, and at the same time perform simultaneous estimation, prediction, and model selection. Typically, the structure/correlation information among predictors can be modeled by the connectivity of an undirected graph. It would be very interesting and useful to study how to use this structure information to improve the performance of variable selection, estimation and prediction.

In general, we can get the structure information of the predictors from prior information or estimation. For example, many biological studies have shown that there may exist some regulatory relationships between genes ((Li and Li, 2008)). An increasing amount of information about gene interaction is organized in databases ((Subramanian et al., 2005)). This biological information can be used to construct the predictor graph where nodes represent genes and edges indicate regulatory relationships. If the prior information is not available in some applications, we can construct the predictor graph by sparse estimation of the covariance (or precision) matrix of the predictors ((Yuan and Lin, 2007; Friedman et al., 2008; Cai et al., 2011)). Then, the estimated significant marginal (or partial) correlational relationships among predictors can be represented by the connectivity of an undirected graph, where nodes represent predictors and edges indicate significant marginal (or partial) correlation. In Chapter 2, we will propose a new sparse regression method that could efficiently use the structure/correlation information among predictors. In Chapter 3, as an extension of the method proposed in Chapter 2, we will propose a new multi-task learning method for joint classification and regression, which is formulated as a multivariate regression problem. As a practical application of our new proposed method, a joint prediction of the class label and clinical scores of the Alzheimer's disease using the Alzheimer's Disease Neuroimaging Initiative (ADNI) data set (www.loni.ucla.edu/ADNI) will be studied in detail.

1.2 New Contributions and Outline

In this dissertation, we investigate some new penalized regression methods for univariate regression and multivariate regression. In addition, we propose a new sparse regression procedure for block-missing multi-modality data. The outline of the dissertation is shown as follows:

- In Chapter 2, we propose a new penalized regression method incorporating the structure/correlation information among predictors directly. Typically, such information can be modeled by the connectivity of an undirected graph using all predictors as nodes of the graph. Our proposed method incorporates this graph information node-by-node by a special latent group Lasso penalty. Theoretical study indicates that our proposed method is very general and it includes adaptive Lasso, group Lasso, and ridge regression as special cases. Furthermore, it acquires tight finite sample bounds for both estimation and prediction, and enjoys model selection consistency for the high dimensional case. Both simulation study and real data analysis demonstrate the effectiveness of the proposed method for simultaneous estimation, prediction and model selection.

- In Chapter 3, we extend the idea of incorporating the structure/correlation information among predictors to a multi-task learning problem. A new multi-task learning method using both the structure/correlation information among predictors and the correlation information among response variables is proposed. Specifically, based on the undirected predictor graph, our new proposed method encourages the correlated predictors to be in or out of the model together. Furthermore, this new method also encourages the correlated response variables to share a common predictor subset. As a practical application of our new proposed method, a joint prediction of class label and clinical scores of the Alzheimer's disease using the ADNI data set will be studied in detail.

- In Chapter 4, we propose a new sparse regression method for block-missing multi-modality data without imputing missing data. Our method includes two steps. In

the first step, we use all available information to estimate the covariance matrix and the cross-covariance matrix. In the second step, based on the estimated covariance matrix and the estimated cross-covariance matrix, we use a modified Lasso estimator to deliver good estimates of the regression coefficients. Both the simulation study and the real data analysis demonstrate the effectiveness of our proposed method. Since our method uses all available information efficiently, it could deliver better performance than many existing methods.

CHAPTER 2: SPARSE REGRESSION INCORPORATING GRAPHICAL STRUCTURE AMONG PREDICTORS

2.1 Introduction

During the last few decades, despite the vast literature on sparse regression, few methods use the structure information of the predictors which can be modeled by the connectivity of an undirected graph. It would be very interesting and useful to study how to use this structure information to improve the performance of variable selection, estimation and prediction. Since the predictor graph can not be represented as some non-overlapping groups, the traditional group Lasso method ((Yuan and Lin, 2006)) cannot make full use of this complicate structure information. To use the entire predictor graph information, most existing methods use the graph edge-by-edge, through adding some penalty terms to encourage coefficients β_i^0 and β_j^0 to be similar for predictors i and j connected by an edge. One type of methods encourages β_i^0 and β_j^0 to be zero or nonzero simultaneously. For example, OSCAR ((Bondell and Reich, 2008)) uses the l_∞ penalty $\max\{|\beta_i^0|, |\beta_j^0|\}$ for every pair of different predictors. (Yang et al., 2012) generalized OSCAR to graph OSCAR (GOSCAR) which only uses the l_∞ penalty for those pairs of predictors connected by an edge in the given predictor graph. (Pan et al., 2010) introduced a weighted L_γ-regularization. (Kim et al., 2013) proposed a new non-convex penalty term based on the truncated lasso penalty.

Another type of methods uses some penalty terms to encourage β_i^0 and β_j^0 have similar values or absolute values. For example, GRACE ((Li and Li, 2008)) uses the penalty $(\beta_i^0/\sqrt{d_i} - \beta_j^0/\sqrt{d_j})^2$ to smooth the weighted β_i^0 over the predictor graph, where d_i is the degree of predictor i. (Zhang et al., 2013) proposed the logistic graph Laplacian net. GFlasso ((Kim and Xing, 2009)) utilizes the penalty $|\beta_i^0 - \text{sign}(\hat{\rho}_{ij})\beta_j^0|$ where $\hat{\rho}_{ij}$ is the sample correlation coefficient between predictors i and j. Other methods of this type include (Yang et al., 2012) and (Zhu et al., 2013) which use some non-convex penalty terms to encourage

$|\beta_i^0|$ and $|\beta_j^0|$ to be similar. Although penalized methods using the predictor graph edge-by-edge are promising in improving regression performance, they also have some drawbacks. On the one hand, these methods do not directly utilize the neighborhood information of the graph. For each neighborhood, it can be preferable to use the corresponding edges jointly rather than separately. On the other hand, the penalty terms in these methods will be more complicate if there are more edges in the graph.

In order to make use of the structure information among predictors, instead of using the predictor graph *edge-by-edge*, we propose a new method, namely Sparse Regression Incorporating Graphical structure among predictors (SRIG), using the graph *node-by-node*. Specifically, according to the predictor graph G, we assume that there is a latent decomposition of β^0 into p parts $V^{(1)}, V^{(2)}, \ldots, V^{(p)}$ such that $\beta^0 = \sum_{i=1}^{p} V^{(i)}$ and each $V^{(i)} \in R^p$. The proposed SRIG imposes a penalty to shrink some $V^{(i)}$ to 0 while the other $V^{(i)}$'s satisfy $supp(V^{(i)}) = \mathcal{N}_i$, where $\mathcal{N}_i$ is a set including predictor i and its neighbors in graph G. For SRIG, if one predictor is important for prediction, the other predictors connected to it are also encouraged to be in the model. Note that our proposed SRIG method is a graph based penalized regression method with a very different motivation, although the corresponding optimization problem can be formulated as a special case of the Latent Group Lasso approach ((Obozinski et al., 2011a)) with each neighborhood $\mathcal{N}_i$ as a group. For computation, besides introducing the predictor duplication method shown in (Obozinski et al., 2011a), we also propose a new iterative proximal algorithm which is very efficient for high dimensional data. Our theoretical study shows that SRIG has close connections with several existing methods: (1) It is the same as the adaptive Lasso method when the predictor graph G has no edge; (2) It is equivalent to the group Lasso method when G consists of multiple complete subgraphs; (3) It has the same nonzero solution set as the ridge regression when G is a complete graph. Under some conditions, SRIG enjoys asymptotic normality, model selection consistency and acquires tight finite sample bounds for both estimation and prediction. In order to evaluate the performance of SRIG, we compare SRIG with many existing methods. Simulation examples with different kinds of predictor graphs are studied. We also analyze a dataset from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (www.loni.ucla.edu/ADNI). The structural magnetic resonance imaging (MRI)

features are used to predict the mini-mental state examination (MMSE) score ((Folstein et al., 1975)). Both the simulation results and the real data application indicate that SRIG has competitive performance in estimation, prediction and model selection.

The rest of the chapter is organized as follows. In Section 2.2, we motivate and introduce our proposed SRIG method. In Section 2.3, we introduce two methods to solve the optimization problem. In Section 2.4, we show some theoretical properties. In Sections 2.5 and 2.6, we demonstrate the use of SRIG on simulated data and the ADNI dataset. We conclude this chapter with some discussion in Section 2.7. Technical proofs are provided in Section 2.8.

2.2 Motivation and Methodology

Consider the following linear regression model:

$$Y = \mathbf{X}\beta^0 + \epsilon, \tag{2.1}$$

where $\epsilon = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^T$ is a vector of i.i.d. random variables with mean 0 and variance σ^2. Here, $\beta^0 = (\beta_1^0, \beta_2^0, \ldots, \beta_p^0)^T$ is a vector of true coefficients, $Y = (y_1, y_2, \ldots, y_n)^T$ is an $n \times 1$ response and $\mathbf{X} = (X_1, X_2, \ldots, X_p) = (x_1, x_2, \ldots, x_n)^T$ is an $n \times p$ design matrix.

For motivation, we first consider the random design setting and assume that each x_k follows some multivariate distribution with mean $0_{p\times 1}$ and covariance matrix $\mathbf{\Sigma}$. The design matrix $\mathbf{X}$ is assumed to be independent of the random error ϵ. Furthermore, denote $\mathbf{\Omega} = (\omega_{ij})_{i,j=1,2,\ldots,p} = \mathbf{\Sigma}^{-1}$ and $\Sigma_{xy} = (c_1, c_2, \ldots, c_p)^T \in R^p$ as the cross-covariance vector between x_k and y_k.

By model (2.1) and the definition of cross-covariance, we have

$$\Sigma_{xy} = E(\mathbf{X}^T Y/n) = E(\mathbf{X}^T\mathbf{X}\beta^0/n) + E(\mathbf{X}^T\epsilon/n) = \mathbf{\Sigma}\beta^0.$$

Then, we observe that $\beta^0 = \mathbf{\Sigma}^{-1}\Sigma_{xy} = \mathbf{\Omega}\Sigma_{xy}$, where $\mathbf{\Omega}$ measures partial correlations among predictors, and Σ_{xy} reflects the marginal correlations between predictors and the response variable. From $\beta^0 = \mathbf{\Omega}\Sigma_{xy}$, we have

$$\beta_1^0 = c_1\omega_{11} + c_2\omega_{12} + \cdots + c_i\omega_{1i} + \cdots + c_p\omega_{1p}$$
$$\beta_2^0 = c_1\omega_{21} + c_2\omega_{22} + \cdots + c_i\omega_{2i} + \cdots + c_p\omega_{2p}$$
$$\vdots$$
$$\beta_p^0 = c_1\omega_{p1} + c_2\omega_{p2} + \cdots + c_i\omega_{pi} + \cdots + c_p\omega_{pp}.$$

As shown in the above equations, β^0 is the sum of p parts, $\{(c_i\omega_{1i}, c_i\omega_{2i}, \ldots, c_i\omega_{pi})^T : 1 \leq i \leq p\}$. For the ith part, $(c_i\omega_{1i}, c_i\omega_{2i}, \ldots, c_i\omega_{pi})^T$, there is a common factor c_i. If the ith predictor and the response variable are uncorrelated marginally, then c_i will be 0 and all the components in the ith part of β^0 will be 0 simultaneously. Furthermore, if c_i is not zero and the predictor graph is defined by Ω, then the support of $(c_i\omega_{1i}, c_i\omega_{2i}, \ldots, c_i\omega_{pi})^T$ becomes $\mathcal{N}_i$, which is a set including predictor i and its neighbors in the predictor graph. Thus, instead of focusing on β^0 in the model, we consider a latent decomposition of β^0 into p parts. After choosing the candidate non-zero components in each part based on $\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_p$, we use the group lasso penalty to encourage the selected components in each part to be zero or nonzero simultaneously.

The above idea can be generalized for an arbitrary predictor graph constructed by the prior information or estimation from data. Given the predictor graph G, we define a $p \times p$ adjacency matrix $\mathbf{E}$, where $E_{ij} = 1$ if predictors i and j are connected and $E_{ij} = 0$ otherwise. For each i, we set $E_{ii} = 1$ and acquire the neighborhood set $\mathcal{N}_i = \{j : E_{ij} = 1\}$. As the previous case, we assume that β^0 can be decomposed into

$$\beta_1^0 = V_1^{(1)}E_{11} + V_1^{(2)}E_{12} + \cdots + V_1^{(i)}E_{1i} + \cdots + V_1^{(p)}E_{1p}$$
$$\beta_2^0 = V_2^{(1)}E_{21} + V_2^{(2)}E_{22} + \cdots + V_2^{(i)}E_{2i} + \cdots + V_2^{(p)}E_{2p}$$
$$\vdots$$
$$\beta_p^0 = V_p^{(1)}E_{p1} + V_p^{(2)}E_{p2} + \cdots + V_p^{(i)}E_{pi} + \cdots + V_p^{(p)}E_{pp}.$$

Here, the ith part is $(V_1^{(i)}E_{1i}, V_2^{(i)}E_{2i}, \ldots, V_p^{(i)}E_{pi})^T$ whose candidate nonzero components are $\{V_j^{(i)}E_{ji} : j \in \mathcal{N}_i\}$. We can view $\{V_j^{(i)} : j \in \mathcal{N}_i\}$ as the effect arising from the marginal correlation between the ith predictor and the response variable. If they are uncorrelated,

$V_j^{(i)}$ will be zero for each $j \in \mathcal{N}_i$ and the components in the set $\{V_j^{(i)} E_{ji} : j \in \mathcal{N}_i\}$ will be zero simultaneously. Therefore, after choosing the candidate non-zero components in each part based on $\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_p$, it is reasonable to use the group lasso penalty to encourage the selected components in each part to be zero or nonzero together. Based on this motivating idea, given the training data $(Y, \mathbf{X})$ and the predictor graph G, we propose a new method, Sparse Regression Incorporating Graphical structure among predictors (SRIG), shown as follows.

SRIG Method

Step 1: Find the neighborhoods $\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_p$ (note that $i \in \mathcal{N}_i$ for each i).

Step 2: Solve the following optimization problem:

$$\min_{\beta, V^{(1)}, \ldots, V^{(p)}} \frac{1}{2n} \|Y - \mathbf{X}\beta\|_2^2 + \lambda \sum_{i=1}^{p} \tau_i \|V^{(i)}\|_2, \tag{2.2}$$

subject to $\sum_{i=1}^{p} V^{(i)} = \beta$ and $\text{supp}(V^{(i)}) \subseteq \mathcal{N}_i$ for each i, where $\text{supp}(V^{(i)})$ is the support of vector $V^{(i)}$ and $\|\cdot\|_2$ is the l_2 norm.

Here, τ_i denotes the positive weight for the i-th group. The choice of τ_i will be discussed in Section 2.4.4.

2.3 Computation

In this section, we introduce two methods to solve the problem (2.2). One is the predictor duplication (PD) method proposed in (Obozinski et al., 2011a) and another one is our proposed iterative proximal (IP) algorithm. The predictor duplication method transforms (2.2) to a traditional group Lasso problem by duplicating predictors while our proposed new algorithm solves problem (2.2) directly without duplicating predictors.

2.3.1 Predictor duplication method

Denote $V^{(i)}_{\mathcal{N}_i}$ as the $|\mathcal{N}_i| \times 1$ sub-vector of $V^{(i)}$ with indices in $\mathcal{N}_i$ and $X_{\mathcal{N}_i}$ as the $n \times |\mathcal{N}_i|$ sub-matrix of $\mathbf{X}$ with column indices in $\mathcal{N}_i$. Denote $\tilde{V} = (V^{(1)^T}_{\mathcal{N}_1}, V^{(2)^T}_{\mathcal{N}_2}, \ldots, V^{(p)^T}_{\mathcal{N}_p})^T$ and $\tilde{\mathbf{X}} = (X_{\mathcal{N}_1}, X_{\mathcal{N}_2}, \ldots, X_{\mathcal{N}_p})$. Then, we can check that $\mathbf{X}\beta = \tilde{\mathbf{X}}\tilde{V}$, and problem (2.2) is equivalent to the following group Lasso problem:

$$\min_{\tilde{V}} \frac{1}{2n}\|Y - \tilde{\mathbf{X}}\tilde{V}\|_2^2 + \lambda \sum_{i=1}^{p} \tau_i \|V^{(i)}_{\mathcal{N}_i}\|_2 \tag{2.3}$$

Many efficient R packages such as **grpreg** ((Breheny and Huang, 2009)) and **gglasso** ((Yang and Zou, 2013)) can be used to solve problem (2.3). After setting $\hat{V}^{(i)}_{\mathcal{N}_i^c} = 0$ for each i, we have $\hat{\beta} = \sum_{i=1}^{p} \hat{V}^{(i)}$. Note that in some cases, some neighborhoods $\{\mathcal{N}_i : i \in F\}$ maybe exactly the same. Then, the vectors $\{V^{(i)}_{\mathcal{N}_i} : i \in F\}$ are indistinguishable and therefore the decomposition of β (i.e., $\{V^{(1)}, V^{(2)}, \ldots, V^{(p)}\}$) is not unique. In this case, although we can not estimate each vector in $\{V^{(i)}_{\mathcal{N}_i} : i \in F\}$ stably, we can estimate $\sum_{i\in F} V^{(i)}_{\mathcal{N}_i}$ directly and stably using the penalty term $(\min_{i\in F} \tau_i)\|\sum_{i\in F} V^{(i)}_{\mathcal{N}_i}\|_2$. Since $\hat{\beta} = \sum_{i=1}^{p} \hat{V}^{(i)}$, different decompositions of β lead to the same estimation of β.

The predictor duplication method shown above is very convenient to use and has good performance in general. However, when the dimensional is high and at the same time the predictor graph is not very sparse, there will be a lot of duplicated predictors in (2.3) and therefore the predictor duplication method can be inefficient ((Obozinski et al., 2011a)). In the following Section 2.3.2, we will propose a new iterative proximal algorithm which does not duplicate predictors. It is stable and very efficient for the high dimensional data, especially when the predictor graph can be decomposed into several disconnected components.

2.3.2 Iterative proximal algorithm

Given the predictor graph G and positive weights τ_i's, for $\beta \in R^p$, define

$$\|\beta\|_{G,\tau} = \min_{\sum_{i=1}^{p} V^{(i)} = \beta,\ \text{supp}(V^{(i)}) \subseteq \mathcal{N}_i} \sum_{i=1}^{p} \tau_i \|V^{(i)}\|_2 \tag{2.4}$$

We can show that $\|\beta\|_{G,\tau}$ is a norm ((Obozinski et al., 2011a)) and (2.2) is equivalent to

$$\min_{\beta \in R^p} \frac{1}{2n}\|Y - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_{G,\tau} \tag{2.5}$$

In problem (2.5), the squared loss function is strictly convex and differentiable. In addition, $\|\beta\|_{G,\tau}$ is a norm and therefore convex. Thus, we can use the Fast Iterative Shrinkage Thresholding Algorithm (FISTA) ((Beck and Teboulle, 2009)) to solve it. For our specific problem (2.5), we propose the following iterative proximal algorithm.

Iterative Proximal (IP) Algorithm

Input: The initial estimate $\beta^{(0)}$ and L= the largest eigenvalue of $\mathbf{X}^T\mathbf{X}/n$.

Step 0: Take $Z^{(1)} = \beta^{(0)} \in R^p$ and $t_1 = 1$.

Step m**:** $(m \geq 1)$ Compute

$$\beta^{(m)} = \arg\min_{\beta} \lambda\|\beta\|_{G,\tau} + \frac{L}{2}\|\beta - (Z^{(m)} - \frac{1}{nL}\mathbf{X}^T(\mathbf{X}Z^{(m)} - Y))\|_2^2, \tag{2.6}$$

$$t_{m+1} = \frac{1 + \sqrt{1 + 4t_m^2}}{2}; \quad Z^{(m+1)} = \beta^{(m)} + \frac{t_m - 1}{t_{m+1}}(\beta^{(m)} - \beta^{(m-1)}).$$

By Theorem 4.4 in (Beck and Teboulle, 2009), the sequences $\{\beta^{(m)}\}$ generated via (2.6) will converge to the optimal solution with rate $O(1/m^2)$. The most time consuming step in the above IP algorithm is to compute the proximal operator of $\lambda\|\beta\|_{G,\tau}$, which is defined as

$$\text{prox}_{\lambda\|\beta\|_{G,\tau}}(h) = \arg\min_{\beta} \lambda\|\beta\|_{G,\tau} + \frac{\|\beta - h\|_2^2}{2}. \tag{2.7}$$

Follow the same proofs of Lemmas 1 and 2 in (Villa et al., 2014), we can show that

$$\text{prox}_{\lambda\|\beta\|_{G,\tau}}(h) = h - \arg\min_{\beta \in \mathcal{S}_{\mathcal{O}}} \|\beta - h\|_2, \tag{2.8}$$

where $\mathcal{S}_{\mathcal{O}} = \{\beta \in R^p : \|\beta_{\mathcal{N}_i}\|_2 \leq \lambda\tau_i \text{ for } i \in \mathcal{O}\}$ and $\mathcal{O} = \{i : \|h_{\mathcal{N}_i}\|_2 > \lambda\tau_i\}$.

In (2.8), we need to solve the following optimization problem

$$u^* = \arg\min_{\beta \in \mathcal{S}_{\mathcal{O}}} \|\beta - h\|_2$$

Based on the number of elements in $\mathcal{O}$, denoted as $M = |\mathcal{O}|$, we use different methods flexibly to find the projection of h onto the convex set $\mathcal{S}_{\mathcal{O}}$ efficiently. If $|\mathcal{O}|$ is small (e.g., smaller than $p/10$ in our simulation study), we calculate the projection by solving the dual problem via the Bertsekas's projected Newton method ((Villa et al., 2014)). The solution is

$$u_j^* = \frac{h_j}{1 + \sum_{i \in \mathcal{O}} t_i^* \mathbf{1}_{i,j}}, \text{ for } j = 1, 2, \ldots, p,$$

where t^* is the solution of

$$\arg\max_{t \in R_+^M} f(t), \text{ with } f(t) = \sum_{j=1}^{p} \frac{-h_j^2}{1 + \sum_{i \in \mathcal{O}} t_i \mathbf{1}_{i,j}} - \sum_{i \in \mathcal{O}} \frac{t_i \lambda^2 \tau_i^2}{L^2},$$

and $\mathbf{1}_{i,j}$ equal to 1 if j belong to $\mathcal{N}_i$ and 0 otherwise. The detailed algorithm to solve the above dual problem is shown in Algorithm 5 in (Villa et al., 2014).

If $|\mathcal{O}|$ is large (e.g., larger than $p/10$), we propose to find the projection by the Parallel Dykstra-like proximal algorithm ((Combettes and Pesquet, 2011)). The detailed algorithm is shown as follows.

Parallel Dykstra-like proximal algorithm

Step 0: Set $u^{(0)} = h$, $z^{1,0} = u^{(0)}, z^{2,0} = u^{(0)}, \ldots, z^{M,0} = u^{(0)}$

Step n: ($n \geq 1$) Compute

$$p^{i,n}_{\mathcal{N}_i^c} = z^{i,n}_{\mathcal{N}_i^c} \text{ for each } i \in \mathcal{O};$$

$$p^{i,n}_{\mathcal{N}_i} = z^{i,n}_{\mathcal{N}_i} \mathbf{1}(\|z^{i,n}_{\mathcal{N}_i}\| \leq \frac{\lambda\tau_i}{L}) + \frac{\lambda\tau_i z^{i,n}_{\mathcal{N}_i}}{L\|z^{i,n}_{\mathcal{N}_i}\|_2} \mathbf{1}(\|z^{i,n}_{\mathcal{N}_i}\| > \frac{\lambda\tau_i}{L}) \text{ for each } i \in \mathcal{O};$$

$$u^{(n+1)} = \sum_{i \in \mathcal{O}} \frac{p^{(i,n)}}{M};$$

$$z^{i,n+1} = u^{(n+1)} + z^{i,n} - p^{i,n} \text{ for each } i \in \mathcal{O}.$$

The sequence $\{u^{(n)}\}$ will converge to the projection of h onto $\mathcal{S}_\mathcal{O}$.

Furthermore, we note that the proposed IP algorithm is scalable to large scale problems when the predictor graph G can be decomposed into several components (i.e., the covariance/precision matrix is block diagonal). Denote the disconnected components in G as $G_1, G_2, \ldots, G_K$ with node sets $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K$, respectively. In this case, we can compute the proximal operator (2.7) efficiently by solving the following K subproblems in parallel:

$$\text{prox}_{\lambda\|\beta_{\mathcal{C}_k}\|_{G_k,\tau_{\mathcal{C}_k}}}(h_{\mathcal{C}_k}) = \arg\min_{\beta_{\mathcal{C}_k}} \lambda\|\beta_{\mathcal{C}_k}\|_{G_k,\tau_{\mathcal{C}_k}} + \frac{\|\beta_{\mathcal{C}_k} - h_{\mathcal{C}_k}\|_2^2}{2},$$

where $\beta_{\mathcal{C}_k}$, $\tau_{\mathcal{C}_k}$, $h_{\mathcal{C}_k}$ are sub-vectors of β, τ, and h, respectively.

The above parallel computation can potentially save a lot of computational cost. In Section 2.5.3, we will compare the computational costs of the PD method with our IP algorithm using several simulated examples. In general, the predictor duplication method is very efficient for small data sets. However, when the dimension is high and the predictor graph G is not very sparse, our proposed IP algorithm is much faster than the predictor duplication method. Furthermore, in some cases, the predictor duplication method may break down since it requires immense working memory.

2.4 Theoretical Properties

In this section, we study the theoretical properties of our proposed SRIG method. For theoretical study, it is convenient to consider (2.5) as the objective function. In (2.5), the optimal decomposition of β minimizing $\|\beta\|_{G,\tau}$ always exists, but may not be unique ((Obozinski et al., 2011a)). Denote $J_0 = \{i : \beta_i^0 \neq 0\}$, $J_0^c = \{i : \beta_i^0 = 0\}$, and $s_0 = |J_0|$ as the true nonzero coefficient set, the true zero coefficient set, and the number of true nonzero coefficients, respectively. For each $\beta \in R^p$, denote $\mathcal{U}(\beta)$ as the set of all optimal decompositions of β, and $K_{G,\tau}(\beta)$ as the number of nonzero $V^{(i)}$'s in the optimal decomposition of β which has the minimal number of nonzero $V^{(i)}$'s, i.e., $K_{G,\tau}(\beta) = \min_{(V^{(1)},V^{(2)},\ldots,V^{(p)})\in\mathcal{U}(\beta)} |\{i : \|V^{(i)}\|_2 \neq 0\}|$. Denote $K_{G,\tau} = \sup_{supp(\beta)\subseteq J_0} K_{G,\tau}(\beta)$. We can check that $K_{G,\tau} = s_0$ if the graph G has no edge, $K_{G,\tau} = K_0$ if G consists of some disconnected complete subgraphs and J_0 is the union of K_0 node sets of those disconnected subgraphs.

2.4.1 Subgradient conditions

The following proposition shows the subgradient conditions for problem (2.5).

Proposition 1. A vector $\beta \in R^p$ is a solution of (2.5) if and only if β can be decomposed as $\beta = \sum_{i=1}^p V^{(i)}$ where $V^{(i)}$'s satisfy that, for all $1 \leq i \leq p$, (a) $V^{(i)}_{\mathcal{N}_i^c} = 0$; (b) either $V^{(i)}_{\mathcal{N}_i} \neq 0$ and $X^T_{\mathcal{N}_i}(Y - \mathbf{X}\beta) = n\lambda\tau_i \frac{V^{(i)}_{\mathcal{N}_i}}{\|V^{(i)}_{\mathcal{N}_i}\|_2}$, or $V^{(i)}_{\mathcal{N}_i} = 0$ and $\|X^T_{\mathcal{N}_i}(Y - \mathbf{X}\beta)\|_2 \leq n\lambda\tau_i$.

The subgradient conditions shown above are similar to the subgradient conditions for the latent group Lasso ((Obozinski et al., 2011a)) and group Lasso ((Nardi and Rinaldo, 2008)). According to Proposition 1, if $(\hat{V}^{(1)}, \hat{V}^{(2)}, \ldots, \hat{V}^{(p)})$ is a solution of problem (2.2), then for each i, either $\hat{V}^{(i)} = 0_{p\times 1}$ or $\text{supp}(\hat{V}^{(i)}) = \mathcal{N}_i$. Thus, the estimate $\hat{\beta} = \sum_{i=1}^p \hat{V}^{(i)}$ acquired by our proposed SRIG method has the same decomposition pattern as we discussed in Section 2.2.

2.4.2 Connections with some existing methods

The following proposition shows the connections between our proposed SRIG method and several other existing penalized methods when the given predictor graph has some special structures.

Proposition 2. (a) If the predictor graph has no edge, the proposed SRIG method is the same as the adaptive Lasso method for each tuning parameter λ; (b) If the predictor graph consists of K disconnected complete subgraphs, our proposed SRIG method is equivalent to the group Lasso method for each λ; (c) If the predictor graph is a complete graph, our proposed SRIG method has the same nonzero solution set as the ridge regression, i.e., for each nonzero solution acquired by ridge regression (or SRIG), SRIG (or ridge regression) could acquire the same solution using a different tuning parameter.

Proposition 2 indicates that the proposed SRIG method includes adaptive Lasso, group Lasso, and ridge regression as special cases. It is much more general and can handle any arbitrary predictor graph structure.

2.4.3 Finite Sample Bounds

In this section, we derive the oracle inequalities for the prediction and estimation loss of our proposed SRIG method. The design matrix $\mathbf{X}$ is treated as fixed in this subsection. For a given graph G, positive weights τ_j's and subset $J \subset \{1, 2 \ldots, p\}$, denote $\mathcal{T}_{G,\tau}(\beta, J)$ as the set of all optimal decompositions of β such that $\sum_{j \in J^c} \tau_j \|V^{(j)}\|_2 \leq 3 \sum_{j \in J} \tau_j \|V^{(j)}\|_2$. For each $1 \leq i \leq p$, denote d_i as the number of predictors in the neighborhood $\mathcal{N}_i$, i.e., $d_i = |\mathcal{N}_i|$. The following conditions are considered in this section.

(A1) The errors $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \overset{i.i.d.}{\sim} N(0, \sigma^2)$.

(A2) The neighborhood $\mathcal{N}_i \subseteq J_0$ for each $i \in J_0$.

(A3) There exists $\kappa > 0$ such that

$$\inf_{|J| \leq s_0, \beta \in R^p \setminus \{0\}} \quad \inf_{(V^{(1)}, V^{(2)}, \ldots, V^{(p)}) \in \mathcal{T}_{G,\tau}(\beta, J)} \frac{\|\mathbf{X}\beta\|_2}{\sqrt{n \sum_{j \in J} \tau_j^2 \|V^{(j)}\|_2^2}} \geq \kappa.$$

Note that condition (A1) is a common condition for linear regression. Condition (A2) assumes that the given predictor graph G is "consistent" with β^0, i.e., predictors connected to the useful predictor are also useful. Condition (A3) is similar to the restricted eigenvalue conditions used for the group Lasso ((Nardi and Rinaldo, 2008; Lounici et al., 2011)) and the

overlapped group Lasso ((Percival, 2012)). It is used to analyze the l_2 consistency property of both estimation and prediction.

Theorem 1. Suppose that conditions (A1), (A2) and (A3) are satisfied. Let $\tau_* = \min_{1\le i\le p}\tau_i$ and denote η_i as the positive square root of the largest eigenvalue of $\frac{1}{n}X_{\mathcal{N}_i}^T X_{\mathcal{N}_i}$. If we choose $\lambda\tau_i \ge \frac{2\sigma\eta_i}{\sqrt{n}}(d_i + Ad_i^{1/2}\log(p))^{1/2}$ where $A > 8$, then, for any optimal solution $\hat{\beta}$ of problem (2.5), we have

$$\frac{1}{n}\|\mathbf{X}(\hat{\beta}-\beta^0)\|_2^2 \le \frac{16\lambda^2 K_{G,\tau}}{\kappa^2},$$
$$\|\hat{\beta}-\beta^0\|_{G,\tau} \le \frac{16\lambda K_{G,\tau}}{\kappa^2},$$
$$\|\hat{\beta}-\beta^0\|_2 \le \frac{16\lambda K_{G,\tau}}{\kappa^2\tau_*},$$

with probability at least $1-p^{1-q}$, where $q = \frac{1}{8}\min\{A, A^2\log(p)\}$.

Remark 1. Note that the above results are very general and have close connections with the results shown in the literature. For example, when the predictor graph G has no edge, we have $K_{G,\tau} = s_0$ and $\|\hat{\beta}-\beta^0\|_{G,\tau} = \|\hat{\beta}-\beta^0\|_1$ if $\tau_i = 1$ for each i. Theorem 1 indicates that our proposed SRIG method acquires the same rates of prediction and estimation as the results shown in (Bickel et al., 2009) for the Lasso method. When the given graph G consists of some disconnected complete subgraphs and J_0 is the union of K_0 node sets of those disconnected subgraphs, we have $K_{G,\tau} = K_0$. In this case, we can also recover the results shown in (Nardi and Rinaldo, 2008) and (Lounici et al., 2011) for the group Lasso.

2.4.4 Asymptotic Normality and Model Selection Consistency

In this section, we first study the asymptotic normality for the case with a fixed dimension p. Then, we study the model selection consistency for the high dimensional case which allows p to grow with n. Both fixed design and random design are considered in these two cases. For every $\beta \in R^p$, denote β_{J_0} and $\beta_{J_0^c}$ as the sub-vectors of β with indices in J_0 and J_0^c respectively.

For the fixed p case, we use the following two common conditions:

(A4) As $n \to \infty$, $\mathbf{X}^T\mathbf{X}/n \to \mathcal{M}$, where $\mathcal{M}$ is a positive matrix.

(A5) The errors $\epsilon_1, \ldots, \epsilon_n$ are i.i.d. random variables with mean 0 and finite variance σ^2.

Theorem 2. Assume conditions (A2), (A4) and (A5) hold. Suppose the tuning parameter λ and weights τ_i's are chosen such that $\sqrt{n}\lambda \to 0$ and $n^{(\gamma+1)/2}\lambda \to \infty$ for some $\gamma > 0$. Furthermore, $\tau_j = O(1)$ for each $j \in J_0$ and $\liminf_{n\to\infty} n^{-\gamma/2}\tau_j > 0$ for each $j \in J_0^c$. Then, with dimension p fixed, as $n \to \infty$, we have

$$\sqrt{n}(\hat{\beta}_{J_0} - \beta^0_{J_0}) \xrightarrow{\text{d}} N(0, \sigma^2 \mathcal{M}^{-1}_{J_0,J_0}), \text{ and } \hat{\beta}_{J_0^c} \xrightarrow{\text{p}} 0,$$

where $\mathcal{M}_{J_0,J_0}$ is the sub-matrix of $\mathcal{M}$ consisting of the entries with row and column indices in J_0.

Remark 2. Theorem 2 indicates that our proposed SRIG method is estimation-consistent for the fixed p case. The estimates of the nonzero coefficients enjoy the asymptotic normality. Theorem 2 also provides a guideline on how to choose the positive weight τ_j. When $n > p$, similar to the weights used for the Adaptive Lasso ((Zou, 2006)), we can choose $\tau_j = \sqrt{d_j}/|\hat{\beta}_j^{\gamma}|$, where $\hat{\beta}_j$ is any $\sqrt{n}$-consistent estimate of β_j^0. Note that Theorem 2 can be extended to the random design setting naturally.

Corollary 1. Consider the random design setting where $x_1, x_2, \ldots, x_n$ are i.i.d. samples from a multivariate distribution with mean 0 and covariance matrix $\mathbf{\Sigma}$. Assume that the design matrix $\mathbf{X}$ and the errors ϵ are independent. Suppose conditions (A2) and (A5) hold. The tuning parameter λ and weights τ_i's are chosen such that $\sqrt{n}\lambda \to 0$ and $n^{(\gamma+1)/2}\lambda \to \infty$ for some $\gamma > 0$. Furthermore, $\tau_j = O(1)$ for each $j \in J_0$ and $\liminf_{n\to\infty} n^{-\gamma/2}\tau_j > 0$ for each $j \in J_0^c$. Then, with p fixed, as $n \to \infty$, we have

$$\sqrt{n}(\hat{\beta}_{J_0} - \beta^0_{J_0}) \xrightarrow{\text{d}} N(0, \sigma^2 \Sigma^{-1}_{J_0,J_0}), \text{ and } \hat{\beta}_{J_0^c} \xrightarrow{\text{p}} 0,$$

where Σ_{J_0,J_0} is the sub-matrix of $\mathbf{\Sigma}$ consisting of the entries with row and column indices in J_0.

For the high dimensional case which allows the dimension p to grow with n, if the design matrix $\mathbf{X}$ is considered to be fixed, we need the following conditions for model selection consistency.

(A6) The number of nonzero coefficients $s_0 = O(n^{\delta_0})$ for some constant $\delta_0 \in (0, 1)$.

(A7) There exists a constant $Q_1 > 0$ such that $\max_{j \in J_0^c} \|X_j\|_2 \leq \sqrt{n} Q_1$ for each n.

(A8) There exists a constant $Q_2 > 0$ such that the smallest eigenvalue of $X_{J_0}^T X_{J_0}/n$ is larger than Q_2 for each n.

(A9) There exists a constant $\xi \in (0, 1)$ such that $\|X_{J_0^c}^T X_{J_0} (X_{J_0}^T X_{J_0})^{-1}\|_\infty \leq 1 - \xi$, where for a $k \times m$ matrix $\mathbf{M}$, $\|\mathbf{M}\|_\infty$ is defined as $\max_{1 \leq i \leq k} \sum_{j=1}^m |M_{ij}|$.

Note that condition (A6) is a common sparsity assumption for the high dimensional regression problem. Condition (A7) can be satisfied by normalizing each predictor. Condition (A8) guarantees that the matrix $X_{J_0}^T X_{J_0}/n$ is invertible and its inverse behaves well. The main condition (A9) is similar to the strong irrepresentable condition used for Lasso ((Zhao and Yu, 2006)).

Theorem 3. Assume conditions (A1), (A2), (A6)-(A9) hold. Suppose the weight τ_j is chosen to be $\sqrt{d_j} m_j$ for each j, where the m_j's satisfy that $\max_{j \in J_0} m_j = O_p(1)$ and $\liminf_{n \to \infty} n^{-\gamma} \min_{j \in J_0^c} m_j > 0$ for some $\gamma > \delta_0$. Furthermore, the selected tuning parameter λ and the minimum absolute nonzero coefficient $\beta_{min}^0 = \min_{j \in J_0} |\beta_j^0|$ satisfy that, as $n \longrightarrow \infty$ and $p = p(n) \longrightarrow \infty$,

$$\frac{1}{\lambda} \sqrt{\frac{\log(p - s_0)}{n}} \max_{j \in J_0^c} \frac{\sqrt{d_j}}{\tau_j} \longrightarrow 0, \text{ and } \frac{1}{\beta_{min}^0} (3\sigma \sqrt{\frac{\log s_0}{n Q_2}} + \lambda \frac{\sqrt{s_0}}{Q_2} \max_{j \in J_0} \tau_j) \longrightarrow 0.$$

Then, as $n \longrightarrow \infty$ and $p = p(n) \longrightarrow \infty$, there exists a solution $\hat{\beta}$ to (2.5) such that sign($\hat{\beta}$)=sign(β^0) with probability tending to 1, where sign($\cdot$) maps a positive entry to 1, a negative entry to -1 and zero to zero.

Remark 3. For clarification, we note that many quantities such as p, s_0, λ, τ_j and d_j depend on n. We use simple notation here for convenience. Theorem 3 indicates that our

proposed SRIG method is model selection consistent for the high dimensional case. For example, suppose the dimension $p = O(e^{n^{\delta_1}})$ for some constant $\delta_1 \in (0, 1)$. Furthermore, for sufficiently large n, the minimum absolute nonzero coefficient β^0_{min} satisfies that $\beta^0_{min} \geq Q_3 n^{(\delta_2-1)/2}$ for some constants $Q_3 > 0$ and $\delta_2 > \delta_1$. If the weights τ_j's are selected as shown in the theorem and the tuning parameter λ is chosen to be $\lambda = O_p(n^{(\delta_1-2\delta_0-1)/2})$, then by Theorem 3 we can show that there exists a solution $\hat{\beta}$ such that sign($\hat{\beta}$)=sign(β^0) with probability tending to 1. In the high dimensional case with $p \gg n$, our simulation study suggests that choosing $\tau_j = \sqrt{d_j}/|\hat{cov}(X_j, Y)|^\gamma$ works well. The positive parameter γ can be chosen by cross-validation.

In Theorem 3, as the Lasso method, we use the irrepresentable condition (A9). In fact, we can also use the following condition (A9′) in order to reflect the use of the weights τ_j's. Following the same proof of Theorem 3, we can achieve the model selection consistency as shown in Corollary 2.

(A9′) There exists a constant $\xi \in (0, 1)$ such that for each $j \in J_0^c$, we have

$$\|X^T_{\mathcal{N}_j} X_{J_0} (X^T_{J_0} X_{J_0})^{-1}\|_\infty \leq \frac{\tau_j}{\sqrt{d_j}}(1 - \xi).$$

Corollary 2. Assume conditions (A1), (A2), (A6)-(A8), (A9′) hold. Suppose the weight τ_j's satisfy that $\sqrt{s_0} \max_{j\in J_0} \tau_j = o_p(1)$. Furthermore, the selected tuning parameter λ and the minimum absolute nonzero coefficient $\beta^0_{min} = \min_{j\in J_0} |\beta^0_j|$ satisfy the same conditions in Theorem 3, then, as $n \longrightarrow \infty$ and $p = p(n) \longrightarrow \infty$, there exists a solution $\hat{\beta}$ to (2.5) such that sign($\hat{\beta}$)=sign(β^0) with probability tending to 1.

Theorem 3 considers the fixed design setting. It can be extended to the random design setting as well. For that setting, the conditions (A6)-(A9) are replaced by the following conditions.

(A10) Let $x_1, x_2, \ldots, x_n \overset{i.i.d.}{\sim} N(0, \mathbf{\Sigma})$ with $\Sigma_{jj} = 1$ for each j. Furthermore, assume that $\mathbf{X}$ and ϵ are independent. The dimension $p < e^{n/(4Q_3^2)}$, where $Q_3 > 4\sqrt{5/3}$.

(A11) Restricted eigenvalue assumption:

$$\Lambda_{min}(s_0) = \frac{16}{17} \min_{J \subseteq \{1,2,\ldots,p\},\ |J| \leq s_0} \ \min_{\theta \neq 0,\ \theta_{J^c}=0} \frac{\theta^T \mathbf{\Sigma} \theta}{\|\theta_J\|_2^2} > 0.$$

(A12) The number of true nonzero coefficients $s_0 < (\Lambda_{min}(s_0)/(16Q_3))\sqrt{n/\log p}$.

Note that conditions (A10)-(A12) are common conditions used in the literature for the random design setting ((Bickel et al., 2009; Zhou et al., 2009)). Under these conditions, we can show that our proposed SRIG method is also model selection consistent for the high dimensional case with random design.

Theorem 4. Assume conditions (A1), (A2), (A10)-(A12) hold. Suppose the weight τ_j is chosen to be $\sqrt{d_j} m_j$ for each j, where $s_0^{3/2} \max_{j \in J_0} m_j = o(\sqrt{\Lambda_{min}(s_0)} \min_{j \in J_0^c} m_j)$. Furthermore, the selected tuning parameter λ and the minimum absolute nonzero coefficient $\beta_{min}^0 = \min_{j \in J_0} |\beta_j^0|$ satisfy that, as $n \longrightarrow \infty$ and $p = p(n) \longrightarrow \infty$,

$$\frac{1}{\lambda}\sqrt{\frac{\log(p - s_0)}{n}} \max_{j \in J_0^c} \frac{\sqrt{d_j}}{\tau_j} \longrightarrow 0, \qquad \frac{1}{\beta_{min}^0}(3\sigma\sqrt{\frac{\log s_0}{n\Lambda_{min}(s_0)}} + \lambda \frac{\sqrt{s_0}}{\Lambda_{min}(s_0)} \max_{j \in J_0} \tau_j) \longrightarrow 0.$$

Then, as $n \longrightarrow \infty$ and $p = p(n) \longrightarrow \infty$, there exists a solution $\hat{\beta}$ to (2.5) such that sign($\hat{\beta}$)=sign(β^0) with probability tending to 1, where sign($\cdot$) maps a positive entry to 1, a negative entry to -1 and zero to zero.

Remark 4. Under conditions (A10)-(A12), we can show that condition (A7) is satisfied with $Q_1 = \sqrt{3/2}$, condition (A8) is satisfied with $Q_2 = \Lambda_{min}(s_0)$, and $\|X_{J_0^c}^T X_{J_0}(X_{J_0}^T X_{J_0})^{-1}\|_\infty \leq \sqrt{3s_0/(2\Lambda_{min}(s_0))}$, with probability greater than $1 - 1/p^2$. Based on these results, we can use a similar proof of Theorem 3 to prove Theorem 4.

2.5 Simulation Study

In this section, we first compare our proposed SRIG method with many existing methods. Then, we conduct a sensitivity study of the SRIG method. Finally, we compare the computational costs of the predictor duplication method and our proposed iterative proximal algorithm using some examples.

2.5.1 Performance Comparison

To examine the performance of SRIG, we compare it with many other methods on three examples. Firstly, we compare SRIG with popular penalized methods such as Lasso, Ridge regression, Adaptive Lasso (ALasso) and Elastic net (Enet) which do not use the predictor graph structure information directly. Secondly, we compare SRIG with some existing methods using the predictor structure information. The competitors are GRACE ((Li and Li, 2008)) and GOSCAR ((Yang et al., 2012)). Thirdly, we compare SRIG with other latent component approaches such as principal component regression (PCR) and sparse partial least squares (SPLS) using the R packages **pls** ((Mevik and Wehrens, 2007)) and **spls** ((Chung et al., 2012)), respectively. In this simulation study, the predictor graph is defined by the precision matrix of the predictors. The performance of GRACE, GOSCAR and SRIG using both the estimated predictor graph and the oracle true predictor graph are evaluated on all examples. We denote GRACE-O, GOSCAR-O and SRIG-O as the GRACE, GOSCAR and SRIG methods using the true predictor graph, respectively. For comparison, we also show the performance of the least square method based on the true model, which is denoted as LS-O.

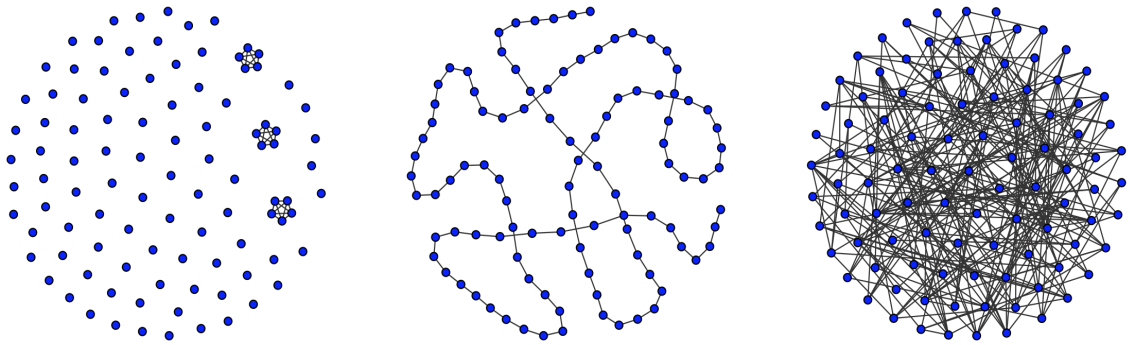


Figure 2.1: True predictor graphs of three simulation examples.

We generate data from model (2.1) with the errors $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \overset{i.i.d.}{\sim} N(0, \sigma^2)$. For each example, our simulated data include a training set, an independent validation set and an independent test set. All the models are fitted on the training data only. The validation data

are used to choose the tuning parameter and the test data set is used to evaluate different methods. We use the notation ./././ to show the sample sizes in the training, validation and test sets, respectively. For each example, we consider three cases: (I) 40/40/400, (II) 80/80/400 and (III) 120/120/400. For each case, we repeat the simulation 50 times. The predictor graph is estimated by the graphical Lasso method ((Friedman et al., 2008)) only using the training data in all cases.

Example 1: (Ω is block diagonal) $p = 100$, $s_0 = 15$, $\sigma = 5$, and the true coefficient vector $\beta^0 = (3, 3, \cdots, 3, 0, 0, \cdots, 0)$. The predictors are generated as:

$$
\begin{aligned}
&X_j = Z_1 + 0.4\epsilon_j^x, \quad Z_1 \sim N(0,1), \quad 1 \le j \le 5; \\
&X_j = Z_2 + 0.4\epsilon_j^x, \quad Z_2 \sim N(0,1), \quad 6 \le j \le 10, \\
&X_j = Z_3 + 0.4\epsilon_j^x, \quad Z_3 \sim N(0,1), \quad 11 \le j \le 15; \quad X_j \overset{i.i.d}{\sim} N(0,1), \quad 16 \le j \le 100,
\end{aligned}
$$

where $\epsilon_j^x \overset{i.i.d}{\sim} N(0,1), \quad j = 1, 2, \ldots, 15$.

Example 2: ($\mathbf{\Omega}$ is banded) $p = 100$, $\sigma = 10$, and β^0 is the same as the β^0 used in Example 1. The predictors $(X_1, X_2, \ldots, X_p)^T \sim N(0, \mathbf{\Sigma})$ with $\Sigma_{ij} = 0.5^{|i-j|}$. For this example, we have $\omega_{ii} = 1.333$, $\omega_{ij} = -0.667$ if $|i-j| = 1$ and $\omega_{ij} = 0$ if $|i-j| > 1$.

Example 3: ($\mathbf{\Omega}$ is sparse) $p = 100$, $\sigma = 5$, and the predictors $(X_1, X_2, \ldots, X_p)^T \sim N(0, \mathbf{\Omega}^{-1})$, where $\mathbf{\Omega} = \mathbf{L} + \delta\mathbf{I}$. Each off-diagonal entry in $\mathbf{L}$ is generated independently and equals to 0.5 with probability 0.05, or 0 with probability 0.95. The diagonal entry of $\mathbf{L}$ is 0. Here, δ is chosen such that the conditional number of $\mathbf{\Omega}$ is equal to p. Finally, $\mathbf{\Omega}$ is standardized to have unit diagonals. We set $\beta^0 = \mathbf{\Omega}\Sigma_{xy}$, where $\Sigma_{xy} = (c_1, c_2, \ldots, c_p)^T$ with $c_i = 10$ for the predictors having the top four largest degrees and $c_i = 0$ otherwise.

To evaluate different methods, we use the following measures:

- l_2 distance $\|\hat{\beta} - \beta^0\|_2$;
- Relative prediction error (RPE) $\frac{1}{\sigma^2 N_{test}}(\hat{\beta} - \beta^0)^T \mathbf{X}_{test}^T \mathbf{X}_{test}(\hat{\beta} - \beta^0)$, where $\mathbf{X}_{test}$ is the test samples and N_{test} is the number of test samples;

- False positive rate (FPR) and False negative rate (FNR);
- Nonzero match ratio (NMR) = $\frac{|\{(i,j):\ \Omega_{ij}\neq 0,\ \hat{\beta}_i\neq 0,\ \hat{\beta}_j\neq 0\}|}{|\{(i,j):\ \Omega_{ij}\neq 0,\ \beta_i^0\neq 0,\ \beta_j^0\neq 0\}|}$, which is used to check whether the estimated coefficients of two connected useful predictors are both nonzero; Zero match ratio (ZMR) = $\frac{|\{(i,j):\ \Omega_{ij}\neq 0,\ \hat{\beta}_i=0,\ \hat{\beta}_j=0\}|}{|\{(i,j):\ \Omega_{ij}\neq 0,\ \beta_i^0=0,\ \beta_j^0=0\}|}$, which is used to check whether the estimated coefficients of two connected useless predictors are both zero. We use NMR and ZMR when there is at least one edge connecting two useful predictors and one edge connecting two useless predictors. Thus, these two ratios are well defined and always between 0 and 1.

Table 2.1: Comparison of estimation and prediction (Example 1).

Methods	l_2 distance			RPE		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	8.378 (0.323)	5.014 (0.124)	4.132 (0.142)	0.595 (0.047)	0.212 (0.010)	0.149 (0.010)
Lasso	8.527 (0.199)	5.635 (0.119)	4.328 (0.153)	1.291 (0.087)	0.530 (0.036)	0.274 (0.014)
Ridge	8.166 (0.050)	7.585 (0.039)	4.325 (0.062)	12.336 (0.215)	10.936 (0.144)	0.946 (0.027)
ALasso	8.822 (0.275)	5.570 (0.167)	4.686 (0.147)	1.032 (0.093)	0.351 (0.041)	0.211 (0.012)
Enet	5.120 (0.201)	3.770 (0.110)	3.265 (0.092)	0.969 (0.071)	0.431 (0.031)	0.239 (0.012)
PCR	7.097 (0.104)	5.730 (0.096)	4.846 (0.080)	5.256 (0.253)	2.714 (0.134)	1.670 (0.092)
SPLS	4.147 (0.307)	3.150 (0.234)	2.752 (0.187)	1.046 (0.141)	0.777 (0.105)	0.494 (0.049)
GOSCAR	4.980 (0.273)	3.218 (0.139)	3.038 (0.108)	0.817 (0.070)	0.362 (0.024)	0.252 (0.010)
GOSCAR-O	5.051 (0.270)	3.220 (0.138)	3.027 (0.107)	0.811 (0.069)	0.363 (0.024)	0.255 (0.010)
GRACE	4.551 (0.142)	3.749 (0.091)	3.378 (0.122)	0.632 (0.050)	0.338 (0.021)	0.222 (0.011)
GRACE-O	4.554 (0.140)	3.743 (0.091)	3.371 (0.123)	0.633 (0.051)	0.338 (0.021)	0.222 (0.011)
SRIG	2.403 (0.065)	1.890 (0.064)	1.610 (0.046)	0.324 (0.037)	0.217 (0.015)	0.175 (0.013)
SRIG-O	2.392 (0.065)	1.820 (0.045)	1.564 (0.043)	0.320 (0.037)	0.208 (0.015)	0.171 (0.012)

Table 2.2: Comparison of model selection (Example 1).

Methods	FPR			FNR		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Lasso	0.087 (0.009)	0.145 (0.014)	0.123 (0.010)	0.171 (0.012)	0.027 (0.005)	0.003 (0.002)
Ridge	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
ALasso	0.039 (0.007)	0.027 (0.006)	0.041 (0.005)	0.173 (0.016)	0.021 (0.006)	0.007 (0.003)
Enet	0.131 (0.013)	0.171 (0.012)	0.148 (0.013)	0.032 (0.010)	0.000 (0.000)	0.000 (0.000)
PCR	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
SPLS	0.140 (0.034)	0.274 (0.043)	0.245 (0.034)	0.043 (0.011)	0.004 (0.002)	0.003 (0.002)
GOSCAR	0.190 (0.025)	0.226 (0.007)	0.307 (0.009)	0.039 (0.011)	0.003 (0.002)	0.000 (0.000)
GOSCAR-O	0.230 (0.032)	0.228 (0.007)	0.310 (0.009)	0.036 (0.011)	0.003 (0.002)	0.000 (0.000)
GRACE	0.136 (0.011)	0.135 (0.009)	0.127 (0.011)	0.005 (0.004)	0.000 (0.000)	0.000 (0.000)
GRACE-O	0.138 (0.011)	0.134 (0.009)	0.127 (0.011)	0.005 (0.004)	0.000 (0.000)	0.000 (0.000)
SRIG	0.001 (0.001)	0.003 (0.001)	0.003 (0.001)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
SRIG-O	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)

Figure 2.1 shows the true predictor graphs (defined by $\mathbf{\Omega}$) of these three examples. The numbers of edges for these three graphs are 30, 99 and 243, respectively. Such graphs were also studied in the literature previously ((Yang et al., 2012; Cai et al., 2011)). It is very interesting to study whether the structure information represented by these predictor graphs could be used to improve the performance of estimation, prediction and model selection. Tables 2.1 and 2.2 show the performance comparison for Example 1. The comparison results indicate that the Elastic net method acquires better estimation and prediction than Lasso, ridge regression and adaptive Lasso methods by using a linear combination of l_1 and ridge penalty. The GOSCAR and GRACE methods further improve the performance of estimation and prediction benefiting from using the addtional estimated predictor graph

directly. However, Elastic net, GOSCAR and GRACE methods still have relatively high FPR. Compared with the other methods (not including methods using the true predictor graph), our proposed SRIG method delivers the best performance of estimation and prediction. Furthermore, SRIG almost always identifies the true model perfectly for this example. Since the estimated predictor graph for this example is almost the same as the true predictor graph, the performance of GOSCAR-O, GRACE-O and SRIG-O are similar to those of GOSCAR, GRACE and SRIG respectively. Due to the strong correlation between different important predictors, the performance of LS-O method on this example is not very good. Compared with LS-O, our proposed SRIG method still acquires better performance of estimation and competitive results for prediction.

Table 2.3: Comparison of estimation and prediction (Example 2).

Methods	l_2 distance			RPE		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	9.312 (0.322)	6.193 (0.213)	4.926 (0.146)	0.575 (0.036)	0.235 (0.015)	0.149 (0.008)
Lasso	9.896 (0.205)	7.440 (0.159)	5.865 (0.130)	1.146 (0.061)	0.536 (0.022)	0.300 (0.012)
Ridge	9.298 (0.065)	8.571 (0.049)	6.496 (0.079)	2.240 (0.045)	1.914 (0.028)	0.500 (0.015)
ALasso	10.072 (0.192)	7.311 (0.181)	6.238 (0.157)	1.065 (0.056)	0.426 (0.021)	0.275 (0.011)
Enet	8.776 (0.197)	6.668 (0.142)	5.176 (0.103)	1.056 (0.057)	0.514 (0.023)	0.280 (0.011)
PCR	9.782 (0.110)	8.842 (0.125)	8.613 (0.132)	2.318 (0.071)	1.763 (0.074)	1.711 (0.077)
SPLS	8.423 (0.261)	5.480 (0.212)	4.062 (0.172)	0.900 (0.056)	0.321 (0.024)	0.194 (0.017)
GOSCAR	8.844 (0.243)	6.280 (0.173)	4.547 (0.123)	0.974 (0.051)	0.438 (0.023)	0.221 (0.009)
GOSCAR-O	5.662 (0.247)	4.666 (0.121)	4.416 (0.102)	0.566 (0.049)	0.287 (0.016)	0.208 (0.010)
GRACE	8.815 (0.235)	6.562 (0.152)	5.270 (0.112)	1.029 (0.055)	0.475 (0.021)	0.267 (0.011)
GRACE-O	8.238 (0.239)	6.353 (0.151)	5.084 (0.108)	0.972 (0.062)	0.453 (0.022)	0.254 (0.010)
SRIG	8.179 (0.200)	5.890 (0.130)	4.942 (0.104)	0.949 (0.068)	0.396 (0.022)	0.236 (0.009)
SRIG-O	7.354 (0.193)	5.257 (0.133)	4.245 (0.097)	0.718 (0.050)	0.284 (0.016)	0.167 (0.008)

Table 2.4: Comparison of model selection (Example 2).

Methods	FPR			FNR		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Lasso	0.154 (0.010)	0.171 (0.014)	0.158 (0.011)	0.304 (0.016)	0.099 (0.010)	0.025 (0.005)
Ridge	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
ALasso	0.121 (0.012)	0.071 (0.010)	0.081 (0.007)	0.303 (0.018)	0.121 (0.014)	0.052 (0.009)
Enet	0.311 (0.032)	0.273 (0.024)	0.223 (0.016)	0.168 (0.019)	0.051 (0.009)	0.005 (0.003)
PCR	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
SPLS	0.196 (0.030)	0.050 (0.011)	0.059 (0.021)	0.181 (0.021)	0.096 (0.013)	0.043 (0.007)
GOSCAR	0.271 (0.028)	0.369 (0.030)	0.354 (0.026)	0.164 (0.016)	0.027 (0.007)	0.005 (0.003)
GOSCAR-O	0.500 (0.038)	0.569 (0.020)	0.715 (0.017)	0.023 (0.008)	0.003 (0.002)	0.000 (0.000)
GRACE	0.440 (0.055)	0.203 (0.014)	0.174 (0.011)	0.109 (0.017)	0.055 (0.008)	0.011 (0.003)
GRACE-O	0.328 (0.045)	0.195 (0.013)	0.170 (0.011)	0.113 (0.016)	0.047 (0.008)	0.009 (0.003)
SRIG	0.283 (0.016)	0.275 (0.017)	0.243 (0.014)	0.112 (0.014)	0.028 (0.005)	0.009 (0.004)
SRIG-O	0.170 (0.016)	0.101 (0.013)	0.067 (0.008)	0.099 (0.012)	0.033 (0.006)	0.013 (0.004)

Tables 2.3 and 2.4 display the results for Example 2. As Example 1, the Elastic net method has better performance of estimation and prediction than Lasso and ridge regression. For the cases with relative large sample sizes, the adaptive Lasso method acquires better prediction than the Elastic net method. GOSCAR, GRACE and our proposed SRIG obtain better estimation and prediction than the methods not incorporating the additional predictor graph information. Methods using the true predictor graph acquire better estimation and prediction than those methods using estimated predictor graph, especially for the small sample cases (I and II). Compared with GOSCAR (GOSCAR-O) and GRACE (GRACE-O), our proposed SRIG (SRIG-O) has competitive performance of estimation and prediction. Furthermore, the results in Table 2.4 show that our proposed SRIG-O method

Table 2.5: Comparison of estimation and prediction (Example 3).

Methods	l_2 distance			RPE		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	2.668 (0.103)	1.769 (0.055)	1.324 (0.048)	0.401 (0.027)	0.172 (0.010)	0.103 (0.007)
Lasso	11.370 (0.131)	7.096 (0.186)	4.772 (0.106)	3.792 (0.080)	1.850 (0.090)	0.846 (0.035)
Ridge	12.140 (0.008)	12.100 (0.013)	11.026 (0.166)	4.006 (0.035)	3.979 (0.046)	3.779 (0.059)
ALasso	11.339 (0.147)	7.070 (0.184)	4.773 (0.105)	3.786 (0.078)	1.840 (0.088)	0.843 (0.035)
Enet	11.366 (0.129)	7.096 (0.186)	4.772 (0.106)	3.795 (0.076)	1.850 (0.090)	0.846 (0.035)
PCR	12.122 (0.010)	12.140 (0.007)	12.139 (0.008)	4.216 (0.044)	4.072 (0.043)	4.076 (0.049)
SPLS	12.080 (0.124)	11.219 (0.137)	10.858 (0.111)	5.990 (0.165)	5.247 (0.112)	4.664 (0.115)
GOSCAR	8.879 (0.220)	5.677 (0.151)	4.001 (0.090)	2.671 (0.117)	1.175 (0.056)	0.600 (0.025)
GOSCAR-O	8.709 (0.220)	5.454 (0.142)	3.900 (0.085)	2.510 (0.102)	1.094 (0.052)	0.571 (0.023)
GRACE	11.166 (0.140)	7.074 (0.184)	4.788 (0.105)	3.753 (0.088)	1.842 (0.089)	0.850 (0.035)
GRACE-O	10.140 (0.159)	7.085 (0.186)	4.787 (0.104)	3.279 (0.071)	1.822 (0.086)	0.848 (0.035)
SRIG	6.398 (0.223)	3.756 (0.131)	2.691 (0.076)	1.607 (0.093)	0.621 (0.040)	0.322 (0.018)
SRIG-O	4.150 (0.301)	2.344 (0.098)	1.736 (0.066)	0.804 (0.103)	0.254 (0.020)	0.141 (0.009)

acquires much lower FPR than the GOSCAR-O and GRACE-O methods. This indicates that GRACE and GOSCAR methods using the predictor graph edge-by-edge may lead to poor model selection results, although they can acquire competitive performance for estimation and prediction. Compared with latent component approaches, SRIG has better performance than PCR while worse performance than SPLS. However, SRIG-O has better performance than PCR and SPLS in most cases.

The performance comparison for Example 3 is shown in Tables 2.5 and 2.6. Methods not using the predictor graph have poor performance for both estimation, prediction and model selection, especially for the cases (I) and (II) with smaller n than p. For this example, the performance of estimation and prediction of the Elastic net method is similar to Lasso, ridge

Table 2.6: Comparison of model selection (Example 3).

Methods	FPR			FNR		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Lasso	0.152 (0.019)	0.467 (0.015)	0.481 (0.013)	0.793 (0.027)	0.129 (0.018)	0.011 (0.005)
Ridge	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
ALasso	0.155 (0.020)	0.469 (0.014)	0.473 (0.014)	0.776 (0.031)	0.124 (0.017)	0.011 (0.005)
Enet	0.233 (0.031)	0.467 (0.015)	0.481 (0.013)	0.716 (0.034)	0.129 (0.018)	0.011 (0.005)
PCR	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
SPLS	0.440 (0.050)	0.351 (0.044)	0.305 (0.042)	0.502 (0.053)	0.493 (0.046)	0.476 (0.049)
GOSCAR	0.292 (0.028)	0.378 (0.022)	0.380 (0.011)	0.438 (0.031)	0.060 (0.010)	0.004 (0.003)
GOSCAR-O	0.261 (0.024)	0.349 (0.016)	0.369 (0.012)	0.424 (0.030)	0.049 (0.009)	0.004 (0.003)
GRACE	0.220 (0.030)	0.472 (0.015)	0.481 (0.014)	0.711 (0.036)	0.120 (0.018)	0.011 (0.005)
GRACE-O	0.677 (0.058)	0.531 (0.028)	0.480 (0.014)	0.296 (0.055)	0.085 (0.015)	0.009 (0.004)
SRIG	0.216 (0.012)	0.266 (0.017)	0.245 (0.016)	0.109 (0.014)	0.015 (0.005)	0.000 (0.000)
SRIG-O	0.163 (0.018)	0.127 (0.018)	0.071 (0.015)	0.031 (0.018)	0.000 (0.000)	0.000 (0.000)

regression and adaptive Lasso. When the additional predictor graph information is used, the GRACE method, which can be considered as a graph version of the Elastic net, still does not acquire improved performance. However, GOSCAR benefits from the additional predictor graph information and acquires better performance. Compared with the other methods (not including SRIG-O), our proposed SRIG method has the best results for both estimation, prediction and model selection. As the previous two examples, each method using the true predictor graph performs better than the corresponding method using the estimated graph. For this example, LS-O acquires the best performance and our proposed SRIG-O method has similar results to the LS-O method when the sample size is large.

Table 2.7: Comparison of NMR and ZMR (Sample sizes: 40/40/400).

Methods	NMR			ZMR		
	Example 1	Example 2	Example 3	Example 1	Example 2	Example 3
LS-O	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	1.000 (0.000)	1.000 (0.000)
Lasso	0.679 (0.020)	0.480 (0.025)	0.149 (0.025)	–	0.717 (0.017)	0.743 (0.031)
Ridge	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	0.000 (0.000)	0.000 (0.000)
Alasso	0.681 (0.027)	0.494 (0.026)	0.167 (0.029)	–	0.779 (0.021)	0.738 (0.032)
Enet	0.939 (0.019)	0.710 (0.032)	0.215 (0.034)	–	0.520 (0.038)	0.642 (0.037)
PCR	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	0.000 (0.000)	0.000 (0.000)
SPLS	0.922 (0.020)	0.703 (0.033)	0.445 (0.057)	–	0.702 (0.038)	0.441 (0.056)
GOSCAR	0.927 (0.019)	0.717 (0.026)	0.491 (0.032)	–	0.593 (0.032)	0.528 (0.029)
GOSCAR-O	0.933 (0.019)	0.966 (0.012)	0.505 (0.032)	–	0.405 (0.040)	0.574 (0.028)
GRACE	0.989 (0.008)	0.813 (0.029)	0.227 (0.036)	–	0.462 (0.048)	0.658 (0.037)
GRACE-O	0.989 (0.008)	0.809 (0.027)	0.676 (0.059)	–	0.552 (0.040)	0.271 (0.052)
SRIG	1.000 (0.000)	0.841 (0.019)	0.864 (0.018)	–	0.579 (0.020)	0.627 (0.018)
SRIG-O	1.000 (0.000)	0.844 (0.017)	0.969 (0.018)	–	0.780 (0.020)	0.713 (0.030)

[– indicates that value is not available since there are no edges between useless predictors.]

The comparison results of NMR and ZMR for the cases with sample sizes 40/40/400, 80/80/400 and 120/120/400 are shown in Table 2.7, Table 2.8 and Table 2.9, respectively. Compared with the other methods (except LS-O which uses the underlying true model), our proposed SRIG-O acquires the best performance in most cases. The NMR's of SRIG-O indicate that our proposed SRIG method incorporates most edges between useful predictors efficiently and therefore chooses those connected useful predictors simultaneously. The ZMR's of SRIG-O indicate that our proposed SRIG-O method also makes use of most edges between useless predictors and therefore excludes those connected useless predictors jointly. Overall, for our proposed SRIG method, the estimated pattern (zero or nonzero) among coefficients agrees with the graphical structure very well.

Table 2.8: Comparison of NMR and ZMR (Sample sizes: 80/80/400).

Methods	NMR			ZMR		
	Example 1	Example 2	Example 3	Example 1	Example 2	Example 3
LS-O	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	1.000 (0.000)	1.000 (0.000)
Lasso	0.947 (0.011)	0.820 (0.018)	0.838 (0.023)	–	0.693 (0.022)	0.300 (0.017)
Ridge	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	0.000 (0.000)	0.000 (0.000)
Alasso	0.958 (0.011)	0.787 (0.024)	0.845 (0.021)	–	0.871 (0.017)	0.295 (0.016)
Enet	1.000 (0.000)	0.906 (0.017)	0.838 (0.023)	–	0.552 (0.031)	0.300 (0.017)
PCR	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	0.000 (0.000)	0.000 (0.000)
SPLS	0.992 (0.005)	0.846 (0.023)	0.447 (0.050)	–	0.914 (0.017)	0.519 (0.051)
GOSCAR	0.995 (0.004)	0.954 (0.012)	0.924 (0.014)	–	0.492 (0.031)	0.403 (0.020)
GOSCAR-O	0.995 (0.004)	0.996 (0.003)	0.938 (0.012)	–	0.341 (0.019)	0.432 (0.018)
GRACE	1.000 (0.000)	0.906 (0.015)	0.849 (0.022)	–	0.641 (0.021)	0.294 (0.018)
GRACE-O	1.000 (0.000)	0.920 (0.014)	0.893 (0.019)	–	0.652 (0.021)	0.261 (0.020)
SRIG	1.000 (0.000)	0.960 (0.008)	0.976 (0.009)	–	0.576 (0.023)	0.559 (0.023)
SRIG-O	1.000 (0.000)	0.949 (0.011)	1.000 (0.000)	–	0.870 (0.016)	0.780 (0.029)

[– indicates that value is not available since there are no edges between useless predictors.]

In conclusion, the simulation results indicate that our proposed SRIG method can make use of the structure information among predictors efficiently and performs well for both estimation, prediction and model selection.

2.5.2 Sensitivity Study

An important condition for our proposed SRIG method is the condition (A2) which requires that the predictor graph G is "consistent" with the true coefficients vector β^0, i.e., predictors connected to the useful predictor are also useful. Since it is difficult to check this condition in practice, it is very important to study the performance of SRIG when the condition (A2) is violated.

Table 2.9: Comparison of NMR and ZMR (Sample sizes: 120/120/400).

Methods	NMR			ZMR		
	Example 1	Example 2	Example 3	Example 1	Example 2	Example 3
LS-O	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	1.000 (0.000)	1.000 (0.000)
Lasso	0.995 (0.004)	0.957 (0.009)	0.985 (0.007)	–	0.711 (0.019)	0.275 (0.015)
Ridge	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	0.000 (0.000)	0.000 (0.000)
Alasso	0.987 (0.006)	0.906 (0.014)	0.985 (0.007)	–	0.845 (0.013)	0.284 (0.016)
Enet	1.000 (0.000)	0.990 (0.005)	0.985 (0.007)	–	0.614 (0.023)	0.275 (0.015)
PCR	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	–	0.000 (0.000)	0.000 (0.000)
SPLS	0.995 (0.004)	0.934 (0.013)	0.462 (0.053)	–	0.909 (0.028)	0.572 (0.049)
GOSCAR	1.000 (0.000)	0.990 (0.005)	0.995 (0.004)	–	0.509 (0.029)	0.373 (0.013)
GOSCAR-O	1.000 (0.000)	1.000 (0.000)	0.995 (0.004)	–	0.198 (0.014)	0.393 (0.015)
GRACE	1.000 (0.000)	0.980 (0.007)	0.985 (0.007)	–	0.685 (0.018)	0.278 (0.015)
GRACE-O	1.000 (0.000)	0.981 (0.007)	0.987 (0.006)	–	0.692 (0.018)	0.277 (0.015)
SRIG	1.000 (0.000)	0.986 (0.006)	1.000 (0.000)	–	0.613 (0.019)	0.593 (0.025)
SRIG-O	1.000 (0.000)	0.979 (0.007)	1.000 (0.000)	–	0.912 (0.011)	0.870 (0.026)

[– indicates that value is not available since there are no edges between useless predictors.]

To this end, we evaluate the performance of SRIG on a series of data sets with changing predictor graphs. Fix $p = 100$, $\sigma = 3$, $s_0 = 20$, and $\beta^0 = (20, 2, 2, \cdots, 2, 0, 0, \cdots, 0)$. For each $p^* = 0, 1, \ldots, 30$, we generate the predictor matrix $\mathbf{X}$ from $N(0, \mathbf{\Omega}^{-1})$, where $\mathbf{\Omega} = \mathbf{L} + 2|\lambda_{max}(\mathbf{L})|I_p$. Here, $L_{ii} = 2$ for each $1 \leq i \leq p$, $L_{1i} = L_{i1} = 0.3$ for each $1 \leq i \leq (s_0 + p^*)$, $L_{(s_0+1)i} = L_{i(s_0+1)} = 0.3$ for each $(s_0 + 1) \leq i \leq p$, and $L_{ij} = 0$ otherwise. Finally, $\mathbf{\Omega}$ is standardized to have unit diagonals.

For this study, the true precision matrix $\mathbf{\Omega}$ is used to construct the predictor graph G. The neighborhoods of the useful predictor X_1 and the useless predictor X_{s_0+1} are $\mathcal{N}_1 = \{1, 2, \ldots, s_0 + p^*\}$ and $\mathcal{N}_{s_0+1} = \{s_0 + 1, s_0 + 2, \ldots, p\}$, respectively. The number of predictors shared by these two neighborhood is $|\mathcal{N}_1 \cap \mathcal{N}_{s_0+1}| = |\{s_0 + 1, s_0 + 2, \ldots, s_0 + p^*\}| = p^*$. The

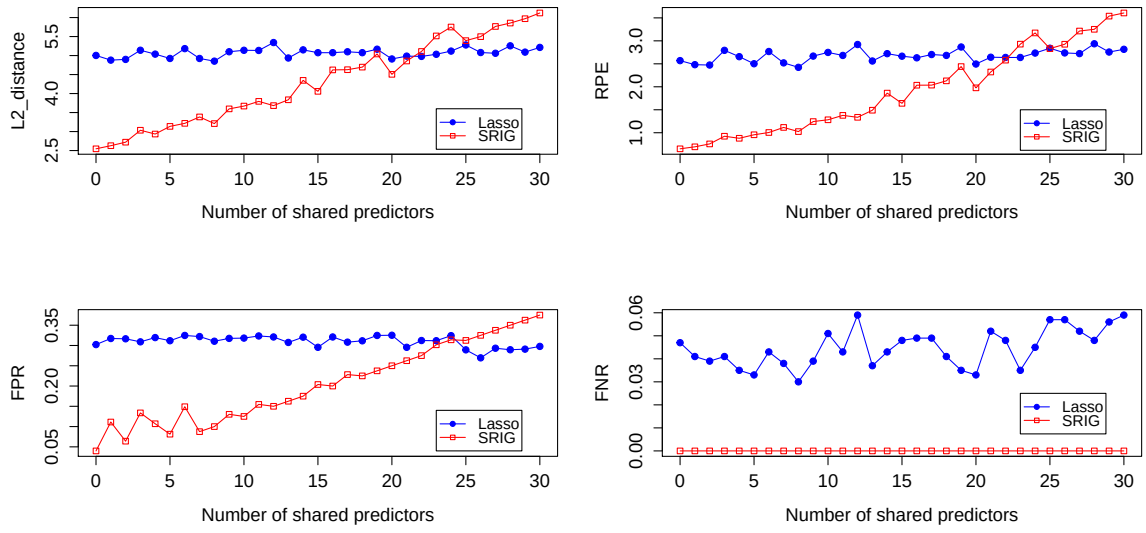


Figure 2.2: Sensitivity study of the SRIG method.

condition (A2) is satisfied when $p^* = 0$ and will be violated more and more seriously as p^* increases. Based on this example, we study the robustness of SRIG as p^* changes gradually from 0 to 30. For each p^*, we also evaluate the performance of Lasso method. The sample sizes are fixed as 80/80/400.

Figure 2.2 shows the performances of SRIG and Lasso method as the number of shared predictors p^* increases. It indicates that Lasso method is more robust than our proposed SRIG method to the intersection between the neighborhood of useful predictors and the neighborhood of useless predictors. One possible reason is that Lasso does not use the predictor graph information directly. For our proposed SRIG method, as p^* increases, the condition (A2) is more and more violated and the performance of SRIG gets worse. As shown in Figure 2.2, if the condition (A2) is not violated seriously, our proposed SRIG method still has better performance than the Lasso method. However, if (A2) is violated seriously (i.e., $p^* > 25$), Lasso method performs better than our proposed SRIG method.

Besides this study, we also compare SRIG with the other methods on the following example:

Table 2.10: Comparison of estimation and prediction (Adjusted Example 2).

Methods	l_2 distance			RPE		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	8.862 (0.263)	6.061 (0.203)	4.572 (0.123)	0.536 (0.027)	0.242 (0.016)	0.139 (0.006)
Lasso	9.935 (0.181)	7.871 (0.150)	6.076 (0.122)	1.137 (0.051)	0.614 (0.030)	0.327 (0.012)
Ridge	9.549 (0.054)	8.936 (0.048)	6.992 (0.088)	1.912 (0.030)	1.652 (0.026)	0.535 (0.015)
ALasso	10.018 (0.190)	7.819 (0.163)	6.298 (0.144)	1.072 (0.050)	0.521 (0.029)	0.311 (0.013)
Enet	8.981 (0.162)	7.270 (0.156)	5.633 (0.115)	1.047 (0.045)	0.598 (0.027)	0.316 (0.012)
PCR	10.036 (0.090)	10.005 (0.082)	9.443 (0.085)	2.011 (0.051)	1.935 (0.051)	1.640 (0.042)
SPLS	9.491 (0.282)	6.846 (0.218)	4.829 (0.172)	1.038 (0.052)	0.454 (0.027)	0.239 (0.016)
GOSCAR	10.458 (0.244)	6.643 (0.161)	4.850 (0.125)	1.156 (0.059)	0.463 (0.022)	0.244 (0.013)
GOSCAR-O	7.167 (0.281)	5.531 (0.132)	4.907 (0.124)	0.703 (0.052)	0.364 (0.021)	0.248 (0.012)
GRACE	9.952 (0.287)	7.072 (0.171)	5.470 (0.110)	1.107 (0.058)	0.544 (0.029)	0.293 (0.012)
GRACE-O	8.840 (0.203)	7.021 (0.149)	5.450 (0.108)	0.987 (0.051)	0.532 (0.024)	0.292 (0.012)
SRIG	9.024 (0.202)	6.065 (0.131)	4.433 (0.110)	0.969 (0.054)	0.395 (0.025)	0.190 (0.010)
SRIG-O	7.843 (0.192)	5.858 (0.140)	4.422 (0.107)	0.777 (0.049)	0.371 (0.025)	0.189 (0.010)

Adjusted Example 2: This example is almost the same as Example 2. We only change the true coefficient vector in Example 2 to

$$\beta^0 = (\underbrace{3, \cdots, 3}_{5}, \underbrace{0, \cdots, 0}_{5}, \underbrace{3, \cdots, 3}_{5}, \underbrace{0, \cdots, 0}_{5}, \underbrace{3, \cdots, 3}_{5}, \underbrace{0, \cdots, 0}_{75}).$$

For the above example, the condition (A2) in Section 2.4.3 is much violated. The simulation results shown in Tables 2.10 and 2.11 indicate that our proposed SRIG method still performs as well as the other methods.

Table 2.11: Comparison of model selection (Adjusted Example 2).

Methods	FPR			FNR		
	(I)	(II)	(III)	(I)	(II)	(III)
LS-O	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Lasso	0.144 (0.009)	0.188 (0.014)	0.184 (0.011)	0.340 (0.017)	0.128 (0.015)	0.029 (0.006)
Ridge	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
ALasso	0.109 (0.010)	0.114 (0.013)	0.122 (0.010)	0.352 (0.019)	0.144 (0.015)	0.051 (0.008)
Enet	0.362 (0.032)	0.343 (0.028)	0.229 (0.012)	0.151 (0.016)	0.045 (0.008)	0.019 (0.005)
PCR	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
SPLS	0.198 (0.034)	0.082 (0.015)	0.076 (0.016)	0.277 (0.026)	0.155 (0.019)	0.049 (0.009)
GOSCAR	0.246 (0.018)	0.496 (0.019)	0.651 (0.019)	0.252 (0.018)	0.013 (0.004)	0.001 (0.001)
GOSCAR-O	0.460 (0.036)	0.575 (0.022)	0.739 (0.018)	0.047 (0.011)	0.003 (0.002)	0.001 (0.001)
GRACE	0.242 (0.030)	0.233 (0.020)	0.193 (0.010)	0.248 (0.021)	0.060 (0.009)	0.011 (0.003)
GRACE-O	0.316 (0.039)	0.234 (0.020)	0.193 (0.011)	0.144 (0.016)	0.064 (0.010)	0.011 (0.003)
SRIG	0.131 (0.010)	0.183 (0.014)	0.132 (0.011)	0.293 (0.019)	0.053 (0.010)	0.007 (0.003)
SRIG-O	0.179 (0.013)	0.164 (0.013)	0.119 (0.010)	0.143 (0.014)	0.039 (0.009)	0.008 (0.004)

2.5.3 PD method v.s. IP algorithm

In this subsection, we compare the computational costs of the PD method and our proposed IP algorithm by some examples. Besides the Examples 1-3 shown in Section 2.5.1, we also consider the following three high dimensional examples:

Example 4: $n = 400$, $p = 1500$, $s_0 = 25$, $\sigma = 5$, and the true coefficient vector $\beta^0 = (1, 1, \cdots, 1, 0, \cdots, 0)$. The predictors are generated as follows.

$$X_j = Z_1 + \epsilon_j^x, \quad Z_1 \sim N(0,1), \ \ 1 \le j \le 25,$$
$$X_j = Z_2 + \epsilon_j^x, \quad Z_2 \sim N(0,1), \ \ 26 \le j \le 50,$$
$$(X_{51}, X_{52}, \ldots, X_p)^T \sim N(0, \mathbf{\Omega}_*^{-1}),$$

Table 2.12: Time comparison between PD method and IP algorithm.

Examples	n	p	Nedges	p_{new}/p	Time_{PD} (seconds)	Time_{IP} (seconds)
1	40	100	30	1.600	0.083	0.436
2	40	100	99	2.980	0.181	14.850
3	40	100	243	5.860	0.485	44.158
4	400	1500	263229	351.972	277.326	74.701
5	500	2000	475289	476.289	796.735	81.051
6	600	2500	750074	601.059	NA	96.436

[Nedges: the number of edges in the graph G; p_{new}: the number of predictors in the duplicated predictor matrix; Time_{PD}: computing time of the PD method; Time_{IP}: computing time of the IP algorithm; NA: out of memory.]

where $\epsilon_j^x \overset{i.i.d}{\sim} N(0,1)$, $j = 1, 2, \ldots, 50$ and $\mathbf{\Omega}_* = \mathbf{L} + \delta\mathbf{I}$. Each off-diagonal entry in $\mathbf{L}$ is generated independently and equals to 0.5 with probability 0.25, or 0 with probability 0.75. The diagonal entry of $\mathbf{L}$ is 0. Here, δ is chosen such that the conditional number of $\mathbf{\Omega}_*$ is equal to $p - 50$. Finally, $\mathbf{\Omega}_*$ is standardized to have unit diagonals.

Example 5: $n = 500$, $p = 2000$ and the other setup is the same as Example 4.

Example 6: $n = 600$, $p = 2500$ and the other setup is the same as Example 4.

For these six examples, we use both the PD method (using **gglasso** R package) and our proposed IP algorithm to compute the solution path of the SRIG method using the true predictor graph. To be specific, we set all the weights τ_i's to be 1 and compute the set of solutions corresponding to 100 different values of the tuning parameter $\lambda_1 > \lambda_2 > \cdots > \lambda_{100}$, where $\lambda_1 = \|X^T Y/n\|_2$ which shrinks all the parameters to be 0 and $\lambda_{100} = 0.05\lambda_1$. The computational times (in seconds) of PD method and IP algorithm are shown in Table 2.9.

As shown in Table 2.12, both methods require more time to compute the solution path as the dimension p and the number of edges in the predictor graph increase. When p is small and at the same time the predictor graph G is sparse (e.g., Examples 1-3), the PD method is faster than the IP algorithm. However, for high dimensional data sets with complicate predictor graphs (e.g., Examples 4-6), our proposed IP algorithm is more efficient than the

PD method. For Example 6, the PD method using **gglasso** package breaks down due to out of memory while our proposed IP algorithm still works well. In this case, the proposed IP algorithm is very desirable.

2.6 Real Data Example

Alzheimer's disease (AD) is one of the most common forms of dementia characterized by progressive cognitive and memory deficits. The increasing incidence of AD makes the disease a very important health issue and a huge financial burden for both patients and governments ((Hebert et al., 2001)). In the practical diagnosis of AD, the Mini Mental State Examination (MMSE) ((Folstein et al., 1975)) score is a very important reference. MMSE is a brief 30-point questionnaire test that is used to screen for cognitive impairment. It can be used to examine patient's arithmetic, memory and orientation. Generally, any score greater than or equal to 27 points (out of 30) indicates a normal cognition. Below this, MMSE score can indicate severe ($\leq$9 points), moderate (10-18 points) or mild (19-24 points) cognitive impairment ((Mungas, 1991)). As more and more treatments are being developed and evaluated, it is very important to develop diagnostic and prognostic biomarkers that can predict which individuals are relatively more likely to progress clinically. At present, structural magnetic resonance imaging (MRI) is one of the most popular and powerful techniques for the diagnosis of AD. It is very interesting to use MRI data to predict MMSE score which can be used to diagnose the current disease status of AD.

The dataset we used in this analysis is the MRI data and MMSE scores of 51 AD patients and 52 normal controls from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (www.loni.ucla.edu/ADNI). The image pre-processing steps for the MRI data include anterior commissure posterior commissure correction, intensity inhomogeneity correction, skull stripping, cerebellum removal, spatial segmentation, and registration. After registration, we obtained the subject-labeled image based a template with 93 manually labeled regions of interest (ROI) ((Kabani et al., 1998)). For each of the 93 ROI in the labeled MRI, we computed the volume of GM tissue as a feature. Therefore, the final dataset has

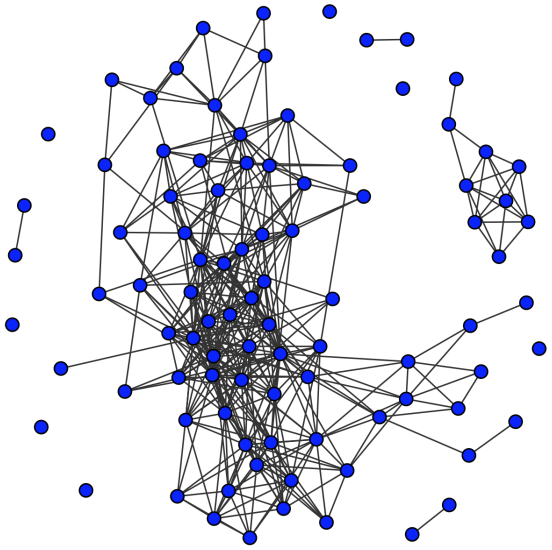


Figure 2.3: Estimated graph of 93 MRI features.

103 subjects. For each subject, there are one MMSE score and 93 MRI features. We treat MMSE score as the response variable and MRI features as predictors in our model.

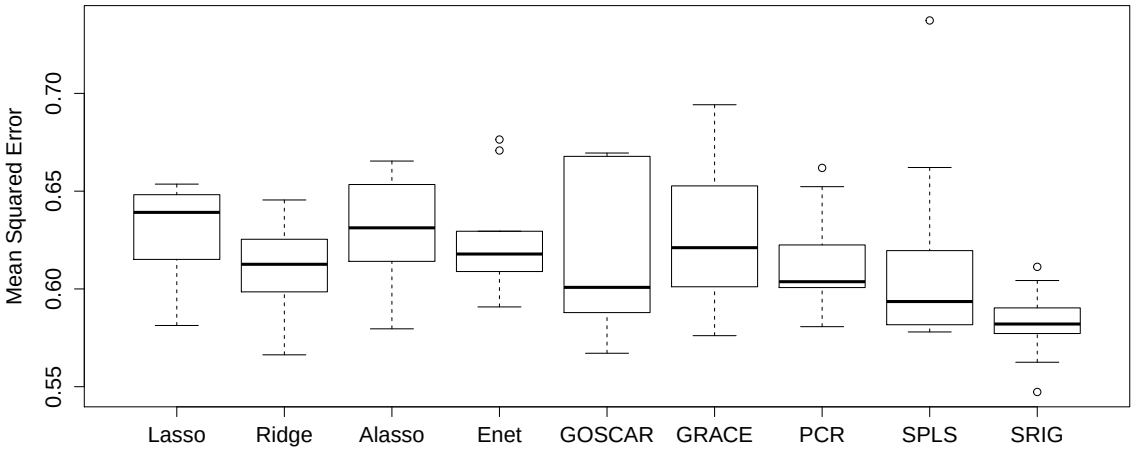


Figure 2.4: Comparison of MSE for various methods on the ADNI data set.

To evaluate the performance of our proposed SRIG method, we compare it with Lasso, ridge regression, Adaptive Lasso, Elastic net, GOSCAR, GRACE, PCR and SPLS. The dataset is first scaled to have mean 0 and variance 1 for the MMSE score and each MRI

feature. The 10-fold cross validation (CV) is used to evaluate different methods. The predictor (MRI feature) graph G is estimated by the graphical Lasso ((Friedman et al., 2008)) only using the training data. Figure 2.3 shows the estimated MRI feature graph using all the data. There are 93 nodes and 419 edges in this graph. Note that all the models are fitted using training data and evaluated by the mean squared error (MSE) calculated from the testing data. To choose the tuning parameters of different methods, an inner 5-fold CV is used. Considering possible bias due to the random splitting, we repeat 10-CV process ten times. Figure 2.4 shows the box plot of the averaged mean squared errors of different methods. Compared with the other methods, our proposed SRIG method delivers the best prediction of MMSE scores. The averaged MSE acquired by our proposed SRIG method is 0.5822, which is about 4.6% percent lower than the smallest MSE acquired by the competitors.

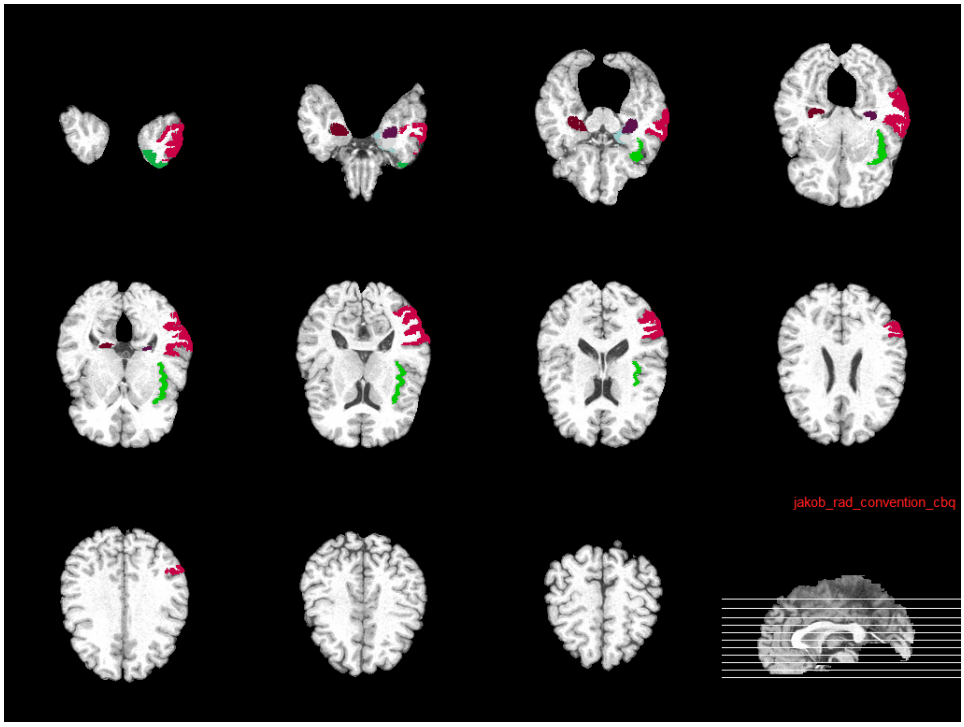


Figure 2.5: The multi-slice view of seven brain regions always selected by SRIG method.

For the ten times of our 10-CV process, we acquires 100 models for each method. For our proposed SRIG method, the averaged number of selected MRI features (with estimated coefficients bigger than 0.01) is almost 36. There are seven MRI features always selected by our proposed SRIG method. The feature indices are 4, 19, 22, 30, 69, 80 and 83. Figure 2.5 shows the multi-slice view of the brain regions corresponding to these seven MRI features. The colored areas are the selected regions. Interestingly, the 30th and 69th features correspond to the hippocampal regions. The 22th and 83th features correspond to the uncus region and the amygdala region respectively. These regions are known to be related to AD by many previous studies based on group comparison methods ((Jack et al., 1999; Misra et al., 2009a; Zhang and Shen, 2012)). Moreover, we notice that the 4th, 19th and 80th features relate to the insula right, temporal pole right and middle temporal gyrus right regions respectively. It would be very interesting to check whether these regions are substantially related to AD by some group comparison studies.

2.7 Conclusion

In this chapter, we propose a new penalized regression method using structure information among predictors. Instead of using the predictor graph *edge-by-edge* as in the existing literature, our proposed SRIG method uses it *node-by-node*. Theoretical study shows that SRIG includes adaptive Lasso, group Lasso and ridge regression as special cases. It is able to make use of the general structure information among predictors efficiently. Furthermore, SRIG acquires tight finite sample bounds for both prediction and estimation. It also enjoys asymptotic normality and model selection consistency. Both simulation study and real data analysis show that SRIG is a competitive tool for estimation, prediction and model selection.

2.8 Proofs

Proof of Proposition 1:

Define $L(\beta) = \frac{1}{2n}\|Y - \mathbf{X}\beta\|_2^2$ and denote $\nabla_{\mathcal{N}_i} L(\beta) \in R^{|\mathcal{N}_i|}$ as the partial gradient of $L(\beta)$ with respect to the predictors in $\mathcal{N}_i$, then we have $\nabla_{\mathcal{N}_i} L(\beta) = -\frac{1}{n} X_{\mathcal{N}_i}^T (Y - \mathbf{X}\beta)$. Proposition 1 is immediate from Lemma 11 in (Obozinski et al., 2011a). □

Proof of Proposition 2:

(a) If the predictor graph has no edge, then $\mathcal{N}_i = \{i\}$ for each i and

$$\|\beta\|_{G,\tau} = \min_{\sum_{i=1}^p V^{(i)} = \beta,\ \ \mathrm{supp}(V^{(i)}) \subseteq \mathcal{N}_i} \sum_{i=1}^p \tau_i \|V^{(i)}\|_2 = \sum_{i=1}^p \tau_i |\beta_i|.$$

Thus, for each given tuning parameter λ, SRIG and adaptive Lasso are equivalent.

(b) Without loss of generality, suppose the nodes in the K disconnected complete subgraphs are $\{1, 2, \ldots, p_1\}$, $\{p_1 + 1, \ldots, p_1 + p_2\}$, $\ldots$, $\{p_{K-1} + 1, \ldots, p_{K-1} + p_K\}$ respectively. Then, for $1 \le k \le K$ and $p_{k-1}+1 \le j \le p_{k-1}+p_k$, we have $\mathcal{N}_j = \{p_{k-1}+1, \ldots, p_{k-1}+p_k\}$.

Furthermore, for each k, we have

$$\sum_{j=p_{k-1}+1}^{p_{k-1}+p_k} V_{\mathcal{N}_j}^{(j)} = \beta^{(k)}, \text{ where } \beta^{(k)} = (\beta_{p_{k-1}+1}, \beta_{p_{k-1}+2}, \ldots, \beta_{p_{k-1}+p_k})^T.$$

Hence,

$$\|\beta\|_{G,\tau} = \sum_{k=1}^K \min_{\sum_{j=p_{k-1}+1}^{p_{k-1}+p_k} V^{(j)} = \beta^{(k)}} \sum_{j=p_{k-1}+1}^{p_{k-1}+p_k} \tau_j \|V^{(j)}\|_2.$$

For each k, let $\tilde{\tau}_k = \min_{p_{k-1}+1 \le j \le p_k} \tau_j$, by the inequality $\|\sum_{i=1}^n a_i\|_2 \le \sum_{i=1}^n \|a_i\|_2$, we have

$$\|\beta\|_{G,\tau} = \sum_{k=1}^K \min_{\sum_{j=p_{k-1}+1}^{p_{k-1}+p_k} V^{(j)} = \beta^{(k)}} \sum_{j=p_{k-1}+1}^{p_{k-1}+p_k} \tau_j \|V^{(j)}\|_2 = \sum_{k=1}^K \tilde{\tau}_k \|\beta^{(k)}\|_2.$$

Hence, in this case, SRIG is equivalent to the group lasso method.

(c) If the predictor graph is a complete graph and let $\tilde{\tau} = \min_{1 \le j \le p} \tau_j$, by the proof of (b) with $K = 1$, our proposed SRIG method is equivalent to the following optimization problem:

$$\min_{\beta \in R^p} \frac{1}{2n} \|Y - \mathbf{X}\beta\|_2^2 + \lambda \tilde{\tau} \|\beta\|_2 \tag{2.9}$$

By the Karush-Kuhn-Tucker conditions, a nonzero $\hat{\beta}_\lambda$ is a solution of (2.9) if and only if

$$-\mathbf{X}^T(Y - \mathbf{X}\hat{\beta}_\lambda) + n\lambda\tilde{\tau}\hat{\beta}_\lambda / \|\hat{\beta}_\lambda\|_2 = 0$$

Thus, $\hat{\beta}_\lambda$ is also the solution of ridge regression

$$\min_{\beta \in R^p} \frac{1}{2n} \|Y - \mathbf{X}\beta\|_2^2 + \lambda_* \beta^T \beta, \tag{2.10}$$

where $\lambda_* = \frac{\lambda\tilde{\tau}}{2\|\hat{\beta}_\lambda\|_2}$.

Furthermore, if $\tilde{\beta}_{\lambda_*}$ is the solution of ridge regression with tuning parameter λ_*, then $\tilde{\beta}_{\lambda_*}$ is also the solution of (2.9) with $\lambda = \frac{2\lambda_* \|\tilde{\beta}_{\lambda_*}\|_2}{\tilde{\tau}}$. Hence, in this case, SRIG and ridge regression have the same nonzero solution set. □

Lemma 1. Let χ_D^2 be a chi-squared random variable with D degrees of freedom. Then, for all $x > 0$, we have

$$P(\chi_D^2 > D + x) \le \exp(-\frac{1}{8} \min\{x, \frac{x^2}{D}\}).$$

Proof of Lemma 1: See Lemma A.1 from (Lounici et al., 2009). □

Lemma 2. For any predictor graph G and positive weights τ_i's, suppose $V^{(1)}, V^{(2)}, \ldots, V^{(p)}$ is an optimal decomposition of $\beta \in R^p$, then for any $S \subset \{1, 2, \ldots, p\}$, $\{V^{(j)} : j \in S\}$ is also an optimal decomposition of $\sum_{j \in S} V^{(j)}$.

Proof of Lemma 2: We prove this statement by contradiction. Suppose $\{V^{(j)} : j \in S\}$ is not an optimal decomposition of $\sum_{j \in S} V^{(j)}$ and denote the optimal decomposition of

$\sum_{j\in S} V^{(j)}$ as $M^{(1)}, M^{(2)}, \ldots, M^{(p)}$. Then, we have

$$\sum_{j\in S} V^{(j)} = \sum_{i=1}^{p} M^{(i)} \text{ and } \sum_{i=1}^{p} \tau_i \|M^{(i)}\|_2 < \sum_{j\in S} \tau_j \|V^{(j)}\|_2.$$

Hence, we have

$$\begin{aligned}
\|\beta\|_{G,\tau} &= \|\sum_{j\in S} V^{(j)} + \sum_{j\in S^c} V^{(j)}\|_{G,\tau} \leq \|\sum_{i=1}^{p} M^{(i)}\|_{G,\tau} + \|\sum_{j\in S^c} V^{(j)}\|_{G,\tau} \\
&= \sum_{i=1}^{p} \tau_i \|M^{(i)}\|_2 + \|\sum_{j\in S^c} V^{(j)}\|_{G,\tau} < \sum_{j\in S} \tau_j \|V^{(j)}\|_2 + \|\sum_{j\in S^c} V^{(j)}\|_{G,\tau} \\
&\leq \sum_{j\in S} \tau_j \|V^{(j)}\|_2 + \sum_{j\in S^c} \|V^{(j)}\|_{G,\tau} \leq \sum_{j\in S} \tau_j \|V^{(j)}\|_2 + \sum_{j\in S^c} \tau_j \|V^{(j)}\|_2 = \|\beta\|_{G,\tau}.
\end{aligned}$$

Contradiction! □

Proof of Theorem 1:

For any $\beta \in R^p$, we have $\frac{1}{2n}\|Y - \mathbf{X}\hat{\beta}\|_2^2 + \lambda\|\hat{\beta}\|_{G,\tau} \leq \frac{1}{2n}\|Y - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_{G,\tau}$. Since $Y = \mathbf{X}\beta^0 + \epsilon$, by simple calculation, we have

$$\frac{1}{n}\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2^2 \leq \frac{1}{n}\|\mathbf{X}(\beta - \beta^0)\|_2^2 + \frac{2}{n}\epsilon^T\mathbf{X}(\hat{\beta} - \beta) + 2\lambda(\|\beta\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau}). \tag{2.11}$$

Furthermore, denote $\{V^{(1)}, V^{(2)}, \ldots, V^{(p)}\}$ as arbitrary optimal decomposition of $\hat{\beta} - \beta$. Then,

$$\begin{aligned}
\|\frac{2}{n}\epsilon^T\mathbf{X}(\hat{\beta} - \beta)\|_2 &= \|\frac{2}{n}\epsilon^T \sum_{i=1}^{p} X_{\mathcal{N}_i} V^{(i)}_{\mathcal{N}_i}\|_2 = \|\frac{2}{n}\epsilon^T \sum_{i=1}^{p} X_{\mathcal{N}_i} \frac{1}{\tau_i}\tau_i V^{(i)}_{\mathcal{N}_i}\|_2 \\
&\leq \frac{2}{n} \sum_{i=1}^{K_{G,\tau}} \|\epsilon^T X_{\mathcal{N}_i}/\tau_i\|_2 \|\tau_i V^{(i)}_{\mathcal{N}_i}\|_2.
\end{aligned}$$

Define event $\mathcal{A} = \{\|\epsilon^T X_{\mathcal{N}_i}\|_2 \leq n\lambda\tau_i/2, \text{ for each } i\}$. We have

$$\mathcal{A}^c = \cup_{i=1}^{K_{G,\tau}} \{\sum_{j\in\mathcal{N}_i} (\frac{1}{\sqrt{n}} \sum_{i=1}^{n} X_{ij}\epsilon_i)^2 > \frac{n\lambda^2\tau_i^2}{4}\}.$$

For the random variables in the set $\{\frac{1}{\sqrt{n}}\sum_{i=1}^{n} X_{ij}\epsilon_i : j \in \mathcal{N}_i\}$, their joint distribution is a multivariate normal distribution with mean 0 and covariance matrix $\frac{\sigma^2}{n} X_{\mathcal{N}_i}^T X_{\mathcal{N}_i}$. Then, we have

$$P(\mathcal{A}^c) \leq \sum_{i=1}^{K_{G,\tau}} P(\sum_{j\in\mathcal{N}_i}(\frac{1}{\sqrt{n}}\sum_{i=1}^{n} X_{ij}\epsilon_i)^2 > \frac{n\lambda^2\tau_i^2}{4}) \leq \sum_{i=1}^{K_{G,\tau}} P(\eta_i^2\chi_{d_i}^2 > \frac{n\lambda^2\tau_i^2}{4\sigma^2})$$

If $\lambda\tau_i \geq \frac{2\sigma\eta_i}{\sqrt{n}}(d_i + Ad_i^{1/2}\log(K_{G,\tau}))^{1/2}$ for each i, then

$$P(\mathcal{A}^c) \leq \sum_{i=1}^{K_{G,\tau}} P(\chi_{d_i}^2 > d_i + Ad_i^{1/2}\log(K_{G,\tau}))$$

By Lemma 1, we have

$$P(\mathcal{A}^c) \leq \sum_{i=1}^{K_{G,\tau}} \exp\{-\frac{1}{8}\min\{Ad_i^{1/2}\log(K_{G,\tau}), A^2(\log(K_{G,\tau}))^2\}\} \leq K_{G,\tau}^{1-q},$$

where $q = \frac{1}{8}\min\{A, A^2\log(K_{G,\tau})\}$.

Let $\beta = \beta^0$ in (2.11). When event $\mathcal{A}$ holds, we have

$$\frac{1}{n}\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2^2 \leq \lambda\|\hat{\beta} - \beta^0\|_{G,\tau} + 2\lambda(\|\beta^0\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau}).$$

Thus,

$$\frac{1}{n}\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2^2 + \lambda\|\hat{\beta} - \beta^0\|_{G,\tau} \leq 2\lambda(\|\hat{\beta} - \beta^0\|_{G,\tau} + \|\beta^0\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau}). \tag{2.12}$$

Denote $S^{(1)}, S^{(2)}, \ldots, S^{(p)}$ as arbitrary optimal decomposition of β^0 and $T^{(1)}, T^{(2)}, \ldots, T^{(p)}$ as arbitrary optimal decomposition of $\hat{\beta} - \beta^0$. Then, by assumption (A2), we get

$$\|\hat{\beta} - \beta^0\|_{G,\tau} + \|\beta^0\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau} = \|\sum_{j\in J_0} T^{(j)}\|_{G,\tau} + \|\sum_{j\in J_0^c} T^{(j)}\|_{G,\tau} + \|\sum_{j\in J_0} S^{(j)}\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau}.$$

Furthermore, we have

$$\|\hat{\beta}\|_{G,\tau} = \|\sum_{j\in J_0} T^{(j)} + \sum_{j\in J_0^c} T^{(j)} + \sum_{j\in J_0} S^{(j)}\|_{G,\tau} \geq \|\sum_{j\in J_0^c} T^{(j)} + \sum_{j\in J_0} S^{(j)}\|_{G,\tau} - \|\sum_{j\in J_0} T^{(j)}\|_{G,\tau}$$
$$= \|\sum_{j\in J_0^c} T^{(j)}\|_{G,\tau} + \|\sum_{j\in J_0} S^{(j)}\|_{G,\tau} - \|\sum_{j\in J_0} T^{(j)}\|_{G,\tau}.$$

Hence, $\|\hat{\beta} - \beta^0\|_{G,\tau} + \|\beta^0\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau} \leq 2\|\sum_{j\in J_0} T^{(j)}\|_{G,\tau}$ and by (2.12), we get

$$\|\hat{\beta} - \beta^0\|_{G,\tau} \leq 2(\|\hat{\beta} - \beta^0\|_{G,\tau} + \|\beta^0\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau}) \leq 4\|\sum_{j\in J_0} T^{(j)}\|_{G,\tau}.$$

By Lemma 2, we have $\|\sum_{j\in J_0} T^{(j)}\|_{G,\tau} = \sum_{j\in J_0} \tau_j \|T^{(j)}\|_2$. Furthermore, by definition, $\|\hat{\beta} - \beta^0\|_{G,\tau} = \sum_{j\in J_0} \tau_j \|T^{(j)}\|_2 + \sum_{j\in J_0^c} \tau_j \|T^{(j)}\|_2$. Thus, we have $\sum_{j\in J_0^c} \tau_j \|T^{(j)}\|_2 \leq 3\sum_{j\in J_0} \tau_j \|T^{(j)}\|_2$.

By Assumption (A3), we get

$$\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2 \geq \sqrt{n}\kappa \sqrt{\sum_{j\in J_0} \tau_j^2 \|T^{(j)}\|_2^2} \tag{2.13}$$

Furthermore, by (2.12), we have

$$\frac{1}{n}\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2^2 \leq 2\lambda(\|\hat{\beta} - \beta^0\|_{G,\tau} + \|\beta^0\|_{G,\tau} - \|\hat{\beta}\|_{G,\tau}) \leq 4\lambda \sum_{j\in J_0} \tau_j \|T^{(j)}\|_2 \tag{2.14}$$

By (2.13), (2.14) and the fact that there is at most $K_{G,\tau}$ nonzero $T^{(j)}$'s where $j \in J_0$, we have

$$\frac{1}{n}\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2^2 \leq 4\lambda K_{G,\tau}^{1/2} \sqrt{\sum_{j\in J_0} \tau_j^2 \|T^{(j)}\|_2^2} \leq \frac{4\lambda K_{G,\tau}^{1/2}}{\sqrt{n}\kappa}\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2$$

Hence,

$$\frac{1}{n}\|\mathbf{X}(\hat{\beta} - \beta^0)\|_2^2 \leq \frac{16\lambda^2 K_{G,\tau}}{\kappa^2}.$$

Furthermore,

$$\|\hat{\beta}-\beta^0\|_2 = \|\sum_{j=1}^{p}\frac{1}{\tau_j}\tau_j T^{(j)}\|_2 \le \frac{\|\hat{\beta}-\beta^0\|_{G,\tau}}{\tau_*} \le \frac{4\|\sum_{j\in J_0}T^{(j)}\|_{G,\tau}}{\tau_*} = \frac{4\sum_{j\in J_0}\tau_j\|T^{(j)}\|_2}{\tau_*}$$
$$\le \frac{4K_{G,\tau}^{1/2}}{\sqrt{n}\kappa\tau_*}\|\mathbf{X}(\hat{\beta}-\beta^0)\|_2 \le \frac{16\lambda K_{G,\tau}}{\kappa^2\tau_*}.$$

Thus,

$$\|\hat{\beta}-\beta^0\|_{G,\tau} \le 4\|\sum_{j\in J_0}T^{(j)}\|_{G,\tau} \le \frac{16\lambda K_{G,\tau}}{\kappa^2}.\square$$

Proof of Theorem 2:

For each $u \in R^p$, define $Q_n(u) = \frac{1}{2}\|\frac{1}{\sqrt{n}}\mathbf{X}u-\epsilon\|_2^2 + n\lambda\|\beta^0+\frac{u}{\sqrt{n}}\|_{G,\tau}$. It's easy to check that

$$\hat{u} = \sqrt{n}(\hat{\beta}-\beta^0) = \arg\min_{u\in R^p} Q_n(u).$$

Furthermore, we have

$$Q_n(u)-Q_n(0) = \frac{1}{2}\|\frac{1}{\sqrt{n}}\mathbf{X}u-\epsilon\|_2^2 + n\lambda\|\beta^0+\frac{u}{\sqrt{n}}\|_{G,\tau} - \frac{1}{2}\|\epsilon\|_2^2 - n\lambda\|\beta^0\|_{G,\tau}$$
$$= \underbrace{\frac{1}{2n}u^T\mathbf{X}^T\mathbf{X}u - \frac{1}{\sqrt{n}}u^T\mathbf{X}^T\epsilon}_{I_1} + \underbrace{n\lambda(\|\beta^0+\frac{u}{\sqrt{n}}\|_{G,\tau}-\|\beta^0\|_{G,\tau})}_{I_2}.$$

By assumptions (A4) and (A5), we get

$$I_1 \xrightarrow{\mathrm{d}} \frac{1}{2}u^T\mathcal{M}u - u^TW,$$

where $W \sim N_p(0,\sigma^2\mathcal{M})$.

Without loss of generality, assume that the first $|J_0|$ elements of β^0 are nonzero and the other $p - |J_0|$ elements are zero, i.e., $\beta^0 = ((\beta^0_{J_0})^T, 0)$. Hence,

$$
\begin{aligned}
I_2 &= n\lambda\Bigg(\left\|\begin{pmatrix} \beta^0_{J_0} + \frac{1}{\sqrt{n}}u_{J_0} \\ \frac{1}{\sqrt{n}}u_{J_0^c} \end{pmatrix}\right\|_{G,\tau} - \left\|\begin{pmatrix} \beta^0_{J_0} \\ 0 \end{pmatrix}\right\|_{G,\tau}\Bigg) \\
&= \underbrace{n\lambda\Bigg(\left\|\begin{pmatrix} \beta^0_{J_0} + \frac{1}{\sqrt{n}}u_{J_0} \\ 0 \end{pmatrix}\right\|_{G,\tau} - \left\|\begin{pmatrix} \beta^0_{J_0} \\ 0 \end{pmatrix}\right\|_{G,\tau}\Bigg)}_{I_3} + \underbrace{\sqrt{n}\lambda\left\|\begin{pmatrix} 0 \\ u_{J_0^c} \end{pmatrix}\right\|_{G,\tau}}_{I_4}.
\end{aligned}
$$

Denote $V^{(1)}, V^{(2)}, \ldots, V^{(p)}$ as arbitrary optimal decomposition of u. Then, by the triangle inequality, we have

$$
|I_3| \leq n\lambda\left\|\begin{pmatrix} \frac{1}{\sqrt{n}}u_{J_0} \\ 0 \end{pmatrix}\right\|_{G,\tau} = \sqrt{n}\lambda\sum_{j\in J_0}\tau_j\|V^{(j)}\|_2.
$$

If $\sqrt{n}\lambda \to 0$ and $\tau_j = O(1)$ for each $j \in J_0$, then for each fixed u, we have $|I_3| \to 0$ as $n \to \infty$.

Furthermore, we observe that

$$
|I_4| = \sqrt{n}\lambda\sum_{j\in J_0^c}\tau_j\|V^{(j)}\|_2 = (n^{(\gamma+1)/2}\lambda)(n^{-\gamma/2}\sum_{j\in J_0^c}\tau_j\|V^{(j)}\|_2).
$$

If $n^{(\gamma+1)/2}\lambda \to \infty$ and $\liminf_{n\to\infty} n^{-\gamma/2}\tau_j > 0$ for each $j \in J_0^c$, then $|I_4| \to \infty$ as $n \to \infty$.

Hence, we get $Q_n(u) - Q_n(0) \xrightarrow{\mathrm{d}} D(u)$, where

$$
D(u) = \begin{cases} \frac{1}{2}u^T\mathcal{M}u - u^TW & \text{if } \mathrm{supp}(u) \subseteq J_0 \\ \infty & \text{else} \end{cases}
$$

Since $\hat{u} = \arg\min_{u\in R^p} Q_n(u) = \arg\min_{u\in R^p}(Q_n(u) - Q_n(0))$ and $u^* = (\mathcal{M}^{-1}_{J_0}W_{J_0}, 0)^T = \arg\min_{u\in R^p} D(u)$, by the argmax theorem ((Van Der Vaart and Wellner, 1996), Corollary

3.2.3), we have $\hat{u} \xrightarrow{\text{d}} (\mathcal{M}_{J_0}^{-1} W_{J_0}, 0)^T$. Thus,

$$\sqrt{n}(\hat{\beta}_{J_0} - \beta_{J_0}^0) \xrightarrow{\text{d}} N(0, \sigma^2 \mathcal{M}_{J_0}^{-1}),$$
$$\sqrt{n}\hat{\beta}_{J_0^c} \xrightarrow{\text{d}} 0 \text{ and therefore } \hat{\beta}_{J_0^c} \xrightarrow{\text{p}} 0. \ \square$$

Proof of Corollary 1:

Since $u^T \mathbf{X}^T \mathbf{X} u/(2n) \to u^T \mathbf{\Sigma} u$ a.s. for each fixed u and

$$\frac{\mathbf{X}^T \epsilon}{\sqrt{n}} = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} x_i \epsilon_i \xrightarrow{\text{d}} N(0, \mathbf{\Sigma}),$$

we can derive the results shown in Corollary 1 by the same proof of Theorem 2.

Proof of Theorem 3:

By Proposition 1, we know that $\hat{\beta}$ is a solution if and only if $\hat{\beta}$ can be decomposed as $\hat{\beta} = \sum_{i=1}^{p} V^{(i)}$ where $V^{(i)}$'s satisfy that, for all $1 \le i \le p$, (a) $V_{\mathcal{N}_i^c}^{(i)} = 0$; (b) either $V_{\mathcal{N}_i}^{(i)} \neq 0$ and $X_{\mathcal{N}_i}^T (Y - \mathbf{X}\hat{\beta}) = n\lambda\tau_i \frac{V_{\mathcal{N}_i}^{(i)}}{\|V_{\mathcal{N}_i}^{(i)}\|_2}$, or $V_{\mathcal{N}_i}^{(i)} = 0$ and $\|X_{\mathcal{N}_i}^T (Y - \mathbf{X}\hat{\beta})\|_2 \le n\lambda\tau_i$.

Denote $\hat{H} = \{i : \|V_{\mathcal{N}_i}^{(i)}\|_2 \neq 0\}$. Then, we have $X_{\mathcal{N}_i}^T (Y - \mathbf{X}\hat{\beta}) = n\lambda\tau_i V_{\mathcal{N}_i}^{(i)}/\|V_{\mathcal{N}_i}^{(i)}\|_2$ for each $i \in \hat{H}$ and $X_{\mathcal{N}_i}^T (Y - \mathbf{X}\hat{\beta}) = n\lambda\tau_i Z_{\mathcal{N}_i}^{(i)}$ for each $i \notin \hat{H}$, where $Z^{(i)}$ is a $p \times 1$ random vector with $\|Z_{\mathcal{N}_i}^{(i)}\|_2 \le 1$. Since some predictors may belong to multiple neighborhoods, the following conditions need to be satisfied:

(i) $\tau_{i_1} V_j^{(i_1)}/\|V_{\mathcal{N}_{i_1}}^{(i_1)}\|_2 = \tau_{i_2} V_j^{(i_2)}/\|V_{\mathcal{N}_{i_2}}^{(i_2)}\|_2$ for each $i_1 \in \hat{H}, i_2 \in \hat{H}$ and $j \in \mathcal{N}_{i_1} \cap \mathcal{N}_{i_2}$;

(ii) $\tau_{i_1} V_j^{(i_1)}/\|V_{\mathcal{N}_{i_1}}^{(i_1)}\|_2 = \tau_{i_2} Z_j^{(i_2)}$ for each $i_1 \in \hat{H}, i_2 \notin \hat{H}$ and $j \in \mathcal{N}_{i_1} \cap \mathcal{N}_{i_2}$;

(iii) $\tau_{i_1} Z_j^{(i_1)} = \tau_{i_2} Z_j^{(i_2)}$ for each $i_1 \notin \hat{H}, i_2 \notin \hat{H}$ and $j \in \mathcal{N}_{i_1} \cap \mathcal{N}_{i_2}$.

For each $1 \le i \le p$, define $\hat{f}_i = \tau_i V_i^{(i)}/\|V_{\mathcal{N}_i}^{(i)}\|_2$ if $i \in \hat{H}$ and $\hat{f}_i = \tau_i Z_i^{(i)}$ if $i \notin \hat{H}$. Then, any solution $\hat{\beta}$ satisfies the following equation

$$\mathbf{X}^T (Y - \mathbf{X}\hat{\beta}) = n\lambda \hat{f}, \text{ where } \hat{f} = (\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_p)^T. \tag{2.15}$$

Define events

$$\mathcal{A}_1 = \{\|\hat{\beta}_{J_0} - \beta^0_{J_0}\|_\infty < \beta^0_{min}\};$$
$$\mathcal{A}_2 = \{\|\hat{f}_{\mathcal{N}_j}\|_2 < \tau_j \text{ for each } j \in J_0^c\}.$$

When event $\mathcal{A}_1$ occurs, we have sign($\hat{\beta}_j$)=sign(β^0_j) for each $j \in J_0$. When event $\mathcal{A}_2$ occurs, we have $V^{(j)}_{\mathcal{N}_j} = 0$ for each $j \in J_0^c$. Furthermore, we know that $V^{(j)}_{\mathcal{N}_j^c} = 0$ for each j. Then, by condition (A2), we have $\hat{\beta}_{J_0^c} = \sum_{j \in J_0^c} V^{(j)}_{J_0^c} = 0$. Thus, if we can show that $P(\mathcal{A}_1 \cap \mathcal{A}_2) \longrightarrow 1$, then we have $P(\text{sign}(\hat{\beta})=\text{sign}(\beta^0)) \longrightarrow 1$ as $n \longrightarrow \infty$.

Note that if events $\mathcal{A}_1$ and $\mathcal{A}_2$ occur, from equation (2.15), we have

$$X^T_{J_0}(X_{J_0}\beta^0_{J_0} + \epsilon - X_{J_0}\hat{\beta}_{J_0}) = n\lambda \hat{f}_{J_0};$$
$$X^T_{\mathcal{N}_j}(X_{J_0}\beta^0_{J_0} + \epsilon - X_{J_0}\hat{\beta}_{J_0}) = n\lambda \hat{f}_{\mathcal{N}_j} \text{ for each } j \in J_0^c.$$

Thus,

$$\hat{\beta}_{J_0} - \beta^0_{J_0} = (X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon - n\lambda(X^T_{J_0}X_{J_0})^{-1}\hat{f}_{J_0}; \tag{2.16}$$
$$\frac{\hat{f}_{\mathcal{N}_j}}{\tau_j} = \frac{1}{n\lambda\tau_j}X^T_{\mathcal{N}_j}(I_n - X_{J_0}(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0})\epsilon + \frac{1}{\tau_j}X^T_{\mathcal{N}_j}X_{J_0}(X^T_{J_0}X_{J_0})^{-1}\hat{f}_{J_0}. \tag{2.17}$$

From (2.16) and condition (A8), we have

$$\begin{aligned}
\|\hat{\beta}_{J_0} - \beta^0_{J_0}\|_\infty &\le \|(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon\|_\infty + \lambda\|(X^T_{J_0}X_{J_0}/n)^{-1}\hat{f}_{J_0}\|_\infty \\
&\le \|(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon\|_\infty + \lambda\|(X^T_{J_0}X_{J_0}/n)^{-1}\|_\infty\|\hat{f}_{J_0}\|_\infty \\
&\le \|(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon\|_\infty + \frac{\lambda\sqrt{s_0}}{Q_2}\|\hat{f}_{J_0}\|_\infty \\
&\le \|(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon\|_\infty + \lambda\frac{\sqrt{s_0}}{Q_2}\max_{j \in J_0}\tau_j,
\end{aligned}$$

where we use the fact that $|\hat{f}_j| \le \tau_j$ for each j in the last inequality.

Then, by Markov inequality, we have

$$P(\|\hat{\beta}_{J_0} - \beta^0_{J_0}\|_\infty \geq \beta^0_{min}) \leq \frac{E(\|\hat{\beta}_{J_0} - \beta^0_{J_0}\|_\infty)}{\beta^0_{min}}$$
$$\leq \frac{1}{\beta^0_{min}}[E(\|(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon\|_\infty) + \lambda\frac{\sqrt{s_0}}{Q_2}\max_{j\in J_0}\tau_j].$$

Since $(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon$ follows the multivariate normal distribution with mean 0 and covariance matrix $\sigma^2(X^T_{J_0}X_{J_0})^{-1}$. Using standard results on the maximum of this Gaussian vector ((Ledoux and Talagrand, 1991)), we have

$$E(\|(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0}\epsilon\|_\infty) \leq 3\sigma\sqrt{\frac{\log s_0}{nQ_2}}.$$

Hence,

$$P(\|\hat{\beta}_{J_0} - \beta^0_{J_0}\|_\infty \geq \beta^0_{min}) \leq \frac{1}{\beta^0_{min}}[3\sigma\sqrt{\frac{\log s_0}{nQ_2}} + \lambda\frac{\sqrt{s_0}}{Q_2}\max_{j\in J_0}\tau_j] \longrightarrow 0 \text{ as } n \longrightarrow \infty.$$

Therefore,

$$P(\mathcal{A}_1) = 1 - P(\|\hat{\beta}_{J_0} - \beta^0_{J_0}\|_\infty \geq \beta^0_{min}) \longrightarrow 1, \text{ as } n \longrightarrow \infty. \tag{2.18}$$

Furthermore, from (2.17), for each $j \in J_0^c$,

$$\frac{\|\hat{f}_{\mathcal{N}_j}\|_2}{\tau_j} \leq \frac{1}{n\lambda\tau_j}\|X^T_{\mathcal{N}_j}(I_n - X_{J_0}(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0})\epsilon\|_2 + \frac{1}{\tau_j}\|X^T_{\mathcal{N}_j}X_{J_0}(X^T_{J_0}X_{J_0})^{-1}\hat{f}_{J_0}\|_2$$
$$\leq \frac{\sqrt{d_j}}{n\lambda\tau_j}\|X^T_{\mathcal{N}_j}(I_n - X_{J_0}(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0})\epsilon\|_\infty + \frac{\|\hat{f}_{J_0}\|_2}{\tau_j}\|X^T_{\mathcal{N}_j}X_{J_0}(X^T_{J_0}X_{J_0})^{-1}\|_2$$
$$\leq \frac{\sqrt{d_j}}{n\lambda\tau_j}\|X^T_{\mathcal{N}_j}(I_n - X_{J_0}(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0})\epsilon\|_\infty + \frac{\sqrt{d_j}\|X^T_{\mathcal{N}_j}X_{J_0}(X^T_{J_0}X_{J_0})^{-1}\|_\infty}{\tau_j}\sqrt{s_0}\max_{j\in J_0}\tau_j.$$

By condition (A9), for each $j \in J_0^c$, we have

$$\frac{\|\hat{f}_{\mathcal{N}_j}\|_2}{\tau_j} \leq \frac{\sqrt{d_j}}{n\lambda\tau_j}\|X^T_{\mathcal{N}_j}(I_n - X_{J_0}(X^T_{J_0}X_{J_0})^{-1}X^T_{J_0})\epsilon\|_\infty + \frac{(1-\xi)\sqrt{d_j}\sqrt{s_0}\max_{j\in J_0}\tau_j}{\tau_j}.$$

Thus,

$$\max_{j\in J_0^c}\frac{\|\hat{f}_{\mathcal{N}_j}\|_2}{\tau_j}\leq\|X_{J_0^c}^T(I_n-X_{J_0}(X_{J_0}^TX_{J_0})^{-1}X_{J_0}^T)\epsilon\|_\infty\max_{j\in J_0^c}\frac{\sqrt{d_j}}{n\lambda\tau_j}+\frac{(1-\xi)\sqrt{s_0}\max_{j\in J_0}\tau_j}{\min_{j\in J_0^c}m_j}.$$

We observe that $X_{J_0^c}^T(I_n-X_{J_0}(X_{J_0}^TX_{J_0})^{-1}X_{J_0}^T)\epsilon$ follows the multivariate normal distribution with mean 0 and covariance matrix $X_{J_0^c}^T(I_n-X_{J_0}(X_{J_0}^TX_{J_0})^{-1}X_{J_0}^T)X_{J_0^c}$. Furthermore, by condition (A7), the variance of each component is bounded by $nQ_1^2\sigma^2$. Thus, by the Markov inequality and the result on the maximum of this Gaussian vector,

$$P(\|X_{J_0^c}^T(I_n-X_{J_0}(X_{J_0}^TX_{J_0})^{-1}X_{J_0}^T)\epsilon\|_\infty\max_{j\in J_0^c}\frac{\sqrt{d_j}}{n\lambda\tau_j}>\xi)\leq\frac{3\sigma Q_1}{\lambda\xi}\sqrt{\frac{\log(p-s_0)}{n}}\max_{j\in J_0^c}\frac{\sqrt{d_j}}{\tau_j}$$
$$\longrightarrow 0 \text{ as } n\longrightarrow\infty.$$

By condition (A6), if m_j's satisfy that $\max_{j\in J_0}m_j = O_p(1)$ and $\liminf_{n\to\infty}n^{-\gamma}\min_{j\in J_0^c}m_j>0$ for some $\gamma>\delta_0$, we have

$$\frac{\sqrt{s_0}\max_{j\in J_0}\tau_j}{\min_{j\in J_0^c}m_j}\leq\frac{s_0\max_{j\in J_0}m_j}{\min_{j\in J_0^c}m_j}=\frac{s_0}{n^\gamma}\frac{\max_{j\in J_0}m_j}{n^{-\gamma}\min_{j\in J_0^c}m_j}\longrightarrow 0,\text{ as } n\longrightarrow\infty.$$

Hence,

$$P(\mathcal{A}_2)=P(\max_{j\in J_0^c}\frac{\|\hat{f}_{\mathcal{N}_j}\|_2}{\tau_j}<1)\longrightarrow 1,\text{ as } n\longrightarrow\infty. \tag{2.19}$$

By (2.18) and (2.19), we conclude that $P(\mathcal{A}_1\cap\mathcal{A}_2)\longrightarrow 1$ and $P(\text{sign}(\hat{\beta})=\text{sign}(\beta^0))\longrightarrow 1$ as $n\longrightarrow\infty$.

Proof of Theorem 4:

For each n, define $\mathbf{\Delta}=\mathbf{X}^T\mathbf{X}/n-\mathbf{\Sigma}$ and $\mathcal{B}_n=\{\max_{j,k}|\Delta_{jk}|<Q_3\sqrt{\frac{\log p}{n}}\}$. By condition (A10) and Lemma 9.3 ((Zhou et al., 2009)), we have $P(\mathcal{B}_n)\geq 1-1/p^2$.

Assume event $\mathcal{B}_n$ holds, then

$$|\frac{X_j^TX_j}{n}-\Sigma_{jj}|\leq\max_{jk}|\Delta_{jk}|<Q_3\sqrt{\frac{\log p}{n}}\leq 1/2,\text{ for } j=1,2,\ldots,p.$$

Thus, $\|X_j\|_2^2/n \leq 3/2$ and therefore $\max_{j\in J_0^c}\|X_j\|_2 \leq \sqrt{3n/2}$. Furthermore, by conditions (A11), (A12), and Lemmas 11.1 and 11.2 ((Zhou et al., 2009)), we have

$$\lambda_{min}(\frac{X_{J_0}^T X_{J_0}}{n}) \geq \Lambda_{min}(s_0) > 0, \text{ and } \|X_{J_0^c}^T X_{J_0}(X_{J_0}^T X_{J_0})^{-1}\|_\infty \leq \sqrt{\frac{3s_0}{2\Lambda_{min}(s_0)}}.$$

In addition, by condition (A1), the standard results on the maximum of Gaussian vector ((Ledoux and Talagrand, 1991)), and the assumption that $\mathbf{X}$ is independent of ϵ, we have

$$\mathrm{E}(\|(X_{J_0}^T X_{J_0})^{-1}X_{J_0}^T\epsilon\|_\infty|\mathbf{X}) \leq 3\sigma\sqrt{\frac{\log s_0}{n\Lambda_{min}(s_0)}}, \text{ and therefore}$$
$$\mathrm{E}(\|(X_{J_0}^T X_{J_0})^{-1}X_{J_0}^T\epsilon\|_\infty) \leq 3\sigma\sqrt{\frac{\log s_0}{n\Lambda_{min}(s_0)}}$$

Similarly, we can prove that

$$\mathrm{E}(\|X_{J_0^c}^T(I_n - X_{J_0}(X_{J_0}^T X_{J_0})^{-1}X_{J_0}^T)\epsilon\|_\infty) \leq 3\sigma\sqrt{\frac{3n\log(p-s_0)}{2}}.$$

Thus, follow the proof of Theorem 3, based on the above results, we can prove that

$$P(\mathcal{A}_1 \cap \mathcal{A}_2 \cap \mathcal{B}_n) \longrightarrow 1 \text{ as } n \to \infty.$$

Hence, for the random design, we also have $P(\text{sign}(\hat{\beta})=\text{sign}(\beta^0)) \longrightarrow 1$ as $n \longrightarrow \infty$.

CHAPTER 3: GRAPH GUIDED MULTI-TASK LEARNING WITH APPLICATIONS IN NEUROSCIENCE

3.1 Introduction

Alzheimer's disease (AD) is one of the most common forms of dementia characterized by progressive cognitive and memory deficits. It has been reported that one in every 85 persons in year 2050 will be likely affected by this disease ((Brookmeyer et al., 2007)). The increasing incidence of AD makes this disease a very important health issue and also huge financial burden for both patients and governments (Hebert et al., 2001; Bain et al., 2008). Thus, it is very important to develop methods for timely diagnosis of AD and its predromal stage, i.e., mild cognitive impairment (MCI). Over the last decade, many machine learning methods have been used for early diagnosis of AD and MCI based on different modalities of biomarkers, e.g., structural brain atrophy delineated by structural magnetic resonance imaging (MRI) (Du et al., 2007; McEvoy et al., 2009; Fjell et al., 2010; Yu et al., 2014), metabolic alterations characterized by fluorodeoxyglucose positron emission tomography (FDG-PET) (De Santi et al., 2001; Morris et al., 2001), and pathological amyloid depositions measured by CerebroSpinal Fluid (CSF) (Bouwman et al., 2007; Fjell et al., 2010). Typically, these methods learn a binary classification model from training data and use this model to predict disease status (i.e., class label) of the testing subjects.

Besides classification of disease status, accurate prediction of clinical scores such as Mini Mental State Examination (MMSE) score and Alzheimer's Disease Assessment Scale-Cognitive Subscale (ADAS-Cog) is also important and useful since they can help evaluate the stage of AD pathology and predict future progression. Specifically, as a brief 30-point questionnaire test, MMSE is commonly used to screen for cognitive impairment. It can be used to examine a patient's arithmetic, memory and orientation ((Folstein et al., 1975)). As another important clinical score of AD, ADAS-Cog is a cognitive testing instrument widely used in clinical trials. It is designed to measure the severity of the most important

symptoms of AD ((Rosen et al., 1984)). Several studies based on regression methods have been conducted to estimate MMSE and ADAS-Cog using the extracted features from MRI and FDG-PET. For example, (Duchesne et al., 2005) used linear regression models, (Wang et al., 2010) developed a high-dimensional kernel-based regression method, and (Cheng et al., 2013) proposed a semi-supervised multi-modal relevance vector regression method. However, almost all of these regression methods model different clinical scores separately and do not use the class label information which is often available in practice.

Although the classification of disease status and the prediction of clinical scores are different tasks, there exists inherent correlation among them since the underlying pathology is the same (Fan et al., 2010; Stonnington et al., 2010). In the literature, (Zhang and Shen, 2012) proposed multi-modal multi-task (M3T) learning to predict both class label and clinical scores jointly. M3T formulates the estimations of class label and clinical scores as different tasks. The $l_{2,1}$ penalty is used to deliver sparse models with a common feature subset for each task. Their experimental results indicate that selecting a common feature subset for different correlated tasks could achieve better prediction of both class label and clinical scores than choosing the feature subset for each task separately. Although benefiting from using the commonality among different correlated tasks, M3T method does not incorporate the correlation information among features. Actually, many features extracted from brain images such as structural MRI are statistically correlated significantly. In this case, feature selection combined with the additional correlation information among features can improve classification/regression performance ((Yang et al., 2012)).

As shown in Chapter 2, we extract effective correlation information among features by constructing a sparse undirected feature graph. This undirected graph uses all features as nodes. Also, two features are connected by an edge in the graph if there is statistically significant partial correlation between them. In practice, we can use many existing high-dimensional precision matrix estimation methods (Friedman et al., 2008; Cai et al., 2011) to construct this undirected graph. Based on this undirected feature graph, we propose a new Graph Guided Multi-task Learning (GGML) method to predict both class label and clinical scores simultaneously. Specifically, we utilize a new latent group Lasso penalty to encourage the significantly-correlated features to be in or out of the models together. This

new penalty also encourages the intrinsic correlated tasks to share a common feature subset. It is very useful for us to acquire robust and accurate feature selection. Computationally, the optimization problem for our proposed GGML method can be solved by the traditional group Lasso algorithm very efficiently ((Yuan and Lin, 2006)). Theoretically, our proposed GGML method includes M3T method as a special case. To validate our proposed GGML method, we have conducted many experiments on the Alzheimer's Disease Neuroimaging Initiative (ADNI) (www.loni.ucla.edu/ADNI) data set. Compared with the other methods, our proposed GGML method acquires very promising results.

The remainder of this chapter is organized as follows. In the Materials section, we introduce the ADNI dataset used in this study. In the Method section, we show how to extract useful correlation information among features and describe our proposed new method. In Sections 3.4 and 3.5, we compare our method with the other methods by simulation study and also the analysis of the ADNI dataset. In the Discussion section, we discuss some possible extensions of our proposed method. Finally, we conclude this chapter in the Conclusion section.

3.2 Materials

3.2.1 Data

Data used in this chapter were obtained from the ADNI database (http://adni.loni.ucla.edu/). As a $60 million, 5-year public-private partnership, the ADNI was launched in 2003 by the National Institute on Aging (NIA), the National Institute of Biomedical Imaging and Bioengineering (NIBIB), the Food and Drug Administration (FDA), private pharmaceutical companies and non-profit organizations. The main goal of ADNI is to test whether serial MRI, PET, other biological markers, and clinical and neuropsychological assessments can be combined to measure the progression of MCI and early AD. To that end, 800 adults with age between 55 and 90 were recruited from over 50 sites across the U.S. and Canada. Approximately, 200 cognitively normal controls and 400 MCI individuals were followed for 3 years and 200 individuals with early AD were followed for 2 years (see www.adni-info.org for up-to-date information).

The general inclusion/exclusion criteria are: 1) each mild AD individual has an MMSE score between 20 and 26, a Clinical Dementia Rating (CDR) of 0.5 or 1.0, and meets the National Institute of Neurological and Communicative Disorders and Stroke and the Alzheimer's Disease and Related Disorders Association (NINCDS/ADRDA) criteria for probable AD; 2) each MCI individual has an MMSE score between 24 and 30, a CDR of 0.5, with a memory complaint, objective memory loss measured by education adjusted scores on Wechsler Memory Scale Logical Memory II, absence of significant levels of impairment in other cognitive domains, essentially preserved activities of daily living, and an absence of dementia; 3) Each Normal Control (NC) individual is non-depressed, non-MCI, non-demented, and has a CDR of 0. The MMSE score of each NC individual is between 24 and 30.

We use data from 199 subjects who have complete baseline MRI, FDG-PET, and CSF data. These 199 subjects include 50 AD subjects, 97 MCI subjects, and 52 NC subjects. The detailed demographic information about these 199 subjects is summarized in Table 3.1.

Table 3.1: Demographic information of the 199 subjects used in this study.

Characteristics	AD (50 subjects)	MCI (97 subjects)	NC (52 subjects)
Gender (F/M)	17/33	32/65	18/34
Age (mean±sd)	75.2±7.6	75.3±7.0	75.1±5.1
Education (mean±sd)	14.7±3.7	15.9±2.9	15.8±3.2
MMSE (mean±sd)	23.7±1.9	27.1±1.7	29.0±1.2
ADAS (mean±sd)	18.5±5.9	11.4±4.4	7.36±3.2

3.2.2 Data Preprocessing

Imaging preprocessing is performed for MRI and PET. For MRI, the preprocessing steps include anterior commissure (AC) -posterior commissure (PC) correction, intensity inhomogeneity correction ((Sled et al., 1998)), skull stripping ((Wang et al., 2011)), cerebellum removal based on registration with atlas, spatial segmentation ((Zhang et al., 2001))

and registration ((Shen and Davatzikos, 2002)). After registration, we obtained the subject-labeled image based on the Jacob template ((Kabani et al., 1998)) with 93 manually labeled ROIs. For each of the 93 ROI regions in the labeled MRI, we computed the volume of gray matter as a feature. For each PET image, we first aligned the PET image to its respective MRI using affine registration. Then, we got the skull-stripping image using the corresponding brain mask of MRI and computed the average intensity of every ROI region in the PET image as a feature.

Besides MRI and PET, the CSF data were collected in the morning after an overnight fast using a 20- or 24-gauge spinal needle, frozen within 1 hour of collection, and transported on dry ice to the ADNI Biomarker Core laboratory at the University of Pennsylvania Medical Center. In this study, we use CSF Aβ42, CSF t-tau and CSF p-tau as features.

Therefore, for each subject, we finally obtained 93 features from MRI, 93 features from PET, and three features from CSF. We also have the class label, MMSE and ADAS-Cog scores for each subject.

3.3 Method

In this section, after introducing some notations, we will first discuss how to extract the correlation information among features. Next, as an extension of the SRIG method introduced in Chapter 2, our proposed graph guided multi-task learning method will be described.

3.3.1 Notation

For a set $\mathcal{A}$, we denote $|\mathcal{A}|$ as the number of elements in $\mathcal{A}$. For a matrix $\mathbf{B}$, we denote $\mathbf{B}^T$ and $\mathbf{B}^{-1}$ as the transpose and the inverse of matrix $\mathbf{B}$, respectively. We also denote $\|\mathbf{B}\|_F = \sqrt{\sum_i \sum_j \mathbf{B}_{ij}^2}$ as the Frobenius norm.

Suppose we have n samples and p features. Let $\mathbf{X} = (X_1, X_2, \ldots, X_p) = (x_1, x_2, \ldots, x_n)^T$ denote the $n \times p$ training data matrix of features, where $x_1, x_2, \ldots, x_n$ are i.i.d. samples generated from a p-dimensional multivariate distribution with mean vector $0_{p\times 1}$ and covariance matrix $\mathbf{\Sigma} = (\sigma_{ij})_{i,j=1}^p$. Also, let $\mathbf{\Omega} = (\omega_{ij})_{i,j=1}^p = \mathbf{\Sigma}^{-1}$

denote the precision matrix. Furthermore, suppose we have q response variables. Let $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_q) = (y_1, y_2, \ldots, y_n)^T$ denote the $n \times q$ training data matrix of response variables, where the response variables can be binary (for classification) or continuous (for regression). Note that, for the ADNI dataset used in our study, we have three response variables, which are class label, MMSE score, and ADAS-Cog score. The class labels are coded as $+1$ and -1 for the binary classification problem considered in this chapter.

3.3.2 Extract the correlation information among features

The correlation information is often measured by the Pearson correlation between each pair of features. We can use sample Pearson correlation coefficients to identify the statistically significant correlated features. One issue with this method is that it only estimates the marginal linear dependence between a pair of features without considering the influence of other features and common driving influences. Such issue can be overcome by using partial correlation which measures the linear dependence between each pair of features after eliminating the linear effect of the other features. In practice, we can compute the sample partial correlation coefficient between features i and j, denoted as $\hat{\rho}^*_{ij}$, which is defined as the sample Pearson correlation coefficient between the residuals R_i and R_j resulting from the linear regression of feature X_i with features $\{X_k : k \neq i, j\}$ and of feature X_j with features $\{X_k : k \neq i, j\}$, respectively. The resulting $\hat{\rho}^*_{ij}$'s can be further thresholded to identify features which are partially correlated statistically significantly.

When the number of features p is small and the sample size n is big enough (bigger than p), it is easy to get good estimates of partial correlation coefficients. In this case, many previous studies (Hampson et al., 2002; Lee et al., 2011) have used partial correlations to identify the statistically significant correlated features. However, in the high dimensional case with the number of features p bigger than the sample size n, the conventional methods for estimating partial correlation may result in over-fitting of the data ((Ryali et al., 2012)). In this case, it is difficult to get accurate estimates of partial correlation coefficients.

For our proposed method introduced in the next section, in order to incorporate the correlation information among features, instead of requiring accurate estimation of ρ^*_{ij}'s, we only need to estimate which pairs of features are partially correlated, i.e., estimate the set

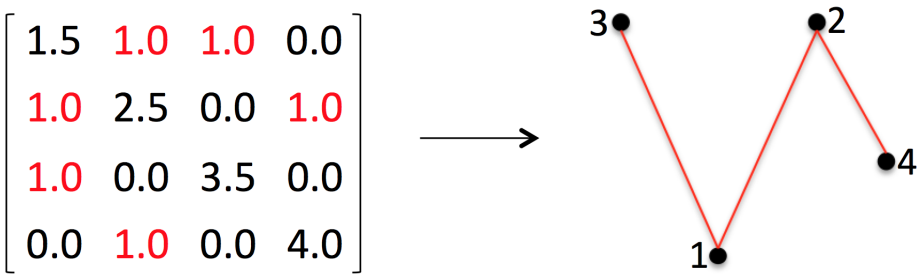


Figure 3.1: Transforming a precision matrix $\hat{\mathbf{\Omega}}$ into an undirected graph $\mathbf{G}$.

$\mathcal{E} = \{(i,j) : i < j \text{ and } \rho^*_{ij} \neq 0\}$. It is well known that the partial correlation coefficients are proportional to the off-diagonal entries of the precision matrix $\mathbf{\Omega}$ ((Meinshausen and Bühlmann, 2006)). Thus, estimating $\mathcal{E}$ is equivalent to estimating the set $\{(i,j) : i < j \text{ and } \omega_{ij} \neq 0\}$. In this way, many existing methods (Meinshausen and Bühlmann, 2006; Friedman et al., 2008; Cai et al., 2011) can be used to estimate $\mathcal{E}$ effectively.

To extract the correlation information among features, we will use the graphical Lasso ((Friedman et al., 2008)) or the neighborhood selection method ((Meinshausen and Bühlmann, 2006)) to estimate $\mathcal{E}$ and denote its estimate as $\hat{\mathcal{E}}$. Furthermore, we represent $\hat{\mathcal{E}}$ as a sparse undirected graph $\mathbf{G}$ with p nodes and $|\hat{\mathcal{E}}|$ edges, where each node represents one feature and each edge indicates that two involved features are partially correlated significantly. Figure 3.1 shows an example on how to transform the estimated precision matrix $\hat{\mathbf{\Omega}}$ into the estimated undirected graph $\mathbf{G}$. In graph $\mathbf{G}$, features i and j are connected if and only if $\hat{\omega}_{ij} \neq 0$.

3.3.3 Graph Guided Multi-task Learning (GGML) method

In this section, we assume that the sparse undirected feature graph $\mathbf{G}$ has been constructed. For each $i = 1, 2, \ldots, p$, denote $\mathcal{N}_i$ as the set including the i-th feature and its neighbors in the feature graph $\mathbf{G}$, i.e, $\mathcal{N}_i = \{j : \hat{\omega}_{ji} \neq 0\}$.

To use the correlation information represented by $\mathbf{G}$, we generalize the idea of SRIG shown in Chapter 2 to multi-task learning. Without loss of generality, considering the t-th

task, we want to use the following linear model to predict the response variable Y_t,

$$Y_t = \mathbf{X}B_t + \epsilon_t, \tag{3.1}$$

where $B_t = (b_{1t}, \ldots, b_{pt})^T \in R^p$ is the coefficient vector of interest and $\epsilon_t = (\epsilon_{1t}, \ldots, \epsilon_{nt}) \in R^n$ is the error vector with $\mathrm{E}(\epsilon_{st}) = 0$ and $\mathrm{Var}(\epsilon_{st}) = \sigma_t^2$ for each $1 \leq s \leq n$.

Suppose the feature matrix $\mathbf{X}$ is independent of the error vector ϵ_t. To use the correlation information among features, the SRIG method proposed in Chapter 2 estimates B_t by solving the following optimization problem:

$$\min_{B_t, V^{1t}, V^{2t}, \ldots, V^{pt} \in R^p} \|Y_t - \mathbf{X}B_t\|_2^2 + \lambda \sum_{i=1}^{p} \tau_{it} \|V^{it}\|_2, \tag{3.2}$$

subject to $B_t = \sum_{i=1}^{p} V^{it}$ and $\mathrm{supp}(V^{it}) \subseteq \mathcal{N}_i$ for each $1 \leq i \leq p$, where $\mathrm{supp}(V^{it})$ is the index set of nonzero components of the vector V^{it}.

In the optimization problem (3.2), τ_{it} is a positive weight for the i-th part and t-th task. Similar with the methods for adaptive Lasso ((Zou, 2006)) and group Lasso ((Yuan and Lin, 2006)), we can set $\tau_{it} = \frac{\sqrt{|\mathcal{N}_i|}}{|\tilde{b}_{it}|^\gamma}$ where γ is a positive parameter and $\tilde{b}_{it}$ is an initial estimate of b_{it}. In our experiments, we choose $\tilde{b}_{it}$ as the sample correlation coefficient between X_i and Y_t. Both the positive parameter γ and the tuning parameter λ are chosen by cross validation. Our experimental results indicate that this method could acquire good performance in general.

Theoretically, the SRIG method is very general and covers many popular methods as special cases. For example, if we ignore the correlation information among features, we can set the undirected graph $\mathbf{G}$ as an empty graph with no edge. In this case, if setting constant weights τ_{it}'s, we can show that $\sum_{i=1}^{p} \tau_{it} \|V^{it}\|_2 \propto |B_t|_1$, and the SRIG method is the same as the Lasso method ((Tibshirani, 1996)). In general, we can estimate a sparse undirected graph $\mathbf{G}$ for modeling the significant partial correlation information among features. The SRIG method can utilize this correlation information effectively and acquires good prediction performance. More theoretical properties of SRIG method are shown in Chapter 2.

For the multi-task learning, we aim at estimating q response variables simultaneously. The multivariate regression model (1.2) shown in Chapter 1 is considered here. Similar to the SRIG method discussed in Chapter 2, for each task, we assume that the coefficient vector B_t can be decomposed as $B_t = \sum_{i=1}^{p} V^{it}$, where each V^{it} is a p-dimensional latent vector satisfying $\text{supp}(V^{it}) \subseteq \mathcal{N}_i$. Furthermore, in order to make use of the intrinsic correlation among these q tasks (response variables), we also assume that the decompositions of q coefficient vectors $B_1, B_2, \ldots, B_q$ have the same pattern, i.e., $\text{supp}(V^{i1}) = \text{supp}(V^{i2}) = \cdots = \text{supp}(V^{iq})$ for each $1 \leq i \leq p$. That is, for each $i = 1, 2, \ldots, p$, we assume that, if both the i-th feature and its partially-correlated features are useful for the prediction of one response variable, they are also useful for the prediction of the other response variables.

Based on the above assumption, denote $\mathbf{B} = (B_1, B_2, \ldots, B_q) \in R^{p \times q}$ and $\mathbf{V}^i = (V^{i1}, V^{i2}, \ldots, V^{iq}) \in R^{p \times q}$ for each $1 \leq i \leq p$, we generalize the SRIG method to the following Graph Guided Multi-task Learning (GGML) method:

$$\min_{\mathbf{B}, \mathbf{V}^1, \mathbf{V}^2, \ldots, \mathbf{V}^p \in R^{p \times q}} \|\mathbf{Y} - \mathbf{X}\mathbf{B}\|_F^2 + \lambda \sum_{i=1}^{p} \tau_i \|\mathbf{V}^i\|_F, \tag{3.3}$$

subject to $\mathbf{B} = \sum_{i=1}^{p} \mathbf{V}^i$ and $\{j : \|\mathbf{V}^i_{j\cdot}\|_2 \neq 0\} \subseteq \mathcal{N}_i$ for each $1 \leq i \leq p$, where $\mathbf{V}^i_{j\cdot}$ is the jth row of matrix $\mathbf{V}^i$.

Similar to the SRIG method, we can set the weight $\tau_i = \frac{\sqrt{|\mathcal{N}_i|}}{\max_{1 \leq t \leq q} |\tilde{b}_{it}|^\gamma}$. The cross validation method can be used to choose the best γ and the best tuning parameter λ for different tasks separately. Note that the penalty term in (3.3) along with the additional constraints not only encourage the significantly partially-correlated features to be in or out of the model jointly, but also choose a common feature subset for different tasks. Due to the use of both the correlation information among features and the intrinsic commonality among different related tasks, our proposed GGML method could acquire better prediction performance than the methods not using or only using part of these two kinds of information.

As an interesting remark, we note that the M3T method ((Zhang and Shen, 2012)) is a special case of our proposed GGML method. In particular, when we ignore the correlation information among features, we can set the undirected graph $\mathbf{G}$ as an empty graph with no edge. In this case, if setting constant weights τ_i's, we can show that $\sum_{i=1}^{p} \tau_i \|\mathbf{V}^i\|_F \propto$

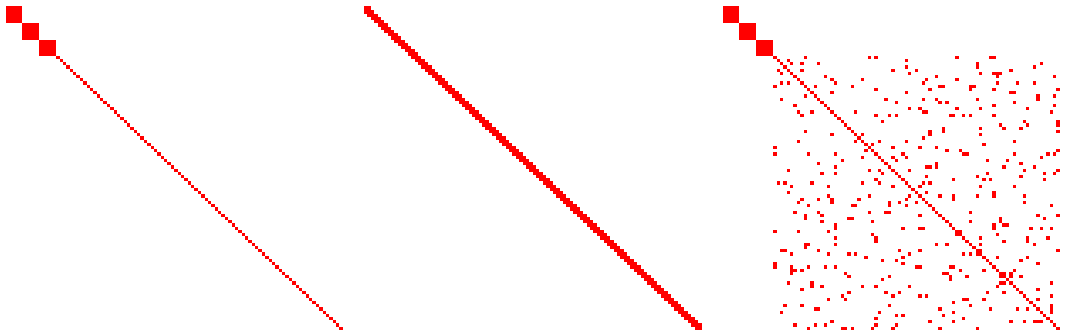


Figure 3.2: Binary maps of the true precision matrices corresponding to these three simulated examples: Left (Example 1), Middle (Example 2), and Right (Example 3).

$\sum_{i=1}^{p} \|\mathbf{B}_{i\cdot}\|_2$, where $\mathbf{B}_{i\cdot}$ is the i-th row of the coefficient matrix $\mathbf{B}$. Thus, our proposed GGML method is exactly the same as the M3T method using the $l_{2,1}$ penalty.

3.3.4 Computation

For our proposed GGML method, we need to solve the optimization problem (3.3). We can transform this constrained optimization problem into a simple unconstrained optimization problem by feature duplication.

Denote $\mathbf{X}_{\cdot\mathcal{N}_i}$ as the sub-matrix of $\mathbf{X}$ with column indices in $\mathcal{N}_i$, and denote $\mathbf{V}^i_{\mathcal{N}_i\cdot}$ as the sub-matrix of $\mathbf{V}^i$ with row indices in $\mathcal{N}_i$. Furthermore, denote $\tilde{\mathbf{X}} = (\mathbf{X}_{\cdot\mathcal{N}_1}, \mathbf{X}_{\cdot\mathcal{N}_2}, \ldots, \mathbf{X}_{\cdot\mathcal{N}_p}) \in R^{n\times(\sum_{i=1}^{p}|\mathcal{N}_i|)}$ as the duplicated feature matrix and $\tilde{\mathbf{V}} = ((\mathbf{V}^1_{\mathcal{N}_1\cdot})^T, (\mathbf{V}^2_{\mathcal{N}_2\cdot})^T, \ldots, (\mathbf{V}^p_{\mathcal{N}_p\cdot})^T)^T$ as the $(\sum_{i=1}^{p}|\mathcal{N}_i|) \times q$ coefficient matrix. Then, we can check that $\mathbf{XB} = \tilde{\mathbf{X}}\tilde{\mathbf{V}}$ and (3.3) is equivalent to the following unconstrained optimization problem:

$$\min_{\tilde{\mathbf{V}}} \|\mathbf{Y} - \tilde{\mathbf{X}}\tilde{\mathbf{V}}\|_F^2 + \lambda \sum_{i=1}^{p} \tau_i \|\mathbf{V}^i_{\mathcal{N}_i\cdot}\|_F, \tag{3.4}$$

The above problem (3.4) is a traditional group Lasso problem which can be solved efficiently by the blockwise majorization decent algorithm ((Yang and Zou, 2013)). Denote the estimate of $\mathbf{B}$ as $\hat{\mathbf{B}}$. In the application stage, given a testing subject x^*, for the t-th task, we can estimate Y_t^* by $\hat{Y}_t^* = \text{sign}(\hat{B}_t^T x^*)$ if Y_t^* is a class label and by $\hat{Y}_t^* = \hat{B}_t^T x^*$ if Y_t^* is a continuous response variable.

3.4 Simulation Study

In this section, we perform numerical studies using simulated examples. For each example, we compare our proposed GGML method with 1) the Lasso method which learns different tasks separately, 2) the SRIG method which uses the correlation information among features and learns different tasks separately, and 3) M3T method which learns different tasks jointly while ignoring the correlation information among features. We implement Lasso, SRIG, and M3T methods as shown in Section 3.3 to predict the response variables.

Similar to the measures used in (Zhang and Shen, 2012), the classification accuracy and the Pearson's correlation coefficient (CC) are also used here to evaluate the classification and regression performances, respectively. In addition, we also use the root-mean-square error (RMSE) to evaluate the regression performance.

3.4.1 Simulated examples

We study three simulated examples. Each example has one classification task and two regression tasks. We set $p = 100$, $B_1 = (2, \ldots, 2, 0, 0, \ldots, 0)^T$, $B_2 = B_3 = (1, \ldots, 1, 0, 0, \ldots, 0)^T$, where only the first 15 elements of each B_t $(t = 1, 2, 3)$ are nonzero. For each t, the errors $\epsilon_{1t}, \epsilon_{2t}, \ldots, \epsilon_{nt} \overset{i.i.d.}{\sim} N(0, 9)$. For $s = 1, 2, \ldots, n$, the feature vector $(x_{s1}, x_{s2}, \ldots, x_{sp})^T$ is generated as follows.

Example 1: For $1 \leq j \leq 5$, $x_{sj} = z_1 + 0.4\epsilon_j^x$. For $6 \leq j \leq 10$, $x_{sj} = z_2 + 0.4\epsilon_j^x$. For $11 \leq j \leq 15$, $x_{sj} = z_3 + 0.4\epsilon_j^x$. For $16 \leq j \leq p$, $x_{sj} \overset{i.i.d}{\sim} N(0, 1)$. Here, $z_1, z_2, z_3, \epsilon_1^x, \epsilon_2^x, \ldots, \epsilon_{15}^x \overset{i.i.d}{\sim} N(0, 1)$.

Example 2: The features $(x_{s1}, x_{s2}, \ldots, x_{sp})^T \sim N(0, \mathbf{\Sigma})$ with $\sigma_{ij} = 0.5^{|i-j|}$. For this example, we have $\omega_{ii} = 1.333$, $\omega_{ij} = -0.667$ if $|i - j| = 1$ and $\omega_{ij} = 0$ if $|i - j| > 1$.

Example 3: The features $\{x_{sj} : 1 \leq j \leq 15\}$ are generated from the same model as shown in Example 1. In addition, the features $(x_{s16}, x_{s17}, \ldots, x_{sp}) \sim N(0, \tilde{\mathbf{\Omega}}^{-1})$, where $\tilde{\mathbf{\Omega}} = \mathbf{M} + \delta\mathbf{I}$. Each off-diagonal entry in $\mathbf{M}$ is generated independently and equals 0.5 with probability 0.05 or 0 with probability 0.95. The diagonal entry of

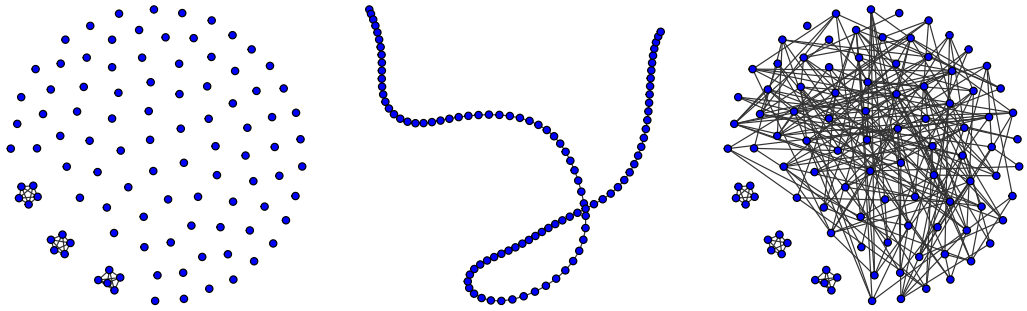


Figure 3.3: True feature graphs corresponding to these three simulated examples: Left (Example 1), Middle (Example 2), and Right (Example 3). Each blue dot indicates a feature.

$\mathbf{M}$ is 0. Here, δ is chosen such that the conditional number of $\tilde{\mathbf{\Omega}}$ is equal to $p - 15$. Finally, $\tilde{\mathbf{\Omega}}$ is standardized to have unit diagonals.

After generating each column of the response matrix $\mathbf{Y}$ by model (1.2), we replace the elements in the first column of $\mathbf{Y}$ by their signs (positive or negative) to simulate class labels. For all examples, we generate 40 training samples, 40 validation samples, and 400 testing samples. All the models are fitted on the training data. The validation data are used to choose the tuning parameters and the testing data are used to evaluate different methods. For each example, we repeat the simulation 30 times.

Figure 3.2 shows the binary maps of the true precision matrices and Figure 3.3 shows the corresponding feature graphs of these three examples. All these three graphs are sparse. For Examples 1 and 3, useful features (i.e., features with nonzero regression coefficients) are only connected with useful features. For Example 2, one useful feature is connected with one useless feature. In addition, for each example, different tasks are highly correlated since they share the same useful features. It is very interesting to study whether correlation information among features represented by the feature graph and the correlation information among tasks can be incorporated to improve the prediction performance.

3.4.2 Simulation results

Table 3.2 shows the comparison of different methods using these three simulated examples. As shown in Table 3.2, for all these three examples, the SRIG method and GGML

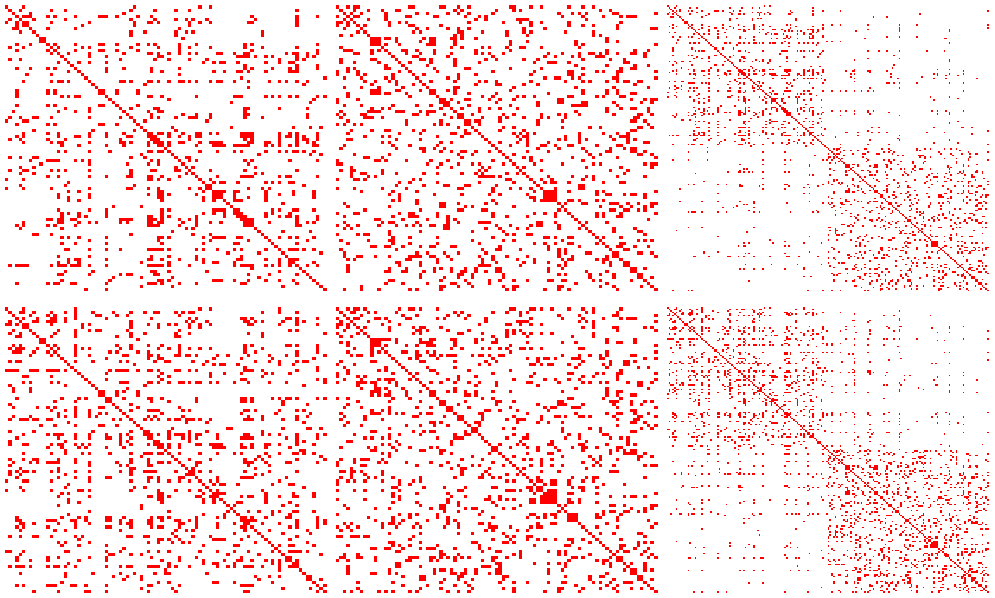


Figure 3.4: Binary maps of the estimated precision matrices. First row: AD/NC data; Second row: MCI/NC data. First column: use only MRI features; Second column: use only PET features; Third column: use both MRI and PET features.

method acquire better performance than the Lasso method and the M3T method, respectively. This indicates that the extracted partial correlation information from features can be utilized to improve the prediction performance. In addition, the GGML method and M3T method also acquire better performance than the SRIG method and the Lasso method, respectively. It indicates that learning different correlated tasks jointly can also improve the prediction performance. For these three simulated examples, since our proposed GGML method incorporates both the partial correlation information among features and the intrinsic correlation information among different related tasks, it delivers the best performance in all cases. In the next section, we will further compare these four methods using the ADNI dataset.

3.5 Analysis of the ADNI dataset

For the ADNI dataset, we estimate one class label and two clinical scores (i.e., MMSE and ADAS-Cog) using the MRI, FDG-PET and/or CSF features. Since there are two binary classification problems (AD vs. NC, and MCI vs. NC), we perform two sets of

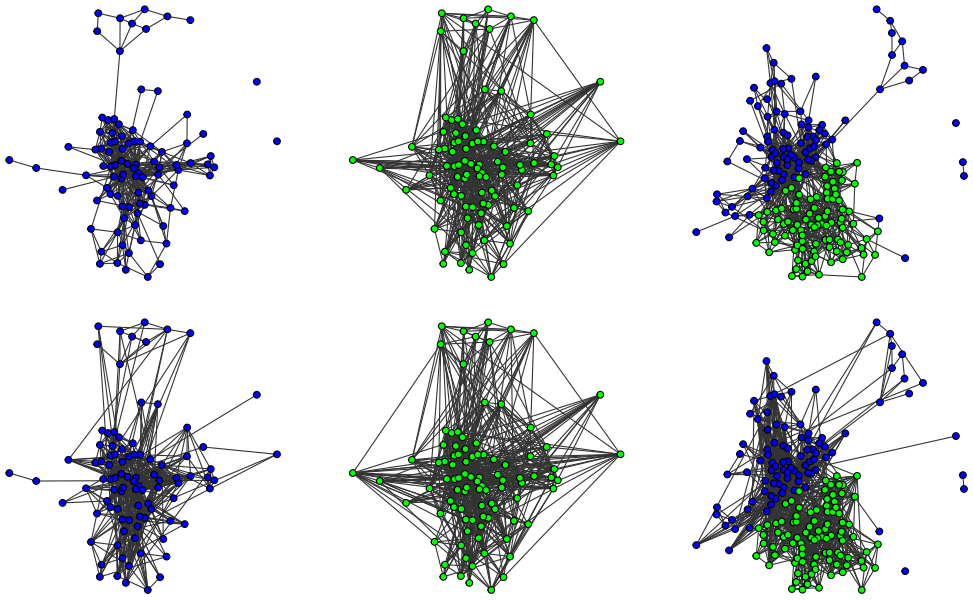


Figure 3.5: Feature graphs corresponding to the estimated precision matrices. First row: AD/NC data; Second row: MCI/NC data. First column: use only MRI features; Second column: use only PET features; Third column: use both MRI and PET features. Each blue dot represents a MRI feature and each green dot represents a PET feature.

experiments. The first set of experiments uses the AD/NC dataset including only AD and NC subjects. The second set of experiments uses the MCI/NC dataset including only MCI and NC subjects. For each set of experiments, we consider four cases: (I) use only MRI features; (II) use only PET features; (III) use both MRI and PET features (denoted as MRI+PET); (IV) use all MRI, PET and CSF features (denoted as MRI+PET+CSF).

To evaluate the performance of different methods, we used the 10-fold cross validation (CV) strategy. Specifically, the whole samples were partitioned randomly into ten subsets. Each time only nine subsets were chosen for training and the remaining one was used for testing. We repeated this process ten times with each of the 10 subsets used exactly once as the testing data. Furthermore, in consideration of possible bias due to the random partition in the 10-fold CV, we repeated the whole 10-CV process 30 times. In the training process, each column of the training data was normalized to have mean 0 and standard deviation 1. For all methods, we performed another inner 5-fold CV on the training data to choose the tuning parameters.

3.5.1 Partial correlation among different features

In the first step of the SRIG and GGML methods, we need to extract the effective correlation information from features. Note that, only the training data matrix of features were used to estimate the sparse undirected graph $\mathbf{G}$ representing the significant partial correlation among features. Figure 3.4 shows the binary maps of the estimated precision matrices. Binary maps in the first two columns indicate that many features within the same modality (e.g., MRI or PET) are partially correlated statistically significantly. However, as shown by the binary maps in the third column, the partial correlation between MRI features and PET features are not statistically significantly in most cases. Furthermore, the comparison between the binary maps in the first row and the second row indicates that the partial correlation information extracted from AD/NC data is similar to that of MCI/NC data. Similar to the example shown in Figure 3.1, we can transform the estimated precision matrices to some undirected graphs. The feature graphs corresponding to the estimated precision matrices are shown in Figure 3.5. This graph information will be used in the GGML and SRIG methods.

3.5.2 Classification results

The classification accuracies of different methods are shown in Table 3.3. All methods deliver higher classification accuracy for the AD/NC dataset than the corresponding classification accuracy for the MCI/NC dataset. For the AD/NC dataset, when we use only MRI features or PET features, the SRIG method and GGML method acquire better classification performance than the Lasso method and the M3T method, respectively. This indicates that the extracted partial correlation information from features can be utilized to improve the classification performance. In addition, when we use both MRI and PET features or all the MRI, PET, and CSF features, since it is relatively easy to discriminate AD subjects from NC subjects in this case, all four methods acquire similar high classification accuracies.

For the MCI/NC dataset, on the one hand, the comparison between SRIG and Lasso (or GGML and M3T) indicates that using the extracted partial correlation information

among features improve the classification performance significantly. On the other hand, the comparison between GGML and SRIG (or M3T and Lasso) shows that the joint classification and regression could provide better classification performance than the separate classification. Since our proposed GGML method incorporates both the partial correlation information among features and the intrinsic correlation information among different related tasks, it delivers the best classification performance.

3.5.3 Regression results

For regression tasks, we need to predict both the MMSE score and the ADAS-Cog score. Tables 3.4 and 3.5 show the comparison of regression performance on the AD/NC data and the MCI/NC data, respectively. As shown in Tables 3.4 and 3.5, our proposed GGML method acquires promising performance in most cases. For example, when we use all the features to predict the MMSE score, for the AD/NC data, our proposed GGML method achieves the highest correlation coefficient 0.745 while the corresponding correlation coefficients for Lasso, SRIG, and M3T are 0.709, 0.723 and 0.724, respectively. For the MCI/NC data, GGML also has the best performance with correlation coefficient 0.382 while the corresponding correlation coefficients for Lasso, SRIG, and M3T are 0.303, 0.325 and 0.364, respectively. In addition, when we use all the features to predict the ADAS-Cog scores, for the AD/NC data, our proposed GGML method achieves the highest correlation coefficient 0.740 while the corresponding correlation coefficients for Lasso, SRIG, and M3T are 0.664, 0.719 and 0.718, respectively. For the MCI/NC data, GGML also has the best performance with correlation coefficient 0.472 while the corresponding correlation coefficients for Lasso, SRIG, and M3T are 0.336, 0.464 and 0.426, respectively.

It is interesting to note that for the MCI/NC dataset, the PET and CSF data seem to be not useful for the prediction of MMSE score. All four methods acquire poor prediction of the MMSE scores when only the PET data are used. In addition, compared with the cases only using MRI data, both M3T and GGML methods acquire worse performance when the additional PET/CSF data are used. Similar to the previous discussion about classification

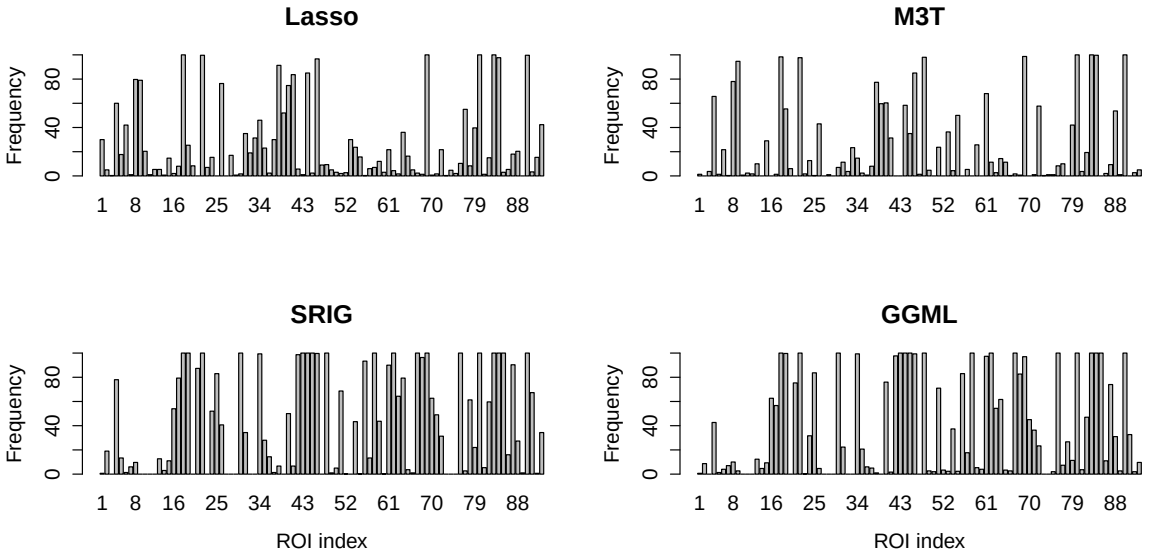


Figure 3.6: Selection frequency of 93 ROIs for the AD/NC classification task.

performance, the comparison between SRIG and Lasso (or GGML and M3T) indicates that using the extracted partial correlation information among features improves the prediction of MMSE and ADAS-Cog scores significantly. In addition, the comparison between GGML and SRIG (or M3T and Lasso) shows that joint classification and regression could deliver better prediction performance than the separate regression of MMSE (or ADAS-Cog) on the features. Since our GGML method incorporates both the partial correlation information among features and the intrinsic correlation information among different tasks, it delivers the best prediction of the MMSE and ADAS-Cog scores.

3.5.4 Most discriminative brain regions

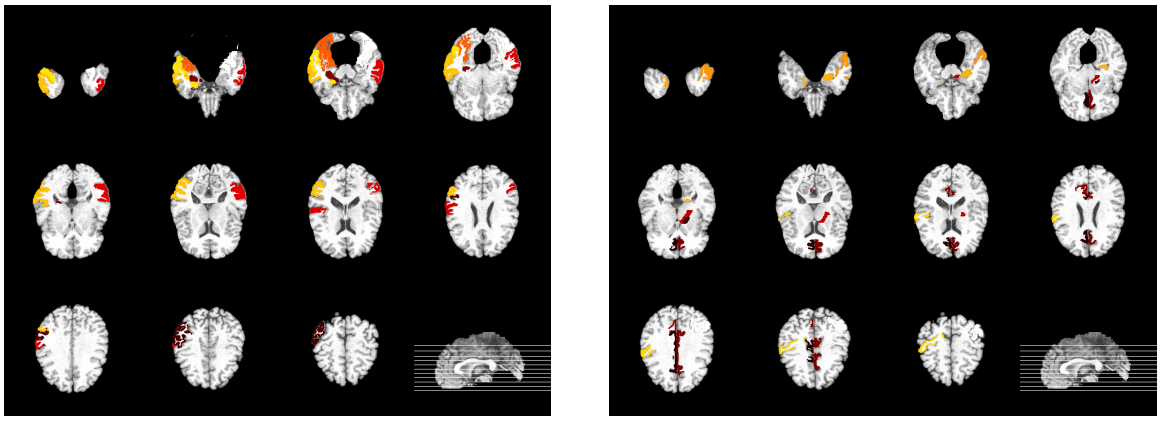
In this subsection, we investigate the most discriminative brain regions for the diagnosis of disease status and the prediction of the MMSE and ADAS-Cog scores. For each method, we repeated the whole 10-CV process 30 times and acquired 300 different models using different training datasets. Figure 3.6 shows the selection frequency of each of 93 ROIs for the AD/NC classification task using only MRI features, where the selection frequency for

each ROI is defined as

$$\text{Frequency} = 100 \times \frac{\text{The times of being selected in the 300 models}}{300}.$$

For each method, some ROIs are always selected while some ROIs are seldom selected. Compared with Lasso and M3T, the SRIG and GGML methods tend to select more ROIs since they use the feature graph information and encourage the significantly partially-correlated features to be selected jointly. According to the selection frequency, we compare the top ten selected ROIs of different methods for different tasks. Tables 3.6-3.8 show the indices of the top ten selected ROIs of the four methods for different tasks (classification or regression), different datasets (AD/NC or MCI/NC) and different modalities (MRI or PET). Table 3.9 contains the full names of the ROIs.

As shown in Tables 3.6-3.8, for different tasks, the top ten selected ROIs of the single task learning methods such as Lasso and SRIG are different while the top ten selected ROIs of the multi-task learning methods such as M3T and GGML are the same. We can also observe that the top ten selected ROIs for the cases using MRI features are not very similar to the top ten selected ROIs for the cases using PET features. One possible reason is that MRI features and PET features provide complementary information for the diagnosis of AD. However, for each case, the top ten selected ROIs of the four methods are similar. For example, for the AD/NC classification task using MRI features, Table 3.6 indicates that the ROIs with indices 18, 80, 83, 84, and 90 are frequently selected by all four methods. It is interesting to point out that both GGML and M3T methods also select the 48-th ROI frequently for the AD/NC classification task while this ROI is not one of the top ten selected ROIs of Lasso and SRIG for this task. However, as shown in Table 3.8, the 48-th ROI is frequently selected by Lasso and SRIG for the regression task (ADAS-Cog) using AD/NC data. This indicates that the multi-task learning methods such as GGML and M3T incorporate the clinical score information for the classification task. On the other hand, as shown in Table 3.8, both GGML and M3T methods select the 22-th ROI frequently for the regression task (ADAS-Cog) using AD/NC data while this ROI is not one of the top ten

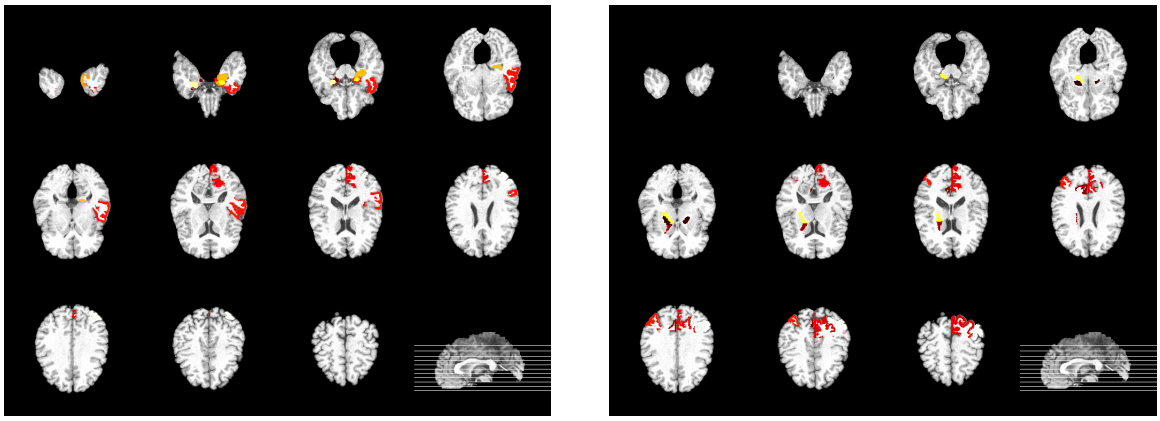


(a) MRI **(b)** PET

Figure 3.7: Top ten most discriminative brain regions (AD/NC dataset).

selected ROIs of Lasso and SRIG for this task. However, as shown in Table 3.6, the 22-th ROI is frequently selected by Lasso and SRIG for the classification task (AD vs NC). This indicates that the multi-task learning methods such as GGML and M3T incorporate the class label information for the regression task.

Furthermore, as shown in Tables 3.6-3.8, for the study using AD/NC data and MRI features, the common top ten selected ROIs of Lasso for different tasks are the ROIs with indices 18, 80, 83, 84 and 90. The common top ten selected ROIs of the SRIG method for different tasks are the ROIs with indices 58, 80, 83, and 84. Most of these ROIs are the top ten selected ROIs of our proposed GGML method. In Figures 3.7-3.8, we visualize the top ten selected ROIs of our proposed GGML method when different datasets (AD/NC or MCI/NC) and different modalities (MRI or PET) are used. Most of the selected regions, e.g., uncus right (22), hippocampal formation right (30), uncus left (46), middle temporal gyrus left (48), hippocampus formation left (69), middle temporal gyrus right (80) and amygdale right (83), are known to be highly correlated with AD and MCI by many studies using group comparison methods (Jack et al., 1999; Misra et al., 2009b; Zhang and Shen, 2012).



(a) MRI **(b)** PET

Figure 3.8: Top ten most discriminative brain regions (MCI/NC dataset).

3.6 Discussion

In this section, we first discuss some issues about constructing the undirected feature graph **G**. Then, some possible extensions of our proposed method will be discussed.

3.6.1 Construction of the undirected feature graph G

Before performing our proposed GGML method, we need to construct an undirected feature graph **G** representing the significant correlation information among features. In Section 3.3.2, we proposed to use the graphical Lasso method to construct this graph. For some datasets, the constructed graph **G** may include many edges corresponding to weak or even wrong partial correlation due to bad estimation of the precision matrix. In this case, by thresholding of the estimated precision matrix, we can construct a sparse undirected graph for representing only the most reliable partial correlation.

Furthermore, besides partial correlation information among features, we can also combine other useful information (e.g., some prior information about features) to construct this graph **G**. Our proposed GGML method can be used for any given undirected feature graph **G** representing the relationships among different features.

3.6.2 Use the structure information among different subjects

Our proposed GGML method utilizes both the correlation information among features and the intrinsic correlation information among different response variables. Actually, we can also generalize GGML method to incorporate the structure information among different subjects. Similar to the locality preserving projection (LPP) method ((He and Niyogi, 2004)), we can model the structure information among different training subjects as another sparse undirected graph $\mathbf{S}$. Here, $\mathbf{S}$ has n nodes and each node represents one subject. The connectivity of the graph $\mathbf{S}$ can be defined by the k nearest neighbors, i.e., subjects x_s and x_l are connected by an edge if x_s is among the k nearest neighbors of x_l, or x_l is among the k nearest neighbors of x_s. In order to use the structure information among different training subjects represented by $\mathbf{S}$, we can preserve the neighborhood structure of subjects, i.e., encouraging the predicted response variables $\hat{y}_s = \mathbf{B}^T x_s$ and $\hat{y}_l = \mathbf{B}^T x_l$ to be close if the s-th and the l-th subjects are connected in the undirected graph $\mathbf{S}$.

3.7 Conclusion

In summary, we propose a new graph guided multi-task learning method to incorporate the correlation information among features and the intrinsic correlation information among different tasks. To use the correlation information among features, our proposed GGML method encourages the partially-correlated features to be in or out of the model jointly. Furthermore, in order to acquire more robust and accurate feature selection, our proposed GGML method encourages different tasks to share a common useful feature subset. Theoretically, our proposed GGML method is very general and includes the M3T method as a special case. The experimental results on the simulated examples and the ADNI dataset also show the advantage of the proposed GGML method over some existing methods.

Table 3.2: Comparison of different methods using the simulated examples

Example	Method	Accuracy	CC1	CC2	RMSE1	RMSE2
1	Lasso	0.828 (0.007)	0.909 (0.004)	0.910 (0.003)	4.091 (0.070)	4.106 (0.064)
	SRIG	0.848 (0.009)	0.932 (0.003)	0.933 (0.002)	3.548 (0.062)	3.620 (0.057)
	M3T	0.840 (0.006)	0.918 (0.002)	0.917 (0.002)	3.916 (0.059)	4.005 (0.059)
	GGML	**0.872 (0.006)**	**0.938 (0.002)**	**0.936 (0.001)**	**3.402 (0.043)**	**3.488 (0.039)**
2	Lasso	0.765 (0.008)	0.781 (0.010)	0.767 (0.012)	4.567 (0.084)	4.596 (0.089)
	SRIG	0.800 (0.008)	0.823 (0.008)	0.810 (0.010)	4.134 (0.075)	4.213 (0.089)
	M3T	0.796 (0.008)	0.814 (0.008)	0.807 (0.008)	4.261 (0.075)	4.290 (0.075)
	GGML	**0.816 (0.008)**	**0.839 (0.007)**	**0.838 (0.007)**	**3.966 (0.069)**	**3.981 (0.073)**
3	Lasso	0.821 (0.005)	0.910 (0.004)	0.903 (0.005)	3.995 (0.066)	4.163 (0.096)
	SRIG	0.846 (0.008)	0.932 (0.003)	0.927 (0.004)	3.506 (0.063)	3.633 (0.084)
	M3T	0.843 (0.006)	0.918 (0.003)	0.913 (0.004)	3.907 (0.049)	3.992 (0.073)
	GGML	**0.872 (0.006)**	**0.938 (0.002)**	**0.934 (0.002)**	**3.388 (0.045)**	**3.464 (0.050)**

[CC1 (CC2) is the Pearson's correlation coefficient of the first (second) regression task; RMSE1 (RMSE2) is the root-mean-square error of the first (second) regression task. The values in the parenthesis are standard deviations.]

Table 3.3: Comparison of the classification performance on the ADNI dataset.

Data	Method	MRI	PET	MRI+PET	MRSRIGI+PET+CSF
AD/NC	Lasso	0.878 (0.003)	0.823 (0.003)	0.903 (0.003)	0.917 (0.003)
	SRIG	0.896 (0.003)	0.830 (0.003)	0.911 (0.002)	0.915 (0.002)
	M3T	0.884 (0.002)	0.821 (0.002)	0.914 (0.002)	0.918 (0.002)
	GGML	**0.906 (0.003)**	**0.832 (0.003)**	**0.919 (0.002)**	**0.926 (0.002)**
MCI/NC	Lasso	0.722 (0.003)	0.677 (0.003)	0.737 (0.004)	0.750 (0.004)
	SRIG	0.737 (0.004)	0.688 (0.004)	0.755 (0.005)	0.769 (0.003)
	M3T	0.738 (0.003)	0.655 (0.003)	0.775 (0.003)	0.776 (0.003)
	GGML	**0.751 (0.003)**	**0.696 (0.003)**	**0.784 (0.003)**	**0.800 (0.003)**

[The reported values are the averaged classification accuracy with standard deviation.]

Table 3.4: Comparison of the regression performance on the AD/NC dataset.

Response	Method	MRI	PET	MRI+PET	MRI+PET+CSF
MMSE	Lasso	0.601 (0.005)	0.601 (0.004)	0.688 (0.003)	0.709 (0.003)
	SRIG	0.656 (0.003)	**0.611 (0.003)**	0.698 (0.003)	0.723 (0.003)
	M3T	0.651 (0.004)	0.585 (0.003)	0.693 (0.002)	0.724 (0.002)
	GGML	**0.671 (0.002)**	0.598 (0.003)	**0.712 (0.002)**	**0.745 (0.002)**
ADAS-Cog	Lasso	0.695 (0.003)	0.611 (0.004)	0.652 (0.004)	0.664 (0.004)
	SRIG	0.703 (0.002)	0.632 (0.004)	0.708 (0.003)	0.719 (0.002)
	M3T	0.703 (0.002)	0.635 (0.003)	0.709 (0.003)	0.718 (0.002)
	GGML	**0.705 (0.002)**	**0.644 (0.003)**	**0.721 (0.002)**	**0.740 (0.002)**

[The reported values are the averaged correlation coefficient with standard deviation.]

Table 3.5: Comparison of the regression performance on the MCI/NC dataset.

Response	Method	MRI	PET	MRI+PET	MRI+PET+CSF
MMSE	Lasso	0.326 (0.006)	0.168 (0.010)	0.303 (0.007)	0.303 (0.007)
	SRIG	0.313 (0.007)	0.181 (0.004)	0.323 (0.005)	0.325 (0.005)
	M3T	0.382 (0.004)	0.182 (0.007)	0.379 (0.004)	0.364 (0.004)
	GGML	**0.394 (0.004)**	**0.213 (0.005)**	**0.392 (0.005)**	**0.382 (0.004)**
ADAS-Cog	Lasso	0.355 (0.006)	0.427 (0.006)	0.343 (0.006)	0.336 (0.006)
	SRIG	0.378 (0.005)	0.451 (0.005)	0.462 (0.004)	0.464 (0.003)
	M3T	0.354 (0.004)	0.406 (0.006)	0.429 (0.003)	0.426 (0.003)
	GGML	**0.391 (0.004)**	**0.469 (0.005)**	**0.462 (0.003)**	**0.472 (0.003)**

[The reported values are the averaged correlation coefficient with standard deviation.]

Table 3.6: Comparison of the top ten selected ROIs for the classification task.

		MRI	PET
AD/NC	Lasso	18, 22, 38, 44, 46, 69, 80, 83, 84, 90	12, 18, 23, 26, 41, 68, 69, 73, 81, 87
	SRIG	18, 22, 30, 44, 58, 69, 80, 83, 84, 90	12, 18, 26, 35, 41, 68, 69, 73, 79, 87
	M3T	9, 18, 22, 46, 48, 69, 80, 83, 84, 90	12, 23, 26, 35, 62, 68, 69, 73, 81, 87
	GGML	18, 22, 30, 44, 48, 67, 80, 83, 84, 90	7, 12, 23, 26, 35, 62, 68, 69, 73, 87
MCI/NC	Lasso	17, 28, 40, 48, 63, 64, 69, 83, 86, 92	2, 37, 39, 41, 54, 55, 63, 68, 81, 87
	SRIG	17, 22, 30, 40, 46, 64, 69, 76, 83, 92	11, 12, 23, 26, 28, 29, 38, 40, 41, 87
	M3T	17, 40, 46, 48, 53, 63, 64, 69, 83, 86	12, 35, 41, 62, 64, 68, 73, 79, 81, 87
	GGML	22, 40, 45, 46, 61, 64, 69, 76, 83, 86	11, 12, 26, 29, 38, 40, 41, 47, 79, 87

Table 3.7: Comparison of the top ten selected ROIs for the prediction of MMSE.

		MRI	PET
AD/NC	Lasso	9, 15, 18, 19, 22, 40, 80, 83, 84, 90	12, 18, 23, 26, 62, 63, 68, 69, 73, 79
	SRIG	19, 22, 48, 58, 62, 67, 80, 83, 84, 85	7, 12, 23, 26, 35, 41, 62, 68, 69, 73
	M3T	9, 18, 22, 46, 48, 69, 80, 83, 84, 90	12, 23, 26, 35, 62, 68, 69, 73, 81, 87
	GGML	18, 22, 30, 44, 48, 67, 80, 83, 84, 90	7, 12, 23, 26, 35, 62, 68, 69, 73, 87
MCI/NC	Lasso	17, 33, 40, 44, 48, 53, 62, 64, 69, 86	4, 23, 24, 33, 41, 61, 62, 68, 84, 87
	SRIG	22, 45, 46, 48, 61, 64, 69, 76, 83, 86	11, 12, 23, 26, 28, 29, 38, 40, 41, 87
	M3T	17, 40, 46, 48, 53, 63, 64, 69, 83, 86	12, 35, 41, 62, 64, 68, 73, 79, 81, 87
	GGML	22, 40, 45, 46, 61, 64, 69, 76, 83, 86	11, 12, 26, 29, 38, 40, 41, 47, 79, 87

Table 3.8: Comparison of the top ten selected ROIs for the prediction of ADAS.

		MRI	PET
AD/NC	Lasso	9, 18, 46, 48, 61, 62, 80, 83, 84, 90	12, 23, 26, 30, 35, 62, 73, 76, 81, 92
	SRIG	18, 30, 48, 58, 62, 67, 80, 83, 84, 85	7, 12, 23, 26, 30, 35, 62, 69, 73, 92
	M3T	9, 18, 22, 46, 48, 69, 80, 83, 84, 90	12, 23, 26, 35, 62, 68, 69, 73, 81, 87
	GGML	18, 22, 30, 44, 48, 67, 80, 83, 84, 90	7, 12, 23, 26, 35, 62, 68, 69, 73, 87
MCI/NC	Lasso	10, 17, 18, 38, 45, 46, 69, 72, 83, 87	10, 12, 14, 19, 35, 39, 41, 62, 64, 88
	SRIG	17, 45, 46, 61, 62, 69, 72, 76, 83, 87	11, 12, 28, 29, 35, 38, 41, 71, 79, 87
	M3T	17, 40, 46, 48, 53, 63, 64, 69, 83, 86	12, 35, 41, 62, 64, 68, 73, 79, 81, 87
	GGML	22, 40, 45, 46, 61, 64, 69, 76, 83, 86	11, 12, 26, 29, 38, 40, 41, 47, 79, 87

Table 3.9: Names of the selected ROIs in this study.

ROI Index	ROI Name	ROI Index	ROI Name
2	middle frontal gyrus right	47	middle occipital gyrus right
4	insula right	48	middle temporal gyrus left
7	cingulate region right	53	postcentral gyrus left
9	medial frontal gyrus left	54	inferior frontal gyrus right
10	superior frontal gyrus right	55	precentral gyrus left
11	globus palladus right	58	perirhinal cortex right
12	globus palladus left	61	perirhinal cortex left
14	inferior frontal gyrus left	62	inferior temporal gyrus left
15	putamen right	63	temporal pole left
17	parahippocampal gyrus left	64	entorhinal cortex left
18	angular gyrus right	67	lateral occipitotemporal gyrus right
19	temporal pole right	68	entorhinal cortex right
22	uncus right	69	hippocampal formation left
23	cingulate region left	71	parietal lobe WM right
24	fornix left	72	insula left
26	precuneus right	73	postcentral gyrus right
28	cerebral peduncle left	76	amygdala left
29	cerebral peduncle right	79	anterior limb of internal capsule right
30	hippocampal formation right	80	middle temporal gyrus right
33	caudate nucleus left	81	occipital pole right
35	anterior limb of internal capsule left	83	amygdala right
37	middle frontal gyrus left	84	inferior temporal gyrus right
38	superior parietal lobule left	85	superior temporal gyrus right
39	caudate nucleus right	86	middle occipital gyrus left
40	cuneus left	87	angular gyrus left
41	precuneus left	88	medial occipitotemporal gyrus right
44	supramarginal gyrus right	90	lateral occipitotemporal gyrus left
45	superior temporal gyrus left	92	occipital pole left
46	uncus left		

CHAPTER 4: SPARSE REGRESSION FOR BLOCK-MISSING MULTI-MODALITY DATA

4.1 Introduction

In modern scientific research, many data are collected from multiple modalities (sources or types). Since different modalities could provide complementary information, sparse regression methods using multi-modality data could deliver better prediction performance. However, one special challenge for using multi-modality data is related to missing data, which is unavoidable due to some reasons such as the high cost of measures or the patients' dropout. Generally, the observations of a certain modality can be missing completely, i.e., a complete block of the data is missing. One example of block-missing multi-modality data is shown in Figure 4.1. In this example, there are n samples (each row is one sample), three modalities and one response variable. The blank regions with question mark indicate missing data.

In regard to the problem of sparse regression for block-missing multi-modality data, the simplest method is to remove all samples with missing observations. However, this approach can greatly reduce the sample size and waste a lot of useful information in the samples with missing observations. Another strategy is to impute the missing data first by some imputation methods such as (Hastie et al., 1999), (Schott et al., 2010), and (Cai et al., 2010). These methods can be effective when the missing locations are random, but they can be ineffective when a complete block of the data is missing.

In the literature, one important recent technique for block-missing multi-modality data is the incomplete Multi-Source Feature learning (iMSF) method proposed by (Yuan et al., 2012). The iMSF method performs classification/regression on block-missing multi-modality data without the need of missing data imputation. It formulates the prediction problem as a multi-task learning problem by first decomposing the prediction problem into a set of tasks (classification or regression), one for each combination of available modalities

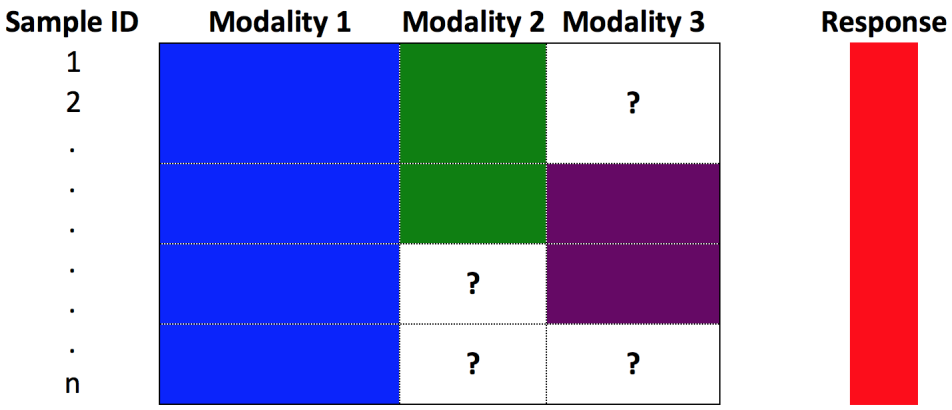


Figure 4.1: An illustration of a block-missing multi-modality data set with three modalities.

(e.g., modality 1, modalities 1 and 2, modalities 1 and 3, modalities 1, 2, and 3 for the example shown in Figure 4.1), and then building the models for all tasks simultaneously. The important assumption in the iMSF method is that all models involving a specific modality share the common set of predictors for that particular modality. However, when different modalities are highly correlated, this assumption could be too strong. In that case, for some modalities, it is more reasonable to choose different predictor subsets for different involved tasks. Therefore, it is desirable to develop flexible and efficient sparse regression methods applicable to block-missing multi-modality data.

In this chapter, we propose a new sparse regression method for block-missing multi-modality data. Our method has two steps. In the first step, we use all available information to estimate the covariance matrix of the predictors and the cross-covariance matrix between the predictors and the response variable. In the second step, based on the estimated covariance matrix and the estimated cross-covariance matrix, we use a modified Lasso estimator to deliver good estimates of the regression coefficients. Both the simulation study and the real data analysis demonstrate the effectiveness of our proposed method. Since our method uses all available information efficiently, it could deliver better performance than many existing methods.

The rest of this chapter is organized as follows. In Section 4.2, we motivate and introduce our proposed method. In Sections 4.3 and 4.4, we demonstrate the use of our method on

simulated data and the ADNI dataset with block-missing entries. We conclude this chapter in Section 4.5.

4.2 Motivation and Methodology

Suppose there are K modalities with $p_1, p_2, \ldots, p_K$ predictors, respectively. Consider the following linear regression model:

$$Y = \mathbf{X}^{(1)}\beta^{(1)} + \mathbf{X}^{(2)}\beta^{(2)} + \cdots + \mathbf{X}^{(K)}\beta^{(K)} + \epsilon, \tag{4.1}$$

where $Y = (y_1, y_2, \ldots, y_n)^T$ is an $n \times 1$ response vector and $\epsilon = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^T$ is a vector of i.i.d. random variables with mean 0 and variance σ^2. For the k-th modality, we use $\mathbf{X}^{(k)} \in R^{n \times p_k}$ and $\beta^{(k)} \in R^{p_k}$ to denote the observations of the p_k predictors and the vector of the true coefficients, respectively. In addition, we use $\mathbf{X} = (\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(K)}) = (x_1, x_2, \ldots, x_n)^T$ to denote the $n \times p$ design matrix, where $p = p_1 + p_2 + \cdots + p_K$. We assume that each x_i follows some multivariate distribution with mean $0_{p\times 1}$ and covariance matrix $\mathbf{\Sigma}$. The design matrix $\mathbf{X}$ is assumed to be independent of the random error ϵ. We use $\mathbf{\Sigma}_{xy} = (c_1, c_2, \ldots, c_p)^T \in R^p$ to denote the cross-covariance vector between x_i and y_i.

For complete data with no missing entries, the classical Lasso method estimates $\beta^0 = ({\beta^{(1)}}^T, {\beta^{(2)}}^T, \ldots, {\beta^{(K)}}^T)^T$ by solving the following optimization problem:

$$\min_{\beta} \frac{1}{2n}\|Y - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1,$$

where $\|Y - \mathbf{X}\beta\|_2$ denotes the ℓ_2 norm of $Y - \mathbf{X}\beta$, $\|\beta\|_1$ denotes the ℓ_1 norm of β, and λ is a tuning parameter.

For the block-missing multi-modality data, the above Lasso method is not applicable since there are many block-missing entries in the design matrix $\mathbf{X}$. However, we can estimate β^0 by solving the following optimization problem

$$\min_{\beta} \frac{1}{2n}E(\|Y - \mathbf{X}\beta\|_2^2) + \lambda\|\beta\|_1,$$

which is equivalent to

$$\min_{\beta} \frac{1}{2}\beta^T \mathbf{\Sigma}\beta - \mathbf{\Sigma}_{xy}^T\beta + \lambda\|\beta\|_1. \tag{4.2}$$

Motivated by the formula (4.2), we propose a new two-step sparse regression procedure for block-missing multi-modality data. In the first step, we use all available data to estimate the covariance matrix $\mathbf{\Sigma}$ and the cross-covariance vector $\mathbf{\Sigma}_{xy}$. The estimates of $\mathbf{\Sigma}$ and $\mathbf{\Sigma}_{xy}$ are denoted as $\hat{\mathbf{\Sigma}}$ and $\hat{\mathbf{\Sigma}}_{xy}$, respectively. In the second step, we estimate β^0 by solving the following optimization problem:

$$\min_{\beta} \frac{1}{2}\beta^T \hat{\mathbf{\Sigma}}\beta - \hat{\mathbf{\Sigma}}_{xy}^T\beta + \sum_{k=1}^{K} \lambda_k\|\beta^{(k)}\|_1, \tag{4.3}$$

where we can use different tuning parameters λ_k's for different modalities.

Next, we discuss how to estimate $\mathbf{\Sigma}$ and $\mathbf{\Sigma}_{xy}$ using the block-missing multi-modality data. For each predictor j, we denote S_j as the set $\{i : x_{ij} \text{ is not missing}\}$. For predictors j and t, we denote S_{jt} as the set $\{i : \text{ both } x_{ij} \text{ and } x_{it} \text{ are not missing}\}$. The number of elements in S_j and S_{jt} are denoted as $|S_j|$ and $|S_{jt}|$, respectively.

A natural initial estimate of $\mathbf{\Sigma}$ using all available data is

$$\tilde{\mathbf{\Sigma}} = (\tilde{\sigma}_{jt})_{j,t=1,2,\ldots,p}, \text{ where } \tilde{\sigma}_{jt} = \frac{1}{|S_{jt}|}\sum_{i\in S_{jt}} x_{ij}x_{it}.$$

For block-missing multi-modality data, the above initial estimate $\tilde{\mathbf{\Sigma}}$ can be ill-conditioned and have negative eigenvalues. Therefore, it may not be a good estimate of $\mathbf{\Sigma}$ and can not be used in (4.3) directly. We will introduce an estimator that is both well-conditioned and more accurate than the initial estimate $\tilde{\mathbf{\Sigma}}$. Denote $\tilde{\mathbf{\Sigma}}_B$ as the block-diagonal matrix with K blocks where the k-th block is the sample covariance matrix of the predictors from the k-th modality. Let $\tilde{\mathbf{\Sigma}}_O = \tilde{\mathbf{\Sigma}} - \tilde{\mathbf{\Sigma}}_B$. We propose to use the following estimate

$$\hat{\mathbf{\Sigma}} = \alpha_1\tilde{\mathbf{\Sigma}}_B + \alpha_2\tilde{\mathbf{\Sigma}}_O + \alpha_3\mathbf{I_p},$$

where α_1, α_2 and α_3 are three nonrandom weights. Our goal is to find the optimal linear combination $\tilde{\mathbf{\Sigma}}^* = \alpha_1^* \tilde{\mathbf{\Sigma}}_B + \alpha_2^* \tilde{\mathbf{\Sigma}}_O + \alpha_3^* \mathbf{I_p}$ whose expected quadratic loss $E[\|\tilde{\mathbf{\Sigma}}^* - \mathbf{\Sigma}\|_F^2]$ is minimum. The optimal weights α_1^*, α_2^* and α_3^* are shown in the following Theorem 4.1.

Theorem 4.1. Consider the following optimization problem:

$$\min_{\alpha_1,\alpha_2,\alpha_3} E[\|\hat{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2]$$
$$s.t. \quad \hat{\mathbf{\Sigma}} = \alpha_1 \tilde{\mathbf{\Sigma}}_B + \alpha_2 \tilde{\mathbf{\Sigma}}_O + \alpha_3 \mathbf{I_p},$$

where the weights α_1, α_2 and α_3 are nonrandom. Denote $\gamma^* = tr(\mathbf{\Sigma})/p$, $\delta_B^2 = E[\|\tilde{\mathbf{\Sigma}}_B - \mathbf{\Sigma}_B\|_F^2]$, $\delta_O^2 = E[\|\tilde{\mathbf{\Sigma}}_O - \mathbf{\Sigma}_O\|_F^2]$, and $\theta^2 = \|\gamma^* \mathbf{I_p} - \mathbf{\Sigma}_B\|_F^2$. The optimal weights are

$$\alpha_1^* = \frac{\theta^2}{\theta^2 + \delta_B^2}, \quad \alpha_2^* = \frac{\|\mathbf{\Sigma}_O\|_F^2}{\|\mathbf{\Sigma}_O\|_F^2 + \delta_O^2}, \quad \alpha_3^* = \gamma^*(1 - \alpha_1^*) = \frac{\gamma^* \delta_B^2}{\theta^2 + \delta_B^2}.$$

In addition, we have

$$E[\|\tilde{\mathbf{\Sigma}}^* - \mathbf{\Sigma}\|_F^2] = \frac{\delta_B^2 \theta^2}{\delta_B^2 + \theta^2} + \frac{\delta_O^2 \|\mathbf{\Sigma}_O\|_F^2}{\delta_O^2 + \|\mathbf{\Sigma}_O\|_F^2} \leq \delta_B^2 + \delta_O^2 = E[\|\tilde{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2].$$

Proof. By changing variables, the optimization problem can be rewritten as

$$\min_{\alpha_1,\alpha_2,\gamma} E[\|\hat{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2]$$
$$s.t. \quad \hat{\mathbf{\Sigma}} = \alpha_1 \tilde{\mathbf{\Sigma}}_B + \alpha_2 \tilde{\mathbf{\Sigma}}_O + (1 - \alpha_1)\gamma \mathbf{I_p}.$$

Denote $\mathbf{\Sigma}_B$ as the block-diagonal matrix with K blocks where the k-th block is the covariance matrix of the predictors from the k-th modality. Let $\mathbf{\Sigma}_O = \mathbf{\Sigma} - \mathbf{\Sigma}_B$. Using the facts that $\mathbf{\Sigma} = \mathbf{\Sigma}_B + \mathbf{\Sigma}_O$ and $E(\tilde{\mathbf{\Sigma}}_B) = \mathbf{\Sigma}_B$, we can rewrite the objective function as

$$\begin{aligned}
E[\|\hat{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2] =& E[\|\alpha_1 \tilde{\mathbf{\Sigma}}_B + \alpha_2 \tilde{\mathbf{\Sigma}}_O + (1 - \alpha_1)\gamma \mathbf{I_p} - \mathbf{\Sigma}\|_F^2] \\
=& E[\|\alpha_1 \tilde{\mathbf{\Sigma}}_B + \alpha_2 \tilde{\mathbf{\Sigma}}_O + (1 - \alpha_1)\gamma \mathbf{I_p} - \alpha_1 \mathbf{\Sigma}_B - (1 - \alpha_1)\mathbf{\Sigma}_B - \mathbf{\Sigma}_O\|_F^2] \\
=& E[\|\alpha_1(\tilde{\mathbf{\Sigma}}_B - \mathbf{\Sigma}_B) + (1 - \alpha_1)(\gamma \mathbf{I_p} - \mathbf{\Sigma}_B)\|_F^2] + E[\|\alpha_2 \tilde{\mathbf{\Sigma}}_O - \mathbf{\Sigma}_O\|_F^2] \\
=& \alpha_1^2 E[\|\tilde{\mathbf{\Sigma}}_B - \mathbf{\Sigma}_B\|_F^2] + (1 - \alpha_1)^2 \|\gamma \mathbf{I_p} - \mathbf{\Sigma}_B\|_F^2 + E[\|\alpha_2 \tilde{\mathbf{\Sigma}}_O - \mathbf{\Sigma}_O\|_F^2].
\end{aligned}$$

Therefore, the optimal value of γ can be obtained by minimizing $\|\gamma \mathbf{I_p} - \mathbf{\Sigma}_B\|_F^2$. Thus, the optimal value is $\gamma^* = tr(\mathbf{\Sigma}_B)/p = tr(\mathbf{\Sigma})/p$. The optimal value of α_2 can be obtained by minimizing $E[\|\alpha_2 \tilde{\mathbf{\Sigma}}_O - \mathbf{\Sigma}_O\|_F^2]$. The optimal value is $\alpha_2^* = \frac{\|\mathbf{\Sigma}_O\|_F^2}{\|\mathbf{\Sigma}_O\|_F^2 + \delta_O^2}$. Replacing γ by its optimal value γ^* in the objective function and taking the derivative of the objective function with respect to α_1, we can find that the optimal value of α_1 is $\alpha_1^* = \frac{\theta^2}{\theta^2 + \delta_B^2}$. Thus, the optimal value of α_3 is $\alpha_3^* = \gamma^*(1 - \alpha_1^*) = \frac{\gamma^* \delta_B^2}{\theta^2 + \delta_B^2}$.

At the optimum, the objective function is equal to $\frac{\delta_B^2 \theta^2}{\delta_B^2 + \theta^2} + \frac{\delta_O^2 \|\mathbf{\Sigma}_O\|_F^2}{\delta_O^2 + \|\mathbf{\Sigma}_O\|_F^2}$, which is less than $\delta_B^2 + \delta_O^2$. Since $E[\|\tilde{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2] = \delta_B^2 + \delta_O^2$, we know that $E[\|\tilde{\mathbf{\Sigma}}^* - \mathbf{\Sigma}\|_F^2] \leq E[\|\tilde{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2]$. $\square$

Theorem 4.1 indicates that $\gamma^* \mathbf{I_p}$ can be viewed as a shrinkage target and the weight $1 - \alpha_1^*$ is the shrinkage intensity. Moreover, it shows that $\mathbf{\Sigma}^*$ is more accurate than the sample covariance matrix. The relative improvement in expected quadratic loss over the sample covariance matrix is equal to

$$\frac{E[\|\tilde{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2] - E[\|\tilde{\mathbf{\Sigma}}^* - \mathbf{\Sigma}\|_F^2]}{E[\|\tilde{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_F^2]} = \frac{\delta_B^2}{\delta_B^2 + \delta_O^2} \cdot (1 - \alpha_1^*) + \frac{\delta_O^2}{\delta_B^2 + \delta_O^2} \cdot (1 - \alpha_2^*).$$

Therefore, if $\tilde{\mathbf{\Sigma}}_B$ is relatively accurate (δ_B^2 is small), then the optimal weight $\alpha_1^* = \frac{\theta^2}{\theta^2 + \delta_B^2}$ should be large and the percentage relative improvement tends to be small. If $\tilde{\mathbf{\Sigma}}_B$ is relatively inaccurate (δ_B^2 is large), then the optimal weight $\alpha_1^* = \frac{\theta^2}{\theta^2 + \delta_B^2}$ should be small and the percentage relative improvement tends to be large. We can also make similar conclusions about $\tilde{\mathbf{\Sigma}}_O$. In addition, for the block-missing multi-modality data, due to the imbalanced sample sizes, the initial estimate $\tilde{\mathbf{\Sigma}}_B$ can be relatively accurate while the estimate $\tilde{\mathbf{\Sigma}}_O$ is relatively inaccurate. In that case, we may need to use different weights for $\tilde{\mathbf{\Sigma}}_B$ and $\tilde{\mathbf{\Sigma}}_O$. As a remark, Theorem 4.1 has some interesting connections with the Theorem 2.1 shown in (Ledoit and Wolf, 2004), where they study the optimal linear combination of the sample covariance matrix and the identity matrix to estimate the covariance matrix using data without missing entries.

Regarding $\mathbf{\Sigma}_{xy}$, we choose the following estimate

$$\hat{\mathbf{\Sigma}}_{xy} = (\hat{c}_1, \hat{c}_2, \cdots, \hat{c}_p)^T, \text{ where } \hat{c}_j = \frac{1}{|S_j|} \sum_{i \in S_j} y_i x_{ij}.$$

After estimating $\mathbf{\Sigma}$ and $\mathbf{\Sigma}_{xy}$, our proposed sparse regression procedure for block-missing multi-modality data estimates β^0 by solving the following optimization problem:

$$\min_{\beta} \frac{1}{2}\beta^T[\alpha_1\tilde{\mathbf{\Sigma}}_B + \alpha_2\tilde{\mathbf{\Sigma}}_O + (1-\alpha_1)\frac{tr(\tilde{\mathbf{\Sigma}})}{p}\mathbf{I_p}]\beta - \hat{\mathbf{\Sigma}}_{xy}^T\beta + \sum_{k=1}^{K}\lambda_k\|\beta^{(k)}\|_1. \tag{4.4}$$

Here, we use $tr(\tilde{\mathbf{\Sigma}})/p$ to estimate γ^*. Both $\alpha_1 \in [0,1]$, $\alpha_2 \in [0,1]$, and λ_k's can be chosen by cross validation or an additional tuning dataset. In practice, we can choose reasonable α_1 and α_2 so that the estimated covariance matrix $\alpha_1\tilde{\mathbf{\Sigma}}_B + \alpha_2\tilde{\mathbf{\Sigma}}_O + (1-\alpha_1)\frac{tr(\tilde{\mathbf{\Sigma}})}{p}\mathbf{I_p}$ is nonnegative and well-conditioned. Our flexible procedure uses the block-missing multi-modality data information efficiently without imputing missing data. It's also easy to solve the quadratic programming problem (4.4). For example, we can use the **scout** ((Witten and Tibshirani, 2011)) R package.

4.3 Simulation Study

In this section, we perform numerical studies using simulated examples. For each example, we compare our proposed method with 1) Lasso: Lasso method which only uses the samples with complete observations; 2) Imputed Lasso: Lasso method which uses all samples with missing values imputed by the Soft-thresholded SVD method ((Mazumder et al., 2010)); 3) Ridge: Ridge regression method which only uses the samples with complete observations; 4) Imputed Ridge: Ridge regression method which uses all samples with missing values imputed by the Soft-thresholded SVD method; and 5) iMSF: the iMSF method which uses all available data without imputing the missing data.

4.3.1 Simulated examples

We study three simulated examples. Data are generated from three modalities and each modality has 100 features. All these examples have the same missing pattern as shown in Figure 4.1. For each example, the training data set is composed of 100 samples with complete observations, 100 samples with observations from the first and the second modalities, 100 samples with observations from the first and the third modalities, and 100 samples with observations only from the first modality. The tuning data set contains 200

samples with complete observations and the testing data set contains 400 samples with complete observations. All methods use the tuning data set to choose the best tuning parameters. Samples with complete observations are generated as follows.

Example 1: The features $(x_{i1}, x_{i2}, \ldots, x_{ip})^T \sim N(0, \mathbf{\Sigma})$ with $\sigma_{jt} = 0.6^{|j-t|}$. The true coefficient vector

$$\beta^0 = (0.5, 0.5, 0.5, \underbrace{0, \cdots, 0}_{97}, 0.5, 0.5, 0.5, \underbrace{0, \cdots, 0}_{97}, 0.5, 0.5, 0.5, \underbrace{0, \cdots, 0}_{97}).$$

The response variables are generated by Model (4.1) with the errors $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \overset{i.i.d}{\sim} N(0, 1)$.

Example 2: The features $(x_{i1}, x_{i2}, \ldots, x_{ip})^T \sim N(0, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ is a block diagonal matrix with $p/5$ blocks. Each block is a 5×5 square matrix with ones on the main diagonal and 0.15 else where. The true coefficient vector

$$\beta^0 = (\underbrace{0.5, \cdots, 0.5}_{5}, \underbrace{0, \cdots, 0}_{95}, \underbrace{0.5, \cdots, 0.5}_{5}, \underbrace{0, \cdots, 0}_{95}, \underbrace{0.5, \cdots, 0.5}_{5}, \underbrace{0, \cdots, 0}_{95}).$$

The response variables are generated by Model (4.1) with the errors $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \overset{i.i.d}{\sim} N(0, 1)$.

Example 3: The features $(x_{i1}, x_{i2}, \ldots, x_{ip})^T \sim N(0, \mathbf{A} \otimes \mathbf{B})$, where

$$\mathbf{A} = \begin{pmatrix} 1 & 0.4 & 0.6 \\ 0.4 & 1 & 0.2 \\ 0.6 & 0.2 & 1 \end{pmatrix},$$

and $\mathbf{B} = (b_{jt})_{j,t=1,2,\ldots,p/3}$ with $b_{jt} = 0.3^{|j-t|}$.

The true coefficient vector

$$\beta^0 = (0.5, 0.5, 0.5, \underbrace{0, \cdots, 0}_{97}, 0.5, 0.5, 0.5, \underbrace{0, \cdots, 0}_{97}, 0.5, 0.5, 0.5, \underbrace{0, \cdots, 0}_{97}).$$

The response variables are generated by Model (4.1) with the errors $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \overset{i.i.d}{\sim} N(0,1)$.

For each example, we repeated the simulation 30 times. To evaluate different methods, we use the following measures:

- ℓ_2 distance $\|\hat{\beta} - \beta^0\|_2$;
- Mean squared error (MSE);
- False positive rate (FPR) and False negative rate (FNR).

4.3.2 Simulated results

The means and the corresponding standard errors of the above four measures are shown in Tables 4.1, 4.2, and 4.3. These results indicate that our proposed method has the best performance of estimation, prediction, and model selection for all three examples. For the Lasso method, using the imputed data can improve performance in most cases. However, as shown in Table 4.1 and Table 4.3, the Lasso method using the imputed data may deliver worse estimate of the true coefficient vector β^0. For the Ridge regression method, our simulated results indicate that using the imputed data can always improve the performance of estimation and prediction.

Compared with the Lasso and Ridge regression methods using the imputed data set or only the samples with complete observations, the iMSF method delivers better estimation and prediction in most cases. However, iMSF method has high false positive rate for these three simulated examples. In addition, the comparison between iMSF and our method shows that our proposed method could use all available data more efficiently and therefore acquires better performance.

4.4 Real Data Analysis

To evaluate our proposed method, we also studied the ADNI dataset with block-missing data. The main goal of ADNI is to test whether serial magnetic resonance imaging (MRI),

Table 4.1: Performance comparison of Example 1.

Methods	ℓ_2 distance	MSE	FPR	FNR
Lasso	0.661 (0.029)	1.436 (0.046)	0.072 (0.004)	0.015 (0.009)
Imputed Lasso	0.668 (0.017)	1.326 (0.019)	0.073 (0.006)	0.000 (0.000)
Ridge	1.268 (0.004)	3.932 (0.058)	1.000 (0.000)	0.000 (0.000)
Imputed Ridge	1.084 (0.012)	2.274 (0.037)	1.000 (0.000)	0.000 (0.000)
iMSF	0.572 (0.020)	1.337 (0.035)	0.179 (0.010)	0.000 (0.000)
Proposed Method	**0.414 (0.013)**	**1.134 (0.014)**	**0.028 (0.003)**	**0.000 (0.000)**

positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessments can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD). In our study, we extracted features from three modalities: structural MRI, fluorodeoxyglucose PET, and CerebroSpinal Fluid (CSF). After data processing, we got 93 features from MRI, 93 features from PET, and 5 features from CSF. There are 805 subjects in total, including 1) 199 subjects with complete MRI, PET, and CSF features, 2) 197 subjects with only MRI and PET features, 3) 201 subjects with only MRI and CSF features, and 4) 208 subjects with only MRI features. The response variables used in our study are the Mini Mental State Examination (MMSE) score and the Alzheimer's Disease Assessment Scale-Cognitive Subscale (ADAS-Cog) score that are very useful to help evaluate the stage of AD pathology and predict future progression. We will use all available observations collected from MRI, PET, and CSF to predict these two clinical scores separately.

In our analysis, we divided the data into three parts: training data set, tuning data set, and testing data set. The training data set consists of all subjects with incomplete observations and 40 randomly selected subjects with complete MRI, PET, and CSF features. The tuning data set consists of another 40 randomly selected subjects (different from the training data set) with complete observations. The testing data set contains the other 119

Table 4.2: Performance comparison of Example 2.

Methods	ℓ_2 distance	MSE	FPR	FNR
Lasso	0.920 (0.025)	1.988 (0.059)	0.133 (0.007)	0.002 (0.002)
Imputed Lasso	0.690 (0.013)	1.546 (0.030)	0.122 (0.007)	0.000 (0.000)
Ridge	1.662 (0.006)	5.262 (0.066)	1.000 (0.000)	0.000 (0.000)
Imputed Ridge	1.332 (0.009)	3.130 (0.048)	1.000 (0.000)	0.000 (0.000)
iMSF	0.777 (0.016)	1.730 (0.040)	0.291 (0.012)	0.000 (0.000)
Proposed Method	**0.597 (0.019)**	**1.373 (0.033)**	**0.083 (0.007)**	**0.000 (0.000)**

subjects with complete observations. The tuning data set was used to choose the best tuning parameters for all methods and the testing data set was used to evaluate different methods. We used different methods as shown in the simulation study to predict MMSE score and ADAS-Cog score using all available MRI, PET, and CSF features. The analysis was repeated 30 times using different partitions of the data.

The results are shown in Tables 4.4 and 4.5. Compared with the other methods, our proposed method acquires the best performance on the prediction of both MMSE score and ADAS-Cog score. The iMSF method has better prediction performance than the Lasso and ridge regression using only samples with complete observations. However, iMSF may not perform as good as Lasso and ridge regression using the imputed data. In addition, the comparison between Lasso and Imputed Lasso (and also the comparison between Ridge and Imputed Ridge) indicates that imputing the missing data could improve the prediction performance.

Regarding the model selection, as shown in Tables 4.4 and 4.5, the Lasso method using the imputed data selected many more features than the method using only samples with complete observations. Both iMSF and our proposed method could deliver a model with relatively small number of features. Figures 4.2 and 4.3 show the selection frequency of all the 191 features for the prediction of MMSE score and ADAS-Cog score, respectively. The

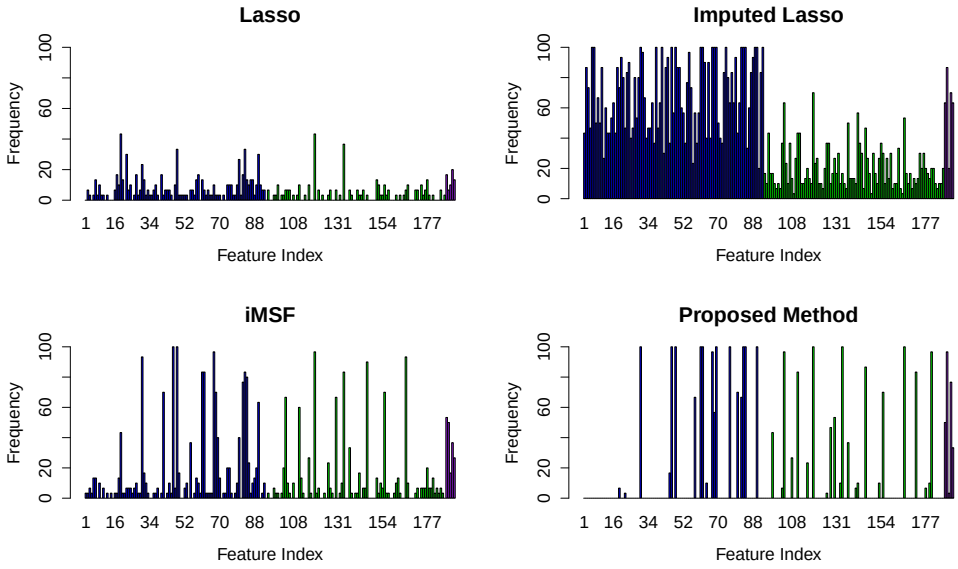


Figure 4.2: Selection frequency of 191 features for the prediction of MMSE score. The 93 blue bars represent 93 MRI features, the 93 green bars represent 93 PET features, and the 5 purple bars represent 5 CSF features.

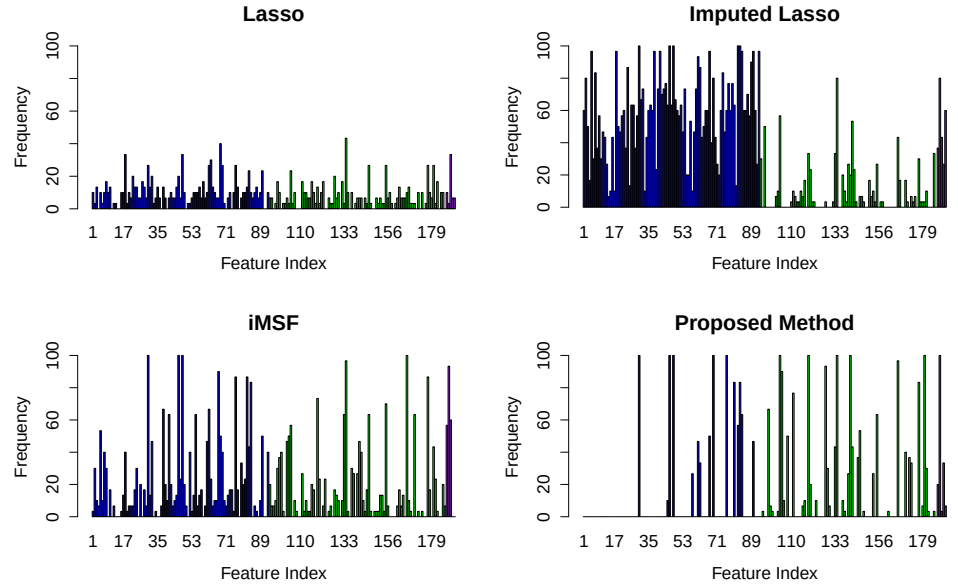


Figure 4.3: Selection frequency of 191 features for the prediction of ADAS-Cog score. The 93 blue bars represent 93 MRI features, the 93 green bars represent 93 PET features, and the 5 purple bars represent 5 CSF features.

Table 4.3: Performance comparison of Example 3.

Methods	ℓ_2 distance	MSE	FPR	FNR
Lasso	0.582 (0.028)	1.358 (0.038)	0.071 (0.005)	0.000 (0.000)
Imputed Lasso	0.713 (0.018)	1.288 (0.022)	0.067 (0.005)	0.000 (0.000)
Ridge	1.227 (0.004)	4.760 (0.071)	1.000 (0.000)	0.000 (0.000)
Imputed Ridge	0.948 (0.011)	1.959 (0.030)	1.000 (0.000)	0.000 (0.000)
iMSF	0.475 (0.017)	1.237 (0.028)	0.137 (0.012)	0.000 (0.000)
Proposed Method	**0.396 (0.011)**	**1.117 (0.015)**	**0.001 (0.001)**	**0.000 (0.000)**

selection frequency for each feature is defined as

$$\text{Selection Frequency} = 100 \times \frac{\text{The times of being selected in the 30 times simulations}}{30}$$

As shown in Figures 4.2 and 4.3, for our proposed method, in the 30 times simulation, some features were always selected and a lot of features were never selected. This means that our method could deliver relatively robust performance on model selection. However, for some other methods such as the Imputed Lasso method, since the majority of features have nonzero selection frequencies, these methods selected very different features in different repetitions. For the Imputed Lasso method, one possible reason for the unstable performance on model selection is due to the randomness involved in the imputation of a lot of block-missing data.

Overall, this real data analysis indicates that our proposed method could make use of all available information efficiently, and therefore deliver good prediction performance. Since our method does not require to impute the block-missing data, the performance of model selection is relatively robust.

Table 4.4: Prediction Performance of MMSE score.

Methods	Mean Squared Error		Number of Selected Features	
	Mean	SD	Mean	SD
Lasso	5.711	0.341	11.733	1.638
Imputed Lasso	4.711	0.082	86.700	8.559
Ridge	5.273	0.204	191.000	0.000
Imputed Ridge	4.478	0.055	191.000	0.000
iMSF	4.630	0.079	28.400	3.025
Proposed Method	**4.178**	**0.058**	27.633	0.908

4.5 Conclusion

In this chapter, we propose a new two-step sparse regression method for block-missing multi-modality data. In the first step, we estimate the covariance matrix of the predictors using a linear combination of the sample covariance matrix and the identity matrix. The proposed estimator of the covariance matrix can be well-conditioned and more accurate than the sample covariance matrix. We also use all available information to estimate the cross covariance vector between the predictors and the response variable. In the second step, based on the estimated covariance matrix and the cross-covariance vector, a modified Lasso estimator is used to deliver a sparse estimate of the regression coefficients in the linear regression model. The effectiveness of the proposed method is demonstrated by both simulated examples and the real data example from the Alzheimer's Disease Neuroimaging Initiative. The comparison between our proposed method and several existing methods also indicates that our method has promising performance on estimation, prediction, and model selection.

Table 4.5: Prediction Performance of ADAS-Cog score.

Methods	Mean Squared Error		Number of Selected Features	
	Mean	SD	Mean	SD
Lasso	31.636	1.647	17.267	1.681
Imputed Lasso	25.332	0.423	65.200	6.626
Ridge	25.692	0.899	191.000	0.000
Imputed Ridge	23.595	0.352	191.000	0.000
iMSF	25.425	0.628	38.567	4.372
Proposed Method	**22.399**	**0.379**	27.967	1.744

BIBLIOGRAPHY

Bain, L. J., Jedrziewski, K., Morrison-Bogorad, M., Albert, M., Cotman, C., Hendrie, H., and Trojanowski, J. Q. (2008). Healthy brain aging: A meeting report from the sylvan m. cohen annual retreat of the university of pennsylvania institute on aging. *Alzheimer's & dementia : the journal of the Alzheimer's Association*, 4:443–446.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202.

Bickel, P. J., Ritov, Y., and Tsybakov, A. B. (2009). Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics*, 37(4):1705–1732.

Bondell, H. D. and Reich, B. J. (2008). Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar. *Biometrics*, 64:115–123.

Bouwman, F. H., van der Flier, W. M., Schoonenboom, N. S., van Elk, E. J., Kok, A., Rijmen, F., Blankenstein, M. A., and Scheltens, P. (2007). Longitudinal changes of csf biomarkers in memory clinic patients. *Neurology*, 69(10):1006–1011.

Breheny, P. and Huang, J. (2009). Penalized methods for bi-level variable selection. *Statistics and Its Interface*, 2:369–380.

Breiman, L. and Friedman, J. H. (1997). Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59(1):3–54.

Brookmeyer, R., Johnson, E., Ziegler-Graham, K., and Arrighi, H. M. (2007). Forecasting the global burden of alzheimers disease. *Alzheimer's & Dementia*, 3:186–191.

Cai, J., Candès, E. J., and Shen, Z. (2010). A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20:1956–1982.

Cai, T., Liu, W., and Luo, X. (2011). A constrained l_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607.

Chen, L. and Huang, J. Z. (2012). Sparse reduced-rank regression for simultaneous dimension reduction and variable selection. *Journal of the American Statistical Association*, 107(500):1533–1545.

Cheng, B., Zhang, D., Chen, S., Kaufer, D. I., and Shen, D. (2013). Semi-supervised multimodal relevance vector regression improves cognitive performance estimation from imaging and biological biomarkers. *Neuroinformatics*, pages 1–15.

Chung, D., Chun, H., and Keles, S. (2012). Spls: sparse partial least squares (spls) regression and classification. *R package, version*, 2:1–1.

Combettes, P. L. and Pesquet, J.-C. (2011). Proximal splitting methods in signal processing. In *Fixed-point algorithms for inverse problems in science and engineering*, pages 185–212. Springer.

De Santi, S., de Leon, M. J., Rusinek, H., Convit, A., Tarshish, C. Y., Roche, A., Tsui, W. H., Kandil, E., Boppana, M., Daisley, K., et al. (2001). Hippocampal formation glucose metabolism and volume losses in mci and ad. *Neurobiology of Aging*, 22(4):529–539.

Du, A.-T., Schuff, N., Kramer, J. H., Rosen, H. J., Gorno-Tempini, M. L., Rankin, K., Miller, B. L., and Weiner, M. W. (2007). Different regional patterns of cortical thinning in alzheimer's disease and frontotemporal dementia. *Brain*, 130:1159–1166.

Duchesne, S., Caroli, A., Geroldi, C., Frisoni, G. B., and Collins, D. L. (2005). Predicting clinical variable from mri features: application to mmse in mci. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2005*, pages 392–399. Springer.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1361.

Fan, Y., Kaufer, D., and Shen, D. (2010). Joint estimation of multiple clinical variables of neurological diseases from imaging patterns. In *Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on*, pages 852–855. IEEE.

Fjell, A. M., Walhovd, K. B., Fennema-Notestine, C., McEvoy, L. K., Hagler, D. J., Holland, D., Brewer, J. B., and Dale, A. M. (2010). Csf biomarkers in prediction of cerebral and clinical change in mild cognitive impairment and alzheimer's disease. *The Journal of Neuroscience*, 30:2088–2101.

Folstein, M. F., Folstein, S. E., and McHugh, P. R. (1975). mini-mental state: a practical method for grading the cognitive state of patients for the clinician. *Journal of Psychiatric Research*, 12(3):189–198.

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

Greenshtein, E. (2006). Best subset selection, persistence in high-dimensional statistical learning and optimization under l1 constraint. *The Annals of Statistics*, 34(5):2367–2386.

Hampson, M., Peterson, B. S., Skudlarski, P., Gatenby, J. C., and Gore, J. C. (2002). Detection of functional connectivity using temporal correlations in mr images. *Human Brain Mapping*, 15(4):247–262.

Hastie, T., Tibshirani, R., Sherlock, G., Eisen, M., Brown, P., and Botstein, D. (1999). Imputing missing data for gene expression arrays. Technical report, Stanford University.

He, X. and Niyogi, P. (2004). Locality preserving projections. In *Neural Information Processing Systems*, volume 16, page 153.

Hebert, L. E., Beckett, L. A., Scherr, P. A., and Evans, D. A. (2001). Annual incidence of alzheimer disease in the united states projected to the years 2000 through 2050. *Alzheimer Disease & Associated Disorders*, 15:169–173.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Jack, C., Petersen, R. C., Xu, Y. C., OBrien, P. C., Smith, G. E., Ivnik, R. J., Boeve, B. F., Waring, S. C., Tangalos, E. G., and Kokmen, E. (1999). Prediction of ad with mri-based hippocampal volume in mild cognitive impairment. *Neurology*, 52(7):1397–1397.

Kabani, N., MacDonald, D., Holmes, C., and Evans, A. (1998). A 3d atlas of the human brain. *NeuroImage*, 7:S717.

Kim, S., Pan, W., and Shen, X. (2013). Network-based penalized regression with application to genomic data. *Biometrics*, 69:582–593.

Kim, S. and Xing, E. P. (2009). Statistical estimation of correlated genome associations to a quantitative trait network. *PLoS Genetics*, 5(8):e1000587.

Knight, K. and Fu, W. (2000). Asymptotics for lasso-type estimators. *The Annals of Statistics*, pages 1356–1378.

Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411.

Ledoux, M. and Talagrand, M. (1991). *Probability in Banach Spaces: isoperimetry and processes*, volume 23. Springer.

Lee, H., Lee, D. S., Kang, H., Kim, B.-N., and Chung, M. K. (2011). Sparse brain network recovery under compressed sensing. *Medical Imaging, IEEE Transactions on*, 30(5):1154–1165.

Lee, W. and Liu, Y. (2012). Simultaneous multiple response regression and inverse covariance matrix estimation via penalized gaussian maximum likelihood. *Journal of Multivariate Analysis*, 111:241–255.

Li, C. and Li, H. (2008). Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, 24(9):1175–1182.

Lounici, K., Pontil, M., Tsybakov, A. B., and van de Geer, S. (2009). Taking advantage of sparsity in multi-task learning. *arXiv preprint arXiv:0903.1468*.

Lounici, K., Pontil, M., Van De Geer, S., Tsybakov, A. B., et al. (2011). Oracle inequalities and optimal inference under group sparsity. *The Annals of Statistics*, 39(4):2164–2204.

Mazumder, R., Hastie, T., and Tibshirani, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. *The Journal of Machine Learning Research*, 11:2287–2322.

McEvoy, L. K., Fennema-Notestine, C., Roddey, J. C., Jr, D. J. H., Holland, D., Karow, D. S., Pung, C. J., Brewer, J. B., and Dale, A. M. (2009). Alzheimer disease: quantitative structural neuroimaging for detection and prediction of clinical and structural changes in mild cognitive impairment. *Radiology*, 251:195–205.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, pages 1436–1462.

Mevik, B.-H. and Wehrens, R. (2007). The pls package: principal component and partial least squares regression in r. *Journal of Statistical Software*, 18(2):1–24.

Misra, C., Fan, Y., and Davatzikos, C. (2009a). Baseline and longitudinal patterns of brain atrophy in mci patients, and their use in prediction of short-term conversion to ad: results from adni. *NeuroImage*, 44:1414–1422.

Misra, C., Fan, Y., and Davatzikos, C. (2009b). Baseline and longitudinal patterns of brain atrophy in mci patients, and their use in prediction of short-term conversion to ad: results from adni. *Neuroimage*, 44(4):1415–1422.

Morris, J. C., Storandt, M., Miller, J. P., McKeel, D. W., Price, J. L., Rubin, E. H., and Berg, L. (2001). Mild cognitive impairment represents early-stage alzheimer disease. *Archives of Neurology*, 58(3):397.

Mungas, D. (1991). In-office mental status testing: A practical guide. *Geriatrics*, 46(7).

Nardi, Y. and Rinaldo, A. (2008). On the asymptotic properties of the group lasso estimator for linear models. *Electronic Journal of Statistics*, 2:605–633.

Obozinski, G., Jacob, L., and Vert, J.-P. (2011a). Group lasso with overlaps: the latent group lasso approach. *arXiv preprint arXiv:1110.0413*.

Obozinski, G., Taskar, B., and Jordan, M. I. (2010). Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252.

Obozinski, G., Wainwright, M. J., Jordan, M. I., et al. (2011b). Support union recovery in high-dimensional multivariate regression. *The Annals of Statistics*, 39(1):1–47.

Pan, W., Xie, B., and Shen, X. (2010). Incorporating predictor network in penalized regression with application to microarray data. *Biometrics*, 66(2):474–484.

Percival, D. (2012). Theoretical properties of the overlapping groups lasso. *Electronic Journal of Statistics*, 6:269–288.

Reinsel, G. C. and Velu, R. P. (1998). *Multivariate reduced-rank regression*. Springer.

Rosen, W. G., Mohs, R. C., and Davis, K. L. (1984). A new rating scale for alzheimer's disease. *The American Journal of Psychiatry*.

Rothman, A. J., Levina, E., and Zhu, J. (2010). Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4):947–962.

Ryali, S., Chen, T., Supekar, K., and Menon, V. (2012). Estimation of functional connectivity in fmri data using stability selection-based sparse partial correlation with elastic net penalty. *NeuroImage*, 59(4):3852–3861.

Schott, J., Bartlett, J., Barnes, J., Leung, K., Ourselin, S., and Fox, N. (2010). Reduced sample sizes for atrophy outcomes in alzheimer's disease trials: baseline adjustment. *Neurobiology of Aging*, 31:1452–1462.

Shen, D. and Davatzikos, C. (2002). Hammer: Hierarchical attribute matching mechanism for elastic registration. *IEEE Transactions on Medical Imaging*, 21(11):1421–1439.

Sled, J. G., Zijdenbos, A. P., and Evans, A. C. (1998). A nonparametric method for automatic correction of intensity nonuniformity in mri data. *IEEE Transactions on Medical Imaging*, 17(1):87–97.

Sohn, K.-A. and Kim, S. (2012). Joint estimation of structured sparsity and output structure in multiple-output regression via inverse-covariance regularization. In *International Conference on Artificial Intelligence and Statistics*, pages 1081–1089.

Stonnington, C. M., Chu, C., Klöppel, S., Jack Jr, C. R., Ashburner, J., and Frackowiak, R. S. (2010). Predicting clinical scores from magnetic resonance scans in alzheimer's disease. *NeuroImage*, 51(4):1405–1413.

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S., and Mesirov, J. P. (2005). Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550.

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

Turlach, B. A., Venables, W. N., and Wright, S. J. (2005). Simultaneous variable selection. *Technometrics*, 47(3):349–363.

Van Der Vaart, A. W. and Wellner, J. A. (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer.

Villa, S., Rosasco, L., Mosci, S., and Verri, A. (2014). Proximal methods for the latent group lasso penalty. *Computational Optimization and Applications*, 58(2):381–407.

Wang, H., Li, G., and Jiang, G. (2007). Robust regression shrinkage and consistent variable selection through the lad-lasso. *Journal of Business & Economic Statistics*, 25(3):347–355.

Wang, Y., Fan, Y., Bhatt, P., and Davatzikos, C. (2010). High-dimensional pattern regression using machine learning: From medical images to continuous clinical variables. *NeuroImage*, 50(4):1519–1535.

Wang, Y., Nie, J., Yap, P.-T., Shi, F., Guo, L., and Shen, D. (2011). Robust deformable-surface-based skull-stripping for large-scale studies. *Medical Image Computing and Computer-Assisted Intervention*, 6893:635–642.

Witten, D. and Tibshirani, R. (2011). scout: Implements the scout method for covariance-regularized regression. *R package version*, 1(3).

Witten, D. M. and Tibshirani, R. (2009). Covariance-regularized regression and classification for high dimensional problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(3):615–636.

Yang, S., Yuan, L., Lai, Y.-C., Shen, X., Wonka, P., and Ye, J. (2012). Feature grouping and selection over an undirected graph. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 922–930. ACM.

Yang, Y. and Zou, H. (2013). *gglasso: Group Lasso Penalized Learning Using A Unified BMD Algorithm*. R package version 1.1.

Yu, G., Liu, Y., Thung, K.-H., and Shen, D. (2014). Multi-task linear programming discriminant analysis for the identification of progressive mci individuals. *PloS one*, 9(5):e96458.

Yuan, L., Wang, Y., Thompson, P. M., Narayan, V. A., and Ye, J. (2012). Multi-source feature learning for joint analysis of incomplete multiple heterogeneous neuroimaging data. *NeuroImage*, 61(3):622–632.

Yuan, M., Ekici, A., Lu, Z., and Monteiro, R. (2007). Dimension reduction and coefficient estimation in multivariate linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(3):329–346.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B*, 68:49–67.

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35.

Zhang, C.-H. (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942.

Zhang, D. and Shen, D. (2012). Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in alzheimer's disease. *NeuroImage*, 59(2):895–907.

Zhang, H. H., Liu, Y., Wu, Y., Zhu, J., et al. (2008). Variable selection for the multicategory svm via adaptive sup-norm regularization. *Electronic Journal of Statistics*, 2:149–167.

Zhang, W., Wan, Y.-w., Allen, G. I., Pang, K., Anderson, M. L., and Liu, Z. (2013). Molecular pathway identification using biological network-regularized logistic models. *BMC Genomics*, 14(Suppl 8):S7.

Zhang, Y., Brady, M., and Smith, S. (2001). Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1):45–57.

Zhao, P. and Yu, B. (2006). On model selection consistency of lasso. *The Journal of Machine Learning Research*, 7:2541–2563.

Zhou, S., van de Geer, S., and Bühlmann, P. (2009). Adaptive lasso for high dimensional regression and gaussian graphical modeling. *arXiv preprint:0903.2515*.

Zhu, Y., Shen, X., and Pan, W. (2013). Simultaneous grouping pursuit and feature selection over an undirected graph. *Journal of the American Statistical Association*, 108(502):713–725.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B*, 67:301–320.