SUPERVISED LEARNING FOR COMPLEX DATA

Haodong Wang

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
2022

Approved by:

Yufeng Liu

Quefeng Li

Nico Fraiman

Quoc Tran-Dinh

Zhengwu Zhang

©2022
Haodong Wang
ALL RIGHTS RESERVED

ABSTRACT

HAODONG WANG: SUPERVISED LEARNING FOR COMPLEX DATA
(Under the direction of Yufeng Liu and Quefeng Li)

Supervised learning problems are commonly seen in a wide range of scientific fields such as medicine and neuroscience. Given data with predictors and responses, an important goal of supervised learning is to find the underlying relationship between predictors and responses for future prediction. In this dissertation, we propose three new supervised learning approaches for the analysis of complex data. For the first two projects, we focus on block-wise missing multi-modal data which contain samples with different modalities. In the first project, we study regression problems with multiple responses. We propose a new penalized method to predict multiple correlated responses jointly, using not only the information from block-wise missing predictors but also the correlation information among responses. In the second project, we study regression problems with censored outcomes. We propose a penalized Buckley-James method that can simultaneously handle block-wise missing covariates and censored outcomes. For the third project, we analyze data streams under reproducing kernel Hilbert spaces. Specifically, we develop a new supervised learning method to learn the underlying model with limited storage space, where the model may be non-stationary. We use a shrinkage parameter and a data sparsity constraint to balance the bias-variance tradeoff, and use random feature approximation to control the storage space.

ACKNOWLEDGEMENTS

Writing this document would have been impossible without the enormous support and encouragement of many people who stood by me during my time at UNC. I would like to express my gratitude to all of them.

I would first like to express my appreciation and deep gratitude to my advisors Professors Yufeng Liu and Quefeng Li for being great advisors and very kind people to work with. I have learned so many things from them both in the academic and personal front. They have been there to share ideas and at the same time encourage me to pursue my own. They have been extremely enthusiastic and supportive throughout my Ph.D. years. I have been very fortunate to have advisors like them.

I would like to convey my sincere thanks to the other members of my dissertation committee: Professors Nicolas Fraiman, Quoc Tran-Dinh, and Zhengwu Zhang, for their time, support, guidance, and insightful comments on my dissertation. Their suggestions and instructions have enabled me to assemble and finish the dissertation effectively.

Last but not least, I am very grateful to my friends and my family for their encouragement and support throughout writing this dissertation and my life in general.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
Introduction

Fast development of modern technology makes it possible to generate and store large-scale and diverse data. Among different types of data, complex data with multiple modalities and censored outcomes are increasingly prevalent across various scientific fields, including genetics and neuroscience. Such data call for efficient statistics and machine learning tools for data analysis. Besides static data, due to the unprecedented speed and volume of generated raw data in many applications, one may need to analyze streaming data in practice, such as in finance and business. This dissertation investigates several supervised learning techniques for multi-modal and streaming data.

In this chapter, we first provide some background knowledge and literature review on machine learning algorithms useful in subsequent chapters and then briefly introduce our problems and main contributions. In Section 1.1, we review some existing multi-response linear regression methods in the literature. In Section 1.2, we describe some literature on the analysis of survival data. In Section 1.3, some existing supervised learning methods for analyzing data streams are discussed. In Section 1.4, we provide an outline of our main contributions in this dissertation.

1.1 Multi-response linear regression methods

In many applications, we may have multiple response variables with the same set of predictors. Multi-response regression is a useful regression technique to solve this problem. In particular, with a q-dimensional vector of response variables for the i–th sample, $\mathbf{Y}_i = (Y_{i1}, \ldots, Y_{iq})^\top$, the multi-response regression model can be formulated as follows,

$$\mathbf{Y}_i = \mathbf{B}^\top \mathbf{X}_i + \boldsymbol{\epsilon}_i \quad \text{for } i = 1, \ldots, n,$$

where $\mathbf{X}_i \in \mathbb{R}^p$ is the vector of predictors for the i-th subject, $\mathbf{B}$ is a $p \times q$ matrix of regression coefficients, and $\boldsymbol{\epsilon}_i$ denotes a q-dimensional error vector for the i-th sample.

The standard approach to estimate the regression parameter matrix $\mathbf{B}$ is to regress each response variable separately on the same set of predictors. All single response regression procedures can be applied to each response separately. For example, one can apply the ordinary least-squares method to each response separately by solving

$$\min_{\boldsymbol{B}_j} \sum_{i=1}^{n} \left(Y_{ij} - \mathbf{X}_i^\top \mathbf{B}_j \right)^2, \quad \text{for } j = 1, \ldots, q,$$

where $\mathbf{B}_j$ is the j-th column of $\mathbf{B}$, and Y_{ij} is the j–th response of the i-th subject. Using some simple linear algebra, it can be shown that the above problem is equivalent to solving the following optimization problem

$$\min_{\mathbf{B}} \operatorname{tr} \left[(\mathbf{Y} - \mathbf{X}\mathbf{B})^T (\mathbf{Y} - \mathbf{X}\mathbf{B}) \right], \tag{1.1}$$

where $\mathbf{Y} = (\mathbf{Y}_1, \ldots \mathbf{Y}_n)^\top$ is the $n \times q$ response matrix and $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_n)^\top$ is the $n \times p$ predictor matrix (Yuan et al., 2007). Although this model is simple to implement, this approach may not be optimal since it does not utilize the joint information among response variables.

To utilize the correlation information among response variables, Breiman and Friedman (1997) proposed an approach called Curd and Whey (C&W). Their method predicts the multiple responses with an optimal linear combination of the ordinary least-squares estimators. In particular, the C&W procedure starts with fitting q separate ordinary least-squares models. Denote the resulting predictor as $\hat{\mathbf{Y}}^{OLS}$. Then the C&W method tries to find another predictor $\tilde{\mathbf{y}} = \hat{\mathbf{Y}}^{OLS}\mathbf{W}$ with an optimal $q \times q$ weight matrix $\mathbf{W}$ so that

$$\mathbf{E}\left\{ \left(Y_j - \left(\hat{\mathbf{Y}}^{OLS}\mathbf{W} \right)_j \right) \right\}^2 \leq \mathbf{E}\left\{ \left(Y_j - \left(\hat{\mathbf{Y}}^{OLS} \right)_j \right) \right\}^2, \quad j = 1, \ldots, q.$$

In other words, $\mathbf{W}$ reduces the mean-squared prediction error for each response. They showed that $\mathbf{W}$ can be obtained by canonical analysis, and their method can outperform separate univariate regression approaches (1.1) when there are correlations among the response variables.

Some other multi-response regression models have been proposed in the regularization framework (Turlach et al., 2005; Yuan et al., 2007). These approaches impose a constraint on the param-

eters to regularize the estimators. In particular, they proposed to solve the following optimization problem

$$\min_{\mathbf{B}} \operatorname{tr}\left[(\mathbf{Y}-\mathbf{XB})^T(\mathbf{Y}-\mathbf{XB})\right] \quad \text{subject to} \quad J(\mathbf{B}) \leq t,$$

where $J(\mathbf{B})$ is a constraint on $\mathbf{B}$ and $t \geq 0$ is a tuning parameter for the constraint. Without any constraint, i.e. when $t = \infty$, the objective function is identical to (1.1) and consequently the method becomes equivalent to the separate least-squares approach. However, by imposing a constraint, we can achieve shrinkage for the resulting estimator. In particular, Yuan et al. (2007) proposed a method performing factor estimation and selection. To encourage sparsity among singular values of the regression parameter matrix, they let $J(\mathbf{B}) = \sum_{i=1}^{\min(p,q)} \sigma_i(\mathbf{B})$, where $\sigma_i(\mathbf{B})$ is the i-th singular value of $\mathbf{B}$. As a result, their method achieves dimension reduction in $\mathbf{B}$. A similar approach to handle multi-response regression is to use the reduced-rank regression by Izenman (1975) to achieve rank reduction. Reduced-rank regression (RRR) introduces a rank constraint on $\mathbf{B}$, namely $J(\mathbf{B}) = \operatorname{rank}(\mathbf{B}) \leq t$, where t is the maximal allowed rank of $\mathbf{B}$. In addition to the regularization purpose, it can also be used as a dimension reduction and data exploration method. If many predictors and responses are available, then RRR constructs "latent factors" in the predictor space for explaining the variance of predictors. This method is also known as redundancy analysis in ecology (Legendre and Anderson, 1999). Turlach et al. (2005) proposed another constraint function, $J(\mathbf{B}) = \sum_{j=1}^{p} \max\left(|\beta_{j1}|, \ldots, |\beta_{jm}|\right)$. By imposing the max-$\ell_1$ penalty, their method can select a common subset of explanatory variables for predicting multiple response variables.

1.2 Statistical analysis of survival data

In this section, we describe some supervised learning algorithms for censored survival data. Survival analysis is an important area of statistical research. One important type of survival analysis is the study of time to event data, in which the response variable is the time until a specific event of interest occurs (Kleinbaum, 1996). Such data are commonly seen in many fields such as biology, medicine, public health, epidemiology, and economics. The most prominent challenge of time to event data is that the response, which is the time until some specified event, cannot always be fully observed. Instead, the response may be right-censored, and consequently the actual response may

not be observed. In particular, the failure time $T \in \mathbb{R}$ is right censored at a censoring point C when $T > C$, and the outcome $Y \in \mathbb{R}$ we observe is recorded as being equal to the censoring point. In contrast, when $T \leq C$, the outcome Y is recorded as the actual failure time. For example, when a patient has been given a certain treatment, a right-censoring time might arise when the patient is still alive at the end of the study or terminate the study due to other reasons (Miller 1976).

The accelerated failure time (AFT) model is one of the most commonly used models in survival data analysis, which assumes that the logarithm of the failure time is linearly related to the covariates. Let $T \in \mathbb{R}$ be the failure time until a certain event of interest occurs, and $\mathbf{X} \in \mathbb{R}^p$ be the p-dimensional covariates vector. The accelerated failure time model assumes that

$$\log(T) = \boldsymbol{X}^\top \boldsymbol{\beta} + \epsilon,$$

where $\boldsymbol{\beta} \in \mathbb{R}^p$ is the p-dimensional regression coefficient vector, and $\epsilon \in \mathbb{R}$ is the random noise. In general, no specific parametric form is assumed for the distribution function of the error term ϵ. Two general estimation strategies to handle censored responses in the AFT model include extensions of least-squares estimators through missing data techniques (Buckley and James, 1979; Koul et al., 1981; Miller and Halpern, 1982; Lai and Ying, 1991) and rank-based methods (Prentice, 1978; Tsiatis, 1990; Lai and Ying, 1991).

Various rank-based methods have been well-studied for the AFT model with right-censored data. Prentice (1978) first proposed the rank-based estimators based on the well-known weighted log-rank statistics. Tsiatis (1990) studied the asymptotic properties of the rank-based estimators. Jin et al. (2003) provided a reliable and accurate estimation procedure by using linear programming techniques to compute the estimator proposed by Gehan (1965), a special version of the weighted log-rank estimator.

By using the conditional expectation to impute the censored outcomes, Buckley and James (1979) proposed a least-squares method to estimate the regression coefficient vector $\boldsymbol{\beta}$ in the model for the AFT model with right-censored data. Lai and Ying (1991) studied the asymptotic properties of the Buckley-James estimators. Jin et al. (2006) proposed an iterative algorithm to compute the Buckley-James estimator. To incorporate high-dimensional covariates, various variable selection

techniques have been applied to the Buckley–James estimators for the AFT model (Datta et al., 2007; Johnson, 2009; Wang et al., 2008).

1.3 Supervised learning methods for data streams

Data are generated at an unprecedented rate and scale these days. The field of streaming data analysis has emerged as a result of new data collection and storage technologies (Anderson, 2008; Wu et al., 2019; Hoi et al., 2021). Streaming data include high-throughput recordings that collect large volumes of observations sequentially and continuously over time. The instances are in an ordered sequence and typically arrive quickly and they may not be completely stored for future study and analysis. In this context, regression models need to be continuously updated as new data arrive. In addition, due to the vast amount of data, it is impossible to store the all the data. Therefore, it is desirable for us to be able to incrementally learn the model without access of the historical data.

Predictive models using such streaming data are widely applied in many fields, such as air pollution monitoring (Hyde et al., 2017), detection of traffic congestion (Arasu et al., 2004), disease surveillance (Althouse et al., 2015), and recommendation systems (Ta et al., 2016).

In this section, we describe a family of supervised learning algorithms for the analysis of data streams.

1.3.1 Online gradient descent

We first describe a linear supervised learning algorithm, online gradient descent, for data streams. Consider a sequence of instances, $\boldsymbol{x}^t \in \mathbb{R}^p$, where t denotes the time and p is the dimensionality of $\boldsymbol{x}^t$, and let y^t be the response. At time t, an instance $\boldsymbol{x}^t$ is observed. Then the model uses $\hat{y}_t = \boldsymbol{x}^{t\top}\hat{\boldsymbol{\beta}}_{t-1}$ to make a prediction, where the coefficient $\hat{\boldsymbol{\beta}}_{t-1}$ is the estimator obtained by the model at time $t-1$. After making the prediction, the true response y_t becomes available. Then one can use the true response y_t to calculate the loss $l(\hat{y}^t, y^t)$. Finally the algorithm updates the coefficient from $\hat{\boldsymbol{\beta}}_{t-1}$ to $\hat{\boldsymbol{\beta}}_t$.

Supervised learning on data streams can be reformulated as an online convex optimization problem. The online gradient descent algorithm (OGD) (Zinkevich, 2003) can be viewed as an

online version of the stochastic gradient descent algorithm (SGD) in convex optimization, which is one of the simplest and most popular methods for convex optimization. At every iteration, based on the loss occurred on the t-th sample $\boldsymbol{x}^t \in \mathbb{R}^p$, the algorithm updates the current model to a new model in the direction of the gradient of the current loss function. A projection may be needed to ensure that the estimated parameters satisfy all constraints on parameters. Algorithm 1 below summarizes the major steps of OGD, where $\eta_t > 0$ is the learning rate parameter.

Algorithm 1: Online Gradient Descent

Initialize $\boldsymbol{\beta}$ with some $\hat{\boldsymbol{\beta}}_0$;

for $t = 1, 2, \ldots T$ **do**

Observe $\boldsymbol{x}^t \in \mathbb{R}^p$, predict $\hat{y}^t$ using $\hat{\boldsymbol{\beta}}_{t-1}$;

Observe $y^t \in \mathbb{R}$, obtain loss $l(\hat{y}_t, y_t)$;

Update $\hat{\boldsymbol{\beta}}_t = \Pi_S(\hat{\boldsymbol{\beta}}_{t-1} - \eta_t \hat{\nabla} l(\hat{y}_t, y_t))$, where $\Pi_S(\cdot)$ is the projection function to constrain the updated model to lie in the feasible domain S of the parameters.

end

OGD is simple and easy to implement, but the projection step may sometimes be computationally intensive, depending on specific tasks.

1.3.2 Online kernel learning algorithm

Classical OGD algorithms focus on linear problems. For many supervised learning problems, however, the response may have a nonlinear relationship with the predictors. Hence a nonlinear model is needed. Online kernel learning algorithms fit the model in a reproducing kernel Hilbert space (RKHS) $\mathcal{H} = \{f | f : \mathbb{R}^p \to \mathbb{R}\}$ (Aronszajn, 1950). Here the RKHS with the reproducing kernel function $K(\cdot, \cdot) : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ satisfies the following properties:

- K has the reproducing property $\langle f, K(\boldsymbol{x}, \cdot) \rangle = f(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathbb{R}^p$,
- $\mathcal{H}$ is the closure of the span of all $K(\boldsymbol{x}, \cdot)$ with $\boldsymbol{x} \in \mathbb{R}^p$.

By the Representer theorem (Kimeldorf and Wahba, 1971), the optimal solution of the kernel optimization problem in RKHS involving some loss functions lies in the span of kernels. Consequently, the goal of a typical online kernel learning algorithm is to learn the kernel-based predictive model $f_t(\boldsymbol{x})$ for predicting the response of a new instance $\boldsymbol{x}^t \in \mathbb{R}^p$ as $f_t(\boldsymbol{x}^t) = \sum_{j=1}^{t-1} \alpha_j^t K(\boldsymbol{x}^j, \boldsymbol{x}^t)$, where α_j^t

is the coefficient of the model. We define support vector (SV) at time t as the set $\mathrm{SV}_t = \{j : \alpha_j^t \neq 0\}$. Then, the model can be written as $f_t(\boldsymbol{x}^{T+1}) = \sum_{j \in SV_t} \alpha_j^t K(\boldsymbol{x}^j, \boldsymbol{x}^t)$. We use the notation $|SV|$ to denote the size of the SV set. In the literature, different online kernel methods have been proposed. We begin by introducing the simplest one, that is, the kernelized online gradient descent (Kivinen et al., 2004), which extends Algorithm 1 by using the kernel trick. Algorithm 2 below outlines the kernelized online gradient descent, where $\eta_t > 0$ is the learning rate parameter.

Algorithm 2: Kernelized Online Gradient Descent

Initialize f with some $\hat{f}_0$;

for $t = 1, 2, \ldots T$ **do**

Observe $\boldsymbol{x}^t \in \mathbb{R}^p$, predict $\hat{y}^t$ using $\hat{f}_{t-1}(\boldsymbol{x}^t)$;

Observe $y^t \in \mathbb{R}$, obtain loss $l(\hat{y}_t, y_t)$;

Update $SV_t = SV_{t-1} \cup (\boldsymbol{x}^t, y^t)$;

Update $\hat{f}_t = \Pi_S(\hat{f}_{t-1} - \eta_t \hat{\nabla} l(f_{t-1}(\boldsymbol{x}^t), y^t)$, where $\Pi_S(\cdot)$ is the projection function to constrain the updated model to lie in the feasible domain S of the parameters.

end

Although online kernel learning described in Algorithm 2 enjoys the clear advantage of flexibility over linear models, it falls short in some critical drawbacks. One crucial issue is that the number of support vectors grows linearly with increasing computational and space complexity. To address this challenge, a family of algorithms, "budget online kernel learning", have been proposed to bound the number of SVs with a fixed budget B by using budget maintenance strategies.

One of the strategies is the "SV removal", which maintains the budget simply and efficiently. It first updates the SV set by adding a new SV whenever necessary. If the SV size exceeds the budget, the SV removal method discards one of the existing SVs and updates the SV set accordingly. The key step of SV removal is to find one of the existing SVs to be removed by minimizing the impact of the resulting model. A straightforward way is to randomly discard one of the existing SVs uniformly with probability $\frac{1}{B}$, as adopted by RBP (Cavallanti et al., 2010) and BOGD (Zhao et al., 2012). Instead of choosing randomly, in "Forgetron" (Dekel et al., 2005), the algorithm discards the oldest SV by assuming an older SV is less representative for the distribution of fresh training data streams. Although these methods are simple and highly efficient, the assumption may not be reasonable in practice for satisfactory performance.

Another strategy is the "SV projection", which was initially introduced by Orabona et al. (2009), where two new algorithms, Projectron and Projectron++, were proposed. These two methods significantly outperformed the previous SV removal based algorithms such as RBP and Forgetron. The SV projection methods follow the setting of SV removal and identify a support vector for removal during the update of the model. It then chooses a subset of SVs as the projection base. Following this, a linear combination of kernels in the projection base is used to approximate the removed SV.

1.4 Main contributions and outline

In this dissertation, we propose several new flexible regression methods for complex data. In particular, the following chapters are organized as follows:

- In Chapter 2, we consider a multi-response regression model for block-wise missing data. The main contribution of this method is to allow missing values in both responses and predictors and correlations among reactions. This method can also handle the case that no subject has complete observation, while most traditional methods do not allow this. Our method includes two steps. The first step is to estimate each element of the covariance and cross-covariance matrices using all available observations without imputation. The second step is to use a penalized likelihood approach to simultaneously estimate the sparse regression coefficient matrix and the precision matrix of the error terms. We show that this method has estimation and model selection consistency under the high-dimensional setting in terms of theoretical studies. Numerical studies and the Alzheimer's Disease Neuroimaging Initiative (ADNI) data application also confirm that the proposed method performs competitively for block-wise missing data. The proofs of several analysis results of this proposed model are given in the Appendix A.

- In Chapter 3, we consider the problem of parameter estimation and variable selection for the semi-parametric accelerated failure time model for high-dimensional block-missing multi-modal data with censored outcomes. We propose a penalized Buckley-James method that simultaneously handles block-wise missing covariates and censored outcomes. This method can perform both variable selection and parameter estimation. The proposed method is

evaluated by simulations and applied to the multi-modal neuroimaging dataset from the ADNI with meaningful results.

- In Chapter 4, we consider a supervised learning model for analyzing data streams in Reproducing Kernel Hilbert Spaces (RKHS). An adaptive supervised learning model is proposed for data streams in RKHS with limited storage space. We use random feature approximation to control the storage space and training time. In addition, our model uses the data sparsity constraint to balance the bias-variance tradeoff of the model and control the error introduced by random feature approximation. Our method can also handle non-stationary models. Numerical studies with simulated and real data confirm that the proposed method performs competitively for data streams in both stationary and non-stationary cases.

CHAPTER 2

Multi-response Regression for Block-missing Multi-modal Data without Imputation

2.1 Introduction

With the prevalence of large-scale multi-modal data in various scientific fields, multi-response linear regression has attracted growing research attentions in statistics and machine learning communities (Rothman et al., 2010; Lee and Liu, 2012; Loh et al., 2013). While linear regression with a scalar response has been well studied, many applications may have a vector as the response. In particular, multi-response models have wide applications in scientific fields, especially for biological problems (Kim et al., 2012). For example, for multi-tissue joint expression quantitative trait loci (eQTL) mapping (Molstad et al., 2021), researchers consider predicting gene expression values in multiple tissues simultaneously by using a weighted sum of eQTL genotypes. Separate prediction for each tissue can be inefficient since same genes in different tissues are often correlated due to the shared genetic variants or other unmeasured common regulators. In order to use data from all tissues simultaneously, a joint eQTL modeling has been proposed to take cross-tissue expression dependence into account (Molstad et al., 2021).

To apply variable selection methods for multi-response problems, one could separately fit each response via a single-response model. There are many well-studied variable selection methods for the single-response linear regression model such as LASSO (Tibshirani, 1996). Although it is simple to apply a single-response linear regression method for each response separately, such a procedure neglects the dependency structure among responses. By incorporating the dependency structure of the response vector, one may obtain a more efficient multi-response linear regression approach in terms of estimation and prediction.

To handle multi-response regression problems, a well-known approach, the Curds and Whey, was proposed by Breiman and Friedman (1997) to improve the prediction performance by utilizing

dependency among responses. Specifically, they first fit a single-response regression model for each response and then modify the predicted values from those regressions by shrinking them using canonical correlations between the response variables and the predictors. Another popular approach to handle multi-response regression is to use dimension reduction. In particular, reduced rank regression (Izenman, 1975) minimizes the least squares criterion subject to the constraint on the rank of regression parameter matrix. Yuan et al. (2007) further extended this method for the high dimensional setting. Their idea is to obtain dimension reduction by encouraging sparsity among singular values of the parameter matrix. Although these methods may achieve better prediction performance than the separate univariate regression, they did not address the problem of variable selection.

In order to handle correlated responses together with variable selection, the precision matrix of response vector given predictors and the regression parameter matrix can be estimated separately or simultaneously (Lee and Liu, 2012). For separate estimation, Cai et al. (2013) used a constrained ℓ_1 minimization that can be treated as a multivariate extension of the Dantzig selector to estimate the regression parameter matrix. After removing the regression effect using the estimated regression parameter matrix, the precision matrix of the error terms can be estimated accordingly. One potential drawback of this indirect method is that it ignores the relationship between different responses given predictors when estimating the regression parameter matrix. In order to use all information more efficiently, it can be desirable to estimate the precision matrix and regression parameter matrix simultaneously. In the literature, various joint estimation techniques were studied by Rothman et al. (2010), Yin and Li (2011) and Lee and Liu (2012). They formulated the multi-response regression problem in a penalized log-likelihood framework, so that the parameter and precision matrices can be estimated simultaneously. Using a similar idea, Chen et al. (2018) proposed an estimation procedure to estimate the parameter and precision matrices simultaneously based on the generalized Dantzig selector.

Despite a lot of development for multi-response linear regression, most existing methods only deal with complete data without missing entries. However, many practical data are incomplete, especially for multi-modal data. For instance, in the study of Alzheimer's Disease (AD), data from different sources are collected. This includes magnetic resonance imaging (MRI) of the brain, positron emission tomography (PET) and cerebrospinal fluid (CSF). In practice, observations of a

certain modality can be missing completely due to patient dropouts or other practical issues. This leads to a block-wise missing data structure. It is important to integrate data from all modalities to improve model prediction and variable selection.

To handle incomplete multi-modal data, one may simply remove those observations with missing entries. However, such a procedure may greatly reduce the number of observations and lead to loss of information. Another approach is to perform data imputation. Existing imputation methods, such as matrix completion (Johnson, 1990) algorithms may possibly be unstable when the missing values happen in blocks. In order to deal with multi-modal block-wise missing data, Yu et al. (2020) proposed a new direct sparse regression procedure using covariance from block-missing multi-modal data (DISCOM). They first used all available information to estimate the covariance matrix of the predictors and the cross-covariance vector between the predictors and the response variable. Based on the estimated covariance matrix and the estimated cross-covariance vector, they then used an extended Lasso-type estimator to estimate the coefficients. However, the DISCOM only considers single-response regression. Recently, Xue and Qu (2021) proposed the Multiple Block-wise Imputation (MBI) method for single-response regression when data are block-wise missing. They developed an estimating equation approach to accommodate block-wise missing patterns in multi-modal data. The method was shown to have high selection accuracy and low estimation error for single-response regression with block-wise missing data. However, since their imputation method requires analyzing all combinations of different blocks, it can be computationally expensive when the number of modalities is large.

In this paper, we consider a multi-response regression model for block-wise missing data. The main contribution of our method is to allow missing values in both responses and predictors and correlations among responses. This method can also handle the case that no subject has complete observations, while most traditional methods do not allow this. Our method includes two steps. The first step is to estimate each element of the covariance and cross-covariance matrices by using all available observations without imputation. The second step is to use a penalized approach to estimate the sparse regression coefficient matrix and the precision matrix of the error terms simultaneously. We show that this method has estimation and model selection consistency under the high-dimensional setting. Numerical studies and the ADNI data application also confirm that the proposed method performs competitively for block-wise missing data.

The remainder of the paper is organized as follows. In Section 2.2, we introduce the problem background and our model. In Section 2.3, we establish some theoretical properties of our proposed method. We present simulation studies and a multi-modal ADNI data example in Sections 2.4 and 2.5.

2.2 Methodology

2.2.1 Problem setup and notations

Consider the following multi-response linear regression model,

$$\mathbf{Y} = \mathbf{X}\mathbf{B}^* + \mathcal{E}, \tag{2.1}$$

where $\mathbf{B}^* = (b_{jk}) \in \mathbb{R}^{p\times q}$ is an unknown $p \times q$ parameter matrix, $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_n)^\top$ is the $n \times q$ response matrix, $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top$ is the $n \times p$ design matrix and $\mathcal{E} = (\boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_n)^\top$ is the $n \times q$ error matrix. We assume that $\{\mathbf{x}_i\}_{i=1}^n$ are i.i.d. realizations of a random vector $(X_1, \ldots, X_p)^\top$ with zero mean and covariance matrix $\mathbf{\Sigma}_{XX} = (\sigma_{ij}^{XX}) \in \mathbb{R}^{p\times p}$. We use $\mathbf{\Sigma}_{XY} = (\sigma_{ij}^{XY}) \in \mathbb{R}^{p\times q}$ to denote the cross-covariance matrix between $\mathbf{x}_i$ and $\mathbf{y}_i$. We assume that the predictors come from multiple modalities and there are p_k predictors in the k-th modality. In addition, $\mathbf{X}$ has block-missing values. That is, for one sample, its measurements in one modality can be entirely missing. We assume elements of $\mathbf{Y}$ can also be missing. The errors $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \ldots, \epsilon_{iq})^\top$ for $i = 1, \ldots, n$ are i.i.d. realizations from a random vector $\boldsymbol{\epsilon}$ with zero mean and covariance matrix $\mathbf{\Sigma}_\epsilon = (\sigma_{ij}^{EE}) \in \mathbb{R}^{q\times q}$. We let $\boldsymbol{C}^* = \mathbf{\Sigma}_\epsilon^{-1}$. Moreover, we further assume $\mathbf{x}_i$ and $\boldsymbol{\epsilon}_i$ are uncorrelated. Denote the support of $\mathbf{B}^*$ and $\mathbf{C}^*$ as $S_B = \{j : \mathrm{vec}(\mathbf{B}^*)_j \neq 0\}$ and $S_C = \{j : \mathrm{vec}(\mathbf{C}^*)_j \neq 0\}$, where "vec" is the vectorization by column operator. For a set S, we denote $|S|$ as its cardinality. Denote $s_B = |S_B|$, $s_C = |S_C|$ and $s = \max(s_B, s_C)$.

We employ the following notation throughout this article. The symbol $\mathbb{S}_+^{d\times d}$ is used to denote the sets of $d \times d$ symmetric positive-definite matrices. For a square matrix $\mathbf{C} = (c_{ii'}) \in \mathbb{R}^{p\times p}$, we denote its trace as $\mathrm{tr}(\mathbf{C}) = \sum_i c_{ii}$ and its diagonal matrix as $\mathrm{diag}(\mathbf{C})$. For a matrix $\mathbf{A} = (a_{ij}) \in \mathbb{R}^{p\times q}$, we define its entrywise ℓ_1–norm as $\|\mathbf{A}\|_1 = \sum_{i,j} |a_{ij}|$ and its entrywise ℓ_∞–norm as $\|\mathbf{A}\|_\infty = \max_{i,j} |a_{ij}|$. In addition, we define its matrix ℓ_1–norm as $\|\mathbf{A}\|_{L_1} = \max_j \sum_i |a_{ij}|$, matrix

ℓ_∞–norm as $\|\mathbf{A}\|_{L_\infty} = \max_i \sum_j |a_{ij}|$, the spectral norm as $\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2$, the Frobenius norm as $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} a_{ij}^2}$ and the number of nonzero elements as $\|\mathbf{A}\|_0 = \sum_{i,j} \mathbb{I}(a_{ij} \neq 0)$. Denote the largest and smallest eigenvalues of $\mathbf{A}$ by $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$ respectively. Denote the sub-matrix of $\mathbf{A}$ with row and column indices in I_1 and I_2 as $\mathbf{A}_{I_1 I_2}$. For a vector $\mathbf{v} \in \mathbb{R}^p$, denote $\mathbf{v}_{I_1}$ as the sub-vector of $\mathbf{v}$ with indices in I_1, $\|\mathbf{v}\|_1 = \sum_i |v_i|$, $\|\mathbf{v}\|_\infty = \max_i |v_i|$, $\|\mathbf{v}\|_{\min} = \min_i |v_i|$ and $\|\mathbf{v}\|_2 = \sqrt{\sum_i v_i^2}$. For a function $h(X)$, we use $\nabla_X h$ to denote a gradient or subgradient of h with respect to X, if it exists. Finally, we write $a_n \lesssim b_n$ if $a_n \leq c b_n$ for some constant c, and write $a_n \asymp b_n$ if $a_n \lesssim b_n$ and $b_n \lesssim a_n$.

2.2.2 Proposed Multi-DISCOM method

For the multi-response linear regression model (2.1), if one separately applies least squares estimation with the ℓ_1-norm penalty to each response, it essentially solves

$$\arg\min_{\mathbf{B}} \mathbb{E}\left[\|\mathbf{Y} - \mathbf{X}\mathbf{B}\|_F^2\right] + \lambda\|\mathbf{B}\|_1 = \arg\min_{\mathbf{B}} \operatorname{tr}\left(\tfrac{1}{2}\mathbf{B}^\top \mathbf{\Sigma}_{XX}\mathbf{B} - \mathbf{\Sigma}_{XY}^\top \mathbf{B}\right) + \lambda\|\mathbf{B}\|_1, \quad (2.2)$$

where λ is a tuning parameter. We refer to this method as the separate LASSO, whose solution is denoted as $\hat{\mathbf{B}}^{LASSO}$. However, such an approach fails to account for the correlations between responses and may lead to poor predictive performance (see, e.g., Breiman and Friedman (1997)). To produce a better estimator, we propose to incorporate $\mathbf{\Sigma}_\epsilon$ into the estimation of $\mathbf{B}^*$and solve the following problem:

$$\hat{\mathbf{B}}^0 = \arg\min_{\mathbf{B}} \operatorname{tr}\left[\mathbf{C}^*\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{C}^*\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{\mathbf{XX}}\mathbf{B} - \mathbf{2}\mathbf{C}^*\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{\mathbf{XY}}\right] + \lambda\|\mathbf{B}\|_1, \quad (2.3)$$

where λ is a tuning parameter, $\hat{\mathbf{\Sigma}}_{YY}$, $\hat{\mathbf{\Sigma}}_{XX}$ and $\hat{\mathbf{\Sigma}}_{XY}$ are some estimators of $\mathbf{\Sigma}_{YY}$, $\mathbf{\Sigma}_{XX}$ and $\mathbf{\Sigma}_{XY}$.

In practice, $\mathbf{C}^*$ is also unknown. It is natural to estimate $\mathbf{C}^*$ first, then plug the estimate $\hat{\mathbf{C}}$ into (2.3) and solve the following problem:

$$\hat{\mathbf{B}}^0 = \arg\min_{\mathbf{B}} \operatorname{tr}\left[\hat{\mathbf{C}}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \hat{\mathbf{C}}\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{\mathbf{XX}}\mathbf{B} - \mathbf{2}\hat{\mathbf{C}}\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{\mathbf{XY}}\right] + \lambda\|\mathbf{B}\|_1. \quad (2.4)$$

We refer this method as the two-step weighted LASSO. But as shown by the toy example in Section 2.2.2.1, the two-step weighted LASSO may perform worse than the separate LASSO in some problems.

In this article, we propose to estimate $\mathbf{B}^*$ and $\mathbf{C}^*$ simultaneously by solving the following optimization problem:

$$(\hat{\mathbf{B}}, \hat{\mathbf{C}}) = \arg \min_{\mathbf{C} \in \mathbb{S}_+^{q \times q}, \mathbf{B}} \operatorname{tr} \left[\mathbf{C}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{C}\mathbf{B}^\top \hat{\mathbf{\Sigma}}_{\mathbf{XX}} \mathbf{B} - 2\mathbf{C}\mathbf{B}^\top \hat{\mathbf{\Sigma}}_{\mathbf{XY}} \right] + \lambda_B \|\mathbf{B}\|_1 + \lambda_C \|\mathbf{C}\|_1 - \log \det \mathbf{C}, \tag{2.5}$$

where λ_B and λ_C are tuning parameters. When λ_C is large enough, Theorem 4 by Banerjee et al. (2008) implies that all off-diagonal entries in $\hat{\mathbf{C}}$ become zero. Then our proposed method (2.5) reduces to the separate LASSO (2.2). For a univariate response regression problem, our proposed method (2.5) reduces to the DISCOM algorithm (Yu et al., 2020). When there is no missing entries, our proposed method (2.5) reduces to the sparse conditional Gaussian graphical model introduced by Yin and Li (2011).

The toy example in Section 2.2.2.1 illustrates that our joint estimation model (2.5) has better estimation performance than the two-step weighted LASSO and the separate LASSO.

2.2.2.1 Toy example

For illustration, we consider a toy example similar to the one in Lee and Liu (2012). Assume $p = q = 2$, $\mathbf{X}^\top \mathbf{X} = \mathbf{I}$ and $\mathbf{\Sigma}_\epsilon = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$, where ρ is an unknown constant. We perform simulation studies for this example with 200 training samples, 300 tuning samples and 1000 testing samples. Set $\mathbf{B}^* = \begin{pmatrix} 0 & 0 \\ 2 & 3.5 \end{pmatrix}$ in Case 1 and $\begin{pmatrix} 0 & 0 \\ -2 & 3.5 \end{pmatrix}$ in Case 2. Figure 2.1 shows the estimation error for the separate LASSO, the two-step weighted LASSO and the joint estimation model (2.5). In Case 1, the two-step weighted LASSO has a smaller estimation error than the separate LASSO when ρ is positive. The result flips when ρ is negative. While in Case 2, the separate LASSO has a smaller estimation error than the two-step weighted LASSO when ρ is positive. The joint estimation model performs the best in all cases.

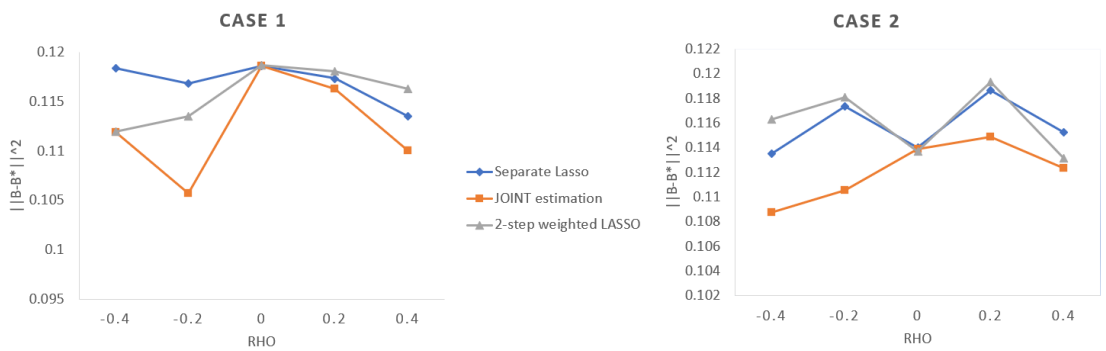

Figure 2.1: Plots of the estimation errors for separated LASSO, two-step weighted LASSO and joint estimation when $\mathbf{\Sigma}_\epsilon = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$. The left panel is for $\mathbf{B}^* = \begin{pmatrix} 0 & 0 \\ 2 & 3.5 \end{pmatrix}$ and the right panel is for $\mathbf{B}^* = \begin{pmatrix} 0 & 0 \\ -2 & 3.5 \end{pmatrix}$.

The simulation results can be explained by the following calculations. With the penalty parameter λ, the solution of the separate LASSO is given by $\hat{B}_{ij}^{\text{LASSO}} = \text{sign}(\hat{B}_{ij}^S)[\hat{B}_{ij}^S - \lambda/2]_+$, where $[u]_+ = u$ if $u \geq 0$, $[u]_+ = 0$ if $u < 0$ and $\hat{\mathbf{B}}^S = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{Y}$.

We can show that the two-step weighted LASSO (2.4) is equivalent to

$$\hat{\mathbf{B}}^{2step} = \arg\min_{\mathbf{B}} \left[(\text{vec}(\mathbf{B}) - \text{vec}(\mathbf{B}^S))^\top (\mathbf{I}_2 \otimes \hat{\mathbf{C}})(\text{vec}(\mathbf{B}) - \text{vec}(\mathbf{B}^S)) + \| \text{vec}(\mathbf{B}) \|_1 \right]. \tag{2.6}$$

When estimate $\hat{\mathbf{C}}$ is accurate, $\hat{\mathbf{B}}^{2step}$ should be very close to the solution of (2.3), where we use $\mathbf{\Sigma}_\epsilon^{-1}$ as the weight. After we plug $\hat{\mathbf{C}} = \mathbf{\Sigma}_\epsilon^{-1}$ into (2.6), the solution is given by $\hat{B}_{ij}^{2step} = \text{sign}(\hat{B}_{ij}^S)[|\hat{B}_{ij}^S| - \lambda(1+\rho)/2]_+$ when $\text{sign}(\hat{B}_{i1}^S \hat{B}_{i2}^S) = 1$ and $\hat{B}_{ij}^{2step} = \text{sign}(\hat{B}_{ij}^S)[|\hat{B}_{ij}^S| - \lambda(1-\rho)/2]_+$ when $\text{sign}(\hat{B}_{i1}^S \hat{B}_{i2}^S) = -1$. Compared with $\hat{B}_{ij}^{\text{LASSO}} = \text{sign}(\hat{B}_{ij}^S)[\hat{B}_{ij}^S - \lambda/2]_+$, $\hat{B}_{ij}^{\text{2step}}$ only differs in the shrinkage amount for each entry. The shrinkage amounts for all entries of the Separate LASSO are the same, which only depend on the tuning parameter λ. The shrinkage amounts for all entries of the two-step weighted LASSO depend on ρ, λ and the sign of $\hat{\mathbf{B}}^S$. Each entry of the two-step weighted LASSO may have different shrinkage amounts.

We consider two cases of ρ in Case 1, where $\mathbf{B}^* = \left(\begin{smallmatrix} 0 & 0 \\ 2 & 3.5 \end{smallmatrix}\right)$. Since B_{21}^*, B_{22}^* are far from 0, for simplicity, we assume that $\text{sign}(\hat{B}_{21}^S) = \text{sign}(\hat{B}_{22}^S) = 1$.

1. Consider $\rho = -0.4$. When $\text{sign}(\hat{B}_{11}^S \hat{B}_{12}^S) = -1$, the shrinkage amounts for $\hat{B}_{21}^{2step}$ and $\hat{B}_{22}^{2step}$ are 0.7λ, while the shrinkage amounts for $\hat{B}_{11}^{2step}$ and $\hat{B}_{12}^{2step}$ are 0.3λ. Thus the shrinkage amounts for $\hat{B}_{21}^{2step}$ and $\hat{B}_{22}^{2step}$ are smaller than the shrinkage amounts for $\hat{B}_{11}^{2step}$ and $\hat{B}_{12}^{2step}$.

This means that with the tuning parameter λ that shrinks $\hat{B}_{11}^{2step}$ and $\hat{B}_{12}^{2step}$ to 0, the shrinkage amounts for $\hat{B}_{21}^{2step}$ and $\hat{B}_{22}^{2step}$ are smaller than the shrinkage amounts for $\hat{B}_{21}^{LASSO}$ and $\hat{B}_{22}^{LASSO}$. Thus the two-step weighted LASSO has a smaller estimation error than separate LASSO in this scenario. When $\text{sign}(\hat{B}_{11}^{S}\hat{B}_{12}^{S}) = 1$, the shrinkage amounts for all entries in $\hat{\mathbf{B}}^{2step}$ are equal.

2. Consider $\rho = 0.4$. When $\text{sign}(\hat{B}_{11}^{S}\hat{B}_{12}^{S}) = -1$, the shrinkage amounts for $\hat{B}_{21}^{2step}$ and $\hat{B}_{22}^{2step}$ are 0.3λ, while the shrinkage amounts for $\hat{B}_{11}^{2step}$ and $\hat{B}_{12}^{2step}$ are 0.7λ. This means that with the tuning parameter λ that shrinks $\hat{B}_{11}^{2step}$ and $\hat{B}_{12}^{2step}$ to 0, the shrinkage amounts for $\hat{B}_{21}^{2step}$ and $\hat{B}_{22}^{2step}$ are larger than the shrinkage amounts for $\hat{B}_{21}^{LASSO}$ and $\hat{B}_{22}^{LASSO}$. Thus the separate LASSO is preferred to the two-step weighted LASSO in this scenario. When $\text{sign}(\hat{B}_{11}^{S}\hat{B}_{12}^{S}) = 1$, all entries in $\hat{\mathbf{B}}^{2step}$ have the same shrinkage amount.

In Case 2, where $\mathbf{B}^* = \begin{pmatrix} 0 & 0 \\ -2 & 3.5 \end{pmatrix}$, the two-step weighted LASSO is preferred to separate LASSO only when ρ is negative. In conclusion, the performance of the two-step weighted LASSO compared with the separate LASSO depends on the sign of $\mathbf{B}^*$ and the covariance matrix $\mathbf{\Sigma}_\epsilon$. In contrast, the joint estimation model (2.5) is more flexible. When $\mathbf{\Sigma}_\epsilon$ and $\mathbf{B}^*$ favor the separate LASSO, the joint estimation model (2.5) can perform better by choosing a large λ_C. Otherwise, the joint estimation model (2.5) can perform better by choosing a relatively small λ_C. Thus the joint estimation model (2.5) can perform competitively in all cases.

2.2.2.2 Covariance estimation

Next we introduce how to obtain $\hat{\mathbf{\Sigma}}_{XX}$, $\hat{\mathbf{\Sigma}}_{XY}$ and $\hat{\mathbf{\Sigma}}_{YY}$ when data have block-missing values. The following notation will be used in this article. For the jth predictor, define $S_j^X = \{i : x_{ij}$ is not missing$\}$. For the jth response, define $S_j^Y = \{i : y_{ij}$ is not missing$\}$. Define $S_{jk}^{XX} = \{i : x_{ij}$ and x_{ik} are not missing$\}$, $S_{jk}^{XY} = \{i : x_{ij}$ and y_{ik} are not missing$\}$, $S_{jkl}^{XX/Y} = \{i : x_{ij}, x_{ik}$ are not missing but y_{il} is missing$\}$, $S_{jkl}^{XY/X} = \{i : x_{ij}, y_{ik}$ are not missing but x_{il} is missing$\}$ and $S_{jk}^{YY} = \{i : y_{ij}$ and y_{ik} are not missing$\}$. Denote the cardinality of S_j^X, S_j^Y, S_{jk}^{XX}, S_{jk}^{XY}, $S_{jkl}^{XX/Y}$, $S_{jkl}^{XY/X}$ and S_{jk}^{YY} as n_j^X, n_j^Y n_{jk}^{XX}, n_{jk}^{XY}, $n_{jkl}^{XX/Y}$, $n_{jkl}^{XY/X}$, and n_{jk}^{YY}, respectively. Denote $n_X = \min_j |S_j^X|$, $n_{XX} = \min_{j,k} |S_{jk}^{XX}|$, $n_{XY} = \min_{j,k} |S_{jk}^{XY}|$, $n_{YY} = \min_{j,k} |S_{jk}^{YY}|$, $n_{XX/Y} = \min_{j,k,l} |S_{jkl}^{XX/Y}|$ and $n_{XY/X} = \min_{j,k,l} |S_{jkl}^{XY/X}|$.

We propose the initial estimators of $\mathbf{\Sigma}_{XX}$, $\mathbf{\Sigma}_{XY}$ and $\mathbf{\Sigma}_{YY}$ to be the sample covariance matrices using all available data, i.e. $\tilde{\mathbf{\Sigma}}_{XX} = (\tilde{\sigma}_{jt}^{XX}), \tilde{\mathbf{\Sigma}}_{XY} = (\tilde{\sigma}_{jt}^{XY}), \hat{\mathbf{\Sigma}}_{YY} = (\hat{\sigma}_{jt}^{YY})$, where $\tilde{\sigma}_{jt}^{XX} = \sum_{i \in S_{jt}^{XX}} x_{ij} x_{it} / n_{jt}^{XX}$, $\tilde{\sigma}_{jt}^{XY} = \sum_{i \in S_{jt}^{XY}} x_{ij} y_{it} / n_{jt}^{XY}$, and

$$\hat{\sigma}_{jt}^{YY} = \frac{1}{n_{jt}^{YY}} \sum_{i \in S_{jt}^{YY}} y_{ij} y_{it}. \tag{2.7}$$

We point out our method requires $\tilde{\mathbf{\Sigma}}_{XX}$, $\tilde{\mathbf{\Sigma}}_{XY}$ and $\hat{\mathbf{\Sigma}}_{YY}$ to be unbiased estimators of their counterparts. When the missingness in $\mathbf{X}$ and Y is missing completely at random, the unbiasedness assumption is satisfied. However, the unbiasedness assumption may also hold under some other missing mechanism. For our theories, we do not specify any particular missing mechanism. The unbiasedness assumption suffices.

For block-missing data $\mathbf{X}$, the above estimate $\tilde{\mathbf{\Sigma}}_{XX}$ can be ill-conditioned and have negative eigenvalues. Therefore, it may not be a good estimate of $\mathbf{\Sigma}_{XX}$ and cannot be used in (2.5) directly. Next, we introduce an estimator that is both well-conditioned and more accurate than the initial estimate $\tilde{\mathbf{\Sigma}}_{XX}$. According to the partition of the predictors into K modalities, $\tilde{\mathbf{\Sigma}}_{XX}$ can be partitioned into K^2 blocks, denoted by $\tilde{\mathbf{\Sigma}}^{k_1 k_2}$ for $1 \leq k_1, k_2 \leq K$ and $\tilde{\mathbf{\Sigma}}^{k_1 k_2}$ being a $p_{k_1} \times p_{k_2}$ matrix. We denote

$$\tilde{\mathbf{\Sigma}}_I = \begin{pmatrix} \tilde{\mathbf{\Sigma}}^{11} & & & \\ & \tilde{\mathbf{\Sigma}}^{22} & & \\ & & \ddots & \\ & & & \tilde{\mathbf{\Sigma}}^{KK} \end{pmatrix} \quad \text{and} \quad \tilde{\mathbf{\Sigma}}_C = \begin{pmatrix} \mathbf{0} & \tilde{\mathbf{\Sigma}}^{12} & \dots & \tilde{\mathbf{\Sigma}}^{1K} \\ \tilde{\mathbf{\Sigma}}^{21} & \mathbf{0} & \dots & \tilde{\mathbf{\Sigma}}^{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathbf{\Sigma}}^{K1} & \tilde{\mathbf{\Sigma}}^{K2} & \dots & \mathbf{0} \end{pmatrix},$$

where $\tilde{\mathbf{\Sigma}}_I$ is called the intra-modality sample covariance matrix, which is a $p \times p$ block-diagonal matrix containing K diagonal blocks of $\tilde{\mathbf{\Sigma}}_{XX}$, and $\tilde{\mathbf{\Sigma}}_C = \tilde{\mathbf{\Sigma}} - \tilde{\mathbf{\Sigma}}_I$ is called the cross-modality sample covariance matrix containing all off-diagonal blocks of $\tilde{\mathbf{\Sigma}}_{XX}$. Let $\mathbf{\Sigma}_I$ and $\mathbf{\Sigma}_C$ be the true intra-modality and cross-modality covariance matrices, respectively. For the block-missing multi-modal data, due to the imbalanced sample sizes, the estimate $\tilde{\mathbf{\Sigma}}_I$ can be relatively accurate while the estimate $\tilde{\mathbf{\Sigma}}_C$ can be inaccurate. In that case, we estimate $\mathbf{\Sigma}_{XX}$ by a linear combination of $\tilde{\mathbf{\Sigma}}_I$ and $\tilde{\mathbf{\Sigma}}_C$ with different weights. In addition, to ensure positive definiteness of our estimation, we

adopt the idea of shrinkage estimation of the covariance matrix (Fisher and Sun, 2011) and add the diagonal matrix $\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)$ to our estimator,

$$\hat{\mathbf{\Sigma}}_{XX} = \alpha_1 \tilde{\mathbf{\Sigma}}_I + (1-\alpha_1)\,\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I) + \alpha_2 \tilde{\mathbf{\Sigma}}_C, \tag{2.8}$$

where $\alpha_1, \alpha_2 \in [0,1]$ are two shrinkage weights. We add the diagonal matrix $\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)$ to ensure the diagonal entries of our estimator are not shrunk.

By Weyl's theorem, the eigenvalues of our estimator are greater than or equal to $\alpha_1 \lambda_{\min}(\tilde{\mathbf{\Sigma}}_I) + (1-\alpha_1)\lambda_{\min}(\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)) + \alpha_2 \lambda_{\min}(\tilde{\mathbf{\Sigma}}_C)$. Since $\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)$ is a positive definite matrix, by carefully selecting the tuning parameters α_1 and α_2, the eigenvalues of our estimator can be guaranteed to be positive.

As we dicussed before, our estimator $\hat{\mathbf{\Sigma}}_{XX}$ is a shrinkage estimator. Using a similar idea, we use a shrinkage estimator to estimate $\mathbf{\Sigma}_{XY}$. That is, we propose to estimate $\mathbf{\Sigma}_{XY}$ by

$$\hat{\mathbf{\Sigma}}_{XY} = \alpha_3 \tilde{\mathbf{\Sigma}}_{XY}, \tag{2.9}$$

where $\alpha_3 \in [0,1]$ is the shrinkage weight. We want to find the optimal linear combination $\hat{\mathbf{\Sigma}}^*_{XY} = \alpha_3^* \tilde{\mathbf{\Sigma}}_{XY}$ whose expected quadratic loss $\mathbb{E}\|\hat{\mathbf{\Sigma}}^*_{XY} - \mathbf{\Sigma}_{XY}\|_F$ is minimized.

In our paper, we only consider a relative low dimension of Y with not too many incomplete observations, so we will use $\hat{\mathbf{\Sigma}}_{YY}$ defined in (2.7) directly. But when the dimension of Y is very high, or there are many incomplete observations of Y, a shrinkage estimator of $\mathbf{\Sigma}_{YY}$ is recommended instead.

Denote $\gamma^* = (\gamma_1^*, \ldots, \gamma_K^*)^\top = (\mathrm{tr}(\mathbf{\Sigma}^{11})/p_1, \ldots, \mathrm{tr}(\mathbf{\Sigma}^{KK})/p_K)^\top$, $\delta_I = \sqrt{\mathbb{E}\|\tilde{\mathbf{\Sigma}}_I - \mathbf{\Sigma}_I\|_F^2}$, $\delta_C = \sqrt{\mathbb{E}\|\tilde{\mathbf{\Sigma}}_C - \mathbf{\Sigma}_C\|_F^2}$, $\delta_{XY} = \sqrt{\mathbb{E}\|\tilde{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\|_F^2}$ and $\theta = \|\,\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I) - \mathbf{\Sigma}_I\|_F$. The optimal choice for the weights of α_1, α_2, and α_3 is shown in the following proposition 2.2.1.

Proposition 2.2.1. *The solutions to the following two optimization problems:*

$$(\alpha_1^*, \alpha_2^*) = \arg\min_{\alpha_1,\alpha_2} \mathbb{E}\|\hat{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\|_F^2 \tag{2.10}$$

$$\alpha_3^* = \arg\min_{\alpha_3} \mathbb{E}\left\|\hat{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\right\|_F^2 \tag{2.11}$$

are

$$\alpha_1^* = \frac{\theta^2}{\theta^2 + \delta_I^2}, \quad \alpha_2^* = \frac{\|\mathbf{\Sigma}_C\|_F^2}{\|\mathbf{\Sigma}_C\|_F^2 + {\delta_C}^2}, \quad \alpha_3^* = \frac{\|\mathbf{\Sigma}_{XY}\|_F^2}{\|\mathbf{\Sigma}_{XY}\|_F^2 + {\delta_{XY}}^2}.$$

In addition, for $\hat{\mathbf{\Sigma}}_{XX}^* = \alpha_1^* \tilde{\mathbf{\Sigma}}_I + (1-\alpha_1^*)\,\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I) + \alpha_2^* \tilde{\mathbf{\Sigma}}_C$ *and* $\hat{\mathbf{\Sigma}}_{XY}^* = \alpha_3^* \tilde{\mathbf{\Sigma}}_{XY}$*, we have*

$$\mathbb{E}\left\|\hat{\mathbf{\Sigma}}_{XX}^* - \mathbf{\Sigma}_{XX}\right\|_F^2 = \frac{\delta_I^2\theta^2}{\delta_I^2 + \theta^2} + \frac{{\delta_C}^2\,\|\mathbf{\Sigma}_C\|_F^2}{{\delta_C}^2 + \|\mathbf{\Sigma}_C\|_F^2} \le \delta_I^2 + {\delta_C}^2 = \mathbb{E}\|\tilde{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\|_F^2,$$

$$\mathbb{E}\left\|\hat{\mathbf{\Sigma}}_{XY}^* - \mathbf{\Sigma}_{XY}\right\|_F^2 = \frac{{\delta_{XY}}^2\,\|\mathbf{\Sigma}_{XY}\|_F^2}{{\delta_{XY}}^2 + \|\mathbf{\Sigma}_{XY}\|_F^2} \le {\delta_{XY}}^2 = \mathbb{E}\|\tilde{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\|_F^2.$$

Define the ℓ_2-error of the estimators $\hat{\mathbf{\Sigma}}_{XX}$ and $\hat{\mathbf{\Sigma}}_{XY}$ as $\mathbb{E}\|\hat{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\|_F^2$ and $\mathbb{E}\|\hat{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\|_F^2$, respectively. Proposition 2.2.1 shows that our estimator is more accurate than the sample covariance matrix.

Proposition 2.2.1 is closely related to Proposition 1 in Yu et al. (2020). They calculated the optimal weight and estimation error for their proposed estimator $\hat{\mathbf{\Sigma}}_{XX,DISCOM}^*$ of $\mathbf{\Sigma}_{XX}$, whose estimation error is

$$\mathbb{E}\|\hat{\mathbf{\Sigma}}_{XX,DISCOM} - \mathbf{\Sigma}_{XX}\|_F^2 = \frac{\delta_I^2\tilde{\theta}^2}{\delta_I^2 + \tilde{\theta}^2} + \frac{{\delta_C}^2\,\|\mathbf{\Sigma}_C\|_F^2}{{\delta_C}^2 + \|\mathbf{\Sigma}_C\|_F^2},$$

where $\tilde{\theta}^2 = \|\mathrm{tr}(\mathbf{\Sigma})\mathbf{I_p}/p - \mathbf{\Sigma}_I\|_F^2$. We can see that our estimator $\hat{\mathbf{\Sigma}}_{XX}$ has smaller ℓ_2-error compared to their estimator. Comparing to their proposition, we also prove that our weighted estimator $\hat{\mathbf{\Sigma}}_{XY}$ is more accurate than the sample covariance matrix.

2.2.3 Computational algorithm

In this section, we describe the computational algorithm to solve the optimization problem (2.5). Since (2.5) is a bi-convex problem, the standard approach to solve this problem is via the alternating minimization method. In particular, starting with some given initial point $(\hat{\mathbf{B}}_0, \hat{\mathbf{C}}_0)$, at the t–th iteration, we solve solving the following problems

$$\hat{\mathbf{B}}_t = \arg\min_{\mathbf{B}} \mathrm{tr}\left[\hat{\mathbf{C}}_{t-1}\hat{\mathbf{\Sigma}}_{YY} + \hat{\mathbf{C}}_{t-1}\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{XX}\mathbf{B} - 2\hat{\mathbf{C}}_{t-1}\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{XY}\right] + \lambda_B\|\mathbf{B}\|_1, \tag{2.12}$$

$$\hat{\mathbf{C}}_t = \arg\min_{\mathbf{C}\in\mathbb{S}_+^{q\times q}} \mathrm{tr}\left[\mathbf{C}\hat{\mathbf{\Sigma}}_{YY} + \mathbf{C}\hat{\mathbf{B}}_t^\top\hat{\mathbf{\Sigma}}_{XX}\hat{\mathbf{B}}_t - 2\mathbf{C}\hat{\mathbf{B}}_t^\top\hat{\mathbf{\Sigma}}_{XY}\right] + \lambda_C\|\mathbf{C}\|_1 - \log\det\mathbf{C}. \tag{2.13}$$

In each iteration of our algorithm, given $\hat{\mathbf{C}}_{t-1}$, we first update the estimator $\hat{\mathbf{B}}_t$ by solving (2.12). Since (2.12) is quadratic in $\mathbf{B}$, we use the coordinate descent algorithm to solve it. Then we adopt the graphical lasso method by Friedman et al. (2008) to solve (2.13). We summarize the above procedures in Algorithm 3 below.

Algorithm 3: Alternating minimization updating algorithm

Input: $\mathbf{X}$, $\mathbf{Y}$, λ_C, λ_B

Output: $\hat{\mathbf{B}}, \hat{\mathbf{C}}$

Obtain $\hat{\mathbf{\Sigma}}_{XX}$ by (2.8), $\hat{\mathbf{\Sigma}}_{XY}$ by (2.9), $\hat{\mathbf{\Sigma}}_{YY}$ by (2.7).

Initialize with

$$\hat{\mathbf{B}}_0 = \arg\min_{\mathbf{B}} \operatorname{tr}\left[\hat{\mathbf{\Sigma}}_{YY} + \mathbf{B}^\top \hat{\mathbf{\Sigma}}_{XX}\mathbf{B} - 2\mathbf{B}^\top \hat{\mathbf{\Sigma}}_{XY}\right] + \lambda_{B_0}\|\mathbf{B}\|_1, \tag{2.14}$$

$$\hat{\mathbf{C}}_0 = \arg\min_{\|\mathbf{C}\|_1 \le R, \mathbf{C}\in\mathbb{S}_+^{d\times d}} \operatorname{tr}(\mathbf{C}\hat{\mathbf{\Sigma}}_0) - \log\det(\mathbf{C}) + \lambda_{C_0}\|\mathbf{C}\|_1, \tag{2.15}$$

where R is a large enough tuning parameter which is usually chosen to be $\lambda_{C_0}^{-1}$ (Loh and Wainwright, 2015) and $\hat{\mathbf{\Sigma}}_0 = \hat{\mathbf{\Sigma}}_{YY} - 2\hat{\mathbf{\Sigma}}_{XY}^\top \hat{\mathbf{B}}_0 + \hat{\mathbf{B}}_0^\top \hat{\mathbf{\Sigma}}_{XX}\hat{\mathbf{B}}_0$.

while $\max\left\{\|\hat{\mathbf{B}}_t - \hat{\mathbf{B}}_{t-1}\|_F, \|\hat{\mathbf{C}}_t - \hat{\mathbf{C}}_{t-1}\|_F\right\} > \textit{threshold}$ **do**

For a given $\hat{\mathbf{C}}_{t-1}$, let

$$\hat{\mathbf{B}}_t = \arg\min_{\mathbf{B}} \operatorname{tr}\left[\hat{\mathbf{C}}_{t-1}\hat{\mathbf{\Sigma}}_{YY} + \hat{\mathbf{C}}_{t-1}\mathbf{B}^\top \hat{\mathbf{\Sigma}}_{XX}\mathbf{B} - 2\hat{\mathbf{C}}_{t-1}\mathbf{B}^\top \hat{\mathbf{\Sigma}}_{XY}\right] + \lambda_B\|\mathbf{B}\|_1;$$

For a given $\hat{\mathbf{B}}_t$, let

$$\hat{\mathbf{C}}_t = \arg\min_{\|\mathbf{C}\|_1 \le R, \mathbf{C}\in\mathbb{S}_+^{q\times q}} \operatorname{tr}\left[\mathbf{C}\hat{\mathbf{\Sigma}}_{YY} + \mathbf{C}\hat{\mathbf{B}}_t^\top \hat{\mathbf{\Sigma}}_{XX}\hat{\mathbf{B}}_t - 2\mathbf{C}\hat{\mathbf{B}}_t^\top \hat{\mathbf{\Sigma}}_{XY}\right] + \lambda_C\|\mathbf{C}\|_1 - \log\det\mathbf{C},$$

end

return $\hat{\mathbf{C}}_t$, $\hat{\mathbf{B}}_t$.

2.3 Theoretical study

We establish the following theoretical results. First, we prove in Theorem 2.3.1 that the proposed estimators $\hat{\mathbf{\Sigma}}_{XX}$, $\hat{\mathbf{\Sigma}}_{XY}$ and $\hat{\mathbf{\Sigma}}_{YY}$ are consistent with high probability. We then show

the convergence rate of our proposed estimators $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$ in Theorem 2.3.2. Finally, the selection consistency of our proposed method is shown in Theorem 2.3.3. The technical assumptions (A1) to (A5), and all proofs are provided in the Supplementary Material. In the following analysis, we allow p and q to diverge as n_{XX}, n_{XY} and n_{YY} increase.

In Theorem 2.3.1, we prove the large deviation bounds for our proposed estimators $\hat{\mathbf{\Sigma}}_{XX}$, $\hat{\mathbf{\Sigma}}_{XY}$ and $\hat{\mathbf{\Sigma}}_{YY}$.

Theorem 2.3.1. *Suppose* $1-\alpha_1 = O(\sqrt{\log p/n_X})$, $1-\alpha_2 = O(\sqrt{\log p/n_{XX}})$, *and* $1-\alpha_3 = O(\sqrt{\log pq/n_{XY}})$. *If Conditions (A1) and (A2) hold, there exists positive constants* v_1', v_2', *and* v_3' *such that*

$$P\left(\left\|\hat{\mathbf{\Sigma}}_{XX}-\mathbf{\Sigma}_{XX}\right\|_\infty \geq v_1'\sqrt{\frac{\log p}{n_{XX}}}\right) \leq \frac{4}{p}, \tag{2.16}$$

$$P\left(\left\|\hat{\mathbf{\Sigma}}_{XY}-\mathbf{\Sigma}_{XY}\right\|_\infty \geq v_2'\sqrt{\frac{\log(pq)}{n_{XY}}}\right) \leq \frac{4}{pq}, \tag{2.17}$$

$$P\left(\left\|\hat{\mathbf{\Sigma}}_{YY}-\mathbf{\Sigma}_{YY}\right\|_\infty \geq v_3'\sqrt{\frac{\log q}{n_{YY}}}\right) \leq \frac{4}{q}. \tag{2.18}$$

If we only use samples with complete observations, sample covariance estimators $\tilde{\mathbf{\Sigma}}_{XX,\text{complete}}$, $\tilde{\mathbf{\Sigma}}_{XX,\text{complete}}$ and $\tilde{\mathbf{\Sigma}}_{XX,\text{complete}}$ have the following convergence rates

$$\begin{aligned}
\left\|\tilde{\mathbf{\Sigma}}_{XX,\text{complete}}-\mathbf{\Sigma}_{XX}\right\|_\infty &= O_p\left(\sqrt{(\log p)/n_{\text{complete}}}\right),\\
\left\|\tilde{\mathbf{\Sigma}}_{XY,\text{complete}}-\mathbf{\Sigma}_{XY}\right\|_\infty &= O_p\left(\sqrt{(\log(pq))/n_{\text{complete}}}\right),\\
\left\|\tilde{\mathbf{\Sigma}}_{YY,\text{complete}}-\mathbf{\Sigma}_{YY}\right\|_\infty &= O_p\left(\sqrt{(\log q)/n_{\text{complete}}}\right),
\end{aligned}$$

where n_{complete} is the number of samples with complete observations; see Yu et al. (2020). For block-missing data, n_{complete} can be much smaller than n_{XX}, n_{XY} and n_{YY}.

Next, we give the properties of initial estimators $\hat{\mathbf{B}}_0$ and $\hat{\mathbf{C}}_0$. The following lemma describes estimation consistency of the initial estimator $\hat{\mathbf{B}}_0$.

Lemma 2.3.1. *Suppose Conditions (A1)–(A4) hold,* $1-\alpha_1 = O(\sqrt{\log p/n_X})$, $1-\alpha_2 = O(\sqrt{\log p/n_{XX}})$, *and* $1-\alpha_3 = O(\sqrt{\log pq/n_{XY}})$. *If we choose* $\lambda_{B_0} =$

$C(\log(pq)/\min(n_{XY}, n_{XX}))^{\frac{1}{2}}\|\mathbf{B}^*\|_{L_1}$ *for some large enough constant* C*, then with probability at least* $1 - 4/p - 4/(pq)$*, the initial estimator* $\hat{\mathbf{B}}_0 = \arg\min_{\mathbf{B}} \operatorname{tr}[\hat{\mathbf{\Sigma}}_{YY} + \mathbf{B}^\top \hat{\mathbf{\Sigma}}_{XX}\mathbf{B} - 2\mathbf{B}^\top \hat{\mathbf{\Sigma}}_{XY}] + \lambda_B\|\mathbf{B}\|_1$ *satisfies*

$$\begin{aligned}\left\|\hat{\mathbf{B}}_0 - \mathbf{B}^*\right\|_F &\lesssim \sqrt{qs_B}\left\|\hat{\mathbf{\Sigma}}_{XY} - \hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^*\right\|_\infty \\ &\lesssim \|\mathbf{B}^*\|_{L_1}\sqrt{\frac{qs_B\log(pq)}{\min(n_{XX}, n_{XY})}}.\end{aligned}$$

Cai et al. (2013) showed that when there is no missing data and the true coefficient $\mathbf{B}^*$ is exactly sparse, their estimator $\hat{\mathbf{B}}_{Cai}$ has the convergence rate of $\|\hat{\mathbf{B}}_{Cai} - \mathbf{B}^*\|_F = O_p(N_p\sqrt{qs_B\log(pq)/n})$, where n is the sample size of the data and N_p is the upper bound of $\|\mathbf{\Sigma}_{XX}^{-1}\|_{L_\infty}$. When there is no missing data, our initial estimator $\hat{\mathbf{B}}_0$ has the convergence rate of $\|\hat{\mathbf{B}}_0 - \mathbf{B}^*\|_F = O_p(\|\mathbf{B}^*\|_{L_1}\sqrt{qs_B\log(pq)/n})$. If we assume $\|\mathbf{B}^*\|_{L_1} \asymp \|\mathbf{\Sigma}_{XX}^{-1}\|_{L_\infty}$, the convergence rate of $\hat{\mathbf{B}}_0$ is the same as that of $\hat{\mathbf{B}}_{Cai}$. When the data are block-wise missing, and we only use complete samples to estimate $\mathbf{B}^*$, we will have $\|\hat{\mathbf{B}}_0 - \mathbf{B}^*\|_F = O_p(\|\mathbf{B}^*\|_{L_1}\sqrt{qs_B\log(pq)/n_{\text{complete}}})$, which can be much slower than the rate in Lemma 2.3.1 as n_{complete} is typically much smaller than n_{XX} and n_{XY} for block-wise missing data.

For the single-response regression with block-wise missing data, the result in Lemma 2.3.1 is the same as Theorem 2 in Yu et al. (2020) and the estimator $\hat{\mathbf{B}}_0$ performs well when the dimension of $\mathbf{Y}$ is small. But when the dimension of $\mathbf{Y}$ becomes large, the estimator $\hat{\mathbf{B}}_0$ may perform poorly.

The following lemma describes consistency of our initial estimator $\hat{\mathbf{C}}_0$.

Lemma 2.3.2. *Suppose Conditions (A1)–(A4) hold,* $1 - \alpha_1 = O(\sqrt{\log p/n_X})$*,* $1 - \alpha_2 = O(\sqrt{\log p/n_{XX}})$*,* $1 - \alpha_3 = O(\sqrt{\log pq/n_{XY}})$*. If we choose* $\lambda_{C_0} = C\|\mathbf{C}^*\|_2^2\|\mathbf{B}^*\|_{L_1}\left(\|\mathbf{B}^*\|_{L_1} + s_B\sqrt{q}\right)(\log(pq)/\min(n_{XX}, n_{XY}))^{1/2}$ *for a large enough* C*, it holds with probability at least* $1 - 4/p - 4/(pq) - 4/q$ *that*

$$\begin{aligned}\left\|\hat{\mathbf{C}}_0 - \mathbf{C}^*\right\|_F &\lesssim \sqrt{s_C}\|\mathbf{C}^*\|_2^2\|\mathbf{\Sigma}_\epsilon - \hat{\mathbf{C}}_0^{-1}\|_\infty \\ &\lesssim \|\mathbf{C}^*\|_2^2\|\mathbf{B}^*\|_{L_1}\left(\|\mathbf{B}^*\|_{L_1} + s_B\sqrt{q}\right)\sqrt{\frac{s_C\log(pq)}{\min(n_{XX}, n_{XY})}}.\end{aligned}$$

There are two terms in the estimation error bound of $\hat{\mathbf{C}}_0$. The first term $\|\mathbf{C}^*\|_2^2\|\mathbf{B}^*\|_{L_1}^2\sqrt{\frac{s_C\log(pq)}{\min(n_{XX}, n_{XY})}}$ comes from the error induced by using incomplete observations to

estimate $\mathbf{\Sigma}_{XX}$ and $\mathbf{\Sigma}_{XY}$. The second term $\|\mathbf{C}^*\|_2^2\|\mathbf{B}^*\|_{L_1} s_B\sqrt{\frac{s_C q\log(pq)}{\min(n_{XX},n_{XY})}}$ comes from the estimation error of $\hat{\mathbf{B}}_0$.

We next derive the convergence rates of $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$. The convergence rates are related to $n_{XX/Y}$ and $n_{XY/X}$, which are fractions of n_{XX} and n_{XY} respectively. Hence, we let $n_{XX/Y} \asymp n_{XX}^{\tau_1}$ and $n_{XY/X} \asymp n_{XY}^{\tau_2}$ with $\tau_1, \tau_2 \in \{-\infty\} \cup [0,1]$. When the responses are complete while the covariates have missing entries, $n_{XX/Y} = 0$ and $\tau_1 = -\infty$, $n_{XY/X} > 0$ and $\tau_2 \in [0,1]$. When the covariates are complete while the responses have missing entries, $n_{XY/X} = 0$ and $\tau_2 = -\infty$, $n_{XX/Y} > 0$ and $\tau_1 \in [0,1]$. When both the responses and covaraites are complete, $n_{XX/Y} = n_{XY/X} = 0$ and $\tau_1 = \tau_2 = -\infty$. Theorem 2.3.2 below establishes the consistency of proposed estimators $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$ in (2.5).

Theorem 2.3.2. *Suppose Conditions (A1)–(A4) hold,* $1-\alpha_1 = O(\sqrt{\log p/n_X})$, $1-\alpha_2 = O(\sqrt{\log p/n_{XX}})$, $1-\alpha_3 = O(\sqrt{\log(pq)/n_{XY}})$. *If we choose* λ_B *and* λ_C *satisfying* $\lambda_B = C((\log p)^{1/2}/\min(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2})\|\mathbf{B}^*\mathbf{C}^*\|_{L_1} + \{\log(pq)/n_{XY}\}^{1/2})$ *and* $\lambda_C = C\|\mathbf{C}^*\|_2^2[\|\mathbf{B}^*\|_{L_1}^2 + s_B\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}/\min(n_{XX}^{1/2-\tau_1/2}, n_{XY}^{1/2-\tau_2/2})]$
$(\log(pq)/\min(n_{XX}, n_{XY}))^{1/2}$ *for a large enough C, then it holds with probability at least* $1-4/p-4/(pq)-4/q$ *that*

$$\left\|\hat{\mathbf{B}}-\mathbf{B}^*\right\|_F \lesssim \sqrt{s_B}\left(\frac{\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}(\log(pq))^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)} + \left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}\right),$$

$$\left\|\hat{\mathbf{C}}-\mathbf{C}^*\right\|_F \lesssim \sqrt{s_C}\|\mathbf{C}^*\|_2^2\left(\frac{s_B\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}(\log(pq))^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)} + \frac{\|\mathbf{B}^*\|_{L_1}^2(\log(pq))^{1/2}}{\min\left(n_{XX}^{1/2}, n_{XY}^{1/2}\right)}\right)$$

$$\left\|\hat{\mathbf{B}}-\mathbf{B}^*\right\|_1 \lesssim s_B\left(\frac{\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}(\log(pq))^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)} + \left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}\right),$$

$$\left\|\hat{\mathbf{C}}-\mathbf{C}^*\right\|_1 \lesssim s_C\|\mathbf{C}^*\|_2^2\left(\frac{s_B\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}(\log(pq))^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)} + \frac{\|\mathbf{B}^*\|_{L_1}^2(\log(pq))^{1/2}}{\min\left(n_{XX}^{1/2}, n_{XY}^{1/2}\right)}\right).$$

Next, we discuss some direct implications of Theorem 2.3.2. First, we show that our estimators are at least as good as the initial estimators under some conditions. Since $\tau_1, \tau_2 \le 1$ as $n_{jkl}^{XX/Y} \le n_{jk}^{XX}$ and $n_{jkl}^{XY/X} \le n_{jk}^{XY}$, the convergence rate of $\|\hat{\mathbf{B}}-\mathbf{B}^*\|_F$ is no slower than $O_p(\max(\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}, 1)$ $\sqrt{s_B\log(pq)/\min(n_{XX}, n_{XY})})$. Similarly, the convergence rate of $\|\hat{\mathbf{C}}-\mathbf{C}^*\|_F$ is no slower than $O_p(\sqrt{s_C}\|\mathbf{C}^*\|_2^2(\|\mathbf{B}^*\|_{L_1}^2 + s_B\|\mathbf{B}^*\mathbf{C}^*\|_{L_1})\sqrt{\frac{\log(pq)}{\min(n_{XX},n_{XY})}})$. Here the two slowest convergence rates

are achieved when $\tau_1 = \tau_2 = 1$. If we assume $\|\mathbf{B}^*\mathbf{C}^*\|_{L_1} = O(\|\mathbf{B}^*\|_{L_1}\sqrt{q})$, the upper bounds of $\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F$ and $\|\hat{\mathbf{C}} - \mathbf{C}^*\|_F$ are at least as tight as $\|\hat{\mathbf{B}}_0 - \mathbf{B}^*\|_F$ and $\|\hat{\mathbf{C}}_0 - \mathbf{C}^*\|_F$.

On the other hand, if $\|\mathbf{B}^*\mathbf{C}^*\|_{L_1} = o(\|\mathbf{B}^*\|_{L_1}\sqrt{q})$ or $\max(\tau_1, \tau_2) < 1$ and $\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}^2 = o(\min(n_{XX}^{1/2-\tau_1/2}, n_{XY}^{1/2-\tau_2/2}))$, the upper bounds of $\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F$ and $\|\hat{\mathbf{C}} - \mathbf{C}^*\|_F$ are strictly tighter than that of $\|\hat{\mathbf{B}}_0 - \mathbf{B}^*\|_F$ and $\|\hat{\mathbf{C}}_0 - \mathbf{C}^*\|_F$. One example is when $\mathrm{Var}(\epsilon_j) > \frac{1}{\sqrt{q}}$ for all $j \leq q$ and $cov(\epsilon_j, \epsilon_k) = 0$ for $j \neq k$. Another example is when $n_{XX/Y} = o(n_{XX})$, $n_{XY/X} = o(n_{XY})$, and $\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}^2 = o(\min(n_{XX}^{1/2-\tau_1/2}, n_{XY}^{1/2-\tau_2/2}))$.

When $\mathbf{Y}$ is complete while $\mathbf{X}$ has missing entries, $\tau_1 = -\infty$ and $\tau_2 \in [0, 1]$. Then convergence rate of $\hat{\mathbf{B}}$ in Theorem 2.3.2 becomes

$$\left\|\hat{\mathbf{B}} - \mathbf{B}^*\right\|_F \lesssim \sqrt{s_B}\left(\frac{\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}(\log(pq))^{1/2}}{n_{XY}^{1-\tau_2/2}} + \left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}\right).$$

When $\mathbf{X}$ are complete while $\mathbf{Y}$ have missing entries, $\tau_2 = -\infty$ and $\tau_1 \in [0, 1]$. In this case, we can set $\alpha_1 = \alpha_2 = 1$ and have

$$\left\|\hat{\mathbf{B}} - \mathbf{B}^*\right\|_F \lesssim \sqrt{s_B}\left(\frac{\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}(\log(pq))^{1/2}}{n_{XX}^{1-\tau_1/2}} + \left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}\right).$$

When both $\mathbf{X}$ and $\mathbf{Y}$ are complete, $\tau_1 = \tau_2 = -\infty$. In this case, we can set $\alpha_1 = \alpha_2 = \alpha_3 = 1$ and have

$$\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F \lesssim \sqrt{s_B \log(pq)/n}, \tag{2.19}$$

where n is the sample size. The error bound in (2.19) is the minimax rate of the ℓ_1-penalized estimator as shown in Raskutti et al. (2011).

In Theorem 2.3.3 below, we show that our proposed method is model selection consistent.

Theorem 2.3.3. *Assume that Conditions (A1)–(A5) hold. Suppose* $1 - \alpha_1 = O(\sqrt{\log p/n_X})$, $1 - \alpha_2 = O(\sqrt{\log p/n_{XX}})$, $1 - \alpha_3 = O(\sqrt{\log(pq)/n_{XY}})$. *If* $(\log(pq)/n_{XY})^{\frac{1}{2}-\gamma_2}/\lambda_B = o(1)$, $\lambda_B\|((\mathbf{C}^* \otimes \mathbf{\Sigma}_{XX})_{S_BS_B})^{-1}\|_{L_\infty}/\min_{j\in S_B}|\boldsymbol{\beta}_j^*| = o(1)$, $s_B\|((\mathbf{C}^*\otimes\mathbf{\Sigma}_{XX})_{S_BS_B})^{-1}\|_{L_\infty}(\log p/n_{XX})^{\frac{1}{2}-\gamma_2} = o(1)$, *and* s_B
$(\log p/n_{XX})^{\frac{1}{2}-\gamma_1-\gamma_2}/\lambda_B = o(1)$, *then with probability at least* $1 - 4/p - 4/(pq) - 4/q$, *there exists a solution* $\hat{\mathbf{B}}$ *to (2.5) such that* $\mathrm{sign}(\hat{\mathbf{B}}) = \mathrm{sign}(\mathbf{B}^*)$.

2.4 Numerical study

In this section, we examine the performance of our proposed method (Multi-DISCOM) related to $\mathbf{\Sigma}_\epsilon$, the signal-to-noise ratio and the distribution of error $\boldsymbol{\epsilon}$ through some numerical studies. We compare the efficiency of our proposed method with some other methods. These methods include (1) Complete Lasso, which separately applies Lasso to each response only using samples with complete observations (both X and Y have no missing values); (2) Imputed-Lasso, which separately applies Lasso to each response using all samples, where missing data are imputed by the Soft-thresholded SVD method; (3) MBI, which separately applies the MBI (Xue and Qu, 2021) to each response using all samples, where missing data are imputed by the Multiple Block-wise Imputation; (4) DISCOM, which separately applies the DISCOM (Yu et al., 2020) to each response; (5) Imputed-MRCE, which runs the MRCE (Rothman et al., 2010) using all samples with missing data imputed by the Soft-thresholded SVD method.

In all examples, we set $q = 4$, $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^\top \sim N(\mathbf{0}, \mathbf{\Sigma})$ with $\sigma_{jt} = 0.6^{|j-t|}$. The ith row of the coefficient matrix $\mathbf{B}^*$ is $(1, 1.5, 1, 1.5)$ for $i = 1, p_1 + 1, p_1 + p_2 + 1$ and 0 otherwise. The response $\mathbf{Y}$ has missing entries completely at random, with the missing proportion 0.01.

For each example, the data were generated from three modalities whose dimensions p_1, p_2 and p_3 are specified below. The training dataset contains n_1 samples with complete observations, n_2 samples from the third modality, n_3 samples from the first and the third modalities and n_4 samples from the first modality. The tuning dataset contains 75 samples with complete observations and the testing dataset includes 300 samples with complete observations. For each method, we train our model with different tuning parameters on the training dataset. Then we choose the optimal tuning parameter minimizing the mean squared error on the tuning dataset.

For each example, we repeat the simulation 50 times. To evaluate the selection performance of the algorithm, we use false-positive rate (FPR) and false-negative rate (FNR) as criteria: $\text{FPR} = \text{FP}/(\text{FP} + \text{TN})$ and $\text{FNR} = \text{FN}/(\text{FN} + \text{TP})$, where FN represents the number of coefficients wrongly detected to be zero, TN are the number coefficients rightfully detected to be zero, TP are the coefficients rightfully detected to be nonzero and FP are the coefficients wrongly detected to be nonzero. Furthermore, to evaluate the accuracy of our estimators, we used the mean squared error (MSE) on the testing dataset and the ℓ_2 distance $\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F$ as criteria.

In Example 1, we examine our method related to $\mathbf{\Sigma}_\epsilon$. Let $n_1 = n_2 = n_3 = n_4 = 30$, $p_1 = p_2 = p_3 = 30$. We set error $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \ldots, \epsilon_{iq}) \sim N(\mathbf{0}, \mathbf{\Sigma}_\epsilon)$ with $\mathbf{\Sigma}_\epsilon = 3\mathbf{I}_2 \otimes \left(\begin{smallmatrix} 1 & \rho \\ \rho & 1 \end{smallmatrix}\right)$. We choose ρ ranging from -0.4 to 0.4.

In Example 2, we examine the performance of our method related to the signal-to-noise ratio. Let $n_1 = n_2 = n_3 = n_4 = 30$, $p_1 = p_2 = p_3 = 30$. We set error $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \ldots, \epsilon_{iq}) \sim N(\mathbf{0}, \mathbf{\Sigma}_\epsilon)$ with $\mathbf{\Sigma}_\epsilon = \alpha\mathbf{I}_2 \otimes \left(\begin{smallmatrix} 1 & -0.4 \\ -0.4 & 1 \end{smallmatrix}\right)$, and range α from 1 to 5.

In Example 3, we examine the robustness of our method when the error follows heavy-tailed distribution. Let $n_1 = n_2 = n_3 = n_4 = 30$ and $p_1 = p_2 = p_3 = 30$. We set error $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \ldots, \epsilon_{iq}) \sim t_{10}(\mathbf{0}, \mathbf{\Sigma}_\epsilon)$ where $\mathbf{\Sigma}_\epsilon = 3\mathbf{I}_2 \otimes \left(\begin{smallmatrix} 1 & -0.4 \\ -0.4 & 1 \end{smallmatrix}\right)$, and $t_\nu(\mathbf{0}, \mathbf{\Sigma}_\epsilon)$ refers to student's t distribution with location vector $\mathbf{0}$ and scale matrix $\mathbf{\Sigma}_\epsilon$.

To demonstrate the results, we focus on the results of Example 1. We report the results of other examples in Appendix A.

The results in Table 2.1 indicate that the Multi-DISCOM delivers the best performance in all settings. Specifically, the Multi-DISCOM produces smaller MSE and estimation errors than the other methods in all settings, especially when the correlations between different responses are large. In addition, the Lasso method using the imputed data may deliver worse selection performance, possibly due to randomness involved in the imputation of block-missing data. The results in Table 4 in the Supplement Materials indicate that the Multi-DISCOM has more advantage when signal-to-noise ratio is small. When the signal-to-noise ratio is smaller, the noise has stronger effect on $\mathbf{Y}$ and hence taking the precision matrix into account is more helpful for our estimation.

2.5 Application to the ADNI study

We apply the Multi-DISCOM to the Alzheimer's Disease Neuroimaging Initiative (ADNI) study (Mueller et al., 2005) and compare it with several existing approaches. A primary goal of this analysis is to identify biological markers and neuropsychological assessments to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD). We are interested in predicting Mini-Mental State Examination (MMSE), ADAS1 and ADAS2. These scores are commonly used diagnotic scores of AD. Data processing steps are summarized in the supplementary materials.

	Method	$\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F$	MSE	FPR	FNR
$\rho = -0.4$	Lasso	1.51(0.06)	3.70(0.06)	0.09(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.73(0.06)	3.57(0.06)	0.11(0.01)	**0.00(0.00)**
	MBI	2.10(0.08)	4.26(0.09)	0.12(0.02)	0.11(0.03)
	DISCOM	1.44(0.04)	3.56(0.06)	0.05(0.00)	0.05(0.01)
	Imputed-MRCE	1.53(0.05)	3.72(0.08)	0.17(0.03)	0.08(0.02)
	Multi-DISCOM	**1.40(0.04)**	**3.39(0.08)**	**0.02(0.01)**	0.09(0.02)
$\rho = 0.4$	Lasso	1.55(0.06)	3.77(0.06)	0.11(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.75(0.06)	3.61(0.06)	0.13(0.01)	**0.00(0.00)**
	MBI	2.14(0.08)	4.30(0.09)	0.13(0.02)	0.11(0.03)
	DISCOM	**1.46(0.04)**	3.59(0.06)	0.06(0.00)	0.05(0.01)
	Imputed-MRCE	1.54(0.05)	3.73(0.08)	0.19(0.03)	0.09(0.02)
	Multi-DISCOM	**1.43(0.04)**	**3.44(0.08)**	**0.04(0.01)**	0.07(0.02)

Table 2.1: Performance comparison of different methods for Example 1 with different ρ's. The values in the parentheses are the standard errors of the measures.

After data processing, we have 93 features from MRI, 93 features from PET and 5 features from CSF. There are 805 subjects in total, including 199 subjects with complete MRI, PET and CSF features, 197 subjects with MRI and PET features only, 201 subjects with MRI and CSF features only and 208 subjects with MRI features only.

In our analysis, we divide the data into training, tuning, and testing sets. The training set consists of all subjects with incomplete observations and 40 randomly selected subjects with complete features. The tuning set consists of another 40 randomly selected subjects with complete observations. The testing set contains the remaining 119 subjects with complete observations. We train our model with different tuning parameters on the training set. Then we choose the tuning parameter which minimizes the mean squared error on the tuning set. The testing set is used to evaluate different methods. We used all methods shown in the simulation study to predict the MMSE score. For each method, the analysis was repeated 30 times using different partitions of the data. In addition to the sum of mean squared errors (MSE) of all three responses, we compare MSEs for each response (MSE_{MMSE}, MSE_{ADAS1} and MSE_{ADAS2}) as criteria. We also compare the number of features selected by each method.

Method	Overall MSE	MSE_{MMSE}	MSE_{ADAS1}	MSE_{ADAS2}	# of Selected Features
Lasso	93.37(3.82)	5.31(0.19)	29.84(1.35)	58.23(2.40)	54.20
Imputed-Lasso	80.40(1.62)	4.54(0.12)	25.80(0.51)	50.07(1.15)	165.00
MBI	91.84(3.02)	5.13(0.14)	28.43(1.17)	58.29(2.16)	59.87
DISCOM	67.47(1.33)	**4.26(0.11)**	21.76(0.51)	41.45(0.86)	72.87
Imputed-MRCE	67.41(2.02)	**4.29(0.10)**	**21.61(0.65)**	41.50(1.33)	218.50
Multi-DISCOM	**65.82(1.21)**	**4.22(0.12)**	**21.18(0.46)**	**40.41(0.80)**	89.67

Table 2.2: Performance comparison for the ADNI data.

As shown in Table 2.2, the Multi-DISCOM delivers better performance than all other methods. The DISCOM has a similar overall MSE as the Multi-DISCOM, but worse MSE_{ADAS1} and MSE_{ADAS2}. One possible reason is that ADAS1 and ADAS2 are highly correlated, so taking the precision matrix into account can help. Since there are 208 subjects with MRI features only, the MBI method may not impute those 208 subjects accurately. As a consequence, the MBI method may not perform well in this case.

Regarding to model selection, both the DISCOM and the Multi-DISCOM can deliver relatively simple models. Figure 2.2 shows the selection frequency of the 191 features when predicting ADAS1. The selection frequency of each feature is defined as the number of times of being selected in the 30 replications. As shown in Figure 2.2, for our method, some features are often selected and many other features are rarely selected. This means that our method could deliver robust model selection. However, for the Imputed-Lasso method, it selects very different features in different replications. One possible reason for the unstable performance on model selection is due to the randomness involved in the imputation of block-missing data. Hippocampus formation left (69th region) and amygdale right (83th feature) are frequently selected by our method and known to be highly correlated with AD and MCI by many existing studies (Jack et al., 1999; Misra et al., 2009; Zhang et al., 2012b), but the DISCOM rarely selects these features.

2.6 Conclusion

In this paper, we propose a joint estimation method in a penalized framework with the entry-wise ℓ_1 regularization using block-missing multi-modal predictors. We first estimate the covariance

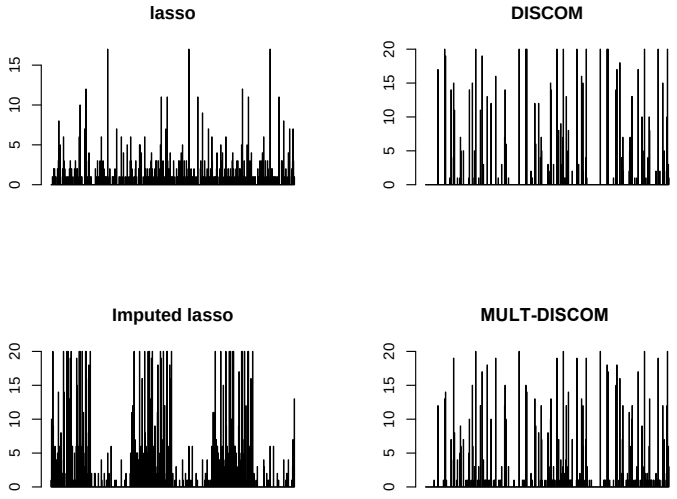

Figure 2.2: Selection frequency of 191 features for prediction of ADAS1 score.

matrix of the predictors using a linear combination of the estimates of the variance of each predictor, the estimates of the intra-modality covariance matrix, and the cross-modality covariance matrix. The proposed estimator of the covariance matrix can be positive semidefinite and more accurate than the sample covariance matrix. In the second step, based on the estimated covariance matrix, a penalized estimator is used to deliver a sparse estimate of the coefficients in the optimal linear prediction. Theoretical studies on the estimation and feature selection consistency are established. Extensive simulation studies also indicate that our method has promising performance on estimation, prediction and model selection for the block-missing multi-modal data. Finally, we apply the Multi-DISCOM to the ADNI dataset and demonstrate that our model has good prediction power and meaningful interpretation.

CHAPTER 3

Regularized Buckley–James method for Right-censored Outcome and Block-missing covariates

3.1 Introduction

Measures of neural activity such as magnetic resonance imaging (MRI) and positron emission tomography (PET) yield thousands of predictor variables for diagnosis and prognosis in patients with diseases such as the Alzheimer's disease (AD). Since not all variables contain helpful information for the model, selecting a parsimonious subset of variables with good prediction accuracy can be very important. While linear regression with a scalar response and complete data has been well studied (Tibshirani, 1996), data with censored outcomes and incomplete covariates present new challenges.

AD is a progressive neurodegenerative disease characterized by overall cognitive decline as well as behavioral and functional changes that eventually impair an individual's ability to perform the basic daily activities. People diagnosed with mild cognitive impairment (MCI), which is generally considered as a transitional stage between healthy cognitive aging and dementia, are at significantly increased risk of clinical AD (Knopman et al., 2003; Gauthier et al., 2006). Thus, MCI is a critical prognostic and therapeutic component in AD study, and it is helpful to develop reliable methods to analyze the conversion time from MCI to AD. Although up to 60% of MCI patients convert to AD within ten years, many return to the normal cognitive function (Manly et al., 2008; Mitchell and Shiri-Feshki, 2009). The AD conversion time of those participants who did not progress to AD during their follow-up period was censored at their last visit time.

Increasing efforts have focused on building predictive models of the AD conversion based on the proportional hazard (PH) model or accelerated failure time (AFT) model. For example, to examine the usage of MRI and CerebroSpinal Fluid (CSF) biomarkers to predict the conversion from MCI to AD, (Vemuri et al., 2009) used a single-predictor Cox PH model to predict the hazard

ratio of the conversion from MCI to AD. They showed that MRI and CSF provide complimentary predictive information about the conversion from MCI to AD. They also showed that combining MRI and CSF can predict better than using either source alone. Liu et al. (2017) used independent component analysis (ICA) and the multivariate Cox PH regression model to identify promising risk factors associated with MCI conversion.

In the literature, many papers also used the AFT model (Kalbfleisch and Prentice, 2011; Cox and Oakes, 2018) to analyze the conversion time of AD, where the response refers to the logarithm of a failure time. The AFT model is based on the linear model and the estimated regression coefficients can help provide useful interpretation (Reid, 1994). It is well-known that the linear model and the PH model cannot hold simultaneously except in the case of the extreme value error distribution. Two general estimation strategies to handle censored responses in the AFT model include extensions of least-squares estimators through missing data techniques (Buckley and James, 1979; Koul et al., 1981; Miller and Halpern, 1982; Lai and Ying, 1991) and rank-based methods (Prentice, 1978; Tsiatis, 1990; Lai and Ying, 1991). For example, (Oulhaj et al., 2009) used the smoothing AFT procedure with G-splines to predict the period of time before cognitive impairment occurs in community-dwelling elderly. (Ning et al., 2011) proposed a generalized Buckley-James type of estimator using right-censored and length-biased data under semiparametric transformation and AFT models. Their proposed method was applied to assess the effect of different diagnostic categories of AD using survival data.

Several authors have also extended the PH and AFT models for variable selection and explored their properties. (Tibshirani, 1997; Gui and Li, 2005) developed regularized Cox regression methods by adding an ℓ_1 penalization term to the partial likelihood function of the Cox model. Similarly, (Datta et al., 2007; Johnson, 2009) addded an ℓ_1 penalization term to the Buckley–James estimators for the AFT model. Wang et al. (2008) added the elastic-net penalty in the Buckley–James method for the AFT model to relate high-dimensional genomic data to censored survival outcomes. (Johnson, 2009) proved that, under suitable regularity conditions, an ℓ_1-penalized Buckley-James estimator with only one iteration yields a root-n consistent solution. Wang and Wang (2010) proposed the Buckley-James boosting method for the semiparametric AFT models with right-censored survival data, which can be used for prediction and variable selection.

In the past few years, there has been extensive research on using neuroimaging data for MCI and AD prediction (Eskildsen et al., 2013; Park and Moon, 2016). However, data in Alzheimer's Disease Neuroimaging Initiative (ADNI) study were collected from different sources, which include MRI, PET, and CSF. Data from a specific modality can be entirely missing due to patient dropouts or other practical issues. This leads to a block-wise missing data structure. Due to block-wise missing structure with high dimensionality and censored response, it is challenging to identify the patients likely to convert from MCI to AD. It is also interesting to further predict the conversion time for an effective risk estimate, which could lead to an efficient intervention of pharmacological treatments for early AD (Jack Jr, 2012).

Most of the AFT and PH models can only work with complete covariates. To handle incomplete multi-modal data in the ADNI study, one may use traditional AFT or PH models by simply removing those observations with missing entries. However, such a procedure may greatly reduce the number of observations and lead to loss of information. Another approach is to perform data imputation, where missing data are replaced by data generated from an imputation model. Imputation methods have been used in both AFT models (Qi et al., 2018) and PH models (Paik and Tsai, 1997; White and Royston, 2009; Hsu and Yu, 2019) to deal with incomplete covariates. Another approach is to use weighted estimating equations for AFT models (Nan et al., 2009; Steingrimsson and Strawderman, 2017) and PH models (Wang and Chen, 2001; Qi et al., 2005; Luo et al., 2009; Xu et al., 2009; Steingrimsson and Strawderman, 2017). They applied the inverse probability weighted (IPW) technique to the existing estimation procedures for the complete covariate cases. In particular, (Yu, 2011) proposed a revised Buckley-James estimator for data missing by design. In order to deal with multi-modal block-wise missing data, (Yu et al., 2020) proposed a new direct sparse regression procedure using the estimated covariance matrix from block-missing multi-modal data (DISCOM). They first used all available information to estimate the covariance matrix of the predictors and the cross-covariance vector between the predictors and the response variable. Then they used an extended LASSO-type estimator to estimate the coefficients based on the estimated covariance matrix and cross-covariance vector. Despite its usefulness, however, the DISCOM only considers the linear regression model for uncensored data.

In this paper, we propose a regularized Buckley-James method for variable selection, parameter estimation, and prediction for right-censored outcomes with block-wise missing data. It extends

the DISCOM method (Yu et al., 2020) to right-censored survival data. Our proposed method has several attractive properties. First, our approach can handle high-dimensional data and perform variable selection. Second, it works with data with block-wise missing covariates and censored outcomes. Third, our method can still deliver reliable results even if our training data have no observation with complete covariates. Our proposed method includes two steps. The first step is to estimate each element of the covariance and cross-covariance matrices using all available observations. The second step is to use a penalized approach to estimate the sparse regression coefficient vector by the Buckley-James method. Numerical studies and the ADNI data application confirm that the proposed method performs competitively for block-wise missing data.

The remainder of this paper is organized as follows. In Section 3.2, we introduce the problem background and our model. Simulation studies and a multi-modal ADNI data example are presented in Sections 3.3 and 3.4. A brief summary of the paper is provided in Section 3.5.

3.2 Methodology

3.2.1 Problem setup and notations

Consider the following semiparametric AFT model,

$$\mathbf{T} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\epsilon}, \tag{3.1}$$

where $\boldsymbol{\beta}^* = (b_1, \ldots, b_p)^\top \in \mathbb{R}^p$ is an unknown p-dimensional vector, $\mathbf{T} = (t_1, \ldots, t_n)^\top \in \mathbb{R}^n$ is the response vector, $\mathbf{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^\top$ is the $n \times p$ design matrix and $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$ is the the error vector. Assume that $\{\boldsymbol{x}_i\}_{i=1}^n$ are i.i.d. realizations of a random vector $\boldsymbol{X} = (X_1, \ldots, X_p)^\top$ with zero mean and covariance matrix $\boldsymbol{\Sigma}_{XX} = (\sigma_{ij}^{XX}) \in \mathbb{R}^{p\times p}$. Denote $\boldsymbol{\Sigma}_{XT} = (\sigma_i^{XT}) \in \mathbb{R}^p$ as the cross-covariance vector between $\boldsymbol{x}_i$ and t_i for $1 \leq i \leq n$. Assume that the predictors come from multiple modalities and there are p_k predictors in the k-th modality. In addition, assume that $\mathbf{X}$ has block-wise missing values. That is, for each sample, its measurements in one modality can be entirely missing. Let $\tilde{\mathbf{X}} = (\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_n)^\top$ be the imputed design matrix, where the missing values in $\mathbf{X}$ are imputed by some imputation methods such as multiple imputation (Rubin, 2004) or the soft-impute algorithm (Mazumder et al., 2010). For simplicity, we use the soft-impute algorithm

to calculate $\tilde{\mathbf{X}}$ in our numerical and case studies. The errors ϵ_i for $1 \leq i \leq n$ are i.i.d. realizations from a random variable ϵ with zero mean and covariance σ_ϵ. Moreover, we further assume that $\mathbf{x}_i$ and ϵ_i are uncorrelated for $1 \leq i \leq n$.

Let $\mathbf{T}$ denote the transformed failure time, e.g., the logarithm of the conversion time from MCI to AD. Suppose that $\mathbf{C} = (c_1, \ldots, c_n)^\top \in \mathbb{R}^n$ is the transformed censoring time which is transformed in the same way as $\mathbf{T}$, with c_i being independent of t_i given $\boldsymbol{x}_i$. When $\mathbf{T}$ is right censored, we can only observe $(y_i, \delta_i, \boldsymbol{x}_i)$ for $1 \leq i \leq n$, where $y_i = \min(t_i, c_i)$, and $\delta_i = 1_{\{t_i \leq c_i\}}$ is the censoring indicator for the i-th observation.

We employ the following notation throughout this article. For a square matrix $\mathbf{C} = (c_{ii'}) \in \mathbb{R}^{p \times p}$, we denote its diagonal matrix as $\mathrm{diag}(\mathbf{C})$. For a matrix $\mathbf{A} = (a_{ij}) \in \mathbb{R}^{p \times q}$, we define the largest and smallest eigenvalues of $\mathbf{A}$ as $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$ respectively. For a vector $\mathbf{v} \in \mathbb{R}^p$, let $\|\mathbf{v}\|_1 = \sum_i |v_i|$, and $\|\mathbf{v}\|_2 = \sqrt{\sum_i v_i^2}$.

3.2.2 Regularized Buckley-James regression for complete observations

If there is no response censored and no covariate missing, then $t_i = y_i$ for $1 \leq i \leq n$ and $\mathbf{X}$ is fully observed. Then the least-squares method can be applied to estimate the parameters in model (3.1) by solving the following optimization problem

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^\top (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}),$$

where $\mathbf{Y} = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$. For a censored response with complete covariates, the key idea of the Buckley-James method is to replace the censored t_i by its expectation conditional on δ_i and $\boldsymbol{x}_i$. Define the pseudo failure time y_i^* as

$$y_i^* = \begin{cases} y_i & \delta_i = 1; \\ \mathbb{E}\left(t_i \mid t_i > y_i, \boldsymbol{x}_i\right) & \delta_i = 0. \end{cases}$$

It can be shown that $\mathbb{E}(y_i^*) = \mathbb{E}(t_i)$ for $1 \leq i \leq n$; for details see (Smith, 2017). With the true $\boldsymbol{\beta}^*$, $\mathbb{E}\left(t_i \mid t_i > y_i, \boldsymbol{x}_i\right)$ has the form of

$$\begin{aligned}\mathbb{E}\left(t_i \mid t_i > y_i, \boldsymbol{x}_i\right) &= \boldsymbol{x}_i^\top \boldsymbol{\beta}^* + \mathbb{E}\left(\epsilon_i \mid \epsilon_i > y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^*\right) \\ &= \boldsymbol{x}_i^\top \boldsymbol{\beta}^* + \int_{y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^*}^{\infty} \frac{t dF(t)}{1 - F\left(y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^*\right)},\end{aligned} \tag{3.2}$$

where F is the distribution function of residual $\epsilon_i(\boldsymbol{\beta}^*) = t_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^*$ for $1 \leq i \leq n$. The distribution of $\epsilon_i(\boldsymbol{\beta}^*)$ can be estimated nonparametrically by the Kaplan-Meier estimator (Kaplan and Meier, 1958)

$$\hat{F}(t) = 1 - \prod_{i:\epsilon_i < t} \left(1 - \frac{d_i}{n_i}\right), \tag{3.3}$$

where $d_i = \sum_{j=1}^n I(\epsilon_j = \epsilon_i \text{ and } \delta_j = 1)$ and $n_i = \sum_{j=1}^n I(\epsilon_j > \epsilon_i)$. After substituting F with $\hat{F}$ in (3.2), the $\tilde{y}_i^*$ can be simplified as

$$\tilde{y}_i^* = \delta_i y_i + (1 - \delta_i)\left(\boldsymbol{x}_i^\top \boldsymbol{\beta}^* + \int_{y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^*}^{\infty} \frac{t d\hat{F}(t)}{1 - \hat{F}\left(y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^*\right)}\right). \tag{3.4}$$

Then the least-squares method can be applied to the following regression model

$$\tilde{y}_i^* = \boldsymbol{x}_i^\top \boldsymbol{\beta}^* + \epsilon_i^*, \tag{3.5}$$

where ϵ_i^* has mean zero. Let $\tilde{\mathbf{Y}}^* = (\tilde{y}_1^*, \ldots, \tilde{y}_n^*)^\top$. The least-squares estimator of $\boldsymbol{\beta}^*$ in model (3.5) is

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}(\tilde{\mathbf{Y}}^* - \mathbf{X}\boldsymbol{\beta})^\top(\tilde{\mathbf{Y}}^* - \mathbf{X}\boldsymbol{\beta}) = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \tilde{\mathbf{Y}}^*.$$

The final solution requires an iterative procedure since values of $\tilde{y}_i^*$ defined in (3.4) contain $\boldsymbol{\beta}$.

In many areas such as genomic, medicine, and bioinformatics, the number of features p is usually much larger than the sample size n and the classical Buckley-James method fails. Regularization is needed to obtain a stable estimator of $\boldsymbol{\beta}$ with small prediction error. In this case, a modified Buckley-James approach by using penalized least-squares with the penalty term $P_\lambda(\boldsymbol{\beta})$ can be used,

where λ is the tuning parameter. To be specific, we consider the following minimization problem

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}(\mathbf{Y}^* - \mathbf{X}\boldsymbol{\beta})^\top(\mathbf{Y}^* - \mathbf{X}\boldsymbol{\beta}) + P_\lambda(\boldsymbol{\beta}), \tag{3.6}$$

where λ is the tuning parameters and can be determined by cross validation. Given an initial value $\boldsymbol{\beta}^{(0)}$, the final estimator of $\boldsymbol{\beta}$ can be calculated (3.3), (3.4) and (3.6) iteratively.

3.2.3 Regularized Buckley-James regression for block-wise missing observations

Next we extend the regularized Buckley-James regression to block-wise missing multi-modal observations. We assume that the predictors are collected from K modalities, and the k-th modality has p_k predictors for $1 \leq k \leq K$.

Recall that the regularized Buckley-James regression for complete observations iteratively estimates y_i^* by (3.4) and then solves the minimization problem (3.6). In order to handle block-wise missing data, given $\tilde{y}_i^*$, we consider the population version of the ℓ_1 penalized least-square estimator

$$\begin{aligned}\boldsymbol{\beta}^0 &= \left(\boldsymbol{\beta}_1^0, \boldsymbol{\beta}_2^0, \ldots, \boldsymbol{\beta}_p^0\right)^T \\ &= \arg\min_{\boldsymbol{\beta}} \mathbb{E}\left[\frac{1}{2}\sum_{i=1}^n \left(\tilde{y}_i^* - \boldsymbol{x}_i^\top \boldsymbol{\beta}\right)^2\right] + \lambda\|\boldsymbol{\beta}\|_1.\end{aligned}$$

If both $\boldsymbol{\Sigma}_{XX}$ and $\boldsymbol{\Sigma}_{X\tilde{Y}^*}$ are known, $\boldsymbol{\beta}^0$ can be equivalently obtained by solving the following optimization problem:

$$\boldsymbol{\beta}^0 = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}\boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{XX}\boldsymbol{\beta} - \boldsymbol{\Sigma}_{X\tilde{Y}^*}^\top \boldsymbol{\beta} + \lambda\|\boldsymbol{\beta}\|_1.$$

Therefore we can obtain the estimator $\hat{\boldsymbol{\beta}}$ if estimators for $\boldsymbol{\Sigma}_{XX}$ and $\boldsymbol{\Sigma}_{X\tilde{Y}^*}$ are available. Denote $\hat{\boldsymbol{\Sigma}}_{XX}$ as the estimator of $\boldsymbol{\Sigma}_{XX}$. Next we explain how to calculate $\hat{\boldsymbol{\Sigma}}_{XX}$ when data are block-wise missing. Define $S_{jk}^{XX} = \{i : x_{ij} \text{ and } x_{ik} \text{ are not missing}\}$, and n_{jt}^{XX} the cardinality of S_{jk}^{XX}. Let $\tilde{\boldsymbol{\Sigma}}_{XX}$ be the sample covariance matrix derived from all observed data, i.e. $\tilde{\boldsymbol{\Sigma}}_{XX} = (\tilde{\sigma}_{jt}^{XX})$, where $\tilde{\sigma}_{jt}^{XX} = \sum_{i \in S_{jt}^{XX}}(x_{ij}x_{it}/n_{jt}^{XX})$. Note that $\tilde{\boldsymbol{\Sigma}}_{XX}$ is required to be an unbiased estimator of $\boldsymbol{\Sigma}_{XX}$. When the elements in $\mathbf{X}$ are missing completely at random, the unbiasedness assumption is satisfied. However, the unbiasedness assumption can also hold under some other missing mechanisms.

Since the data $\mathbf{X}$ are block-wise missing, the estimator $\tilde{\mathbf{\Sigma}}_{XX}$ defined above can be ill-conditioned. As a result, $\tilde{\mathbf{\Sigma}}_{XX}$ is not a good estimator of $\mathbf{\Sigma}_{XX}$. Thus it cannot be used directly in our optimization problem. To resolve this problem, we partition $\tilde{\mathbf{\Sigma}}_{XX}$ into K^2 blocks, denoted as $\tilde{\mathbf{\Sigma}}^{k_1k_2} \in \mathbb{R}^{p_{k_1} \times p_{k_2}}$ for $1 \leq k_1, k_2 \leq K$. We let

$$\tilde{\mathbf{\Sigma}}_I = \begin{pmatrix} \tilde{\mathbf{\Sigma}}^{11} & & & \\ & \tilde{\mathbf{\Sigma}}^{22} & & \\ & & \ddots & \\ & & & \tilde{\mathbf{\Sigma}}^{KK} \end{pmatrix} \quad \text{and} \quad \tilde{\mathbf{\Sigma}}_C = \begin{pmatrix} \mathbf{0} & \tilde{\mathbf{\Sigma}}^{12} & \dots & \tilde{\mathbf{\Sigma}}^{1K} \\ \tilde{\mathbf{\Sigma}}^{21} & \mathbf{0} & \dots & \tilde{\mathbf{\Sigma}}^{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathbf{\Sigma}}^{K1} & \tilde{\mathbf{\Sigma}}^{K2} & \dots & \mathbf{0} \end{pmatrix},$$

where $\tilde{\mathbf{\Sigma}}_I$ is a $p \times p$ block-diagonal matrix containing K diagonal blocks of $\tilde{\mathbf{\Sigma}}_{XX}$, and $\tilde{\mathbf{\Sigma}}_C = \tilde{\mathbf{\Sigma}}_{XX} - \tilde{\mathbf{\Sigma}}_I$ is a $p \times p$ matrix containing all off-diagonal blocks of $\tilde{\mathbf{\Sigma}}_{XX}$. Here, $\tilde{\mathbf{\Sigma}}_I$ and $\tilde{\mathbf{\Sigma}}_C$ are called the intra-modality and cross-modality sample covariance matrices, respectively. Since data are block-wise missing, we use more data to estimate the entries in $\tilde{\mathbf{\Sigma}}_I$ than those in the $\tilde{\mathbf{\Sigma}}_C$. Thus the estimator $\tilde{\mathbf{\Sigma}}_I$ can be relatively more accurate than $\tilde{\mathbf{\Sigma}}_C$. We linearly combine $\tilde{\mathbf{\Sigma}}_I$ and $\tilde{\mathbf{\Sigma}}_C$ with different weights to estimate $\mathbf{\Sigma}_{XX}$. In addition, as in (Yu et al., 2020), we adopt the idea of shrinkage estimation of the covariance matrix (Fisher and Sun, 2011) and add the diagonal matrix $\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)$ to our estimator to ensure the resulting estimator to be positive definite. We let

$$\hat{\mathbf{\Sigma}}_{XX} = \alpha_1 \tilde{\mathbf{\Sigma}}_I + (1 - \alpha_1)\,\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I) + \alpha_2 \tilde{\mathbf{\Sigma}}_C, \tag{3.7}$$

where $\alpha_1, \alpha_2 \in [0, 1]$ are two shrinkage weights. The diagonal matrix $(1 - \alpha_1)\,\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)$ in (3.7) ensures that the diagonal entries of our estimator are not shrunk. The eigenvalues of $\hat{\mathbf{\Sigma}}_{XX}$ is larger than or equal to $\alpha_1 \lambda_{\min}(\tilde{\mathbf{\Sigma}}_I) + (1 - \alpha_1)\lambda_{\min}(\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)) + \alpha_2 \lambda_{\min}(\tilde{\mathbf{\Sigma}}_C)$ by Weyl's theorem, where $(1 - \alpha_1)\lambda_{\min}(\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)) > 0$ since $\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I)$ is a positive-definite matrix. Thus $\hat{\mathbf{\Sigma}}_{XX}$ is guaranteed to be positive definite by carefully selecting the tuning parameters α_1 and α_2. In practice, α_1 and α_2 can be chosen from the set $\{(\alpha_1, \alpha_2) : \alpha_1 \in [0, 1], \alpha_2 \in [0, 1], \hat{\mathbf{\Sigma}}_{XX}$ is positive semidefinite$\}$ by cross-validation or using an additional tuning dataset.

Let $\tilde{y}_i^{*(m)}$ be the i-th failure time calculated in the m-th step of Buckley-James method, $\tilde{\mathbf{Y}}_i^{*(m)} = (\tilde{y}_1^{*(m)}, \dots, \tilde{y}_n^{*(m)})^\top$, $\mathbf{\Sigma}_{X\tilde{Y}^*}^{(m)}$ be the covariance vector between $\mathbf{X}$ and $\tilde{\mathbf{Y}}^{*(m)}$, and $\hat{\mathbf{\Sigma}}_{X\tilde{Y}^*}^{(m)}$ be an estimator of $\mathbf{\Sigma}_{X\tilde{Y}^*}^{(m)}$. Next we discuss how to calculate $\hat{\mathbf{\Sigma}}_{X\tilde{Y}^*}^{(m)}$ when $\mathbf{X}$ is block-wise missing.

let $\boldsymbol{\beta}^{(m-1)}$ be the coefficient vector derived in the $(m-1)$-th step. In the m-th step, $\tilde{y}^{*(m)}$ is defined as

$$\tilde{y}_i^{*(m)} = \delta_i y_i + (1-\delta_i)\left(\boldsymbol{x}_i^\top \boldsymbol{\beta}^{(m-1)} + \int_{y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^{(m-1)}}^{\infty} \frac{t d\tilde{F}^{(m)}(t)}{1-\tilde{F}^{(m)}\left(y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^{(m-1)}\right)}\right),$$

where $\tilde{F}^{(m)}$ is the estimated distribution function of $t_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^{(m-1)}$. However, since $\mathbf{X}$ is block-wise missing, $\tilde{y}_i^{*(m)}$ can not be calculated directly. In order to estimate $\boldsymbol{\Sigma}_{X\tilde{Y}^*}^{(m-1)}$, we decompose it as

$$\begin{aligned}
\boldsymbol{\Sigma}_{X\tilde{Y}^*}^{(m-1)} &= \mathbb{E}(\mathbf{X}^\top \tilde{\mathbf{Y}}^{*(m)}) \\
&= \mathbb{E}(\mathbf{X}^\top (\mathbf{X}(\boldsymbol{\beta}^{(m-1)}) + \tilde{\mathbf{E}}^{*(m)})) \\
&= \mathbb{E}(\mathbf{X}^\top \mathbf{X})\boldsymbol{\beta}^{(m-1)} + \mathbb{E}(\mathbf{X}\tilde{\mathbf{E}}^{*(m)}),
\end{aligned}$$

where $\tilde{\mathbf{E}}^{*(m)} = (\tilde{e}_1^*(\boldsymbol{\beta}^{(m-1)}), \ldots, \tilde{e}_n^*(\boldsymbol{\beta}^{(m-1)}))^\top$ and

$$\tilde{e}_i^*(\boldsymbol{\beta}^{(m-1)}) = \begin{cases} y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^{(m-1)} & \delta_i = 1; \\ \int_{y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^{(m-1)}}^{\infty} \frac{t d\tilde{F}^{(m)}(t)}{1-\tilde{F}^{(m)}\left(y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}^{(m-1)}\right)} & \delta_i = 0. \end{cases}$$

Let $\boldsymbol{\Sigma}_{X\tilde{E}^*}^{(m)}$ be the covariance vector between $\mathbf{X}$ and $\tilde{\mathbf{E}}^{*(m)}$, and $\hat{\boldsymbol{\Sigma}}_{X\tilde{E}^*}^{(m)}$ be an estimator of $\boldsymbol{\Sigma}_{X\tilde{E}^*}^{(m)}$. Then we can estimate $\boldsymbol{\Sigma}_{X\tilde{Y}^*}^{(m)}$ as

$$\hat{\boldsymbol{\Sigma}}_{X\tilde{Y}^*}^{(m)} = \hat{\boldsymbol{\Sigma}}_{XX}\boldsymbol{\beta}^{(m-1)} + \hat{\boldsymbol{\Sigma}}_{X\tilde{E}^*}^{(m)}. \tag{3.8}$$

Define $S_j^X = \{i : x_{ij} \text{ is not missing}\}$ and let n_j^X as the cardinality of S_j^X. In order to estimate $\boldsymbol{\Sigma}_{X\tilde{E}^*}^{(m)}$, let $\hat{\mathbf{E}}^{*(m)} = (\hat{e}_1^*(\boldsymbol{\beta}^{(m-1)}), \ldots, \hat{e}_n^*(\boldsymbol{\beta}^{(m-1)}))$ and

$$\hat{e}_i^*(\boldsymbol{\beta}^{(m-1)}) = \begin{cases} y_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)} & \delta_i = 1; \\ \int_{y_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)}}^{\infty} \frac{t d\hat{F}^{(m)}(t)}{1-\hat{F}^{(m)}\left(y_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)}\right)} & \delta_i = 0. \end{cases}$$

Here $\tilde{\boldsymbol{x}}_i$ are the imputed predictors and $\hat{F}^{(m)}$ is the estimated distribution function of $t_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)}$. Define $\tilde{\boldsymbol{\Sigma}}_{X\tilde{E}^*}^{(m)}$ as the sample covariance matrix using all available data, i.e. $\tilde{\boldsymbol{\Sigma}}_{X\tilde{E}^*}^{(m)} = (\tilde{\sigma}_j^{X\tilde{E}^*,(m)})$, where $\tilde{\sigma}_j^{X\tilde{E}^*,(m)} = \sum_{i \in S_j^X} x_{ij}\hat{e}_i^*/n_j^X$. Since our estimator $\hat{\boldsymbol{\Sigma}}_{XX}$ in (3.7) is a shrinkage estimator, we

also use a shrinkage estimator to estimate $\mathbf{\Sigma}^{(m)}_{X\tilde{E}^*}$ by

$$\hat{\mathbf{\Sigma}}^{(m)}_{X\tilde{E}^*} = \alpha_3 \tilde{\mathbf{\Sigma}}^{(m)}_{X\tilde{E}^*}, \tag{3.9}$$

where $\alpha_3 \in [0, 1]$ is the shrinkage weight. In practice, α_3 can also be chosen by cross-validation or using an additional tuning dataset.

In summary, given $\hat{\mathbf{\Sigma}}_{XX}$ and $\hat{\mathbf{\Sigma}}^{(m)}_{X\tilde{E}^*}$ as defined in (3.7) and (3.9), in the m-th iteration of the regularized Buckley-James method, we solve the optimization problem

$$\boldsymbol{\beta}^{(m)} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}\boldsymbol{\beta}^\top \hat{\mathbf{\Sigma}}_{XX}\boldsymbol{\beta} - \boldsymbol{\beta}^{(m-1)\top}\hat{\mathbf{\Sigma}}^\top_{XX}\boldsymbol{\beta} + \hat{\mathbf{\Sigma}}^{(m)\top}_{X\tilde{E}^*}\boldsymbol{\beta} + \lambda\|\boldsymbol{\beta}\|_1 \tag{3.10}$$

by the proximal gradient descent algorithm (Parikh et al., 2014).

In Algorithm 4, we summarized the major steps for our proposed method, DISCOM-BJ, given a set of tuning parameters $(\alpha_1, \alpha_2, \alpha_3, \lambda)$.

We make two important remarks about the proposed procedure. First, our method applies to any penalty for linear models, including LASSO and elastic net (Zou and Hastie, 2005). Secondly, to be numerically effective, the starting values $\boldsymbol{\beta}^{(0)}$ may be obtained by using the least-squares estimator treating all observations as uncensored (Buckley and James, 1979). Other choices, e.g. using only uncensored observations, are also feasible.

3.3 Numerical Study

We perform some numerical studies to compare our proposed method (DISCOM-BJ) with some other methods, which include

1. ℓ_2-BJ, which applies the regularized Buckley-James regression to samples with complete observations and uses $P_\lambda(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_2$;

2. Imputed-ℓ_2-BJ, which applies the regularized Buckley-James regression to all samples with missing values being imputed by the soft-thresholded SVD method and uses $P_\lambda(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_2$;

3. ℓ_1-BJ, which applies the regularized Buckley-James regression to samples with complete observations and uses $P_\lambda(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_1$;

Algorithm 4: Regularized Buckley-James method by using covariance from multi-modality data

Input: $\{(y_i, \delta_i, \boldsymbol{x}_i), 1 \leq i \leq n\}$, λ

Output: $\boldsymbol{\beta}^{(m)}$

Let $\boldsymbol{\beta}^{(0)}$ be the initial value of $\boldsymbol{\beta}$.

while $\left|\boldsymbol{\beta}^{(m)} - \boldsymbol{\beta}^{(m-1)}\right| > d$ **do**

Compute $\epsilon_i(\boldsymbol{\beta}^{(m-1)})$ for $1 \leq i \leq n$ by

$$\epsilon_i\left(\boldsymbol{\beta}^{(m-1)}\right) = y_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)},$$

where $\tilde{\boldsymbol{x}}_i$ is the imputed predictors of the i-th observation.

Compute $\hat{F}^{(m)}(t)$ by

$$\hat{F}^{(m)}(t) = 1 - \prod_{i:\epsilon_i(\boldsymbol{\beta}^{(m-1)})<t} \left(1 - \frac{d_i}{n_i}\right),$$

where $d_i = \sum_{j=1}^n I[\epsilon_j(\boldsymbol{\beta}^{(m-1)}) = \epsilon_i(\boldsymbol{\beta}^{(m-1)})$ and $\delta_j = 1]$ and $n_i = \sum_{j=1}^n I(\epsilon_j(\boldsymbol{\beta}^{(m-1)}) > \epsilon_i(\boldsymbol{\beta}^{(m-1)}))$.

Compute $\tilde{\mathbf{E}}^{*(m)} = (e_1^*(\boldsymbol{\beta}^{(m-1)}), \ldots, e_n^*(\boldsymbol{\beta}^{(m-1)}))^\top$ by

$$e_i^*(\boldsymbol{\beta}^{(m-1)}) = \begin{cases} y_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)} & \delta_i = 1; \\ \int_{y_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)}}^{\infty} \frac{t d\hat{F}^{(m)}(t)}{1 - \hat{F}^{(m)}\left(y_i - \tilde{\boldsymbol{x}}_i^\top \boldsymbol{\beta}^{(m-1)}\right)} & \delta_i = 0. \end{cases}$$

Compute $\hat{\boldsymbol{\Sigma}}_{XX}$ and $\hat{\boldsymbol{\Sigma}}_{X\tilde{E}^*}^{(m)}$ by (3.7) and (3.9) respectively.

Update $\boldsymbol{\beta}^{(m)}$ by

$$\boldsymbol{\beta}^{(m)} = \min_{\boldsymbol{\beta}} \frac{1}{2}\boldsymbol{\beta}^\top \hat{\boldsymbol{\Sigma}}_{XX}\boldsymbol{\beta} - \boldsymbol{\beta}^{(m-1)\top}\hat{\boldsymbol{\Sigma}}_{XX}^\top \boldsymbol{\beta} + \hat{\boldsymbol{\Sigma}}_{X\tilde{E}^*}^{(m)\top}\boldsymbol{\beta} + \lambda\|\boldsymbol{\beta}\|_1.$$

end

return $\boldsymbol{\beta}^{(m)}$.

4. Imputed-ℓ_1-BJ, which applies the regularized Buckley-James regression to all samples with missing values being imputed by the soft-thresholded SVD method and uses $P_\lambda(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_1$;

5. Boosting-BJ, which applies the Buckley-James boosting method with linear least-squares (Wang and Wang, 2010) to samples with complete observations;

6. Imputed-Boosting-BJ, which applies the Buckley-James boosting method with linear-least squares (Wang and Wang, 2010) to all samples with missing values being imputed by the soft-thresholded SVD method.

For all examples, we generate the natural logarithm of the true survival time by

$$T = \boldsymbol{x}^\top \boldsymbol{\beta} + \epsilon, \text{ where } \epsilon \sim N(0,1).$$

and set $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^\top \sim N(\mathbf{0}, \boldsymbol{\Sigma})$ with $\boldsymbol{\Sigma} = (\sigma_{jt})$, where $\sigma_{jt} = 0.6^{|j-t|}$. The data are generated from three modalities whose dimensions p_1, p_2 and p_3 are specified in each example. The true coefficient vector is

$$\boldsymbol{\beta} = (b, b, b, \underbrace{0, \cdots, 0}_{p_1-3}, b, b, b, \underbrace{0, \cdots, 0}_{p_2-3}, b, b, b, \underbrace{0, \cdots, 0}_{p_3-3}),$$

where b is a constant. We generate $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \overset{\text{iid}}{\sim} N(0,1)$. The censoring time $\mathbf{C}$ is generated from unif(τ_l, τ_u), where τ_l, τ_u are tuned to achieve the desired censoring rate. The censoring rates are specified in each example.

The training dataset contains 25 samples with complete observations, 25 samples with observations from the third modality, 25 samples with observations from the first and the third modalities and 25 samples with observations from the first modality. The tuning dataset contains 100 samples with complete observations without censoring response and the testing dataset includes 400 samples with complete observations without censoring response. For each method, we train our model with different tuning parameters on the training dataset. Then we choose the optimal tuning parameters minimizing the mean squared error on the tuning dataset.

For each example, the experiment is repeated 50 times. To evaluate the selection performance of the algorithm, we use false-positive rate (FPR) and false-negative rate (FNR) defined

as FPR $=$ FP$/$(FP $+$ TN) and FNR $=$ FN$/$(FN $+$ TP), where FN is the number of coefficients wrongly estimated as zero, TN is the number coefficients rightfully estimated as zero, TP is the number of coefficients rightfully estimated as nonzero and FP is the number of coefficients wrongly estimated as nonzero. Furthermore, to evaluate the accuracy of our estimators, the mean squared error MSE $= \|\mathbf{T}_{\text{test}} - \hat{\mathbf{T}}_{\text{test}}\|_2$ on the testing dataset and the ℓ_2 distance $\|\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}\|_2$ are used as the criteria, where T_{test} is the logarithm of the survival time vector in the test dataset, $\hat{T}_{\text{test}}$ is the logarithm of the predicted survival time vector in the test dataset, and $\hat{\boldsymbol{\beta}}$ is the estimated coefficient vector.

In Example 1, we examine how our method performs with various signal-to-noise ratios. We set $p = 90$, $p_1 = p_2 = p_3 = 30$ and the censoring rate equal to 50%. In Example 1(a) and 1(b), we set b to be 0.5 and 2 respectively.

In Example 2, we examine how our method performs with various p. We set $b = 1$ and the censoring rate equal to 50%. In Example 2(a), we set p to be 60, where $p_1 = p_2 = p_3 = 20$. In Example 2(b), we set p to be 120, where $p_1 = p_2 = p_3 = 40$.

In Example 3, we examine how our method performs with various censoring rates. We set $p = 90$, $p_1 = p_2 = p_3 = 30$ and $b = 1$. In Example 3(a) to 3(f), we respectively let $(\tau_l, \tau_u) \in \{(1, 6.8950), (1, 3.64), (1, 1.21), (-5, 5), (-5, 2.515), (-5, 0.16)\}$ such that the yielding censoring rate $\mathbb{P}(T > C)$ ranges from 0.2 to 0.7 with an increment of 0.1.

We report the simulation results in Tables 3.1, 3.2 and 3.3. Table 3.1 shows the results of Example 1 with two different signal to noise ratios. Table 3.2 shows the results of Example 2 with two different dimensions. Table 3.3 shows the results of Example 3 with different censoring rates. Based on the results, we can see that imputed versions of ℓ_2-BJ, ℓ_1-BJ and Boosting-BJ perform better than the un-imputed version of these methods in terms of the parameter estimation and variable selection. Compared with other exisiting methods, our proposed DISCOM-BJ delivers the best performance in all these three examples.

3.4 Application to the ADNI study

We apply the DISCOM-BJ to the ADNI study (Mueller et al., 2005) and compare it with several other approaches. A primary goal of this analysis is to identify biological markers and

	Example 1(a) [low signal to noise ratio]			
	MSE	EST	FPR	FNR
ℓ_2-BJ	4.14 (0.09)	1.34 (0.01)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	2.91 (0.07)	1.21 (0.01)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	3.64 (0.12)	1.40 (0.02)	0.17 (0.02)	0.49 (0.03)
Imputed-ℓ_1-BJ	**2.56** (0.09)	1.27 (0.03)	0.16 (0.01)	0.31 (0.02)
Boosting-BJ	4.37 (0.15)	1.54 (0.03)	0.07 (0.00)	0.58 (0.02)
Imputed-Boosting-BJ	2.85 (0.09)	1.22 (0.02)	**0.06** (0.00)	0.34 (0.02)
DISCOM-BJ	**2.51** (0.09)	**1.21**(0.03)	0.18 (0.02)	**0.26** (0.02)
	Example 1(b) [high signal to noise ratio]			
ℓ_2-BJ	47.19 (1.22)	5.31 (0.05)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	26.26 (0.80)	4.62 (0.05)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	29.93 (1.49)	4.54 (0.09)	0.23 (0.02)	0.24 (0.02)
Imputed-ℓ_1-BJ	16.84 (0.80)	4.22 (0.09)	0.20 (0.01)	0.14 (0.01)
Boosting-BJ	41.40 (1.70)	5.32 (0.09)	0.06 (0.00)	0.42 (0.03)
Imputed-Boosting-BJ	25.67 (0.98)	4.37 (0.08)	**0.04**(0.00)	0.22 (0.02)
DISCOM-BJ	**15.16** (0.63)	**3.87** (0.08)	0.19 (0.01)	**0.09** (0.01)

Table 3.1: Performance comparison of different methods for Example 1 with different signal to noise ratios. The values in the parentheses are the standard errors of the measures.

	Example 2(a) [$p = 60$]			
	MSE	EST	FPR	FNR
ℓ_2-BJ	10.87 (0.34)	2.49 (0.03)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	5.81 (0.19)	2.16 (0.03)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	7.58 (0.40)	2.38 (0.04)	0.24 (0.02)	0.28 (0.03)
Imputed-ℓ_1-BJ	4.77 (0.17)	2.14 (0.04)	0.25 (0.01)	0.14 (0.02)
Boosting-BJ	10.28 (0.39)	2.65 (0.04)	0.08 (0.00)	0.43 (0.02)
Imputed-Boosting-BJ	6.91 (0.22)	2.16 (0.04)	**0.06** (0.00)	0.23 (0.02)
DISCOM-BJ	**4.46** (0.18)	**1.97** (0.04)	0.29 (0.02)	**0.09** (0.02)
	Example 2(b) [$p = 120$]			
ℓ_2-BJ	13.73 (0.31)	2.73 (0.02)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	8.84 (0.24)	2.43 (0.02)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	9.91 (0.45)	2.52 (0.05)	0.14 (0.01)	0.37 (0.03)
Imputed-ℓ_1-BJ	5.83 (0.22)	2.26 (0.05)	0.16 (0.01)	0.19 (0.01)
Boosting-BJ	12.65 (0.51)	2.86 (0.05)	0.06 (0.00)	0.48 (0.02)
Imputed-Boosting-BJ	7.23 (0.26)	2.20 (0.04)	**0.04** (0.00)	0.24 (0.02)
DISCOM-BJ	**5.52** (0.24)	**2.08** (0.05)	0.16 (0.01)	**0.13** (0.02)

Table 3.2: Performance comparison of different methods for Example 2 with different dimensions. The values in the parentheses are the standard errors of the measures.

	Example 3(a) [$\mathbb{P}(T > C) = 0.2$]			
	MSE	EST	FPR	FNR
ℓ_2-BJ	9.39 (0.25)	2.43 (0.02)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	5.88 (0.13)	2.15 (0.02)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	5.85 (0.31)	2.07 (0.05)	0.26 (0.02)	0.15 (0.02)
Imputed-ℓ_1-BJ	4.18 (0.16)	1.99 (0.04)	0.18 (0.01)	0.10 (0.01)
Boosting-BJ	8.10 (0.34)	2.35 (0.05)	0.06 (0.00)	0.31 (0.02)
Imputed-Boosting-BJ	5.18 (0.19)	1.93 (0.03)	**0.04** (0.00)	0.13 (0.01)
DISCOM-BJ	**3.81** (0.14)	**1.81** (0.04)	0.16 (0.01)	**0.07** (0.01)
	Example 3(b) [$\mathbb{P}(T > C) = 0.3$]			
ℓ_2-BJ	10.40 (0.25)	2.52 (0.03)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	6.31 (0.14)	2.20 (0.02)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	6.82 (0.31)	2.16 (0.04)	0.26 (0.02)	0.19 (0.02)
Imputed-ℓ_1-BJ	4.40 (0.16)	2.01 (0.04)	0.19 (0.01)	0.10 (0.01)
Boosting-BJ	9.08 (0.35)	2.43 (0.04)	0.06 (0.00)	0.35 (0.02)
Imputed-Boosting-BJ	5.71 (0.20)	1.98 (0.03)	**0.04** (0.00)	0.16 (0.02)
DISCOM-BJ	**4.14** (0.13)	**1.85** (0.03)	0.19 (0.01)	**0.09** (0.01)
	Example 3(c) [$\mathbb{P}(T > C) = 0.4$]			
ℓ_2-BJ	11.56 (0.28)	2.57 (0.02)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	6.92 (0.16)	2.27 (0.02)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	7.76 (0.32)	2.27 (0.04)	0.24 (0.02)	0.24 (0.02)
Imputed-ℓ_1-BJ	4.78 (0.16)	2.08 (0.04)	0.21 (0.01)	0.11 (0.01)
Boosting-BJ	10.36 (0.36)	2.56 (0.04)	0.06 (0.00)	0.40 (0.02)
Imputed-Boosting-BJ	6.47 (0.19)	2.07 (0.03)	**0.05** (0.00)	0.17 (0.02)
DISCOM-BJ	**4.48** (0.14)	**1.90** (0.04)	0.22 (0.02)	**0.08** (0.01)
	Example 3(d) [$\mathbb{P}(T > C) = 0.5$]			
ℓ_2-BJ	12.65 (0.31)	2.64 (0.03)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	7.46 (0.22)	2.32 (0.03)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	8.79 (0.39)	2.42 (0.05)	0.17 (0.01)	0.34 (0.03)
Imputed-ℓ_1-BJ	5.41 (0.28)	2.19 (0.05)	0.19 (0.01)	0.18 (0.02)
Boosting-BJ	11.46 (0.53)	2.72 (0.06)	0.06 (0.00)	0.43 (0.02)
Imputed-Boosting-BJ	7.14 (0.29)	2.17 (0.05)	**0.04** (0.00)	0.24 (0.02)
DISCOM-BJ	**5.05** (0.23)	**2.03** (0.05)	0.21 (0.02)	**0.12** (0.01)
	Example 3(e) [$\mathbb{P}(T > C) = 0.6$]			
ℓ_2-BJ	14.36 (0.34)	2.75 (0.03)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	8.95 (0.26)	2.44 (0.03)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	10.62 (0.39)	2.54 (0.04)	0.20 (0.02)	0.37 (0.03)
Imputed-ℓ_1-BJ	6.36 (0.31)	2.31 (0.05)	0.18 (0.01)	0.22 (0.02)
Boosting-BJ	13.95 (0.55)	2.99 (0.05)	0.06 (0.00)	0.52 (0.02)
Imputed-Boosting-BJ	8.81 (0.32)	2.39 (0.05)	**0.04** (0.00)	0.30 (0.02)
DISCOM-BJ	**5.84** (0.29)	**2.16** (0.05)	0.22 (0.02)	**0.17** (0.02)
	Example 3(f) [$\mathbb{P}(T > C) = 0.7$]			
ℓ_2-BJ	15.76 (0.36)	2.83 (0.03)	1.00 (0.00)	0.00 (0.00)
Imputed-ℓ_2-BJ	10.82 (0.30)	2.61 (0.03)	1.00 (0.00)	0.00 (0.00)
ℓ_1-BJ	12.09 (0.40)	2.61 (0.03)	0.21 (0.03)	0.44 (0.02)
Imputed-ℓ_1-BJ	7.42 (0.31)	2.39 (0.05)	0.18 (0.01)	0.28 (0.02)
Boosting-BJ	17.29 (0.61)	3.36 (0.05)	0.06 (0.00)	0.59 (0.02)
Imputed-Boosting-BJ	11.39 (0.33)	2.74 (0.04)	**0.04** (0.00)	0.36 (0.02)
DISCOM-BJ	**6.83** (0.28)	**2.24** (0.04)	0.36 (0.04)	**0.12** (0.02)

Table 3.3: Performance comparison of different methods for Example 3 with different censoring rates. The values in the parentheses are the standard errors of the measures.

neuropsychological assessments to measure the progression of MCI and early AD. We are interested in predicting the time to convert to state AD of patients who was initially diagnosed as MCI in the ADNI study. We extract biomarkers from three complementary data sources: MRI, PET and CSF. Note that, as (Xue and Qu, 2021) stated, our sparsity assumption of the proposed method may not be suitable for raw imaging data or imaging data at small scales since images have to show some visible atrophy for AD. However, the sparsity assumption can still be reasonable for the region of interest (ROI) level data. Thus, we apply the DISCOM-BJ to the ROI level data in ADNI.

We process the image data following the similar procedure as in (Yu et al., 2020). For the MRI, after correction, spatial segmentation and registration steps, we obtain the image for each subject based on the Jacob template with 93 manually labeled ROIs. For each of the 93 ROIs in the labeled MRI, we compute the volume of gray matter as a feature. For each PET image, we first align the PET image to its respective MRI image using affine registration. Then, we calculate the average intensity of every ROI in the PET image as a feature. For the CSF modality, five biomarkers are used in this study, namely amyloid $\beta(A\beta 42)$, CSF total tau (t-tau), tau hyperphosphorylated at threonine 181 (p-tau), and two tau ratios with respective to $A\beta 42$ (i.e., t-tau/$A\beta 42$ and p-tau /$A\beta 42$)

After data processing, we have 93 features from MRI, 93 features from PET and 5 features from CSF. There are 376 subjects in total, including 56 subjects with complete MRI, PET, CSF features and uncensored response, 38 subjects with complete MRI, PET, CSF features and censored response, 101 subjects with MRI and PET features only, 89 subjects with MRI and CSF features only, and 92 subjects with MRI features only.

In our analysis, we divide the data into training, tuning, and testing sets. The training set consists of all subjects with incomplete observations and 40 randomly selected subjects with complete features. The tuning set consists of another 18 randomly selected subjects with complete observations. The testing set contains the remaining 36 subjects with complete observations. We train our model with different tuning parameters on the training set. Then we choose the tuning parameters which minimize the mean squared error on the tuning set. The testing set is used to evaluate different methods. We used all methods shown in the simulation study to predict the conversion time from MCI to AD. For each method, the analysis is repeated 50 times using different

method	ℓ_2-BJ	Imputed-ℓ_2-BJ	ℓ_1-BJ	Imputed-ℓ_1-BJ	DISCOM-BJ
MSE	0.99(0.04)	1.01(0.04)	0.88(0.03)	0.88(0.04)	**0.84(0.02)**

Table 3.4: Performance comparison for the ADNI data. The values in the parentheses are the standard errors of the measures.

Top features selected by DISCOM-BJ
Uncus left
Hippocampal formation left
Middle temporal gyrus right;
Precuneus left;
Angular gyrus left;
amyloid $\boldsymbol{\beta}$ (A$\boldsymbol{\beta}$42)
CSF total tau(t-tau);
tau hyperphosphorylated at threonine 181

Table 3.5: Top 8 features selected by DISCOM-BJ.

partitions of the data. In addition to the sum of MSE of all three responses. We also compare the number of features selected by each method.

The results in Table 3.4 show that our proposed DISCOM-BJ method acquires the best prediction performance with smaller MSE than ℓ_1-BJ, Imputed-ℓ_1-BJ , ℓ_2-BJ and Imputed-ℓ_2-BJ. To further understand our results, since each MRI and PET features correspond to one ROI, we can examine whether the selected features are meaningful by studying their corresponding brain regions. Table 3.5 shows the names of top 8 features selected by our method, where the first 5 features are ROIs, and the last 3 features correspond to the CSF modality. Figure 3.1 shows these 5 ROIs of the brain. Among these 5 brain regions, some regions such as uncus left, middle temporal gyrus left and hippocampus formation left are known to be highly correlated with AD and MCI by many studies using group comparison methods (Misra et al., 2009; Zhang et al., 2012a). It would be interesting to study whether the other two brain regions (Middle temporal gyrus right and Angular gyrus left) are truly related to the conversion from MCI to AD.

3.5 Conclusion

In this paper, we propose an ℓ_1-penalized Buckley-James method using block-missing multi-modal predictors and censored responses. In each iteration of Buckley-James method, with pseudo responses, we first estimate the covariance matrix of the predictors using a linear combination

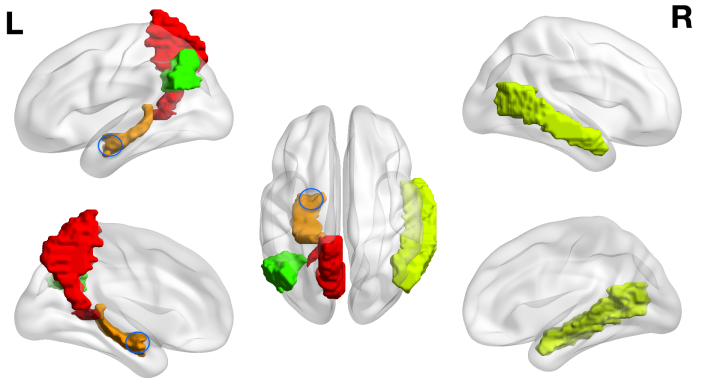

Figure 3.1: Top 5 brain regions selected by DISCOM-BJ, where the uncus left region is highlighted by the blue circle.

of the estimates of the variance of each predictor, the intra-modality covariance matrix, and the cross-modality covariance matrix. The proposed estimator of the covariance matrix can be positive semidefinite and more accurate than the sample covariance matrix. In the second step of each iteration, based on the estimated covariance matrix, a penalized estimator is used to deliver a sparse estimate of the coefficients. Extensive simulation studies also indicate that our method has promising performance in estimation, prediction and model selection for the block-missing multi-modal data. Finally, we apply the DISCOM-BJ method to the ADNI dataset to predict the conversion time of the patients from MCI to AD. We demonstrate that our model has accurate prediction and meaningful interpretation.

CHAPTER 4

Adaptive Supervised Learning on Data Streams in Reproducing Kernel Hilbert Spaces with Data Sparsity Constraint

4.1 Introduction

With the advance in technology, the volume of data generation is increasing at a very rapid rate. Due to the challenges of big data in many applications, streaming data analysis has attracted considerable attention. Supervised learning methods analyzing streaming data need to address several challenges, such as limited storage and concept drift. Specifically, the amount of memory required by the algorithms becomes infeasible as the number of samples in the data streams increases (Langford et al., 2009). Moreover, sometimes the data stream exhibits a phenomenon referred to as concept drift (Tsymbal, 2004), in which the underlying model evolves, causing the model constructed using old samples to become not applicable to new observations. Traditional machine learning algorithms may not be able to provide a good model as they may not adapt to the new changes.

The stochastic gradient descent (SGD) algorithm (Robbins and Monro, 1951), which can efficiently handle large-scale data sets, has gained increasing attention in developing supervised learning tools for data streams (Rosenblatt, 1958; Littlestone, 1988; Hazan et al., 2007). Given a convex loss function and a training set, researchers can use the SGD to obtain a sequence of models that converge to the optimal model. For many supervised learning problems, linear models can be suboptimal when the response has a nonlinear relationship with the predictors.

To improve the flexibility of the model, various nonlinear regression models (Friedman et al., 2001) can be used. Online learning with kernels (Kivinen et al., 2004) embeds the model in a reproducing kernel Hilbert space (RKHS) (Aronszajn, 1950) to address the nonlinear relationship in the model. Since the regression function is assumed to be in a RKHS, it is common to take the squared norm of the regression function as the penalty. By the representer theorem (Kimeldorf

and Wahba, 1971), the resulting regression function can be represented as a linear combination of kernel functions determined by the training data. In addition to the typical squared norm penalty, Zhang et al. (2016) introduced a data sparsity constraint. Zhang et al. (2016) showed that the regression model with the data sparsity constraint can have competitive prediction performance for various problems, especially when the sample size is small or moderate, or a sparse representation of the data can reasonably approximate the underlying function. The data points corresponding to kernel functions with non-zero coefficients are called support vectors (SVs).

The size of SVs grows linearly over time, posing storage and computational problems for these models. This may result in increasing storage space and training time. To resolve this issue, researchers have developed several different approaches. A family of algorithms, called "budget online kernel learning", has been proposed to bound the number of SVs with a fixed budget. Cavallanti et al. (2007) and Zhao et al. (2012) discarded one of the existing SVs uniformly during the training process. Dekel et al. (2005) discarded the oldest SVs during the training process. Orabona et al. (2008) used a new kernel function to approximate the removed SVs. These methods may suffer information loss when removing or approximating the SVs.

Another promising strategy is to explore the functional approximation techniques for achieving scalable kernel learning (Lu et al., 2016). The key idea is to construct a kernel-induced feature representation such that the inner product of instances in the new feature space can effectively approximate the kernel function. Because of the approximation, the model can suffer from high variation. As pointed out by Sun et al. (2018), the number of random features needed for a consistent estimation grows when the number of SVs increases.

Many machine learning algorithms focus on a fixed model, where the relationship between the responses and the covariates doesn't change over time. However, learning a fixed function may not always be suitable for data streams. Many data streams are non-stationary. As a result, the underlying model may change over time. This problem is also known as concept drift, where the conditional distribution of the response given the predictors changes over time. Concept drift can affect the learner's performance if not handled properly. There are many algorithms in the literature for this issue. Schaul et al. (2013) introduced the vSGD for non-stationary models. In particular, in each step of the vSGD, the algorithm determines the learning rate adaptively to minimize the

loss function by using a quadratic approximation of the objective function. One drawback of vSGD is that it may not be able to capture the model correctly when it changes rapidly.

Another common way to deal with concept drift is to detect changes and react accordingly. Concept drift can be detected by its effect on characteristic features of the model, such as the regression or classification accuracy. Such quantitative features can be accompanied by statistical tests to assess the significance. Such tests can rely on some well-known statistics, such as the Hoeffding bound (Frias-Blanco et al., 2014), or suitable distances such as the Hellinger distance (Ditzler and Polikar, 2011). These indirect methods rely on the statistical power of the tests.

In this paper, we consider a supervised learning problem on data streams with the regression function in a RKHS. Our proposed method has several important features. First, by using random feature approximation, the proposed method doesn't need to store all the previous data and uses limited storage space and training time even when the total sample size is enormous. In addition, the variation of our model and the error induced by random feature approximation is reduced by using the data sparsity constraint and a shrinkage parameter. Finally, this method can also handle non-stationary models. In particular, at time t, our approach finds the best model in a RKHS by using the previously estimated model and kernel functions generated by the data we observe at time t. It updates the model by a shrinkage parameter and random feature approximation. Numerical studies in simulated and real data applications also confirm that the proposed method performs competitively for data streams in both stationary and non-stationary problems.

The remainder of this chapter is organized as follows. In Section 4.2, the problem background and the model are introduced. The simulated and real data examples are used to demonstrate the effectiveness of our proposed method in Sections 4.3 and 4.4, respectively.

4.2 Methodology

4.2.1 Problem setup and notation

We consider the supervised learning problem when the observations arrive sequentially. The goal is to recover the underlying mean function. At each time t, we are given a set of n_t instances $\{(x_i^t, y_i^t), i = 1, \ldots, n_t\}$ as our training set, where $t = 1, \ldots T$, n_t is the number of data we receive at time t, T is the total number of times we observe, $\boldsymbol{x}_i^t \in \mathbb{R}^p$ is the p-dimensional covariate vector

of the i-th observation, and $y_i^t \in \mathbb{R}$ is the response of the i-th observation. We consider fitting the model in a RKHS $\mathcal{H} = \{f | f : \mathbb{R}^p \to \mathbb{R}\}$ with a reproducing kernel function $K(\cdot, \cdot)$. The data at time t are observed according to the model

$$Y^t = f_t(\mathbf{X}^t) + \epsilon, \tag{4.1}$$

where $Y^t \in \mathbb{R}, \mathbf{X}^t \in \mathbb{R}^p$, $f_t \in \mathcal{H}$, and $\epsilon \in \mathbb{R}$ is the random noise. While traditional learning algorithms assume the data are sampled from a fixed model, here we assume that the model f_t may vary as a function of time t. Since we want to fit the model on data streams, with possibly an infinite number of observations, it is unrealistic to store all the data. Our goal is to fit our model with limited storage space.

4.2.2 Proposed method

4.2.2.1 Adaptive kernel learning on data streams

First, we describe the general adaptive kernel learning model on data streams. Given the training data $\{(x_i^t, y_i^t), i = 1, \ldots, n_t\}$ at time t, we consider the penalized regression problem which only uses these n_t samples

$$\tilde{f}_t(\boldsymbol{x}) = \arg\min_{f_t \in \mathcal{H}} \frac{1}{n_t} \sum_{i=1}^{n_t} L(f_t(\boldsymbol{x}_i^t), y_i^t) + \lambda J(f_t), \tag{4.2}$$

where L is a convex and differentiable loss function which measures the goodness of fit of f_t, J is a penalty function on f_t in order to avoid overfitting, and λ is a tuning parameter that controls the magnitude of penalty $J(f_t)$. By the representer theorem (Kimeldorf and Wahba, 1971), the estimated function in (4.2) can be written as

$$\tilde{f}_t(\boldsymbol{x}) = \sum_{i=1}^{n_t} \widetilde{\alpha}_{t,i} K\left(\boldsymbol{x}_i^t, \boldsymbol{x}\right), \tag{4.3}$$

where $\tilde{\alpha}_{t,i}$ is the coefficients to be estimated, and we let $\tilde{\boldsymbol{\alpha}}_t = (\tilde{\alpha}_{t,1}, \ldots, \tilde{\alpha}_{t,n_t})^\top$. To learn our model in RKHS, it is common to use the regular squared norm penalty, which aims to solve the following

optimization problem

$$\tilde{f}_t(\boldsymbol{x}) = \arg\min_{f_t \in \mathcal{H}} \frac{1}{n_t} \sum_{i=1}^{n_t} L(f_t(\boldsymbol{x}_i^t), y_i^t) + \lambda \left\| f_t \right\|_{\mathcal{H}}^2 , \tag{4.4}$$

where $\|f_t\|_{\mathcal{H}}$ is the norm of f_t in RKHS $\mathcal{H}$.

The kernel representation of the regression function is similar to the knot structure in the smoothing splines. Each observation in the training data can be regarded as a knot in a multidimensional space. For large sample size problems, the solution to (4.4) is known to be consistent with desirable theoretical properties. However, since the sample size, n_t in each time is usually small in practice, using all kernel functions for the representation may introduce a similar issue as using too many knots in spline regressions. For spline regression, it is known that too many knots may lead to overfitting and unnecessary fluctuation in the resulting estimator. To obtain the estimators with a sparse kernel function representation, Zhang et al. (2016) proposed the data sparsity penalty to constrain the estimated kernel function coefficient vector $\tilde{\boldsymbol{\alpha}}_t$ in an ℓ_1-ball. As shown in Zhang et al. (2016), the data sparsity model is desirable in this case since it can deliver estimators with a sparse kernel function representation. Hence we follow their method and use the data sparsity penalty in our model as well. By (4.2) and (4.3), we aim to solve the following optimization problem with data sparsity constraint

$$\hat{\boldsymbol{\alpha}}_t = \arg\min_{\boldsymbol{\alpha}_t} \left[\frac{1}{n_t} \sum_{i=1}^{n_t} L \left(\sum_{j=1}^{n_t} \alpha_{t,j} K(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t), y_i^t \right) + \lambda \|\boldsymbol{\alpha}_t\|_1 \right] , \tag{4.5}$$

where $\boldsymbol{\alpha}_t = (\alpha_{t,1}, \ldots, \alpha_{t,n_t})^\top$, and $\|\boldsymbol{\alpha}_t\|_1$ refers to the ℓ_1-norm of $\boldsymbol{\alpha}_t$. However, model (4.5) only uses the training data at time t without previous information. For $t > 1$, in order to use both the observations we receive at time t, and the previous models we estimated before time t, we rewrite our estimated function as

$$\hat{f}_t(\boldsymbol{x}) = \gamma_t \hat{f}_{t-1}(\boldsymbol{x}) + \sum_{j=1}^{n_t} \hat{\alpha}_{t,j} K\left(\boldsymbol{x}_j^t, \boldsymbol{x}\right) , \qquad \text{where } \gamma_t \in [0, 1]. \tag{4.6}$$

Here the adaptive weight γ_t illustrates how the model changes from time $t-1$ to time t. If the underlying true model f_t doesn't change, γ_t is expected to be 1 when $\hat{f}_{t-1}$ is a good estimator of f_{t-1}.

In summary, for the training dataset $\{(x_i^t, y_i^t), i = 1, \ldots, n_t\}$ and the model $\hat{f}_{t-1}(\boldsymbol{x})$ estimated at time $t-1$, our adaptive kernel learning model solves the following optimization at time t

$$(\hat{\gamma}_t, \hat{\boldsymbol{\alpha}}_t) = \arg\min_{\gamma_t, \boldsymbol{\alpha}_t} \left[\frac{1}{n_t} \sum_{i=1}^{n_t} L\left(\gamma_t \hat{f}_{t-1}(\boldsymbol{x}_i^t) + \sum_{j=1}^{n_t} \alpha_{t,j} K(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t), y_i^t \right) + \lambda \|\boldsymbol{\alpha}_t\|_1 \right] \quad \text{subject to } \gamma_t \in [0,1]. \tag{4.7}$$

4.2.2.2 Adaptive kernel learning on data streams with adjusted learning rate

When the underlying true model f_t doesn't change from time $t-1$ to time t, the model (4.7) uses $\sum_{j=1}^{n_t} \hat{\alpha}_{t,j} K(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t)$ to fit the residual of our last model $y_i^t - \hat{f}_{t-1}(\boldsymbol{x})$ at time $t-1$. Hence the bias of our model is reduced. However, $\hat{f}_t(\boldsymbol{x})$ is highly correlated to $\hat{f}_{t-1}(\boldsymbol{x})$ due to the sequential modeling process when $\gamma_t = 1$. Compared to the model (4.5) which only uses the data at time t, our model (4.7) has a smaller bias but a relatively larger variation. One advantage of the data sparsity constraint is that it can deliver estimators with a sparse kernel function representation. Hence it is a much simpler model with a relatively small variation.

In order to balance between the bias and variation, we introduce a shrinkage parameter ν. After we solve the optimization problem (4.7), if the solution $\hat{\gamma}_t = 1$, then the estimator $\hat{f}_t(\boldsymbol{x})$ is updated as

$$\hat{f}_t(\boldsymbol{x}) = \hat{f}_{t-1}(\boldsymbol{x}) + \nu \sum_{j=1}^{n_t} \hat{\alpha}_{t,j} K\left(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t\right). \tag{4.8}$$

The shrinkage parameter $0 < \nu \leq 1$ controls the learning rate of our model.

If $\gamma_t \neq 1$, the underlying true model f_t may be changed from time $t-1$ to t and it is not necessary to use the shrinkage parameter. Then the estimator $\hat{f}_t(\boldsymbol{x})$ is still updated as

$$\hat{f}_t(\boldsymbol{x}) = \hat{f}_{t-1}(\boldsymbol{x}) + \sum_{j=1}^{n_t} \hat{\alpha}_{t,j} K\left(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t\right). \tag{4.9}$$

The learning rate parameter ν balances the learning speed and convergence rate tradeoff of our model. With a large ν, our model can estimate f_t well with only a few batches of data but

may converge to a suboptimal model. On the other hand, with a small ν, our model needs more batches of data to estimate f_t well but will converge to a model with better prediction. Hence, in practice, if the number of batches T is small, or if we want to have a good estimation with only a few batches of training data for frequently changing model f_t, it is recommended to use a larger ν such as 1. If the number of batches T is large and the model f_t doesn't change frequently, our model can eventually have a better prediction with a smaller ν such as 0.1.

4.2.2.3 Adaptive kernel learning on data streams with limited storage space

In order to solve the optimization problem (4.7), we need to evaluate $K(\boldsymbol{x}_i^t, \boldsymbol{x}_j^{t'})$ for all the covariates $\boldsymbol{x}_j^{t'}$ we have received until time step $t-1$ to calculate $\hat{f}_{t-1}(\boldsymbol{x}_i)$, where time $t' \leq t-1$. Since the total sample size $n = \sum_{t=1}^{T} n_t$ can be very large, it is impossible for us to store all the data due to the limited storage. Here, we adopt random feature approximation (Lu et al., 2016) to store our model $\hat{f}_t(\boldsymbol{x})$ with limited storage for future use. The key idea is to construct a kernel-induced feature representation $z(\boldsymbol{x})$ such that the inner product of instances in the new feature space can effectively approximate the kernel function as

$$K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) \approx z\left(\boldsymbol{x}_i\right)^\top z\left(\boldsymbol{x}_j\right),$$

where $z(\boldsymbol{x}) \in \mathbb{R}^D$ is a function of $\boldsymbol{x}$, where D is the dimension of the function. A common random feature approximation technique, random Fourier features, can be used in shift-invariant kernels (Rahimi et al., 2007). A shift-invariant kernel is a family of reproducing kernel functions that can be written as $K\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = k(\Delta \boldsymbol{x})$, where k is some function and $\Delta \boldsymbol{x} = \boldsymbol{x}_1 - \boldsymbol{x}_2$ is the difference between two instances. Examples of shift-invariant kernels include some widely used kernels, such as the Gaussian and Laplace kernels. By performing an inverse Fourier transform of the shift-invariant kernel function, one can obtain:

$$K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = k\left(\boldsymbol{x}_i - \boldsymbol{x}_j\right) = \int_{\mathbb{R}^p} p(\mathbf{u}) e^{i\mathbf{u}^\top(\boldsymbol{x}_i - \boldsymbol{x}_j)} d\mathbf{u},$$

where

$$p(\mathbf{u}) = \left(\frac{1}{2\pi}\right)^p \int_{\mathbb{R}^p} e^{-i\mathbf{u}^\top(\Delta \boldsymbol{x})} k(\Delta \boldsymbol{x}) d(\Delta \boldsymbol{x}),$$

which is a proper probability density function calculated from the Fourier transform of function $k(\Delta \boldsymbol{x})$. More specifically, for a Gaussian kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2/(2\sigma^2))$, where σ is the width of the Gaussian kernel, we have the corresponding random Fourier component $\mathbf{u}$ with the distribution $p(\mathbf{u}) = \mathcal{N}\left(0, \sigma^{-2} I\right)$. Then for a continuous, positive-definite and shift-invariant kernel function, according to the Bochner theorem (Rudin, 1962), the kernel function can be expressed as

$$\begin{aligned} K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = & \int_{\mathbb{R}^p} p(\mathbf{u}) e^{i \mathbf{u}^\top (\boldsymbol{x}_i - \boldsymbol{x}_j)} d\mathbf{u} \\ = & \mathbb{E}_{\mathbf{u}}\left[\cos\left(\mathbf{u}^\top \boldsymbol{x}_i\right) \cos\left(\mathbf{u}^\top \boldsymbol{x}_j\right) + \sin\left(\mathbf{u}^\top \boldsymbol{x}_i\right) \sin\left(\mathbf{u}^\top \boldsymbol{x}_j\right)\right] \\ = & \mathbb{E}_{\mathbf{u}}\left[\left[\sin\left(\mathbf{u}^\top \boldsymbol{x}_i\right), \cos\left(\mathbf{u}^\top \boldsymbol{x}_i\right)\right] \cdot \left[\sin\left(\mathbf{u}^\top \boldsymbol{x}_j\right), \cos\left(\mathbf{u}^\top \boldsymbol{x}_j\right)\right]\right], \end{aligned}$$

where the operator $\cdot$ refers to the dot product between two vectors. Then any shift-invariant kernel function can be expressed by the expectation of the inner product between original data's new representation, where the new representation of the data is $z(\boldsymbol{x}) = \left[\sin\left(\mathbf{u}^\top \boldsymbol{x}\right), \cos\left(\mathbf{u}^\top \boldsymbol{x}\right)\right]^\top$. We can sample $D \in \mathbb{N}$ number of random Fourier components $\mathbf{u}_1, \ldots \mathbf{u}_D$ independently for constructing the new representation as

$$z(\boldsymbol{x}) = \left(\sin\left(\mathbf{u}_1^\top \boldsymbol{x}\right), \cos\left(\mathbf{u}_1^\top \boldsymbol{x}\right), \ldots, \sin\left(\mathbf{u}_D^\top \boldsymbol{x}\right), \cos\left(\mathbf{u}_D^\top \boldsymbol{x}\right)\right)^\top.$$

The kernel learning task in the original input space can be approximated by solving a linear learning task in the new feature space.

Using the above approximation, when $t = 1$, the model $\hat{f}_1(\boldsymbol{x}) = \sum_{i=1}^{n_1} \hat{\alpha}_{1,i} K(\boldsymbol{x}_i^1, \boldsymbol{x})$ can be rewritten as

$$\hat{f}_1(\boldsymbol{x}) = \sum_{i=1}^{n_1} \hat{\alpha}_{1,i} K\left(\boldsymbol{x}_i^1, \boldsymbol{x}\right) \approx \sum_{i=1}^{n_1} \hat{\alpha}_{1,i} z\left(\boldsymbol{x}_i^1\right)^\top z(\boldsymbol{x}) = \hat{\mathbf{u}}_1^\top z(\boldsymbol{x}),$$

where $\hat{\mathbf{u}}_1 = \sum_{i=1}^{n_1} \hat{\alpha}_{1,i} z(\boldsymbol{x}_i^1)$. Let $\mu_t = \nu$ when $\gamma_t = 1$ and $\mu_t = 1$ when $\gamma_t \neq 1$. Similarly, the model $\hat{f}_2(\boldsymbol{x}) = \gamma_2 \hat{f}_1(\boldsymbol{x}) + \mu_2 \sum_{i=1}^{n_2} \hat{\alpha}_{2,i} K(\boldsymbol{x}_i^2, \boldsymbol{x})$ can be rewritten as

$$\hat{f}_2(\boldsymbol{x}) \approx \gamma_2 \hat{\mathbf{u}}_1^\top z(\boldsymbol{x}) + \mu_2 \sum_{i=1}^{n_2} \hat{\alpha}_{2,i} z\left(\boldsymbol{x}_i^2\right)^\top z(\boldsymbol{x}) = \hat{\mathbf{u}}_2^\top z(\boldsymbol{x}),$$

where $\hat{\mathbf{u}}_2 = \gamma_2 \hat{\boldsymbol{u}}_1 + \mu_2 \sum_{i=1}^{n_2} \hat{\alpha}_{2,i} z(\boldsymbol{x}_i^2)$. By induction, when $t > 1$, for given model at time step $t-1$ as $\hat{f}_{t-1}(\boldsymbol{x}) = \hat{\mathbf{u}}_{t-1}^\top z(\boldsymbol{x})$, our estimated function $\hat{f}_t(\boldsymbol{x}) = \gamma_t \hat{f}_t(\boldsymbol{x}) + \mu_t \sum_{i=1}^{n_t} \hat{\alpha}_{t,i} K(\boldsymbol{x}_i^t, \boldsymbol{x})$ can be written as

$$\hat{f}_t(\boldsymbol{x}) \approx \gamma_t \hat{\boldsymbol{u}}_{t-1} z(\boldsymbol{u}) + \mu_t \sum_{i=1}^{n_t} \tilde{\alpha}_{t,i} z\left(\boldsymbol{x}_i^t\right)^\top z(\boldsymbol{x}) = \hat{\boldsymbol{u}}_t^\top z(\boldsymbol{x}),$$

where $\hat{\mathbf{u}}_t = \gamma_t \hat{\boldsymbol{u}}_{t-1} + \mu_t \sum_{i=1}^{n_t} \hat{\alpha}_{t,i} z(\boldsymbol{x}_i^t)$. Then the optimization problem (4.7) can be written as

$$(\hat{\gamma}_t, \hat{\boldsymbol{\alpha}}_t) = \arg\min_{\gamma_t, \boldsymbol{\alpha}_t} \left[\frac{1}{n_t} \sum_{i=1}^{n_t} L\left(\gamma_t \hat{\mathbf{u}}_{t-1}^\top z(\boldsymbol{x}_i^t) + \sum_{j=1}^{n_t} \alpha_{t,j} z\left(\boldsymbol{x}_j^t\right)^\top z(\boldsymbol{x}_i^t), y_i^t \right) + \lambda \|\boldsymbol{\alpha}\|_1 \right] \text{ subject to } \gamma_t \in [0,1], \tag{4.10}$$

where $\hat{\mathbf{u}}_{t-1}$ is the coefficient vector of the previous model we estimated at time $t-1$. If $\hat{\gamma}_t = 1$, then the estimator $\hat{\mathbf{u}}_t$ is updated as

$$\hat{\mathbf{u}}_t = \gamma_t \hat{\mathbf{u}}_{t-1} + \nu \sum_{i=1}^{n_t} \hat{\alpha}_{t,i} z(\boldsymbol{x}_i^t).$$

If $\gamma_t \neq 1$, the estimator $\hat{\mathbf{u}}_t$ is updated as

$$\hat{\mathbf{u}}_t = \gamma_t \hat{\mathbf{u}}_{t-1} + \sum_{i=1}^{n_t} \hat{\alpha}_{t,i} z(\boldsymbol{x}_i^t).$$

Then instead of keeping all the data to evaluate kernel functions at each time step, we need to keep a D-dimensional vector $\hat{\mathbf{u}}_t$.

Data sparsity constraint can also reduce the approximation error induced by the random feature approximation. When we use the random feature approximation, the more kernel functions we use, the larger approximation error it may generate. As pointed out by Sun et al. (2018), the number of random features D needed for a consistent estimation grows when the number of kernel functions increases. Thus when we use fewer kernel functions to estimate the model by using data sparsity constraint, we can also reduce the error induced by the random feature approximation.

Algorithm 5 below describes the major steps of the Incremental Adaptive Data Sparsity Kernel (IADSK) learning method for a given shift-invariant kernel function $K(\boldsymbol{x}_1, \boldsymbol{x}_2) = k(\Delta \boldsymbol{x})$, the number of random Fourier components D, and the learning rate ν, loss function L.

Algorithm 5: incremental Adaptive Data Sparsity Kernel learning (IADSK) method

Input: data stream $\{(\boldsymbol{x}_i^t, y_i^t), i = 1, \ldots, n_t\}_{t=1}^T$, a shirft-invariant kernel function $K(\boldsymbol{x}_1, \boldsymbol{x}_2) = k(\Delta \boldsymbol{x})$, the number of random Fourier components D and the learning rate ν, loss function L.

Output: $\{\hat{f}_t\}_{t=1}^T$

Generate D random Fourier components $\mathbf{u}_1, \ldots \mathbf{u}_D$ independently with the distribution

$$p(\mathbf{u}) = \left(\frac{1}{2\pi}\right)^d \int e^{-i\mathbf{u}^\top(\Delta \boldsymbol{x})} k(\Delta \boldsymbol{x}) d(\Delta \boldsymbol{x}).$$

Constructing the random feature function as

$$z(\boldsymbol{x}) = \left(\sin\left(\mathbf{u}_1^\top \boldsymbol{x}\right), \cos\left(\mathbf{u}_1^\top \boldsymbol{x}\right), \ldots, \sin\left(\mathbf{u}_D^\top \boldsymbol{x}\right), \cos\left(\mathbf{u}_D^\top \boldsymbol{x}\right)\right)^\top.$$

Let $\hat{\mathbf{u}}_0 = 0$. **for** $t = 1$ *to* T **do**

Compute γ_t and $\hat{\boldsymbol{\alpha}}_t$ by

$$\min_{\gamma_t, \hat{\boldsymbol{\alpha}}_t} \left[\frac{1}{n_t} \sum_{i=1}^{n_t} L\left(\gamma_t \hat{\mathbf{u}}_{t-1}^\top z(\boldsymbol{x}_i^t) + \sum_{j=1}^{n_t} \hat{\alpha}_{t,j} z\left(\boldsymbol{x}_j^t\right)^\top z(\boldsymbol{x}_i^t), y_i^t\right) + \lambda_t \|\hat{\boldsymbol{\alpha}}_t\|_1\right] \qquad \text{subject to } \gamma_t \in [0, 1],$$

where tuning parameter λ_t is chosen by cross validation

Compute $\hat{\boldsymbol{u}}_t$ by

$$\hat{\mathbf{u}}_t = \begin{cases} \gamma_t \hat{\mathbf{u}}_{t-1} + \sum_{i=1}^{n_t} \hat{\alpha}_{t,i} z\left(\boldsymbol{x}_i^t\right) & \text{if } \gamma_t < 1; \\ \gamma_t \hat{\mathbf{u}}_{t-1} + \nu \sum_{i=1}^{n_t} \hat{\alpha}_{t,i} z\left(\boldsymbol{x}_i^t\right) & \text{if } \gamma_t = 1. \end{cases}$$

Compute $\hat{f}_t$ by

$$\hat{f}_t(\boldsymbol{x}) = \hat{\mathbf{u}}_t^\top z(\boldsymbol{x}).$$

end

4.3 Numerical study

In this section, we perform three numerical studies to compare the efficiency of our proposed method (IADSK) with different learning rates ν and two other methods. In particular, we choose $\nu = 1, 0.5$ and 0.3 for IADSK. The other two methods include

- Fourier online gradient descent (FouGD) method (Lu et al., 2016), which is an online kernel learning method using random Fourier features for approximating kernel functions.
- Incremental Adaptive Ridge Kernel (IARK) learning method with different learning rate ν, which uses the squared norm penalty $\|f\|_{\mathcal{H}}^2$ instead of the data sparsity penalty, where $\|f\|_{\mathcal{H}}$ is the norm of f in RKHS $\mathcal{H}$. In particular, we also choose learning rate $\nu = 1, 0.5$ and 0.3 for our proposed method.

In our numerical study, we use the Gaussian kernel and the ℓ_2-loss as our loss function in our training model (4.10). Then $L(\hat{f}(\boldsymbol{x}), y) = (y - \hat{f}(\boldsymbol{x}))^2$ and $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2/(2\sigma^2))$. Let the number of random features D be 30. For the first two examples, we aim to compare our method with other methods when the model is stationary. For the first example, we generate the data by

$$Y_t = 3\exp((X_t - 0.5)^2) + \exp((X_t - 1)^2) + \epsilon.$$

For the second example, we generate the data by

$$Y_t = 10\exp(X_t^2) + \epsilon.$$

Let $\epsilon \sim N(0, 0.1)$ and $X_t \in \mathbb{R}$ follow a uniform distribution within $[-1, 1]$. In both examples, we let the size of each batch of our training samples be 10, 20, or 40, and generate 3000 batches of training samples in total.

For the third example, we aim to compare our method with other methods when the model is non-stationary. We generate the data by

$$Y_t = 3\exp((X_t - 0.5)^2) + \exp((X_t - 1)^2) + \epsilon,$$

when $t \in [1, 500]$, and

$$Y_t = 3\exp((X_t + 0.5)^2) + \exp((X_t + 1)^2) + \epsilon,$$

when $t \in [501, 1000]$. Let $\epsilon \sim N(0, 0.1)$ and $X_t \in \mathbb{R}$ follows a uniform distribution within $[-1, 1]$. We let the size of the each batch of our training samples be 20, and generate 1000 batches of training samples in total.

For each example, we repeat the simulation 50 times. To evaluate the prediction performance of the algorithms at time t, we generate 100 testing samples $\{(X^t_{i,\text{text}}, Y^t_{i,\text{test}}), i = 1, \ldots, 100\}$. Then we use the average testing error from time 1 until time t as the criterion

$$\frac{1}{t}\sum_{i=1}^{t}\frac{1}{100}\sum_{j=1}^{100}(Y^i_{j,\text{test}} - \hat{Y}^i_{j,\text{test}})^2,$$

where $\hat{Y}^t_{j,\text{test}} = \hat{f}_t(X^t_{j,\text{text}})$ is the prediction using our estimated model $\hat{f}_t$ at time t. In addition, after we plot the performance of all methods, we zoom in some parts of the plot to highlight the comparison of different methods. In particular for the first and second examples, we first plot the performance of all te methods for all batches. Secondly, we plot the performance of IADSK with 3 different learning rates for the first 50 batches. Then we plot the performance of FouGD and IADSK with 3 different learning rates for the last 1000 batches. Finally we plot the performance of IADSK with 3 different learning rates for the last 1000 batches.

For the third example, we first plot the performance of IADSK with 3 different learning rates, IARK with 3 different learning rates, and FouGD for all batches. Then the performance of FouGD and IADSK with 3 different learning rates for time $t \in [400, 600]$. Finally, we plot the performance of IADSK with 3 different learning rates for time $t \in [400, 600]$.

We report the simulation results in Figures 4.1 to 4.7. Figures 4.1, 4.2, 4.3 and 4.4 show the results of Example 1 with 10, 20 and 40 training samples in each batch respectively. Figures 4.4, 4.5, 4.6 and 4.4 show the results of Example 2 with 10, 20 and 40 training samples in each batch respectively. Figure 4.7 shows the result of Example 3.

Compared with the other two methods, our proposed IADSK method always delivers better prediction than FouGD and IARK. Specifically, the average testing error of IADSK with $\nu = 1$

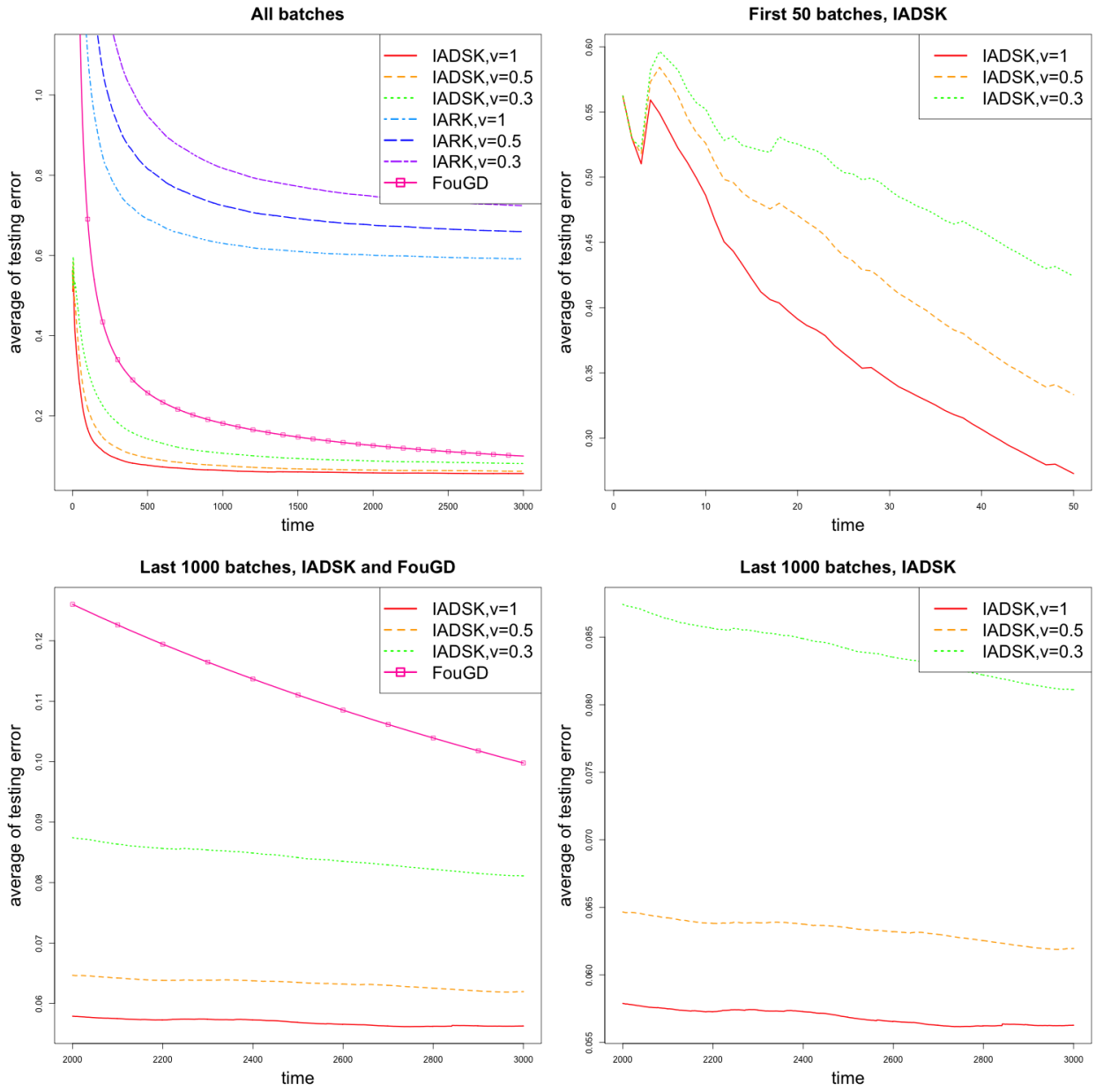

Figure 4.1: Performance comparison of different methods for Example 1 with 10 samples in each batch. The top left figure compare the performance of all methods for all 3000 batches of data. The top right figure compare the performance of IADSK with different learning rate for the first 50 batches of data. The bottom left figure compare the performance of FouGD and IADSK with different learning rate for the last 1000 batches of data. The bottom right figure compare the performance of IADSK with different learning rate for the last 1000 batches of data.

and 0.8 decrease much faster than the other methods when $t \leq 50$, and IADSK with $\nu = 1$ and 0.8 perform better when t is small.

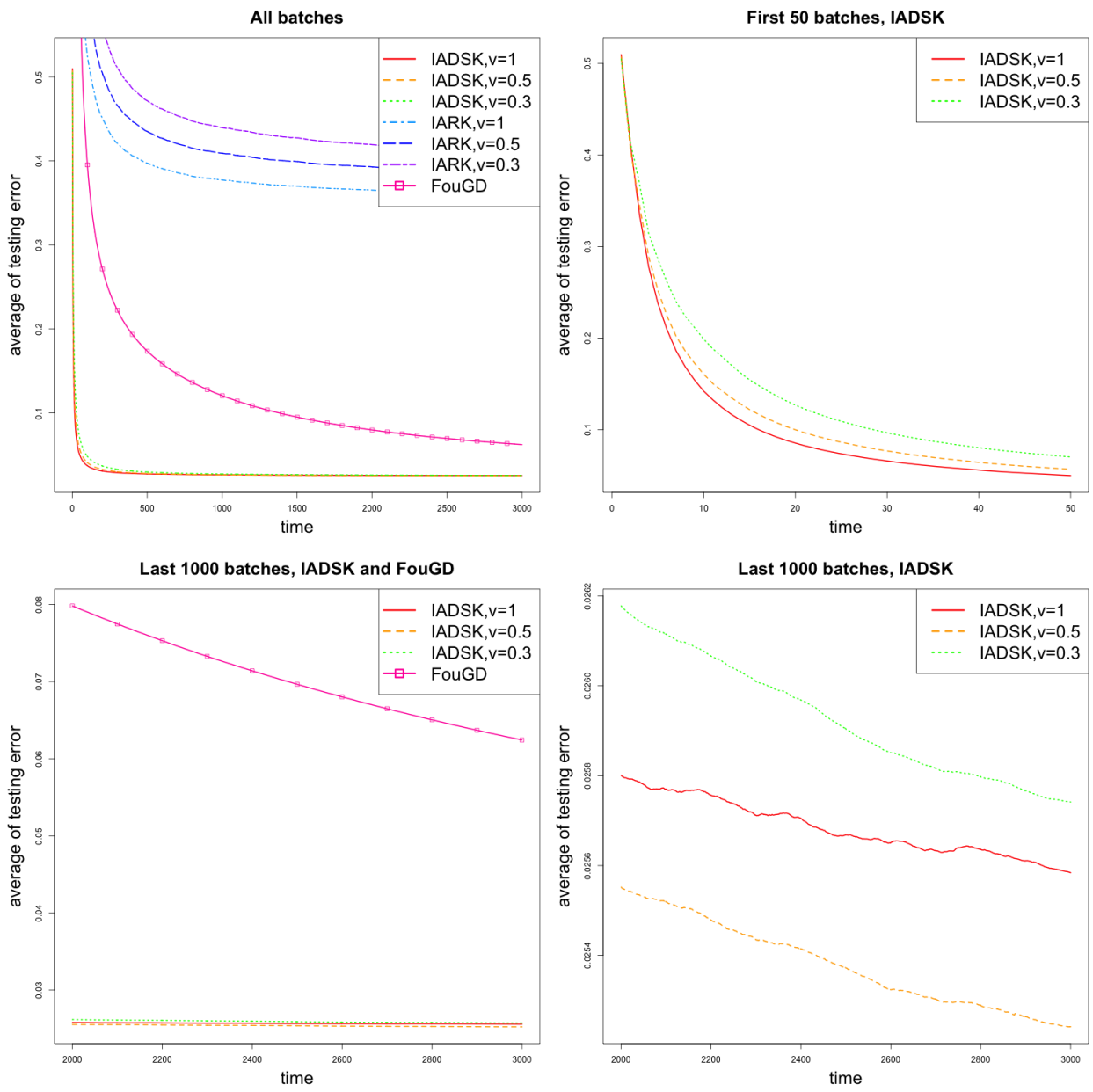

Figure 4.2: Performance comparison of different methods for Example 1 with 20 samples in each batch. The top left figure compare the performance of all methods for all 3000 batches of data. The top right figure compare the performance of IADSK with different learning rate for the first 50 batches of data. The bottom left figure compare the performance of FouGD and IADSK with different learning rate for the last 1000 batches of data. The bottom right figure compare the performance of IADSK with different learning rate for the last 1000 batches of data.

For the first example when $n_t = 10$, Figure 4.1 shows that the IADSK with $\nu = 1$ or 0.8 always produces smaller average testing errors than the other methods. But as time t becomes larger, the testing error of FouGD decreases faster than that of IADSK and IARK.

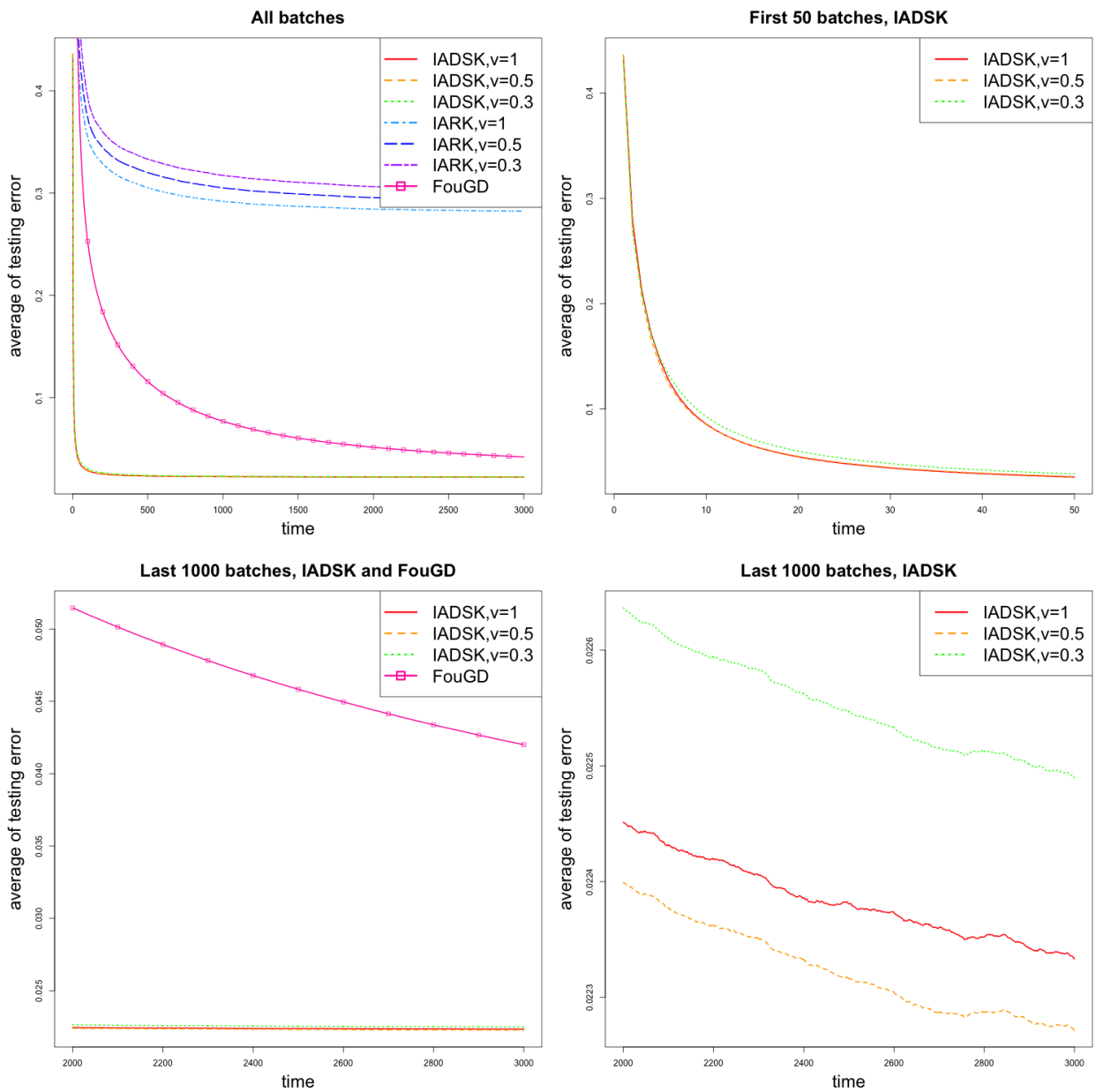

Figure 4.3: Performance comparison of different methods for Example 1 with 40 samples in each batch. The top left figure compare the performance of all methods for all 3000 batches of data. The top right figure compare the performance of IADSK with different learning rate for the first 50 batches of data. The bottom left figure compare the performance of FouGD and IADSK with different learning rate for the last 1000 batches of data. The bottom right figure compare the performance of IADSK with different learning rate for the last 1000 batches of data.

For the first example when $n_t = 20$ or 40 and the second example when $n_t = 10, 20$ or 40, Figures 4.2, 4.3, 4.4, 4.5 and 4.6 show that although the average testing error of IADSK with $\nu = 1$ is smaller than all the other methods when $t \leq 50$, when $t > 2000$, the testing error of IADSK with $\nu = 0.3$ or 0.5 become smaller than IADSK with $\nu = 1$.

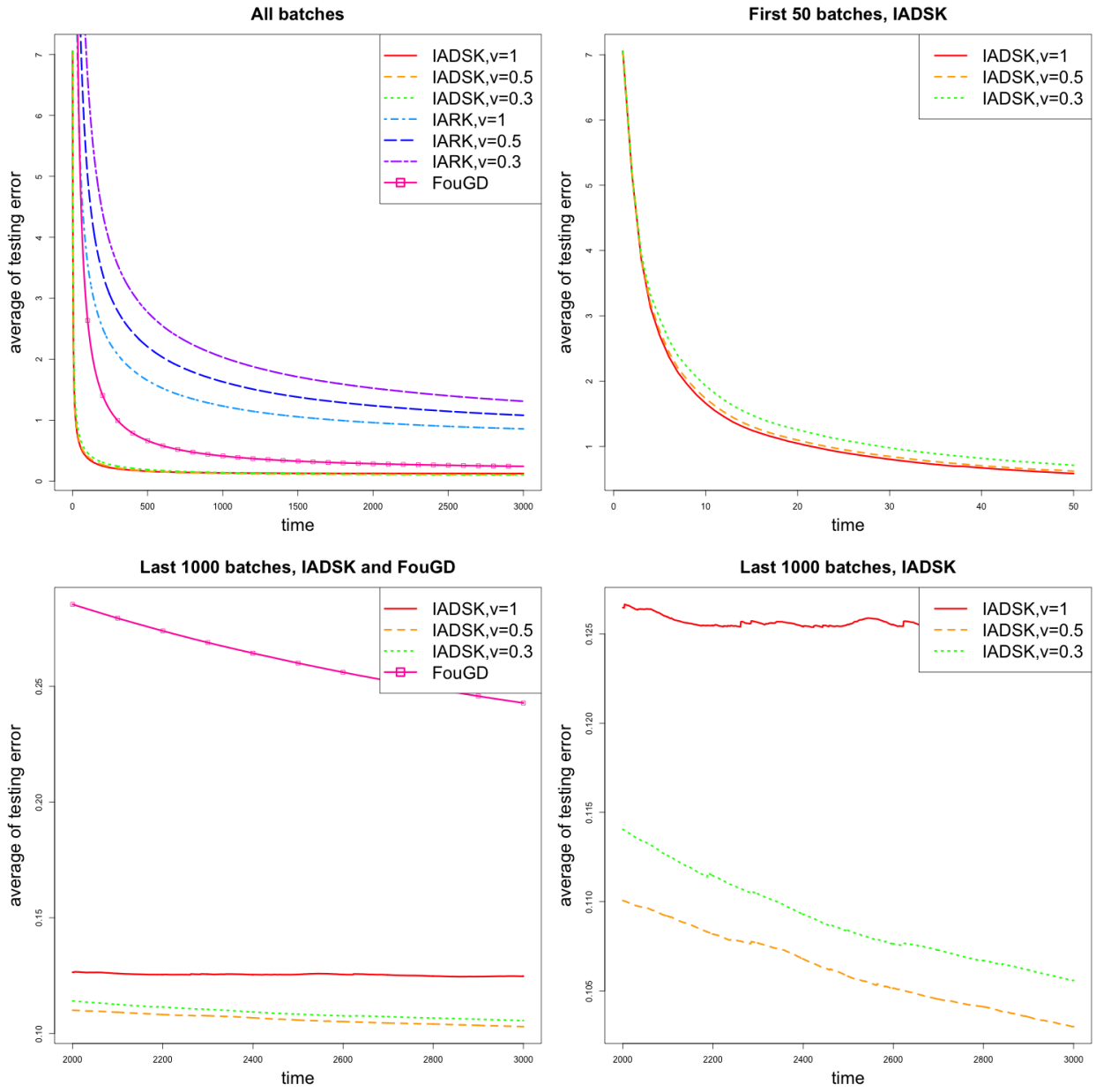

Figure 4.4: Performance comparison of different methods for Example 2 with 10 samples in each batch. The top left figure compare the performance of all methods for all 3000 batches of data. The top right figure compare the performance of IADSK with different learning rate for the first 50 batches of data. The bottom left figure compare the performance of FouGD and IADSK with different learning rate for the last 1000 batches of data. The bottom right figure compare the performance of IADSK with different learning rate for the last 1000 batches of data.

For the third example, the lower right plot in Figure 4.7 shows that the model change at time $t = 500$ has more impact on the performance of FouGD than our proposed method. In addition, the lower-left plot shows that the average testing error of IADSK with $\nu = 1$ or 0.8 decrease faster than IADSK with $\nu = 0.5$ or 0.3 after the model changes.

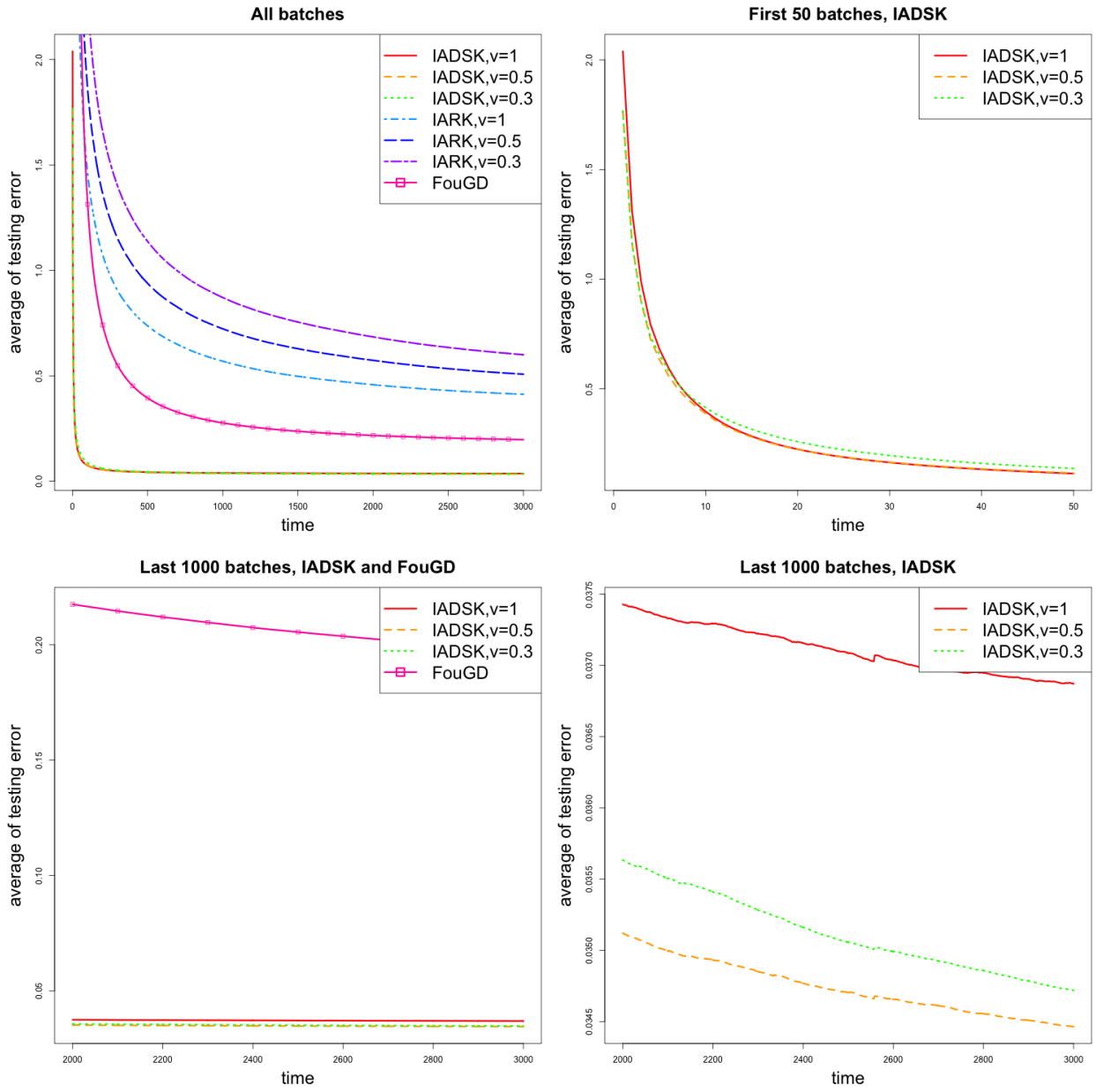

Figure 4.5: Performance comparison of different methods for Example 2 with 20 samples in each batch. The top left figure compare the performance of all methods for all 3000 batches of data. The top right figure compare the performance of IADSK with different learning rate for the first 50 batches of data. The bottom left figure compare the performance of FouGD and IADSK with different learning rate for the last 1000 batches of data. The bottom right figure compare the performance of IADSK with different learning rate for the last 1000 batches of data.

4.4 Experiments on real data

In this section, we demonstrate the performance of our proposed model using the abalone dataset from UCI datasets. Abalone is a mollusk with a peculiar ear-shaped shell lined with

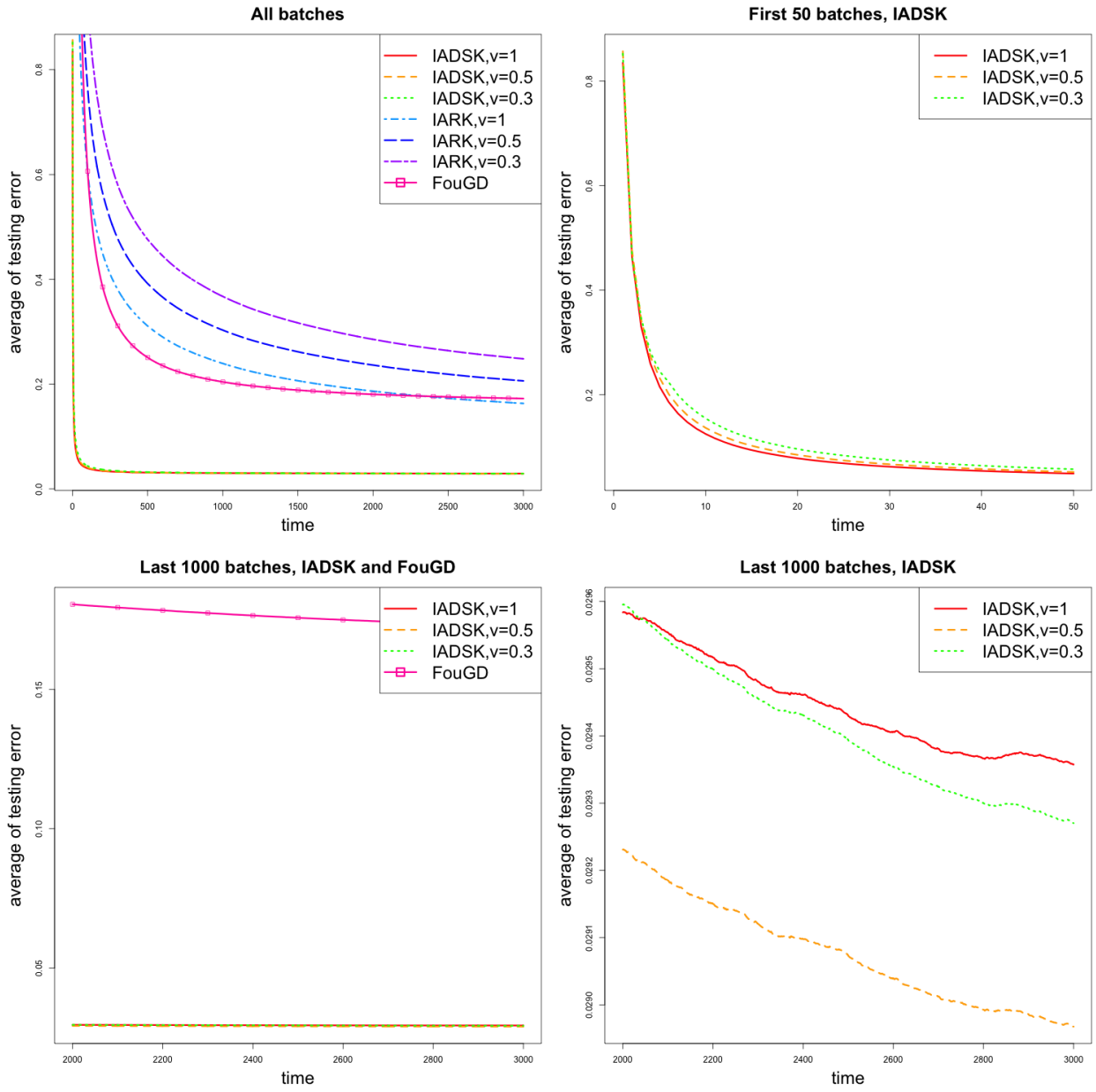

Figure 4.6: Performance comparison of different methods for Example 2 with 40 samples in each batch. The top left figure compare the performance of all methods for all 3000 batches of data. The top right figure compare the performance of IADSK with different learning rate for the first 50 batches of data. The bottom left figure compare the performance of FouGD and IADSK with different learning rate for the last 1000 batches of data. The bottom right figure compare the performance of IADSK with different learning rate for the last 1000 batches of data.

mother of pearl. Researchers can estimate its age by counting the number of rings in its shell with a microscope, but it is time-consuming. In this section, we will use different methods to predict the age by using physical measurements. The sample data contains 4177 observations and eight features. All of the predictor variables are continuous except for sex, which is a categorical variable

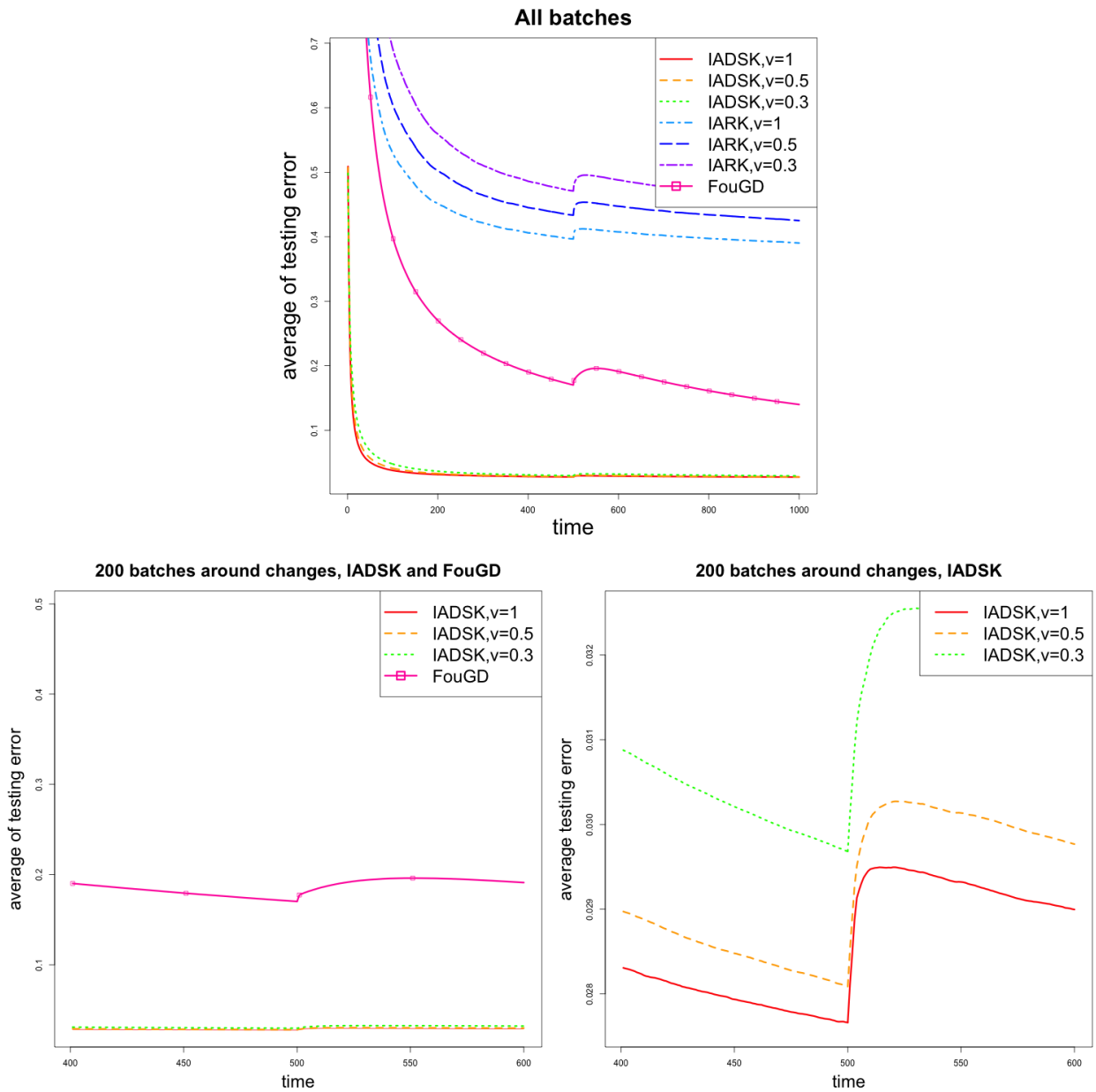

Figure 4.7: Performance comparison of different methods for Example 3 with 20 samples in each batch. The top figure compare the performance of all methods for all 3000 batches of data. The bottom left figure compare the performance of FouGD and IADSK with different learning rate for time $t \in [400, 600]$. The bottom right figure compare the performance of IADSK with different learning rate for time $t \in [400, 600]$.

with possible values 'M' (for males), 'F' (for females), and 'I' (for infants). The goal is to predict the number of rings on the abalone and thereby determine its age.

In our experiments, we divide the data into the training set and testing set. The testing set consists of 57 randomly selected subjects, and the training set consists of T batches of samples, and each batch contains n_t subjects, where T and n_t are specified in each experiment. The analyses

were repeated 30 times for each method using different data partitions. We use the Gaussian kernel and the ℓ_2-loss in our training model (4.10). To evaluate the result, we use the average testing error from time 1 until time t as the criterion

$$\frac{1}{t}\sum_{i=1}^{t}\frac{1}{100}\sum_{j=1}^{100}(Y^i_{j,\text{test}} - \hat{Y}^i_{j,\text{test}})^2.$$

In the first experiment, we let $n_t = 20$ and $T = 206$. In the second experiment, we let $n_t = 40$ and $T = 103$. The results are plotted in Figure 4.8. Both results indicate that our proposed method IADSK with $\nu = 1$ and $\nu = 0.8$ deliver the best prediction among all methods. In addition, although the total numbers of training samples are the same in both experiments, the performance of both IADSK and IARK in the second experiment is better than those in the first experiment.

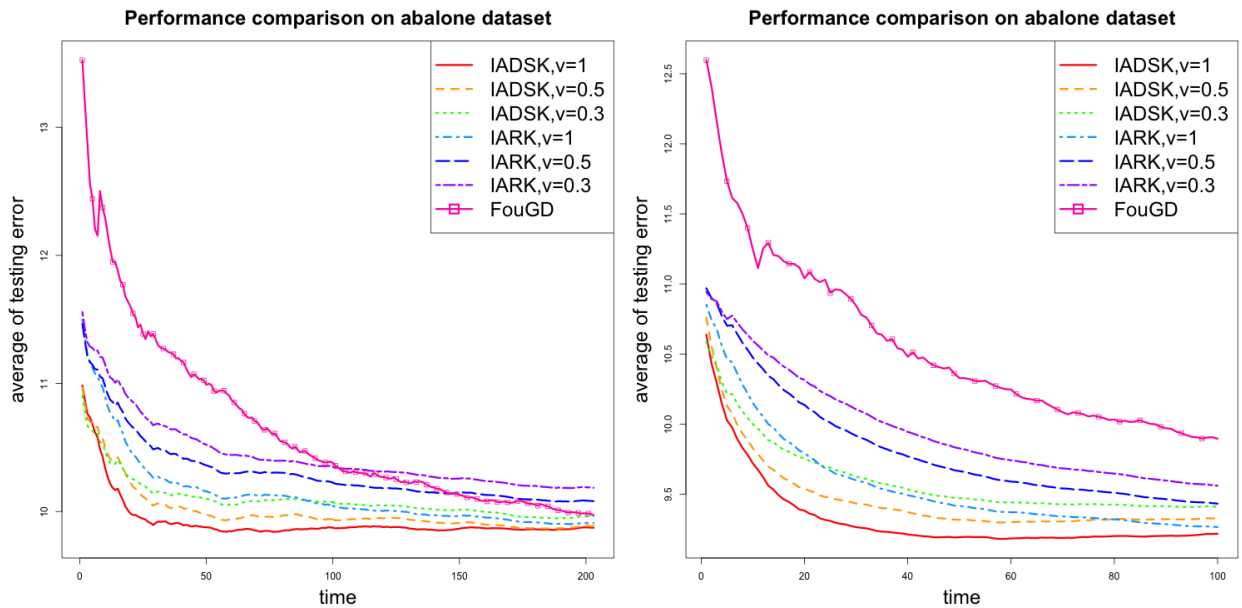

Figure 4.8: Performance comparison of different methods for Abalone data. The left figure compare the performance of all methods for Abalone data with 20 samples in each batch. The right figure compare the performance of all methods for Abalone data with 40 samples in each batch.

APPENDIX A: SUPPLEMENTS TO CHAPTER 2

A.1 Toy example with adaptive LASSO penalty

The advantage of joint estimation is not pertained to the choice of penalty. If we choose other penalty functions, we can still see such an advantage. To illustrate that, we did some further simulation experiments in the toy example with the adaptive LASSO penalty. In particular, we plot the estimation errors of the original adaptive LASSO method ("Separate LASSO" in Figure 1) and the adaptive LASSO penalty with precision matrix as the adjusted weight ("2-step weighted LASSO" in Figure 1). We use cross-validation to choose the tuning parameter. The resulting estimation errors are shown in Figure A.1. It shows that the two-step weighted adaptive LASSO may perform worse than separate adaptive LASSO, so it also has the same problem as LASSO. Jointly estimate $\mathbf{B}^*$ and $\mathbf{C}^*$ with the adaptive LASSO penalty can solve this problem.

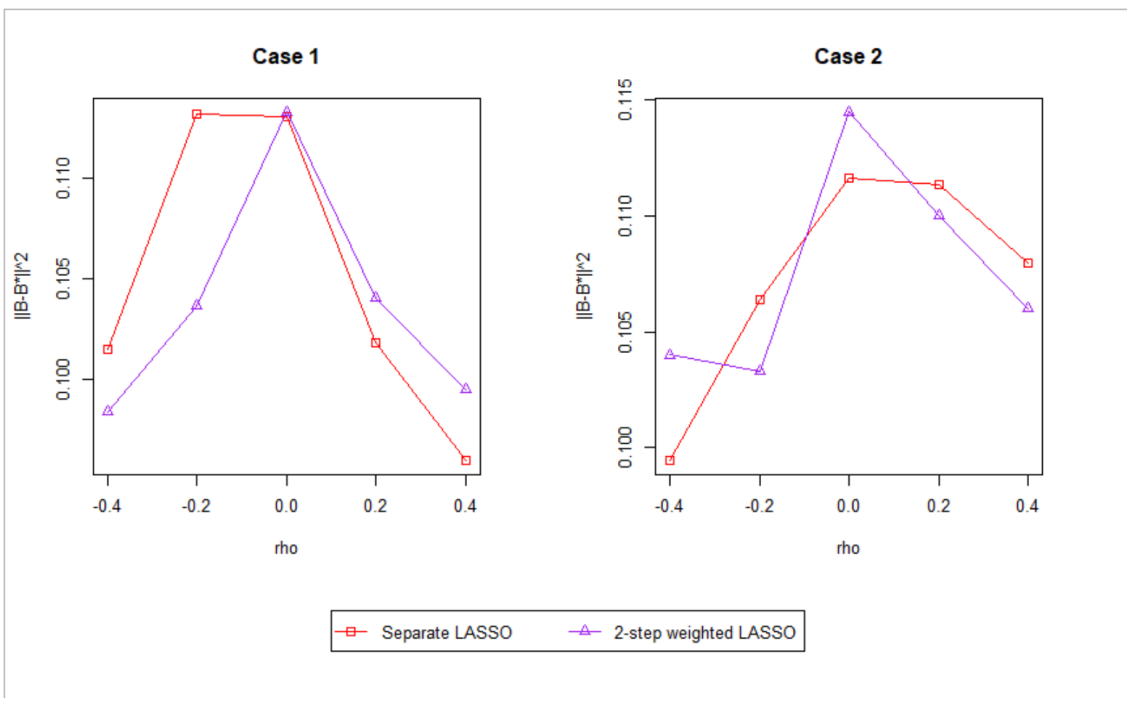

Figure A.1: Plots of the estimation errors for separated adaptive LASSO, two-step weighted adaptive LASSO and joint estimation when $\mathbf{\Sigma}_\epsilon = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$. The left panel is for $\mathbf{B}^* = \begin{pmatrix} 0 & 0 \\ 2 & 3.5 \end{pmatrix}$ and the right panel is for $\mathbf{B}^* = \begin{pmatrix} 0 & 0 \\ -2 & 3.5 \end{pmatrix}$.

A.2 Regularity Conditions

Assumption A.2.1. *Suppose there exists two positive constants L_1 and L_2 such that for any $\mathbf{u}_1 \in \mathbb{R}^p$, $\mathbf{u}_2 \in \mathbb{R}^q$, and $t \in \mathbb{R}$, $\mathbb{E}\left(\exp\left(t\mathbf{u}_1^\top \mathbf{x}_i\right)\right) \leq \exp\left(\frac{L_1^2\|\mathbf{u}_1\|_2^2 t^2}{2}\right)$ and $\mathbb{E}(\exp(t\mathbf{u}_2^\top \mathbf{y}_i)) \leq \exp\left(\frac{L_2^2\|\mathbf{u}_2\|_2^2 t^2}{2}\right)$.*

Assumption A.2.2. *$n_{XX} \geq 6\log p$, $n_{XY} \geq 4\log(pq)$ and $n_{YY} \geq 6\log q$.*

Under Condition A.2.1, the predictor and the response vectors follow sub-Gaussian distributions. Condition A.2.2 ensures that the missing proportion of the data is not too large in order to get consistent estimators of $\mathbf{B}^*$ and $\mathbf{C}^*$. If we further assume that $(\log(pq))/n_0 = O(1)$, with $n_0 = \min\{n_{XX}, n_{XY}, n_{YY}\}$, Condition A.2.2 is satisfied when n_0 is sufficiently large.

In order to prove Lemma 2.3.1 and 2.3.2, we need the following additional assumptions.

Assumption A.2.3. *$\|\mathbf{B}^*\|_{L_1} \leq c_0^{\gamma_1}$ and $\|\mathbf{C}^*\|_{L_1} \leq c_0^{\gamma_2}$ where $0 < \gamma_1, \gamma_2 < \frac{1}{16}$ and $c_0 = \min\{\frac{n_{XY}}{\log(pq)}, \frac{n_{XX}}{\log p}, \frac{n_{YY}}{\log q}\}$. $\|\mathbf{B}^*\|_2 \leq c$ for some positive constant c.*

Assumption A.2.4. *Suppose that $\mathbf{\Sigma}_{XX}$ and $\mathbf{C}^*$ satisfy $c \leq \lambda_{\min}(\mathbf{\Sigma}_{XX}) \leq \lambda_{\max}(\mathbf{\Sigma}_{XX}) \leq C$ and $c \leq \lambda_{\min}(\mathbf{C}^*) \leq \lambda_{\max}(\mathbf{C}^*) \leq C$ for some positive constants c and C.*

Condition A.2.3 makes a weak assumption on the upper bounds of the norms of the true parameters, where the two upper bounds can diverge as $(\log(pq))/n_0 \to 0$, with $n_0 = \min\{n_{XX}, n_{XY}, n_{YY}\}$. We impose the sub-Gaussian assumption on y_i in Condition A.2.1. We essentially assume that it has bounded variance. Since it is the response from a linear model, it is reasonable to assume that $\mathrm{Var}(y_i)$ is bounded. Since $\mathrm{Var}(y_i) \geq \mathbf{B}^{*\top}\mathrm{Var}(x_i)\mathbf{B}^*$, the boundness of $\mathrm{Var}(y_i)$ implies that $\|\mathbf{B}^*\|_2$ is bounded, if we assume $\lambda_{\max}(\mathrm{Var}(x_i)) < \infty$. Condition A.2.4 ensures that the eigenvalues of $\mathbf{\Sigma}_{XX}$ and $\mathbf{C}^*$ are bounded away from 0 and infinity.

Assumption A.2.5. *$\left\|(\mathbf{C}^* \otimes \mathbf{\Sigma}_{XX})_{S_B^C S_B}(\mathbf{C}^* \otimes \mathbf{\Sigma}_{XX})^{-1}_{S_B S_B}\right\|_\infty \leq 1 - \eta$ holds for a constant $\eta \in (0, 1)$.*

Condition A.2.5 can be viewed as a population version of the strong irrepresentable condition proposed in Zhao and Yu (2006).

A.3 Proof of Proposition 2.1

We use a similar argument as the proof of Proposition 1 in Yu et al. (2020), we first decompose the objective function into the estimation error of intra-modality sample covariance matrix, the estimation error of diagonal entries and the estimation error of cross-modality sample covariance matrix. Then we find the optimal value of each term.

By using the facts that $\mathbf{\Sigma}_{XX} = \mathbf{\Sigma}_I + \mathbf{\Sigma}_C$ and $\mathbb{E}(\tilde{\mathbf{\Sigma}}_I) = \mathbf{\Sigma}_I$, we can rewrite the objective function in (2.10) as

$$\begin{aligned}
&\arg\min_{\alpha_1,\alpha_2} \mathbb{E}\|\hat{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\|_F^2 \\
&= \arg\min_{\alpha_1,\alpha_2} \left\{ \alpha_1^2 \mathbb{E}\left\|\tilde{\mathbf{\Sigma}}_I - \mathbf{\Sigma}_I\right\|_F^2 + (1-\alpha_1)^2\, \mathbb{E}\left\|\mathrm{diag}(\tilde{\mathbf{\Sigma}}_I) - \mathbf{\Sigma}_I\right\|_F^2 + \mathbb{E}\left\|\alpha_2 \tilde{\mathbf{\Sigma}}_C - \mathbf{\Sigma}_C\right\|_F^2 \right\}.
\end{aligned}$$

The optimal value of α_2 can be obtained by minimizing $\mathbb{E}\|\alpha_2\tilde{\mathbf{\Sigma}}_C - \mathbf{\Sigma}_C\|_F^2$. Thus, the optimal value is $\alpha_2^* = \frac{\|\mathbf{\Sigma}_C\|_F^2}{\|\mathbf{\Sigma}_C\|_F^2 + \delta_C^2}$. Then taking the derivative of the objective function with respect to α_1, we can find that the optimal value of α_1 is $\alpha_1^* = \frac{\theta^2}{\theta^2+\delta_I^2}$.

At the optimum, the value of the objective function is equal to $\frac{\delta_I^2\theta^2}{\delta_I^2+\theta^2} + \frac{\delta_C^2\|\mathbf{\Sigma}_C\|_F^2}{\delta_C^2+\|\mathbf{\Sigma}_C\|_F^2} \leq \delta_I^2 + \delta_C^2$. Since $\mathbb{E}\|\tilde{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\|_F^2 = \delta_I^2 + \delta_C^2$, we have $\mathbb{E}\|\hat{\mathbf{\Sigma}}_{XX}^* - \mathbf{\Sigma}_{XX}\|_F^2 \leq \mathbb{E}\|\tilde{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\|_F^2$.

By taking the derivative of the objective function of (2.11) with respect to α_3, the optimal value of α_3 is $\alpha_3^* = \frac{\|\mathbf{\Sigma}_{XY}\|_F^2}{\|\mathbf{\Sigma}_{XY}\|_F^2 + \delta_{XY}^2}$. At the optimum, the value of the objective function is equal to $\frac{\delta_{XY}^2\|\mathbf{\Sigma}_{XY}\|_F^2}{\delta_{XY}^2+\|\mathbf{\Sigma}_{XY}\|_F^2}$, which is less than δ_{XY}^2. Since $\mathbb{E}\|\tilde{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\|_F^2 = \delta_{XY}^2$, we have $\mathbb{E}\|\hat{\mathbf{\Sigma}}_{XY}^* - \mathbf{\Sigma}_{XY}\|_F^2 \leq \mathbb{E}\|\tilde{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\|_F^2$.

A.4 Proof of Theorem 2.3.1

We first gives the large deviation bounds for the sample covariance matrices $\tilde{\mathbf{\Sigma}}_{XX}$ and $\tilde{\mathbf{\Sigma}}_{XY}$ by a similar argument as the proof of Lemma 1 in Yu et al. (2020). Then we calculate the convergence rate of entries in the estimated intra-modality sample covariance matrix, entries in the estimated cross-modality sample covariance matrix and estimated diagonal entries using the previous bound, and then calculate the overall convergence rate of using the union bound.

Without loss of generality, we assume $\sigma_{jj}^{XX} = 1$ for $1 \leq j \leq p$. Then, under Condition A.2.1, we know that X_j is sub-Gaussian with parameter L_1. Let $\delta_1 = 8\sqrt{6}\left(1 + 4L_1^2\right)\sqrt{\frac{\log p}{n_{jk}^{XX}}}$. If $n_{XX} > 6\log p$,

we have $\delta_1 < 8\left(1+4L_1^2\right)$. By letting $\nu_1 = 8\sqrt{6}\left(1+4L_1^2\right)$, it follows from Lemma A.9.2 that

$$\begin{aligned}
P\left(\left|\tilde{\sigma}_{jk}^{XX} - \sigma_{jk}^{XX}\right| \geq \delta_1\right) &\leq 4\exp\left\{-\frac{n_{jk}^{XX}\delta_1^2}{128\left(1+4L_1^2\right)^2}\right\} \\
&= 4\exp\left\{-\frac{\nu_1^2 \log p}{128\left(1+4L_1^2\right)^2}\right\} \\
&= 4p^{-\frac{\nu_1^2}{128\left(1+4L_1^2\right)^2}} \\
&\leq \frac{4}{p^3}.
\end{aligned}$$

Hence, under Conditions A.2.1 and A.2.2, we have

$$\max_{j,k} P\left(\left|\tilde{\sigma}_{jk}^{XX} - \sigma_{jk}^{XX}\right| \geq \nu_1\sqrt{\tfrac{\log p}{n_{jk}^{XX}}}\right) \leq \tfrac{4}{p^3}.$$

By the union bound, we have

$$P\left(\|\tilde{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\|_\infty \geq \nu_1\sqrt{\tfrac{\log p}{n_{XX}}}\right) \leq \tfrac{4}{p}.$$

Let Y_j denote the jth response. Without loss of generality, we assume that Y_j has finite variance. Under Condition A.2.1, $Y_j/\sqrt{\mathrm{Var}(Y_j)}$ is sub-Gaussian with parameter $L_2/\sqrt{\mathrm{Var}(Y_j)}$. Let $\delta_3 = 16(1+4\max\{L_1^2, \frac{L_2^2}{\min_j(\mathrm{Var}(Y_j))}\})\sqrt{\frac{\log(pq)}{n_{jk}^{XY}}}\max\{\max_j(\mathrm{Var}(Y_j)), 1\}$. When $n_{XY} > 4\log(pq)$, we have

$$\delta_3 < 8\left(1+4\max\left\{L_1^2, \frac{L_2^2}{\min_j\left(\mathrm{Var}(Y_j)\right)}\right\}\right)\max_j\left(\mathrm{Var}(Y_j), 1\right).$$

By choosing $\nu_2 = 16(1+4\max\{L_1^2, \frac{L_2^2}{\min_j(\mathrm{Var}(Y_j))}\})\max\{\max_j(\mathrm{Var}(Y_j)), 1\}$, it follows from Lemma A.9.2 that for any $1 \leq j, k \leq pq$, we have

$$\begin{aligned}
P\left(\left|\tilde{\sigma}_{jk}^{XY} - \sigma_{jk}^{XY}\right| \geq \delta_3\right) &\leq 4\exp\left\{-\frac{\nu_2^2\log(pq)}{128\left(1+4\max\left\{L_1^2, \frac{L_2^2}{\min_j(\mathrm{Var}(\mathbf{y}_j))}\right\}\right)^2\max_j\left(\mathrm{Var}(Y_j), 1\right)}\right\} \\
&\leq \frac{4}{(pq)^2}.
\end{aligned}$$

Hence, by Condition A.2.1 and $n_{XY} > 4\log(pq)$, there exists a positive constant ν_2 such that

$$\max_{j,k} P\left(\left|\tilde{\sigma}_{jk}^{XY} - \sigma_{jk}^{XY}\right| \geq \nu_2\sqrt{\tfrac{\log(pq)}{n_{jk}^{XY}}}\right) \leq \tfrac{4}{(pq)^2}.$$

By the union bound, we have

$$P\left(\|\tilde{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\|_\infty \geq \nu_2\sqrt{\tfrac{\log(pq)}{n_{XY}}}\right) \leq \tfrac{4}{pq}.$$

Based on the definition of $\hat{\mathbf{\Sigma}}_{XX}$, we have

$$\hat{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX} = \begin{cases} \tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX} & \text{if } j = t; \\ \alpha_1\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX} & \text{if } j \neq t, j \text{ and } t \text{ are in the same modality;} \\ \alpha_2\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX} & \text{if } j \text{ and } t \text{ are in different modalities.} \end{cases}$$

Thus, if $j = t$, there exists a positive constant ν_1 such that with probability at least $1 - 4/p^3$, it holds that

$$\left|\hat{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| = \left|\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| \leq \nu_1\sqrt{\log p/n_X}.$$

If $j \neq t$ and j and t are in the same modality, it holds with probability at least $1 - 4/p^3$ that

$$\begin{aligned} \left|\hat{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| = \left|\alpha_1\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| &\leq \alpha_1\left|\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| + (1-\alpha_1)\left|\sigma_{jt}^{XX}\right| \\ &\leq \alpha_1\left|\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| + 1 - \alpha_1 \\ &\leq \alpha_1\nu_1\sqrt{\log p/n_X} + 1 - \alpha_1. \end{aligned}$$

Similarly, if $j \neq t$ and j and t are in different modalities, it holds with probability at least $1 - 4/p^3$ that

$$\begin{aligned} \left|\hat{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| = \left|\alpha_2\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| &\leq \alpha_2\left|\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| + (1-\alpha_2)\left|\sigma_{jt}^{XX}\right| \\ &\leq \alpha_2\left|\tilde{\sigma}_{jt}^{XX} - \sigma_{jt}^{XX}\right| + 1 - \alpha_2 \\ &\leq \alpha_2\nu_1\sqrt{\log p/n_{XX}} + 1 - \alpha_2. \end{aligned}$$

Therefore, by the union bound, there exists a constant ν_1' such that

$$P\left(\left\|\hat{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}\right\|_\infty \geq v_1'\sqrt{\frac{\log p}{n_{XX}}}\right) \leq \frac{4}{p}.$$

Similarly, it holds with probability at least $1 - 4/(pq)^2$ that

$$\begin{aligned}\left|\hat{\sigma}_{jk}^{XY} - \sigma_{jt}^{XY}\right| = \left|\alpha_3 \tilde{\sigma}_{jt}^{XY} - \sigma_{jt}^{XY}\right| &\leq \alpha_3 \left|\tilde{\sigma}_{jt}^{XY} - \sigma_{jt}^{XY}\right| + (1-\alpha_3)\left|\sigma_{jt}^{XY}\right| \\ &\leq \alpha_3 \left|\tilde{\sigma}_{jt}^{XY} - \sigma_{jt}^{XY}\right| + 1 - \alpha_3 \\ &\leq \alpha_3 \nu_2 \sqrt{\log(pq)/n_{XY}} + 1 - \alpha_3.\end{aligned}$$

Therefore, by the union bound, there exists a constant ν_2' such that

$$P\left(\left\|\hat{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}\right\|_\infty \geq v_2' \sqrt{\frac{\log(pq)}{n_{XY}}}\right) \leq \frac{4}{pq}.$$

Let $\delta_2 = 8\sqrt{6}(1 + 4\frac{L_2^2}{\min_j(\mathrm{Var}(Y_j))}) \sqrt{\frac{\log q}{n_{jk}^{YY}}} \max_j(\mathrm{Var}(Y_j))$. If $n_{YY} > 6\log q$, we have

$$\delta_2 < 8\left(1 + 4\frac{L_2^2}{\min_j\left(\mathrm{Var}(Y_j)\right)}\right) \max_j\left(\mathrm{Var}(Y_j)\right).$$

By choosing $\nu_3' = 8\sqrt{6}(1 + 4\frac{L_2^2}{\min_j(\mathrm{Var}(Y_j))}) \max_j(\mathrm{Var}(Y_j))$, it follows from Lemma A.9.2 that

$$\begin{aligned}P\left(\left|\hat{\sigma}_{jk}^{YY} - \sigma_{jk}^{YY}\right| \geq \delta_2\right) &\leq 4\exp\left\{-\frac{\nu_3'^2 \log q}{128\left(1 + 4\frac{L_2^2}{\min_j(\mathrm{Var}(Y_j))}^2\right)^2}\right\} \\ &\leq \frac{4}{q^3}.\end{aligned}$$

Hence, under Conditions A.2.1 and A.2.2, we have

$$\max_{j,k} P\left(\left|\hat{\sigma}_{jk}^{YY} - \sigma_{jk}^{YY}\right| \geq \nu_3' \sqrt{\tfrac{\log q}{n_{jk}^{YY}}}\right) \leq \tfrac{4}{q^3},$$

where ν_3' is a positive constant. By the union bound, we have

$$P\left(\|\hat{\mathbf{\Sigma}}_{YY} - \mathbf{\Sigma}_{YY}\|_\infty \geq \nu_3' \sqrt{\tfrac{\log q}{n_{YY}}}\right) \leq \tfrac{4}{q}.$$

A.5 Proof of Lemma 2.3.1

We use a similar argument as the proof of Theorem 2 in Yu et al. (2020). By Theorem 2 in Yu et al. (2020), we have $\|\hat{\mathbf{B}}_i - \mathbf{B}_i^*\|_2 = O_p(\sqrt{s_B}\lambda_B)$. In order to prove the ℓ_2–error bound, we only need to prove $\|\hat{\mathbf{\Sigma}}_{XY,i} - \hat{\mathbf{\Sigma}}_{XX}\hat{\mathbf{B}}_i\|_\infty \leq \lambda_B$, where $\hat{\mathbf{\Sigma}}_{XY,i}$ and $\hat{\mathbf{B}}_i$ are the ith column of $\hat{\mathbf{\Sigma}}_{XY}$ and $\hat{\mathbf{B}}$, respectively. Let $\mathbf{\Delta}^{XX} = \hat{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}$ and $\mathbf{\Delta}^{XY} = \hat{\mathbf{\Sigma}}_{XY} - \mathbf{\Sigma}_{XY}$. Let $\mathbf{\Delta}_i^{XX}$ and $\mathbf{\Delta}_i^{XY}$ be the ith column of $\mathbf{\Delta}^{XX}$ and $\mathbf{\Delta}^{XY}$, respectively. We have

$$
\begin{aligned}
&\left\|\hat{\mathbf{\Sigma}}_{XY,i} - \hat{\mathbf{\Sigma}}_{XX}\mathbf{B}_i^*\right\|_\infty \\
=&\left\|\mathbf{\Delta}_i^{XY} - \mathbf{\Delta}^{XX}\mathbf{B}_i^*\right\|_\infty \\
\leq&\left\|\mathbf{\Delta}_i^{XY}\right\|_\infty - \left\|\mathbf{\Delta}^{XX}\right\|_\infty \|\mathbf{B}_i^*\|_{L_1} \\
\leq&(\|\mathbf{B}^*\|_{L_1} v_1' + v_3')\sqrt{\frac{\log(pq)}{\min(n_{XX}, n_{XY})}} \\
\lesssim&\lambda_{B_0}.
\end{aligned}
$$

Denote the ith column of $\hat{\mathbf{B}}_0$ as $\hat{\mathbf{B}}_{0,i}$. By Theorem 2 in Yu et al. (2020), we have

$$
\left\|\hat{\mathbf{B}}_{0,i} - \mathbf{B}_i^*\right\|_F^2 = O_p\left(s_B \left\|\hat{\mathbf{\Sigma}}_{XY,i} - \hat{\mathbf{\Sigma}}_{XX}\mathbf{B}_i^*\right\|_\infty^2\right) = O_p\left(\|\mathbf{B}_i^*\|_1^2 s_B \frac{\log(pq)}{\min(n_{XX}, n_{XY})}\right).
$$

Adding all q columns together, we have

$$
\left\|\hat{\mathbf{B}}_0 - \mathbf{B}^*\right\|_F = O\left(\|\mathbf{B}^*\|_{L_1}\sqrt{\frac{s_B q \log(pq)}{\min(n_{XX}, n_{XY})}}\right).
$$

A.6 Proof of Lemma 2.3.2

We first verify the RSC conditions of the objective function, see (A.29) and (A.30) in Theorem A.9.1. Then we use Theorem 1 of Loh and Wainwright (2015) to prove the convergence rate.

Recall that $\hat{\mathbf{\Sigma}}_0 = \tilde{\mathbf{\Sigma}}_{YY} - 2\hat{\mathbf{\Sigma}}_{XY}^\top\hat{\mathbf{B}}_0 + \hat{\mathbf{B}}_0^\top\hat{\mathbf{\Sigma}}_{XX}\hat{\mathbf{B}}_0$. Let $\mathcal{L}_n(\mathbf{C}) = \mathrm{tr}(\hat{\mathbf{\Sigma}}_0\mathbf{C}) - \log\det(\mathbf{C})$. Its Hessian matrix is $\nabla^2\mathcal{L}_n(\mathbf{C}) = (\mathbf{C}\otimes\mathbf{C})^{-1}$.

For any $\mathbf{\Delta}^{C_0}$ such that $\|\mathbf{\Delta}^{C_0}\|_F \leq 1$. By Mean Value Theorem, there exists some $t \in [0,1]$ such that

$$\begin{aligned}
&\left\langle \nabla \mathcal{L}_n(\mathbf{C}^* + \mathbf{\Delta}^{C_0}) - \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle \\
=& \operatorname{vec}(\mathbf{\Delta}^{C_0})^\top \left(\nabla^2 \mathcal{L}_n\left(\mathbf{C}^* + t\mathbf{\Delta}^{C_0}\right)\right) \operatorname{vec}(\mathbf{\Delta}^{C_0}) \\
\geq& \lambda_{\min}\left(\nabla^2 \mathcal{L}_n\left(\mathbf{C}^* + t\mathbf{\Delta}^{C_0}\right)\right) \|\mathbf{\Delta}^{C_0}\|_F^2 \\
=& \left\|\mathbf{C}^* + t\mathbf{\Delta}^{C_0}\right\|_2^{-2} \|\mathbf{\Delta}^{C_0}\|_F^2 \\
\geq& \left(\|\mathbf{C}^*\|_2 + t\|\mathbf{\Delta}^{C_0}\|_2\right)^{-2} \|\mathbf{\Delta}^{C_0}\|_F^2 \\
\geq& \left(\|\mathbf{C}^*\|_2 + 1\right)^{-2} \|\mathbf{\Delta}^{C_0}\|_F^2.
\end{aligned}$$

Thus, (A.29) holds. Moreover, since $\mathcal{L}_n$ is convex, the function $f(t) := \mathcal{L}_n(\mathbf{C}^* + t\mathbf{\Delta}^{C_0})$ is also convex. So, $f'(1) - f'(0) \geq f'(t) - f'(0)$ for all $t \in [0,1]$. Since

$$\begin{aligned}
f'(1) - f'(0) =& \left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + \mathbf{\Delta}^{C_0}\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle - \left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle \\
=& \left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + \mathbf{\Delta}^{C_0}\right) - \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle, \\
f'(t) - f'(0) =& \left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + t\mathbf{\Delta}^{C_0}\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle - \left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle \\
=& \frac{1}{t}\left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + t\mathbf{\Delta}^{C_0}\right) - \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), t\operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle,
\end{aligned}$$

we have

$$\left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + \mathbf{\Delta}^{C_0}\right) - \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle \geq \frac{1}{t}\left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + t\mathbf{\Delta}^{C_0}\right) - \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), t\operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle.$$

For any $\|\mathbf{\Delta}^{C_0}\|_F \geq 1$, take $t = \frac{1}{\|\mathbf{\Delta}^{C_0}\|_F} \in (0,1]$. Since $\|t\mathbf{\Delta}^{C_0}\|_F = 1$, we have

$$\begin{aligned}
&\left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + \mathbf{\Delta}^{C_0}\right) - \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), \operatorname{vec}\left(\mathbf{\Delta}^{C_0}\right)\right\rangle \\
\geq& \|\mathbf{\Delta}^{C_0}\|_F \left\langle \nabla \mathcal{L}_n\left(\mathbf{C}^* + \frac{\mathbf{\Delta}^{C_0}}{\|\mathbf{\Delta}^{C_0}\|_F}\right) - \nabla \mathcal{L}_n\left(\mathbf{C}^*\right), \operatorname{vec}\left(\frac{\mathbf{\Delta}^{C_0}}{\|\mathbf{\Delta}^{C_0}\|_F}\right)\right\rangle \\
\geq& \|\mathbf{\Delta}^{C_0}\|_F \left(\|\mathbf{C}^*\|_2 + 1\right)^{-2}.
\end{aligned}$$

Thus, (A.30) holds. Denote $\mathbf{\Delta}^{XX} = \mathbf{\Sigma}_{XX} - \hat{\mathbf{\Sigma}}_{XX}$, $\mathbf{\Delta}^{XY} = \mathbf{\Sigma}_{XY} - \hat{\mathbf{\Sigma}}_{XY}$ and $\mathbf{\Delta}^{YY} = \mathbf{\Sigma}_{YY} - \hat{\mathbf{\Sigma}}_{YY}$. Theorem 2.3.1 implies that with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, we have

$$\|\mathbf{\Delta}^{XX}\|_\infty \le v_1' \sqrt{\frac{\log p}{n_{XX}}}, \quad \|\mathbf{\Delta}^{XY}\|_\infty \le v_2' \sqrt{\frac{\log(pq)}{n_{XY}}}, \quad \|\mathbf{\Delta}^{YY}\|_\infty \le v_3' \sqrt{\frac{\log q}{n_{YY}}}.$$

Then, we have

$$\begin{aligned}
&\|\nabla \mathcal{L}_n(\mathbf{C}^*)\|_\infty \\
=&\left\|\mathbf{\Sigma}_\epsilon - \hat{\mathbf{\Sigma}}_0\right\|_\infty \\
\le&\|\mathbf{\Sigma}_\epsilon - \hat{\mathbf{\Sigma}}_{YY} + 2\hat{\mathbf{\Sigma}}_{XY}^\top \mathbf{B}^* - \mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^*\|_\infty + \Big\|\hat{\mathbf{\Sigma}}_{YY} - 2\hat{\mathbf{\Sigma}}_{XY}^\top \mathbf{B}^* + \mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^* - \\
&(\hat{\mathbf{\Sigma}}_{YY} - 2\hat{\mathbf{\Sigma}}_{XY}^\top \hat{\mathbf{B}}_0 + \hat{\mathbf{B}}_0^\top \hat{\mathbf{\Sigma}}_{XX}\hat{\mathbf{B}}_0)\Big\|_\infty \\
\le&2\|\mathbf{B}^* - \hat{\mathbf{B}}_0\|_{L_1}\|\hat{\mathbf{\Sigma}}^{XY}\|_\infty + 2\|\mathbf{B}^* - \hat{\mathbf{B}}_0\|_{L_1}\|\mathbf{B}^*\|_{L_1}\|\hat{\mathbf{\Sigma}}^{XX}\|_\infty + \|\mathbf{B}^* - \hat{\mathbf{B}}_0\|_{L_1} \\
&\|\mathbf{B}^* - \hat{\mathbf{B}}_0\|_{L_1}\|\hat{\mathbf{\Sigma}}^{XX}\|_\infty + \|\mathbf{\Delta}^{YY}\|_\infty + 2\|\mathbf{B}^*\|_{L_1}\|\mathbf{\Delta}^{XY}\|_\infty + \|\mathbf{B}^*\|_{L_1}^2\|\mathbf{\Delta}^{XX}\|_\infty \\
\lesssim&\|\mathbf{B}^*\|_{L_1}^2 \sqrt{\frac{\log(pq)}{\min(n_{XX}, n_{XY})}} + \|\mathbf{B}^*\|_{L_1} s_B \sqrt{\frac{q\log(pq)}{\min(n_{XX}, n_{XY})}}.
\end{aligned}$$

Then, the result follows from Theorem A.9.1.

A.7 Proof of Theorem 2.3.2

We first rely on verifying the RSC conditions of our loss function to express the upper bound of $\|\hat{\mathbf{B}} - \mathbf{B}^*\|_1$ as a function of $\|\hat{\mathbf{C}} - \mathbf{C}^*\|_1$; see (A.15). Similarly, we show that the upper bound of $\|\hat{\mathbf{C}} - \mathbf{C}^*\|_1$ can also be expressed as a function of $\|\hat{\mathbf{B}} - \mathbf{B}^*\|_1$; see (A.18). Combining these two results with some algebra proves the theorem.

For $\mathcal{L}_n(\mathbf{B}, \mathbf{C}) = \mathrm{tr}[\mathbf{C}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{BCB}^\top \hat{\mathbf{\Sigma}}_{\mathbf{XX}} - \mathbf{2CB}^\top \hat{\mathbf{\Sigma}}_{\mathbf{XY}}] - \log\det(\mathbf{C})$, we have $\nabla_B^2 \mathcal{L}_n(\mathbf{B}, \mathbf{C}) = 2\hat{\mathbf{\Sigma}}_{XX} \otimes \mathbf{C}$ and $\nabla_C^2 \mathcal{L}_n(\mathbf{B}, \mathbf{C}) = \mathbf{C}^{-1} \otimes \mathbf{C}^{-1}$.

For $\mathcal{L}(\mathbf{B}, \mathbf{C}) = \mathrm{tr}\left[\mathbf{C}\mathbf{\Sigma}_{\mathbf{YY}} + \mathbf{BCB}^\top \mathbf{\Sigma}_{\mathbf{XX}} - \mathbf{2CB}^\top \mathbf{\Sigma}_{\mathbf{XY}}\right] - \log\det(\mathbf{C})$, we have $\nabla_B^2 \mathcal{L}(\mathbf{B}, \mathbf{C}) = 2\mathbf{\Sigma}_{XX} \otimes \mathbf{C}$ and $\nabla_C^2 \mathcal{L}(\mathbf{B}, \mathbf{C}) = \mathbf{C}^{-1} \otimes \mathbf{C}^{-1}$.

Denote $\mathbf{\Delta}^B = \mathbf{B}^* - \hat{\mathbf{B}}$ and $\mathbf{\Delta}^C = \mathbf{C}^* - \hat{\mathbf{C}}$. For any $t \in [0,1]$, denote $\hat{\mathbf{B}}^t = \mathbf{B}^* + t\mathbf{\Delta}^B$. For any vector $\mathbf{v}_{I_1} \in \mathbb{R}^{pq}$, we have

$$\begin{aligned}
&\mathbf{v}_{I_1}^\top \nabla_B^2 \mathcal{L}(\hat{\mathbf{B}}^t, \hat{\mathbf{C}})\mathbf{v}_{I_1} = 2\mathbf{v}_{I_1}^\top (\mathbf{\Sigma}_{XX} \otimes \hat{\mathbf{C}})\mathbf{v}_{I_1} \\
\geq & 2\|\mathbf{v}_{I_1}\|_2^2 \lambda_{\min}(\mathbf{\Sigma}_{XX})\lambda_{\min}(\hat{\mathbf{C}}) \geq \lambda_{\min}(\mathbf{\Sigma}_{XX})\lambda_{\min}(\hat{\mathbf{C}})\|\mathbf{v}_{I_1}\|_2^2.
\end{aligned}$$

In addition, define

$$\begin{aligned}
\tilde{\epsilon}_n^B &= \max_{t' \in [0,1]} \left\{ \|\nabla_B^2 \mathcal{L}(\hat{\mathbf{B}}^t, \hat{\mathbf{C}}) - \nabla_B^2 \mathcal{L}_n(\hat{\mathbf{B}}^t, \hat{\mathbf{C}})\|_\infty \right\} \\
&= \left\| 2\mathbf{\Delta}^{XX} \otimes \hat{\mathbf{C}} \right\|_\infty,
\end{aligned}$$

where $\mathbf{\Delta}^{XX} = \hat{\mathbf{\Sigma}}_{XX} - \mathbf{\Sigma}_{XX}$. Then, we have

$$\begin{aligned}
&\frac{\mathbf{v}_{I_1}^\top \nabla_B^2 \mathcal{L}_n(\hat{\mathbf{B}}^t, \hat{\mathbf{C}})\mathbf{v}_{I_1}}{\|\mathbf{v}_{I_1}\|_2^2} \\
=& \frac{\mathbf{v}_{I_1}^\top \nabla_B^2 \mathcal{L}(\hat{\mathbf{B}}^t, \hat{\mathbf{C}})\mathbf{v}_{I_1}}{\|\mathbf{v}_{I_1}\|_2^2} + \frac{\mathbf{v}_{I_1}^\top (\nabla_B^2 \mathcal{L}_n(\hat{\mathbf{B}}^t, \hat{\mathbf{C}}) - \nabla_B^2 \mathcal{L}(\hat{\mathbf{B}}^t, \hat{\mathbf{C}}))\mathbf{v}_{I_1}}{\|\mathbf{v}_{I_1}\|_2^2} \\
\geq & \alpha_B - \frac{\tilde{\epsilon}_n^B \|\mathbf{v}_{I_1}\|_1^2}{\|\mathbf{v}_{I_1}\|_2^2},
\end{aligned}$$

where $\alpha_B = \lambda_{\min}(\mathbf{\Sigma}_{XX})(\lambda_{\min}(\mathbf{C}^*) - \lambda_{\min}(\mathbf{\Delta}^C))$.

Let $\boldsymbol{\delta}_B = \mathrm{vec}(\mathbf{\Delta}^B)$ and $\boldsymbol{\delta}_C = \mathrm{vec}(\mathbf{\Delta}^C)$. Then, we have

$$\begin{aligned}
&\left\langle \boldsymbol{\delta}_B, \mathrm{vec}\left(\nabla_B \mathcal{L}_n(\hat{\mathbf{B}}, \hat{\mathbf{C}}) - \nabla_B \mathcal{L}_n(\mathbf{B}^*, \hat{\mathbf{C}}) \right) \right\rangle \\
=& \left\langle \boldsymbol{\delta}_B, \mathrm{vec}\left(\int_0^1 \nabla_B^2 \mathcal{L}_n \left(\mathbf{B}^* + t(\hat{\mathbf{B}} - \mathbf{B}^*), \hat{\mathbf{C}} \right) \mathbf{\Delta}^B dt \right) \right\rangle \\
\geq & \langle \boldsymbol{\delta}_B, \alpha_B \boldsymbol{\delta}_B \rangle - \tilde{\epsilon}_n^B \|\boldsymbol{\delta}_B\|_1^2 \\
=& \alpha_B \|\boldsymbol{\delta}_B\|_2^2 - \tilde{\epsilon}_n^B \|\boldsymbol{\delta}_B\|_1^2.
\end{aligned} \tag{A.1}$$

For any matrix $\mathbf{B} = (B_{ij}) \in \mathbb{R}^{p \times q}$, define $f_1(\mathbf{B}) = (b_{ij})$, where $b_{ij} = 1$ if $B_{ij} > 0$, $b_{ij} = -1$ if $B_{ij} < 0$ and $b_{ij} = 0$ if $B_{ij} = 0$. Similarly, for any matrix $\mathbf{C} = (C_{ij}) \in \mathbb{R}^{q \times q}$, define $f_2(\mathbf{C}) = (c_{ij})$, where $c_{ij} = 1$ if $C_{ij} > 0$, $c_{ij} = -1$ if $C_{ij} < 0$ and $c_{ij} = 0$ if $C_{ij} = 0$. Then $f_1(\mathbf{B}) \in \nabla_B(\|\mathbf{B}\|_1)$ and $f_2(\mathbf{C}) \in \nabla_C(\|\mathbf{C}\|_1)$. Since $\hat{\mathbf{B}}$ is a stationary point of $\mathcal{L}_n + \lambda_B \|\mathbf{B}\|_1$ and $\hat{\mathbf{C}}$ is a stationary point of

$\mathcal{L}_n + \lambda_C \|\mathbf{C}\|_1$, we have

$$\langle \mathrm{vec}(\nabla_B \mathcal{L}_n(\hat{\mathbf{B}}, \hat{\mathbf{C}}) + \lambda_B f_1(\hat{\mathbf{B}})), \boldsymbol{\delta}_B \rangle \geq 0, \tag{A.2}$$

and

$$\langle \mathrm{vec}(\nabla_C \mathcal{L}_n(\hat{\mathbf{B}}, \hat{\mathbf{C}}) + \lambda_C f_2(\hat{\mathbf{C}})), \boldsymbol{\delta}_C \rangle \geq 0. \tag{A.3}$$

By (A.1) and (A.2), we have

$$\begin{aligned}
&\alpha_B \|\boldsymbol{\delta}_B\|_2^2 - \tilde{\epsilon}_n^B \|\boldsymbol{\delta}_B\|_1^2 \\
\leq &\langle \mathrm{vec}(\nabla_B \mathcal{L}_n(\hat{\mathbf{B}}, \hat{\mathbf{C}}) - \nabla_B \mathcal{L}_n(\mathbf{B}^*, \hat{\mathbf{C}})), \boldsymbol{\delta}_B \rangle \\
= &\langle \mathrm{vec}(\nabla_B \mathcal{L}_n(\hat{\mathbf{B}}, \hat{\mathbf{C}})), \boldsymbol{\delta}_B \rangle - \langle \mathrm{vec}(\nabla_B \mathcal{L}_n(\mathbf{B}^*, \hat{\mathbf{C}})), \boldsymbol{\delta}_B \rangle \\
\leq &\langle \mathrm{vec}(\nabla_B (\lambda_B \|\hat{\mathbf{B}}\|_1 + \lambda_C \|\hat{\mathbf{C}}\|_1)), \boldsymbol{\delta}_B \rangle - \langle \mathrm{vec}(\nabla_B \mathcal{L}_n(\mathbf{B}^*, \hat{\mathbf{C}})), \boldsymbol{\delta}_B \rangle \\
\leq &\lambda_B \|\mathbf{B}^*\|_1 - \lambda_B \|\hat{\mathbf{B}}\|_1 + \|\nabla_B \mathcal{L}_n(\mathbf{B}^*, \hat{\mathbf{C}})\|_\infty \|\boldsymbol{\delta}_B\|_1.
\end{aligned} \tag{A.4}$$

Define

$$\begin{aligned}
\tilde{\lambda}_B = &C_\lambda (\log p)^{1/2} / \min(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}) (\|\mathbf{B}^* \mathbf{C}^*\|_{L_1} + \|\mathbf{B}^*\|_{L_1} \|\boldsymbol{\delta}_C\|_1) \\
&+ C_\lambda \max\{\lambda_{\max}(\mathbf{C}^*), 1/\lambda_{\min}(\mathbf{C}^*)\} \{\frac{\log(pq)}{n_{XY}}\}^{1/2} (1 + \|\boldsymbol{\delta}_C\|_1),
\end{aligned} \tag{A.5}$$

where C_λ is a constant only depending on $\lambda_{\max}(\mathbf{C}^*), 1/\lambda_{\min}(\mathbf{C}^*), L_1, L_2$. Then with a large enough constant C_λ, we have $\lambda_B < \tilde{\lambda}_B$. By Lemma A.9.5, we have

$$\|\nabla_B \mathcal{L}_n(\mathbf{B}^*, \hat{\mathbf{C}})\|_\infty \lesssim \tilde{\lambda}_B. \tag{A.6}$$

By Lemma A.9.3, we have

$$\begin{aligned}
\|\boldsymbol{\delta}_B\|_1 = &\left\|\boldsymbol{\Delta}^B\right\|_1 \\
= &\left\|\mathrm{vec}\left(\boldsymbol{\Delta}^B\right)_{S_B}\right\|_1 + \left\|\mathrm{vec}\left(\boldsymbol{\Delta}^B\right)_{S_B^C}\right\|_1 \\
\lesssim &4 \left\|\mathrm{vec}\left(\boldsymbol{\Delta}^B\right)_{S_B}\right\|_1 \\
\lesssim &4\sqrt{s_B} \left\|\mathrm{vec}\left(\boldsymbol{\Delta}^B\right)_{S_B}\right\|_2 \\
\lesssim &\sqrt{s_B} \left\|\boldsymbol{\delta}_B\right\|_2.
\end{aligned} \tag{A.7}$$

Then by (A.4), (A.6) and (A.7), it holds with probability at least $1-\frac{4}{p}-\frac{4}{pq}$ that

$$
\begin{aligned}
&\left\{\alpha_B - 16\tilde{\epsilon}_n^B s_B\right\}\|\boldsymbol{\delta}_B\|_2^2 \\
\leq&\lambda_B\|\mathbf{B}^*\|_1 - \lambda_B\|\hat{\mathbf{B}}\|_1 + \|\nabla_B\mathcal{L}_n(\mathbf{B}^*,\hat{\mathbf{C}})\|_\infty\|\boldsymbol{\delta}_B\|_1 \\
\lesssim&\lambda_B\|\mathbf{B}^*\|_1 - \lambda_B\|\hat{\mathbf{B}}\|_1 + \tilde{\lambda}_B\|\boldsymbol{\delta}_B\|_1 \\
\lesssim&\lambda_B\|\mathbf{B}^*\|_1 - \lambda_B\|\hat{\mathbf{B}}\|_1 + \tilde{\lambda}_B\|\operatorname{vec}\left(\boldsymbol{\Delta}^B\right)_{S_B}\|_1 \\
\lesssim&\tilde{\lambda}_B\|\operatorname{vec}(\boldsymbol{\Delta}^B)_{S_B}\|_1 - \lambda_B\|\operatorname{vec}(\hat{\mathbf{B}})_{S_B^C}\|_1 \\
\lesssim&\tilde{\lambda}_B\sqrt{s_B}\|\boldsymbol{\delta}_B\|_2.
\end{aligned}
\tag{A.8}
$$

Next we show that with large enough n_{XX} and q, $\alpha_B - 16\tilde{\epsilon}_n^B s_B$ is bounded away from 0. To show α_B is bounded away from 0, we first prove that $\mathcal{L}_n(\mathbf{B},\mathbf{C})$ satisfies the RSC condition (A.29) and (A.30) with respect to $\mathbf{C}$ for any $\mathbf{B}$.

For any $t'\in[0,1]$, denote $\hat{\mathbf{C}}^{t'} = \mathbf{C}^* + t'\boldsymbol{\Delta}^C$. For any vector $\mathbf{v}_{I_2}\in\mathbb{R}^{q^2}$, we have

$$
\begin{aligned}
&\mathbf{v}_{I_2}^\top\nabla_C^2\mathcal{L}(\hat{\mathbf{B}},\hat{\mathbf{C}}^{t'})\mathbf{v}_{I_2} \\
=&\mathbf{v}_{I_2}^\top((\hat{\mathbf{C}}^{t'})^{-1}\otimes(\hat{\mathbf{C}}^{t'})^{-1})\mathbf{v}_{I_2} \\
\geq&(\|\mathbf{C}^*\|_2 + t'\|\boldsymbol{\Delta}^C\|_2)^{-2}\|\mathbf{v}_{I_2}\|_2^2,
\end{aligned}
$$

where we use the Weyl's inequality that $\lambda_{\max}(\mathbf{C}^*) - t'\lambda_{\max}(\boldsymbol{\Delta}^{\mathbf{C}}) \geq \lambda_{\max}(\hat{\mathbf{C}}^{t'})$. Then, for all $\|\boldsymbol{\Delta}^C\|_F\leq 1$ and any $\mathbf{B}$, we have

$$
\frac{\mathbf{v}_{I_2}^\top\nabla_C^2\mathcal{L}_n(\mathbf{B},\hat{\mathbf{C}}^{t'})\mathbf{v}_{I_1}}{\|\mathbf{v}_{I_2}\|_2^2} \geq \quad (\|\mathbf{C}^*\|_2 + t'\|\boldsymbol{\Delta}^C\|_2)^{-2} \geq(\|\mathbf{C}^*\|_2+1)^{-2}.
$$

Then, for any $\|\boldsymbol{\Delta}^C\|_F\leq 1$ and any $\mathbf{B}$ we have

$$
\begin{aligned}
&\langle\boldsymbol{\delta}_C, \operatorname{vec}(\nabla_C\mathcal{L}_n(\mathbf{B},\hat{\mathbf{C}}) - \nabla_C\mathcal{L}_n(\mathbf{B},\mathbf{C}^*))\rangle \\
=&\left\langle\boldsymbol{\delta}_C, \operatorname{vec}\left(\int_0^1\nabla_C^2\mathcal{L}_n\left(\mathbf{B},\mathbf{C}^*+t'(\hat{\mathbf{C}}-\mathbf{C}^*)\right)\boldsymbol{\Delta}^C dt\right)\right\rangle \\
\geq&\alpha_C\|\boldsymbol{\delta}_C\|_2^2,
\end{aligned}
\tag{A.9}
$$

where $\alpha_C = (\|\mathbf{C}^*\|_2+1)^{-2}$. If $\|\mathbf{\Delta}^C\|_F > 1$, since $\mathcal{L}_n(\mathbf{B},\mathbf{C})$ is convex with respect to $\mathbf{C}$, the function $f:[0,1]\to\mathbb{R}$ given by $f(t) := \mathcal{L}_n(\mathbf{B},\mathbf{C}^*+t'\mathbf{\Delta}^C)$ is also convex, so $f'(1)-f'(0)\geq f'(t')-f'(0)$ for all $t'\in[0,1]$. Computing the derivatives of f yields

$$\begin{aligned}&\langle \mathrm{vec}(\nabla_C\mathcal{L}_n(\mathbf{B},\hat{\mathbf{C}})-\nabla_C\mathcal{L}_n(\mathbf{B},\mathbf{C}^*)),\boldsymbol{\delta}_C\rangle\\ \geq&\frac{1}{t'}\left\langle \mathrm{vec}(\nabla\mathcal{L}_n\left(\mathbf{B},\mathbf{C}^*+t'\mathbf{\Delta}^C\right)-\nabla\mathcal{L}_n\left(\mathbf{B},\mathbf{C}^*\right)),t'\boldsymbol{\delta}_C\right\rangle.\end{aligned}$$

Taking $t'=\frac{1}{\|\mathbf{\Delta}^C\|_F}\in(0,1]$, for any $\|\mathbf{\Delta}^C\|_F>1$ and any $\mathbf{B}$, we have

$$\langle \mathrm{vec}(\nabla_C\mathcal{L}_n(\mathbf{B},\hat{\mathbf{C}})-\nabla_C\mathcal{L}_n(\mathbf{B},\mathbf{C}^*)),\boldsymbol{\delta}_C\rangle\geq\alpha_C\|\boldsymbol{\delta}_C\|_2. \tag{A.10}$$

Combining (A.9) and (A.10), we show that $\mathcal{L}_n(\mathbf{B},\mathbf{C})$ satisfies the RSC conditions (A.29) and (A.30) with respect to $\mathbf{C}$ for any $\mathbf{B}$. Next, following the proof of Lemma A.9.1 from Loh and Wainwright (2015), we can prove $\|\boldsymbol{\delta}_C\|_2\leq 3(\|\mathbf{C}^*\|_2+1)^2/2$. For completeness, we prove it as follows.

By (A.3) and (A.10), we have

$$\left\langle \mathrm{vec}(-\lambda_C f_2(\hat{\mathbf{C}})-\nabla_C\mathcal{L}_n(\hat{\mathbf{B}},\mathbf{C}^*)),\boldsymbol{\delta}_C\right\rangle\geq\alpha_C\|\boldsymbol{\delta}_C\|_2.$$

By Hölder's inequality and the triangle inequality, we also have

$$\left\langle \mathrm{vec}(-\lambda_C f_2(\hat{\mathbf{C}})-\nabla_C\mathcal{L}_n(\hat{\mathbf{B}},\mathbf{C}^*)),\boldsymbol{\delta}_C\right\rangle\leq\frac{3}{2}\lambda_C\|\boldsymbol{\delta}_C\|_1.$$

Combining the above two inequalities yields

$$\|\boldsymbol{\delta}_C\|_2\leq\frac{3\|\boldsymbol{\delta}_C\|_1\lambda_C}{2\alpha_C}\leq\frac{3R\lambda_C}{2\alpha_C}. \tag{A.11}$$

With our choice of λ_C and R, and large enough n_{XX},n_{XY},n_{YY}, we have $\|\boldsymbol{\delta}_C\|_2\leq 3(\|\mathbf{C}^*\|_2+1)^2/2$. Since $\sqrt{\sum_{i=1}^q|\lambda_i(\mathbf{\Delta}^C)|^2}\leq\|\mathbf{\Delta}^C\|_F$, where $\lambda_i(\mathbf{\Delta}^C)$ denotes all the q eigenvalues of $\mathbf{\Delta}^C$, we have $\lambda_{\min}(\mathbf{\Delta}^C)\leq\frac{3(\|\mathbf{C}^*\|_2+1)^2}{2q}$. Thus with large enough q, $\alpha_B=\lambda_{\min}(\mathbf{\Sigma}_{XX})(\lambda_{\min}(\mathbf{C}^*)-\lambda_{\min}(\mathbf{\Delta}^C))$ is bounded away from 0 by Condition A.2.4.

Denote $\mathbf{\Delta}^{YY} = \mathbf{\Sigma}_{YY} - \hat{\mathbf{\Sigma}}_{YY}$. Theorem 2.3.1 implies that with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, we have

$$\|\mathbf{\Delta}^{XX}\|_\infty \leq v_1' \sqrt{\frac{\log p}{n_{XX}}}; \tag{A.12}$$

$$\|\mathbf{\Delta}^{XY}\|_\infty \leq v_2' \sqrt{\frac{\log(pq)}{n_{XY}}}; \tag{A.13}$$

$$\|\mathbf{\Delta}^{YY}\|_\infty \leq v_3' \sqrt{\frac{\log q}{n_{YY}}}. \tag{A.14}$$

By inequalities (A.12), (A.13), (A.14) and Condition A.2.3, with probability at least $1 - \frac{4}{p}$, it holds that

$$\tilde{\epsilon}_n^B = 2\|\mathbf{\Delta}^{XX}\|_\infty \|\hat{\mathbf{C}}\|_\infty \lesssim v_1' \left(\frac{\log p}{n_{XX}}\right)^{\frac{1}{2}-\gamma_2}.$$

Thus when n_{XX} and q are large enough, $\alpha_B - 16\tilde{\epsilon}_n^B s_B$ is bounded away from 0. Then by (A.8), it holds with probability at least $1 - \frac{4}{p} - \frac{4}{pq}$ that

$$\|\boldsymbol{\delta}_B\|_2 \lesssim \tilde{\lambda}_B \sqrt{s_B}.$$

By (A.7), it holds with probability at least $1 - \frac{4}{p} - \frac{4}{pq}$ that

$$\|\boldsymbol{\delta}_B\|_1 \lesssim \sqrt{s_B}\, \|\boldsymbol{\delta}_B\|_2 \lesssim \tilde{\lambda}_B s_B, \tag{A.15}$$

where $\tilde{\lambda}_B$ is as stated in (A.5). Next, we show that the upper bound of $\|\mathbf{C}^* - \hat{\mathbf{C}}\|_1$ can also be expressed as a function of $\|\hat{\mathbf{B}} - \mathbf{B}^*\|_1$. By (A.3) and (A.9), we have

$$\alpha_C \|\boldsymbol{\delta}_C\|_2^2 \leq \lambda_C \|\mathbf{C}^*\|_1 - \lambda_C \|\hat{\mathbf{C}}\|_1 - \langle \mathrm{vec}(\nabla_C \mathcal{L}_n(\hat{\mathbf{B}}, \mathbf{C}^*)), \boldsymbol{\delta}_C \rangle.$$

By (A.12), (A.13) and (A.14), it holds with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$ that

$$
\begin{aligned}
&\left\|\nabla_C \mathcal{L}_n(\hat{\mathbf{B}}, \mathbf{C}^*)\right\|_\infty \\
=&\left\|\mathbf{\Sigma}_\epsilon - \hat{\mathbf{\Sigma}}_0\right\|_\infty \\
\leq&\|\mathbf{\Sigma}_\epsilon - \hat{\mathbf{\Sigma}}_{YY} + 2\hat{\mathbf{\Sigma}}_{XY}^\top \mathbf{B}^* - \mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^*\|_\infty + \left\|\hat{\mathbf{\Sigma}}_{YY} - 2\hat{\mathbf{\Sigma}}_{XY}^\top \mathbf{B}^* + \mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^* - \right. \\
&\left.(\hat{\mathbf{\Sigma}}_{YY} - 2\hat{\mathbf{\Sigma}}_{XY}^\top \hat{\mathbf{B}} + \hat{\mathbf{B}}^\top\hat{\mathbf{\Sigma}}_{XX}\hat{\mathbf{B}})\right\|_\infty \\
\lesssim&\|\mathbf{B}^*\|_{L_1}^2\sqrt{\frac{\log(pq)}{\min(n_{XX}, n_{XY})}} + \|\boldsymbol{\delta}_B\|_1.
\end{aligned}
$$

Then, with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, it holds that

$$
\|\boldsymbol{\delta}_C\|_2 \lesssim \sqrt{s_C}\|\mathbf{C}^*\|_2^2\|\mathbf{B}^*\|_{L_1}^2\sqrt{\frac{\log(pq)}{\min(n_{XX}, n_{XY})}} + \sqrt{s_C}\|\mathbf{C}^*\|_2^2\|\boldsymbol{\delta}_B\|_1. \tag{A.16}
$$

By Lemma A.9.3, we have

$$
\begin{aligned}
\|\boldsymbol{\delta}_C\|_1 =& \left\|\mathrm{vec}\left(\mathbf{\Delta}^C\right)_{S_C}\right\|_1 + \left\|\mathrm{vec}\left(\mathbf{\Delta}^C\right)_{S_C^C}\right\|_1 \\
\lesssim& 4\left\|\mathrm{vec}\left(\mathbf{\Delta}^C\right)_{S_C}\right\|_1 \\
\lesssim& \sqrt{s_C}\left\|\boldsymbol{\delta}_C\right\|_2.
\end{aligned} \tag{A.17}
$$

Finally, we combine (A.15), (A.16) and (A.17) to show the upper bounds of the estimation errors of $\hat{\mathbf{C}}$ and $\hat{\mathbf{B}}$.

By (A.5), (A.16) and (A.17, with large enough n_{XX}, n_{XY}, it holds with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$ that

$$
\begin{aligned}
&\|\boldsymbol{\delta}_C\|_1 \\
\lesssim & s_C\|\mathbf{C}^*\|_2^2\|\mathbf{B}^*\|_{L_1}^2\sqrt{\frac{\log(pq)}{\min(n_{XX},n_{XY})}}+s_Bs_C\|\mathbf{C}^*\|_2^2\left(\frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2},n_{XY}^{1-\tau_2/2}\right)}\right. \\
&(\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}+\|\mathbf{B}^*\|_{L_1}\|\boldsymbol{\delta}_C\|_1)+\max\{\lambda_{\max}(\mathbf{C}^*),1/\lambda_{\min}(\mathbf{C}^*)\}\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2} \\
&\left.(1+\|\boldsymbol{\delta}_C\|_1)\right) \\
\lesssim & \|\mathbf{C}^*\|_2^2 s_C\left(\|\mathbf{B}^*\|_{L_1}^2+\frac{\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}s_B}{\min\left(n_{XX}^{1/2-\tau_1/2},n_{XY}^{1/2-\tau_2/2}\right)}\right)\sqrt{\frac{\log(pq)}{\min(n_{XX},n_{XY})}} \\
&+\|\boldsymbol{\delta}_C\|_1 s_Bs_C\|\mathbf{C}^*\|_2^2\left(\frac{(\log p)^{1/2}\|\mathbf{B}^*\|_{L_1}}{\min\left(n_{XX}^{1-\tau_1/2},n_{XY}^{1-\tau_2/2}\right)}+\max\{\lambda_{\max}(\mathbf{C}^*),1/\lambda_{\min}(\mathbf{C}^*)\}\right. \\
&\left.\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}\right).
\end{aligned}
$$

With large enough n_{XX}, n_{XY}, we have

$$
s_Bs_C\|\mathbf{C}^*\|_2^2(\frac{(\log p)^{1/2}\|\mathbf{B}^*\|_{L_1}}{\min\left(n_{XX}^{1-\tau_1/2},n_{XY}^{1-\tau_2/2}\right)}+\max\{\lambda_{\max}(\mathbf{C}^*),1/\lambda_{\min}(\mathbf{C}^*)\}\{\frac{\log(pq)}{n_{XY}}\}^{1/2})=o(1),
$$

so we have

$$
\begin{aligned}
&\|\boldsymbol{\delta}_C\|_1 \\
\lesssim & \|\mathbf{C}^*\|_2^2 s_C\left(\|\mathbf{B}^*\|_{L_1}^2+\frac{\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}s_B}{\min\left(n_{XX}^{1/2-\tau_1/2},n_{XY}^{1/2-\tau_2/2}\right)}\right)\sqrt{\frac{\log(pq)}{\min(n_{XX},n_{XY})}} \\
\lesssim & \lambda_C s_C.
\end{aligned}
\tag{A.18}
$$

By choosing large enough n_{XX}, n_{XY} and n_{YY}, it holds with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$ that

$$
\|\boldsymbol{\delta}_C\|_1 \lesssim 1. \tag{A.19}
$$

By (A.16) and (A.19), it holds with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$ that

$$
\begin{aligned}
&\|\boldsymbol{\delta}_C\|_2 \\
\lesssim &\sqrt{s_C}\|\mathbf{C}^*\|_2^2\|\mathbf{B}^*\|_{L_1}^2\sqrt{\frac{\log(pq)}{\min(n_{XX},n_{XY})}}+s_B\sqrt{s_C}\|\mathbf{C}^*\|_2^2\left(\frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2},n_{XY}^{1-\tau_2/2}\right)}\right. \\
&(\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}+\|\mathbf{B}^*\|_{L_1}\|\boldsymbol{\delta}_C\|_1)+\max\left\{\lambda_{\max}(\mathbf{C}^*),1/\lambda_{\min}(\mathbf{C}^*)\right\}\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2} \\
&\left.(1+\|\boldsymbol{\delta}_C\|_1)\right) \\
\lesssim &\|\mathbf{C}^*\|_2^2\sqrt{s_C}\left(\|\mathbf{B}^*\|_{L_1}^2+\frac{\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}s_B}{\min\left(n_{XX}^{1/2-\tau_1/2},n_{XY}^{1/2-\tau_2/2}\right)}\right)\sqrt{\frac{\log(pq)}{\min(n_{XX},n_{XY})}} \\
\lesssim &\lambda_C\sqrt{s_C}.
\end{aligned}
$$

By Lemma A.9.5 and (A.19), with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, we have $\|\nabla_B\mathcal{L}_n(\mathbf{B}^*,\hat{\mathbf{C}})\|_\infty\lesssim\lambda_B$. Then by (A.15), it holds with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$ that

$$
\begin{aligned}
&\|\boldsymbol{\delta}_B\|_2^2 \\
\leq &\lambda_B\|\mathbf{B}^*\|_1-\lambda_B\|\hat{\mathbf{B}}\|_1+\|\nabla_B\mathcal{L}_n(\mathbf{B}^*,\hat{\mathbf{C}})\|_\infty\|\boldsymbol{\delta}_B\|_1 \\
\lesssim &\lambda_B\|\mathbf{B}^*\|_1-\lambda_B\|\hat{\mathbf{B}}\|_1+\lambda_B\|\boldsymbol{\delta}_B\|_1 \\
\lesssim &\lambda_B\sqrt{s_B}\|\boldsymbol{\delta}_B\|_2.
\end{aligned}
$$

So with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, it holds that

$$\|\boldsymbol{\delta}_B\|_2\lesssim\lambda_B\sqrt{s_B},$$

and

$$\|\boldsymbol{\delta}_B\|_1\lesssim\lambda_B s_B.$$

This completes the proof.

A.8 Proof of Theorem 2.3.3

We use a similar argument as the proof of Theorem 6 in Yu et al. (2020). We first transfer the objective function. Then we show the upper bounds of $\|(\mathbf{\Gamma}_{S_B S_B})^{-1}\|_{L_\infty}$ and $\left\|\widehat{\boldsymbol{\gamma}}_{S_B^C} - \widehat{\mathbf{\Gamma}}_{S_B^C S_B}\boldsymbol{\beta}^*_{S_B}\right\|_\infty$. We use them to show that $\|\hat{\boldsymbol{\beta}}_{S_B} - {\boldsymbol{\beta}_{S_B}}^*\|_\infty < \min_{j\in S_B}|\boldsymbol{\beta}^*_j|$ with probability close to 1. Then we show that $\left\|\widehat{\boldsymbol{\gamma}}_{S_B^C} - \widehat{\mathbf{\Gamma}}_{S_B^C S_B}\hat{\boldsymbol{\beta}}_{S_B}\right\|_\infty \le \lambda_B$ with probability close to 1.

By properties of trace and vectorization, we can rewrite (2.5) as

$$
\begin{aligned}
&(\hat{\mathbf{B}}, \hat{\mathbf{C}}) \\
=&\arg\min_{\mathbf{C}\in\mathbb{S}_+^{q\times q},\mathbf{B}} \left\{\operatorname{tr}\left[\mathbf{C}\hat{\mathbf{\Sigma}}_{YY}\right] + \operatorname{tr}\left[\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}\mathbf{C}\right] - 2\operatorname{tr}\left[\mathbf{B}^\top\hat{\mathbf{\Sigma}}_{XY}\mathbf{C}\right] + \lambda_B\|\mathbf{B}\|_1 + \lambda_C\|\mathbf{C}\|_1 - \log\det\mathbf{C}\right\} \\
=&\arg\min_{\mathbf{C}\in\mathbb{S}_+^{q\times q},\mathbf{B}} \left\{\operatorname{tr}\left[\mathbf{C}\hat{\mathbf{\Sigma}}_{YY}\right] + \operatorname{vec}(\mathbf{B})^\top\left(\mathbf{C}\otimes\hat{\mathbf{\Sigma}}_{XX}\right)\operatorname{vec}(\mathbf{B}) - 2\operatorname{vec}(\mathbf{B})^\top\operatorname{vec}\left(\hat{\mathbf{\Sigma}}_{XY}\mathbf{C}\right) + \lambda_B\|\mathbf{B}\|_1 + \right. \\
&\left.\lambda_C\|\mathbf{C}\|_1 - \log\det\mathbf{C}\right\}.
\end{aligned}
$$

Denote $\boldsymbol{\beta} = \operatorname{vec}(\mathbf{B})$, (2.5) is equivalent to solving

$$
\begin{aligned}
(\hat{\boldsymbol{\beta}}, \hat{\mathbf{C}}) = \arg\min_{\boldsymbol{\beta},\mathbf{C}\in\mathbb{S}_+^{q\times q}} &\left\{\operatorname{tr}\left[\mathbf{C}\hat{\mathbf{\Sigma}}_{YY}\right] - \log\det\mathbf{C} - 2\boldsymbol{\beta}^\top\operatorname{vec}(\hat{\mathbf{\Sigma}}_{XY}\mathbf{C})\right. \\
&\left.+\boldsymbol{\beta}^\top\left(\mathbf{C}\otimes\hat{\mathbf{\Sigma}}_{XX}\right)\boldsymbol{\beta} + \lambda_B\|\boldsymbol{\beta}\|_1 + \lambda_C\|\mathbf{C}\|_1\right\},
\end{aligned}
\tag{A.20}
$$

For an optimal solution $(\hat{\boldsymbol{\beta}}, \hat{\mathbf{C}})$ to (A.20), $\hat{\boldsymbol{\beta}}$ should satisfy

$$
\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}}\left\{-2\boldsymbol{\beta}^\top\hat{\boldsymbol{\gamma}} + \boldsymbol{\beta}^\top\hat{\mathbf{\Gamma}}\boldsymbol{\beta} + \lambda_B\|\boldsymbol{\beta}\|_1\right\}, \tag{A.21}
$$

where $\widehat{\mathbf{\Gamma}} = \hat{\mathbf{C}}\otimes\hat{\mathbf{\Sigma}}_{XX}$ and $\widehat{\boldsymbol{\gamma}} = \operatorname{vec}(\hat{\mathbf{\Sigma}}_{XY}\hat{\mathbf{C}})$. This can be proved by contradiction. If $\hat{\boldsymbol{\beta}}$ does not satisfy (A.21), let $\boldsymbol{\beta}_1$ be a solution of (A.21). Denote $\mathcal{L}_n(\boldsymbol{\beta}, \mathbf{C}) = \operatorname{tr}\left[\mathbf{C}\hat{\mathbf{\Sigma}}_{YY}\right] - \log\det\mathbf{C} - 2\boldsymbol{\beta}^\top\operatorname{vec}(\hat{\mathbf{\Sigma}}_{XY}\mathbf{C}) + \boldsymbol{\beta}^\top\left(\mathbf{C}\otimes\hat{\mathbf{\Sigma}}_{XX}\right)\boldsymbol{\beta} + \lambda_B\|\boldsymbol{\beta}\|_1 + \lambda_C\|\mathbf{C}\|_1$. Then

$$
\begin{aligned}
&\mathcal{L}_n(\hat{\boldsymbol{\beta}}, \hat{\mathbf{C}}) - \mathcal{L}_n(\boldsymbol{\beta}_1, \hat{\mathbf{C}}) \\
=&\left(2\hat{\boldsymbol{\beta}}^\top\hat{\boldsymbol{\gamma}} + \hat{\boldsymbol{\beta}}^\top\hat{\mathbf{\Gamma}}\hat{\boldsymbol{\beta}} + \lambda_B\|\hat{\boldsymbol{\beta}}\|_1\right) - \left(2\boldsymbol{\beta}_1^\top\hat{\boldsymbol{\gamma}} + \boldsymbol{\beta}_1^\top\hat{\mathbf{\Gamma}}\boldsymbol{\beta}_1 + \lambda_B\|\boldsymbol{\beta}_1\|_1\right) \\
>&0,
\end{aligned}
$$

which is a contradiction. Thus $\hat{\boldsymbol{\beta}}$ should satisfy (A.21). Since $\hat{\mathbf{C}}$ is the optimal solution to (A.20), it is positive definite. By our construction, $\hat{\boldsymbol{\Sigma}}_{XX}$ is also positive definite. Thus (A.21) is a strictly convex problem, which has a unique solution. Thus $\hat{\boldsymbol{\beta}}$ is the unique solution to (A.21).

By the Karush–Kuhn–Tucker (KKT) conditions of (A.21), we know that $\hat{\boldsymbol{\beta}}$ is a solution to $(A.21)$ if there exists a subgradient $\boldsymbol{\omega}^B \in \mathbb{R}^{pq}$ such that

$$\widehat{\boldsymbol{\gamma}} - \widehat{\boldsymbol{\Gamma}}\hat{\boldsymbol{\beta}} = \lambda_B \boldsymbol{\omega}^B, \tag{A.22}$$

where $\boldsymbol{\omega}_j^B = \text{sign}(\hat{\beta}_j)$ if $\hat{\beta}_j \neq 0$, and $\boldsymbol{\omega}_j^B \in [-1, 1]$ if $\hat{\beta}_j = 0$.

First, we show that for $j \in S_B$, with high probability, there exists a solution $\hat{\boldsymbol{\beta}}$ to (A.21) s.t.

$$\left\| \hat{\boldsymbol{\beta}}_{S_B} - \boldsymbol{\beta}^*_{S_B} \right\|_\infty < \min_{j \in S_B} |\beta_j^*|,$$

where $\hat{\boldsymbol{\beta}}_{S_B}$ is the sub-vector of $\hat{\boldsymbol{\beta}}$ with indices in S_B. Then letting $\hat{\boldsymbol{\beta}}_{S_B^C} = \mathbf{0}$, we show that $\hat{\boldsymbol{\beta}}$ also satisfies the KKT conditions with high probability for $j \notin S_B$. Then, by construction, $\text{sign}(\hat{\boldsymbol{\beta}}) = \text{sign}(\boldsymbol{\beta}^*)$. Define events $\mathcal{A}_1 = \{\|\hat{\boldsymbol{\beta}}_{S_B} - \boldsymbol{\beta}^*_{S_B}\|_\infty < \min_{j \in S_B} |\beta_j^*|\}$ and $\mathcal{A}_2 = \{\|\hat{\boldsymbol{\gamma}}_{S_B^C} - \hat{\boldsymbol{\Gamma}}_{S_B^C S_B}\hat{\boldsymbol{\beta}}_{S_B}\|_\infty \leq \lambda_B\}$, where $\boldsymbol{\beta}^* = \text{vec}(\mathbf{B}^*)$. We show that $P(\mathcal{A}_1)$ and $P(\mathcal{A}_2)$ are close to 1.

Denote $V = \|(\boldsymbol{\Gamma}_{S_B S_B})^{-1}\|_{L_\infty}$, where $\boldsymbol{\Gamma} := \mathbf{C}^* \otimes \boldsymbol{\Sigma}_{XX}$. Since

$$\begin{aligned}
&\left\| \left(\widehat{\boldsymbol{\Gamma}}_{S_B S_B}\right)^{-1} - (\boldsymbol{\Gamma}_{S_B S_B})^{-1} \right\|_{L_\infty} \\
\leq &\left\| (\boldsymbol{\Gamma}_{S_B S_B})^{-1} \right\|_{L_\infty} \left\| \left(\widehat{\boldsymbol{\Gamma}}_{S_B S_B}\right)^{-1} \right\|_{L_\infty} \left\| \widehat{\boldsymbol{\Gamma}}_{S_B S_B} - \boldsymbol{\Gamma}_{S_B S_B} \right\|_{L_\infty} \\
\leq &\left\| (\boldsymbol{\Gamma}_{S_B S_B})^{-1} \right\|_{L_\infty} \left(\left\| (\boldsymbol{\Gamma}_{S_B S_B})^{-1} \right\|_{L_\infty} + \left\| \left(\widehat{\boldsymbol{\Gamma}}_{S_B S_B}\right)^{-1} - (\boldsymbol{\Gamma}_{S_B S_B})^{-1} \right\|_{L_\infty} \right) \\
&\left\| \widehat{\boldsymbol{\Gamma}}_{S_B S_B} - \boldsymbol{\Gamma}_{S_B S_B} \right\|_{L_\infty} \\
= &V \left(V + \left\| \left(\widehat{\boldsymbol{\Gamma}}_{S_B S_B}\right)^{-1} - (\boldsymbol{\Gamma}_{S_B S_B})^{-1} \right\|_{L_\infty} \right) \left\| \widehat{\boldsymbol{\Gamma}}_{S_B S_B} - \boldsymbol{\Gamma}_{S_B S_B} \right\|_{L_\infty},
\end{aligned}$$

by some algebra, we have,

$$\left\|\left(\widehat{\mathbf{\Gamma}}_{S_BS_B}\right)^{-1}-(\mathbf{\Gamma}_{S_BS_B})^{-1}\right\|_{L_\infty} \leq \frac{V^2\left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_{L_\infty}}{1-V\left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_{L_\infty}} \leq \frac{s_BV^2\left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_{\infty}}{1-s_BV\left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_{\infty}},$$

and

$$\left\|\left(\widehat{\mathbf{\Gamma}}_{S_BS_B}\right)^{-1}\right\|_{L_\infty} \leq V+\frac{s_BV^2\left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_{\infty}}{1-s_BV\left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_{\infty}} = \frac{V}{1-s_BV\left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_{\infty}}.$$

By Theorem 2.3.2, with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, it holds that

$$\left\|\hat{\mathbf{C}}-\mathbf{C}^*\right\|_1 \lesssim \lambda_C s_C, \tag{A.23}$$

where $\lambda_C = C\|\mathbf{C}^*\|_2^2[\|\mathbf{B}^*\|_{L_1}^2 + s_B\|\mathbf{B}^*\mathbf{C}^*\|_{L_1}/\min(n_{XX}^{1/2-\tau_1/2}, n_{XY}^{1/2-\tau_2/2})](\log(pq)/\min(n_{XX}, n_{YY}))^{1/2}$. Denote $\mathbf{\Delta}^C = \hat{\mathbf{C}} - \mathbf{C}^*$. By (A.23) and Condition A.2.3, with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, it holds that

$$\left\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}\right\|_\infty \leq \left(\|\mathbf{C}^*\|_\infty + \|\mathbf{\Delta}^C\|_\infty\right)\|\hat{\mathbf{\Sigma}}_{XX}-\mathbf{\Sigma}_{XX}\|_\infty \lesssim \left(\frac{\log p}{n_{XX}}\right)^{\frac{1}{2}-\gamma_2}. \tag{A.24}$$

Define $\boldsymbol{\gamma} := \mathrm{vec}(\mathbf{\Sigma}_{XY}\mathbf{C}^*)$. Then, with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, it holds that

$$\begin{aligned}\left\|\widehat{\boldsymbol{\gamma}}_{S_B}-\widehat{\mathbf{\Gamma}}_{S_BS_B}\boldsymbol{\beta}^*_{S_B}\right\|_\infty &\leq \left\|\widehat{\boldsymbol{\gamma}}_{S_B}-\boldsymbol{\gamma}_{S_B}\right\|_\infty + \left\|\left(\mathbf{\Gamma}_{S_BS_B}-\widehat{\mathbf{\Gamma}}_{S_BS_B}\right)\boldsymbol{\beta}^*_{S_B}\right\|_\infty \\ &\leq \left\|\widehat{\boldsymbol{\gamma}}_{S_B}-\boldsymbol{\gamma}_{S_B}\right\|_\infty + \left\|\mathbf{\Gamma}_{S_BS_B}-\widehat{\mathbf{\Gamma}}_{S_BS_B}\right\|_\infty \left\|\boldsymbol{\beta}^*_{S_B}\right\|_\infty \\ &\leq \left\|\widehat{\boldsymbol{\gamma}}_{S_B}-\boldsymbol{\gamma}_{S_B}\right\|_\infty + s_B\|\mathbf{B}^*\|_\infty \left\|\widehat{\mathbf{\Gamma}}_{S_BS_B}-\mathbf{\Gamma}_{S_BS_B}\right\|_\infty.\end{aligned}$$

By (A.23), with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, it holds that

$$\begin{aligned}
&\|\widehat{\boldsymbol{\gamma}}-\boldsymbol{\gamma}\|_\infty \\
&\leq \left(\|\mathbf{C}^*\|_{L_\infty}+\|\boldsymbol{\Delta}^C\|_{L_\infty}\right)\|\hat{\boldsymbol{\Sigma}}_{XY}-\boldsymbol{\Sigma}_{XY}\|_\infty \\
&\lesssim \left(\frac{\log(pq)}{n_{XY}}\right)^{\frac{1}{2}-\gamma_2}. \qquad \text{(A.25)}
\end{aligned}$$

Since $\hat{\boldsymbol{\beta}}_{S_B} = (\widehat{\boldsymbol{\Gamma}}_{S_BS_B})^{-1}\widehat{\boldsymbol{\gamma}}_{S_B} - \lambda_B(\hat{\boldsymbol{\Sigma}}_{XX,S_BS_B})^{-1}\operatorname{sign}(\hat{\boldsymbol{\beta}}_{S_B})$, $\frac{s_B}{\lambda_B}(\frac{\log p}{n_{XX}})^{\frac{1}{2}-\gamma_1-\gamma_2} = O(1)$, $\frac{1}{\lambda_B}(\frac{\log(p+q)}{n_{XY}})^{\frac{1}{2}-\gamma_2} = O(1)$, $s_BV(\frac{\log p}{n_{XX}})^{\frac{1}{2}-\gamma_2} = O(1)$, with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, it holds that

$$\begin{aligned}
\left\|\hat{\boldsymbol{\beta}}_{S_B}-\boldsymbol{\beta}^*_{S_B}\right\|_\infty &= \left\|\left(\widehat{\boldsymbol{\Gamma}}_{S_BS_B}\right)^{-1}\widehat{\boldsymbol{\gamma}}_{S_B}-\lambda_B\left(\widehat{\boldsymbol{\Gamma}}_{S_BS_B}\right)^{-1}\cdot\operatorname{sign}\left(\hat{\boldsymbol{\beta}}_{S_B}\right)-\boldsymbol{\beta}^*_{S_B}\right\|_\infty \\
&\leq \left\|\left(\widehat{\boldsymbol{\Gamma}}_{S_BS_B}\right)^{-1}\widehat{\boldsymbol{\gamma}}_{S_B}-\boldsymbol{\beta}^*_{S_B}\right\|_\infty+\lambda_B\left\|\left(\widehat{\boldsymbol{\Gamma}}_{S_BS_B}\right)^{-1}\right\|_{L_\infty} \\
&\leq \left(\left\|\widehat{\boldsymbol{\gamma}}_{S_B}-\widehat{\boldsymbol{\Gamma}}_{S_BS_B}\boldsymbol{\beta}^*_{S_B}\right\|_\infty+\lambda_B\right)\cdot\left\|\left(\widehat{\boldsymbol{\Gamma}}_{S_BS_B}\right)^{-1}\right\|_{L_\infty} \\
&\leq \left(\left\|\widehat{\boldsymbol{\gamma}}_{S_B}-\boldsymbol{\gamma}_{S_B}\right\|_\infty+s_B\|\mathbf{B}^*\|_\infty\left\|\widehat{\boldsymbol{\Gamma}}_{S_BS_B}-\boldsymbol{\Gamma}_{S_BS_B}\right\|_\infty+\lambda_B\right) \\
&\quad\frac{V}{1-s_BV\left\|\widehat{\boldsymbol{\Gamma}}_{S_BS_B}-\boldsymbol{\Gamma}_{S_BS_B}\right\|_\infty} \\
&\leq \frac{2\lambda_BV}{1-s_BV\left\|\widehat{\boldsymbol{\Gamma}}_{S_BS_B}-\boldsymbol{\Gamma}_{S_BS_B}\right\|_\infty}\leq 4\lambda_BV<\min_{j\in S_B}|\boldsymbol{\beta}^*_j|,
\end{aligned}$$

for sufficiently large p,q, n_{XX},n_{XY} and n_{YY}. The last step holds because we assume that $\lambda_BV/\min_{j\in S_B}|\boldsymbol{\beta}^*_j| = o(1)$. Thus we have $P(\mathcal{A}_1)\geq 1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$ for sufficiently large p,q, n_{XX},n_{XY} and n_{YY}.

For $\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1} - \mathbf{\Gamma}_{S_B^C S_B}\left(\mathbf{\Gamma}_{S_B S_B}\right)^{-1}\|_{L_\infty}$, we have

$$
\begin{aligned}
&\left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1} - \mathbf{\Gamma}_{S_B^C S_B}\left(\mathbf{\Gamma}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \\
\leq &\left\|\mathbf{\Gamma}_{S_B^C S_B}\left(\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1} - \left(\mathbf{\Gamma}_{S_B S_B}\right)^{-1}\right)\right\|_{L_\infty} + \left\|\left(\widehat{\mathbf{\Gamma}}_{S_B^C S_B} - \mathbf{\Gamma}_{S_B^C S_B}\right)\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \\
\leq &\left\|\mathbf{\Gamma}_{S_B^C S_B}\left(\mathbf{\Gamma}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \cdot \left\|\mathbf{\Gamma}_{S_B S_B} - \widehat{\mathbf{\Gamma}}_{S_B S_B}\right\|_{L_\infty} \cdot \left\|\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \\
&+ \left\|\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \cdot \left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B} - \mathbf{\Gamma}_{S_B^C S_B}\right\|_{L_\infty} \\
\leq &\left\|\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \cdot \left(\left\|\widehat{\mathbf{\Gamma}}_{S_B S_B} - \mathbf{\Gamma}_{S_B S_B}\right\|_{L_\infty} + \left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B} - \mathbf{\Gamma}_{S_B^C S_B}\right\|_{L_\infty}\right) \\
\leq &\frac{2 s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}{1 - s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}.
\end{aligned}
\tag{A.26}
$$

Since $\hat{\boldsymbol{\beta}}_{S_B} = (\widehat{\mathbf{\Gamma}}_{S_B S_B})^{-1}\widehat{\boldsymbol{\gamma}}_{S_B} - \lambda_B(\hat{\mathbf{\Sigma}}_{XX,S_B S_B})^{-1} \cdot \mathrm{sign}(\hat{\boldsymbol{\beta}}_{S_B})$, with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, it holds that

$$
\begin{aligned}
&\left\|\widehat{\boldsymbol{\gamma}}_{S_B^C} - \widehat{\mathbf{\Gamma}}_{S_B^C S_B}\hat{\boldsymbol{\beta}}_{S_B}\right\|_\infty \\
\leq &\left\|\widehat{\boldsymbol{\gamma}}_{S_B^C} - \widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\widehat{\boldsymbol{\gamma}}_{S_B}\right\|_\infty + \lambda_B\left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \\
\leq &\left\|\widehat{\boldsymbol{\gamma}}_{S_B^C} - \boldsymbol{\gamma}_{S_B^C}\right\|_\infty + \left\|\left(\mathbf{\Gamma}_{S_B^C S_B}\left(\mathbf{\Gamma}_{S_B S_B}\right)^{-1} - \widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right)\boldsymbol{\gamma}_{S_B}\right\|_\infty \\
&+ \left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\left(\boldsymbol{\gamma}_{S_B} - \widehat{\boldsymbol{\gamma}}_{S_B}\right)\right\|_\infty + \lambda_B\left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right\|_{L_\infty} \\
\leq &\underbrace{\left\|\widehat{\boldsymbol{\gamma}}_{S_B^C} - \boldsymbol{\gamma}_{S_B^C}\right\|_\infty}_{\text{(I)}} + \underbrace{\left\|\left(\mathbf{\Gamma}_{S_B^C S_B}\left(\mathbf{\Gamma}_{S_B S_B}\right)^{-1} - \widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right)\mathbf{\Gamma}_{S_B S_B}\boldsymbol{\beta}^*_{S_B}\right\|_\infty}_{\text{(II)}} \\
&+ \underbrace{\left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\left(\boldsymbol{\gamma}_{S_B^C} - \widehat{\boldsymbol{\gamma}}_{S_B^C}\right)\right\|_\infty + \lambda_B\left\|\widehat{\mathbf{\Gamma}}_{S_B^C S_B}\left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1}\right\|_{L_\infty}}_{\text{(III)}}.
\end{aligned}
$$

By Condition A.2.3, Condition A.2.5, (A.24), (A.25) and (A.26), with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, it holds that

$$
(I) \lesssim \left(\frac{\log(pq)}{n_{XY}}\right)^{\frac{1}{2} - \gamma_2},
$$

$$
\begin{aligned}
(II) \leq & s_B \|\mathbf{B}^*\|_\infty \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty \left(1 + \left\| \widehat{\mathbf{\Gamma}}_{S_B^C S_B} \left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1} \right\|_{L_\infty}\right) \\
\lesssim & s_B \left(\frac{\log p}{n_{XX}}\right)^{\frac{1}{2}-\gamma_1-\gamma_2} \left(2 - \eta + \frac{2 s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}{1 - s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}\right),
\end{aligned}
$$

$$
\begin{aligned}
(III) \leq & \left(\lambda_B + \left\| \boldsymbol{\gamma}_{S_B^C} - \widehat{\boldsymbol{\gamma}}_{S_B^C} \right\|_\infty\right) \left\| \widehat{\mathbf{\Gamma}}_{S_B^C S_B} \left(\widehat{\mathbf{\Gamma}}_{S_B S_B}\right)^{-1} \right\|_{L_\infty} \\
\lesssim & \left(\lambda_B + \left(\frac{\log(pq)}{n_{XY}}\right)^{\frac{1}{2}-\gamma_2}\right) \left(1 - \eta + \frac{2 s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}{1 - s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}\right).
\end{aligned}
$$

Since $\frac{s_B}{\lambda_B}(\frac{\log p}{n_{XX}})^{\frac{1}{2}-\gamma_1-\gamma_2} = O(1)$, $\frac{1}{\lambda_B}(\frac{\log(p+q)}{n_{XY}})^{\frac{1}{2}-\gamma_2} = O(1)$, and $s_B V(\frac{\log p}{n_{XX}})^{\frac{1}{2}-\gamma_2} = O(1)$, when p, q, n_{XY}, n_{XX} and n_{YY} are sufficiently large, with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, it holds that

$$
\frac{(I)}{\lambda_B} \leq \frac{\eta}{4}, \qquad \frac{(II)}{\lambda_B} \leq \frac{\eta}{4},
$$

$$
\begin{aligned}
\frac{(III)}{\lambda_B} \leq & \frac{1}{\lambda_B} \left(\lambda_B + \left\| \boldsymbol{\gamma}_{S_B^C} - \widehat{\boldsymbol{\gamma}}_{S_B^C} \right\|_\infty\right) \left(1 - \eta + \frac{2 s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}{1 - s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}\right) \\
= & 1 - \eta + \frac{1}{\lambda_B} \frac{2 s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}{1 - s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty} + \frac{1}{\lambda_B} \left\| \boldsymbol{\gamma}_{S_B^C} - \widehat{\boldsymbol{\gamma}}_{S_B^C} \right\|_\infty (1 - \eta + \\
& \left(\frac{2 s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}{1 - s_B V \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_\infty}\right) \\
\leq & 1 - \frac{\eta}{2}.
\end{aligned}
$$

Thus, with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, it holds that

$$
\frac{\left\| \widehat{\boldsymbol{\gamma}}_{S_B^C} - \widehat{\mathbf{\Gamma}}_{S_B^C S_B} \hat{\boldsymbol{\beta}}_{S_B} \right\|_\infty}{\lambda_B} = \frac{(I) + (II) + (III)}{\lambda_B} \leq \frac{\eta}{4} + \frac{\eta}{4} + 1 - \frac{\eta}{2} = 1.
$$

Therefore, $P(\mathcal{A}_2) = P\left(\|\hat{\boldsymbol{\gamma}}^{\mathbf{C}}_{S_B^C} - \hat{\mathbf{\Gamma}}^{\mathbf{C}}_{S_B^C S_B} \hat{\boldsymbol{\beta}}_{S_B}\|_\infty \leq \lambda_B\right) \geq 1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$.

Since $P(\|\hat{\boldsymbol{\beta}}_{S_B} - \boldsymbol{\beta}^*_{S_B}\|_\infty < \min_{j \in S_B} |\boldsymbol{\beta}^*_j|) \geq 1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$ it holds that $|\hat{\boldsymbol{\beta}}_j - \boldsymbol{\beta}^*_j| < |\boldsymbol{\beta}^*_j|$ for $j \in S_B$. Thus, we have $P(\text{sign}(\hat{\boldsymbol{\beta}}_{S_B}) = \text{sign}(\boldsymbol{\beta}^*_{S_B})) \geq 1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$. Let $\hat{\boldsymbol{\beta}} \in \mathbb{R}^{pq}$ which satisfies $\hat{\boldsymbol{\beta}}_{S_B^C} = \mathbf{0}$ and $\hat{\boldsymbol{\beta}}_{S_B} = \hat{\boldsymbol{\beta}}_{S_B}$. Since $P(\|\hat{\boldsymbol{\gamma}}^{\mathbf{C}}_{S_B^C} - \hat{\mathbf{\Gamma}}^{\mathbf{C}}_{S_B^C S_B} \hat{\boldsymbol{\beta}}_{S_B}\|_\infty \leq \lambda_B) \geq$

$1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$, $\hat{\boldsymbol{\beta}}$ satisfies (A.22) with probability at least $1-\frac{4}{p}-\frac{4}{pq}-\frac{4}{q}$. Thus, we have verified that $\operatorname{sign}(\hat{\boldsymbol{\beta}})=\operatorname{sign}(\boldsymbol{\beta}^*)$ with high probability. This completes the proof.

A.9 Supporting lemmas

Lemma A.9.1. *(Lemma 1 from Cai et al. (2013)) Let $\xi_1,\ldots,\xi_n$ be independent random variables with mean zero. Suppose that there exists some $t>0$ and $\bar{B}_n$ such that $\sum_{k=1}^n E\left\{\xi_k^2 e^{t|\xi_k|}\right\}\leqslant \bar{B}_n^2$. Then uniformly for $0<x\leqslant \bar{B}_n$,*

$$\operatorname{pr}\left(\sum_{k=1}^n \xi_k \geqslant C_t \bar{B}_n x\right)\leqslant \exp\left(-x^2\right),$$

where $C_t=t+t^{-1}$.

Lemma A.9.2. *(Lemma 1 from Ravikumar et al. (2011)) Consider a zero-mean random vector $\mathbf{X}=(X_1,\ldots,X_p)^\top$ with covariance $\boldsymbol{\Sigma}=(\sigma_{ij})$ such that $X_j/\sqrt{\sigma_{jj}}$ is sub-Gaussian with parameter L for $1\leq j\leq p$. Let $\{\mathbf{X}_i\}_{i=1}^n$ be i.i.d. samples of $\mathbf{X}$, the sample covariance $\hat{\boldsymbol{\Sigma}}=(\hat{\sigma}_{ij})$ satisfies the tail bound that*

$$P\left(|\hat{\sigma}_{jt}-\sigma_{jt}|\geq\delta\right)\leq 4\exp\left\{-\frac{n\delta^2}{128\left(1+4L^2\right)^2\max_j\left(\sigma_{jj}\right)^2}\right\},$$

for all $\delta\in\left(0,8\max_j\left(\sigma_{jj}\right)\left(1+4L^2\right)\right)$.

Lemma A.9.3. *(Lemma 1 of Negahban et al. (2012)) Suppose that $\mathcal{L}$ is a convex and differentiable function and consider any optimal solution $\widehat{\boldsymbol{\theta}}_{\lambda_n}$ to the following optimization problem*

$$\widehat{\boldsymbol{\theta}}_{\lambda_n}\in\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p}\left\{\mathcal{L}\left(\boldsymbol{\theta};\mathbf{Z}_1^n\right)+\lambda_n\mathcal{R}(\boldsymbol{\theta})\right\},$$

where $\lambda_n>0$ is a constant and $\mathcal{R}:\mathbb{R}^p\to\mathbb{R}_+$ is a decomposable norm. For a given inner product $\langle\cdot,\cdot\rangle$, define the dual norm of $\mathcal{R}$ as

$$\mathcal{R}^*(\mathbf{v}):=\sup_{\mathbf{u}\in\mathbb{R}^p\setminus\{0\}}\frac{\langle\mathbf{u},\mathbf{v}\rangle}{\mathcal{R}(\mathbf{u})}=\sup_{\mathcal{R}(\mathbf{u})\leq 1}\langle\mathbf{u},\mathbf{v}\rangle.$$

If $\lambda_n \geq 2\mathcal{R}^*\left(\nabla\mathcal{L}\left(\boldsymbol{\theta}^*;\mathbf{Z}_1^n\right)\right)$ *and for any pair of sets* $(\mathcal{M},\overline{\mathcal{M}}^{\perp})$ *over which* $\mathcal{R}$ *is decomposable, the error* $\widehat{\boldsymbol{\Delta}} = \widehat{\boldsymbol{\theta}}_{\lambda_n} - \boldsymbol{\theta}^*$ *belongs to the set*

$$S\left(\mathcal{M},\overline{\mathcal{M}}^{\perp};\boldsymbol{\theta}^*\right) := \left\{\boldsymbol{\Delta}\in\mathbb{R}^p \mid \mathcal{R}\left(\boldsymbol{\Delta}_{\overline{\mathcal{M}}^{\perp}}\right) \leq 3\mathcal{R}\left(\boldsymbol{\Delta}_{\overline{\mathcal{M}}}\right) + 4\mathcal{R}\left(\boldsymbol{\theta}^*_{\mathcal{M}^{\perp}}\right)\right\}.$$

Lemma A.9.4. *Under assumptions of Theorem 2.3.2,* $\boldsymbol{\Delta}^B = \hat{\mathbf{B}} - \mathbf{B}^*$ *belongs to the set*

$$\mathcal{C}_B := \left\{\boldsymbol{\Delta}^B \in \mathbb{R}^{p\times q} \,\middle|\, \left\|\boldsymbol{\Delta}^B_{S_B^C}\right\|_1 \leq 3\left\|\boldsymbol{\Delta}^B_{S_B}\right\|_1\right\}, \tag{A.27}$$

and $\boldsymbol{\Delta}^C = \hat{\mathbf{C}} - \mathbf{C}^*$ *belongs to the set*

$$\mathcal{C}_C := \left\{\boldsymbol{\Delta}^C \in \mathbb{R}^{q\times q} \,\middle|\, \left\|\boldsymbol{\Delta}^C_{S_C^C}\right\|_1 \leq 3\left\|\boldsymbol{\Delta}^C_{S_C}\right\|_1\right\}. \tag{A.28}$$

Proof of Lemma A.9.4. Since $\mathbf{Y} = \mathbf{X}\mathbf{B}^* + \mathcal{E}$, we have

$$\begin{aligned}
\boldsymbol{\Sigma}_{YY} &= \mathrm{Cov}(\mathbf{Y},\mathbf{Y}) = \mathrm{Cov}(\mathbf{X}\mathbf{B}^* + \mathbf{E}, \mathbf{X}\mathbf{B}^* + \mathcal{E}) = \mathbf{B}^{*\top}\boldsymbol{\Sigma}_{XX}\mathbf{B}^* + \mathrm{Cov}(\mathcal{E},\mathcal{E}) \\
&= \mathbf{B}^{*\top}\boldsymbol{\Sigma}_{XX}\mathbf{B}^* + \mathbf{C}^{*-1}, \\
\boldsymbol{\Sigma}_{XY} &= \mathrm{Cov}(\mathbf{X},\mathbf{Y}) = \mathrm{Cov}(\mathbf{X},\mathbf{X}\mathbf{B}^* + \mathcal{E}) = \boldsymbol{\Sigma}_{XX}\mathbf{B}^*.
\end{aligned}$$

Thus, by Theorem 2.3.1 and Condition A.2.3, it holds with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$ that

$$\begin{aligned}
&\|\nabla_B\mathcal{L}_n(\mathbf{B}^*,\mathbf{C}^*)\|_\infty \\
=&\left\|2\mathbf{C}^{*\top}\mathbf{B}^{*\top}\hat{\boldsymbol{\Sigma}}_{XX} - 2\mathbf{C}^{*\top}\hat{\boldsymbol{\Sigma}}_{XY}^\top\right\|_\infty \\
=&\left\|2\mathbf{C}^{*\top}\mathbf{B}^{*\top}\boldsymbol{\Delta}^{XX} - 2\mathbf{C}^{*\top}\boldsymbol{\Delta}^{XY\top}\right\|_\infty \\
\leq& 2\|\mathbf{C}^*\|_{L_1}(\|\mathbf{B}^*\|_{L_1}\|\boldsymbol{\Delta}^{XX}\|_\infty + \|\boldsymbol{\Delta}^{XY}\|_\infty) \\
\lesssim& \max\left\{\left(\frac{\log p}{n_{XX}}\right)^{\frac{1}{2}-\gamma_1-\gamma_2}, \left(\frac{\log(pq)}{n_{XY}}\right)^{\frac{1}{2}-\gamma_2}\right\} \\
\lesssim& \lambda_B,
\end{aligned}$$

and

$$
\begin{aligned}
&\|\nabla_C \mathcal{L}_n(\mathbf{B}^*, \mathbf{C}^*)\|_\infty \\
=&\left\|\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^* + \hat{\mathbf{\Sigma}}_{YY} - \mathbf{C}^{*-1} - 2\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{XY}\right\|_\infty \\
=&\left\|\mathbf{B}^{*\top}\mathbf{\Delta}^{XX}\mathbf{B}^* + \mathbf{\Delta}^{YY} - 2\mathbf{B}^{*\top}\mathbf{\Delta}^{XY}\right\|_\infty \\
\leq&\|\mathbf{B}^*\|_{L_1}^2\|\mathbf{\Delta}^{XX}\|_\infty + \|\mathbf{\Delta}^{YY}\|_\infty + 2\|\mathbf{B}^*\|_{L_1}\|\mathbf{\Delta}^{XY}\|_\infty \\
\lesssim&\max\left\{\left(\frac{\log p}{n_{XX}}\right)^{\frac{1}{2}-2\gamma_1}, \left(\frac{\log(pq)}{n_{XY}}\right)^{\frac{1}{2}-\gamma_1}, \left(\frac{\log q}{n_{YY}}\right)^{\frac{1}{2}}\right\} \\
\lesssim&\lambda_C.
\end{aligned}
$$

Since L_1 penalty is decomposable, by applying Lemma A.9.3, we have

$$
\left\|\mathbf{\Delta}^B_{S_B^C}\right\|_1 \leq 3\|\mathbf{\Delta}^B_{S_B}\|_1 + 4\|\mathbf{B}^*_{S_B^C}\|_1 = 3\|\mathbf{\Delta}^B_{S_B}\|_1,
$$

$$
\left\|\mathbf{\Delta}^C_{S_C^C}\right\|_1 \leq 3\|\mathbf{\Delta}^C_{S_C}\|_1 + 4\|\mathbf{C}^*_{S_C^C}\|_1 = 3\|\mathbf{\Delta}^B_{S_C}\|_1.
$$

□

Theorem A.9.1. *(Theorem 1 of Loh and Wainwright (2015)) Consider the optimization problem*

$$
\tilde{\boldsymbol{\beta}} = \arg\min_{\|\boldsymbol{\beta}\|_1 \leq R, \boldsymbol{\beta}\in\Omega} \mathcal{L}_n(\boldsymbol{\beta}) + \lambda\|\boldsymbol{\beta}\|_1,
$$

where Ω is some convex set and the empirical loss $\mathcal{L}_n$ satisfies the RSC conditions

$$
\langle \nabla\mathcal{L}_n(\boldsymbol{\beta}^* + \mathbf{\Delta}) - \nabla\mathcal{L}_n(\boldsymbol{\beta}^*), \mathbf{\Delta}\rangle \geq \begin{cases} \alpha_1\|\mathbf{\Delta}\|_2^2 - \tau_1\dfrac{\log p}{n}\|\mathbf{\Delta}\|_1^2, & \forall\|\mathbf{\Delta}\|_2 \leq 1; \quad \text{(A.29)} \\ \alpha_2\|\mathbf{\Delta}\|_2 - \tau_2\sqrt{\dfrac{\log p}{n}}\|\mathbf{\Delta}\|_1, & \forall\|\mathbf{\Delta}\|_2 \geq 1; \quad \text{(A.30)} \end{cases}
$$

α_1, α_2 are positive constants and τ_1, τ_2 are non-negative constants. Suppose $n \geq \frac{16R^2\max(\tau_1^2,\tau_2^2)}{\alpha_2^2}\log p$, $\|\boldsymbol{\beta}^*\|_1 \leq R$ and

$$
\frac{4}{L}\max\left\{\|\nabla\mathcal{L}_n(\boldsymbol{\beta}^*)\|_\infty, \alpha_2\sqrt{\frac{\log p}{n}}\right\} \leq \lambda \leq \frac{\alpha_2}{6RL},
$$

where L is a constant. Then for any vector $\tilde{\boldsymbol{\beta}}$ with $\|\tilde{\boldsymbol{\beta}}\|_1 \leq R$ and satisfies the first-order necessary condition

$$\left\langle \nabla \mathcal{L}_n(\widetilde{\boldsymbol{\beta}}) + \nabla \|\tilde{\boldsymbol{\beta}}\|_1, \boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}} \right\rangle \geq 0, \quad \textit{for all } \|\boldsymbol{\beta}\|_1 \leq 1,$$

it holds that

$$\left\|\widetilde{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\right\|_2 \leq \frac{6\lambda\sqrt{k}}{4\alpha_1}, \quad \textit{and} \quad \left\|\widetilde{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\right\|_1 \leq \frac{24\lambda k}{4\alpha_1},$$

where $k = \|\boldsymbol{\beta}^*\|_0$.

Lemma A.9.5. *Let $n_{XX/Y} \asymp n_{XX}^{\tau_1}$ and $n_{XY/X} \asymp n_{XY}^{\tau_2}$ with $\tau_1, \tau_2 \in \{-\infty\} \cup [0,1]$, $\boldsymbol{\delta}_C = \mathrm{vec}(\mathbf{C}^* - \hat{\mathbf{C}})$, $1 - \alpha_1 = O(\sqrt{\log p / n_X})$, $1 - \alpha_2 = O(\sqrt{\log p / n_{XX}})$ and $1 - \alpha_3 = O(\sqrt{\log(pq)/n_{XY}})$. With probability at least $1 - \frac{4}{p} - \frac{4}{pq}$, we have*

$$\begin{aligned}
&\left\|\nabla_B \left\{\mathrm{tr}[\hat{\mathbf{C}}\hat{\boldsymbol{\Sigma}}_{\mathbf{YY}} + \mathbf{B}^*\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\boldsymbol{\Sigma}}_{\mathbf{XX}} - \mathbf{2}\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\boldsymbol{\Sigma}}_{\mathbf{XY}}] - \log\det(\hat{\mathbf{C}})\right\}\right\|_\infty \\
\lesssim &\frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)}(\|\mathbf{B}^*\mathbf{C}^*\|_{L_1} + \|\mathbf{B}^*\|_{L_1}\|\boldsymbol{\delta}_C\|_1) \\
&+ \max\{\lambda_{\max}(\mathbf{C}^*), 1/\lambda_{\min}(\mathbf{C}^*)\}\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}(1 + \|\boldsymbol{\delta}_C\|_1).
\end{aligned}$$

Proof. Denote $\boldsymbol{\Delta}^{YY} = \boldsymbol{\Sigma}_{YY} - \hat{\boldsymbol{\Sigma}}_{YY}$, Theorem 2.3.1 implies that with probability at least $1 - \frac{4}{p} - \frac{4}{pq} - \frac{4}{q}$, we have

$$\|\boldsymbol{\Delta}^{XX}\|_\infty \leq v_1' \sqrt{\frac{\log p}{n_{XX}}}; \tag{A.31}$$

$$\|\boldsymbol{\Delta}^{XY}\|_\infty \leq v_2' \sqrt{\frac{\log(pq)}{n_{XY}}}; \tag{A.32}$$

$$\|\boldsymbol{\Delta}^{YY}\|_\infty \leq v_3' \sqrt{\frac{\log q}{n_{YY}}}. \tag{A.33}$$

Define $\tilde{y}_{ij}$ and $\tilde{x}_{ij}$ to be the underlying complete data without missing entries. Define the observed-data indicator matrix as $M^X = (m_{ij}^X)$ and $M^Y = (m_{ij}^Y)$ such that $m_{ij}^X = 1$ when x_{ij} is observed, $m_{ij}^X = 0$ when x_{ij} is missing, $m_{ij}^Y = 1$ when y_{ij} is observed and $m_{ij}^Y = 0$ when y_{ij} is missing. Then we can write the observed data as $x_{ij} = m_{ij}^X \tilde{x}_{ij}$, $y_{ij} = m_{ij}^Y \tilde{y}_{ij}$. Define α_{ij}^{XX} to be the adjusting

weight we use to estimate $(\hat{\mathbf{\Sigma}}_{XX})_{ij}$, that is

$$\alpha_{ij}^{XX} = \begin{cases} 1 & \text{if } i = j; \\ \alpha_1 & \text{if } i \neq j, i \text{ and } j \text{ are in the same modality;} \\ \alpha_2 & \text{if } i \text{ and } j \text{ are in different modalities.} \end{cases}$$

Then we have

$$\begin{aligned} &\left\| \nabla_B \left\{ \mathrm{tr}[\hat{\mathbf{C}}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{B}^*\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XX}} - \mathbf{2}\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XY}}] - \log\det(\hat{\mathbf{C}}) \right\} \right\|_\infty \\ =&\left\| 2\hat{\mathbf{C}}^\top\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{XX} - 2\hat{\mathbf{C}}^\top\hat{\mathbf{\Sigma}}_{XY}^\top \right\|_\infty . \end{aligned} \tag{A.34}$$

When either $\mathbf{Y}$ or $\mathbf{X}$ has missing entries, we have

$$\begin{aligned} &(\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^* - \hat{\mathbf{\Sigma}}_{XY})_{ij} \\ =&\sum_{l=1}^{p} \frac{\alpha_{il}^{XX} \sum_{k\in S_{il}^{XX}} x_{ki}x_{kl}}{n_{il}^{XX}} \mathbf{B}_{lj}^* - \frac{\alpha_3 \sum_{k\in S_{ij}^{XY}}^{n} x_{ki} m_{kj}^Y \tilde{y}_{kj}}{n_{ij}^{XY}} \\ =&(\hat{\mathbf{\Sigma}}_{XX}\mathbf{B}^* - \hat{\mathbf{\Sigma}}_{X\tilde{X}}\mathbf{B}^*)_{ij} - (\hat{\mathbf{\Sigma}}_{X\epsilon})_{ij}, \end{aligned}$$

where $(\hat{\mathbf{\Sigma}}_{X\tilde{X}})_{ij} = \alpha_3 \sum_{k\in S_{ij}^{XY}} x_{ki}\tilde{x}_{kj}/n_{ij}^{XY}$, and $(\hat{\mathbf{\Sigma}}_{X\epsilon})_{ij} = \alpha_3 \sum_{k\in S_{ij}^{XY}} x_{ki}\epsilon_{kj}/n_{ij}^{XY}$. Then by (A.34) we have

$$\begin{aligned} &\left\| \nabla_B \left\{ \mathrm{tr}[\hat{\mathbf{C}}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{B}^*\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XX}} - \mathbf{2}\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XY}}] - \log\det(\hat{\mathbf{C}}) \right\} \right\|_\infty \\ =&2\left\| (\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})\mathbf{B}^*\hat{\mathbf{C}} - \hat{\mathbf{\Sigma}}_{X\epsilon}\hat{\mathbf{C}} \right\|_\infty \\ \leq&2\left\| (\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})\mathbf{B}^*\mathbf{C}^* \right\|_\infty + 2\left\| (\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})\mathbf{B}^*(\mathbf{C}^* - \hat{\mathbf{C}}) \right\|_\infty \\ &+2\left\| \hat{\mathbf{\Sigma}}_{X\epsilon}\mathbf{C}^* \right\|_\infty + 2\left\| \hat{\mathbf{\Sigma}}_{X\epsilon}(\mathbf{C}^* - \hat{\mathbf{C}}) \right\|_\infty . \end{aligned}$$

We first derive an upper bound for the first term $\|(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})\mathbf{B}^*\mathbf{C}^*\|_\infty$. Define $S_{jkl}^{XXY} = \{i : x_{ij}, x_{ik} \text{ and } y_{il} \text{ are not missing}\}$ and $n_{jkl}^{XXY} = |S_{jkl}^{XXY}|$.

When $n_{ijl}^{XX/Y} \neq 0$ and $n_{ilj}^{XY/X} \neq 0$, for each entry in matrix $\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}}$ and $1 \leq l \leq q$, with probability at least $1 - \frac{4}{p^3}$, we have

$$
\begin{aligned}
&(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})_{ij} \\
=&\frac{1}{n_{ijl}^{XXY}} \sum_{k \in S_{ijl}^{XXY}} X_{ik}X_{jk} \frac{n_{ijl}^{XXY}(\alpha_{ij}^{XX} n_{ij}^{XX} - \alpha_3 n_{il}^{XY})}{n_{ij}^{XX} n_{il}^{XY}} \\
&+ \frac{1}{n_{ijl}^{XX/Y}} \sum_{k \in S_{ijl}^{XX/Y}} X_{ik}X_{jk} \frac{\alpha_{ij}^{XX} n_{ijl}^{XX/Y}}{n_{ij}^{XX}} - \frac{1}{n_{ilj}^{XY/X}} \sum_{k \in S_{ilj}^{XY/X}} X_{ik}X_{jk} \frac{\alpha_3 n_{ilj}^{XY/X}}{n_{il}^{XY}} \\
\leq& \left(\frac{1}{n_{ijl}^{XXY}} \sum_{k \in S_{ijk}^{XXY}} X_{ik}X_{jk} - \mathbf{\Sigma}_{XX,ij} \right) \frac{n_{ijl}^{XXY}(\alpha_{ij}^{XX} n_{ij}^{XX} - \alpha_3 n_{il}^{XY})}{n_{ij}^{XX} n_{il}^{XY}} \\
&+ \left(\frac{1}{n_{ijl}^{XX/Y}} \sum_{k \in S_{ijl}^{XX/Y}} X_{ik}X_{jk} - \mathbf{\Sigma}_{XX,ij} \right) \frac{\alpha_{ij}^{XX} n_{ijl}^{XX/Y}}{n_{ij}^{XX}} \\
&- \left(\frac{1}{n_{ilj}^{XY/X}} \sum_{k \in S_{ilj}^{XY/X}} X_{ik}X_{jk} - \mathbf{\Sigma}_{XX,ij} \right) \frac{\alpha_3 n_{ilj}^{XY/X}}{n_{il}^{XY}} + 2(1 - \alpha_{ij}^{XX}) + 2(1 - \alpha_3)
\end{aligned}
$$

$$
\begin{aligned}
\lesssim& \sqrt{\frac{\log p}{n_{ijl}^{XXY}}} \frac{n_{ijl}^{XXY}(n_{ij}^{XX} - n_{il}^{XY})}{n_{ij}^{XX} n_{il}^{XY}} + \sqrt{\frac{\log p}{n_{ijl}^{XX/Y}}} \frac{n_{ijl}^{XX/Y}}{n_{ij}^{XX}} + \sqrt{\frac{\log p}{n_{ilj}^{XY/X}}} \frac{n_{ilj}^{XY/X}}{n_{il}^{XY}} + \alpha_{ij}^{XX} - \alpha_3 \\
\lesssim& \sqrt{\log p} \max \left[\max_{ijl} \left(\frac{(n_{ijl}^{XX/Y})^{1/2}}{n_{ij}^{XX}} \right), \max_{ijl} \left(\frac{(n_{ilj}^{XY/X})^{1/2}}{n_{il}^{XY}} \right) \right] - (1 - \alpha_{ij}^{XX}) + (1 - \alpha_3) \\
\lesssim& \frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2} \right)},
\end{aligned}
$$

where $\tau_1, \tau_2 \in [0, 1]$.

When $n_{ijl}^{XX/Y} = 0$ and $n_{ilj}^{XY/X} \neq 0$, for each entry in matrix $\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}}$ and $1 \leq l \leq q$, with probability at least $1 - \frac{4}{p^3}$, we have

$$
\begin{aligned}
&(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})_{ij} \\
=&\frac{1}{n_{ijl}^{XXY}} \sum_{k \in S_{ijl}^{XXY}} X_{ik} X_{jk} \frac{n_{ijk}^{XXY} (\alpha_{ij}^{XX} n_{ij}^{XX} - \alpha_3 n_{il}^{XY})}{n_{ij}^{XX} n_{il}^{XY}} \\
&- \frac{1}{n_{ilj}^{XY/X}} \sum_{k \in S_{ilj}^{XY/X}} X_{ik} X_{jk} \frac{\alpha_3 n_{ilj}^{XY/X}}{n_{il}^{XY}} \\
\lesssim&\frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)},
\end{aligned}
$$

where $\tau_2 \in [0,1]$, $\tau_1 \in \{-\infty\} \cup [0,1]$.

When $n_{ijl}^{XX/Y} \neq 0$ and $n_{ijl}^{XY/X} = 0$, for each entry in matrix $\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}}$ and $1 \leq l \leq q$, with probability at least $1 - \frac{4}{p^3}$, we have

$$
\begin{aligned}
&(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})_{ij} \\
=&\frac{1}{n_{ijl}^{XXY}} \sum_{k \in S_{ijl}^{XXY}} X_{ik} X_{jk} \frac{n_{ijk}^{XXY} (\alpha_{ij}^{XX} n_{ij}^{XX} - \alpha_3 n_{il}^{XY})}{n_{ij}^{XX} n_{il}^{XY}} \\
&+ \frac{1}{n_{ijl}^{XX/Y}} \sum_{k \in S_{ijl}^{XX/Y}} X_{ik} X_{jk} \frac{\alpha_{ij}^{XX} n_{ijl}^{XX/Y}}{n_{ij}^{XX}} \\
\lesssim&\frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)},
\end{aligned}
$$

where $\tau_1 \in [0,1]$, $\tau_2 \in \{-\infty\} \cup [0,1]$.

When $n_{ijl}^{XX/Y} = n_{ijl}^{XY/X} = 0$, $n_{ijl}^{XXY} = n_{ij}^{XX} = n_{il}^{XY}$. Then for each entry in matrix $\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}}$ and $1 \le l \le q$, with probability at least $1 - \frac{4}{p^3}$, we have

$$
\begin{aligned}
&(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})_{ij} \\
=&\frac{(\alpha_{ij}^{XX} - \alpha_3)}{n_{ijl}^{XXY}} \sum_{k \in S_{ijl}^{XXY}} X_{ik} X_{jk} \\
\lesssim& \max(1-\alpha_1, 1-\alpha_2, 1-\alpha_3) \sqrt{\frac{\log p}{n_{ijl}^{XXY}}} \\
\lesssim& \frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)},
\end{aligned}
$$

where $\tau_1, \tau_2 \in \{-\infty\} \cup [0,1]$.

If we combine the above four cases, for each entry in matrix $\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}}$ and $1 \le l \le q$, with probability at least $1 - \frac{4}{p^3}$, we have

$$
(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})_{ij} \lesssim \frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)},
$$

where $\tau_1, \tau_2 \in \{-\infty\} \cup [0,1]$.

Then by Holder's inequality and the union bound, with probability at least $1 - 4/p$ we have

$$
\|(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})\mathbf{B}^*\mathbf{C}^*\|_\infty \lesssim \frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)} \|\mathbf{B}^*\mathbf{C}^*\|_{L_1}. \tag{A.35}
$$

Similarly, for the second term $\left\|(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})\mathbf{B}^*(\mathbf{C}^* - \hat{\mathbf{C}})\right\|_\infty$, with probability at least $1 - 4/p$ we have

$$
\|(\hat{\mathbf{\Sigma}}_{XX} - \hat{\mathbf{\Sigma}}_{X\tilde{X}})\mathbf{B}^*(\mathbf{C}^* - \hat{\mathbf{C}})\|_\infty \lesssim \frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)} \|\mathbf{B}^*\|_{L_1} \|\boldsymbol{\delta}_C\|_1. \tag{A.36}
$$

Next we focus on the third term $\|\hat{\mathbf{\Sigma}}_{X\epsilon}\mathbf{C}^*\|_\infty$. Each entry of the matrix $(\hat{\mathbf{\Sigma}}_{X\epsilon}\mathbf{C}^*)_{ij}$ can be written as $\frac{\alpha_3}{n_{ij}^{XY}} \sum_{k \in S_{ij}^{XY}} X_{ki}(\boldsymbol{\epsilon}_k \mathbf{C}^*)_j$. By Condition A.2.1 and monotone convergence theorem, for any $t \in \mathbb{R}$, we have

$$
\mathbb{E}\left[\exp\left(\frac{X_{ki}^2}{8L_1^2}\right)\right] = \mathbb{E}\left[\sum_{l=0}^{\infty} \frac{X_{ki}^{2l}}{\left(4L_1^2\right)^l l!} \frac{1}{2^l}\right] \le \sum_{l=0}^{\infty} \frac{1}{2^l} = 2.
$$

By Condition A.2.1, the error vectors also follow sub-Gaussian distribution. Assume $\mathbb{E}(\exp(t\mathbf{u}_2^\top \boldsymbol{\epsilon}_i)) \leq \exp\left(\frac{L_3^2\|\mathbf{u}_2\|_2^2 t^2}{2}\right)$. Then we have

$$\mathbb{E}\left[\exp\left(\frac{(\boldsymbol{\epsilon}_k \mathbf{C}^*)_j^2}{8L_3^2\|\mathbf{C}_j^*\|_2^2}\right)\right] \leq 2.$$

By Young's inequality and the simple inequality $s^2 e^s \leq e^{2s}$ for $s > 0$, we have

$$\begin{aligned}
&\mathbb{E}\left[(X_{ki}(\boldsymbol{\epsilon}_k \mathbf{C}^*)_j|)^2 \exp\left(\frac{|X_{ki}(\boldsymbol{\epsilon}_k \mathbf{C}^*)_j|}{8L_1L_3\lambda_{\max}(\mathbf{C}^*)}\right)\right] \\
\leq &\mathbb{E}\left[\exp\left(\frac{|X_{ki}(\boldsymbol{\epsilon}_k \mathbf{C}^*)_j|}{4L_1L_3\lambda_{\max}(\mathbf{C}^*)}\right)\right] \\
\leq &\mathbb{E}\left[\exp\left(\frac{X_{ki}^2}{8L_1^2}\right)\exp\left(\frac{(\boldsymbol{\epsilon}_k \mathbf{C}^*)_j^2}{8L_3^2\|\mathbf{C}^*\|_2^2}\right)\right] \\
\leq &\frac{1}{2}\left[\mathbb{E}\exp\left(\frac{X_{ki}^2}{8L_1^2}\right)\right]^2 + \frac{1}{2}\left[\mathbb{E}\exp\left(\frac{(\boldsymbol{\epsilon}_k \mathbf{C}_j^*)^2}{8L_3^2\|\mathbf{C}^*\|_2^2}\right)\right]^2 \\
\leq &4.
\end{aligned}$$

By Lemma A.9.1, let $\bar{B}_n = 2\sqrt{n_{XY}}$, $t = \frac{1}{8L_1L_3\lambda_{\max}(\mathbf{C}^*)}$ and $x = \sqrt{2\log(pq)}$, we have

$$\max_{i,j}\mathbb{P}\left[\left|\frac{1}{n_{ij}^{XY}}\sum_{k\in S_{ij}^{XY}}\left(X_{ki}\left(\epsilon_k^\top \mathbf{C}^*\right)_j\right)\right| \geq C_1\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}\right] \leq 2(pq)^{-2}. \tag{A.37}$$

where $C_1 = \frac{\sqrt{2}}{8L_1L_3\lambda_{\max}(\mathbf{C}^*)} + 8\sqrt{2}L_1L_3\lambda_{\max}(\mathbf{C}^*)$. So with probability at least $1 - 2(pq)^{-1}$, we can bound the third term by

$$\|\hat{\boldsymbol{\Sigma}}_{X\epsilon}\mathbf{C}^*\|_\infty \lesssim C_1\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}. \tag{A.38}$$

Similarly, for the last term $\left\|\hat{\boldsymbol{\Sigma}}_{X\epsilon}(\mathbf{C}^* - \hat{\mathbf{C}})\right\|_\infty$, we have

$$\left\|\hat{\boldsymbol{\Sigma}}_{X\epsilon}(\mathbf{C}^* - \hat{\mathbf{C}})\right\|_\infty \lesssim C_2\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}\|\boldsymbol{\delta}_C\|_1, \tag{A.39}$$

where $C_2 = \frac{\sqrt{2}}{8L_1L_3\lambda_{\min}(\mathbf{C}^*)^{-1}} + 8\sqrt{2}L_1L_3\lambda_{\min}(\mathbf{C}^*)^{-1}$.

By (A.35), (A.36), (A.38), (A.39), with probability at least $1 - \frac{4}{p} - \frac{4}{pq}$ we have

$$\begin{aligned}
&\left\|\nabla_B \left\{\mathrm{tr}[\hat{\mathbf{C}}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{B}^*\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XX}} - \mathbf{2}\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XY}}] - \log\det(\hat{\mathbf{C}})\right\}\right\|_\infty \\
\lesssim &\frac{(\log p)^{1/2}}{\min\left(n_{XX}^{1-\tau_1/2}, n_{XY}^{1-\tau_2/2}\right)}(\|\mathbf{B}^*\mathbf{C}^*\|_{L_1} + \|\mathbf{B}^*\|_{L_1}\|\boldsymbol{\delta}_C\|_1) \\
&+ \max\{\lambda_{\max}(\mathbf{C}^*), 1/\lambda_{\min}(\mathbf{C}^*)\}\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}(1 + \|\boldsymbol{\delta}_C\|_1).
\end{aligned}$$

We remark that, when both $\mathbf{X}$ and $\mathbf{Y}$ are complete, we can set $\alpha_1 = \alpha_2 = \alpha_3 = 1$. Then by (A.34) we have

$$\begin{aligned}
&\left\|\nabla_B \left\{\mathrm{tr}[\hat{\mathbf{C}}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{B}^*\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XX}} - \mathbf{2}\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XY}}] - \log\det(\hat{\mathbf{C}})\right\}\right\|_\infty \\
\leq &\left\|2\mathbf{C}^{*\top}\tilde{\mathbf{\Sigma}}_{X\epsilon}\right\|_\infty + \left\|2(\mathbf{C}^* - \hat{\mathbf{C}})^\top\tilde{\mathbf{\Sigma}}_{X\epsilon}\right\|_\infty,
\end{aligned}$$

where $(\tilde{\mathbf{\Sigma}}_{X\epsilon})_{ij} = \sum_{k=1}^n x_{ki}\epsilon_{kj}/n$. By (A.37), with probability at least $1 - 2(pq)^{-1}$, we have

$$\|\mathbf{C}^{*\top}\tilde{\mathbf{\Sigma}}_{X\epsilon}\|_\infty \lesssim C_1\left\{\frac{\log(pq)}{n}\right\}^{1/2},$$

and

$$\left\|2(\mathbf{C}^* - \hat{\mathbf{C}})^\top\tilde{\mathbf{\Sigma}}_{X\epsilon}\right\|_\infty \lesssim C_2\left\{\frac{\log(pq)}{n}\right\}^{1/2}\|\boldsymbol{\delta}_C\|.$$

Hence with probability at least $1 - \frac{4}{p} - \frac{4}{pq}$ we also have

$$\begin{aligned}
&\left\|\nabla_B \left\{\mathrm{tr}[\hat{\mathbf{C}}\hat{\mathbf{\Sigma}}_{\mathbf{YY}} + \mathbf{B}^*\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XX}} - \mathbf{2}\hat{\mathbf{C}}\mathbf{B}^{*\top}\hat{\mathbf{\Sigma}}_{\mathbf{XY}}] - \log\det(\hat{\mathbf{C}})\right\}\right\|_\infty \\
\lesssim &\max\{\lambda_{\max}(\mathbf{C}^*), 1/\lambda_{\min}(\mathbf{C}^*)\}\left\{\frac{\log(pq)}{n_{XY}}\right\}^{1/2}(1 + \|\boldsymbol{\delta}_C\|_1),
\end{aligned}$$

which is the same as stated in Lemma A.9.5 if we set $\tau_1 = \tau_2 = -\infty$ as both $\mathbf{X}$ and $\mathbf{Y}$ are complete. □

A.10 Numerical study

In this section, we show some additional results of our numerical studies. The complete results for Example 1 are shown in Table A.3. The results for Example 2 are shown in Table A.4. The results for Example 3 are shown in Table A.5.

A.11 Data processing details in the ADNI study

In Section 2.5, we are interested in predicting Mini-Mental State Examination (MMSE), ADAS1 and ADAS2 in the Alzheimer's Disease Neuroimaging Initiative (ADNI) study (Mueller et al., 2005). These scores are commonly used diagnotic scores of AD. We extract biomarkers from three complementary data sources: serial magnetic resonance imaging (MRI), positron emission tomography (PET) and CerebroSpinal Fluid (CSF). Note that, as Xue and Qu (2021) stated, our sparsity assumption of the proposed method might not be suitable for raw imaging data or imaging data at small scales since images have to show some visible atrophy for AD. However, the sparsity assumption can still be reasonable for the region of interest (ROI) level data. Thus, we apply the Multi-DISCOM to the ROI level data in ADNI instead of the raw data.

We process the image data following the similar procedure as Yu et al. (2020). For the MRI, after correction, spatial segmentation and registration steps, we obtain the image for each subject based on the Jacob template with 93 manually labeled ROIs. For each of the 93 ROIs in the labeled MRI, we compute the volume of gray matter as a feature. For each PET image, we first align the PET image to its respective MRI using affine registration. Then, we calculate the average intensity of every ROI in the PET image as a feature. For the CSF modality, five biomarkers were used in this study, namely amyloid $\beta(A\beta 42)$, CSF total tau (t-tau), tau hyperphosphorylated at threonine 181 (p-tau), and two tau ratios with respective to $A\beta 42$ (i.e., t-tau/$A\beta 42$ and p-tau /$A\beta 42$).

	Method	$\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F$	MSE	FPR	FNR
$\rho = -0.4$	Lasso	1.51(0.06)	3.70(0.06)	0.09(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.73(0.06)	3.57(0.06)	0.11(0.01)	**0.00(0.00)**
	MBI	2.10(0.08)	4.26(0.09)	0.12(0.02)	0.11(0.03)
	DISCOM	1.44(0.04)	3.56(0.06)	0.05(0.00)	0.05(0.01)
	Imputed-MRCE	1.53(0.05)	3.72(0.08)	0.17(0.03)	0.08(0.02)
	Multi-DISCOM	**1.40(0.04)**	**3.39(0.08)**	**0.02(0.01)**	0.09(0.02)
$\rho = -0.2$	Lasso	1.50(0.06)	3.73(0.06)	0.10(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.71(0.06)	3.59(0.06)	0.11(0.01)	**0.00(0.00)**
	MBI	2.15(0.08)	4.25(0.09)	0.12(0.02)	0.11(0.03)
	DISCOM	1.43(0.04)	3.52(0.06)	0.05(0.00)	0.05(0.01)
	Imputed-MRCE	1.52(0.05)	3.78(0.08)	0.16(0.03)	0.07(0.02)
	Multi-DISCOM	**1.41(0.04)**	**3.40(0.08)**	**0.02(0.01)**	0.09(0.02)
$\rho = 0$	Lasso	1.49(0.06)	3.67(0.06)	0.08(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.71(0.06)	3.55(0.06)	0.10(0.01)	**0.00(0.00)**
	MBI	2.05(0.08)	4.21(0.09)	0.10(0.02)	0.09(0.03)
	DISCOM	1.42(0.04)	3.53(0.06)	0.04(0.00)	0.05(0.01)
	Imputed-MRCE	1.51(0.05)	3.70(0.08)	0.15(0.03)	0.09(0.02)
	Multi-DISCOM	**1.42(0.04)**	**3.43(0.08)**	**0.03(0.01)**	0.10(0.02)
$\rho = 0.2$	Lasso	1.54(0.06)	3.75(0.06)	0.10(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.74(0.06)	3.59(0.06)	0.13(0.01)	**0.00(0.00)**
	MBI	2.10(0.08)	4.29(0.09)	0.11(0.02)	0.10(0.03)
	DISCOM	1.43(0.04)	3.57(0.06)	0.05(0.00)	0.05(0.01)
	Imputed-MRCE	1.53(0.05)	3.73(0.08)	0.19(0.03)	0.08(0.02)
	Multi-DISCOM	**1.41(0.04)**	**3.42(0.08)**	**0.04(0.01)**	0.07(0.02)
$\rho = 0.4$	Lasso	1.55(0.06)	3.77(0.06)	0.11(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.75(0.06)	3.61(0.06)	0.13(0.01)	**0.00(0.00)**
	MBI	2.14(0.08)	4.30(0.09)	0.13(0.02)	0.11(0.03)
	DISCOM	1.46(0.04)	3.59(0.06)	0.06(0.00)	0.05(0.01)
	Imputed-MRCE	1.54(0.05)	3.73(0.08)	0.19(0.03)	0.09(0.02)
	Multi-DISCOM	**1.43(0.04)**	**3.44(0.08)**	**0.04(0.01)**	0.07(0.02)

Table A.3: Performance comparison of different methods for Example 1 with different ρ's. The values in the parentheses are the standard errors of the measures.

	Method	$\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F$	MSE	FPR	FNR
$\alpha = 1$	Lasso	1.33(0.08)	2.19(0.06)	0.12(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.44(0.06)	2.28(0.06)	0.15(0.01)	**0.00(0.00)**
	MBI	1.68(0.19)	3.56(0.07)	0.14(0.02)	0.13(0.03)
	DISCOM	**1.29(0.06)**	**1.86(0.06)**	0.05(0.00)	0.05(0.01)
	Imputed-MRCE	1.49(0.05)	2.13(0.08)	0.18(0.03)	0.07(0.02)
	Multi-DISCOM	**1.26(0.04)**	**1.77(0.09)**	**0.03(0.02)**	0.07(0.01)
$\alpha = 3$	Lasso	1.51(0.06)	3.70(0.06)	0.09(0.02)	**0.00(0.00)**
	Imputed-Lasso	1.73(0.06)	3.57(0.06)	0.11(0.01)	**0.00(0.00)**
	MBI	2.10(0.08)	4.26(0.09)	0.12(0.02)	0.11(0.03)
	DISCOM	1.44(0.04)	3.56(0.06)	0.05(0.00)	0.05(0.01)
	Imputed-MRCE	1.53(0.05)	3.72(0.08)	0.17(0.03)	0.08(0.02)
	Multi-DISCOM	**1.40(0.04)**	**3.39(0.08)**	**0.02(0.01)**	0.09(0.02)
$\alpha = 5$	Lasso	1.81(0.06)	5.70(0.06)	0.11(0.02)	**0.01(0.00)**
	Imputed-Lasso	1.89(0.06)	5.77(0.06)	0.15(0.01)	**0.01(0.00)**
	MBI	2.37(0.10)	5.95(0.12)	0.15(0.03)	0.12(0.02)
	DISCOM	1.71(0.04)	5.41(0.08)	0.06(0.02)	0.07(0.01)
	Imputed-MRCE	1.93(0.05)	5.66(0.09)	0.18(0.03)	0.10(0.02)
	Multi-DISCOM	**1.64(0.05)**	**5.19(0.12)**	**0.04(0.03)**	0.10(0.02)

Table A.4: Performance comparison of different methods for Example 2 with different signal-to-noise ratios. The values in the parentheses are the standard errors of the measures.

Method	$\|\hat{\mathbf{B}} - \mathbf{B}^*\|_F$	MSE	FPR	FNR
Lasso	1.50(0.06)	3.68(0.06)	0.09(0.02)	**0.00(0.00)**
Imputed-Lasso	1.72(0.06)	3.56(0.06)	0.12(0.01)	**0.00(0.00)**
MBI	2.11(0.08)	4.26(0.09)	0.12(0.02)	0.11(0.03)
DISCOM	1.45(0.04)	3.56(0.06)	0.05(0.00)	0.05(0.01)
Imputed-MRCE	1.55(0.05)	3.74(0.08)	0.18(0.03)	0.08(0.02)
Multi-DISCOM	**1.41(0.04)**	**3.42(0.08)**	**0.03(0.01)**	0.09(0.02)

Table A.5: Performance comparison of different methods for Example 3 with heavy-tailed error. The values in the parentheses are the standard errors of the measures.

BIBLIOGRAPHY

Althouse, B. M., Scarpino, S. V., Meyers, L. A., Ayers, J. W., Bargsten, M., Baumbach, J., Brownstein, J. S., Castro, L., Clapham, H., Cummings, D. A., et al. (2015). Enhancing disease surveillance with novel data streams: challenges and opportunities. *EPJ Data Science*, 4(1):1–8.

Anderson, T. (2008). *The theory and practice of online learning*. Athabasca University Press.

Arasu, A., Cherniack, M., Galvez, E., Maier, D., Maskey, A. S., Ryvkina, E., Stonebraker, M., and Tibbetts, R. (2004). Linear road: a stream data management benchmark. In *Proceedings of The Thirtieth International Conference on Very Large Data Bases-Volume 30*, pages 480–491.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404.

Banerjee, O., Ghaoui, L. E., and d'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research*, 9(Mar):485–516.

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.

Buckley, J. and James, I. (1979). Linear regression with censored data. *Biometrika*, 66(3):429–436.

Cai, T. T., Li, H., Liu, W., and Xie, J. (2013). Covariate-adjusted precision matrix estimation with an application in genetical genomics. *Biometrika*, 100(1):139–156.

Cavallanti, G., Cesa-Bianchi, N., and Gentile, C. (2007). Tracking the best hyperplane with a simple budget perceptron. *Machine Learning*, 69(2):143–167.

Cavallanti, G., Cesa-Bianchi, N., and Gentile, C. (2010). Linear algorithms for online multitask classification. *The Journal of Machine Learning Research*, 11:2901–2934.

Chen, J., Xu, P., Wang, L., Ma, J., and Gu, Q. (2018). Covariate adjusted precision matrix estimation via nonconvex optimization. In *International Conference on Machine Learning*, pages 922–931.

Cox, D. R. and Oakes, D. (2018). *Analysis of survival data*. Chapman and Hall/CRC.

Datta, S., Le-Rademacher, J., and Datta, S. (2007). Predicting patient survival from microarray data by accelerated failure time modeling using partial least squares and lasso. *Biometrics*, 63(1):259–271.

Dekel, O., Shalev-Shwartz, S., and Singer, Y. (2005). The forgetron: A kernel-based perceptron on a fixed budget. *Advances in Neural Information Processing Systems*, 18.

Ditzler, G. and Polikar, R. (2011). Hellinger distance based drift detection for nonstationary environments. In *2011 IEEE Symposium on Computational Intelligence in Dynamic and Uncertain Environments (CIDUE)*, pages 41–48. IEEE.

Eskildsen, S. F., Coupé, P., García-Lorenzo, D., Fonov, V., Pruessner, J. C., Collins, D. L., Initiative, A. D. N., et al. (2013). Prediction of alzheimer's disease in subjects with mild cognitive impairment from the adni cohort using patterns of cortical thinning. *Neuroimage*, 65:511–521.

Fisher, T. J. and Sun, X. (2011). Improved stein-type shrinkage estimators for the high-dimensional multivariate normal covariance matrix. *Computational Statistics and Data Analysis*, 55(5):1909–1918.

Frias-Blanco, I., del Campo-Ávila, J., Ramos-Jimenez, G., Morales-Bueno, R., Ortiz-Diaz, A., and Caballero-Mota, Y. (2014). Online and non-parametric drift detection methods based on hoeffding's bounds. *IEEE Transactions on Knowledge and Data Engineering*, 27(3):810–823.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Friedman, J., Hastie, T., Tibshirani, R., et al. (2001). *The elements of statistical learning*, volume 1. Springer Series in Statistics New York.

Gauthier, S., Reisberg, B., Zaudig, M., Petersen, R. C., Ritchie, K., Broich, K., Belleville, S., Brodaty, H., Bennett, D., Chertkow, H., et al. (2006). Mild cognitive impairment. *The Lancet*, 367(9518):1262–1270.

Gehan, E. A. (1965). A generalized wilcoxon test for comparing arbitrarily singly-censored samples. *Biometrika*, 52(1-2):203–224.

Gui, J. and Li, H. (2005). Penalized cox regression analysis in the high-dimensional and low-sample size settings, with applications to microarray gene expression data. *Bioinformatics*, 21(13):3001–3008.

Hazan, E., Rakhlin, A., and Bartlett, P. (2007). Adaptive online gradient descent. *Advances in Neural Information Processing Systems*, 20.

Hoi, S. C., Sahoo, D., Lu, J., and Zhao, P. (2021). Online learning: A comprehensive survey. *Neurocomputing*, 459:249–289.

Hsu, C.-H. and Yu, M. (2019). Cox regression analysis with missing covariates via nonparametric multiple imputation. *Statistical Methods in Medical Research*, 28(6):1676–1688.

Hyde, R., Angelov, P., and MacKenzie, A. R. (2017). Fully online clustering of evolving data streams into arbitrarily shaped clusters. *Information Sciences*, 382:96–114.

Izenman, A. J. (1975). Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, 5(2):248–264.

Jack, C. R., Petersen, R. C., Xu, Y. C., O'Brien, 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.

Jack Jr, C. R. (2012). Alzheimer disease: new concepts on its neurobiology and the clinical role imaging will play. *Radiology*, 263(2):344–361.

Jin, Z., Lin, D., Wei, L., and Ying, Z. (2003). Rank-based inference for the accelerated failure time model. *Biometrika*, 90(2):341–353.

Johnson, B. A. (2009). On lasso for censored data. *Electronic Journal of Statistics*, 3:485–506.

Johnson, C. R. (1990). Matrix completion problems: a survey. In *Matrix Theory and Applications*, volume 40, pages 171–198. Amer. Math. Soc.

Kalbfleisch, J. D. and Prentice, R. L. (2011). *The statistical analysis of failure time data*. John Wiley & Sons.

Kaplan, E. L. and Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, 53(282):457–481.

Kim, S., Xing, E. P., et al. (2012). Tree-guided group lasso for multi-response regression with structured sparsity, with an application to eqtl mapping. *The Annals of Applied Statistics*, 6(3):1095–1117.

Kimeldorf, G. and Wahba, G. (1971). Some results on tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33(1):82–95.

Kivinen, J., Smola, A. J., and Williamson, R. C. (2004). Online learning with kernels. *IEEE Transactions on Signal Processing*, 52(8):2165–2176.

Knopman, D. S., Boeve, B. F., and Petersen, R. C. (2003). Essentials of the proper diagnoses of mild cognitive impairment, dementia, and major subtypes of dementia. In *Mayo Clinic Proceedings*, volume 78, pages 1290–1308. Elsevier.

Koul, H., Susarla, V., and Van Ryzin, J. (1981). Regression analysis with randomly right-censored data. *The Annals of Statistics*, pages 1276–1288.

Lai, T. L. and Ying, Z. (1991). Rank regression methods for left-truncated and right-censored data. *The Annals of Statistics*, pages 531–556.

Langford, J., Li, L., and Zhang, T. (2009). Sparse online learning via truncated gradient. *Journal of Machine Learning Research*, 10(3).

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.

Legendre, P. and Anderson, M. J. (1999). Distance-based redundancy analysis: testing multispecies responses in multifactorial ecological experiments. *Ecological Monographs*, 69(1):1–24.

Littlestone, N. (1988). Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318.

Liu, K., Chen, K., Yao, L., and Guo, X. (2017). Prediction of mild cognitive impairment conversion using a combination of independent component analysis and the cox model. *Frontiers in Human Neuroscience*, 11:33.

Loh, P.-L. and Wainwright, M. J. (2015). Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *The Journal of Machine Learning Research*, 16(1):559–616.

Loh, W.-Y., Zheng, W., et al. (2013). Regression trees for longitudinal and multiresponse data. *The Annals of Applied Statistics*, 7(1):495–522.

Lu, J., Hoi, S. C., Wang, J., Zhao, P., and Liu, Z.-Y. (2016). Large scale online kernel learning. *Journal of Machine Learning Research*, 17(47):1.

Luo, X., Tsai, W. Y., and Xu, Q. (2009). Pseudo-partial likelihood estimators for the cox regression model with missing covariates. *Biometrika*, 96(3):617–633.

Manly, J. J., Tang, M.-X., Schupf, N., Stern, Y., Vonsattel, J.-P. G., and Mayeux, R. (2008). Frequency and course of mild cognitive impairment in a multiethnic community. *Annals of Neurology: Official Journal of the American Neurological Association and the Child Neurology Society*, 63(4):494–506.

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.

Miller, R. and Halpern, J. (1982). Regression with censored data. *Biometrika*, 69(3):521–531.

Misra, C., Fan, Y., and Davatzikos, C. (2009). 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.

Mitchell, A. J. and Shiri-Feshki, M. (2009). Rate of progression of mild cognitive impairment to dementia–meta-analysis of 41 robust inception cohort studies. *Acta Psychiatrica Scandinavica*, 119(4):252–265.

Molstad, A. J., Sun, W., and Hsu, L. (2021). A covariance-enhanced approach to multi-tissue joint eqtl mapping with application to transcriptome-wide association studies. *The Annals of Applied Statistics*, 15(2):998.

Mueller, S. G., Weiner, M. W., Thal, L. J., Petersen, R. C., Jack, C., Jagust, W., Trojanowski, J. Q., Toga, A. W., and Beckett, L. (2005). The alzheimer's disease neuroimaging initiative. *Neuroimaging Clinics*, 15(4):869–877.

Nan, B., Kalbfleisch, J. D., and Yu, M. (2009). Asymptotic theory for the semiparametric accelerated failure time model with missing data. *The Annals of Statistics*, 37(5A):2351–2376.

Negahban, S. N., Ravikumar, P., Wainwright, M. J., Yu, B., et al. (2012). A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557.

Ning, J., Qin, J., and Shen, Y. (2011). Buckley–james-type estimator with right-censored and length-biased data. *Biometrics*, 67(4):1369–1378.

Orabona, F., Keshet, J., and Caputo, B. (2008). The projectron: a bounded kernel-based perceptron. In *Proceedings of the 25th International Conference on Machine Learning*, pages 720–727.

Orabona, F., Keshet, J., and Caputo, B. (2009). Bounded kernel-based online learning. *Journal of Machine Learning Research*, 10(11).

Oulhaj, A., Wilcock, G. K., Smith, A. D., and de Jager, C. A. (2009). Predicting the time of conversion to mci in the elderly: role of verbal expression and learning. *Neurology*, 73(18):1436–1442.

Paik, M. C. and Tsai, W.-Y. (1997). On using the cox proportional hazards model with missing covariates. *Biometrika*, 84(3):579–593.

Parikh, N., Boyd, S., et al. (2014). Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239.

Park, M. and Moon, W.-J. (2016). Structural mr imaging in the diagnosis of alzheimer's disease and other neurodegenerative dementia: current imaging approach and future perspectives. *Korean Journal of Radiology*, 17(6):827–845.

Prentice, R. L. (1978). Linear rank tests with right censored data. *Biometrika*, 65(1):167–179.

Qi, L., Wang, C., and Prentice, R. L. (2005). Weighted estimators for proportional hazards regression with missing covariates. *Journal of the American Statistical Association*, 100(472):1250–1263.

Qi, L., Wang, Y.-F., Chen, R., Siddique, J., Robbins, J., and He, Y. (2018). Strategies for imputing missing covariates in accelerated failure time models. *Statistics in Medicine*, 37(24):3417–3436.

Rahimi, A., Recht, B., et al. (2007). Random features for large-scale kernel machines. In *NIPS*, volume 3, page 5. Citeseer.

Raskutti, G., Wainwright, M. J., and Yu, B. (2011). Minimax rates of estimation for high-dimensional linear regression over lq-balls. *IEEE Transactions on Information Theory*, 57(10):6976–6994.

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

Reid, N. (1994). A conversation with sir david cox. *Statistical Science*, 9(3):439–455.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386.

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.

Rubin, D. B. (2004). *Multiple imputation for nonresponse in surveys*, volume 81. John Wiley & Sons.

Rudin, W. (1962). *Fourier analysis on groups*, volume 121967. Wiley Online Library.

Schaul, T., Zhang, S., and LeCun, Y. (2013). No more pesky learning rates. In *International Conference on Machine Learning*, pages 343–351. PMLR.

Smith, P. J. (2017). *Analysis of failure and survival data*. Chapman and Hall/CRC.

Steingrimsson, J. A. and Strawderman, R. L. (2017). Estimation in the semiparametric accelerated failure time model with missing covariates: improving efficiency through augmentation. *Journal of the American Statistical Association*, 112(519):1221–1235.

Sun, Y., Gilbert, A., and Tewari, A. (2018). But how does it work in theory? linear svm with random features. *Advances in Neural Information Processing Systems*, 31.

Ta, V.-D., Liu, C.-M., and Nkabinde, G. W. (2016). Big data stream computing in healthcare real-time analytics. In *2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pages 37–42. IEEE.

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

Tibshirani, R. (1997). The lasso method for variable selection in the cox model. *Statistics in Medicine*, 16(4):385–395.

Tsiatis, A. A. (1990). Estimating regression parameters using linear rank tests for censored data. *The Annals of Statistics*, pages 354–372.

Tsymbal, A. (2004). The problem of concept drift: definitions and related work. *Computer Science Department, Trinity College Dublin*, 106(2):58.

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

Vemuri, P., Wiste, H., Weigand, S., Shaw, L., Trojanowski, J., Weiner, M., Knopman, D. S., Petersen, R. C., Jack, C., et al. (2009). Mri and csf biomarkers in normal, mci, and ad subjects: predicting future clinical change. *Neurology*, 73(4):294–301.

Wang, C. and Chen, H. Y. (2001). Augmented inverse probability weighted estimator for cox missing covariate regression. *Biometrics*, 57(2):414–419.

Wang, S., Nan, B., Zhu, J., and Beer, D. G. (2008). Doubly penalized buckley–james method for survival data with high-dimensional covariates. *Biometrics*, 64(1):132–140.

Wang, Z. and Wang, C. (2010). Buckley-james boosting for survival analysis with high-dimensional biomarker data. *Statistical Applications in Genetics and Molecular Biology*, 9(1).

White, I. R. and Royston, P. (2009). Imputing missing covariate values for the cox model. *Statistics in Medicine*, 28(15):1982–1998.

Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., and Fu, Y. (2019). Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 374–382.

Xu, Q., Paik, M. C., Luo, X., and Tsai, W.-Y. (2009). Reweighting estimators for cox regression with missing covariates. *Journal of the American Statistical Association*, 104(487):1155–1167.

Xue, F. and Qu, A. (2021). Integrating multisource block-wise missing data in model selection. *Journal of the American Statistical Association*, 116(536):1914–1927.

Yin, J. and Li, H. (2011). A sparse conditional gaussian graphical model for analysis of genetical genomics data. *The Annals of Applied Statistics*, 5(4):2630.

Yu, G., Li, Q., Shen, D., and Liu, Y. (2020). Optimal sparse linear prediction for block-missing multi-modality data without imputation. *Journal of the American Statistical Association*, 115(531):1406–1419.

Yu, M. (2011). Buckley–james type estimator for censored data with covariates missing by design. *Scandinavian Journal of Statistics*, 38(2):252–267.

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.

Zhang, C., Liu, Y., and Wu, Y. (2016). On quantile regression in reproducing kernel hilbert spaces with the data sparsity constraint. *The Journal of Machine Learning Research*, 17(1):1374–1418.

Zhang, D., Shen, D., and Initiative, A. D. N. (2012a). Predicting future clinical changes of mci patients using longitudinal and multimodal biomarkers. *PloS One*, 7(3):e33182.

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

Zhao, P., Wang, J., Wu, P., Jin, R., and Hoi, S. C. (2012). Fast bounded online gradient descent algorithms for scalable kernel-based online learning. *arXiv preprint arXiv:1206.4633*.

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

Zinkevich, M. (2003). Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (icml-03)*, pages 928–936.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320.