Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

_____ _____

Xin Ma Date

Prediction Approaches for High-dimensional and Complex Neuroimaging Data

By

Xin Ma

Doctor of Philosophy

Biostatistics

Suprateek Kundu, Ph.D.
Advisor

Ying Guo, Ph.D.
Committee Member

John Hanfelt, Ph.D.
Committee Member

Deqiang Qiu, Ph.D.
Committee Member

Accepted:

Kimberly Jacob Arriola, Ph.D, MPH
Dean of the James T. Laney School of Graduate Studies

Date

Prediction Approaches for High-dimensional and Complex Neuroimaging Data

By

Xin Ma
M.A. Columbia University, 2012

Advisor: Suprateek Kundu, Ph.D.

An Abstract of
a dissertation submitted to the Faculty of the
James T. Laney Graduate School of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Biostatistics
2022

Abstract

Prediction Approaches for High-dimensional and Complex Neuroimaging Data

By Xin Ma

Neuroimaging studies continue to scale up with more participants, multiple follow-up visits, and higher scanning resolutions. High-dimensionality, spatial distribution and low signal-to-noise ratio make neuroimaging data challenging to work with, requiring development of novel and flexible methodology for prediction and feature selection.

In topic 1, we develop a novel two-stage Bayesian regression framework using functional connectivity networks as covariates and a scalar continuous outcome variable. The approach first finds a lower dimensional node-specific representation for the networks, then embeds these representations in a flexible Gaussian process regression framework with node selection via spike-and-slab prior. Extensive simulations and a real application show distinct advantages of the proposed approach regarding prediction, coverage, and node selection. To our knowledge, the proposed approach is one of the first nonlinear semi-parametric Bayesian regression models based on high-dimensional functional connectivity features.

In topic 2, we propose a novel joint scalar-on-image regression framework involving wavelet-based image representations with grouped penalties to pool information across inter-related images for joint learning. We explicitly account for noise in images via a corrected objective function. We derive non-asymptotic statistical error bounds under the grouped penalties, allowing the number of voxels to increase exponentially with sample size. A projected gradient descent algorithm is used for computation and shown to approximate the optimal solution via non-asymptotic optimization error bounds under noisy images. Extensive simulations and an application to Alzheimer's study demonstrate significantly improved predictability and greater power to detect signals.

In topic 3, we generalize the idea in topic 2 to Lipschitz continuous loss functions, including logistic loss, hinge loss and quantile regression loss. We propose a unified sparse learning framework in high-dimensional setting with built-in strategy for measurement errors. Unlike the approach with corrected objective function for linear models in topic 2, we find a sparse estimator in a confidence set based on the gradient of empirical loss function. We derive the non-asymptotic statistical error bounds and sign consistency results for the proposed estimator. We develop a Newton-Raphson type algorithm with linear programming and conduct extensive numerical experiments to illustrate the superior performance of our proposed estimator in various settings.

Prediction Approaches for High-dimensional and Complex Neuroimaging Data

By

Xin Ma
M.A. Columbia University, 2012

Adviser: Suprateek Kundu, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney Graduate School of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Biostatistics
2022

Acknowledgements

The past five and a half years has been a wonderful journey and one of the most memorable parts of my life. I have many people to thank who have ridden along with me during this journey.

First, I would like to thank my dissertation advisor Dr. Kundu. He has led me into the academia world, introduced me to new research topics and always encouraged me to think further. He has offered valuable advice on various aspects of research work and come to help when I feel struggled.

Second, I would like to thank my dissertation committee members Dr. Guo, Dr. Hanfelt, and Dr. Qiu. They always have good suggestions for my dissertation work based on their different scientific perspectives. They are always encouraging and provide me with positive feedback.

Next, I would like to thank the faculty and staff administrators of our BIOS department, for creating a secure and enjoyable environment for a doctoral student to grow. Special thanks to our three department chairs, Dr. Waller, Dr. Hanfelt, Dr. Krafty, our DGS Dr. Qin, our ADAP Angela, and our wonderful staffs Mary, Melissa, Joy and Bob.

In addition, I would like to thank the faculty members Dr. Guo, Dr. Kundu, Dr. Risk, and the doctoral students from the CBIS group. It is within this group that I set my foot into the neuroimaging field and become fascinated by the studies that help us learn our brain like never before.

I also would like to thank all my fellow students in the BIOS department, especially my former officemates Jin, Bo, Josh, Nancy, Yingtian, Zhengyi, Andrea, Yunxiao, Thomas, Lindsay and Lin. They have filled this journey with laugh and happiness.

Finally, I would like to thank my husband Teng. I am so lucky to have met him at Emory and shared this journey together. Through all ups and downs, he is always there to support,

as a family and a fellow researcher. I am sure we will be sharing many more years of our beautiful lives ahead.

We are still in a global pandemic. Actually I am trying to fight off one COVID-19 infection right now. But I believe we will find a path out of the pandemic eventually. We just need to keep calm and carry on.

Xin

March, 2022

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Development of neuroimaging techniques has provided researchers an opportunity to investigate the function and anatomy of human brain in a safe and non-invasive way. There has been growing interest using neuroimaging data in pattern recognition, classification and prediction of psychiatric disorders. However, the high dimensionality and complex structure of spatial correlation of the neuroimaging data has posed special challenges for statistical analysis. In this dissertation, we focus on developing novel prediction approaches that address these challenges and yield meaningful results in prediction, coefficient estimation as well as important feature selection.

The dissertation is organized as follows: the rest of Chapter 1 introduces the magnetic resonance imaging (MRI) technique, the motivating examples and discusses challenges of prediction modeling using neuroimaging data. Chapter 2 presents a novel two stage Bayesian framework for prediction and node selection using functional network data. Chapter 3 presents a novel approach for the joint analysis of multiple scalar-on-image regression models. Chapter 4 presents a unified framework of sparse learning for a class of smooth Lipschitz loss functions encompassing binary classification and quantile regression.

1.2 Magnetic Resonance Imaging of Brain

Magnetic Resonance Imaging technique can be used to illustrate the anatomical and physiological characteristics of the human brain. It utilizes the fact that the nuclei of hydrogen atoms, which are abundant in the fat and water of human brain, are able to absorb radio frequency (RF) energy and re-emit it when placed in a magnetic field. To obtain a brain MRI scan, a subject is send into the MRI scanner with a strong magnetic field which aligns the nuclei of hydrogen atoms in one direction. Then a RF pulse is applied to knock the nuclei of hydrogen atoms off the aligned direction. Once the RF pulse is removed, the nuclei of hydrogen atoms spin gradually back to the original aligned direction and as a consequence emit RF signal which is captured by receiver coils. These received signals are then processed and transformed into readable images. By varying designs of the RF sequence, we can obtain MRI images with different contrast on tissue types. The two main parameters are the repetition time (TR) and the echo time (TE).

With a short TR and a short TE, T1-weighted spin-echo sequence provides good contrast between gray and white matter tissues, while cerebrospinal fluid (CSF) is void of signal. This type of MRI images is useful for brain segmentation. We can also obtain measurement such as intracranial volume (ICV), and cortical thickness in downstream analysis.

Functional MRI measures brain activity with blood oxygenation level dependent (BOLD) contrast. When brain is activated in one area, there is an increase in blood flow to send in glucose, together with an increase in oxygenated hemoglobin (OxyHb) molecules. The increase in OxyHb molecules usually outpaces the metabolic needs and results in an increase in the ratio of OxyHb molecules to deoxygenated hemoglobin (deOxyHb) molecules. As deOxyHb is paramagnetic while OxyHb is diamagnetic, the increase in ratio between these two types of molecules changes the magnetic property of blood making the T2* time longer and MRI signal stronger relative to normal state.

There are task functional MRI and resting-state (task-free) functional MRI sessions. During one session, each voxel within the brain has a BOLD signal time course. Functional connectivity (FC) measures the temporal correlation of the BOLD time series in different

regions of interest. Research on resting-state FC has led to the discovery of several resting state networks, including the default mode network.

1.3 Motivating Examples

There are two motivating studies for our dissertation, each conducted on a specific type of psychiatric disorder. Depending on the difference in disease pathology, we will use different types of neuroimaging data in our investigation.

1.3.1 Grady Trauma Project

The Grady Trauma Project (GTP) recruited African American females to study the risk factors for post-traumatric stress disorder (PTSD) in a low-socioeconomic status (Stevens et al., 2013). The resting state functional MRI (rs-fMRI) scans were obtained on a 3.0T Siemens Trio with echo-planar imaging (Siemens, Malvern, PA) and the functional images were collected in an ascending interleaved sequence with 37 3mm axial slices and no gap between slices (TR/TE=2000/30ms, FA=90°, $3mm^3$ voxel size). The study has also acquired data on demographic factors such as the participants' age and several clinical scales relating to PTSD, including the Connor-Davidson Resilience Scale (Connor and Davidson, 2003) for measuring reselience as individual's ability to thrive in the face of adversity, traumatic events inventory (TEI) score (Sprang, 1997) and the childhood trauma questionnaire (CTQ) total score (Scher et al., 2001).

It is now increasingly believed that mental disorders may not be associated with abnormalities in specific local regions but may result due to disruptions in functional connectivity as captured by changes in the brain network (Zhan and Yu, 2015). In this respect, there has been some progress in classifying the disease status in mental disorder studies based on brain networks (Du et al., 2018). However, such existing classification approaches may not be fully satisfactory for a heterogeneous mental disorder such as PTSD, where the definition of disease phenotypes itself has been a stumbling block. Hence instead of focusing on

classification, it is appealing to develop prediction approaches for continuous measures of mental disorders based on the brain network and other risk factors. The GTP data has been investigated in the first topic, utilizing functional connectivity networks estimated from its rs-fMRI scans and the demographic and clinical data been collected.

1.3.2 Alzheimer's Disease Neuroimaging Initiative

The Alzheimer's Disease Neuroimaging Initiative (ADNI) "was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of ADNI has been to test whether serial magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD). For up-to-date information, see www.adni-info.org." The ADNI data can be obtained from its database (adni.loni.usc.edu).

The ADNI study has undertaken three phases (ADNI 1, ADNI GO/2 and ADNI 3) up to date. The first phase ADNI 1 study was conducted between 2004 and 2009 and aimed to enroll 400 subjects with early MCI, 200 subjects with early AD and 200 normal control subjects. T1- and dual echo T2-weighted MRI images were collected on 1.5T scanners at baseline, 6 month, 12 month and 24 month for all three groups of subjects. The images underwent quality control and its standardized collections are available on the ADNI database. In our second topic, we plan to use the ADNI 1 T1-weighted imaging data as well as other biomarkers and assessment to illustrate the performance of our proposed method.

1.4 Prediction Modeling on Neuroimaging Data

There has been growing interest using neuroimaging data in pattern recognition, classification and prediction of psychiatric disorders (Zhan and Yu, 2015; Du et al., 2018; Falkai et al., 2018). However, using neuroimaging data in predictive models encounters several challenges.

First is the high dimensionality of the data structure. For structural MRI data such as T1-weighted images, the data has three dimensions. While for functional MRI data, the dimension is 4D as it consists of a series of 3D images along the time. This type of predictors is difficult for conventional regression models to handle. One way to get around is to use summary metrics or voxel-wise data. However, this results in information loss and potential false positives from multiple testing. Another challenge comes from the complex structure of imaging data's spatial correlation. It is not straightforward in modeling the spatial correlation and linking it to the response variable. Additionally, the imaging data usually involves hundreds of thousands of voxels, which is high dimensional in a different sense. This has posed a high requirement on the efficiency of the computing algorithm. We also note the importance of interpretability which can lead to desirable implication for research in disease mechanism and diagnosis. Nevertheless, the recently emerged machine learning and deep learning methods usually lack clear interpretation and act more like a black box process.

In order to tackle these challenges, we develop novel prediction approaches that adapt to the characteristics of the neuroimaging data and yield meaningful prediction and interpretation.

Chapter 2

Semi-parametric Bayes Regression with Network-valued Covariates

2.1 Introduction

Recent studies have given rise to a rich variety of network data involving fraud detection (Akoglu et al., 2015), brain networks (Lukemire et al., 2020), social networks (Liben-Nowell and Kleinberg, 2007), traffic forecasting (Cui et al., 2019; Li et al., 2017), computer vision (Monti et al., 2017) and so on, that arise in diverse applications. Networks provide inherently richer insights by delving into relationships between nodes that represent different entities in different application domains (nodes can represent brain regions in neuroimaging problems, an individual in social networks, sensors in traffic forecasting, and so on). By using the web of relationships represented by the network, one can obtain more accurate prediction and analysis in neuroimaging applications (Guha and Rodriguez, 2020), traffic forecasting (Li et al., 2017), synthetic identity dectection (Zhang et al., 2017), financial fraud detection (Zhou et al., 2017), and other scientific applications. Similar to other domains, classification and prediction problems based on networks has gained increasing prominence in our motivating neuroimaging applications involving mental health, where network differences between disease versus control groups resulting from disruptions in the

brain functional and structural connectivity are well-established (Higgins et al., 2019; Zhan and Yu, 2015). Such brain network disruptions are clearly evident in our motivating Grady Trauma project (GTP) application that shows considerable differences in brain connectivity between individuals with high and low posttraumatic stress disorder (PTSD) resilience (see Figure 2.1).

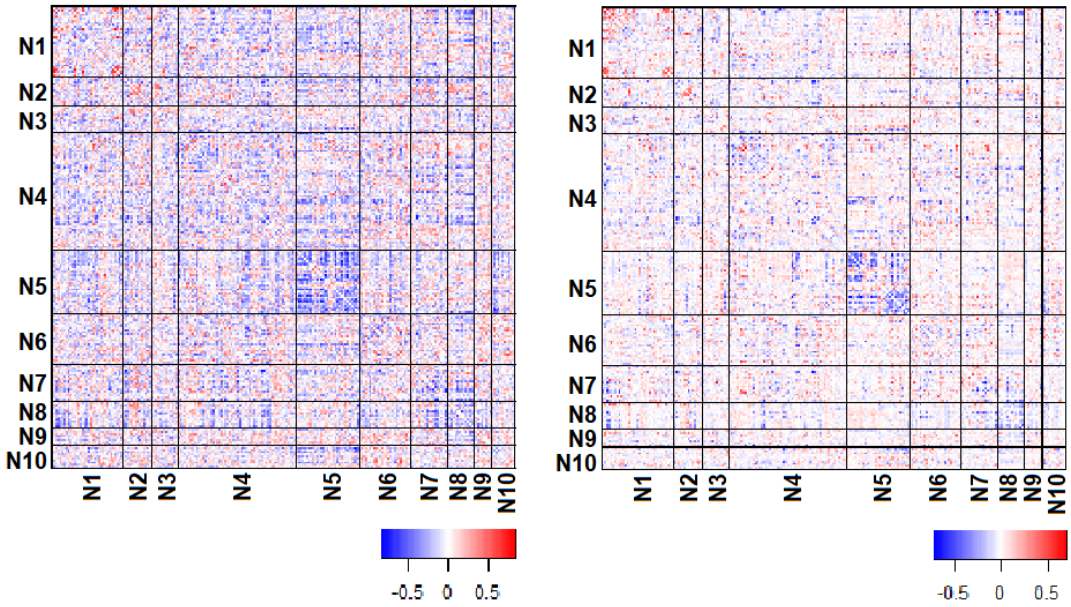

Figure 2.1: Differences in absolute correlations (left panel) and fitted edge probabilities (right panel) between participants with highest and lowest resilience score. N1 through N10 respectively denote the following functional networks: motor, cingulo-opercular, auditory, default mode, visual, fronto-parietal, salience, sub-cortical, ventral attention, and dorsal attention.

While a limited number of classification methods using high-dimensional network-valued covariates have been proposed (Du et al., 2018; Relión et al., 2019), the literature on prediction models using networks is unfortunately even more sparse and has several crucial pitfalls. For example, most classification and prediction approaches using network valued covariates assume linear relationships, and they often do not account for the complex structures inherent in the networks. Linear models are restrictive in terms of not accommodating non-linear relationships between the outcome and the network, and they also do not account for unknown interactions between the network and other supplementary covariates, and even between different network features. The presence of such non-linear associations between the clinical outcome and some edge strengths are evident in our motivating mental

health neuroimaging applications (see Figure 2.2). One possible avenue for bypassing these constraints involve deep learning approaches such as convolutional neural networks (CNN) as in Meng and Xiang (2018), which extract deep non-linear embeddings from the network that is subsequently used to predict the outcome. However, deep learning methods are data-hungry and require enormous sample sizes that may not always be available especially for neuroimaging problems of interest in this chapter. Moreover, deep learning techniques typically do not provide the ability for inference and feature selection, which is desirable in our applications of interest. Hence, novel flexible and interpretable non-linear approaches with the added capability to perform inference are required.

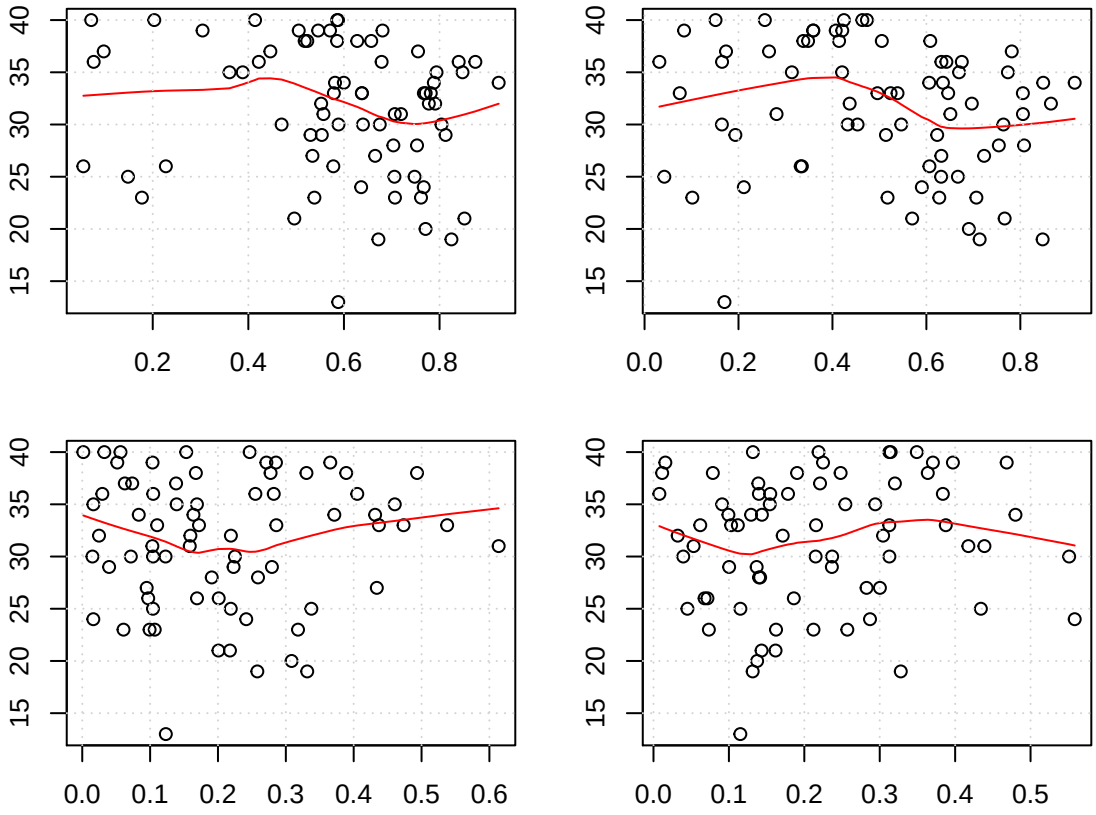

Figure 2.2: Scatter plots from the Grady Trauma Project data. The vertical axis represents the resilience score. The horizontal axis represents edge probabilities from four selected edges. The red lines are obtained via Locally Weighted Scatterplot Smoothing (LOWESS) (Cleveland, 1979).

The second important limitation with current classification and prediction methods that use

network-valued covariates results from the fact the networks are non-Euclidean objects that have an inherently complex structure. This feature results in sub-optimal performance for standard prediction and classification techniques that typically treat network edges/features as interchangeable. In addition, using the entire edge set for the network results in an overly inflated model where the number of candidate parameters increases quadratically with the number of nodes. One potential remedy is to extract lower dimensional embeddings from the high-dimensional networks with minimal loss of information (see Cui et al. (2018) for a survey of graph embedding methods), and subsequently use these embeddings for prediction. However, these graph embedding methods typically result in loss of information and it is not immediately clear how well these embedding approaches are suited for classification and/or prediction purposes and their effect on uncertainty quantification. Moreover, projection of the network onto a lower dimensional manifold leads to loss of interpretability that may result in difficulties in terms of network feature selection. To our knowledge, there is very limited literature on Bayesian classification and prediction methods using lower dimensional projections of high-dimensional networks, which also accommodates network feature selection. Such an approach clearly warrants further consideration and investigation, and is the main focus of this chapter.

Our goal in this chapter is to propose a novel two-stage Bayesian methodology to address the aforementioned gaps in network analysis literature. In particular, we develop a flexible Gaussian process modeling framework for scalar continuous outcomes based on lower dimensional manifold projections of high-dimensional networks. Our two stage method uses a manifold learning approach in the first stage to project the high-dimensional network onto lower dimensional latent scale space (Hoff, 2005) in order to address the curse of dimensionality. The first stage projection approach results in interpretability at the node level and is remarkable in terms of preserving the original network characteristics given small to moderate number of channels. In the second stage, the projected latent scale features are used to predict continuous scalar outcomes via a flexible Gaussian process regression, which naturally accommodates non-linear relationships and incorporates unknown interactions. In addition to being able to provide a systematic way for extracting the dimension of

the underlying manifold space, the proposed approach is able to perform feature selection at the network node level via spike and slab priors on the lengthscale parameters of the Gaussian process, which provides significant advantages over competing network embedding approaches that typically do not preserve interpretability and hence are not equipped to explicitly perform network feature selection.

We denote our approach as latent scale Gaussian process regression (ls-GPR) and develop a fast computational method for implementation. An efficient Expectation-Maximization (EM) algorithm for estimating the latent scales from the given network data is proposed for the first stage model, by leveraging the Polya-Gamma data augmentation scheme in Polson et al. (2013). The Gaussian process regression in the second step is implemented via Markov chain Monte Carlo (MCMC) tools. We note that while it is tempting to propose a joint model that simultaneously updates the latent scales and the regression parameters, such a model runs into computational challenges when updating the latent scales due to the difficulties in deriving closed form posteriors, which is a well-known problem in literature (Yang et al., 2016). One can possibly discretize the latent scales to facilitate computational updates, or alternatively use Metropolis-Hastings based strategies or their more efficient variants (Robert, 2015) for such a joint analysis. However, both strategies have drawbacks in high dimensions: the former may lead to shrinkage of prior support, while the latter may result in inefficient mixing. Hence we adopt a more practical two-stage implementation that alleviates these computational challenges and is scalable to high dimensional networks. We perform extensive simulations under a variety of network structures and compared the performance to existing linear and non-linear approaches. Our results illustrate the clear advantages of the proposed approach in terms of higher prediction accuracy, superior predictive uncertainty quantification in terms of coverage intervals for test samples that also often have much lower interval widths, and power to detect truly significant predictors while controlling the false positives. Our neuroimaging application involving PTSD resilience modeling based on brain networks reveals significantly higher prediction accuracy and better predictive uncertainty quantification under the proposed method, and identifies important brain regions that are associated with PTSD.

We would note that this chapter makes several important contributions. To our knowledge, the proposed approach is one of the first to develop a Bayesian non-parametric regression approach based on network-valued covariates with the added capability of node level feature selection. The superior numerical performance of the approach results from successfully resolving the question of whether accurate lower dimensional manifold representations of high-dimensional networks can be used for improved prediction in conjunction with flexible non-linear regression models (in our case, Gaussian process regression), and whether they can be used to infer important network features. The latent scale representation provides a desirable balance between two extreme scenarios involving an edge-level analysis that is not appealing due to the reasons mentioned previously, and alternate dimension reduction techniques that do not preserve interpretability and hence precludes network feature selection. To our knowledge, given that there is a limited literature on variable selection in Gaussian process latent variable models (GP-LVMs) and limited or no variable selection methods for Bayesian non-parametric regression methods using network-valued covariates, our contribution is of independent interest.

2.2 Related Literature

A common strategy for regression based on network valued covariates is to use summary network measures as explanatory variables (see, for example, Bullmore and Sporns (2009) and references therein). However, the success of such an approach depends heavily on the choice of the network metrics. Moreover, these approaches have reduced exploratory value and potentially sub-optimal performance due to decreased resolution of the summary statistics. Another alternative is to include all the edges in the network as a vectorized predictor, and use these high dimensional features for modeling the clinical phenotype (Craddock et al., 2009). Although penalized regression approaches (Tibshirani, 1996) and Bayesian shrinkage (Chang et al., 2018) may be used to model the regression coefficients in these high-dimensional applications, these approaches often treat the edges as interchangeable and fail to respect the inherent network structure that may show properties such as small-

worldedness (Bassett and Bullmore, 2006) or other patterns of organization. Disregarding these inherent structures and the associated correlations may lead to sub-optimal performance. Recent developments such as Guha and Rodriguez (2020) address this issue by using a tensor-based representation for the regression parameters corresponding to the edges in the network, while (Relión et al., 2019) propose a linear classification model that encourages sparsity in the number of nodes and edges of the coefficient matrix. However, these linear approaches still require estimating as many regression coefficients as there are edges, and hence are challenging to implement for high dimensional networks including hundreds of regions containing tens of thousands of edges. Going beyond linear approaches, a contemporary work by Weaver et al. (2021) presented a regularized single index modeling approach that links the outcome to a linear combination of covariates using an unspecified smooth function modeled via splines. Although more flexible than linear regression approaches, single index models do not mitigate the curse of dimensionality presented by a massive number of edges corresponding to our high-dimensional networks of interest (involving 264 nodes and 34716 edges), which may potentially lead to overfitting and sub-optimal performance. An additional limitation for the approach by Weaver et al. (2021) is that it only reports point estimates without presenting uncertainty quantification and lacks inferential capabilities for feature selection that is often critical in neuroimaging studies.

A possible alternative approach to tackle the curse of dimensionality in regression problems involving high-dimensional networks involves manifold learning, where the network is first projected onto a lower dimensional manifold, which is subsequently embedded within a flexible regression framework. Frequentist examples of reducing the dimension of feature space include principal component analysis and more elaborate methods that accommodate non-linear subspaces, such as isomap (Tenenbaum et al., 2000) and Laplacian eigenmaps (Belkin and Niyogi, 2003). Bayesian manifold approaches characterizing predictive uncertainty have also been developed. Page et al. (2013) proposed a Bayesian nonparametric model for learning of an affine subspace in classification problems, and Kundu and Dunson (2014) proposed a Gaussian process latent variable model to accommodate non-linear subspaces for prediction with scalar covariates. However, there may be a heavy computational

price to extricate the number and distribution of the latent variables, and then simultaneously learning the mapping functions while keeping identifiability restrictions. These factors typically restrict such manifold learning based approaches involving Gaussian process models to a small to moderate number of features, although some limited number of scalable approaches are available that avoid having to learn the mapping to the lower-dimensional subspace (Yang et al., 2016). The lack of scalabilty under Gaussian process regression to more than a few hundred or thousand predictors is not surprising, given that it is not trivial to sample a large number of lengthscale parameters under an anisotropic Gaussian process model via Metropolis-Hastings updates in such high-dimensions.

Another important limitation of existing manifold learning approaches is that the features in the lower dimensional manifold are typically not interpretable in terms of the original network features. This may be restrictive in neuroimaging applications where it is often important to identify important brain regions that are associated with mental illnesses via localized changes in network configurations. Similar limitations apply for non-linear deep learning approaches for constructing prediction models using network-valued covariates, which have the added complication of requiring large sample sizes to train the deep layers. These efforts are further complicated by the fact that the generalization of deep learning tools such as CNN to irregular or non-Euclidean network data is non-trivial. In fact, there is very limited literature on this topic (Meng and Xiang, 2018), and advanced tools such as graph neural networks (Wu et al., 2020) require more testing and validation for brain network applications of interest. Even then, such existing deep learning approaches are not applicable to our problems of interest that focus on interpretable Bayesian methods incorporating network feature selection through inference, and quantification of uncertainty.

2.3 Proposed Methods

Suppose we have data on n participants. For the ith participant ($i = 1, \ldots, n$), the data includes a continuous scalar variable $y_i \in \Re$ representing the clinical outcome of interest, an undirected brain network having p nodes represented as a symmetric binary matrix

$G_i(p \times p)$, with $g_i(k,l) = 1/0$ depending on whether the edge $(k,l), k \neq l$, is present or absent in the network, and supplemental covariates $\mathbf{z}_i$ representing environmental exposures and demographic factors. The binary network G can be based on structural or functional connectivity, while the number of regions depends on the chosen atlas ($p = 264$ under the Power atlas for our applications). We denote the vector of elements in the upper triangle excluding the diagonals for G_i, or edge set, as $\mathbf{e}_i$ of length $p(p-1)/2$. The diagonal elements are excluded since they do not represent connections between distinct nodes and are irrelevant to our problem of interest. The method is described in detail below, and Figure 2.3 provides a diagrammatic illustration of our two-stage model.

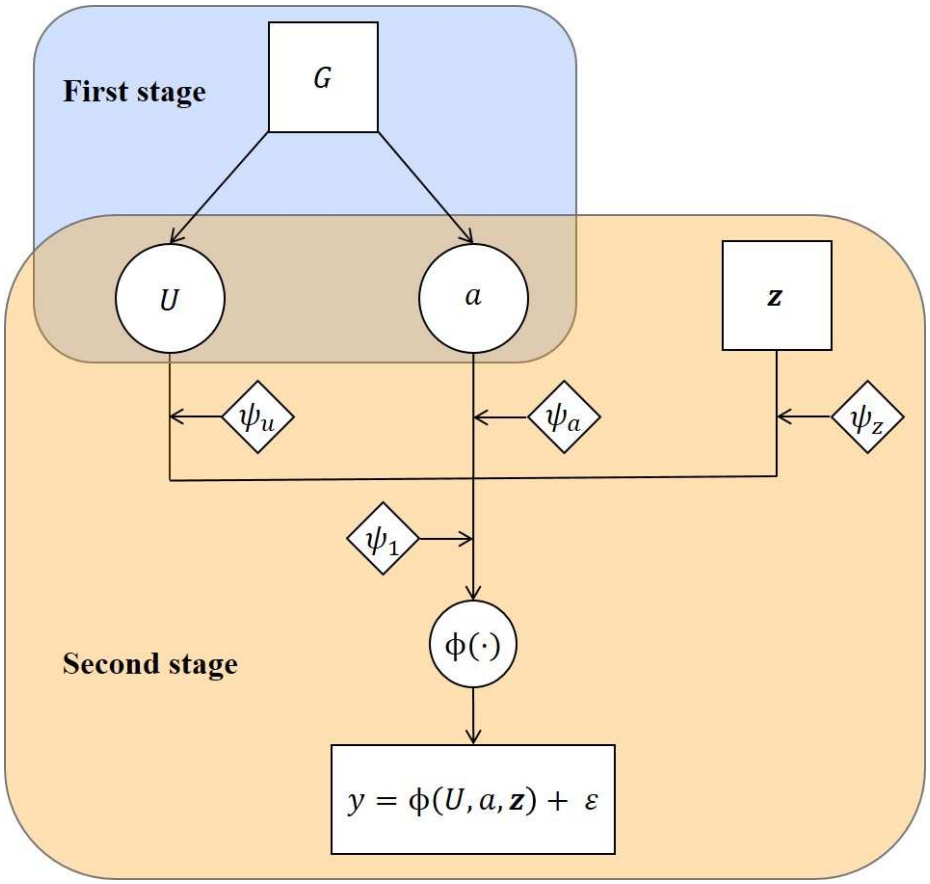

Figure 2.3: Schematic Diagram of the Two-stage Model. G represents the given network that is projected onto a lower dimensional manifold involving latent scales U and intercept a in the first stage. These parameters are then combined with observed environmental exposures z to model the outcome via the unknown mean function $\phi(\cdot)$ that is modeled via a Gaussian process regression, in the second stage.

2.3.1 Model formulation

First Stage: Latent Scale Representation of Brain Networks

Our goal is to have a parsimonious probability model for the binary networks represented by the edge sets $\mathbf{e}_i, i = 1, \ldots, n$. Clearly there are $2^{p(p-1)/2}$ possible models for the graph space $\mathcal{G}$ that grows exponentially with the number of nodes. In order to tackle this curse of dimensionality, we project the network into a lower dimensional space via a meaningful mapping that avoids restrictive assumptions and fits the data reasonably well. Motivated by the above considerations, we represent the edge probabilities in terms of node level latent scales. In particular, we fit the following latent scale model separately for each sample

$$P(\mathbf{e}_i) = \prod_{k<l,k,l=1}^{p} \pi_{i,kl}^{e_{i,kl}}(1-\pi_{i,kl})^{1-e_{i,kl}}, \ \log(\frac{\pi_{i,kl}}{1-\pi_{i,kl}}) = a_i + \mathbf{u}_{ik}^T \Lambda_i \mathbf{u}_{il},$$
$$a_i \sim N(0, \sigma_a^2), \ u_{ik,r} \sim \mathrm{N}(0, \sigma_u^2), \ k = 1, \ldots, p, i = 1, \ldots, n, \tag{2.3.1}$$

where $\mathbf{u}_{ik} = (u_{ik,1}, \cdots, u_{ik,d})^T$ is the vector of latent scale having d channels for node k, a_i denotes the subject-specific intercept common across edges, and Λ_i represents the $d \times d$ diagonal weight matrix with elements $(\lambda_{i1}, \ldots, \lambda_{id})$ that controls the contribution of the latent scales to the inner product in (2.3.1) corresponding to the ith participant. The intercept term controls the overall density of the network and is learnt by pooling data across all edges. The latent scale $\mathbf{u}_l$ captures the importance of node l in the network. If both nodes l and k have activations in the same directions, captured via $u_{k,r}$ and $u_{l,r}$ having the same signs, then they can be construed as functionally connected. In order to preserve identifiability with respect to rotation, we fix the first element of all latent scales (0.5 in our implementations), and the first diagonal element in weight matrix Λ_i is also fixed to be one. The remaining diagonal elements in the weight matrix are either pre-specified, or assigned a Bernoulli prior $\lambda_{ir} \sim Ber(\pi), \ r = 2, \cdots, d$, that allows one to adaptively select the important channels specific to the network for each individual. The unknown prior inclusion probability π, which controls the number of channels that are expected to be included when fitting the network, is estimated adaptively under a Beta(a_π,b_π) prior.

Model (2.3.1) results in a dramatic reduction in the number of parameters from $p(p-1)/2$ to the order of $p \times d + 1$. In particular, the latent scales $U_i = (\mathbf{u}_{i1}, \ldots, \mathbf{u}_{ip})$, for participant i, have dimension $d \times p$ where p is the number of brain regions and d is the intrinsic dimension of the latent scale (or channels) that needs to be determined. In general, finding the intrinsic dimension of the manifold is a difficult problem (Yang et al., 2016). However, we are able to systematically choose the manifold dimensions under our approach. The number of channels d can be chosen from a range of possible values, as one that results in the smallest BIC score in the subsequent second stage regression model. Such a choice of d represents a common latent dimension across all samples that results in the best out of sample prediction performance, although the relative importance of individuals channels when fitting the network in the first stage model is expected to vary across samples.

Model (2.3.1) is inspired by the latent space modeling of a single network (Hoff, 2005) that has been shown to provide a more general characterization of interconnection structures and network properties than stochastic block model (Nowicki and Snijders, 2001) and latent distance model (Hoff et al., 2002). A similar model for undirected binary networks appeared in Durante et al. (2017), who used a mixture of latent scale probabilities to model a population of networks. For our neuroimaging applications of interest, it may be too simplistic to assume that groups of participants in the sample share the same latent scales exactly, due to the well-known inherent heterogeneity in PTSD (Lanius et al., 2006). Moreover, our primary objective is to predict the clinical outcome based on the network, which requires a distinct lower dimensional representation for the network corresponding to each participant. Hence for our application, having individual network specific parameters in (2.3.1) seems appropriate. In addition, instead of multiplicative gamma priors on the diagonal elements of Λ in Durante et al. (2017) that are specified for adaptive estimation of the weight matrix, our prior construction uses Bernoulli priors on the diagonal elements of Λ that is computationally more straightforward, and allows one to actually select the important channels for fitting the individual level networks.

Second Stage: Latent Scale Gaussian Process Regression

Once the high-dimensional network has been projected onto the lower dimensional latent scale space in the first stage, we embed these latent scales (along with supplementary covariates $\mathbf{z}$) into a Gaussian process regression framework for predicting the clinical outcome as:

$$y_i = \phi(\Lambda_i^{1/2} U_i, a_i, \mathbf{z}_i) + \epsilon_i,\ \epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \tau^{-1}),\ i = 1, \cdots, n. \tag{2.3.2}$$

where ϵ_i denotes the residual error normally distributed with precision $\tau \sim \pi(\tau)$, $\phi(\cdot)$ denotes the unknown mean that is a function of the brain network via the estimated weighted latent scales $\Lambda_i^{1/2} U_i$ and intercept a_i derived via model (2.3.1), as well as supplementary demographics and environmental exposures $\mathbf{z}_i$. We note that the parameters $(\Lambda_i^{1/2} U_i, a_i)$ capture the entirety of information about the network for the i-th sample, and hence are collectively included in the Gaussian process regression. The function $\phi(\cdot)$ is assumed to have a Gaussian process prior with mean $\mathbf{0}$ and covariance kernel K that has the following squared-exponential structure:

$$K(i, i') = \psi_1 exp\Big\{ -\psi_u \big|\big|\Lambda_i^{1/2} U_i - \Lambda_{i'}^{1/2} U_{i'} \big|\big|_F^2 - \psi_a (a_i - a_{i'})^2 - \psi_z \big|\big|\mathbf{z}_i - \mathbf{z}_{i'} \big|\big|_2^2 \Big\},\ i \neq i' \tag{2.3.3}$$

where $||\cdot||_F$ and $||\cdot||_2$ denote the Frobenius and L_2 norms respectively, ψ_1 denotes the scale parameter controlling the variance of the mean function, and ψ_u, ψ_a and ψ_z respectively denote the distinct lengthscale parameters corresponding to the latent scales, the intercept term and the supplementary covariates that control the smoothness of the mean function under the Gaussian process. The Gaussian process prior for the mean function allows flexible non-linear relationships between the outcome and the covariates, and can also accommodate unknown interactions between the network and supplemental covariates that is crucial in order to achieve good prediction accuracy. We note that the Frobenius norm in the first term in (2.3.3) represents the distance between matrices, whereas the L_2 norm represents the distance between vectors.

Node Selection with Respect to Outcome

Given the node-specific latent scales representation of the network, we are also interested in investigating which nodes in the network contribute to significant differences with respect to the outcome of interest. To achieve this, we extend the proposed model in (2.3.2) by using a modified kernel function as

$$K(i,i') = \psi_1 exp\Bigg\{ -\psi_u \Bigg[\sum_{j=1}^{p} \beta_j \big|\big| \Lambda_i^{1/2} \mathbf{u}_{ij} - \Lambda_{i'}^{1/2} \mathbf{u}_{i'j} \big|\big|_2^2 \Bigg] - \psi_a (a_i - a_{i'})^2 - \psi_z \big|\big| \mathbf{z}_i - \mathbf{z}_{i'} \big|\big|_2^2 \Bigg\}, \; i \neq i', \tag{2.3.4}$$

where β_j represents the overall contribution of node j in the network towards modeling the outcome. If nodes k and l are both significant related to the outcome (i.e. β_k and β_l are non-zero), then it automatically implies that edge (k,l) is significant. However even then, it is possible for subsets of edges associated with significant nodes $\{k : \beta_k \neq 0\}$ to be unrelated to the outcome, since β_k represents the node level contributions corresponding to the k-th node and is not equipped to directly identify edge-level associations.

Similar to Savitsky et al. (2011), we assume $\beta_j = -\log(\rho_j)$ where $\rho_j \in [0,1]$ for $j = 1, \cdots, p$. Further we assume a spike-and-slab prior on ρ_j's as

$$\pi(\rho_j|\gamma_j) = \gamma_j \mathrm{I}\{0 < \rho_j \leq 1\} + (1-\gamma_j)\delta_1(\rho_j), \; \gamma_j \sim Ber(\pi^*), \; \pi^* \sim Beta(a_{\pi^*}, b_{\pi^*}), \tag{2.3.5}$$

where $\delta_1(\cdot)$ denotes a point mass at 1, in which case the corresponding β parameter would have a point mass at 0. Equation (2.3.5) specifies that with probability $1 - \pi^*$ (that is unknown), the j-th network node will have no effect on the regression model (i.e. $\beta_j = 0$), while it will have a non-negligible influence with probability π^*. The unknown prior inclusion probability is estimated under a Beta hyperprior. One can perform feature selection at the node level by including those nodes whose posterior inclusion probabilities lie above a certain threshold, where the threshold can be chosen as 0.5, or can be determined adaptively using a post-hoc approach that controls for false discovery rates as in Kundu et al. (2019b). The spike and slab specification is designed to result in dimension reduction via a node selection procedure that completely eliminates the contribution of all network edges related to nodes

that are not significantly associated with the outcome. The feature selection at the node level (rather than the edge level), is more suitable in high-dimensional network studies (such as in our neuroimaging applications), and is motivated by the fact that node level summaries are commonly used to study brain network properties (Higgins et al., 2019), which justifies this approach. Hence, the proposed approach works best when the association between the network and the outcome can be primarily characterized via node level network measures.

2.3.2 Computation Framework

The estimation of the first stage model is implemented through an EM algorithm with data augmentation utilizing Theorem 1 in Polson et al. (2013). The EM algorithm is more computationally efficient compared to the previous Markov chain Monte Carlo (MCMC) implementation (Hoff, 2005), and leads to significant speed-ups in applications involving high-dimensional networks. The details of the EM algorithm can be found below. For completeness, we have also included details of the MCMC algorithm for the first stage model in the Appendices, but the MCMC based results for the first stage model are not included in the report. Conditional on the estimated latent scales in the first stage, we use a MCMC sampling scheme to estimate the parameters for the second stage regression model. For the MCMC sampling scheme, the scale parameters $\tau \sim Ga(a_\tau, b_\tau)$ and $\psi_1 \sim Ga(a_{\psi_1}, b_{\psi_1})$ are assigned conjugate priors and updated under closed form posteriors. The lengthscale parameters ψ_u, ψ_a and ψ_z are updated via Metropolis-Hastings steps. For the posterior computation involving node selection in model (2.3.4), the MCMC proceeds as in model (2.3.2) with additional steps included to update γ's, ρ's, $\beta's$ and π^*, as described in the sequel.

EM Algorithm for the First Stage Model

Denoting $\{\mathbf{e}\} = \{\mathbf{e}_1, \ldots, \mathbf{e}_n\}$, $\{a\} = \{a_1, \ldots, a_n\}$, $\{U\} = \{U_1, \ldots, U_n\}$, we can express the likelihood as follows:

$$\pi\big(\{\mathbf{e}\}\big|\{a\}, \{U\}, \Lambda\big) = \prod_{i=1}^{n} \prod_{k<l,k,l=1}^{p} \frac{\Big[exp\big(a_i + \mathbf{u}_{ik}^T \Lambda_i \mathbf{u}_{il}\big)\Big]^{e_{i,kl}}}{\Big[1 + exp\big(a_i + \mathbf{u}_{ik}^T \Lambda_i \mathbf{u}_{il}\big)\Big]}, k < l, k, l = 1, \ldots, p.$$

Using Theorem 1 in Polson et al. (2013), we can introduce edge-specific latent Pólya-Gamma (PG(0, 1)) variables $\omega_i = \{\omega_{i,kl} : k < l\}, i = 1, \cdots, n$ and denote $\{\omega\}$=$\{\omega_1, \ldots, \omega_n\}$, and write the augmented likelihood as:

$$\pi\big(\{\mathbf{e}\}\big|\{a\}, \{U\}, \Lambda_i, \{\omega\}\big) = \prod_{i=1}^{n} \prod_{k<l,k,l=1}^{p} \frac{1}{2} \exp\big\{(e_{i,kl} - 0.5)\big(a_i + \mathbf{u}_{ik}^T \Lambda_i \mathbf{u}_{il}\big) - 0.5\omega_{i,kl}\big(a_i + \mathbf{u}_{ik}^T \Lambda_i \mathbf{u}_{il}\big)^2\big\}.$$

The posterior distribution is proportional to the product of the likelihood and priors $\pi(\{a\})$, $\pi(\{U\})$, $\pi(\Lambda_i)$, $\pi(\{\omega\})$ as specified earlier. The EM algorithm treats the log-posterior as the objective function and maximizes it to obtain the MAP (maximum a posteriori) estimates for a_i, U_i and Λ_i via the M-step, while treating $\{\omega\}$ as missing variables that are imputed via the E-step. The q-th iteration of the EM algorithm is described below.

E step: We calculate the conditional expectation of the Pólya–Gamma variables as

$$\omega_{i,kl}^{(q)} = \frac{1}{2\delta_{i,kl}^{(q-1)}} \left[\frac{e^{\delta_{i,kl}^{(q-1)}} - 1}{e^{\delta_{i,kl}^{(q-1)}} + 1}\right], \; \delta_{i,kl}^{(q-1)} = a_i^{(q-1)} + \mathbf{u}_{ik}^{(q-1)T} \Lambda_i^{(q-1)} \mathbf{u}_{il}^{(q-1)}, k < l, k, l = 1, \ldots, p.$$

and also for the diagonal elements in the weight matrix as

$$\lambda_{ir}^{(q)} = \frac{\pi^{(q-1)} \mathcal{L}_1}{\pi^{(q-1)} \mathcal{L}_1 + (1 - \pi^{(q-1)}) \mathcal{L}_0}$$

where

$$\mathcal{L}_1 = \prod_{k<l,k,l=1}^{p} \frac{1}{2} \exp\big\{(e_{i,kl} - 0.5)\big(a_i^{(q-1)} + \mathbf{u}_{ik}^{(q-1)T} \Lambda_i^{r1} \mathbf{u}_{il}^{(q-1)}\big) - 0.5\omega_{i,kl}^{(q)}\big(a_i^{(q-1)} + \mathbf{u}_{ik}^{(q-1)T} \Lambda_i^{r1} \mathbf{u}_{il}^{(q-1)}\big)^2\big\}$$

and $\Lambda_i^{r1} = diag(1, \lambda_{i2}^{(q)}, \cdots, \lambda_{i(r-1)}^{(q)}, 1, \lambda_{i(r+1)}^{(q-1)}, \cdots, \lambda_{id}^{(q-1)})$; also on the other hand,

$$\mathcal{L}_0 = \prod_{k<l,k,l=1}^{p} \frac{1}{2} \exp\left\{(e_{i,kl} - 0.5)\left(a_i^{(q-1)} + \mathbf{u}_{ik}^{(q-1)T} \Lambda_i^{r0} \mathbf{u}_{il}^{(q-1)}\right) - 0.5\omega_{i,kl}^{(q)}\left(a_i^{(q-1)} + \mathbf{u}_{ik}^{(q-1)T} \Lambda_i^{r0} \mathbf{u}_{il}^{(q-1)}\right)^2\right\}$$

and $\Lambda_i^{r0} = diag(1, \lambda_{i2}^{(q)}, \cdots, \lambda_{i(r-1)}^{(q)}, 0, \lambda_{i(r+1)}^{(q-1)}, \cdots, \lambda_{id}^{(q-1)})$. The diagonal elements are updated in sequence from index 2 to d.

M step: We plug in $\{\omega_{i,kl}^{(q)} : k < l\}$ and $\{\lambda_{ir}^{(q)} : r = 2, \cdots, d\}$ to obtain the values for the remaining model parameters as those that maximize the augmented log-posterior. We first find the estimate for a_i $(i = 1, \ldots, n)$ as:

$$a_i^{(q)} = \left(\sum_{k<l,k,l=1}^{p} \left[e_{i,kl} - 0.5 - \omega_{i,kl}^{(q)} \mathbf{u}_{ik}^{(q-1)T} \Lambda_i^{(q)} \mathbf{u}_{il}^{(q-1)}\right]\right)\left(\sigma_a^{-2} + \sum_{k<l,k,l=1}^{p} \omega_{i,kl}^{(q)}\right)^{-1}$$

As noted earlier, the first element of every latent scale is fixed at a certain pre-specified value b and does not need to be updated. Thus we denote latent scale for the kth node omitting the first element as $\mathbf{u}_{ik(-1)}$. Correspondingly, we fix the first diagonal element of Λ_i at 1 and denote the diagonal weight matrix omitting the first row and column of Λ_i as Λ_{i0}. The latent scales are updated iteratively for $k = 1, \cdots, p$ as $\mathbf{u}_{ik(-1)}^{(q)} = A_{ik}^{-1} B_{ik}$ where

$$\begin{aligned}
A_{ik} &= \sum_{1\le j<k} \left[\omega_{i,jk}^{(q)} \Lambda_{i0}^{(q)} \mathbf{u}_{ij(-1)}^{(q)} \mathbf{u}_{ij(-1)}^{(q)T} \Lambda_{i0}^{(q)} \right. \\
&+ \left. \sigma_u^{-2} \mathrm{I}_{(d-1)}\right] + \sum_{k<j\le p} \left[\omega_{i,jk}^{(q)} \Lambda_{i0}^{(q)} \mathbf{u}_{ij(-1)}^{(q-1)} \mathbf{u}_{ij(-1)}^{(q-1)T} \Lambda_{i0}^{(q)} + \sigma_u^{-2} \mathrm{I}_{(d-1)}\right] \\
B_{ik} &= \sum_{1\le j<k} \left[e_{i,jk} - 0.5 - (a_i^{(q)} + b^2)\omega_{i,jk}^{(q)}\right] \Lambda_{i0}^{(q)} \mathbf{u}_{ij(-1)}^{(q)} \\
&+ \sum_{k<j\le p} \left[e_{i,jk} - 0.5 - (a_i^{(q)} + b^2)\omega_{i,jk}^{(q)}\right] \Lambda_{i0}^{(q)} \mathbf{u}_{ij(-1)}^{(q-1)}
\end{aligned}$$

Finally, the prior inclusion probability π corresponding to the channels is updated as $\pi^{(q)} = (a_\pi - 1 + \sum_{r=2}^{d} \lambda_{ir}^{(q)})/(a_\pi + b_\pi + d - 3)$.

Gibbs Sampler for Second Stage Parameters

We denote the Gaussian process atoms as $\boldsymbol{\phi} = \left(\phi(\hat{a}_1, \hat{\Lambda}_1^{1/2} \hat{U}_1, \boldsymbol{z}_1), \cdots, \phi(\hat{a}_n, \hat{\Lambda}_n^{1/2} \hat{U}_n, \boldsymbol{z}_n)\right)^T$. The algorithm iterates between the following steps:

1. Update the noise precision τ from Gamma distribution with parameters $(a_\tau + 0.5n)$ and $(b_\tau + 0.5\big|\big|\boldsymbol{y} - \boldsymbol{\phi}\big|\big|_2^2)$.

2. Update the global scale parameter ψ_1 from inverse Gamma distribution with parameters $(a_{\psi_1} + 0.5n)$ and $(b_{\psi_1} + 0.5\boldsymbol{\phi}^T E_0^{-1}\boldsymbol{\phi})$ where $E_0 = \exp\Big(-\psi_u E_u - \psi_a E_a - \psi_z E_z\Big)$. Here we define $E_u(i,i') = \big|\big|\hat{\Lambda}_i^{1/2}\hat{U}_i - \hat{\Lambda}_{i'}^{1/2}\hat{U}_{i'}\big|\big|_F^2$, $E_a(i,i') = (\hat{a}_i - \hat{a}_{i'})^2$, $E_z(i,i') = \big|\big|\boldsymbol{z}_i - \boldsymbol{z}_{i'}\big|\big|_2^2$, $i, i' = 1, \cdots, n$.

3. Draw a candidate ψ_u^* where $log\psi_u^* \sim \mathrm{N}(log\psi_u, 0.01^2)$. Accept the candidate with probability

$$min\left(1, \frac{\big|\psi_1 E_0^* + \tau^{-1}\mathrm{I}_n\big|^{-0.5}\exp\{-0.5\boldsymbol{y}^T(\psi_1 E_0^* + \tau^{-1}\mathrm{I}_n)^{-1}\boldsymbol{y}\}}{\big|\psi 1 E_0 + \tau^{-1}\mathrm{I}_n\big|^{-0.5}\exp\{-0.5\boldsymbol{y}^T(\psi 1 E_0 + \tau^{-1}\mathrm{I}_n)^{-1}\boldsymbol{y}\}}\right)$$

   where $E_0^* = \exp\Big(-\psi_u^* E_u - \psi_a E_a - \psi_z E_z\Big)$. Same procedure for updating ψ_a and ψ_z.

4. Update the Gaussian process atoms $\boldsymbol{\phi}$ from multivariate normal distribution with mean $\Big[\tau^{-1}\psi_1^{-1}E_0^{-1} + \mathrm{I}_n\Big]^{-1}\boldsymbol{y}$ and covariance $\Big[\psi_1^{-1}E_0^{-1} + \tau\mathrm{I}_n\Big]^{-1}$.

For our additional node selection analysis using modified kernel function, we need two more steps in the Gibbs sampler to update γ_j, ρ_j and β_j for $j = 1, \cdots, p$ in sequence, as well as update the hyperparameter π^* as illustrated below:

5. Update γ_j, ρ_j and β_j for $j = 1, \cdots, p$ in sequence. While keeping parameters for $k \neq j$ fixed, we have two moves to make.

   - Between-model move: if currently $\gamma_j = 1$, we propose to have $\gamma_j^{'} = 0$, $\rho_j^{'} = 1$ and $\beta_j^{'} = 0$; otherwise if currently $\gamma_j = 0$, we propose to have $\gamma_j^{'} = 1$, draw $\rho_j^{'}$ from $Unif(0,1)$ and let $\beta_j^{'} = -\log(\rho_j^{'})$. We accept the proposal with probability

$$\min\left\{1, \frac{\pi(\gamma_j^{'})\big|\psi_1 E_j^{'} + \tau^{-1}\mathrm{I}_n\big|^{-0.5}\exp\{-0.5\boldsymbol{y}^T(\psi_1 E_j^{'} + \tau^{-1}\mathrm{I}_n)^{-1}\boldsymbol{y}\}}{\pi(\gamma_j)\big|\psi_1 E_j + \tau^{-1}\mathrm{I}_n\big|^{-0.5}\exp\{-0.5\boldsymbol{y}^T(\psi_1 E_j + \tau^{-1}\mathrm{I}_n)^{-1}\boldsymbol{y}\}}\right\}$$

     where $E_j = \exp\Big(-\psi_u E_u^* - \psi_a E_a - \psi_z E_z\Big)$, $E_u^*(i,i') = \big|\big|\hat{\Lambda}_i^{1/2}\hat{U}_i B - \hat{\Lambda}_{i'}^{1/2}\hat{U}_{i'} B\big|\big|_F^2$ and $B = diag(\sqrt{\beta_1}, \cdots, \sqrt{\beta_p})$ is a diagonal matrix with the j-th diagonal element

as $\sqrt{\beta_j}$. On the other hand, we have $E_j^{'} = \exp\Big(- \psi_u E_u^{*'} - \psi_a E_a - \psi_z E_z\Big)$, $E_u^{*'}(i,i') = \big|\big|\hat{\Lambda}_i^{1/2}\hat{U}_i B^{'} - \hat{\Lambda}_{i'}^{1/2}\hat{U}_{i'} B^{'}\big|\big|_F^2$ and $B^{'} = diag(\sqrt{\beta_1}, \cdots, \sqrt{\beta_p})$ where the j-th diagonal element is $\sqrt{\beta_j^{'}}$.

- Within-model move: this move is only triggered when we sample $\gamma_j^{'} = 1$. Then we further draw another $\rho_j^{''}$ from $Unif(0,1)$ and $\beta_j^{''} = -\log(\rho_j^{''})$. And we accept this proposal with probability

$$\min\left\{1, \frac{\big|\psi_1 E_j^{''} + \tau^{-1}\mathrm{I}_n\big|^{-0.5}\exp\{-0.5\boldsymbol{y}^T(\psi_1 E_j^{''} + \tau^{-1}\mathrm{I}_n)^{-1}\boldsymbol{y}\}}{\big|\psi_1 E_j^{'} + \tau^{-1}\mathrm{I}_n\big|^{-0.5}\exp\{-0.5\boldsymbol{y}^T(\psi_1 E_j^{'} + \tau^{-1}\mathrm{I}_n)^{-1}\boldsymbol{y}\}}\right\}$$

and here $E_j^{''} = \exp\Big(- \psi_u E_u^{*''} - \psi_a E_a - \psi_z E_z\Big)$, $E_u^{*''}(i,i') = \big|\big|\hat{\Lambda}_i^{1/2}\hat{U}_i B^{''} - \hat{\Lambda}_{i'}^{1/2}\hat{U}_{i'} B^{''}\big|\big|_F^2$ and $B^{''} = diag(\sqrt{\beta_1}, \cdots, \sqrt{\beta_p})$ where the j-th diagonal element is $\sqrt{\beta_j^{''}}$.

6. Update the prior inclusion probability π^* from Beta distribution with parameters $(a_{\pi^*} + \sum_{j=1}^p \gamma_j)$ and $(b_{\pi^*} + p - \sum_{j=1}^p \gamma_j)$.

2.3.3 Prediction for Testing Samples

The prediction for additional subjects involves two steps. First step takes the upper triangular vector of binary network matrices $\boldsymbol{e}_1^*, \cdots, \boldsymbol{e}_m^*$ as input of first stage model and obtain the estimates of intercept and latent scales. These estimates are then used in the second step, together with other supplementary covariates, as input of second stage model and eventually obtain the predicted values of the response variable. From the property of Gaussian process, the response of the additional subjects $\boldsymbol{y}^* = (y_1^*, \cdots, y_m^*)^T$ and the response

of the original n subjects $\boldsymbol{y}$ jointly follow a multivariate normal distribution as

$$\begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{y}^* \end{pmatrix} \sim \mathrm{N}\Big(\mathbf{0}_{(n+m)}, \begin{pmatrix} K + \tau^{-1}\mathrm{I}_n & K_*^T \\ K_* & K_{**} + \tau^{-1}\mathrm{I}_n \end{pmatrix}\Big),$$

$$K_*(q,i) = \psi_1 exp\Big\{ -\psi_u \big|\big|\Lambda_q^{1/2}U_q^* - \Lambda_i^{1/2}U_i\big|\big|_F^2 - \psi_a(a_q^* - a_i)^2 - \psi_z \big|\big|\boldsymbol{z}_q^* - \boldsymbol{z}_i\big|\big|_2^2 \Big\},$$

$$K_{**}(q,q') = \psi_1 exp\Big\{ -\psi_u \big|\big|\Lambda_q^{1/2}U_q^* - \Lambda_{q'}^{1/2}U_{q'}^*\big|\big|_F^2 - \psi_a(a_q^* - a_{q'}^*)^2 - \psi_z \big|\big|\boldsymbol{z}_q^* - \boldsymbol{z}_{q'}^*\big|\big|_2^2 \Big\},$$

$$i = 1, \cdots, n,\ q, q' = 1, \cdots, m$$

When working with the modified kernel (2.3.4), we need to replace the $\Lambda^{1/2}U$ in the covariance kernel with $\Lambda^{1/2}UB$ where B is a diagonal matrix of $\sqrt{\beta_1}, \cdots, \sqrt{\beta_p}$. Then the conditional mean can be expressed as:

$$\mathbf{E}(\boldsymbol{y}^*|\boldsymbol{y}) = K_*(K + \tau^{-1}\mathrm{I}_n)^{-1}\boldsymbol{y}$$

The expression above is based on the marginal distribution of the outcome. We can also base the prediction expression on the Gaussian process atoms $\boldsymbol{\phi}$. Then the computation would be

$$\begin{pmatrix} \boldsymbol{\phi} \\ \boldsymbol{y}^* \end{pmatrix} \sim \mathrm{N}\Big(\mathbf{0}_{(n+m)}, \begin{pmatrix} K & K_*^T \\ K_* & K_{**} + \tau^{-1}\mathrm{I}_n \end{pmatrix}\Big),\ \mathbf{E}(\boldsymbol{y}^*|\boldsymbol{\phi}) = K_*K^{-1}\boldsymbol{\phi}$$

This expression can be used for in-sample prediction with the Gibbs sampler algorithm.

2.3.4 Hyper-parameter Selection

An optimal implementation of our method requires a careful choice of several hyperparameters in order to optimize performance. In the first stage EM algorithm, we set $\sigma_a^2 = \sigma_u^2 = 2$ for the priors on a and elements of U, which worked well across a range of hyper-parameter combinations and several network structures. We also present (in Appendix) the sensitivity analysis for different combinations of (σ_a^2, σ_u^2) in the first stage model, which illustrates the robustness in performance under reasonable choices of these hyper-parameters as long as

these values are not extremely large or small. For the weight matrix Λ, we can (i) fix it at identity matrix or (ii) update it under Bernoulli priors and examine the performance under two different hyperparameter choices on the prior inclusion probability π that is chosen to be $Beta(1, 10)$ or $Beta(1, 25)$ in order to facilitate a sensitivity analysis for channel selection. The prior on the spike and slab probability π^* set at $Beta(1, 1)$ that translated to a prior mean inclusion probability of 0.5 and hence it does not favor either inclusion or exclusion of the nodes.

For simulation studies, we performed sensitivity analysis for different choices of d in the working model, whereas for the GTP data analysis, we choose the number of channels using a data adaptive approach via cross-validation, in a manner that minimizes the prediction error under the second stage regression model in a validation sample. For the second stage Gibbs sampler, the prior on ψ_1 is set to inverse gamma $(0.1, 80)$ that results in overall good prediction performance and coverage at the risk of wider predictive intervals in some settings. However in the analysis of GTP data, we also implemented the method under $\psi_1 \sim Inv - Ga(0.1, 10)$ for sensitivity analysis. The prior on τ is pre-specified as Gamma $(0.1, 80)$ that mimics a non-informative variance, which enables good prediction and coverage.

2.4 Empirical Experiments

2.4.1 Simulation Studies

We considered different scenarios corresponding to varying true network structures. Scenarios 1 and 2 correspond to scale-free and small-world networks respectively that are frequently encountered in real-world neuroimaging applications, whereas Scenario 3 uses the networks obtained from resting state fMRI data from our GTP application. When generating the data, a latent scale model (2.3.1) was first fit to the true network in order to obtain estimates of latent scales U, the intercept term a and the weight matrix Λ. Subsequently, the response was generated as $y = 0.5a + \exp(\mathbf{b}_1 \Lambda \mathbf{u}_{s_1}) + \sum_{j=2}^{n_{act}} \mathbf{b}_j \Lambda \mathbf{u}_{s_j} + \epsilon_i^*, \epsilon_i^* \sim \mathrm{N}(0, 0.5^2)$,

where each vector in the set $\{\mathbf{b}_1, \ldots, \mathbf{b}_{n_{act}}\}$ had length d and was sparse with non-zero values as 1 or -1. The set $\{s_1, \cdots, s_{n_{act}}\}$ contained the indices of the active nodes, where the number of active nodes were chosen at three levels 10, 30, 50 for the simulations.

The total number of channels d for the true model was chosen as 3, while different values of d were used in the working model for sensitivity analysis. Moreover, we only present the simulation results for Λ fixed to identity matrix in the working model, since the performance with Bernoulli priors on the weight matrix for channel selection resulted in inferior prediction, although we did present a sensitivity analysis with channel selection in the analysis of GTP data reported in the next section. Supplementary covariates were not included for our simulation examples in Scenario 1-3, although we did include additional covariates (such as trauma exposure) in our analysis of GTP data as detailed in the sequel. For Scenarios 1-2, the training and test sample sizes were 50 each, whereas for Scenario 3, the training and test sample sizes correspond to 41 and 40 respectively corresponding to the GTP dataset.

True Network Generation

We used 100 nodes for the simulated networks in Scenarios 1 and 2 that is common in brain connectome literature (Lukemire et al., 2020), while we used 264 nodes from the Power atlas for Scenario 3. Hence our studies showcase the generalizability of the proposed method across with varying network sizes. The network generation is described in detail below.

Scenario 1: Here, scale-free networks were generated by function `sample_pa` in `R` package `igraph` (Csardi et al., 2006). The number of nodes was set to be 100 (p) with size of initial graph varying between 2 and 10, and number of edges to add in each time step varying between 4 and 10.

Scenario 2: Here, the network was generated with small-world structure using function `watts.strogatz.game` in `R` package `igraph`. The size of starting lattice was kept at 1 and the total size of lattice was set to 100 (p). The rewiring probability was kept at 0.5 while the neighborhood size varied between 4 and 12.

Scenario 3: Here, we generated the binary network based on resting state fMRI connectivity using GTP data. A description of the study can be found in the PTSD Data Application section. We calculated the resting state network for each participant corresponding to the Power atlas with 264 nodes using the graphical lasso algorithm (Friedman et al., 2008) under varying sparsity levels corresponding to regularization parameter values $\lambda = 0.05, 0.10$, and 0.15, with a larger λ value corresponding to a sparser network.

We note that the current data generation settings in Scenario 1 yields simulated networks with density lying between 0.07 and 0.19; while for Scenario 2, the network density varies between 0.08 and 0.24. For Scenario 3 involving the GTP data, the network density varies between [0.18, 0.25] corresponding to $\lambda = 0.05$ and it decreases for higher values of λ. These simulated network densities represent acceptable levels of sparsity in brain networks that are encountered in literature (see Hallquist and Hillary (2018)), which typically vary between 5% to 25%.

Competing Methods and Performance Metrics

We denote the proposed latent scale Gaussian process regression approach without node selection as lsGPR, and denote the corresponding version with node selection via spike and slab priors as sparse lsGPR or sp-lsGPR. Our proposed method was compared to the linear shrinkage methods including lasso (Tibshirani, 1996), ridge (Hoerl and Kennard, 1970), elastic net (Zou and Hastie, 2005), and Bayesian horseshoe prior (Carvalho et al., 2010), which all used the full edge set as the predictors. The models were implemented through `R` packages `glmnet` (Friedman et al., 2010) and `monomvn` (Gramacy, 2018). We also compared with non-linear approaches that used GPR on the full edge set (edge-GPR), on the reduced representation from principal component analysis (pca-GPR), and on the reduced representation from Laplacian Eigenmap (Belkin and Niyogi, 2003) (mf-GPR). The dimension reduction of the pca-GPR and mf-GPR methods were implemented using `R` function `prcomp` and `R` package `dimRed` (Kraemer et al., 2018) respectively. We used the squared exponential kernel (Rasmussen and Williams, 2006) for all GPR approaches, with similar priors on the scale parameter of the kernel for fair comparisons. We evaluated

prediction performance in terms of out of sample predictive MSE, as well as coverage and 95% predictive interval widths for the test samples under all Bayesian approaches. Here the coverage was defined as the ratio of test samples where the predictive intervals covered the observed value of these test samples to the total number of test samples. We also evaluated the node selection performance for our method and compared to Bayesian horseshoe prior method in terms of the area under the curve (AUC). For computing the AUC, we varied the threshold for posterior inclusion probabilities in order to detect significant nodes under the proposed sparse lsGPR method. Similarly for the Bayesian horseshoe that is a continuous shrinkage approach, we denoted a node as significant if one or more edges associated with that particular node was significant, where an edge is defined as significant if the estimated regression coefficient had absolute value greater than some threshold. In addition to AUC, the ROC curves themselves are also presented for the simulation examples in the Appendix. The total number of MCMC iterations for the second stage method was 10,000 with a burn in of 5000 under all Bayesian methods.

We note that the above competing methods are designed to cover a gamut of potential competitors that are state-of-the art in existing literature. The linear regression approaches are perhaps most widely used in literature. Moreover, the comparisons with alternate Gaussian process regression approaches with and without dimension reduction are natural in the context of the proposed method that involves dimension reduction via latent scales coupled with a Gaussian process regression framework. Such comparisons are particularly pertinent in the context of previous literature that suggests that Gaussian process regression methods which involve overly large number of features without dimension reduction perform sub-optimally for high-dimensional regression problems (Jiang et al., 2007).

Results

The prediction and coverage results for Scenarios 1-2 are presented in Figure 2.4. The horizontal axis represents the different levels for number of active nodes involved in generating the response. From the boxplots, it is immediately clear that both the lsGPR and the sparse lsGPR have superior predictive performance that is significantly improved compared to ex-

isting methods under Scenarios 1-2. Although the lsGPR including all nodes has greater predictive accuracy compared to sparse lsGPR for some cases, it is important to note that the sparse lsGPR method consistently has greater coverage accuracy compared to lsGPR, that points to a better characterization of predictive uncertainty under node selection via the spike and slab prior compared to when all nodes are included. The sparse lsGPR consistently has greater coverage compared to different approaches with varying number of active nodes in almost all settings under Scenarios 1-2. Moreover, the credible interval width is reasonable and often lower than competing GP regression approaches that use alternate dimension reduction methods. In contrast, the edge level GPR has extremely narrow predictive intervals that is impractical and results in very poor coverage. Finally, the sparse lsGPR method has a significantly higher AUC compared to the Bayesian horseshoe consistently across different number of active nodes, although the node selection performance becomes less accurate as the number of active nodes increases, which is expected.

Figure 2.5 illustrates the results of Scenario 3 where the horizontal axis represents the number of active nodes as in Figure 2.4, and the three rows correspond to the three levels of the regularization parameter λ of the graphical lasso in obtaining the binary networks, with higher values implying a sparser network. The sparse lsGPR method results in significantly improved prediction accuracy compared to all approaches, including the lsGPR method involving all nodes. Further, the coverage under the sparse lsGPR method is significantly improved compared to all other methods for the majority of settings, although the pcaGPR and mfGPR methods have slightly improved coverage in a few cases that results from extremely wide predictive intervals under these methods, which may not be desirable. Moreover, both the edgeGPR and the lsGPR involving all nodes have extremely poor predictive coverage that results from narrow predictive intervals, which is not desirable. In addition to superior performance in prediction and coverage, our proposed method also shows a consistent advantage in terms of node selection based on the AUC metric over the Bayesian horseshoe prior method, as in Scenarios 1-2. Furthermore, the ROC curves presented in Figure 2.12 in the Appendix also clearly show the superior sensitivity and specificity for variable selection under the proposed approach across all network sparsity

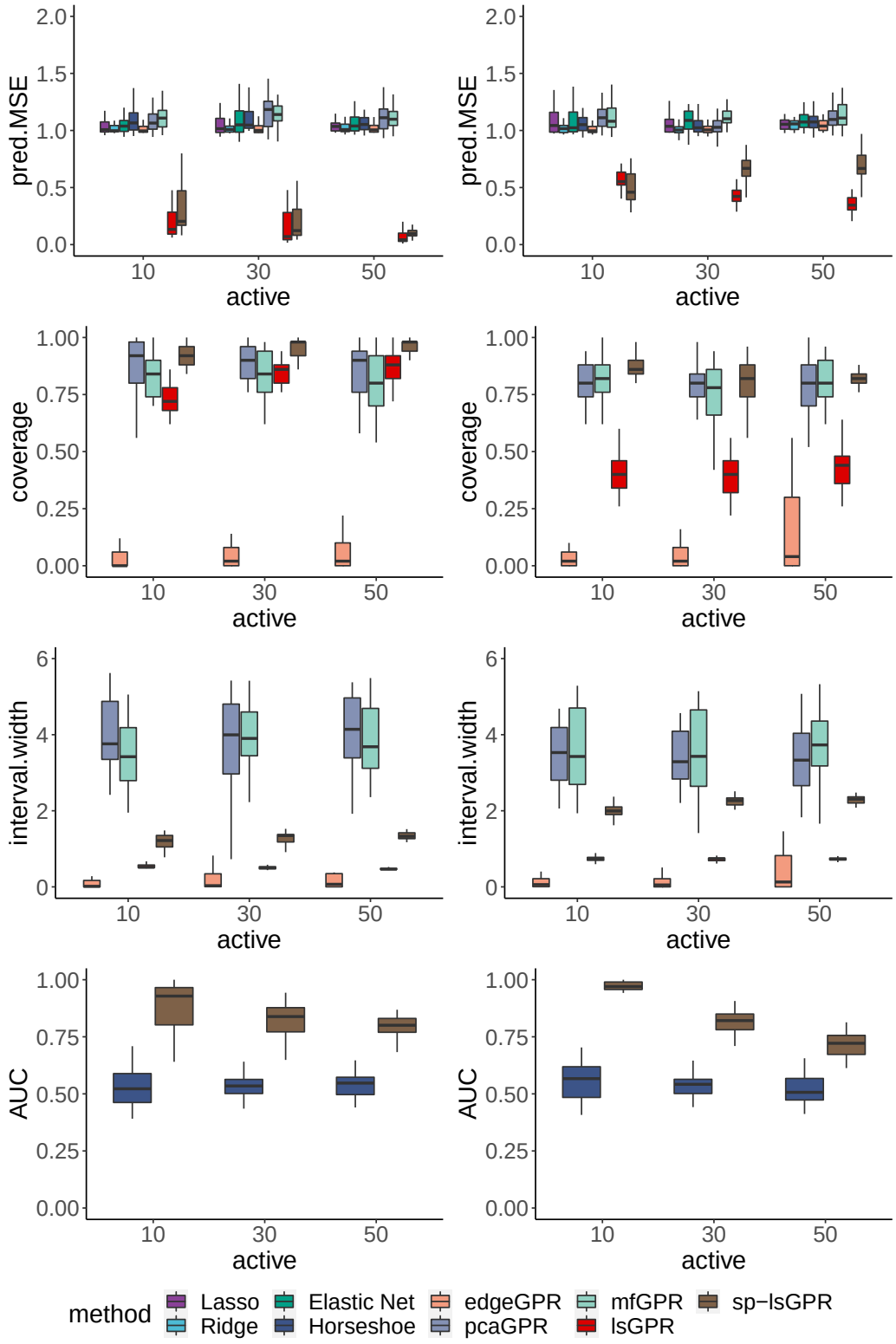

Figure 2.4: Boxplots for MSE, coverage, credible interval width and node selection AUC with varying number of activated nodes, under simulation Scenario 1 (left column) and Scenario 2 (right column).

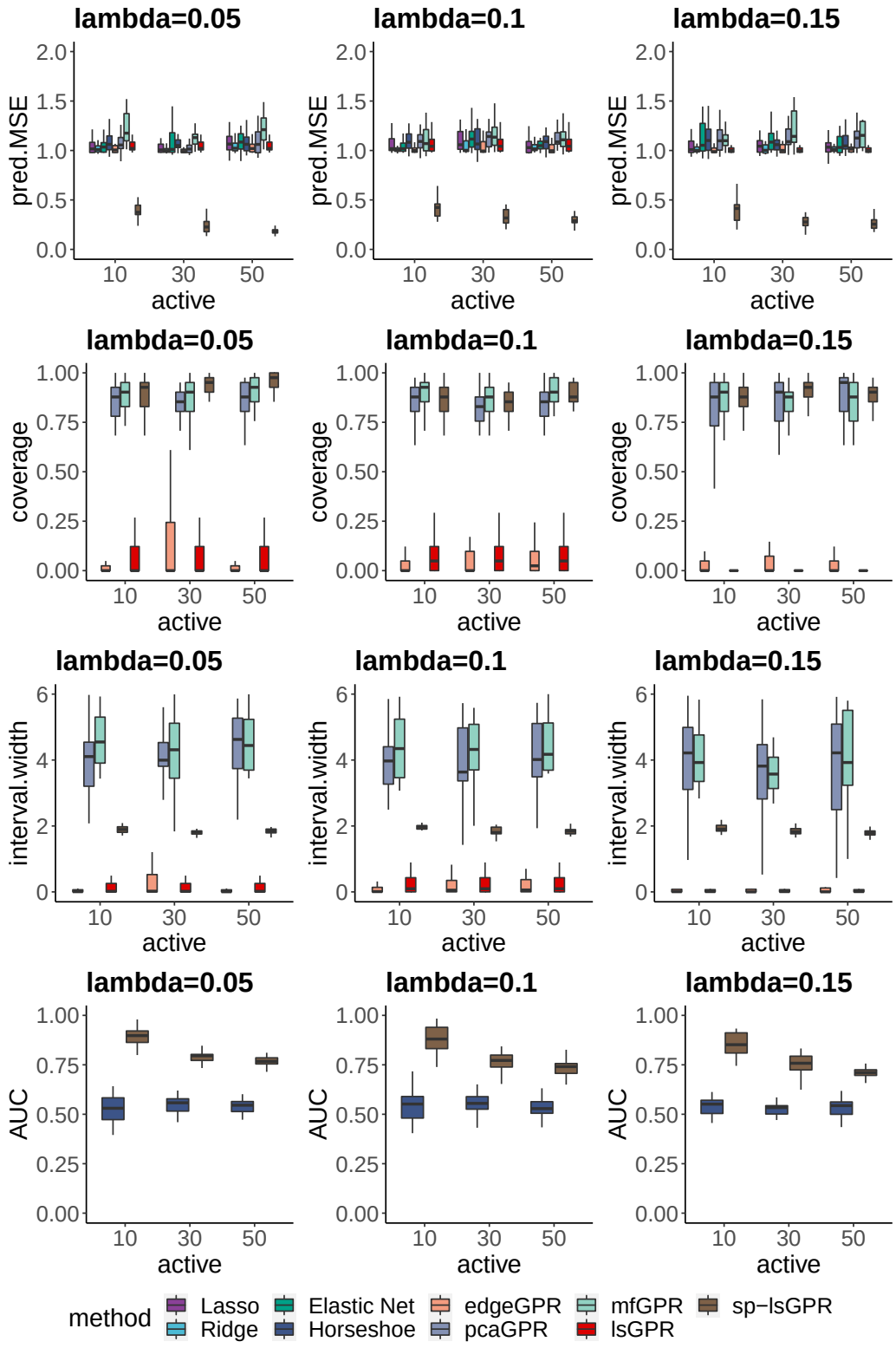

Figure 2.5: Boxplots for MSE, coverage, credible interval width , and AUC under Scenario 3 involving GTP networks with varying sparsity levels and different number of activated nodes.

levels in Scenario 3.

In summary, the lsGPR and the sparse lsGPR with node selection has significantly improved prediction performance compared to all approaches for Scenarios 1-2, but the performance of lsGPR involving all nodes declines under Scenario 3 involving high-dimensional networks derived from real fMRI data. Moreover, the sparse lsGPR has a decided advantage over lsGPR involving all nodes in terms of characterizing predictive uncertainty as reflected by higher coverage under predictive intervals for test samples that have reasonable width. The edgeGPR method suffers from an overwhelmingly large number of edges, and the performance of the lsGPR method involving all nodes deteriorates when the number of nodes is increased in Scenario 3 (compared to Scenarios 1-2). Competing dimension reduction techniques relying on PCA or other manifold projections have inferior prediction performance, as well as inferior coverage for the majority of settings, in spite of having sufficiently wide predictive intervals, which suggests a poor ability to characterize predictive uncertainty. All linear methods have inferior predictive performance as expected, and the variable selection under Bayesian horseshoe results in inaccurate node selection that is potentially due to an overwhelmingly large number of edges in the high-dimensional networks considered.

We conducted a sensitivity test for the hyper-parameters σ_a^2 and σ_u^2 needed in the first stage EM algorithm. Table 2.5 in the Appendix reports the performance metrics for different hyper-parameter combinations under simulation scenario 1. From the table we can see that the suggested hyper-parameter values ($\sigma_a^2 = 2$, $\sigma_u^2 = 2$) enjoy a generally good performance in terms of prediction and variable selection. Also in general, the overall prediction and variable selection are not sensitive to the choice of hyper-parameters in the first stage EM algorithm, as long as some extreme choices are not used, for example, $\sigma_u^2 = 0.2$ or $\sigma_u^2 = 20$.

We also compared the performance of the proposed approach using EM algorithm in the first stage versus using MCMC for fitting the first stage model, under simulation scenario 1. The scatterplots of Figure 2.10 in the Appendix show high concordance between prediction and uncertainty quantification results under the latent scales derived from the EM algorithm and the MCMC approach for fitting the first stage model. Since both methods for estimating the first stage model leads to a comparable performance under the second stage regression

model (which is our main focus of interest), we recommend using the EM algorithm in the first stage since it is computationally scalable for high-dimensional networks.

Finally, a sensitivity analysis with varying number of channels in Figure 2.6 suggests that the performance of the sparse lsGPR method remains consistently better than competing approaches even when the value of d used in the working model is different than the number of true channels used when generating the data. However, the performance of the proposed approach in terms of node selection (AUC) seems to deteriorate significantly when the number of channels used in the working model is less than the true number of channels used for generating the data. Additional numerical examples are included in the Appendices section in Figure 2.11, which illustrate the effect of mis-specification of the number of channels. Based on these results, it is clear that the proposed approach performs best when the value of d used in the working model is at least as large as the true number of channels.

Moreover, Table 2.1 shows the computation time under stage 1 for different methods, corresponding to an implementation on a high performance computing (HPC) environment utilizing Intel Xeon CPU at 2.80GHz. The results in Table 2.1 are averaged over several simulations and only serve as approximation. The computation time for the Bayesian approaches are reported for 10,000 iterations. We observe that although the computation time increases almost linearly with the number of channels, the overall approach is computationally feasible for up to $d = 25$ or even $d = 50$ channels. However we don't anticipate requiring so many channels in order to obtain a good prediction performance under our approach. In our experience, a small to moderate number of channels in the first stage model is often adequate to deal with prediction involving high dimensional networks. We note that although the additional computational burden for fitting the first stage latent scale model results in an increase in the overall computation time that is higher than competing methods, the proposed approach is still scalable to high-dimensional networks of interest.

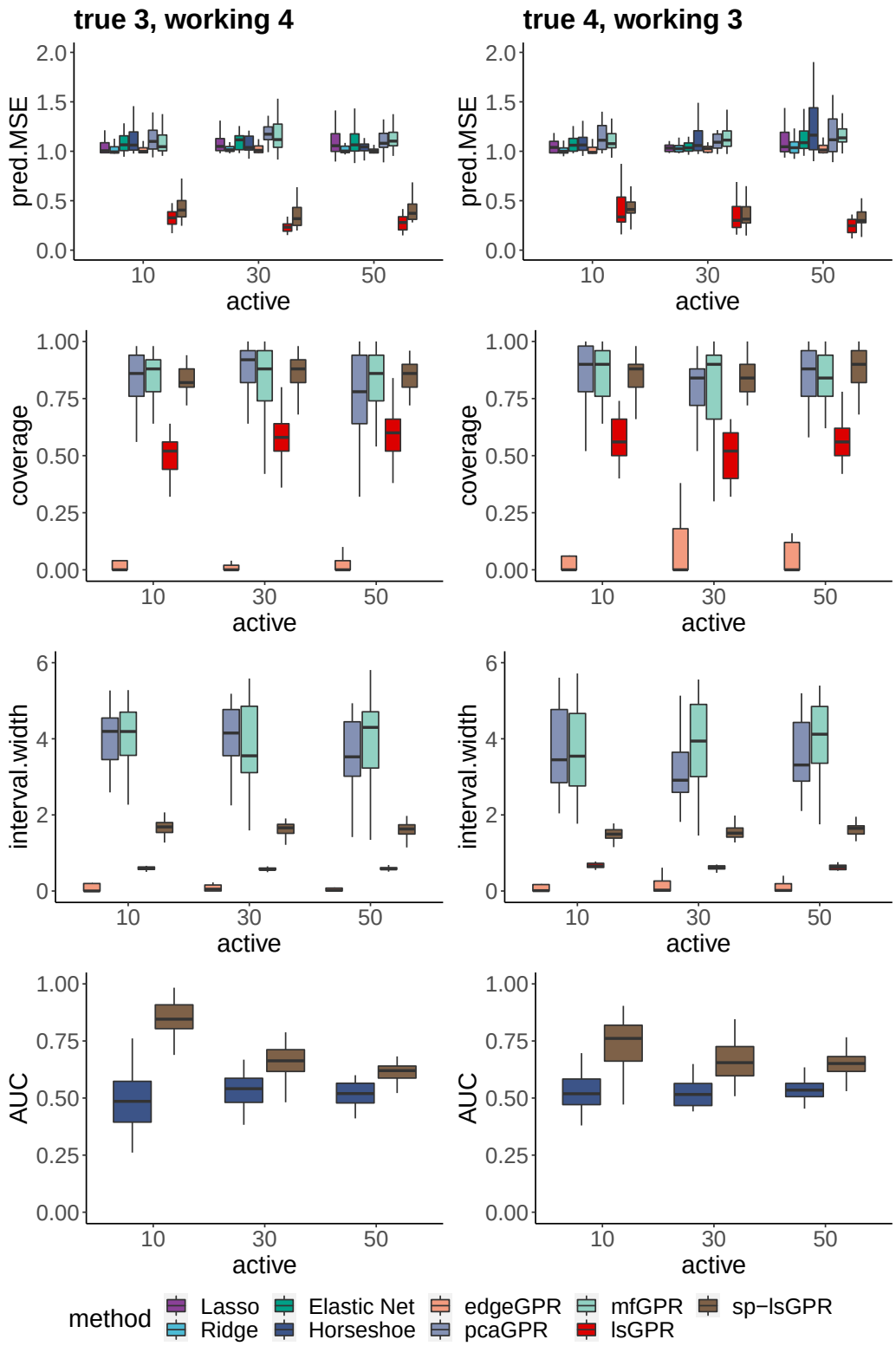

Figure 2.6: Sensitivity analysis on the number of channels under Scenario 1. The left column is under situation where the data generation uses 3 channels while estimation uses 4 channels. The right column is under situation where data generation uses 4 channels and estimation with 3 channels.

Table 2.1: Computation time summary for different number of channels under Scenario 1 with Stage1 and Stage2 referring to the two stages in our lsGPR method.

Method	*Lasso*	*Ridge*	*Elastic Net*	*Horseshoe*
CompTime	<1s	4s	10s	4min
Method	*edgeGPR*	*pcaGPR*	*mfGPR*	
CompTime	0.5min	0.5min	0.5min	
Method	*Stage1 (d=3)*	*Stage1 (d=5)*	*Stage1 (d=7)*	*Stage1 (d=10)*
CompTime	7.5min	9min	10.5min	13min
Method	*Stage1 (d=25)*	*Stage1 (d=50)*	*Stage2*	
CompTime	26min	58min	0.5min	

2.4.2 PTSD Data Application

Grady Trauma Project and Resilience Scores

The Grady Trauma Project (GTP) recruited African American females to study the risk factors for PTSD in a low-socioeconomic status (Stevens et al., 2013). The resting state functional magnetic resonance imaging (rs-fMRI) brain scans for each individual were obtained on a 3.0T Siemens Trio with echo-planar imaging (Siemens, Malvern, PA). T1-weighted anatomical scans were gathered with 176 contiguous 1mm sagittal slices using 3D MP-RAGE sequence (TR/TE/TI=2000/3.02/900 MS, $1mm^3$ voxel size). The functional images were collected in an ascending interleaved sequence with 37 3mm axial slices and no gap between slices (TR/TE=2000/30ms, FA=90°, $3mm^3$ voxel size).

We used the preprocessing script released from the 1000 Functional Connectomes Project to preprocess the brain images using standard steps. We first performed skull stripping on the T1-weighted images. Then we removed the first four volumes of the functional scans for signal stabilization, with 146 volumes remaining for the downstream preprocessing. We registered the T1-weighted image and functional images to the MNI standard space with a 6 mm FWHM Gaussian kernel for registration and smoothing. Motion corrections and removing of nuisance signals were also performed on the images. Finally, we put the functional images through band-pass filter to retain frequencies between 0.01 and 0.1 Hz.

After removing data with movement or drowsiness issues, we have 81 participants with available rs-fMRI data. For our analysis, we use the whole brain parcellation presented

in Power et al. (2011) for the brain images, involving 264 region of interest (ROIs). These regions are further organized into ten functional modules including motor, cingulo-opercular (CON), auditory, default mode (DMN), visual, fronto-parietal (FPN), salience (SAN), sub-cortical, ventral attention (VAN) and dorsal attention (DAN) (Cole et al., 2013). These functional modules have been assigned based on resting state fMRI studies (Power et al., 2011), which is well-suited for our data. Using these 264 regions (nodes), a network was computed separately for each individual using the graphical lasso algorithm (Friedman et al., 2008) with regularization parameter λ (higher λ represents networks with greater sparsity) and subsequently these networks were used for analysis. The GTP study has also acquired data on the Connor-Davidson Resilience Scale (Connor and Davidson, 2003) for measuring resilience as individual's ability to thrive in the face of adversity. Our goal is to model resilience as a continuous clinical measure of well-being in PTSD using resting state functional connectivity as well as demographic factors such as the participants' age, and environmental exposure including traumatic events inventory (TEI) score (Sprang, 1997) and the childhood trauma questionnaire (CTQ) total score (Scher et al., 2001). The resilience score of interest is only available for 73 participants, and hence we focus our analysis on this subset.

For our analysis, we used 10 channels for the latent scales in the first stage model that was chosen via cross-validation and yielded desirable results for prediction and uncertainty quantification under the second stage model. We note that the ability of the latent scales to reconstruct the network can be quantified via the area under the ROC curve (AUC) as illustrated in Figure 2.7. In particular, one can first fit the latent scale model to the given network and subsequently reconstruct the edge probabilities under the fitted latent scale model. These edge probabilities can be thresholded under varying cut-offs to inform whether an edge was present or absent under the reconstructed edge set as per the latent scales model. Then, this reconstructed edge set can be compared to the observed network edges and the sensitivity and specificity are computed for a series of thresholds that can then be used for computing the AUC. We note that while an increasing number of channels is expected to lead to greater AUC, it may not necessarily translate to gains in predictive accuracy in the

second stage model, and instead may result in an inflated number of parameters without providing any tangible benefits in terms of prediction and feature selection. This is evident from our GTP analysis, where the predictive accuracy and uncertainty quantification with 10 channels was often superior compared to an alternate analysis with 25 channels in the first stage model (results not presented due to space constraints).

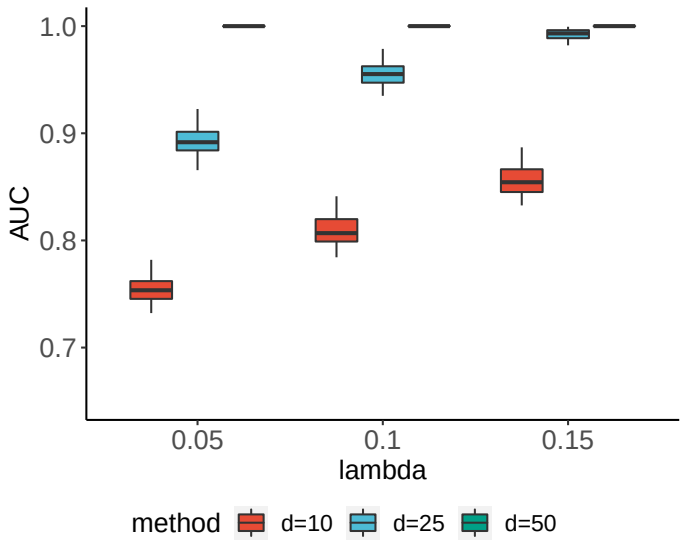

Figure 2.7: Network Recovery Comparisons for the GTP Data under varying number of channels and network densities

Results

The MCMC in stage 2 of the proposed model converged rapidly as evident from Geweke's convergence diagnostic test (Geweke, 1992). Figure 2.8 shows the trace plots for different model parameters. Moreover, Table 2.2 shows the prediction performance of different methods when modeling the resilience score. The results are obtained from 50 random splits into training and testing samples. Both the lsGPR and the sparse lsGPR yield considerable lower predictive MSE compared to all competing methods across varying network densities represented via different λ settings ($\lambda = 0.05, 0.1, 0.15$). Permutation tests show that both the lsGPR and the sparse lsGPR method have significant reduction in MSE at 5% level of significance compared to all competing methods, and across all levels of network sparsity. Moreover, the prediction under the sparse lsGPR is significantly lower compared to the

lsGPR involving all nodes for network densities represented by $\lambda = 0.05, 0.1$, while the predictive MSE is similar for both methods when $\lambda = 0.15$. However, the coverage of the sparse lsGPR is significantly higher compared to the lsGPR method involving all nodes as well as all other competing methods, across all settings. This suggests the superior ability for characterization of predictive uncertainty using node selection under the proposed sparse lsGPR method, which results from superior predictive ability combined with sufficiently wide predictive intervals.

In contrast, approaches that employ the full edge set, including the linear models as well as the edge-GPR method have extremely inferior prediction performance, illustrating the perils of regression using high dimensional edge space that does not respect the inherent dependency structure of the network. Moreover, the edge-GPR method often has poor prediction performance that is sometime subdued even compared to linear models, which highlights the drawbacks of using the full edge set in linear or non-linear models. Moreover, the edge-GPR approach has considerably poor coverage due to tight intervals compared to other non-linear regression approaches. These results suggest a strong justification for non-linear regression modeling with dimension reduction using networks that is able to accommodate unknown interactions with additional exposure variables, and has orders of magnitudes improvements over linear models. Importantly, among all the dimension reduction approaches considered, the advantage of dimension reduction using a latent scale manifold approach coupled with sparse node selection (i.e. the sparse lsGPR method) is most prominent that highlights its' considerable benefits over standard dimension reduction techniques.

The performance of the lsGPR and the sparse lsGPR methods were evaluated under different hyperparameter settings as reported in Table 2.3, which serves as sensitivity analysis of our proposed methods. We consider the following settings: sp-lsGPR1 refers to updating the channels with prior on π at $Beta(1, 10)$ and setting the prior on ψ_1 at inverse Gamma $(0.1, 80)$; sp-lsGPR2 refers to updating the channels with prior on π at $Beta(1, 25)$ and setting the prior on ψ_1 at inverse Gamma $(0.1, 80)$; sp-lsGPR3 refers to fixing the values of the weight matrix to identity while using the prior on ψ_1 as inverse Gamma $(0.1, 10)$;

sp-lsGPR4 refers to updating the channels with prior on π at $Beta(1, 10)$ and specifying the prior on ψ_1 at inverse Gamma $(0.1, 10)$; sp-lsGPR5 refers to updating the channels with prior on π at $Beta(1, 25)$ and specifying the prior on ψ_1 at inverse Gamma $(0.1, 10)$; lsGPR1 refers to updating the channels with prior on π at $Beta(1, 10)$ and setting the prior on ψ_1 at inverse Gamma $(0.1, 80)$ but without node selection; lsGPR2 refers to updating the channels with prior on π at $Beta(1, 25)$ and setting the prior on ψ_1 at inverse Gamma $(0.1, 80)$ and without node selection. This sensitivity analysis reveals that although the out of sample predictive accuracy is less under these hyperparameter settings compared to the performance reported in Table 2.2, the characterization of predictive uncertainty as reflected by the coverage of test samples is significantly higher when using channel selection under different priors on the channel selection probability (see first two columns in Table 2.3). We also discovered that changing the prior on the scale parameter of the Gaussian process from $\psi_1 \sim InverseGamma(0.1, 80)$ as in Table 2.2 to $\psi_1 \sim InverseGamma(0.1, 10)$ in Table 2.3 results in a decrease in predictive accuracy as well as coverage of test samples, due to the tapering of predictive intervals for test samples. These results suggest that overall, the hyperparameter choices made in Table 2.2 work well for the proposed approaches.

Table 2.2: GTP study analysis results for predictive mean squared error (MSE), coverage and (interval) width over 50 random splits. The sp-lsGPR and lsGPR methods here have fixed the weight matrix to identity matrix in first stage and set prior on ψ_1 at inverse Gamma $(0.1, 80)$.

		sp-lsGPR	*lsGPR*	*mfGPR*	*pcaGPR*	*edgeGPR*	*Horseshoe*	*Elastic net*	*Ridge*	*Lasso*
$\lambda = 0.05$	*MSE*	0.885	0.926	0.973	1.090	1.013	1.092	1.052	1.005	1.046
	Coverage	0.986	0.859	0.766	0.774	0.110				
	Width	5.230	4.001	3.872	4.014	0.323				
$\lambda = 0.1$	*MSE*	0.864	0.890	0.912	1.067	1.013	1.099	1.107	1.010	1.081
	Coverage	0.965	0.882	0.822	0.691	0.099				
	Width	5.086	3.964	4.273	3.508	0.299				
$\lambda = 0.15$	*MSE*	0.875	0.873	0.968	1.042	1.014	1.105	1.086	1.024	1.057
	Coverage	0.963	0.859	0.752	0.651	0.098				
	Width	5.053	3.561	3.840	3.284	0.291				

We also examined the node selection results from our sparse lsGPR method based on the posterior inclusion probabilities. Table 2.4 includes the nodes appear in the top ten percent of the ranked posterior inclusion probabilities from analysis of the binary networks with shrinkage parameters λ at 0.05, 0.1 and 0.15. A total of 12 nodes were selected including

Table 2.3: Sensitivity analysis for GTP study analysis results with alternative hyperparameter settings.

		sp-lsGPR1	*sp-lsGPR2*	*sp-lsGPR3*	*sp-lsGPR4*	*sp-lsGPR5*	*lsGPR1*	*lsGPR2*
$\lambda = 0.05$	*MSE*	0.927	0.901	0.933	0.925	0.911	0.963	0.943
	Coverage	0.977	0.975	0.723	0.794	0.799	0.691	0.844
	Width	5.170	5.188	2.267	2.689	2.645	2.916	3.733
$\lambda = 0.1$	*MSE*	0.890	0.924	0.916	0.931	0.937	0.890	0.979
	Coverage	0.990	0.978	0.756	0.811	0.758	0.871	0.745
	Width	5.295	5.175	2.382	2.749	2.546	3.939	3.744
$\lambda = 0.15$	*MSE*	0.911	0.905	0.926	0.943	0.933	1.001	1.008
	Coverage	0.991	0.990	0.739	0.790	0.804	0.775	0.552
	Width	5.228	5.360	2.363	2.686	2.690	3.340	2.264

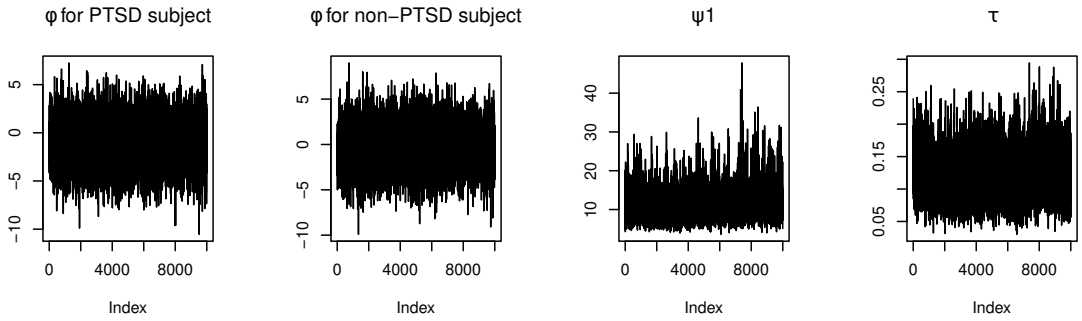

Figure 2.8: Trace plots for gaussian process atom ϕ of one PTSD and one non-PTSD subjects, and hyperparameters ψ_1 and τ.

one from auditory module, three from default mode, one from visual module, one from fronto-parietal module, one from salience, one from subcortical, and four from unknown module. These regions are visually illustrated in the top panel of Figure 2.9, whereas the bottom panel highlights the largest edge differences among these regions between most and least resilient participants. We note that the behavioral patterns listed above have all been shown to be compromised in trauma exposed individuals (Sripada et al., 2013; Falconer et al., 2008) and hence our analysis has a direct relevance in linking the brain network with behavior in trauma exposed individuals. These network differences support our node selection results by highlighting the connectivity differences between individuals with high and low resilience levels. In contrast, the Bayesian horseshoe method has edge-level posterior inclusion probabilities no greater than 0.04 thus we choose not to report its selection results.

Table 2.4: Information on selected nodes in predicting resilience for the GTP study using sp-lsGPR method. (L) and (R) represent left and right cerebrum respectively.

Power atlas index	*Functional module*	*ROI location*	*Average posterior inclusion probability*
64	Auditory	(L) Sub-lobar, insula	0.1139
91	Default mode	(L) Limbic lobe, posterior cingulate	0.1138
102	Default mode	(R) Frontal lobe, superior frontal gyrus	0.1139
137	Default mode	(L) Frontal lobe, inferior frontal gyrus	0.1139
140	Unknown	(R) Occipital lobe, lingual gyrus	0.1136
172	Visual	(L) Occipital lobe, middle occipital gyrus	0.1137
177	Fronto-parietal	(L) Parietal lobe, inferior parietal lobule	0.1137
185	Unknown	(R) Cerebellum posterior lobe, tuber	0.1137
210	Salience	(R) Frontal lobe, inferior frontal gyrus	0.1143
233	Subcortical	(R) Sub-lobar, extra-nuclear	0.1139
243	Unknown	(L) Cerebellum posterior lobe, declive	0.1139
248	Unknown	(L) Limbic lobe, uncus	0.1142

2.5 Conclusion and Future Direction

In this chapter, we make major contributions in the network analysis literature by proposing one of the first flexible non-linear Bayesian non-parametric methods for regression with network-valued covariates under a Gaussian process framework, which also has the added advantage of network node selection via spike and slab priors. The proposed latent scale

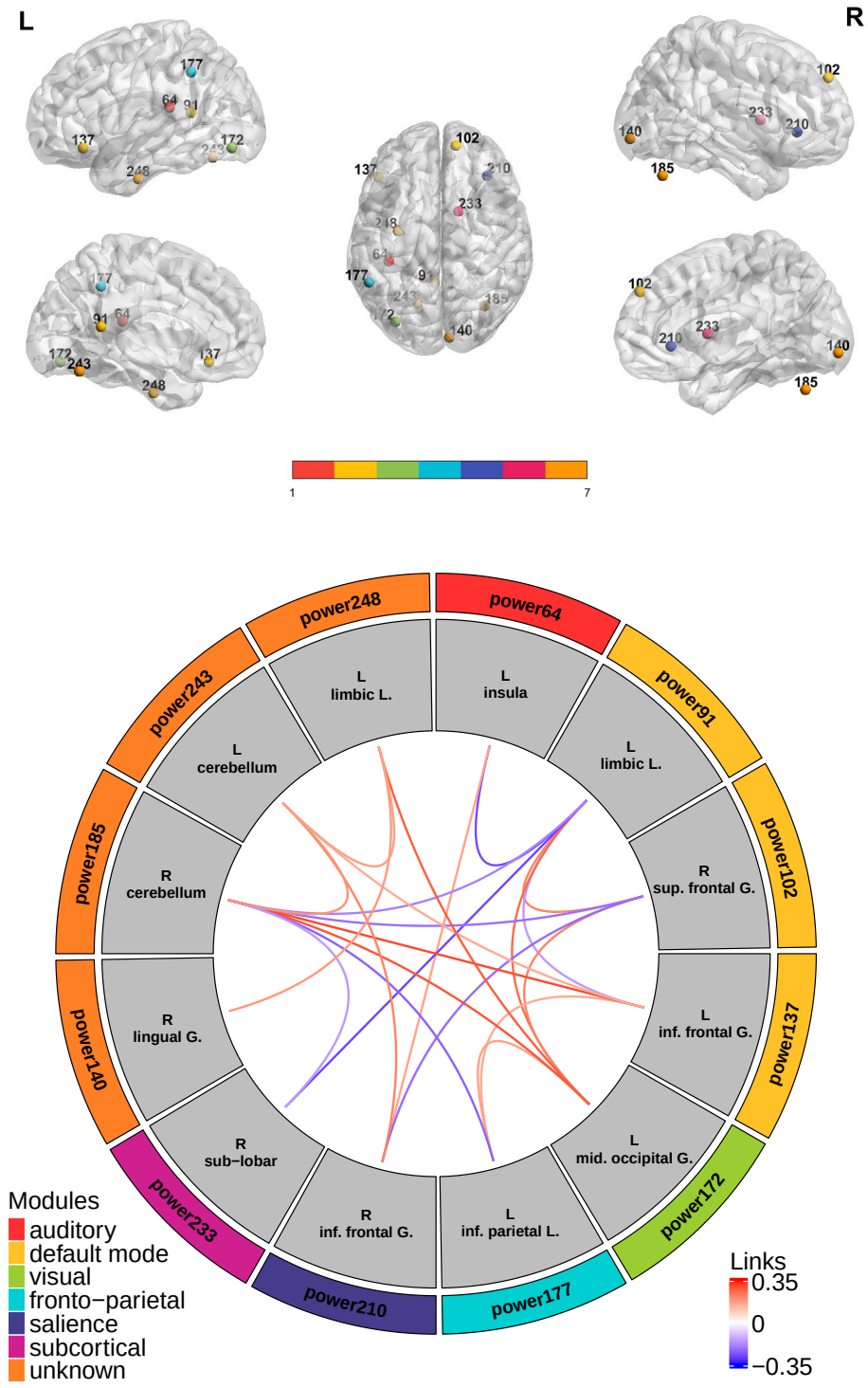

Figure 2.9: Brain maps of selected nodes with respect to resilience using our proposed method (upper) and circular plot for differences in functional connectivity between most and least resilient participants among selected nodes (lower). The connections in the lower panel are red or blue depending on whether the difference in edge strengths between the most and least resilient participants is positive or negative.

Gaussian process regression approach has the advantage of preserving interpretability at the node level that also facilitates node level feature selection, while allowing flexible non-linear relationships between the network and the outcome while accommodating unknown interactions between the network and supplementary covariates that is simply not feasible under existing linear approaches. Hence, the proposed method provides a desirable middle ground between interpretable linear models equipped with feature selection, and highly flexible non-linear models lacking interpretability and feature selection capabilities, in the context of regression with high-dimensional networks. Our extensive numerical studies revealed the significant advantages under the proposed approach over competing linear and non-linear methods. Interestingly, the version of the proposed method incorporating node selection had better predictive performance as well as higher accuracy in characterising predictive uncertainty compared to the version of the proposed method without node selection, which reinforces existing evidence in literature regarding the merits of feature selection and dimension reduction under a Gaussian process framework (Jiang et al., 2007).

Although we adopt a two stage approach for computational ease and scalability to high-dimensional networks, we note that it is possible that the two-step approach could lead to error propagation. However, our primary goal is to regress the clinical outcome on lower dimensional representations of the brain network, and some inadequacies in the estimation of the latent scales are tolerated as long as it does not lead to significant decrease in prediction performance in the second stage. Future research may involve more flexible and scalable approaches that jointly update the mapping to the manifold as well as the regression parameters. Given the scarcity of flexible regression approaches involving high-dimensional covariates in literature, the proposed method is expected to have a significant impact in network analysis literature.

2.6 Appendices

A1. MCMC Algorithm for First Stage Model

The Gibbs Sampler for the first stage model iterates between the following steps:

1. Update the Pólya–Gamma variables $\omega_{i,kl} \sim PG(1, \delta_{i,kl})$ where $PG(\cdot)$ denotes the Pólya–Gamma distribution and $\delta_{i,kl} = a_i + \mathbf{u}_{ik}^T \Lambda_i \mathbf{u}_{il}$ for $1 \leq k < l \leq p$.

2. Update the diagonal elements in the weight matrix

$$\lambda_{ir} \sim Bernoulli(\frac{\pi \mathcal{L}_1}{\pi \mathcal{L}_1 + (1-\pi)\mathcal{L}_0})$$

   where for $r = 2, \cdots, d$,

$$\mathcal{L}_1 = \prod_{k<l,k,l=1}^{p} \frac{1}{2} \exp\left\{(e_{i,kl} - 0.5)\big(a_i + \mathbf{u}_{ik}^T \Lambda_i^{r1} \mathbf{u}_{il}\big) - 0.5\omega_{i,kl}\big(a_i + \mathbf{u}_{ik} T \Lambda_i^{r1} \mathbf{u}_{il}\big)^2\right\}$$

   and $\Lambda_i^{r1} = diag(1, \lambda_{i2}, \cdots, \lambda_{i(r-1)}, 1, \lambda_{i(r+1)}, \cdots, \lambda_{id})$, also

$$\mathcal{L}_0 = \prod_{k<l,k,l=1}^{p} \frac{1}{2} \exp\left\{(e_{i,kl} - 0.5)\big(a_i + \mathbf{u}_{ik}^T \Lambda_i^{r0} \mathbf{u}_{il}\big) - 0.5\omega_{i,kl}\big(a_i + \mathbf{u}_{ik} T \Lambda_i^{r0} \mathbf{u}_{il}\big)^2\right\}$$

   and $\Lambda_i^{r0} = diag(1, \lambda_{i2}, \cdots, \lambda_{i(r-1)}, 0, \lambda_{i(r+1)}, \cdots, \lambda_{id})$.

3. Update the intercept term $a_i \sim N(\mu_i, \sigma_i^2)$ where $\sigma_i^2 = (\sigma_a^{-2} + \sum_{k<l} \omega_{i,kl})^{-1}$ and $\mu_i = \sigma_i^2 \sum_{k<l} (e_{i,kl} - 0.5 - \omega_{i,kl} \mathbf{u}_{ik}^T \Lambda_i \mathbf{u}_{il})$.

4. Update the latent scales $\mathbf{u}_{ik(-1)} \sim N(A_{ik}^{-1} B_{ik}, A_{ik}^{-1})$ where

   $A_{ik} = \sum_{j \neq k} \left[\omega_{i,jk} \Lambda_{i0} \mathbf{u}_{ij(-1)} \mathbf{u}_{ij(-1)}^T \Lambda_{i0} + \sigma_u^{-2} \mathrm{I}_{(d-1)}\right]$ and $B_{ik} = \sum_{j \neq k} \Big[e_{i,jk} - 0.5 - (a_i + b^2)\omega_{i,jk}\Big] \Lambda_{i0} \mathbf{u}_{ij(-1)}$.

5. Update $\pi \sim Beta(a_\pi + \sum_{r=2}^{d} \lambda_{ir}, b_\pi + d - 1 - \sum_{r=2}^{d} \lambda_{ir})$.

Figure 2.10 includes the scatterplots between the prediction and uncertainty quantification results corresponding to the latent scales derived from fitting the EM algorithm and the

MCMC to the first stage model fitting.

A2. Additional Results for Sensitivity Analysis of Channel Number Selection

The following Figure 2.11 includes additional results when the difference between the channel number in the true model and the working model is larger than 1.

A3. Sensitivity Analysis for Hyper-parameters

The following Table 2.5 illustrates the performance of the proposed approach under varying choices of hyperparameters (σ_a^2, σ_u^2) in the EM algorithm for fitting the first stage model.

$[\sigma_a^2,\sigma_u^2]$	active=10			active=30			active=50		
	lsGPR PMSE	sp-lsGPR PMSE	sp-lsGPR AUC	lsGPR PMSE	sp-lsGPR PMSE	sp-lsGPR AUC	lsGPR PMSE	sp-lsGPR PMSE	sp-lsGPR AUC
[2, 2]	0.35	0.32	0.92	0.11	0.16	0.79	0.07	0.11	0.76
[0.2, 2]	0.25	0.25	0.90	0.10	0.18	0.80	0.12	0.13	0.74
[0.5, 2]	0.74	0.27	0.91	0.11	0.14	0.82	0.07	0.13	0.77
[1, 2]	0.28	0.25	0.85	0.19	0.22	0.80	0.08	0.13	0.76
[4, 2]	0.42	0.34	0.85	0.12	0.17	0.78	0.09	0.13	0.78
[8, 2]	0.26	0.21	0.87	0.11	0.16	0.81	0.07	0.13	0.79
[20, 2]	0.29	0.22	0.87	0.09	0.13	0.83	0.09	0.12	0.77
[2, 0.2]	0.61	0.66	0.72	0.29	0.37	0.64	0.26	0.29	0.58
[2, 0.5]	0.41	0.39	0.79	0.12	0.16	0.75	0.11	0.17	0.68
[2, 1]	0.26	0.26	0.91	0.10	0.13	0.83	0.15	0.18	0.76
[2, 4]	0.39	0.25	0.86	0.17	0.18	0.79	0.08	0.13	0.76
[2, 8]	0.30	0.32	0.83	0.07	0.16	0.73	0.07	0.14	0.68
[2, 20]	0.35	0.46	0.77	0.14	0.20	0.69	0.13	0.19	0.64

Table 2.5: Sensitivity analysis corresponding to hyperparameters (σ_a^2, σ_u^2) for the First Stage EM Algorithm.

A4. ROC Curves in Supporting the Node Selection Results

We present here in Figure 2.12 ROC curves from simulation scenario 3 under different network sparsity levels and different number of active nodes in generating the response variable.

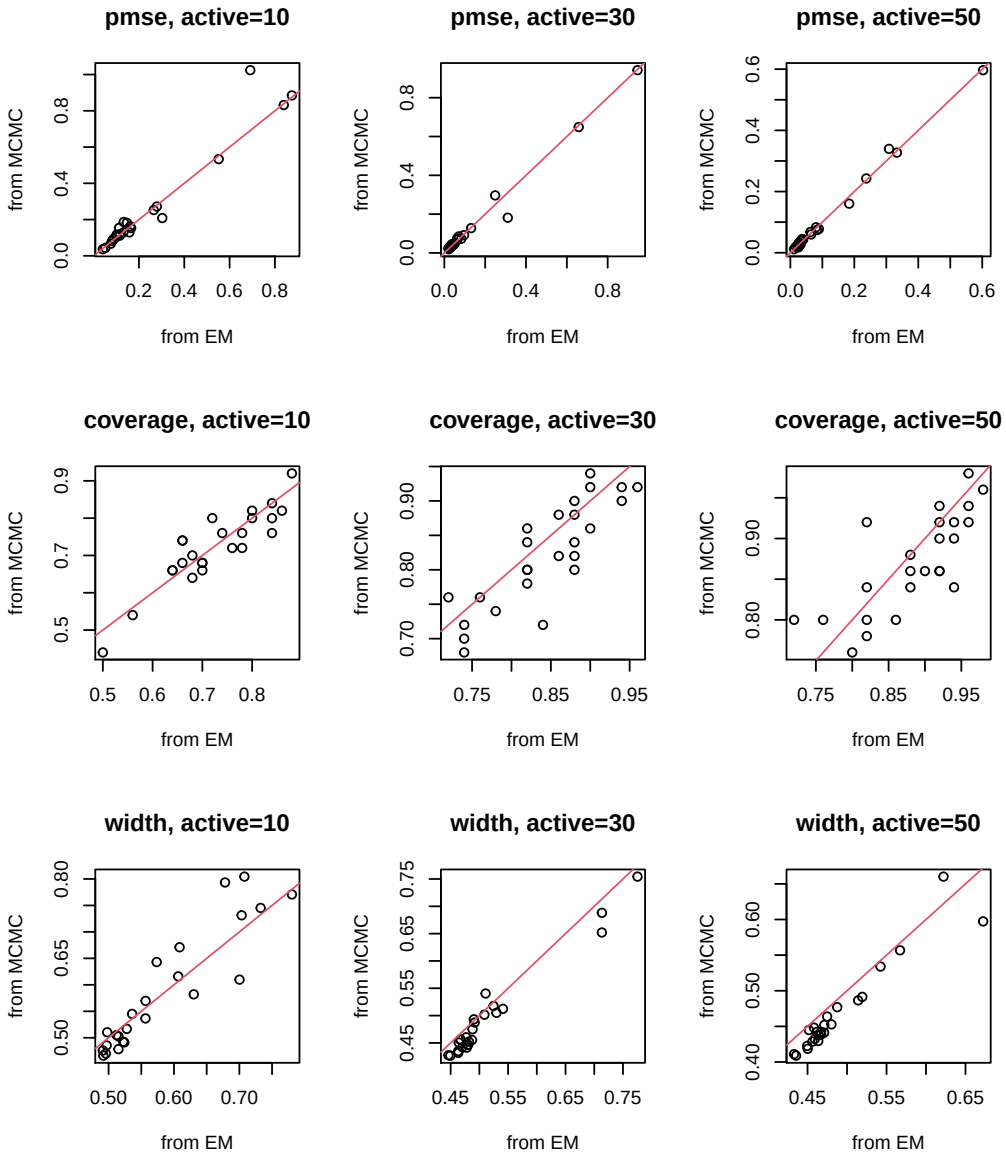

Figure 2.10: Scatterplots comparing the prediction and uncertainty quantification results corresponding to the latent scales derived from fitting the EM algorithm and the MCMC to the first stage model fitting.

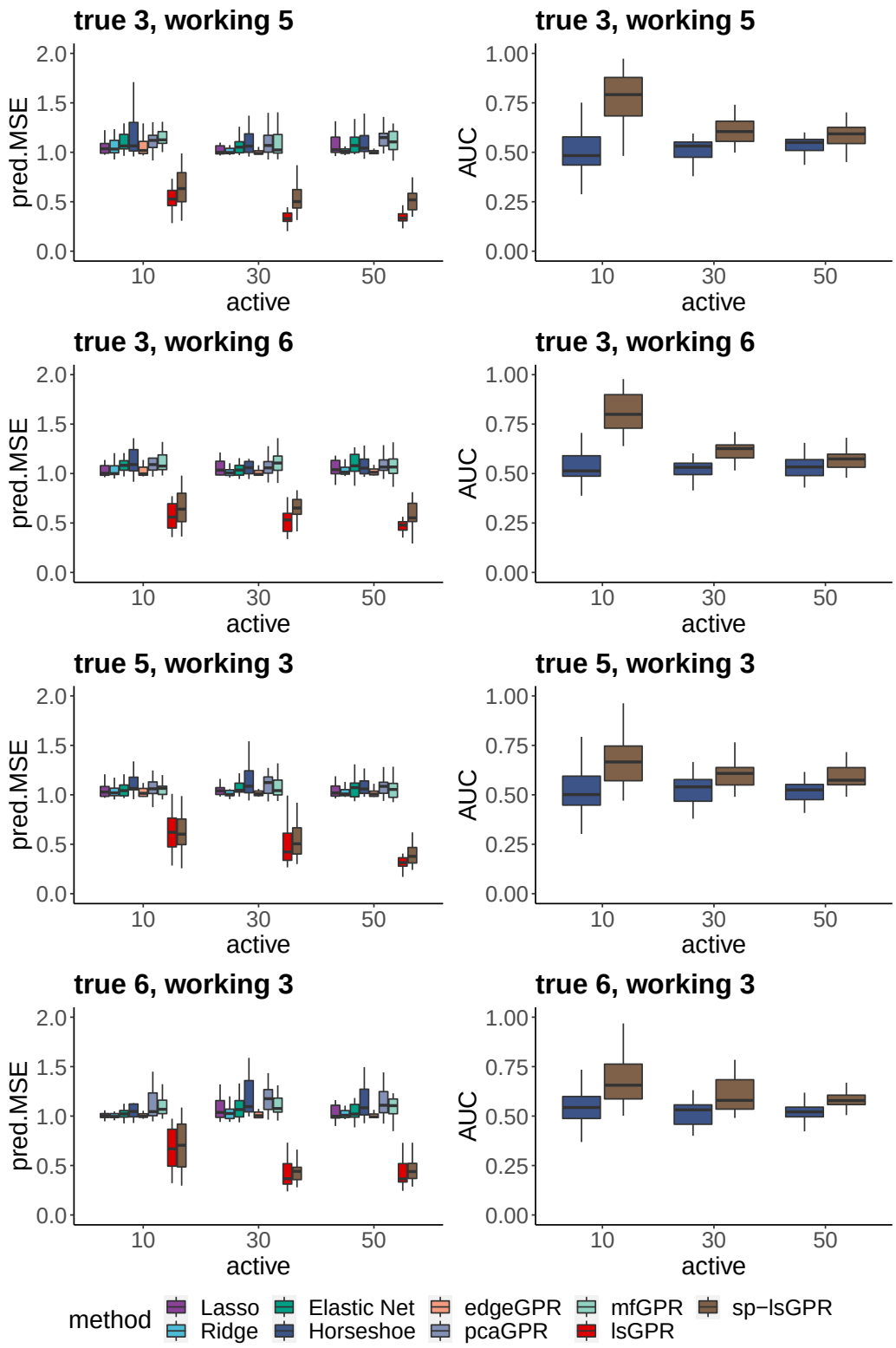

Figure 2.11: Additional Results for Channel Number Sensitivity Tests

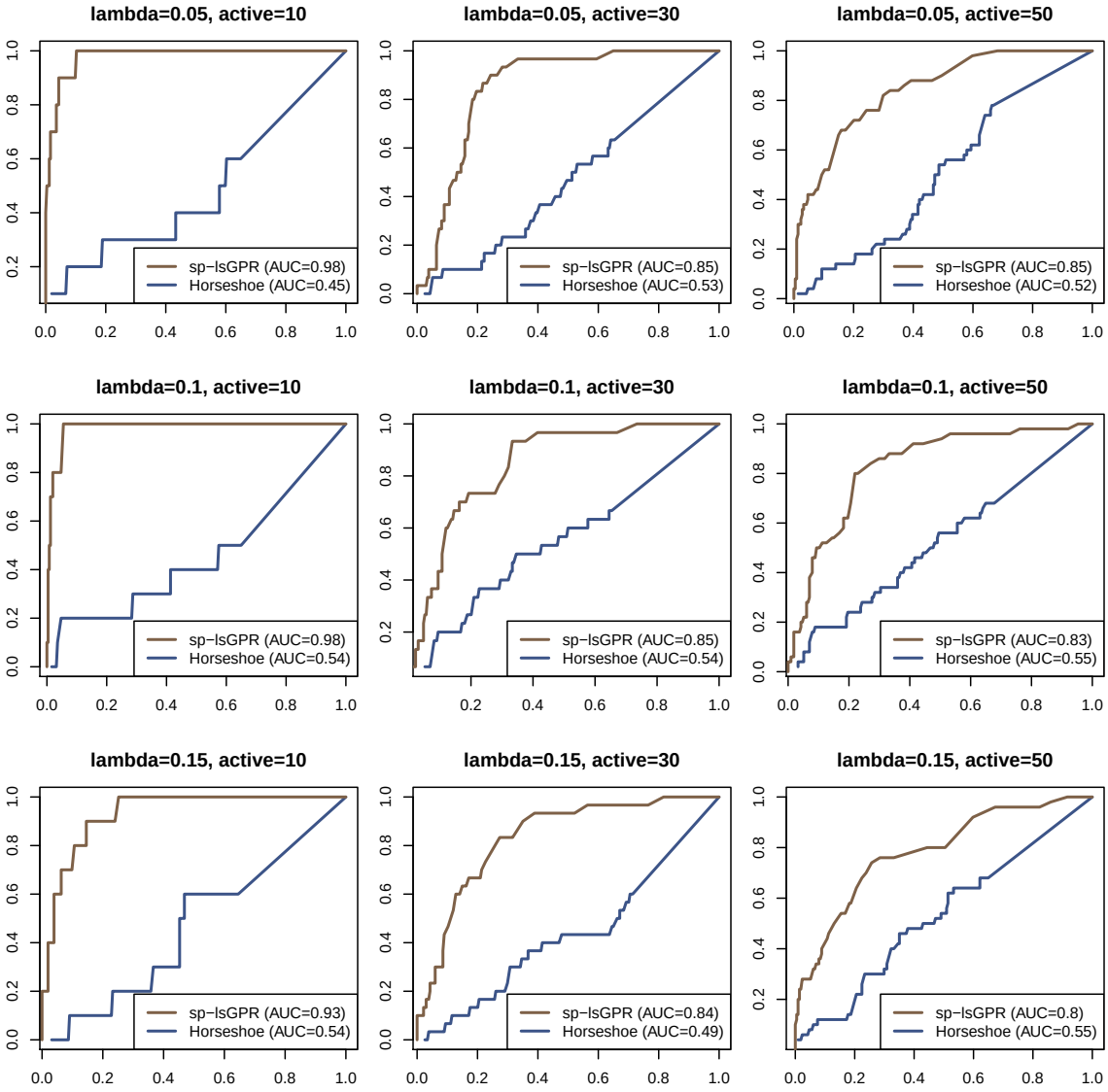

Figure 2.12: Selected ROC Curves from Simulation Scenario 3. The Y-axis represents sensitivity and the X-axis represents 1-specificity.

Chapter 3

Multi-task Learning with High-Dimensional Noisy Images

3.1 Introduction

Methods for functional data analysis (Ramsay and Dalzell, 1991) have become ubiquitous with the growth of recent technologies that are able to generate high-dimensional functional data. Although, the vast majority of literature has focused on one-dimensional functional curves (Morris, 2015), recent literature has started investigating models involving more complex types of functional data such as images. For example, neuroimaging analysis using entire brain images as covariates (scalar-on-image regression) for prediction of a continuous outcome have gained in popularity (Feng et al., 2021). Such approaches are able to discover significantly activated brain voxels and are clearly more attractive compared to methods that evaluate the association between the outcome and each voxel separately (Lazar, 2008).

Typical scalar-on-image regression approaches need to carefully account for the spatial configuration of hundreds of thousands of voxels, and hence often involve some type of lower dimensional representation for the images such as principal components, wavelet representations, or tensors, along with additional sparsity or shrinkage assumptions designed to tackle the curse of dimensionality. Approaches involving functional principal component analysis

(FPCA) (Zipunnikov et al., 2011; Feng et al., 2019) often assume that the components driving variability in the images are related to the outcome that may not always be practical, and they are computationally burdensome for high-dimensional images. Moreover, they only use a subset of principal components resulting in information loss that is potentially exacerbated in the presence of noise in images. A limited number of alternate methods involving wavelet-based representations have been proposed for scalar-on-image regression (Wang et al., 2014; Reiss et al., 2015) that provides a desirable avenue to preserve the spatial properties of the image when modeling regression coefficients. Tensor-based representations for regression coefficients have also been proposed (Feng et al., 2021), which massively reduce the number of parameters needed to be estimated in the model and are shown to possess desirable asymptotic properties. However, the finite sample properties of the existing tensor-based approaches are not well understood, particularly for high-dimensional applications where the tensor decomposition may not provide an adequate characterization.

Unfortunately in spite of the growing practical interest, there is negligible development of methods for joint learning of multiple scalar-on-image regression models corresponding to inter-related high-dimensional imaging datasets. Our joint learning goals are motivated by an increasing interest in data fusion techniques in medical imaging (Lahat et al., 2015), which may involve data on task and rest experiments, or longitudinal neuroimaging data collected in mental health studies such as the Alzheimer's Disease Neuroimaging Initiative (ADNI) (Weiner and Veitch, 2015), among others. A joint analysis of such inter-related datasets can leverage common threads of information across experiments or visits that is expected to lead to greater predictive accuracy and higher power to detect true signals, and produce reliable estimates that are biologically interpretable (Kundu et al., 2019a). One can potentially leverage multi-task learning methods in machine learning literature (Zhang and Yang, 2018; Tang and Song, 2016; Li et al., 2014; Lounici et al., 2011) and related methods for our purposes. Unfortunately, these existing approaches are not designed to tackle integrative analysis involving high dimensional images with spatially distributed voxels, and their theoretical and numerical properties are have not been investigated in the presence of noisy functional covariates. Our extensive numerical studies and ADNI

analysis reveal that the presence of noise may potentially mask the common patterns across inter-related images, which consequently hinders the ability of existing multi-task learning approaches to learn such patterns and eventually results in poor performance.

The presence of noise in brain images is not surprising, given that measurement errors are likely to arise due to technological limitations, operator performance, equipment, environment, and other factors (Vaishali et al., 2015). Although standard pre-processing steps are applied to neuroimaging data prior to analysis, they are not expected to completely alleviate the noise in these images. Unfortunately, existing neuroimaging studies do not account for noise in pre-processed images, which is consistent with the predominant practices in biomedical studies. Inadequate noise correction can result in estimation bias in the direction of zero that is known as attenuation to the null (Carroll and Stefanski, 1994). This phenomenon is also clearly evident for our ADNI analysis (Section 3.5) where standard approaches without noise-correction discover negligible brain activations. We note that standard denoising steps in scalar-on-function regression approaches (Ramsay and Silverman, 2005) may not be biologically meaningful for our neuroimaging applications that already involve a very specific set of pre-processing steps for the images, and they induce additional computational burden.

Although there is a rich literature on measurement error models with scalar covariates (Carroll and Stefanski, 1994), there is (unfortunately) a limited literature on scalar-on-function regression with noisy functional predictors, which can not be directly adapted to our settings of interest involving multi-task learning with high-dimensional noisy images. Such approaches often rely on corrected least squares (OLS) estimators that account for bias due to the presence of noise (Crambes et al., 2009) that may not be applicable to settings when the model dimensions increase much faster than the sample size without additional regularization, or add an additional level of hierarchy by assigning a probability model on the observed functional covariates (James, 2002; Goldsmith et al., 2011). These limited approaches have not been adapted to applications with high-dimensional noisy images involving unknown error variances, and their finite sample theoretical properties remain unclear in such settings. In addition, it is not evident whether these methods for noisy

functional predictors can be directly applied to multi-task learning problems with an added goal of ensuring model parsimony. Alternative Monte Carlo simulation based approaches such as simulation-extrapolation (SIMEX) (Cook and Stefanski, 1994) that were originally designed for univariate or lower dimensional covariates, also suffer from similar drawbacks.

In this chapter, we develop a fundamentally novel approach for the joint analysis of multiple scalar-on-image regression models with high-dimensional noisy images that uses wavelet expansions and grouped penalties for sparse multi-task learning. In particular, we propose a corrected M-estimation approach that adjusts for the bias arising due to noisy images by projecting the solution onto a space of admissible solutions. The proposed approach uses minimal assumptions that involve sub-Gaussian distributions on the true image and additive noise terms with unstructured covariance structures. In order to tackle the curse of dimensionality arising due to high-dimensional images and to enable multi-task learning, we employ grouped penalties such as the non-convex group bridge (Huang et al., 2009) as well as the convex $L_{1,q}$ penalty, on the functional regression coefficients. The group bridge penalty promotes differential sparsity patterns across datasets, whereas the group lasso penalty encourages more similar sparsity patterns designed for robust learning across datasets. Since a closed form solution under the corrected optimization criteria is challenging, we propose a computationally efficient projected gradient descent algorithm that approximates the optimal solution of the model parameters in the presence of noisy images. The proposed approaches translate to locally sparse brain activations, i.e. functional regression coefficients that are zero or non-zero over spatially contiguous regions, which adhere to the biological reality of locally concentrated brain activations in our motivating neuroimaging applications.

We establish attractive theoretical justifications for the proposed methods in high dimensional applications where the number of voxels (p) increases exponentially with sample size (n) for all the M inter-related imaging datasets. Beginning with the case without measurement error in images, we establish weak oracle properties under the group bridge penalty, and we justify the choice of the $L_{1,q}$ penalty by appealing to the desirable theoretical properties that has already been established in literature in the case of covariates without measurement error (Lounici et al., 2011; Negahban et al., 2012). Moving on to the case with

images having voxel-specific additive errors, we derive finite sample statistical error bounds for the optimal solutions explicitly in terms of (n, p, M), for both non-convex and convex grouped penalty functions, which become vanishingly small with high probability as n grows to infinity. In addition, we derive finite sample optimization error bounds which illustrate that the iterations of the projected gradient descent under the convex grouped penalties converges with high probability to the optimum solution, which ensures the legitimacy of the computed parameter estimates. Extensive numerical studies conclusively illustrate the gains under the proposed methods over competing multi-task learning approaches without noise correction as well as noise corrected scalar-on-image regression without multi-task learning, in terms of recovery of true signals and predictive performance. We apply the proposed methods to analyze the longitudinal ADNI brain MRI images, which illustrate the predictive gains under the proposed approach when modeling cognitive outcomes, and provides clear evidence regarding the ability of the proposed noise corrected multi-task learning method to detect biologically meaningful brain activations. In contrast, other multi-task learning methods without noise correction result in poor prediction, and negligible or absent brain activations that is consistent with the attentuation to the null phenomenon in literature.

This chapter makes several significantly novel contributions. First, to our knowledge, the proposed approach is one of the first methods for integrative analysis of multiple scalar-on-image regressions involving inter-related high-dimensional noisy images that provides significant practical gains over existing methods. Hence this approach expands the literature on scalar-on-image regression models without measurement error to scenarios involving multi-task learning in the presence of noisy images. Second, we derive finite sample error bounds for the model parameters explicitly in terms of (n, p, M), which is one of the first such results under grouped non-convex and convex penalties involving noisy functional covariates. Such results provide non-trivial generalizations of previous results in Loh and Wainwright (2012) who focused on linear regression under L_1 penalties involving noisy scalar covariates without multi-task learning. We note that deriving such error bounds is not straightforward due to the inherent non-convexity in the loss function that results from

the presence of noise (see Section 3.3). Third, we derive optimization error bounds under the computationally efficient projected gradient descent algorithm to approximate the optimal solution under the grouped $L_{1,q}$ penalty, which guarantees that the computed parameter estimates are well-behaved. To our knowledge, this is one of the first such results involving noisy functional predictors and under grouped penalties, and is motivated by developments in Agarwal et al. (2012).

Section 3.2 develops the joint scalar-on-image regression approach and theory corresponding uncorrupted images, while Section 3.3 extends this approach for noisy images. Section 3.4 involves extensive simulation studies, Section 3.5 applies the methods to ADNI data, and Section 3.6 contains further discussions. Section 3.7 contains additional materials.

3.2 Multi-task learning without Measurement Errors

Our goal is to propose an approach for joint learning for multiple scalar-on-surface regressions. Denote the scalar continuous outcome $y \in \Re$ that is regressed on an image X defined over a d-dimensional surface and observed at a discrete set of voxels $\{\mathbf{v}_1, \ldots, \mathbf{v}_p\}$. Here $\mathbf{v}_l \in [0,1]^d$ without loss of generality, where $d = 2$ or 3 in practice corresponding to two- or three-dimensional (2-D or 3-D) images. Moreover, the images for all the subjects and data sources are registered to a common template that is standard in medical imaging applications (Avants et al., 2011). For the purposes of exposition and illustration, we will consider the situation with 2-D images as functional predictors. However, our framework is naturally applicable to 3-D images (see Section 3.7). Further, it is straightforward to include additional datasource-specific scalar covariates in the modeling framework, but they are omitted in the following discussions in order to preserve simplicity of notations. Let y_{mi} and X_{mi} denote the outcome and the imaging predictor for subject i ($i = 1, \cdots, n_m$) from data source m ($m = 1, \cdots, M$), where $x_{mi}(\mathbf{v}_k)$ denotes the observed MRI image at voxel $\mathbf{v}_k$ corresponding to X_{mi}. We assume that $|\int X_{mi}(\mathbf{v})\mathrm{d}\mathbf{v}| < \infty$, which is reasonable when $\mathbf{v} \in [0,1]^d$. The scalar-on-image regression model for the mth data source can be written

as:

$$y_{mi} = \beta_{m0} + \int X_{mi}(\mathbf{v})\beta_m(\mathbf{v})\mathrm{d}\mathbf{v} + \epsilon_{mi},\ \epsilon_{mi} \overset{i.i.d.}{\sim} N(0, \sigma_m^2),\ m = 1, \cdots, M; i = 1, \cdots, n_m, \tag{3.2.1}$$

where ϵ_{mi} denotes the random error term that is assumed to be normally distributed with data source-specific residual variance, and $\beta_m(\cdot)$ is the functional regression coefficient that captures the effects of the functional predictor on the outcome corresponding to the mth data source. These functional regression coefficients are estimated jointly across M datasets under an integrative learning framework, as elaborated in the sequel. We allow the number of subjects to vary across the data sources, which gives us flexibility in handling missing data at follow-up visits that is encountered in our motivating ADNI study.

We propose a wavelet-based decomposition for the 2-D images that provides a multiscale representation to accommodate varying degrees of smoothness (Reiss et al., 2015) as:

$$x_{mi}(\mathbf{v}) = \sum_{k,l=0}^{2^{j_0}-1} c^0_{mi,j_0,\{k,l\}}\phi_{j_0,\{k,l\}}(\mathbf{v}) + \sum_{j=j_0}^{J}\sum_{k,l=0}^{2^j-1}\sum_{q=1}^{3} c^q_{mi,j,\{k,l\}}\psi^q_{j,\{k,l\}}(\mathbf{v}) \tag{3.2.2}$$

where j_0 is the primary level of decomposition that controls the number of basis elements in the multi-scale representation, J denotes the maximum level of decomposition, and $\{\phi_{j_0,\{k,l\}}, k, l = 1, \ldots, 2^{j_0} - 1\}$ and $\{\psi^q_{j,\{k,l\}}, j = j_0, \cdots, J, k, l = 0, \ldots, 2^j - 1, q = 1, \cdots, 3\}$ denote pairwise orthonormal wavelets. The wavelet basis functions in (3.2.2) can also be expressed as $\phi_{j_0,\{k,l\}}(\mathbf{v}) = \phi_{j_0,k}(v_1)\phi_{j_0,l}(v_2)$, $\psi^1_{j,\{k,l\}}(\mathbf{v}) = \psi_{j,k}(v_1)\phi_{j,l}(v_2)$, $\psi^2_{j,\{k,l\}}(\mathbf{v}) = \phi_{j,k}(v_1)\psi_{j,l}(v_2)$, $\psi^3_{j,\{k,l\}}(\mathbf{v}) = \psi_{j,k}(v_1)\psi_{j,l}(v_2)$, where $\mathbf{v} = (v_1, v_2)$ and $\phi_{j,\cdot}(\cdot), \psi_{j,\cdot}(\cdot)$ are the one-dimensional father and mother wavelets of level j. Using orthonormality, the wavelet coefficients in (3.2.2) can be calculated by $c^0_{mi,j_0,\{k,l\}} = \langle \mathbf{x}_{mi}, \phi_{j_0,\{k,l\}} \rangle$ and $c^q_{mi,j,\{k,l\}} = \langle \mathbf{x}_{mi}, \psi^q_{j,\{k,l\}} \rangle$, where we use $\langle f_1, f_2 \rangle = \int f_1(\mathbf{v})f_2(\mathbf{v})d\mathbf{v}$ to denote the inner product and $\mathbf{x}_{mi}$ denotes the vectorized image. Due to discrete wavelet transform, we usually require the observed number of locations along all dimensions to be the same (power of 2). If the original functional data does not fulfill this requirement, we can easily pad zero values around it and increase the dimension to the nearest high power of 2 (Reiss et al., 2015), which we denote as p_0. Then the maximum level $J = \log_2(p_0) - 1$, and $p = p_0^2$ is also the total number of

wavelet coefficients in (3.2.2).

We assume the true functional regression coefficients to have bounded total variation, i.e. $\|\beta_m^0(\mathbf{v})\|_V = \int_0^1 \int_0^1 \left|\nabla\beta_m^0(\mathbf{v})\right|\mathrm{d}\mathbf{v} < \infty$, and bounded amplitude, where ∇ denotes the partial derivative in the general sense of distributions (Ziemer, 2012). This implies that the true regression coefficients lie in the space of functions $\mathcal{T} := \left\{\beta(\mathbf{v}) \in L^2[0,1]^2 : \|\beta(\mathbf{v})\|_V < \infty, \left\|\beta(\mathbf{v})\right\|_\infty < \infty\right\}$. We can express $\beta_m^0 \in \mathcal{T}$ as $\beta_m^0(\mathbf{v}) = \sum_{k,l=0}^{2^{j_0}-1} a_{m,j_0,\{k,l\}}^0 \phi_{j_0,\{k,l\}}(\mathbf{v}) + \sum_{j=j_0}^{\infty}\sum_{k,l=0}^{2^j-1}\sum_{q=1}^{3} d_{m,j,\{k,l\}}^{0q}\psi_{j,\{k,l\}}^q(\mathbf{v}) = B^T(\mathbf{v})\boldsymbol{\eta}_m^0 + e_m^0$ for a separable, compactly supported and orthonormal wavelet basis of $L^2[0,1]^2$, where $\{a_{m,j_0,\{k,l\}}^0\}$ and $\{d_{m,j,\{k,l\}}^{0q}\}$ are the true scaling and dilation coefficients, the primary decomposition level is assumed to be known at j_0, $\boldsymbol{\eta}_m^0$ is corresponds to the first p coefficients of the true wavelet coefficient vector, and $e_m^0(\cdot)$ is the approximation term such that $\|e_m^0(\cdot)\|_\infty = O(2^{-J}) = O(p^{-1/2})$ based on Theorem 9.18 in Mallat (1999). Similar specifications can be found in Wang et al. (2017). Since in practice, the functional data are only observed at discrete locations, and given that $e_m^0(\cdot)$ rapidly decreases to zero for large p, we will focus on recovering the truncated coefficient vector $\boldsymbol{\eta}_m^0$ and we will view it as the true wavelet coefficients for our subsequent discussions.

In the above spirit, we will use a finite basis expansion for model fitting. In particular, we use $\beta_m(\mathbf{v}) = \sum_{k,l=0}^{2^{j_0}-1} a_{m,j_0,\{k,l\}}\phi_{j_0,\{k,l\}}(\mathbf{v}) + \sum_{j=j_0}^{J}\sum_{k,l=0}^{2^j-1}\sum_{q=1}^{3} d_{m,j,\{k,l\}}^q\psi_{j,\{k,l\}}^q(\mathbf{v})$, where the unknown coefficients $a_{m,j_0,\{k,l\}} = \langle\beta_m, \phi_{j_0,\{k,l\}}\rangle$ and $d_{m,j,\{k,l\}}^q = \langle\beta_m, \psi_{j,\{k,l\}}^q\rangle$ can be directly computed from the images for a given choice of basis functions. The wavelet representation is flexible enough to allow the functional regression coefficient to be estimated at different levels of smoothness via different levels of j_0. One can rewrite the model (3.2.1) as:

$$y_{mi} = \beta_{m0} + \boldsymbol{c}_{mi}^T\boldsymbol{\eta}_m + \epsilon_{mi},\ \epsilon_{mi} \overset{i.i.d.}{\sim} N(0,\sigma_m^2),\ i = 1,\ldots,n_m,\ m = 1,\ldots,M, \qquad (3.2.3)$$

where $\boldsymbol{c}_{mi}^T\boldsymbol{\eta}_m = \sum_{k,l=0}^{2^{j_0}-1} c_{mi,j_0,\{k,l\}}^0 a_{m,j_0,\{k,l\}} + \sum_{j=j_0}^{J}\sum_{k,l=0}^{2^j-1}\sum_{q=1}^{3} c_{mi,j,\{k,l\}}^q d_{m,j,\{k,l\}}^q$ is the linear mean term, $\boldsymbol{\eta}_m = (\eta_{m1},\ldots,\eta_{mp})'$ denotes the vector of unknown wavelet coefficients corresponding to $\beta_m(\mathbf{v})$, and $\boldsymbol{c}_{mi}(p\times 1)$ denotes the collection of coefficients corresponding to the decomposition of X_{mi} in (3.2.2) that can be computed explicitly. Model (3.2.3) is

now a standard linear regression model with known design matrix $\boldsymbol{C}_m = (\boldsymbol{c}_{m1}, \cdots, \boldsymbol{c}_{mn_m})^T$ and unknown wavelet coefficients $\boldsymbol{\eta}_m, m = 1, \cdots, M$, corresponding to the mth data source, which are jointly estimated across datasets (as elaborated in the sequel) and can be used to reconstruct the functional regression coefficients. We note that by location transformation, we can assume $\beta_{m0} = 0, m = 1, \cdots, M$, without loss of generality, and hence we will ignore the intercept terms in the following discussions. Model (3.2.3) represents a discretized version of the original functional linear model in (3.2.1) that will be used throughout this chapter. We note that in contrast to the working model (3.2.3), the true model is given as $y_{mi} = \int X_{mi}(\mathbf{v})\beta_m^0(\mathbf{v})\mathrm{d}\mathbf{v} + \epsilon_{\mathrm{mi}} = \mathrm{c}_{\mathrm{mi}}^{\mathrm{T}}\boldsymbol{\eta}_{\mathrm{m}}^0 + \int \mathrm{X}_{\mathrm{mi}}(\mathbf{v})\mathrm{e}_{\mathrm{m}}^0(\mathbf{v})\mathrm{d}\mathbf{v} + \epsilon_{\mathrm{mi}}$, using the above discussions.

One can use different types of wavelet bases in (3.2.2)–(3.2.3) - see (Walker, 2008) for more details on different choices of wavelet basis. One possible choice is the Haar wavelets (Wang et al., 2014, 2017) that results in piecewise constant approximations of the signal due to one vanishing moment. The Haar wavelets can be generalized to accommodate higher number of vanishing moments via the Daubechies wavelets (Reiss et al., 2015), which is able to capture diverse types of signals while preserving model parsimony. The choice of the wavelet bases can be tuned to the particular application, as needed.

In order to ensure sparsity in the estimated coefficients that reflects the biological reality of a small subset of activated brain locations driving the outcome, suitable grouped penalty functions $\rho(\cdot)$ are imposed that facilitate joint learning. In particular, we propose to solve the optimization problem: $\max_{\boldsymbol{\eta}} \big\{ -\frac{1}{2}\sum_{m=1}^{M}\sum_{i=1}^{n_m}\Big(y_{mi} - \langle \boldsymbol{c}_{mi}, \boldsymbol{\eta}_m\rangle\Big)^2 - \lambda\rho(\boldsymbol{\eta})\big\}$, where $\rho(\boldsymbol{\eta})$ may correspond to convex penalty functions such as the $L_{1,q}$ penalty $\rho(\boldsymbol{\eta}) = \sum_{j=1}^{p}\Big(\sum_{m=1}^{M}|\eta_{mj}|^q\Big)^{1/q} (q > 1)$, that includes the group lasso when $q = 2$, as well as non-convex penalties such as the group bridge, i.e. $\rho(\boldsymbol{\eta}) = \sum_{j=1}^{p}\Big(\sum_{m=1}^{M}|\eta_{mj}|\Big)^{1/2}$. The convex and non-convex penalties lead to different modes of joint learning by promoting varying sparsity patterns. The group lasso penalty is expected to work better in cases with greater homogeneity across datasets, while the group bridge penalty is recommended for scenarios with more heterogeneous data sources. The choice of these penalties is motivated by our primary goal of data fusion and associated theoretical properties that are already established

in literature for the case without measurement error. For example, consistency properties under the group lasso penalty (Nardi et al., 2008; Lounici et al., 2011) are well-known, while the asymptotic properties (Huang et al., 2009) and weak oracle properties for binary outcomes (Li et al., 2014) of the group bridge penalty have been established in literature.

3.2.1 Weak Oracle Properties under Group Bridge with Uncorrupted Images

We will establish weak oracle properties (Lv et al., 2009) under group bridge that will extend the results in Li et al. (2014) corresponding to binary outcomes involving scalar covariates to the case of scalar-on-image regression with continuous outcomes. For the ease of notation and without loss of generality, we assume that all M data sources have n samples in the following discussion. However, the proposed methodology and theoretical developments are equally applicable for unequal sample sizes across data sources. One can rewrite the optimization problem as

$$\max_{\boldsymbol{\eta}} \left\{ \boldsymbol{y}^T \boldsymbol{C}\boldsymbol{\eta} - \frac{1}{2}\boldsymbol{\eta}^T \boldsymbol{C}^T \boldsymbol{C}\boldsymbol{\eta} - n\lambda_n \rho(\boldsymbol{\eta}) \right\} \tag{3.2.4}$$

where $\rho(\boldsymbol{\eta}) = \sum_{j=1}^{p} \left(\sum_{m=1}^{M} |\eta_{mj}| \right)^{1/2}$, and $\boldsymbol{C}_{(Mn \times Mp)}$ is a block-diagonal design matrix whose mth block $\boldsymbol{C}_m$ corresponds to known wavelet coefficients from the mth data source. Consider the following partition of the index set $\{1, \ldots, p\}$: $I = \{(m,j) | \eta^0_{mj} \neq 0, \boldsymbol{\eta}^0_{(j)} \neq \mathbf{0}\}$, $II = \{(m,j) | \eta^0_{mj} = 0, \boldsymbol{\eta}^0_{(j)} \neq \mathbf{0}\}$ and $III = \{(m,j) | \boldsymbol{\eta}^0_{(j)} = \mathbf{0}\}$ where $\boldsymbol{\eta}^0_{(j)} = (\eta^0_{1j}, \cdots, \eta^0_{Mj})^T$ denotes the true wavelet coefficients corresponding to the j-th wavelet basis function. Set I denotes the indices for the true nonzero coefficients across all data sources, set II denotes the indices for those true wavelet coefficients that are zero for some data sources but not others, while set III denotes the indices for those wavelet coefficients that are zero across all data sources. It is clear that the three sets are mutually exclusive. Further, let $s = |I|$ be the true sparsity level, let $d = 0.5 \min\{|\eta^0_{mj}| : \eta^0_{mj} \in I\}$ with $d/\log n \asymp n^{-\alpha_d}$ where $\alpha_d \leq \gamma, \gamma \in (0, 1/2]$, and denote $l = \min_{\{j:\boldsymbol{\eta}^0_{(j)} \neq \mathbf{0}\}} \|\boldsymbol{\eta}^0_{(j)}\|_1^{1/2}$, $L = \max_{\{j:\boldsymbol{\eta}^0_{(j)} \neq \mathbf{0}\}} \|\boldsymbol{\eta}^0_{(j)}\|_1^{1/2}$. Define the neighborhood $\mathcal{N}_0 = \{\boldsymbol{\delta} \in \mathbb{R}^s : \|\boldsymbol{\delta} - \boldsymbol{\eta}^0_I\|_\infty \leq d\}$, and

the constant $\kappa_0 = \max_{\boldsymbol{\delta} \in \mathcal{N}_0} \max_{\{j|\boldsymbol{\delta}_{(j)} \neq \mathbf{0}\}} 4^{-1} \|\boldsymbol{\delta}_{(j)}\|_1^{-3/2}$, where $\boldsymbol{\delta}_{(j)} = (\delta_{1j}, \cdots, \delta_{Mj})^T$ with $\delta_{mj} = 0$ for $(m, j) \notin I$.

Consider the standardized design matrix $\boldsymbol{C}$ denoted as $\tilde{\boldsymbol{C}}$, such that $\|\tilde{\boldsymbol{c}}_{mj}\|_2 = \sqrt{n}$. We denote $\|\cdot\|_\infty$ as the supremum norm, and denote $\mathcal{O}$ and o as the big-O and little-o notations. Also $a \asymp b$ implies a, b, are on the same order. We will assume the following conditions:

(C1) $\alpha_p = \min(1/2, 2\gamma - \alpha_s)$, where $s \asymp n^{\alpha_s}, \alpha_s < 1, \log p \asymp n^{1-2\alpha_p}$ and $\gamma \in (0, 1/2]$;
(C2) $\|(\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I)^{-1}\|_\infty = \mathcal{O}(b_s n^{-1})$, $b_s = o(n^{1/2-\gamma}\sqrt{\log n})$;
(C3) $\|\tilde{\boldsymbol{C}}_{II}^T \tilde{\boldsymbol{C}}_I (\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I)^{-1}\|_\infty \leq l/(2L)$.

Assumption *(C1)* places conditions on the rate of growth for the true sparsity level (s) and allows p to grow much faster than n. *(C2)* essentially requires $\frac{1}{n}\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I$ to be non-singular and that the supremum norm of $(\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I)^{-1}$ has a lower bound as in equation (15) in Fan and Lv (2011), while *(C3)* is similar to the irrepresentability condition in literature (Zhao and Yu, 2006). Given *(C1)-(C3)*, Theorem 3.2.1 formalizes the weak oracle property.

Theorem 3.2.1. *Suppose the conditions (C1)-(C3) hold. For λ_n satisfying $\lambda_n \asymp n^{-\alpha_\lambda}$ with $\alpha_\lambda < \alpha_p$, $\lambda_n b_s = o(n^{-\alpha_d/2-\gamma} \log n)$ and $\lambda_n \kappa_0 = o(\tau_0)$, where $\tau_0 = \lambda_{\min}(n^{-1}\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I)$, there exists a local maximizer $\hat{\boldsymbol{\eta}}$ of (3.2.4), such that: (a) $\hat{\boldsymbol{\eta}}_{II \cup III} = \mathbf{0}$; and (b) $\|\hat{\boldsymbol{\eta}}_I - \boldsymbol{\eta}_I^0\|_\infty \leq n^{-\gamma} \log n$, with probability greater than $1 - 2\{sn^{-1} + (Mp - s)e^{-n^{1-2\alpha_p} \log n}\}$ for sufficiently large n.*

Property (a) indicates that the oracle estimator for the truly zero wavelet coefficients are estimated correctly with high probability tending to one as $n \to \infty$. Property (b) indicates that with high probability tending to one as n increases, the error under the oracle estimator (in terms of the supremum norm) corresponding to the truly nonzero wavelet coefficients is bounded by a term that goes to zero. Together properties (a) and (b) in Theorem 3.2.1 imply the weak oracle property, which hold for certain strict local maximizers that satisfies the KKT conditions as detailed in Lemma 1 in the Supplementary Materials. The results for the wavelet coefficients in Theorem 3.2.1 can be used to deduce the non-asymptotic error bounds for the corresponding oracle estimators of the functional regression coefficients $\{\hat{\boldsymbol{\beta}}_1(\cdot), \ldots, \hat{\boldsymbol{\beta}}_M(\cdot)\}$ and the predicted means, as captured via the following result.

Corollary 3.2.1. *If Theorem 3.2.1 holds, then* $\left|\hat{\beta}_m(\mathbf{v}) - \beta_m^0(\mathbf{v})\right| \leq \tau_m(\mathbf{v}) s n^{-\gamma} \log n + O(p^{-1/2})$ *and* $\left| \int \boldsymbol{X}_{mi}(\mathbf{v})\hat{\beta}_m(\mathbf{v})\mathrm{d}\mathbf{v} - \int \boldsymbol{X}_{mi}(\mathbf{v})\beta_m^0(\mathbf{v})\mathrm{d}\mathbf{v} \right| \leq \iota_m s n^{-\gamma} \log n + O(p^{-1/2})$*, for all* $m \in \{1, \cdots, M\}$*, where* $\tau_m(\mathbf{v})$ *and* ι_m *can be calculated from the images.*

3.2.2 Computation under Group Bridge with Uncorrupted Images

We can give the wavelet coefficients $\boldsymbol{\eta}$ a hierarchical representation as $\eta_{mj} = g_j \zeta_{mj}$, which involves a differential wavelet coefficient (ζ_{mj}) that is data source-specific, and a shared wavelet effect (g_j) that is common across all the data sources. The hierarchical specification enables us to borrow information across data sources to learn the shared wavelet effects, resulting in joint learning. We note that a zero value for g_j results in a null effect for the j-th wavelet corresponding to all the data sources. Clearly, the differential and shared wavelet effects are not identifiable in the model, but this is tolerated since the goal of the model is to estimate the functional regression coefficient.

Utilizing Lemma 1 and Theorem 1 in Zhou and Zhu (2010), the optimization problem in Section 2 under the group bridge penalty has two equivalent expressions in terms of the hierarchical specification as follows:

$$\begin{aligned}
&\max_{\boldsymbol{g},\boldsymbol{\zeta}} \left[-\frac{1}{2} \sum_{m=1}^{M} \sum_{i=1}^{n_m} \Big\{ y_{mi} - \langle \boldsymbol{c}_{mi}, (\boldsymbol{g} \cdot \boldsymbol{\zeta}_m) \rangle \Big\}^2 - \lambda_g \sum_{j=1}^{p} |g_j| - \lambda_\zeta \sum_{j=1}^{p} \sum_{m=1}^{M} |\zeta_{mj}| \right] \\
&= \max_{\boldsymbol{g},\boldsymbol{\zeta}} \left[-\frac{1}{2} \sum_{m=1}^{M} \sum_{i=1}^{n_m} \Big\{ y_{mi} - \langle \boldsymbol{c}_{mi}, (\boldsymbol{g} \cdot \boldsymbol{\zeta}_m) \rangle \Big\}^2 - \sum_{j=1}^{p} |g_j| - \lambda \sum_{j=1}^{p} \sum_{m=1}^{M} |\zeta_{mj}| \right] \quad (3.2.5)
\end{aligned}$$

where $\boldsymbol{g} = (g_1, \cdots, g_p)^T, \boldsymbol{\zeta}_m = (\zeta_{m1}, \cdots, \zeta_{mp})^T, \boldsymbol{\zeta} = (\boldsymbol{\zeta}_1^T, \cdots, \boldsymbol{\zeta}_M^T)^T$, '$\cdot$' denotes the element-wise product, $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ here denotes the inner product between the two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, $\lambda = \lambda_g \lambda_\zeta$. As seen from the second equivalent expression, we only need one tuning parameter to control sparsity (λ). Since a sparse set of wavelet coefficients also results in zero or close to zero estimated values for the functional regression coefficients (Wang et al., 2014), our model is expected to result in biologically interpretable results by ensuring that only a small percentage of spatially contiguous voxels are related to the outcome.

Taking the advantage of the hierarchical specification, an efficient optimization algorithm is

enabled to estimate the wavelet coefficients as detailed in the following Algorithm 1. In our simulations and real data analysis, we set the stopping threshold $\epsilon^* = 10^{-8}$ that works well in practice. We also set a maximum iteration number at 200 so the algorithm would stop when the iteration rounds reach 200 or the stopping threshold has been reached, whichever comes first. The tuning parameter can be selected via five fold cross validation procedure based on the out-of-sample prediction performance. If further the minimum level of wavelet transform and even the type of wavelet basis functions also need to be selected, one can conduct the cross validation procedure with a grid search for the optimal combination of these parameters.

Algorithm 1 Optimization for model without measurement errors under group bridge penalty

1. Initialize $\hat{\zeta}_{mj}^{(0)} = 1$ $(m = 1, \cdots, M; j = 1, \cdots, p)$.

2. For the kth iteration, let $\tilde{c}_{mi,j} = c_{mi,j}\hat{\zeta}_{mj}^{(k-1)}$ and estimate g_j by

$$\hat{g}_j^{(k)} = \text{argmax}_{g_j}\left[l(\boldsymbol{g}) - \sum_{j=1}^{p}|g_j|\right], j = 1, \cdots, p,$$

   using the lasso algorithm with penalty parameter set to one, where

$$l(\boldsymbol{g}) = -\frac{1}{2}\sum_{m=1}^{M}\sum_{i=1}^{n_m}\left\{y_{mi} - \sum_{j=1}^{p}\tilde{c}_{mi,j}g_j\right\}^2$$

3. Let $\breve{c}_{mi,j} = c_{mi,j}\hat{g}_j^{(k)}$ and estimate ζ_{mj} by

$$\hat{\zeta}_{mj}^{(k)} = \text{argmax}_{\zeta_{mj}}\left[l(\boldsymbol{\zeta}) - \lambda\sum_{j=1}^{p}\sum_{m=1}^{M}|\zeta_{mj}|\right], m = 1, \cdots, M; j = 1, \cdots, p,$$

   using the lasso algorithm with penalty parameter λ where

$$l(\boldsymbol{\zeta}) = -\frac{1}{2}\sum_{m=1}^{M}\sum_{i=1}^{n_m}\left\{y_{mi} - \sum_{j=1}^{p}\breve{c}_{mi,j}\zeta_{mj}\right\}^2$$

4. Compute $\hat{\eta}_{mj}^{(k)} = \hat{g}_j^{(k)}\hat{\zeta}_{mj}^{(k)}, m = 1, \cdots, M; j = 1, \cdots, p.$

5. Repeat step 2 through step 4 until the following convergence criteria is met: $\max_{1\le m\le M, 0\le j\le p}\left|\hat{\eta}_{mj}^{(k)} - \hat{\eta}_{mj}^{(k-1)}\right| < \epsilon^*$ where ϵ^* is a pre-specified threshold.

3.3 Multi-task learning with Measurement Errors

We now generalize our multi-task learning scalar-on-image regression approach to the case of images with measurement errors that is the main focus of this chapter. We assume an additive measurement error model, i.e. $z_{mi}(\mathbf{v}) = x_{mi}(\mathbf{v}) + u_{mi}(\mathbf{v})$, $i = 1, \cdots, m_n$, $m = 1, \cdots, M$, where $x_{mi}(\mathbf{v})$ denotes the true unobserved image at voxel $\mathbf{v}$, while $z_{mi}(\mathbf{v})$ denotes the observed noisy image with the measurement error $u_{mi}(\mathbf{v})$. While we still assume the true model to be $y_{mi} = \beta_{m0} + \int X_{mi}(\mathbf{v})\beta_m(\mathbf{v})\mathrm{d}\mathbf{v} + \epsilon_{mi}$, we relax the distribution of the random error ϵ_{mi} to be sub-Gaussian with parameter σ_m^2. Moreover, the working model used for fitting the data would replace the true image $x_{mi}(\mathbf{v})$ (that is unobserved) by its noisy counterpart $z_{mi}(\mathbf{v})$ in (3.2.2), which results in a different M-estimation criteria compared to (3.2.4) (see equation (3.3.1)). We capture the randomness of the noisy images by assuming that the vectorized true image $\boldsymbol{x}_{mi} = \big(x_{mi}(\mathbf{v}_1), \ldots, x_{mi}(\mathbf{v}_p)\big)$ and the measurement errors $\boldsymbol{u}_{mi} = \big(u_{mi}(\mathbf{v}_1), \ldots, u_{mi}(\mathbf{v}_p)\big)$ are independently distributed as sub-Gaussian random variables (see definition below) with parameters (Σ_m^x, σ_x^2) and (Σ_u, σ_u^2) respectively, which ensures bounded tails.

Definition 3.3.1. *A random vector $\boldsymbol{s} \in \mathbb{R}^p$ is sub-Gaussian with parameters (Σ, σ^2) if: (a) $\boldsymbol{s}$ is distributed with zero mean and covariance Σ; and (b) for any unit vector $\boldsymbol{e} \in \mathbb{R}^p$, the random variable $\boldsymbol{e}^T\boldsymbol{s}$ is sub-Gaussian with parameter at most σ^2, i.e. $\mathsf{P}(|\boldsymbol{e}^T\boldsymbol{s}| > t) \leq 2e^{-t^2/(2\sigma^2)}$.*

Let us denote the matrix of wavelet basis corresponding to the p discrete locations as $B = (\boldsymbol{b}_1, \cdots, \boldsymbol{b}_p)$ where $\boldsymbol{b}_j = \big(b_j(\mathbf{v}_1), \cdots, b_j(\mathbf{v}_p)\big)^T$ denotes the realizations of the j-th basis function at all p discrete locations ($j = 1, \cdots, p$), so that $B^TB = I$. We define $\boldsymbol{w}_{mi} = B^T\boldsymbol{z}_{mi} = B^T\boldsymbol{x}_{mi} + B^T\boldsymbol{u}_{mi} = \boldsymbol{c}_{mi} + B^T\boldsymbol{u}_{mi}$ as the inner product between the noisy observed image and the wavelet basis functions over all voxels. Hence, $\boldsymbol{w}_{mi}$ can be considered as an adaptation of the image wavelet coefficients $\boldsymbol{c}_{mi}$ in (3.2.2) to the case of noisy images. A naive solution that ignores the noise in the images would be to replace $\boldsymbol{c}_{mi}$ with $\boldsymbol{w}_{mi}$ in the scalar-on-image regression model (3.2.3), but this strategy would result in inconsistent estimation under the criteria (3.2.4) (Sørensen et al., 2015). A potential

solution to remedy the problem that is motivated by Loh and Wainwright (2012), is to use a corrected M-estimator that resembles (3.2.4) but adjusts for the additive noise. In this context, note that the matrix $\boldsymbol{C}_m^T\boldsymbol{C}_m$ in (3.2.4) corresponding to the unobserved true images can be approximated by $\hat{\boldsymbol{\Gamma}}_m = \frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{W}_m - B^T\boldsymbol{\Sigma}_u B$, via the relationship $\boldsymbol{w}_{mi} = B^T\boldsymbol{z}_{mi} = \boldsymbol{c}_{mi} + B^T\boldsymbol{u}_{mi}$, where $\boldsymbol{W}_m = (\boldsymbol{w}_{m1}, \cdots, \boldsymbol{w}_{mn})^T$. We propose the following noise corrected version of (3.2.4) as

$$\min_{\rho(\boldsymbol{\eta})\leq R}\Big[\sum_{m=1}^{M}\Big\{\frac{1}{2}\boldsymbol{\eta}_m^T\hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m - \langle\hat{\boldsymbol{\gamma}}_m, \boldsymbol{\eta}_m\rangle\Big\} + \lambda_n\rho(\boldsymbol{\eta})\Big], \tag{3.3.1}$$

where $\hat{\boldsymbol{\gamma}}_m = \frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{y}_m$, $\boldsymbol{y}_m = (y_{m1}, \cdots, y_{mn})^T$, $\rho(\boldsymbol{\eta})$ denotes grouped penalty functions for multi-task learning, and the remaining terms other than $\rho(\boldsymbol{\eta})$ in (3.3.1) represents the loss function $\mathcal{L}$ that makes use of a corrected variance term $\hat{\boldsymbol{\Gamma}}_m$ and a cross-product $\hat{\boldsymbol{\gamma}}_m$. Lemma 3.3.1 illustrates that $(\hat{\boldsymbol{\Gamma}}_m, \hat{\boldsymbol{\gamma}}_m)$ serve as surrogates for $[Var(\mathbf{c}_{mi})]$ and $[Var(\mathbf{c}_{mi})]\boldsymbol{\eta}_m^0$ respectively, where approximation error is shown to decrease to zero as $n \to \infty$ even when $p >> n$. This property indicates the resemblance between the corrected criteria (3.3.1) and criteria (3.2.4) corresponding to no measurement error, since the terms $C^T\mathbf{y}$ and C^TC in (3.2.4) are also unbiased estimators for $[Var(\mathbf{c}_{mi})]\boldsymbol{\eta}_m^0$ and $Var(\mathbf{c}_{mi})$ respectively, in the absence of noise.

Lemma 3.3.1. *(deviation condition) The surrogates* $(\hat{\boldsymbol{\Gamma}}_m, \hat{\boldsymbol{\gamma}}_m)$ *satisfy*

(i) $\left\|\hat{\boldsymbol{\gamma}}_m - B^T\boldsymbol{\Sigma}_m^x B\boldsymbol{\eta}_m^0\right\|_\infty \leq \phi\sqrt{\frac{\log p}{n}}$;

(ii) $\left\|\left(\hat{\boldsymbol{\Gamma}}_m - B^T\boldsymbol{\Sigma}_m^x B\right)\boldsymbol{\eta}_m^0\right\|_\infty \leq \phi\sqrt{\frac{\log p}{n}}$

with probability at least $1 - c_1\exp\{-c_2\log p\}$, *where* $\sigma^2 = \sigma_x^2 + \sigma_u^2$, $\phi = \max_m\left\{c_0\sigma(\sigma_m + \sigma\|\boldsymbol{\eta}_m^0\|_1)\right\} + c_0'\sigma p^{-1/2}$, *and constants* $c_0, c_0', c_1, c_2 > 0$.

Another unique feature of (3.3.1) is that it requires the solution to be restricted to the ball $\rho(\boldsymbol{\eta}) \leq R$, defined in terms of the penalty $\rho(\cdot)$. This restriction is imposed to ensure the stability of solutions, since $\hat{\boldsymbol{\Gamma}}_m$'s are generally not positive semi-definite in the presence of noise, making the loss $\mathcal{L}$ in (3.3.1) non-convex. The restricted solution enables one to tackle a potentially large number of negative eigen values in $\hat{\boldsymbol{\Gamma}}_m$ due to noisy images,

which could otherwise lead to the objective function $\mathcal{L}$ being unbounded from below in an extreme case. In what follows, we will first use the assumption that the covariance matrix for the measurement error $\mathbf{\Sigma}_u$ is known to develop our model and theoretical properties, and subsequently relax this assumption and generalize the results to unknown noise covariance matrices that are empirically estimated. Lemma 3.3.1 and the above discussions provide an intuition regarding the proposed corrected criteria in (3.3.1), which is motivated by Loh and Wainwright (2012).

3.3.1 Theoretical properties under noisy images

In order to establish theoretical properties corresponding to images with measurement errors under criteria (3.3.1), it is necessary to characterize the behavior of the matrix $\hat{\mathbf{\Gamma}}$ (referring to any of $\hat{\mathbf{\Gamma}}_1, \cdots, \hat{\mathbf{\Gamma}}_M$) via some lower restricted eigen value (lower-RE) conditions that places lower bounds on quadratic terms $\boldsymbol{\eta}^T \hat{\mathbf{\Gamma}} \boldsymbol{\eta}$ in (3.3.1). Such conditions prevent the objective function from being unbounded from below when there are a large number of negative eigen values for $\hat{\mathbf{\Gamma}}$ in the presence of noise. Also, the lower-RE condition ensures that the curvature is not overly flat, since $\hat{\mathbf{\Gamma}}$ represents the curvature of the loss function (equivalent to a Hessian matrix in classical literature). Sufficiently curved loss functions are needed to ensure that optimum solutions are able to converge sufficiently close to the true parameter values, given that a small loss difference $|\mathcal{L}(\hat{\boldsymbol{\eta}}) - \mathcal{L}(\boldsymbol{\eta}^0)|$ will translate to a small error $|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0|$. To this effect, we define the following lower-RE condition that is similar to those that have been used extensively in penalized regression literature (Van De Geer et al., 2009), and can be considered a substitute for global strong convexity that can not be guaranteed when $p >> n$.

Definition 3.3.2. *(Lower-RE condition). The matrix $\hat{\mathbf{\Gamma}}$ satisfies a lower restricted eigen-value condition with curvature $\alpha_1 > 0$ and tolerance $\tau > 0$ if $\boldsymbol{\theta}^T \hat{\mathbf{\Gamma}} \boldsymbol{\theta} \geq \alpha_1 \|\boldsymbol{\theta}\|_2^2 - \tau \|\boldsymbol{\theta}\|_1^2, \forall \boldsymbol{\theta} \in \mathbb{R}^p$.*

It turns out that the lower-RE condition holds with high probability in the presence of noise, given the sub-Gaussian assumptions on the true images and the additive errors. This

is shown by Lemma 3.3.2 below under certain choices for α_1 and τ, which follows from the results in Loh and Wainwright (2012).

We are now in a position to formally establish the finite sample error bounds corresponding to the optimal estimators obtained under (3.3.1). We denote $\boldsymbol{\eta}^0 = (\boldsymbol{\eta}_1^{0T}, \cdots, \boldsymbol{\eta}_M^{0T})^T$ as the true wavelet coefficients concatenated over M data sources. Further denote the index set for true wavelet coefficients that have at least one non-zero signal across data sources as $S = \{j : \|\boldsymbol{\eta}_{(j)}^0\|_1 \neq 0\}$ where the cardinality of S represents the group sparsity of $\boldsymbol{\eta}^0$ (denoted by k). Similarly, denote the index set of unimportant wavelet coefficients as $S^C = \{j : \|\boldsymbol{\eta}_{(j)}^0\|_1 = 0\}$, let $l = \min_{j \in S} \|\boldsymbol{\eta}_{(j)}^0\|_1^{1/2}$, $h_1 = 1 + 3l^{-1}R$ and $h_2 = 1 + 3M^{(q-1)/q}$. The following result establishes the statistical (L_1 and L_2) error bounds corresponding to the group bridge penalty explicitly in terms of n, p, M, R, and other parameters.

Theorem 3.3.1. *(statistical error under group bridge) For any $\boldsymbol{\eta}^0$ with group sparsity at most k, the global optimum $\hat{\boldsymbol{\eta}}$ of the problem (3.3.1) under the group bridge penalty $\rho(\cdot)$ satisfies the following error bounds with probability at least $1 - c_1 \exp\{-c_2 \log p\}$ for constants $c_1, c_2 > 0$, $R \geq \rho(\boldsymbol{\eta}^0)$, $\alpha_1/\tau \geq 2h_1^2 Mk$, and $\lambda_n \geq 2\phi\sqrt{\frac{\log p}{n}} \max\{l, R\}$:*
$\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2 \leq \frac{8h_1\sqrt{Mk}}{\alpha_1} \max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n l^{-1}\right\}$, $\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_1 \leq \frac{8h_1^2 Mk}{\alpha_1} \max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n l^{-1}\right\}$

Remark 3.1: While Theorem 3.3.1 is stated for a global optimum, we note that the result also holds for any local optimum that satisfies the constraint $\mathcal{L}(\hat{\boldsymbol{\eta}}) \leq \mathcal{L}(\boldsymbol{\eta}^0)$ for $\mathcal{L}$ as in (3.3.1).

Corollary 3.3.1. *The error bounds in Theorem 3.3.1 hold when the space of admissible solutions in (3.3.1) is restricted to a L_1 ball $\{\boldsymbol{\eta} : \|\boldsymbol{\eta}\|_1 \leq R^2\}$ in (3.3.1), provided that $\|\boldsymbol{\eta}^0\|_1 \leq R^2$ holds.*

We note that Theorem 3.3.1 guarantees that the bound on the statistical error under a non-convex penalty goes to zero even when $p >> n$ and in the presence of measurement errors. In addition, Remark 3.1 suggests the existence of local optima corresponding to (3.3.1) that come arbitrarily close to the true parameters in terms of bounded statistical errors in Theorem 3.3.1. Moreover, Corollary 3.3.1 illustrates that the results in Theorem 3.3.1 are valid when the space of admissible solutions in (3.3.1) is modified in terms of a

L_1 ball, which provides computational benefits when deriving parameter estimates under a projected gradient descent algorithm. Similar to Theorem 3.3.1, we now establish finite sample error bounds under the convex $L_{1,q}$ penalty, which includes the group lasso ($q = 2$) as a special case. We note that although the $L_{1,q}$ penalty is convex, the issues of non-convexity arising due to noise in the images that are encountered in Theorem 3.3.1 still persist in this scenario as well. We note that Theorems 3.3.1-3.3.2 provide novel finite sample error bounds that go beyond existing results in literature by accommodating non-convexity arising due to high-dimensional noisy images in the setting of multi-task learning involving non-convex and convex grouped penalties.

Theorem 3.3.2. *(statistical error under $L_{1,q}$ penalty) For any $\boldsymbol{\eta}^0$ with group sparsity at most k, the global optimum $\hat{\boldsymbol{\eta}}$ of the problem (3.3.1) under the $L_{1,q}$ penalty $\rho(\cdot)$ satisfies the following error bounds with probability at least $1 - c_1 \exp\{-c_2 \log p\}$ for constants $c_1, c_2 > 0$, $R \geq \rho(\boldsymbol{\eta}^0)$, $\alpha_1/\tau \geq 2h_2^2 Mk$, $\lambda_n \geq 2\phi\sqrt{\frac{\log p}{n}} M^{(q-1)/q}$:*
$\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2 \leq \frac{8h_2\sqrt{Mk}}{\alpha_1} \max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n\right\}$, $\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_1 \leq \frac{8h_2^2 Mk}{\alpha_1} \max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n\right\}$

Remark 3.2: Although Theorems 3.3.1-3.3.2 guarantee accurate recovery of the estimated wavelet coefficients, no such finite sample guarantees are available for prediction error. More discussions can be found in Sørensen et al. (2015). However, the proposed methods have good predictive performance as evident from the extensive numerical studies in the sequel.

Remark 3.3: The L_2 error bounds can be expressed as $\frac{4}{\tau h_1\sqrt{Mk}} \max\{\phi\sqrt{\log(p)/n}, \lambda_n l^{-1}\}$ under Theorem 3.1, and as $\frac{4}{\tau h_2\sqrt{Mk}} \max\{\phi\sqrt{\log(p)/n}, \lambda_n\}$ under Theorem 3.2, using the fact that $h_1^2 Mk/\alpha_1 \leq 1/(2\tau)$ and $h_2^2 Mk/\alpha_1 \leq 1/(2\tau)$ respectively. This implies a tightening of the bounds as M increases and directly highlights the benefits of integrative learning.

Although Theorems 3.3.1-3.3.2 establish statistical error bounds for any global optimum of (3.3.1), it is not immediately clear how to computationally obtain such an optimum. This is partly due to non-convexity in the loss function, which hinders a closed form solution to (3.3.1). Hence one needs to resort to some type of projected gradient descent algorithm in order to approximate the solution, which generates a sequence of iterates via the following

recursions:

$$\boldsymbol{\eta}^{(t+1)} = \underset{\rho(\boldsymbol{\eta}) \leq R}{\operatorname{argmin}} \left\{ \mathcal{L}(\boldsymbol{\eta}^{(t)} + \langle \hat{\boldsymbol{\Gamma}} \boldsymbol{\eta} - \hat{\boldsymbol{\gamma}}, \boldsymbol{\eta} - \boldsymbol{\eta}^{(t)} \rangle) + \frac{\delta^*}{2} \|\boldsymbol{\eta} - \boldsymbol{\eta}^{(t)}\|_2^2 \right\}, \tag{3.3.2}$$

where δ^* denotes the step size. However for non-convex problems, the projected gradient descent may get trapped in local minima. While some local optima may lie close to the global optimum as per Remark 3.1, not all local optima are guaranteed to converge to optimum solutions that satisfy the statistical error bounds in Theorems 3.3.1-3.3.2. Fortunately, it is possible to show that the local optima under the projected gradient descent algorithm in (3.3.2) involving the $L_{1,q}$ penalty converges (after a suitable number of iterations) to a solution that is arbitrarily close to the global optimum in Theorem 3.3.2, which ensures the legitimacy of the approximate solution. We first state an additional upper restricted eigen value (upper-RE) condition below that holds with high probability for our settings (Lemma 3.3.2) and is needed in order to derive such a result, followed by the Theorem statement. We note that it is possible to check whether the upper-RE and lower-RE hold in practice using certain sufficient conditions as described in Section 3.3.3.

Definition 3.3.3. *(Upper-RE condition). The matrix $\hat{\boldsymbol{\Gamma}}$ satisfies an upper restricted eigen-value condition with curvature $\alpha_2 > 0$ and tolerance $\tau > 0$ if $\boldsymbol{\theta}^T \hat{\boldsymbol{\Gamma}} \boldsymbol{\theta} \leq \alpha_2 \|\boldsymbol{\theta}\|_2^2 + \tau \|\boldsymbol{\theta}\|_1^2, \forall \boldsymbol{\theta} \in \mathbb{R}^p$.*

Theorem 3.3.3. *(optimization error) Let $\hat{\boldsymbol{\eta}}$ denote an optimum solution in Theorem 3.3.2 under $L_{1,q}$ penalty. Then, the estimate $\boldsymbol{\eta}^{(t)}$ under the projected gradient descent in (3.3.2) with initial choice $\boldsymbol{\eta}^*$ satisfies the error bound $\|\boldsymbol{\eta}^{(t)} - \hat{\boldsymbol{\eta}}\|_2^2 \leq c_3 \frac{k \log p}{n} \|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2^2$ for all iterates $t \geq 2[\log(1/\kappa)]^{-1} \log \frac{\mathcal{L}(\boldsymbol{\eta}^*) - \mathcal{L}(\hat{\boldsymbol{\eta}})}{\delta^2} + \log_2 \log_2 (\frac{R\lambda_n}{\delta^2})(1 + \frac{\log 2}{\log(1/\kappa)})$ with probability at least $1 - c_1 \exp\{-c_2 \log p\}$, for positive constants $c_1, c_2, c_3 > 0$, $\delta^2 = c_3 \frac{k \log p}{n} \|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2^2$, and $\kappa \in (0, 1)$.*

Theorem 3.3.3 shows that the L_2 optimization error under the projected gradient descent algorithm with $L_{1,q}$ penalty is bounded by the statistical error, which is already shown to be well behaved and bounded in Theorem 3.3.2. This result essentially guarantees that the iterate t under the projected gradient descent that is easily computed in polynomial-time

(especially when $q = 2$ resulting in group lasso penalty), converges to a global optimum for the criteria (3.3.1) that may be difficult to compute and may not have a closed form solution. In other words, Theorem 3.3.3 guarantees that when the gradient descent is run long enough, the iterations under the projected gradient descent will produce an estimate that is essentially as good as any global optimum for (3.3.1) in terms of statistical error. This is indeed a desirable practical feature in our applications of interest. Moreover, the number of iterations needed to be run before the approximate solution starts to converge to the optimal solution will depend on the initial choice of the parameters $\boldsymbol{\eta}^*$, with a good choice resulting in faster convergence. This is clear from the fact that a choice of $\boldsymbol{\eta}^*$ close to the global optimum will result in a small difference in $\mathcal{L}(\boldsymbol{\eta}^*) - \mathcal{L}(\hat{\boldsymbol{\eta}})$, subject to the curvature of the loss function.

3.3.2 Case with unknown noise covariance

In practical scenarios, Σ_u is unknown and needs to be estimated. Fortunately, under certain scenarios involving replicated validation data, it is possible to empirically estimate the noise covariance in a manner that ensures that the theoretical properties are preserved. In particular, if we observe n_0 i.i.d. noise vectors $\mathbf{u}$, or in the case of repeated observations from healthy controls, i.e. $\boldsymbol{z}_{mi} = \boldsymbol{x}_i + \boldsymbol{u}_{mi}$ with $\boldsymbol{u}_{mi} \overset{i.i.d.}{\sim} N(0, \boldsymbol{\Sigma}_u)$, Theorems 3.3.1-3.3.3 will hold under empirical estimates of $\boldsymbol{\Sigma}_u$ provided that the validation data has a reasonably large sample size, as stated below.

Corollary 3.3.2. *Theorems 3.1-3.3 hold if we replace $\boldsymbol{\Sigma}_u$ in (3.3.1) by the estimate $\hat{\boldsymbol{\Sigma}}_u = \frac{1}{n_0} U_0^T U_0$ as well as $\hat{\boldsymbol{\Sigma}}_u = \frac{1}{n^*(M-1)} \sum_{i=1}^{n^*} \sum_{m=1}^{M} (\boldsymbol{z}_{mi} - \bar{\boldsymbol{z}}_{\cdot i})(\boldsymbol{z}_{mi} - \bar{\boldsymbol{z}}_{\cdot i})^T$ where $\bar{\boldsymbol{z}}_{\cdot i} = \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{z}_{mi}$ with $\boldsymbol{z}_{mi} = \boldsymbol{x}_i + \boldsymbol{u}_{mi}$ for healthy controls and $\boldsymbol{u}_{mi} \overset{i.i.d.}{\sim} N(0, \boldsymbol{\Sigma}_u)$ for the second estimator, given that $n_0 > n$ and $n^*(M-1) > n$.*

The proof of the above result (provided in Section 3.7) proceeds by showing that the deviation condition in Lemma 3.3.1 as well as the lower- and upper-RE conditions still hold with high probability under the modified covariance estimator $\hat{\boldsymbol{\Sigma}}_u$ in Corollary 3.3.2. We use this strategy in our analysis of ADNI data that comprises longitudinal visits for healthy

controls in addition to individuals with AD, as detailed in Section 3.5.

3.3.3 Lower- and Upper-RE Conditions

Lemma 3.3.2. *Lower- and upper-RE conditions hold with probability at least* $1 - c_1 \exp\Big(-c_2 n \min\big\{\frac{\delta_{\min}^2}{\sigma^4}, 1\big\}\Big)$*, for parameters* $\delta_{\min} = \min_{m\in\{1,\cdots,M\}} \lambda_{\min}(B^T \mathbf{\Sigma}_m^x B)$*,* $\delta_{\max} = \max_{m\in\{1,\cdots,M\}} \lambda_{\max}(B^T \mathbf{\Sigma}_m^x B)$*,* $\alpha_1 = \frac{1}{2}\delta_{\min}$*,* $\alpha_2 = \frac{3}{2}\delta_{\max}$*,* $\tau = c_0 \max\big\{\frac{\sigma^4}{\delta_{\min}}, \delta_{\min}\big\}\frac{\log p}{n}$ *and universal constants* $c_0, c_1, c_2 > 0$*.*

From a practical perspective, it is possible to check whether the restricted eigenvalue conditions holds in practice for a given dataset. We use $\hat{\mathbf{\Gamma}}$ to denote any of $\hat{\mathbf{\Gamma}}_1, \cdots, \hat{\mathbf{\Gamma}}_M$ in the following discussion. Note that the lower-RE condition on the matrix $\hat{\mathbf{\Gamma}}$ requires that $\boldsymbol{\theta}^T \hat{\mathbf{\Gamma}} \boldsymbol{\theta} \geq \alpha_1 \|\boldsymbol{\theta}\|_2^2 - \tau \|\boldsymbol{\theta}\|_1^2, \forall \boldsymbol{\theta} \in \mathbb{R}^p$, and the upper-RE condition requires that $\boldsymbol{\theta}^T \hat{\mathbf{\Gamma}} \boldsymbol{\theta} \leq \alpha_2 \|\boldsymbol{\theta}\|_2^2 + \tau \|\boldsymbol{\theta}\|_1^2, \forall \boldsymbol{\theta} \in \mathbb{R}^p$, with curvatures $\alpha_1, \alpha_2 > 0$ and tolerance $\tau > 0$. Given that $\boldsymbol{\theta}^T \hat{\mathbf{\Gamma}} \boldsymbol{\theta} \geq \lambda_{\min}(\hat{\mathbf{\Gamma}}) ||\boldsymbol{\theta}||_2^2$, where $\lambda_{\min}(A)$ denotes the minimum eigen value of the matrix A, it is clear that the lower-RE condition would be satisfied if $\lambda_{\min}(\hat{\mathbf{\Gamma}}) ||\boldsymbol{\theta}||_2^2 \geq \alpha_1 \|\boldsymbol{\theta}\|_2^2 - \tau \|\boldsymbol{\theta}\|_1^2$ or equivalently if $\lambda_{\min}(\hat{\mathbf{\Gamma}}) \geq \alpha_1 - \tau \|\boldsymbol{\theta}\|_1^2 / \|\boldsymbol{\theta}\|_2^2 = \alpha_1 - \tau(1 + \theta^*)$, where $\theta^* = \frac{2}{\|\boldsymbol{\theta}\|_2^2} \sum_{k<l} |\theta_k||\theta_l| > 0$. Clearly, this condition is satisfied when $\lambda_{\min}(\hat{\mathbf{\Gamma}}) \geq \alpha_1 - \tau$. However if this condition is not satisfied, then one can not say with certainty whether the lower-RE condition is satisfied or not.

In order to check whether the condition $\lambda_{\min}(\hat{\mathbf{\Gamma}}) \geq \alpha_1 - \tau$ holds, one can substitute the parameters in this inequality with their corresponding empirical estimates from the data. As noted in Lemma 3.3.2, we have that $\alpha_1 = 0.5 \min_{m\in\{1,\cdots,M\}} \lambda_{\min}(B^T \mathbf{\Sigma}_m^x B)$ and $\tau = c_0 \max\big\{\frac{\sigma^4}{2\alpha_1}, 2\alpha_1\big\}\frac{\log p}{n}$ where c_0 needs to be sufficiently small such that $c_0 \leq \frac{n}{2 \log p} \min\{\frac{4\alpha_1^2}{\sigma^4}, 1\}$. It is possible to estimate σ^2 by $(1/nM) \sum_{m=1}^M \|\boldsymbol{W}_m^T \boldsymbol{W}_m\|_{op}$ where $\|\cdot\|_{op}$ denotes the spectral norm of matrix, and to estimate $B^T \mathbf{\Sigma}_m^x B$ by $\hat{\mathbf{\Gamma}}_m = (1/n) \mathbf{W}_m^T \mathbf{W}_m - B^T \hat{\Sigma}_u B$. One can then plug in $\hat{\alpha}_1, \hat{\tau}$, in place of the original parameters to check the condition $\lambda_{\min}(\hat{\mathbf{\Gamma}}) \geq \alpha_1 - \tau$. If this inequality is satisfied, then this would imply that the lower-RE condition holds. A similar practical check can be implemented for the upper-RE condition by following the above steps and noting that $\alpha_2 = 1.5 \max_{m\in\{1,\cdots,M\}} \lambda_{\max}(B^T \mathbf{\Sigma}_m^x B)$.

3.3.4 Computational Algorithms

We implement the projected gradient descent approach under the group bridge and the group lasso penalties, whose convergence can be impacted by the choice of the step size δ^* in (3.3.2). We utilize the non-monotone spectral projected gradient (SPG) method for the computation under the group lasso penalty as in Sra (2011) and Duchi et al. (2008), which adopts a spectral choice of the step size with non-monotone line search technique that is known to speed up the convergence.

We tackle the optimization problem of model with measurement errors under group lasso penalty by utilizing the SPG algorithm with projection onto $L_{1,2}$ ball. Details are listed in the following three algorithms. The first two algorithms (Algorithms 2, 3) provide details on projection onto a L_1 ball and $L_{1,2}$ ball respectively, whereas Algorithm 4 outlines the steps for SPG under the group lasso penalty. In Algorithm 4, the function $g(\cdot)$ denotes the gradient related to the loss function $\mathcal{L}$ as in (3.3.1).

Algorithm 2 Projection onto L_1 ball with radius R and penalty term λ_0

Input: vector $\boldsymbol{v}$ of length p, radius R, penalty term λ_0.

if $\|\boldsymbol{v}\|_1 \leq R$ **then**
  $proj(\boldsymbol{v}) = \max\{|\boldsymbol{v}| - \lambda_0, 0\}$, stop.
else
  set $U = \{1, \cdots, p\}$, $s = 0$, $\rho = 0$.
  while $U \neq \emptyset$ **do**
    randomly sample $k \in U$.
    partition U as $G = \{j \in U \big| |v_j| \geq |v_k|\}$, and $L = \{j \in U \big| |v_j| < |v_k|\}$.
    compute $\Delta\rho = card\{G\}$, $\Delta s = \sum_{j \in G} |v_j|$.
    if $s + \Delta s - (\rho + \Delta\rho)|v_k| < R$ **then**
      $s = s + \Delta s$, $\rho = \rho + \Delta\rho$, $U = L$.
    else
      $U = G \setminus \{k\}$.
    end if
  end while
  set $\theta = (s - R)/\rho$, $\lambda = \max\{\theta, \lambda_0\}$, and $proj(\boldsymbol{v}) = \max\{|\boldsymbol{v}| - \lambda, 0\}$.
end if

Output: projected vector $\prod_1(\boldsymbol{v}; R, \lambda_0) = proj(\boldsymbol{v}) * sgn(\boldsymbol{v})$.

Moreover under the non-convex group bridge penalty, we develop a novel projected gradient descent algorithm which restricts the space of admissible solutions to an L_1 ball of radius R^2 by leveraging Corollary 3.3.1. Using the data augmentation strategy in Huang et al. (2009),

Algorithm 3 Projection onto $L_{1,2}$ ball with radius R and penalty term λ_0

Input:
vector $\boldsymbol{\eta}$ of length M (number of groups) by p (number of elements), radius R, penalty term λ_0.
Compute
vector $\boldsymbol{c} = (\|\boldsymbol{\eta}_{(1)}\|_2, \cdots, \|\boldsymbol{\eta}_{(p)}\|_2)^T$, where $\boldsymbol{\eta}_{(j)} = (\eta_{1j}, \cdots, \eta_{Mj})^T$, $j = 1, \cdots, p$.
Compute
$\boldsymbol{w} = \prod_1(\boldsymbol{c}; R, \lambda_0)$, and $\eta^*_{mj} = \eta_{mj}\frac{w_j}{c_j}$, $m = 1, \cdots, M$, $j = 1, \cdots, p$.
Output:
projected vector $\prod_{1,2}(\boldsymbol{\eta}; M, p, R, \lambda_0) = (\eta^*_{11}, \cdots, \eta^*_{M1}, \cdots, \eta^*_{1p}, \cdots, \eta^*_{Mp})^T$.

Algorithm 4 Spectral Projected Gradient Algorithm for Group Lasso Penalty

Input: starting value $\boldsymbol{\eta}^{(0)}$, $\alpha^{(0)} = 1/\|g_1(\boldsymbol{\eta}^{(0)})\|_\infty$ where

$$g_1(\boldsymbol{\eta}^{(0)}) = \prod_{1,2}(\boldsymbol{\eta}^{(0)} - g(\boldsymbol{\eta}^{(0)}); M, p, R, \lambda_n) - \boldsymbol{\eta}^{(0)},$$

set $\gamma = 10^{-4}$, $\alpha_{\min} = 10^{-30}$, $\alpha_{\max} = 10^{30}$, and $tol = 10^{-5}$.

Iterate between the following steps.

Step 1: check if stationary.
if $\|\prod_{1,2}(\boldsymbol{\eta}^{(k)} - g(\boldsymbol{\eta}^{(k)}); M, p, R, \lambda_n) - \boldsymbol{\eta}^{(k)}\|_\infty \le tol$ **then** Stop, output $\boldsymbol{\eta}^{(k)}$.
end if

Step 2: backtracking.
Step 2.1: compute $d^{(k)} = \prod_{1,2}(\boldsymbol{\eta}^{(k)} - \alpha^{(k)} g(\boldsymbol{\eta}^{(k)}); M, p, R, \lambda_n) - \boldsymbol{\eta}^{(k)}$, set $\lambda = 1$.
Step 2.2: set $\boldsymbol{\eta}^+ = \boldsymbol{\eta}^{(k)} + \lambda d^{(k)}$.
Step 2.3:
if $\mathcal{L}(\boldsymbol{\eta}^+) \le \max_{0\le s\le \min\{k,9\}} \mathcal{L}(\boldsymbol{\eta}^{(k-s)}) + \gamma\lambda\langle d^{(k)}, g(\boldsymbol{\eta}^{(k)})\rangle$ **then**
define $\lambda^{(k)} = \lambda$, $\boldsymbol{\eta}^{(k+1)} = \boldsymbol{\eta}^+$, $s^{(k)} = \boldsymbol{\eta}^{(k+1)} - \boldsymbol{\eta}^{(k)}$, $y^{(k)} = g(\boldsymbol{\eta}^{(k+1)}) - g(\boldsymbol{\eta}^{(k)})$ and go to Step 3.
else
define $\lambda_{new} \in [0.1\lambda, 0.9\lambda]$, set $\lambda = \lambda_{new}$ and go to Step 2.2.
end if

Step 3: compute $b^{(k)} = \langle s^{(k)}, y^{(k)}\rangle$.
if $b^{(k)} \le 0$ **then** set $\alpha^{(k+1)} = \alpha_{\max}$.
else
compute $a^{(k)} = \langle s^{(k)}, s^{(k)}\rangle$, $\alpha^{(k+1)} = \min\{\alpha_{\max}, \max\{\alpha_{\min}, \frac{a^{(k)}}{b^{(k)}}\}\}$.
end if

Output: $\boldsymbol{\eta}^{(k)}$ either when the stationary criterion is satisfied in Step 1, or the maximum iteration number is hit.

criteria (3.3.1) under group bridge can be expressed as the following equivalent problem :

$$\min_{\|\boldsymbol{\eta}\|_1 \le R^2} \Big[\sum_{m=1}^{M} \Big\{ \frac{1}{2} \boldsymbol{\eta}_m^T \hat{\boldsymbol{\Gamma}}_m \boldsymbol{\eta}_m - \langle \hat{\boldsymbol{\gamma}}_m, \boldsymbol{\eta}_m \rangle \Big\} + \sum_{j=1}^{p} \theta_j^{-1} \Big(\sum_{m=1}^{M} |\eta_{mj}| \Big) + \tau_n \sum_{j=1}^{p} \theta_j \Big], \ \tau_n = \lambda_n^2/4, \tag{3.3.3}$$

where $\theta_j \ge 0$ for $j = 1, \cdots, p$. This can be solved via the SPG method which is detailed in Algorithm 5 below. The SPG step in Algorithm 5 is very similar to Algorithm 4, with only minor changes including: (a) the loss function and the corresponding gradient function need to be replaced according to the formula in step 3 of Algorithm 5; and (b) the projection onto $L_{1,2}$ ball in Algorithm 4 needs to be replaced by a projection onto L_1 ball with fixed penalty equal to 1 and a large enough radius, which we set to $R = 10^{30}$ in our implementation.

Algorithm 5 Projected Gradient Descent with Data Augmentation under Group Bridge Penalty

1. Initialize the values of the wavelet coefficients at $\boldsymbol{\eta}^*$.
2. For the k-th iteration, update θ_j for $j = 1, \cdots, p$ as $\theta_j^{(k)} = \tau_n^{-1/2} \Big(\sum_{m=1}^{M} |\eta_{mj}^{(k-1)}| \Big)^{1/2}$.
3. Use SPG method to find the solution for $\boldsymbol{\eta}^\theta$ where $\eta_{mj}^\theta = \theta_j^{-1} \eta_{mj}$ as

$$\hat{\boldsymbol{\eta}}^\theta = \underset{\boldsymbol{\eta}^\theta}{\operatorname{argmin}} \Big[\sum_{m=1}^{M} \Big\{ \frac{1}{2} (\boldsymbol{\eta}_m^\theta)^T \Big(I_\theta^{(k)} \hat{\boldsymbol{\Gamma}}_m I_\theta^{(k)} \Big) \boldsymbol{\eta}_m^\theta - \Big(I_\theta^{(k)} \hat{\boldsymbol{\gamma}}_m \Big)^T \boldsymbol{\eta}_m^\theta \Big\} + \|\boldsymbol{\eta}^\theta\|_1 \Big]$$

   where $I_\theta^{(k)} = diag\{\theta_1^{(k)}, \cdots, \theta_p^{(k)}\}$. Then update $\boldsymbol{\eta}$ with $\eta_{mj}^{(k)} = \theta_j^{(k)} \hat{\eta}_{mj}^\theta$, $m = 1, \cdots, M$, $j = 1, \cdots, p$ and project it onto an L_1 ball with radius R^2.
4. Repeat step 2 through 3 until convergence.

It is evident from the inequality $||\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}_0||_2 \le \frac{4}{\tau h_1 \sqrt{Mk}} \max\{\phi\sqrt{\log p/n}, \lambda_n l^{-1}\}$ in Theorem 3.3.1, that the denominator for the error bound contains $h_1 = 1 + 3l^{-1}R$ that increases linearly with R. Moreover when $R > l$, we have $\lambda_n > 2\phi\sqrt{\frac{\log p}{n}} R > \phi\sqrt{\frac{\log p}{n}}$ for $R > \rho(\boldsymbol{\eta}^0) > 1$, and further $\lambda_n l^{-1} > \phi\sqrt{\frac{\log p}{n}} R l^{-1} > \phi\sqrt{\frac{\log p}{n}}$. Hence when $R > l$, the error bounds reduces to a ratio $4\frac{\lambda_n l^{-1}}{\tau\sqrt{Mk}(1+3l^{-1}R)}$, which decreases with R when λ_n is chosen to be large enough. Hence our theory benefits from a large choice of R and in general, one can not choose R to be extremely small if we would like our non-asymptotic results to hold, since Theorems 3.3.1-3.3.2 require $R > \rho(\boldsymbol{\eta}^0)$. Similarly it is not useful

to choose R as extremely large, since the optimization error bounds in Theorem 3.3.3 are only applicable to all iterates t in the projected gradient descent algorithm that satisfy $t \geq 2[\log(1/\kappa)]^{-1} \log \frac{\mathcal{L}(\boldsymbol{\eta}^*)-\mathcal{L}(\hat{\boldsymbol{\eta}})}{\delta^2} + \log_2 \log_2(\frac{R\lambda_n}{\delta^2})(1 + \frac{\log 2}{\log(1/\kappa)})$, and an extremely large choice of R would imply slower convergence for this projected gradient descent algorithm. In our applications, we choose R to be large enough that works well in diverse practical applications.

In practice, one can get a better sense about the lower bound of the constraints R and R^2 from the estimates of the group lasso and group bridge methods without noise correction in (3.2.4), which can be used to guide the choice of R. Alternatively, the correction proposed in Datta et al. (2017) avoids tuning on R and the projection step, which may be potentially interesting to consider in future work. Moreover, the shrinkage parameter λ_n in the penalty term can be selected via five fold cross validation. In our implementations, we fix the primary level of wavelet transform (j_0) informed by extensive empirical studies. However, in the situation when one is uncertain about the choice of j_0, a cross validation can be conducted based on cross-validation or goodness of fit scores.

3.4 Simulations

In this section we conduct extensive simulations involving three data sources ($M = 3$) with 2-D images of size 64×64 that mimics the 2-D brain slices and evaluate the performance of the proposed approach with respect to competing methods. The functional image predictors are generated by first generating wavelet coefficients independently from normal distribution with mean 0 and variance 1, and followed by an inverse wavelet transform implemented via the `R` package `wavethresh` (Nason, 2008). We choose the Daubechies Least Asymmetric wavelet with 4 vanishing moments and $j_0 = 3$ as the wavelet basis function in both data generation and model fitting under all wavelet-based approaches. We generated three types of true 2-D functional regression coefficients with different shapes including round, square and triangle, and varying degrees of overlap in the regions with non-zero signals between the three images across the data sources as shown in Figure 3.1. We also considered two extreme

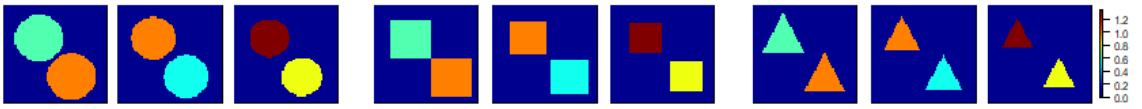

Figure 3.1: Partially overlapping true signals used in simulations, where the size of the signals varying across data sources. Other signals types (homogeneous and minimally overlapping) were also considered with these results presented in the Supplementary Materials.

simulation settings with homogeneous and minimally-overlapping signals, but we present these results in the Supplementary Materials due to space constraints. The scalar outcome variable is then generated under a scalar-on-image regression model based on the true image without noise. The ratio of the mean function variance and residual term variance is set to be 9. For model fitting, we use a working image that is obtained by corrupting the true image with additive noise, and explore the performance for both known and unknown error covariances. We simulate data with training and test sample sizes as 200 for all three groups.

Competing Methods and Evaluation Metrics: We compare the performance of the proposed projected group lasso ('p_glasso') and the projected group bridge ('p_gbridge') methods with several competing methods that (i) fit the model separately for each data source without accounting for noise in images, including the WNET method (Reiss et al., 2015) which first performs wavelet transform and then applies elastic net penalty regression, and the WPCR method (Reiss et al., 2015) which first performs wavelet transform then applies principal component regression; (ii) fit the model separately to each data source with noise correction, as in the method in Loh and Wainwright (2012) that uses a similar projected gradient descent algorithm as in (3.3.2), but with L_1 penalty on the wavelet coefficients that is not equipped for multi-task learning ('p_lasso'); and (iii) multi-task learning approaches involving group lasso and group bridge penalties, but without noise correction as in (3.2.4). The WNET and WPCR methods are implemented in `R` package `refund.wave` (Reiss et al., 2015). We utilize the `R` package `grpreg` (Breheny and Huang, 2015) for implementing the group lasso without noise correction, while we use a hierarchical representation to implement the group bridge without measurement error, as detailed in the Supplementary Materials.

We evaluate the performance of different methods using out-of-sample prediction in a testing sample measured via the prediction mean squared error (PMSE), as well as the accuracy in recovering the functional regression coefficient in terms of bias and area under the curve (AUC) that measures the ability to distinguish non-zero and zero signals (results in Table 3.1). The PMSE is calculated for the test set and standardized by the variance of the training set. Figures 3.2-3.4 illustrate the recovery of the true signals for $SNR = 0$ and $SNR = 3$, while Table 3.1 presents replicate-averaged PMSE, AUC and bias.

3.4.1 Scenario with Known Noise Covariance

The true images are generated independently for each of the data sources, and additive noise is introduced to the true image to obtain the working image used for model fitting. The noise is generated by first generating wavelet coefficients independently from normal distribution with mean 0 and variance at 1/4 (SNR=4) or 1/3 (SNR= 3), and then followed by an inverse wavelet transform. Results are also reported for an ideal case where the true image is used to fit the data. We report results averaged over 100 replicates for each setting.

Results: For the scenarios where the true images are observed, the group bridge method without noise correction shows the best predictive performance. However, the group lasso approach without noise correction, as well as the projected group lasso methods with noise correction have comparable performance with respect to the group bridge method, in terms of coefficient estimation (bias and AUC) that is superior to remaining methods. While it is expected that methods without noise correction will have improved predictive performance when in fact the true image is observed, it is impressive to see that the projected group lasso approach with noise correction performs equally well in terms of signal recovery under these settings, although it cannot guarantee accurate prediction due to the assumption of noise in the images when there is in fact none. For settings of greater interest involving noisy images used for model fitting, the projected group lasso approach has (by far) the best performance in terms of out of sample prediction, bias and AUC, which are almost always significantly improved compared to competing methods. While the empirical performance of the projected group lasso approach in terms of coefficient estimation is supported by

our theoretical results, the superior predictive performance under the projected group lasso approach in the presence of noisy images further highlights the utility of this method. In contrast, the signal recovery accuracy under the projected group bridge is variable and the prediction performance is not always ideal, which is likely due to the lack of theoretical guarantees under the projected gradient descent algorithm that used to compute parameter estimates corresponding to the group bridge penalty. The following Figures 3.2-3.4 illustrate that while the proposed approaches are able to adequately recover the true signals, some competing approaches (such as WPCR) have a particularly poor signal recovery.

3.4.2 Scenario with Unknown Noise Covariance

In this scenario, we assume Σ_u is unknown and needs to be estimated from an external validation sample, which is assumed to be a valid estimate for the noise covariance in the training and the test data sets as well, as in our motivating ADNI analysis. This set-up is designed to mimic the scenario for ADNI data analysis (see Section 3.5 for more details), and does not use the validation data to inform any other aspect of the modeling or prediction conducted on the training and test data sets beyond computing the error covariance. Unlike the set-up with known noise-covariance, the images for each sample were linked across the three data sources by first generating a true image independently for each sample, and then corrupting these images with additive noise across the three data sources. This scenario leads to common patterns in images across the three data sources that are distorted by noise, and enables one to empirically estimate the noise covariance. We generate the data with varying signal-to-noise ratios (6, 4, or 3) and for 100 replicates for each simulation set-up.

Results: The projected group lasso method consistently has the best prediction performance across all signal shapes and signal-to-noise ratios that is significantly improved compared to the other approaches. In terms of coefficient estimation, both the projected group bridge as well as the projected group lasso methods consistently have significantly improved performance in terms of bias and AUC compared to the other methods. Compared to the case with known noise covariance, the relative performance (bias and AUC) under the projected

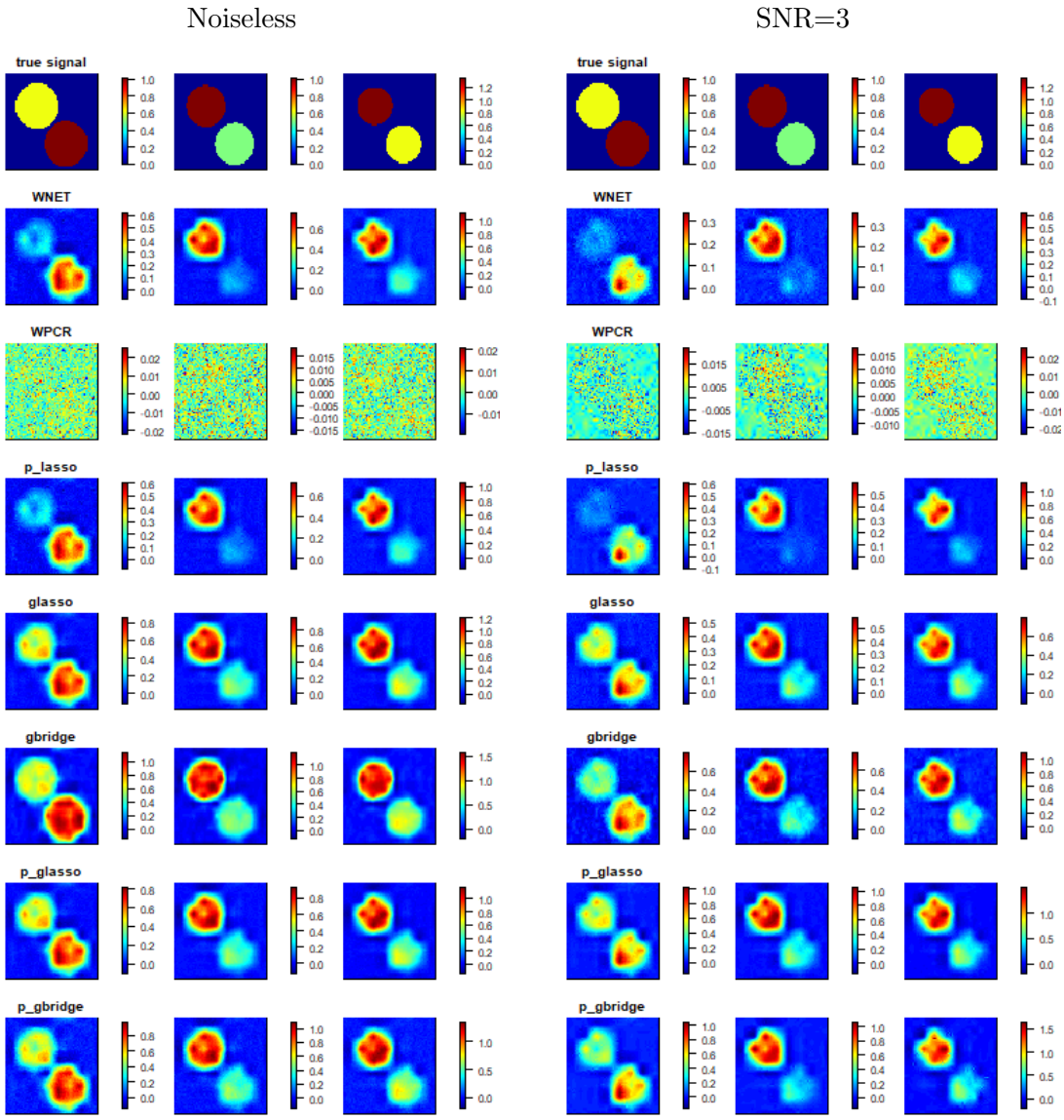

Figure 3.2: Estimated Functional Regression Coefficients with Known Noise Covariance (Round Type) corresponding to the case with images with no measurement error (noiseless) and corresponding to noisy images with SNR=3. The different rows in the Figure depict the true signal, and the estimated signals under WNET, WPCR, projected Lasso, group lasso without noise correction, group bridge without noise correction, projected group lasso with noise correction, and projected group bridge with noise correction.

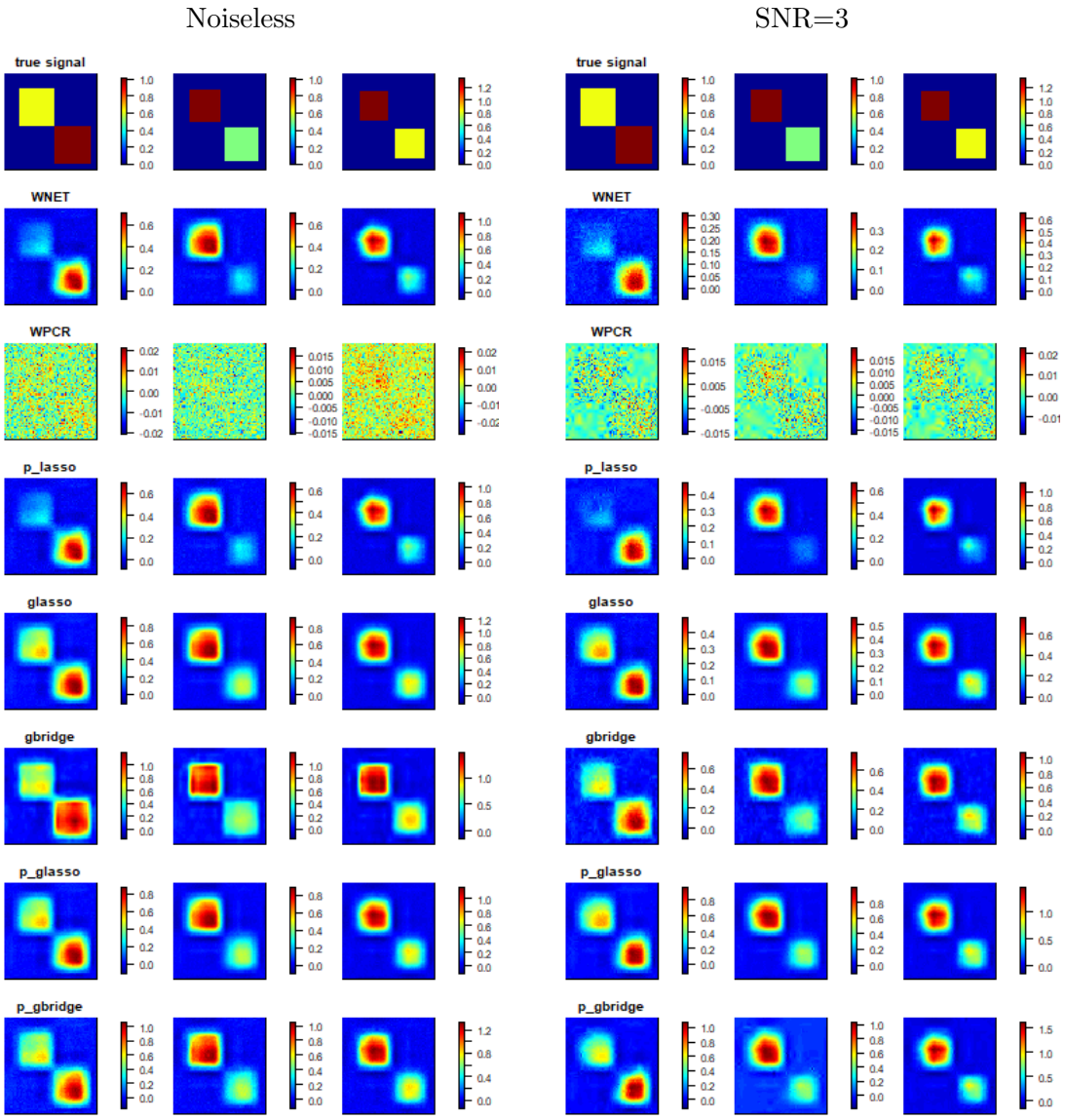

Figure 3.3: Estimated Functional Regression Coefficients with Known Noise Covariance (Square Type) corresponding to the case with images with no measurement error (noiseless) and corresponding to noisy images with SNR=3. The different rows in the Figure depict the true signal, and the estimated signals under WNET, WPCR, projected Lasso, group lasso without noise correction, group bridge without noise correction, projected group lasso with noise correction, and projected group bridge with noise correction.

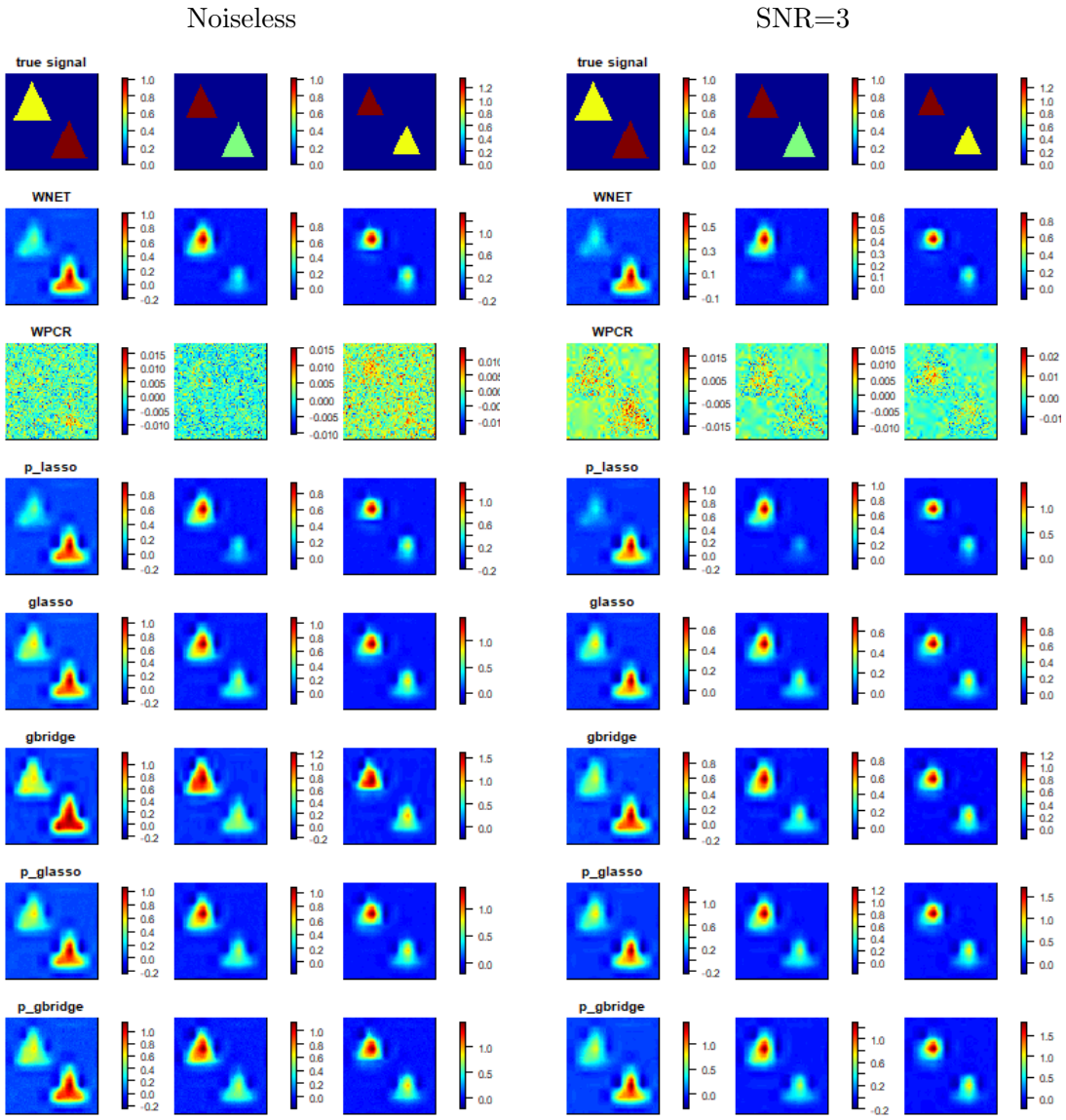

Figure 3.4: Estimated Functional Regression Coefficients with Known Noise Covariance (Triangle Type) corresponding to the case with images with no measurement error (noiseless) and corresponding to noisy images with SNR=3. The different rows in the Figure depict the true signal, and the estimated signals under WNET, WPCR, projected Lasso, group lasso without noise correction, group bridge without noise correction, projected group lasso with noise correction, and projected group bridge with noise correction.

group lasso method slightly deteriorates in the case of unknown noise covariance and becomes more at par with the projected group bridge in terms of signal recovery. However in our experience, the performance of the projected group bridge was occasionally sensitive to the starting values, whereas the projected group lasso method produces more stable results that is consistent with the theoretical guarantees under Theorem 3.3.3.

3.4.3 Additional Simulations with Other Signal Patterns

To aim with understanding of the utility of the proposed methods on various signal patterns, we have included two extreme simulation scenarios. One scenario utilizes a true signal pattern of homogeneous type, while the other utilizes a minimally-overlapping type, as shown in Figure 3.5. We implemented the same simulation settings as in the main manuscript for these two additional signal patterns in the cases of known and unknown noise covariance. The results for prediction mean squared errors (PMSE), bias and AUC metrics are summarized in Table 3.2 and 3.3.

In the scenario with homogeneous signals and in the presence of measurement error, the projected group lasso has the best performance in the case of unknown and known noise covariance, with the projected group bridge also performing well in the case of unknown noise covariance. In the noiseless case, the group bridge and group lasso penalties without noise correction often have the best prediction performance, as expected. The relative improvements under the projected approaches with noise correction is quite similar, and sometimes stronger, compared to the results presented in the main manuscript corresponding to partially overlapping signals.

The scenario with minimally-overlapping signals where the non-zero regions barely intersect across data sources is expected to be more challenging for the group penalty based methods, due to the largely disjoint signal patterns. In spite of this, the projected group lasso has the superior or close to optimal prediction performance across all settings involving noisy images, corresponding to the round and square signal types. For the triangular signals and in the presence of measurement error, the performance is somewhat mixed with superior or

			Known Noise Covariance									*Unknown Noise Covariance*								
			Noiseless			SNR=4			SNR=3			SNR=6			SNR=4			SNR=3		
			G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3
PMSE	round	WNET	0.67	0.58	0.51	0.83	0.77	0.75	0.89	0.82	0.77	0.81	0.73	0.69	0.85	0.77	0.72	0.89	0.82	0.78
		WPCR	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01
		p_lasso	0.68	0.60	0.53	0.81	0.74	0.70	0.92	0.83	0.74	0.77	0.68	0.62	0.83	0.72	0.67	0.89	0.79	0.74
		glasso	0.42	0.36	0.33	0.65	0.57	0.56	0.71	0.63	0.61	0.73	0.66	0.62	0.80	0.71	0.68	0.84	0.77	0.74
		gbridge	**0.32**	**0.34**	**0.29**	0.71	0.67	0.64	0.81	0.74	0.69	**0.64**	0.63	0.56	0.77	0.72	0.68	0.87	0.79	0.74
		p_glasso	0.44	0.37	0.35	**0.58**	**0.52**	**0.50**	**0.67**	**0.60**	**0.58**	**0.62**	**0.54**	**0.49**	**0.69**	**0.57**	**0.55**	**0.75**	**0.63**	**0.61**
		p_gbridge	0.47	0.43	0.37	0.72	0.65	0.65	0.83	0.73	1.00	0.70	0.59	0.55	0.81	0.61	0.63	0.89	0.70	0.71
	square	WNET	0.69	0.59	0.53	0.84	0.77	0.74	0.88	0.81	0.78	0.79	0.73	0.69	0.84	0.78	0.72	0.86	0.82	0.78
		WPCR	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01
		p_lasso	0.69	0.61	0.54	0.81	0.73	0.70	0.88	0.79	0.75	0.75	0.68	0.63	0.81	0.73	0.67	0.84	0.79	0.77
		glasso	0.45	0.37	0.35	0.67	0.59	0.57	0.71	0.63	**0.61**	0.72	0.65	0.62	0.79	0.72	0.68	0.83	0.78	0.75
		gbridge	**0.36**	**0.35**	**0.32**	0.73	0.67	0.65	0.80	0.73	0.71	**0.63**	0.60	0.58	0.74	0.71	0.66	0.84	0.78	0.75
		p_glasso	0.47	0.39	0.36	**0.63**	**0.52**	**0.52**	**0.68**	**0.59**	**0.60**	**0.61**	**0.53**	**0.50**	**0.68**	**0.59**	**0.55**	**0.72**	**0.64**	**0.64**
		p_gbridge	0.52	0.44	0.40	1.13	0.61	0.66	0.86	0.78	0.81	0.72	0.56	0.56	0.80	0.64	0.64	0.83	0.70	0.71
	triangle	WNET	0.46	0.54	0.55	0.67	0.72	0.70	0.72	0.75	0.74	0.60	0.66	0.65	0.66	0.71	0.70	0.71	0.75	0.74
		WPCR	1.01	1.01	1.01	1.01	1.02	1.01	1.01	1.01	1.01	1.01	1.01	1.00	1.01	1.01	1.01	1.01	1.01	1.01
		p_lasso	0.48	0.56	0.55	0.62	0.70	0.66	0.68	0.73	0.71	0.55	0.61	0.61	0.61	0.66	0.66	0.67	0.70	0.71
		glasso	0.36	0.39	0.41	0.57	0.58	0.59	0.62	0.63	0.62	0.56	0.60	0.58	0.64	0.66	0.65	0.69	0.71	0.69
		gbridge	**0.30**	**0.35**	**0.39**	0.55	0.58	0.59	0.59	0.62	0.63	0.48	0.54	0.53	0.56	0.60	0.63	0.62	0.65	0.67
		p_glasso	0.38	0.40	0.42	**0.51**	**0.55**	**0.56**	**0.57**	**0.59**	**0.61**	**0.46**	**0.51**	**0.51**	**0.52**	**0.56**	**0.57**	**0.58**	**0.61**	**0.63**
		p_gbridge	0.42	0.49	0.50	0.59	0.62	0.66	0.69	0.68	0.72	0.55	0.60	0.64	0.58	0.60	0.67	0.63	0.66	0.70
Bias	round	WNET	0.25	0.19	0.21	0.27	0.20	0.24	0.27	0.21	0.25	0.27	0.20	0.23	0.27	0.21	0.24	0.27	0.21	0.24
		WPCR	0.29	0.24	0.27	0.29	0.23	0.27	0.29	0.23	0.27	0.29	0.23	0.27	0.29	0.23	0.27	0.29	0.23	0.27
		p_lasso	0.25	0.19	0.21	0.25	0.19	0.22	0.26	0.20	0.22	0.25	0.19	0.21	0.25	0.19	0.22	0.26	0.20	0.22
		glasso	0.18	**0.13**	**0.16**	0.23	0.17	0.21	0.24	0.18	0.22	0.26	0.20	0.24	0.27	0.21	0.24	0.28	0.22	0.26
		gbridge	**0.17**	0.14	0.17	0.23	0.19	0.23	0.24	0.19	0.23	0.23	0.19	0.23	0.24	0.20	0.24	0.26	0.21	0.24
		p_glasso	0.18	0.14	0.17	**0.19**	**0.14**	**0.17**	**0.20**	**0.14**	**0.18**	**0.21**	0.16	0.19	**0.21**	**0.15**	**0.18**	**0.22**	**0.16**	0.19
		p_gbridge	0.20	0.16	0.19	0.19	0.15	0.18	0.21	0.16	0.20	**0.21**	**0.15**	**0.18**	**0.21**	**0.15**	**0.18**	**0.22**	**0.16**	**0.18**
	square	WNET	0.22	0.16	0.18	0.23	0.18	0.20	0.24	0.18	0.20	0.23	0.17	0.19	0.24	0.18	0.20	0.24	0.18	0.20
		WPCR	0.26	0.20	0.23	0.25	0.19	0.23	0.25	0.19	0.22	0.26	0.19	0.22	0.26	0.19	0.23	0.25	0.19	0.22
		p_lasso	0.22	0.16	0.18	0.22	0.16	0.19	0.23	0.17	0.18	0.22	0.16	0.18	0.22	0.16	0.18	0.22	0.17	0.19
		glasso	**0.17**	**0.12**	**0.15**	0.20	0.15	0.18	0.21	0.15	0.18	0.23	0.17	0.20	0.24	0.18	0.21	0.25	0.19	0.22
		gbridge	**0.17**	0.13	0.17	0.22	0.17	0.21	0.22	0.17	0.20	0.21	0.17	0.21	0.22	0.17	0.21	0.23	0.18	0.21
		p_glasso	**0.17**	**0.12**	**0.15**	**0.17**	**0.12**	**0.15**	**0.18**	**0.13**	**0.15**	**0.19**	0.14	**0.16**	**0.19**	0.14	**0.16**	**0.19**	0.14	**0.17**
		p_gbridge	0.19	0.15	0.17	0.19	0.13	0.16	0.19	0.14	0.16	**0.19**	**0.13**	**0.16**	**0.19**	**0.13**	**0.16**	**0.19**	**0.13**	**0.16**
	triangle	WNET	0.11	0.09	0.10	0.12	0.10	0.10	0.12	0.09	0.10	0.12	0.09	0.10	0.12	0.10	0.10	0.12	0.10	0.10
		WPCR	0.14	0.10	0.11	0.14	0.11	0.11	0.14	0.10	0.11	0.14	0.11	0.11	0.14	0.11	0.11	0.14	0.10	0.11
		p_lasso	0.11	0.09	0.10	0.11	0.09	0.09	0.11	0.09	0.09	0.11	0.09	0.10	0.11	0.09	0.09	0.11	0.09	**0.09**
		glasso	0.09	**0.08**	**0.09**	0.11	0.08	0.09	0.11	0.09	0.09	0.12	0.09	0.10	0.13	0.10	0.11	0.13	0.10	0.11
		gbridge	**0.09**	0.08	0.10	0.11	0.08	0.10	0.11	0.08	0.10	0.11	0.09	0.11	0.11	0.09	0.11	0.12	0.09	0.11
		p_glasso	0.10	**0.08**	**0.09**	**0.09**	**0.08**	**0.09**	**0.10**	**0.08**	**0.09**	**0.10**	**0.08**	**0.09**	**0.10**	**0.08**	**0.09**	**0.10**	**0.08**	**0.09**
		p_gbridge	0.12	0.10	0.12	0.10	0.08	0.10	**0.10**	**0.08**	0.09	0.11	0.09	0.11	**0.10**	**0.08**	0.10	**0.10**	**0.08**	**0.09**
AUC	round	WNET	0.81	0.81	0.89	0.76	0.77	0.83	0.74	0.76	0.82	0.77	0.78	0.85	0.76	0.76	0.83	0.73	0.75	0.83
		WPCR	0.53	0.53	0.54	0.62	0.62	0.61	0.62	0.61	0.61	0.62	0.61	0.60	0.62	0.62	0.61	0.63	0.62	0.61
		p_lasso	0.79	0.80	0.89	0.77	0.78	0.85	0.74	0.73	0.84	0.79	0.80	0.87	0.78	0.77	0.84	0.74	0.75	0.83
		glasso	0.94	**0.95**	**0.97**	0.90	0.92	0.93	0.88	0.90	0.92	0.81	0.84	0.87	0.79	0.81	0.85	0.76	0.78	0.83
		gbridge	**0.95**	0.94	**0.97**	0.86	0.86	0.90	0.85	0.85	0.88	0.85	0.86	0.91	0.84	0.84	0.88	0.82	0.82	0.87
		p_glasso	0.94	**0.95**	**0.97**	**0.93**	**0.94**	**0.96**	**0.92**	**0.93**	**0.95**	0.89	0.90	0.93	0.90	**0.91**	**0.93**	**0.89**	**0.89**	**0.92**
		p_gbridge	0.90	0.91	0.94	0.91	0.91	0.94	0.88	0.87	0.90	**0.90**	**0.91**	**0.94**	**0.91**	**0.91**	**0.93**	**0.89**	**0.89**	**0.92**
	square	WNET	0.81	0.83	0.90	0.77	0.78	0.84	0.74	0.77	0.85	0.78	0.80	0.87	0.77	0.78	0.86	0.76	0.78	0.84
		WPCR	0.53	0.54	0.54	0.66	0.64	0.63	0.64	0.64	0.63	0.65	0.64	0.64	0.65	0.64	0.63	0.65	0.64	0.63
		p_lasso	0.81	0.82	0.89	0.78	0.79	0.85	0.73	0.77	0.85	0.80	0.81	0.88	0.79	0.79	0.87	0.77	0.77	0.85
		glasso	**0.94**	**0.95**	**0.97**	0.90	0.91	0.94	0.89	0.91	0.93	0.82	0.85	0.89	0.79	0.81	0.86	0.77	0.79	0.84
		gbridge	**0.94**	**0.95**	**0.97**	0.86	0.88	0.91	0.85	0.86	0.90	0.87	0.88	0.92	0.85	0.86	0.90	0.83	0.85	0.88
		p_glasso	**0.94**	**0.95**	**0.97**	**0.93**	**0.94**	**0.95**	**0.92**	**0.93**	**0.95**	0.90	0.91	**0.94**	**0.90**	**0.90**	**0.93**	**0.90**	**0.90**	**0.92**
		p_gbridge	0.89	0.91	0.94	0.91	0.91	0.93	0.89	0.88	0.91	**0.91**	**0.92**	**0.94**	**0.90**	**0.91**	**0.94**	**0.90**	**0.90**	**0.93**
	triangle	WNET	0.92	0.90	0.94	0.89	0.86	0.92	0.87	0.85	0.91	0.90	0.89	0.93	0.88	0.86	0.91	0.87	0.85	0.91
		WPCR	0.54	0.55	0.56	0.66	0.65	0.65	0.66	0.65	0.65	0.65	0.64	0.64	0.66	0.64	0.65	0.66	0.65	0.65
		p_lasso	0.92	0.90	0.94	0.90	0.87	0.93	0.87	0.84	0.91	0.91	0.90	0.94	0.89	0.88	0.92	0.87	0.87	0.92
		glasso	**0.98**	**0.98**	**0.98**	0.95	0.96	0.97	0.95	0.96	0.97	0.93	0.94	0.95	0.91	0.91	0.94	0.89	0.90	0.93
		gbridge	**0.98**	**0.98**	**0.98**	0.95	0.95	0.97	0.95	0.95	0.96	**0.96**	0.96	0.97	0.94	0.95	0.96	0.94	0.94	0.95
		p_glasso	**0.98**	**0.98**	**0.98**	**0.97**	**0.97**	**0.98**	**0.96**	**0.97**	**0.97**	**0.96**	**0.96**	**0.97**	**0.96**	**0.96**	**0.97**	**0.95**	**0.96**	**0.97**
		p_gbridge	0.95	0.95	0.96	0.95	0.95	0.96	0.95	0.95	0.97	**0.96**	0.96	0.96	**0.95**	**0.95**	0.96	**0.95**	0.95	0.96

Table 3.1: Summary for simulation results with known and unknown noise covariances

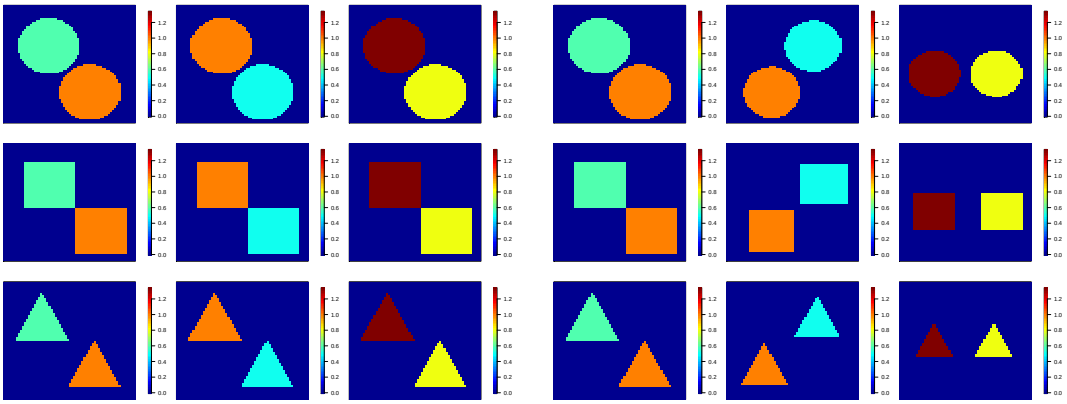

Figure 3.5: True Regression Coefficient Maps for Additional Signal Patterns. Left: Homogeneous Type; Right: Minimally-overlapping Type.

comparable prediction under the projected group lasso compared to other approaches for the overwhelming majority of cases. In terms of parameter estimation, the projected group bridge has the lowest bias in the overwhelmingly large number of cases in the presence of measurement error. However, the projected lasso that performs analysis separately for each data source also has good estimation performance in several cases, which is somewhat expected due to the disjoint nature of the non-zero signal regions. The performance with respect to AUC is more varied, and no particular methods seems to have an edge over other approaches with respect to feature selection.

We see that overall, the advantages of noise correction under grouped penalties are partially eroded when the true signal is disjoint across data sources, as expected. However, the proposed projected group lasso and/or group bridge still have improved or comparable predictive and estimation performance across a majority of settings, which is encouraging. We would note that the proposed approaches are best suited for applications involving some degree of shared patterns across data sources, such as in our motivating neuroimaging application that involves longitudinal progression of Alzheimer's disease where one expects a non-negligible number of brain regions with signals that do not change considerably across visits.

			Known Noise Covariance									*Unknown Noise Covariance*								
			Noiseless			SNR=4			SNR=3			SNR=6			SNR=4			SNR=3		
			G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3
PMSE	round	WNET	0.71	0.65	0.61	0.87	0.84	0.82	0.88	0.87	0.85	0.82	0.79	0.78	0.86	0.81	0.81	0.90	0.85	0.84
		WPCR	1.01	1.01	1.01	1.02	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.02	1.01	1.01	1.01	1.01	1.01	1.01
		p_lasso	0.71	0.66	0.62	0.87	0.79	0.79	0.87	0.86	0.86	0.78	0.73	0.74	0.84	0.78	0.76	0.88	0.82	0.81
		glasso	0.41	**0.37**	0.35	0.67	0.61	0.62	0.71	0.66	0.67	0.81	0.76	0.75	0.86	0.80	0.79	0.90	0.83	0.82
		gbridge	**0.38**	**0.38**	**0.28**	0.82	0.79	0.71	0.88	0.84	0.81	0.76	0.69	**0.63**	0.82	0.78	0.75	0.90	0.83	0.83
		p_glasso	0.44	**0.39**	0.38	**0.60**	**0.55**	**0.54**	**0.68**	**0.61**	**0.64**	**0.72**	**0.64**	**0.64**	**0.77**	**0.68**	**0.69**	**0.81**	**0.72**	**0.73**
		p_gbridge	0.46	0.45	0.38	0.73	0.73	0.72	0.82	0.75	0.83	0.86	0.73	0.79	0.92	0.77	0.90	0.97	0.80	0.91
	square	WNET	0.73	0.64	0.57	0.84	0.82	0.82	0.87	0.83	0.83	0.81	0.78	0.75	0.84	0.81	0.78	0.89	0.85	0.84
		WPCR	1.01	1.02	1.01	1.01	1.01	1.01	1.01	1.00	1.01	1.02	1.01	1.01	1.01	1.01	1.01	1.02	1.01	1.01
		p_lasso	0.74	0.65	0.58	0.83	0.79	0.77	0.86	0.79	0.83	0.79	0.74	0.70	0.81	0.77	0.75	0.88	0.84	0.81
		glasso	0.46	**0.39**	0.37	0.68	0.61	0.62	0.71	0.65	**0.66**	0.81	0.76	0.74	0.86	0.79	0.77	0.91	0.86	0.85
		gbridge	**0.40**	**0.38**	**0.29**	0.78	0.75	0.69	0.85	0.81	0.75	**0.74**	0.69	**0.63**	0.80	0.77	0.73	0.91	0.83	0.82
		p_glasso	0.47	0.41	0.39	**0.62**	**0.56**	**0.56**	**0.67**	**0.60**	**0.68**	**0.72**	**0.64**	**0.62**	**0.74**	**0.66**	**0.64**	**0.80**	**0.71**	**0.71**
		p_gbridge	0.54	0.47	0.41	0.73	0.67	0.79	0.80	0.69	0.78	0.83	0.71	0.73	0.84	0.72	0.80	0.92	0.79	0.88
	triangle	WNET	0.51	0.51	0.43	0.69	0.67	0.65	0.73	0.73	0.70	0.64	0.62	0.60	0.69	0.71	0.67	0.73	0.73	0.69
		WPCR	1.02	1.01	1.02	1.02	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01
		p_lasso	0.53	0.52	0.45	0.64	0.64	0.59	0.69	0.71	0.66	0.58	0.57	0.55	0.64	0.66	0.61	0.73	0.69	0.64
		glasso	0.37	**0.36**	0.31	0.56	0.54	0.53	0.61	0.59	**0.58**	0.65	0.61	0.59	0.71	0.71	0.68	0.76	0.75	0.72
		gbridge	**0.34**	**0.35**	**0.28**	0.56	0.54	0.53	0.59	0.58	**0.58**	0.57	0.54	0.51	0.62	0.62	0.59	0.66	0.63	0.62
		p_glasso	0.38	0.37	0.33	**0.51**	**0.49**	**0.49**	**0.57**	**0.56**	**0.57**	**0.51**	**0.49**	**0.46**	**0.54**	**0.56**	**0.52**	**0.60**	**0.57**	**0.56**
		p_gbridge	0.45	0.43	0.36	0.58	0.54	0.60	0.62	0.61	0.63	0.58	0.57	0.58	0.58	**0.58**	0.59	0.65	0.62	0.68
Bias	round	WNET	0.25	0.23	0.31	0.27	0.25	0.36	0.27	0.25	0.36	0.27	0.24	0.35	0.27	0.25	0.35	0.28	0.25	0.36
		WPCR	0.29	0.28	0.39	0.29	0.27	0.39	0.29	0.27	0.39	0.29	0.27	0.39	0.29	0.27	0.39	0.29	0.27	0.39
		p_lasso	0.26	0.23	0.32	0.26	0.23	0.33	0.26	0.24	0.34	0.25	0.23	0.32	0.25	0.23	0.33	0.26	0.24	0.33
		glasso	**0.18**	**0.15**	0.23	0.23	0.20	0.31	0.24	0.21	0.31	0.28	0.25	0.36	0.29	0.26	0.37	0.30	0.27	0.38
		gbridge	**0.18**	0.17	**0.22**	0.26	0.24	0.33	0.26	0.24	0.34	0.25	0.23	0.31	0.26	0.24	0.33	0.27	0.24	0.34
		p_glasso	**0.18**	0.16	0.24	**0.19**	**0.16**	**0.24**	**0.20**	**0.18**	**0.26**	**0.23**	**0.20**	**0.29**	**0.23**	**0.20**	**0.29**	**0.24**	**0.21**	**0.30**
		p_gbridge	0.20	0.18	0.25	0.20	0.18	0.26	0.22	0.19	0.28	**0.23**	**0.20**	**0.30**	**0.24**	**0.20**	**0.30**	**0.24**	**0.21**	**0.31**
	square	WNET	0.22	0.20	0.27	0.24	0.22	0.32	0.24	0.22	0.32	0.24	0.22	0.30	0.24	0.22	0.31	0.24	0.22	0.31
		WPCR	0.26	0.24	0.34	0.26	0.24	0.34	0.25	0.24	0.34	0.26	0.24	0.34	0.26	0.24	0.34	0.25	0.24	0.34
		p_lasso	0.23	0.20	0.28	0.23	0.20	0.29	0.23	0.20	0.30	0.22	0.20	0.28	0.22	0.21	0.28	0.23	0.21	0.29
		glasso	**0.17**	**0.15**	0.21	0.20	0.18	0.27	0.21	0.19	0.28	0.25	0.23	0.32	0.26	0.24	0.33	0.27	0.24	0.34
		gbridge	0.18	0.16	**0.21**	0.23	0.21	0.29	0.23	0.21	0.30	0.23	0.21	0.29	0.23	0.22	0.30	0.24	0.21	0.30
		p_glasso	0.17	0.15	0.22	**0.17**	**0.15**	**0.22**	**0.18**	**0.16**	**0.24**	**0.21**	**0.18**	**0.26**	**0.20**	**0.18**	**0.26**	**0.21**	**0.18**	**0.26**
		p_gbridge	0.20	0.18	0.24	0.18	0.16	0.25	0.19	0.17	**0.24**	**0.21**	**0.18**	**0.26**	**0.20**	**0.17**	**0.26**	**0.21**	**0.17**	**0.26**
	triangle	WNET	0.11	0.11	0.15	0.12	0.12	0.17	0.13	0.12	0.17	0.12	0.12	0.16	0.12	0.12	0.17	0.12	0.12	0.17
		WPCR	0.15	0.14	0.20	0.14	0.14	0.19	0.14	0.13	0.19	0.14	0.14	0.19	0.14	0.14	0.19	0.14	0.14	0.19
		p_lasso	0.11	0.11	0.15	0.11	0.11	0.15	0.11	0.11	0.15	0.11	0.11	0.15	0.11	0.11	0.15	0.12	0.11	0.15
		glasso	**0.09**	**0.09**	**0.13**	0.11	0.10	0.15	0.11	0.11	0.15	0.13	0.13	0.18	0.14	0.14	0.19	0.15	0.14	0.20
		gbridge	0.10	0.09	**0.13**	0.10	0.10	0.14	0.10	0.10	0.14	0.12	0.11	0.16	0.12	0.12	0.17	0.11	0.11	0.16
		p_glasso	**0.09**	**0.09**	**0.13**	**0.09**	**0.09**	**0.13**	**0.09**	**0.09**	**0.13**	**0.10**	**0.10**	**0.14**	0.10	0.10	**0.14**	**0.10**	**0.09**	**0.14**
		p_gbridge	0.12	0.11	0.15	**0.09**	**0.09**	0.13	**0.09**	**0.09**	**0.13**	**0.10**	**0.10**	0.15	**0.10**	**0.09**	**0.14**	**0.10**	**0.09**	**0.14**
AUC	round	WNET	0.79	0.78	0.81	0.76	0.73	0.73	0.74	0.72	0.76	0.76	0.75	0.77	0.77	0.74	0.76	0.74	0.72	0.75
		WPCR	0.53	0.53	0.53	0.63	0.62	0.62	0.63	0.63	0.62	0.62	0.61	0.62	0.62	0.62	0.63	0.63	0.62	0.62
		p_lasso	0.79	0.78	0.81	0.76	0.75	0.76	0.73	0.72	0.74	0.78	0.77	0.80	0.79	0.74	0.78	0.74	0.72	0.77
		glasso	**0.95**	**0.95**	**0.95**	0.90	0.89	0.88	0.89	0.88	0.87	0.76	0.76	0.77	0.73	0.73	0.74	0.70	0.70	0.71
		gbridge	0.94	0.92	**0.95**	0.82	0.80	0.83	0.83	0.81	0.82	0.82	0.81	0.85	0.81	0.80	0.81	0.81	0.79	0.80
		p_glasso	**0.95**	**0.95**	**0.95**	**0.93**	**0.91**	**0.92**	**0.92**	**0.91**	**0.91**	**0.86**	**0.85**	**0.86**	0.86	**0.84**	**0.85**	**0.85**	**0.84**	**0.85**
		p_gbridge	0.91	0.90	0.92	0.91	0.89	0.89	0.86	0.83	0.85	**0.87**	**0.86**	**0.87**	**0.88**	**0.85**	**0.86**	**0.86**	**0.84**	**0.85**
	square	WNET	0.80	0.78	0.84	0.77	0.75	0.77	0.76	0.75	0.77	0.76	0.76	0.79	0.77	0.72	0.79	0.75	0.75	0.77
		WPCR	0.53	0.53	0.53	0.65	0.66	0.66	0.67	0.66	0.66	0.66	0.66	0.66	0.66	0.67	0.66	0.65	0.66	0.66
		p_lasso	0.77	0.78	0.83	0.78	0.77	0.78	0.76	0.75	0.77	0.80	0.78	0.81	0.78	0.73	0.80	0.76	0.75	0.78
		glasso	**0.94**	**0.93**	0.94	0.90	0.89	0.88	0.89	0.88	0.87	0.76	0.76	0.78	0.73	0.73	0.75	0.71	0.71	0.72
		gbridge	0.94	**0.92**	**0.96**	0.84	0.82	0.84	0.84	0.82	0.84	0.83	0.82	0.85	0.83	0.81	0.83	0.81	0.79	0.81
		p_glasso	**0.94**	**0.93**	0.93	**0.93**	**0.92**	**0.92**	**0.92**	**0.91**	**0.91**	0.87	0.86	**0.87**	0.88	**0.86**	**0.87**	**0.87**	**0.86**	**0.87**
		p_gbridge	0.89	0.88	0.90	0.89	0.88	0.88	0.88	0.84	0.87	**0.88**	**0.88**	**0.87**	**0.89**	**0.86**	**0.88**	**0.88**	**0.85**	**0.87**
	triangle	WNET	0.92	0.91	0.94	0.87	0.86	0.89	0.87	0.84	0.87	0.90	0.88	0.91	0.89	0.85	0.89	0.86	0.84	0.88
		WPCR	0.54	0.54	0.55	0.65	0.66	0.64	0.66	0.66	0.66	0.65	0.65	0.65	0.66	0.65	0.65	0.66	0.66	0.66
		p_lasso	0.92	0.90	0.94	0.88	0.86	0.91	0.88	0.84	0.88	0.91	0.89	0.92	0.90	0.86	0.90	0.87	0.85	0.90
		glasso	**0.98**	**0.98**	**0.98**	0.97	0.96	0.96	0.96	0.95	0.95	0.90	0.89	0.90	0.87	0.86	0.88	0.85	0.84	0.86
		gbridge	0.98	**0.97**	**0.98**	0.96	0.94	0.96	0.96	0.94	0.95	0.95	0.94	0.95	0.93	0.92	0.94	0.94	0.93	0.94
		p_glasso	**0.98**	**0.98**	**0.98**	**0.98**	**0.97**	**0.97**	**0.97**	**0.97**	**0.96**	**0.96**	**0.95**	**0.96**	**0.96**	**0.95**	**0.96**	**0.96**	**0.95**	**0.95**
		p_gbridge	0.96	0.95	0.96	0.97	0.95	0.97	0.95	0.94	0.94	**0.96**	**0.95**	**0.96**	**0.96**	**0.95**	**0.96**	**0.96**	0.94	**0.95**

Table 3.2: Summary for simulation results for homogeneous signals

			Known Noise Covariance									*Unknown Noise Covariance*								
			Noiseless			SNR=4			SNR=3			SNR=6			SNR=4			SNR=3		
			G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3	G1	G2	G3
PMSE	round	WNET	0.69	0.60	0.51	0.84	0.79	0.74	0.88	0.81	0.78	0.83	0.76	0.66	0.85	0.76	0.72	0.89	0.82	0.78
		WPCR	1.02	1.02	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.02	1.01	1.01	1.01	1.01
		p_lasso	0.70	0.61	0.52	0.82	0.77	0.67	0.89	0.77	0.76	0.80	0.69	0.60	0.83	0.73	0.67	0.89	0.80	0.74
		glasso	0.57	0.52	0.42	0.78	0.74	0.66	**0.82**	0.76	0.71	0.76	0.70	0.62	0.81	0.73	0.68	0.85	0.78	0.74
		gbridge	**0.35**	**0.41**	**0.33**	0.80	0.84	0.74	0.88	0.86	0.79	**0.70**	0.69	0.61	0.79	0.79	0.74	0.91	0.87	0.78
		p_glasso	0.59	0.55	0.44	**0.75**	**0.68**	**0.57**	**0.81**	**0.72**	**0.66**	**0.70**	**0.63**	**0.53**	**0.75**	**0.67**	**0.60**	**0.83**	**0.72**	**0.65**
		p_gbridge	0.66	0.63	0.47	0.91	0.76	0.70	1.01	0.79	0.85	0.83	0.69	0.60	0.90	0.73	0.74	0.99	0.78	0.74
	square	WNET	0.70	0.60	0.50	0.83	0.78	0.70	0.86	0.83	0.74	0.80	0.73	0.66	0.85	0.78	0.71	0.87	0.83	0.76
		WPCR	1.02	1.01	1.01	1.01	1.01	1.01	1.01	1.02	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01
		p_lasso	0.71	0.61	0.52	0.81	0.75	0.65	0.85	0.80	0.73	0.76	0.69	0.60	0.82	0.73	0.65	0.87	0.81	0.71
		glasso	0.64	0.57	0.43	0.82	0.77	0.65	0.85	0.81	0.70	0.78	0.71	0.62	0.82	0.76	0.67	0.86	0.81	0.72
		gbridge	**0.42**	**0.43**	**0.34**	0.80	0.79	0.68	0.88	0.82	0.73	**0.70**	0.70	0.62	0.80	0.80	0.68	0.89	0.86	0.75
		p_glasso	0.66	0.58	0.45	**0.76**	**0.71**	**0.57**	**0.82**	**0.76**	**0.63**	**0.72**	**0.65**	**0.54**	**0.76**	**0.70**	**0.59**	**0.82**	**0.76**	**0.63**
		p_gbridge	0.74	0.66	0.48	0.91	0.77	0.66	0.96	0.98	0.84	0.84	0.70	0.58	0.89	0.75	0.70	0.97	0.83	0.71
	triangle	WNET	0.46	0.53	0.46	0.69	0.68	0.64	0.73	0.71	0.68	0.61	0.63	0.57	0.67	0.68	0.64	0.73	0.72	0.67
		WPCR	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.01	1.02	1.01	1.01
		p_lasso	0.48	0.54	0.48	**0.62**	**0.65**	**0.58**	0.73	0.70	0.66	**0.54**	0.59	**0.52**	0.61	**0.62**	**0.59**	0.72	**0.68**	**0.62**
		glasso	0.47	0.55	0.46	0.70	0.71	0.64	0.73	0.74	0.68	0.62	0.65	0.58	0.69	0.70	0.64	0.74	0.75	0.69
		gbridge	**0.33**	**0.42**	**0.37**	**0.62**	**0.65**	**0.60**	**0.69**	**0.67**	**0.64**	**0.53**	**0.57**	0.53	**0.59**	**0.64**	0.61	**0.67**	**0.68**	0.67
		p_glasso	0.49	0.57	0.48	**0.63**	**0.64**	**0.59**	**0.70**	0.71	**0.65**	**0.55**	0.60	**0.51**	**0.60**	**0.64**	**0.58**	**0.68**	**0.69**	**0.64**
		p_gbridge	0.54	0.66	0.54	0.82	0.71	0.86	0.76	0.72	0.69	0.60	0.64	0.57	0.65	**0.65**	0.65	0.70	0.70	0.68
Bias	round	WNET	0.25	0.18	0.20	0.27	0.21	0.24	0.27	0.21	0.25	0.27	0.21	0.23	0.27	0.21	0.24	0.28	0.21	0.25
		WPCR	0.29	0.24	0.28	0.29	0.23	0.28	0.29	0.24	0.27	0.29	0.23	0.27	0.29	0.24	0.28	0.29	0.23	0.27
		p_lasso	0.25	0.19	0.21	0.25	0.20	0.21	0.26	0.20	0.23	0.25	0.19	0.21	0.26	0.19	0.21	**0.26**	0.20	0.22
		glasso	0.23	0.18	**0.19**	0.26	0.21	0.23	0.27	0.21	0.24	0.27	0.21	0.23	0.27	0.21	0.24	0.28	0.22	0.25
		gbridge	**0.18**	**0.17**	**0.19**	0.26	0.22	0.26	0.26	0.22	0.25	**0.25**	0.21	0.25	0.26	0.22	0.27	0.28	0.23	0.26
		p_glasso	0.24	0.19	**0.20**	**0.24**	0.19	**0.20**	**0.25**	0.19	**0.21**	**0.25**	0.19	0.20	**0.25**	0.19	**0.21**	**0.26**	0.20	0.21
		p_gbridge	0.26	0.21	0.22	**0.25**	**0.18**	**0.20**	**0.26**	**0.19**	**0.21**	**0.25**	**0.18**	**0.19**	**0.24**	**0.18**	**0.20**	**0.26**	**0.18**	**0.20**
	square	WNET	0.22	**0.16**	0.17	0.24	0.18	0.20	0.24	0.19	0.20	0.23	0.17	0.19	0.24	0.18	0.19	0.24	0.19	0.20
		WPCR	0.26	0.20	0.23	0.26	0.20	0.23	0.26	0.20	0.22	0.25	0.20	0.22	0.26	0.20	0.22	0.25	0.20	0.22
		p_lasso	0.23	0.16	0.18	**0.22**	0.17	0.18	**0.23**	**0.17**	**0.18**	**0.22**	**0.16**	0.18	**0.22**	**0.17**	0.18	**0.23**	0.17	0.18
		glasso	0.22	0.17	**0.17**	0.24	0.19	0.19	0.24	0.19	0.20	0.24	0.18	0.19	0.25	0.19	0.20	0.25	0.19	0.20
		gbridge	**0.19**	**0.16**	0.17	0.24	0.19	0.21	0.24	**0.18**	0.20	0.23	0.19	0.22	0.24	0.20	0.22	0.24	0.19	0.21
		p_glasso	0.22	0.17	**0.17**	**0.23**	0.18	**0.17**	**0.23**	**0.18**	**0.18**	**0.22**	0.17	0.17	**0.23**	0.18	**0.17**	**0.23**	0.18	0.18
		p_gbridge	0.25	0.20	0.20	**0.23**	**0.16**	**0.17**	**0.23**	**0.17**	**0.17**	**0.22**	**0.16**	**0.17**	**0.22**	**0.16**	**0.17**	**0.23**	**0.16**	**0.16**
	triangle	WNET	0.11	**0.09**	**0.09**	0.12	**0.09**	**0.09**	0.12	0.10	0.09	0.12	**0.09**	0.09	0.12	0.09	0.10	0.13	0.10	0.09
		WPCR	0.14	0.11	0.11	0.14	0.11	0.11	0.14	0.11	0.11	0.14	0.11	0.11	0.14	0.11	0.11	0.14	0.11	0.11
		p_lasso	0.11	**0.09**	**0.09**	**0.11**	**0.09**	**0.09**	0.12	**0.09**	0.09	**0.11**	**0.09**	**0.09**	**0.11**	**0.09**	**0.09**	0.12	**0.09**	**0.09**
		glasso	0.12	0.10	0.10	0.13	0.10	0.10	0.13	0.10	0.10	0.13	0.10	0.10	0.13	0.10	0.10	0.13	0.10	0.10
		gbridge	**0.10**	**0.09**	0.10	0.12	0.10	0.11	0.12	0.10	0.10	**0.11**	**0.09**	0.10	0.12	0.10	0.10	0.12	0.09	0.10
		p_glasso	0.12	0.10	0.10	0.12	0.10	0.10	0.12	0.10	0.10	0.12	0.10	0.10	0.12	0.10	0.10	0.12	0.10	0.10
		p_gbridge	0.14	0.13	0.13	**0.12**	**0.09**	**0.10**	**0.11**	**0.09**	**0.09**	**0.11**	0.10	0.10	**0.11**	**0.09**	**0.09**	**0.11**	**0.09**	**0.09**
AUC	round	WNET	0.80	0.82	0.90	0.76	0.74	0.84	0.74	**0.73**	0.82	0.77	0.77	0.86	0.77	**0.77**	0.83	0.73	**0.74**	0.82
		WPCR	0.53	0.53	0.54	0.62	0.53	0.56	0.63	0.54	0.57	0.62	0.54	0.56	0.63	0.53	0.56	0.61	0.54	0.55
		p_lasso	0.79	0.81	0.89	0.78	0.75	0.86	0.75	**0.73**	0.82	0.77	0.79	0.87	0.76	**0.77**	0.84	0.73	**0.73**	0.83
		glasso	0.83	0.82	0.94	0.77	0.74	0.88	0.74	0.72	0.85	0.75	0.75	0.87	0.74	0.74	0.84	0.72	**0.72**	0.83
		gbridge	**0.93**	**0.92**	**0.96**	**0.80**	**0.78**	0.86	**0.77**	0.71	0.84	**0.82**	**0.81**	0.89	**0.80**	**0.78**	0.85	**0.76**	**0.73**	0.83
		p_glasso	0.82	0.81	0.94	**0.80**	0.76	**0.90**	**0.78**	**0.73**	**0.88**	0.79	0.77	0.90	**0.79**	0.76	0.88	**0.76**	**0.73**	0.87
		p_gbridge	0.79	0.79	0.91	**0.79**	0.75	0.88	**0.77**	0.70	**0.87**	0.80	**0.80**	**0.91**	**0.80**	0.75	**0.89**	**0.77**	**0.71**	**0.89**
	square	WNET	0.79	0.83	0.91	0.77	0.75	0.85	**0.75**	**0.73**	0.86	0.78	0.78	0.88	0.76	0.75	0.86	**0.78**	**0.74**	0.85
		WPCR	0.53	0.54	0.54	0.65	0.60	0.57	0.66	0.60	0.58	0.65	0.59	0.57	0.65	0.59	0.58	0.65	0.58	0.57
		p_lasso	0.78	0.82	0.91	**0.79**	0.75	0.87	**0.76**	**0.74**	0.86	0.80	0.79	0.89	0.78	**0.77**	0.88	**0.78**	**0.73**	0.86
		glasso	0.79	0.80	0.94	0.73	0.73	0.89	0.72	**0.72**	0.88	0.75	0.75	0.90	0.72	0.72	0.87	0.72	**0.72**	0.87
		gbridge	**0.91**	**0.90**	**0.97**	**0.80**	0.75	0.89	**0.76**	0.70	0.87	**0.84**	**0.81**	0.90	**0.79**	0.75	0.88	0.76	**0.71**	0.87
		p_glasso	0.79	0.80	0.94	0.78	0.74	**0.92**	**0.76**	**0.73**	**0.91**	0.79	0.76	0.92	0.77	0.73	0.91	0.77	**0.72**	0.90
		p_gbridge	0.77	0.79	0.91	**0.80**	**0.77**	**0.92**	**0.77**	0.70	**0.90**	0.81	**0.82**	**0.93**	**0.79**	0.74	**0.92**	**0.78**	**0.71**	**0.92**
	triangle	WNET	0.93	0.90	0.96	0.88	0.85	0.94	**0.86**	**0.85**	0.94	0.90	0.87	0.96	0.89	0.83	0.94	0.86	**0.83**	0.94
		WPCR	0.54	0.54	0.55	0.66	0.59	0.58	0.66	0.59	0.59	0.66	0.60	0.58	0.66	0.60	0.58	0.65	0.59	0.58
		p_lasso	0.92	0.89	0.96	**0.90**	**0.86**	0.95	**0.86**	0.83	0.94	0.91	0.88	0.96	0.90	**0.84**	0.95	0.87	**0.83**	0.95
		glasso	0.92	0.88	0.96	0.86	0.83	0.94	0.84	0.81	0.93	0.88	0.85	0.95	0.86	0.81	0.93	0.84	0.81	0.92
		gbridge	**0.97**	**0.94**	**0.98**	**0.91**	**0.87**	**0.96**	**0.87**	**0.84**	**0.95**	**0.92**	**0.90**	**0.97**	**0.91**	0.83	0.95	**0.88**	0.80	0.94
		p_glasso	0.91	0.87	0.96	0.89	0.85	0.95	**0.87**	0.83	0.94	0.90	0.86	0.96	0.89	0.82	0.95	0.86	0.81	0.94
		p_gbridge	0.90	0.86	0.94	**0.91**	**0.87**	**0.96**	**0.86**	0.81	**0.95**	**0.92**	**0.89**	**0.96**	**0.91**	**0.84**	**0.96**	**0.88**	0.80	**0.96**

Table 3.3: Summary for simulation results for minimally overlapping signals

3.4.4 Sensitivity Analysis to Noise Covariance Estimation Bias

As a sensitivity test, we have reported the error in noise covariance estimation and investigated the trade-off between the covariance estimation error and performance metrics corresponding to simulation settings in the unknown noise covariance scenario in the main manuscript. In order to obtain a range of values for covariance error estimation that is required to investigate the trade-off between variance estimation error and performance metrics, we used three different validation sample sizes (200,100, and 50) corresponding to the simulation scenario with unknown noise covariance and signal-to-noise ratio of 3. Figure 3.6 reports the trade-off under the projected group lasso method for 50 replicates across different validation sample sizes. In particular, this Figure reports the ratio of PMSE/Bias/AUC metrics between the group lasso method without noise correction and the projected group lasso, with a lower value in PMSE/Bias metrics or a higher value in AUC metric, implying the improvement under the projected group lasso over the uncorrected version. From Figure 3.6, it is clear that decreasing the validation sample resulted in an increase in the covariance estimation error as expected; however, this increase was fairly limited and in general the covariance estimation error values were manageable in our experience for diagonal error covariances. More importantly, the deterioration in the performance metrics was limited as the validation sample size was decreased, although the rates of change in performance varied across the three signal types, which is to be expected.

We also investigated one additional simulation scenario where we intentionally used a mis-specified measurement error covariance to fit the data, which corresponds to the set-up where the noise covariance can not be confidently estimated. Random errors were independently added to each diagonal element in the true noise covariance matrix to make the working noise covariance. The added errors were normally distributed, and the standard deviation of the normal distribution was made to be 0.1, 0.2 or 0.3 times of the value in the corresponding cell of the true noise covariance. Figure 3.7 shows the scatterplots between the bias in the working noise covariance averaged over voxels versus the ratio of PMSE/Bias/AUC metrics between the projected group lasso method and the group lasso method without noise correction, over varying degrees of mis-specification in the er-

ror covariance. These results revealed a robust performance under the proposed method employing a biased working noise covariance, which is similar to our findings presented in the scenario above investigating the trade-off between noise covariance estimation bias and model performance.

Based on these results, we conclude that the methods with noise correction are fairly robust in their performance with respect to the estimation accuracy of the unknown noise covariance matrix. In extreme settings with no or limited validation samples to estimate the measurement error covariance, we expect the performance of the proposed approaches to deteriorate but we would note that this is expected of essentially all measurement error corrected approaches that depend on error covariance estimation. It is important to note that error covariance estimation is not the primary goal of our work - instead we propose an error covariance estimate that results in theoretically justified and numerically accurate regression parameter estimates, even when the error covariance estimation is not perfect.

3.4.5 Summary of Results

The results clearly illustrate the benefits of multi-task learning in scalar-on-image regression, given that the performance of those approaches that fit the model separately for each data source are often inferior, even when they account for the presence of noise. In addition, multi-task learning without noise correction results in sub-optimal performance in the presence of noisy images, compared to projected group lasso and group bridge approaches. Moreover, the advantages of noise correction under grouped penalties are accentuated under homogeneous signals, and are partially eroded under minimally overlapping true signals across data sources, as expected. However, the proposed projected group lasso and/or group bridge still have improved or comparable predictive and estimation performance across the overwhelming majority of settings for the latter case. Another factor that influences the performance of the proposed methods is the accuracy of the estimated $\mathbf{\Sigma}_u$ used in the working model. In our experience, the prediction and selection performance are largely robust to mis-specification of the noise covariance, as long as the biases for the estimated noise covariance are not overly pronounced. In practical applications, the proposed approach is

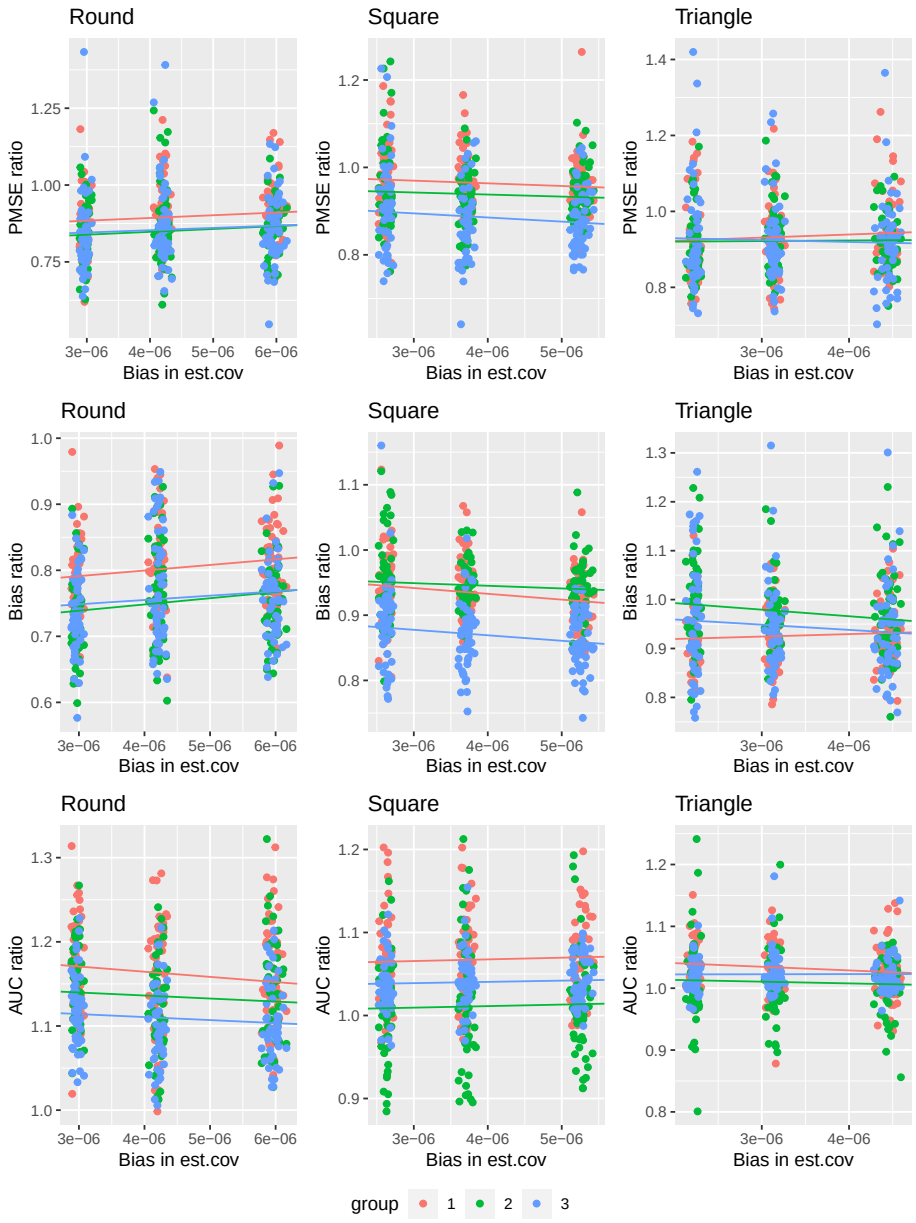

Figure 3.6: Trade-off between Noise Covariance Estimation Error and MSE/Bias/AUC Metrics, shown as the ratios between the projected group lasso method and group lasso method without noise correction, over varying validation sets used to compute the noise covariance. The three colors represent the three datasets.

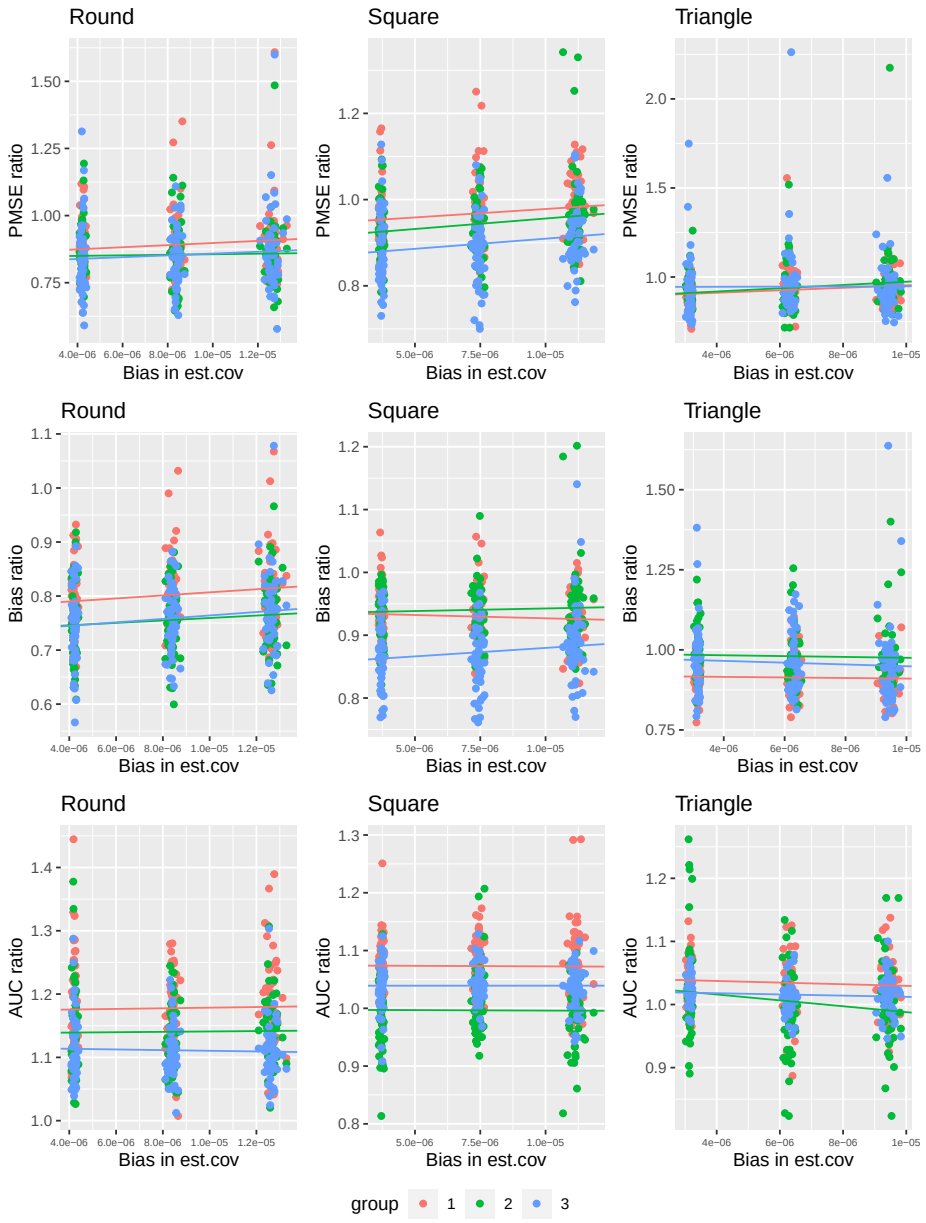

Figure 3.7: Scatterplots between Bias in Noise Covariance Estimation and MSE/Bias/AUC Metrics, shown as the ratios between the projected group lasso method and group lasso method without noise correction. The different colors correspond to the three datasets and the dots represent different replicates.

best suited in settings where validation datasets with non-negligible sample sizes are available for estimating the unknown $\mathbf{\Sigma}_u$. For our simulations, the proposed methods with noise correction often converge within 3 minutes on a machine with 1.90GHz Intel i7 processor and 16GB RAM. Figure 3.8 shows the convergence plots.

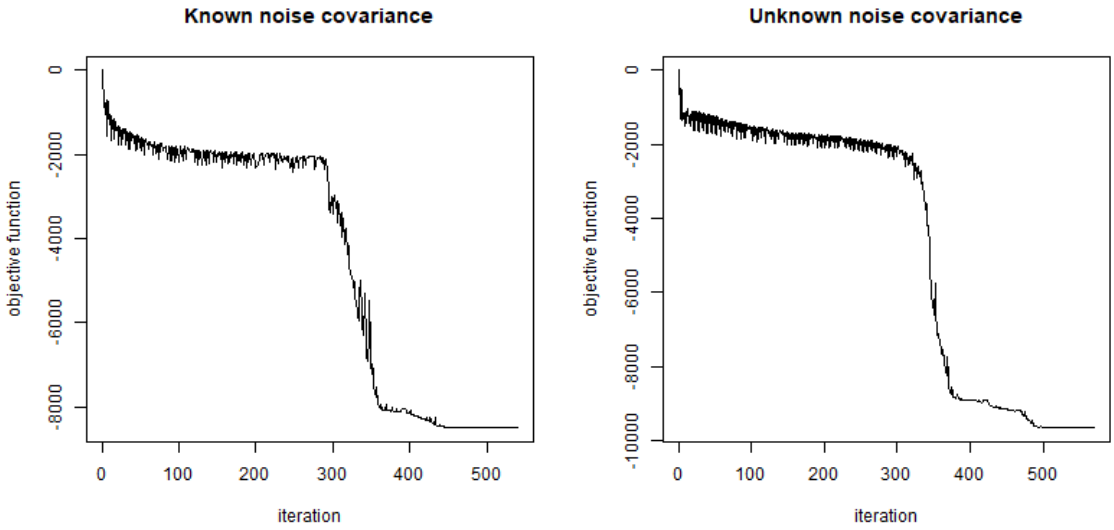

Figure 3.8: Convergence Plots for Simulation Scenarios with Known and Unknown Noise Covariance (SNR=3). The convergence appears to be slightly faster for the setting with known noise covariance, as expected.

3.5 Analysis of ADNI Data

The Alzheimer's Disease Neuroimaging Initiative (ADNI) longitudinal study is designed to develop and validate neuroimaging, clinical and genetic biomarkers in clinical trials of Alzheimer's disease (AD) therapies (Weiner and Veitch, 2015). The primary goal of the ADNI analysis in this chapter is to discover neuroimaging biomarkers in the form of localized brain regions that are significantly related to longitudinal changes in cognition for AD individuals, using magnetic resonance imaging (MRI) scans that measure the brain structure and brain volumes at the voxel level with dimensions $256 \times 256 \times 170$.

3.5.1 Data Pre-processing

Our analysis used 1.5T T1-weighted MRI volumetric scans from ADNI-1, created by the ADNI MRI Core. The downloaded data included MRI scans acquired from 192 healthy controls (NC), and 133 Alzheimer's disease (AD) individuals from screening visit (baseline), month 6 visit and month 12 visit, in addition to age, gender, and APOE status. The T1-weighted MRI images were processed with the Advanced Normalization Tools (ANTs) registration pipeline (Tustison et al., 2014). All images were registered to a population-based template image to ensure that the brain locations from different participants were normalized to the same template space. The population-based template image was created based on 52 normal control participants from ADNI 1 and shared to us from the ANTs group (Tustison et al., 2019). Among other things, the ANTs pipeline (i) uses the N4 bias correction step to correct for intensity nonuniformity (Tustison et al., 2010), which inherently normalizes the intensity across samples; and (ii) implements a symmetric diffeomorphic image registration algorithm that performs spatial normalization (Avants et al., 2008), which aligns each participant's T1 images to a template brain image so that the images across different participants can be spatially comparable. We also used the ANTs joint label fusion pipeline to produce the AAL atlas (Rolls et al., 2020) in the ADNI-specific template space. The fused atlas was then used to locate the significantly associated clusters in our downstream analysis. In Table 3.4, we provide a summary of the demographic information for the three groups in the ADNI data.

	Normal Control Arm (N=192)	Alzheimer's Disease Arm (N=133)
Baseline Age (sd)	75.89 (5.09)	74.75 (7.59)
Gender (% Female)	47.9	48.1
Education Years (sd)	16.04 (2.80)	14.73 (3.11)
APOE number		
0	138	44
1	50	58
2	4	31

Table 3.4: Demographic Information of ADNI1 Individuals

3.5.2 Analysis Outline

The outcome used for our analysis is the Mini-Mental State Exam (MMSE) score that measures cognitive abilities. We conducted our analysis separately for 9 two-dimensional axial slices each of size 128×128, which covers the hippocampus and amygdala and is our targeted area of interest (depicted in Figure 3.9). A supplementary 3-D analysis was also conducted to further support the performance of the proposed methods. Our goal is to study how the relationship between MMSE and brain structures at the voxel level change across time, by jointly analyzing the imaging data across the three longitudinal visits. Due to the fact that age, gender and APOE status did not produce significant associations with the outcome after accounting for the variability due to the brain image, and in order to boost the power to detect important regions, we chose not to adjust for these additional variables in our final scalar-on-image regression model as in Wang et al. (2014). The goals of our analysis are to identify brain regions significantly associated with MMSE, and evaluate the out of sample prediction in the presence of noisy MRI scans under the proposed approaches and the same set of competing approaches as in the simulation studies.

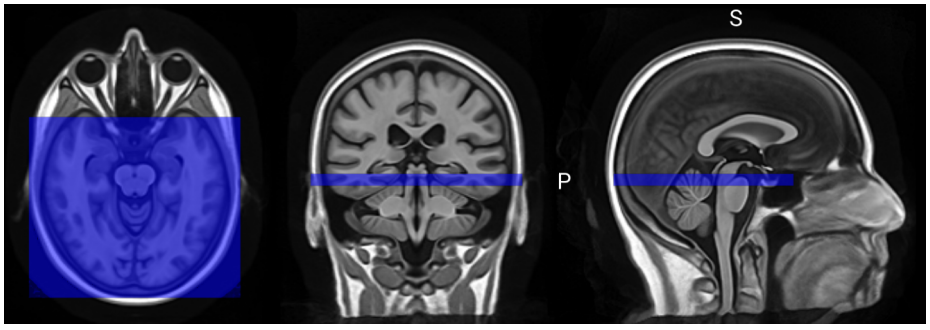

Figure 3.9: Illustration of brain region used for ADNI analysis using the axial (left), saggital (middle) and the coronal (right) slices.

For our analysis, we focus our modeling efforts on 133 AD individuals who have data at baseline, 6 months and 12 months. In addition, we used MRI scans from 192 healthy NC individuals over three longitudinal visits to obtain an estimate for the noise covariance matrix as $\hat{\mathbf{\Sigma}}_u = \frac{1}{n^*(M-1)} \sum_{i=1}^{n^*} \sum_{m=1}^{M} (\boldsymbol{z}_{mi} - \bar{\boldsymbol{z}}_{\cdot i})(\boldsymbol{z}_{mi} - \bar{\boldsymbol{z}}_{\cdot i})^T$ as in Corollary 3.2, which was subsequently used for the analysis of the AD cohort. The extrapolation of the noise covariance from the NC cohort to the AD cohort is valid under the assumption that the noise

in the MRI scans is related to scanner properties and does not depend on the disease status or other individual-specific characteristics. We note that the data on the NC individuals was not used to inform the analysis under other competing approaches, since the model parameters under these methods are specific to the analysis of AD individuals and can not be generalized to other cohorts. For model fitting and out of sample prediction under all the methods, we randomly split the 133 AD individuals into training and test groups (50-50), and consider multiple (25) such splits. The significant voxel-level associations were inferred via a two-sided t-test ($\alpha = 0.05$) with Bonferroni corrections using the estimated signals over the 25 splits. In order to eliminate clinically weak signals from the association map, all signals with absolute values less than 10^{-3} were thresholded to zero before performing the t-test.

3.5.3 Results

Table 3.5 reports the out of sample prediction, and the association maps corresponding to the significant voxels are plotted in Figures 3.10 and 3.11. Table 3.5 also reports the number of significantly associated voxels across different methods. From the results, it is clear that while the projected group lasso with noise correction is able to detect significantly associated voxels in biologically interpretable regions (see below), all other competing methods report negligible or no significant associations after multiplicity corrections that is potentially due to the *attenuation to the null phenomenon* in the presence of noisy images. Our conjecture is that multi-task learning methods without noise correction lose their ability to detect common association patterns across longitudinal visits due to non-negligible noise-to-signal ratio in the brain images. The longitudinal association maps in Figures 3.10 and 3.11 illustrate the increase in the number of associated voxels over time under the projected group lasso (also see Table 3.6). It is seen that the significant voxels in early visits are highly likely to still be significant at later visits, and these are concentrated in the hippocampus, amygdala, and parahippocampal gyrus regions that is consistent with evidence in literature. The fusiform gyrus show significant associations that supports prior evidence linking this region to visual cognition deficits and work memory tasks in AD (Yetkin et al., 2006).

	Methods	Prediction PMSE									Number of significantly associated voxels									
		s131	s130	s129	s128	s127	s126	s125	s124	s123	s131	s130	s129	s128	s127	s126	s125	s124	s123	total
Baseline	WNET	1.00	1.01	0.97	1.00	1.00	1.00	1.03	1.00	1.03	0	0	0	0	0	0	0	0	7	7
	WPCR	1.03	1.33	1.07	1.12	1.17	1.20	1.19	1.22	1.17	19	0	0	0	0	0	0	56	458	533
	p_lasso	0.98	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	46	0	0	0	0	0	0	0	0	46
	glasso	0.97	0.93	0.96	0.91	0.97	**0.95**	0.98	0.98	0.99	16	0	0	25	0	0	0	0	0	41
	gbridge	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0	0	0	0	0	0	0	0	0	0
	p_glasso	**0.92**	**0.86**	**0.89**	**0.90**	**0.94**	0.97	0.99	**0.96**	**0.97**	563	550	455	740	733	631	811	516	1212	6211
	p_gbridge	1.00	1.00	1.00	1.00	1.00	0.99	1.00	1.00	1.00	0	0	0	0	0	0	0	0	0	0
Month 6	WNET	1.00	1.00	1.00	**0.94**	0.98	1.00	1.01	0.99	1.00	0	0	0	0	0	0	0	0	68	68
	WPCR	1.35	1.09	1.22	1.50	1.42	1.43	1.42	1.51	1.48	318	432	149	308	282	531	472	348	344	3184
	p_lasso	1.00	1.00	1.00	0.98	1.00	1.00	1.00	1.00	1.00	0	0	0	0	0	0	0	0	0	0
	glasso	0.98	1.00	**0.97**	**0.94**	0.95	0.94	0.97	0.93	**0.95**	8	0	0	12	0	0	0	0	0	20
	gbridge	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0	0	0	0	0	0	0	0	0	0
	p_glasso	**0.89**	**0.97**	**0.97**	**0.94**	**0.90**	**0.92**	**0.95**	**0.91**	1.00	1257	1124	592	1169	1322	1049	1837	1279	2212	11841
	p_gbridge	1.00	1.00	1.00	1.00	1.00	0.99	1.00	1.00	1.00	0	0	0	0	0	486	0	0	0	486
Month 12	WNET	1.01	1.00	0.95	0.95	0.99	1.00	1.00	1.00	0.98	0	0	0	3	0	0	0	0	81	84
	WPCR	1.22	1.69	1.02	1.70	1.45	1.33	1.07	1.09	1.07	1231	874	176	509	799	569	700	568	695	6121
	p_lasso	1.00	1.00	0.93	1.00	1.00	0.99	1.00	0.98	1.00	0	0	0	0	0	0	0	0	11	11
	glasso	1.00	1.00	0.97	0.94	1.00	0.99	0.99	0.99	0.95	2	0	0	17	0	0	0	0	0	19
	gbridge	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0	0	0	0	0	0	0	0	0	0
	p_glasso	**0.96**	**0.92**	**0.89**	**0.87**	**0.91**	**0.97**	**0.98**	**0.90**	**0.88**	1370	1329	976	1646	1718	702	2024	1915	2617	14297
	p_gbridge	1.00	1.00	1.00	1.00	1.00	0.99	1.00	1.00	1.00	0	0	0	0	0	896	0	0	0	896

Table 3.5: Left half of the Table shows the prediction MSE for ADNI data analysis, whereas the right half shows the number of significantly associated voxels, for each of the 9 axial slices. The bolded numbers imply significantly improved PMSE compared to other methods.

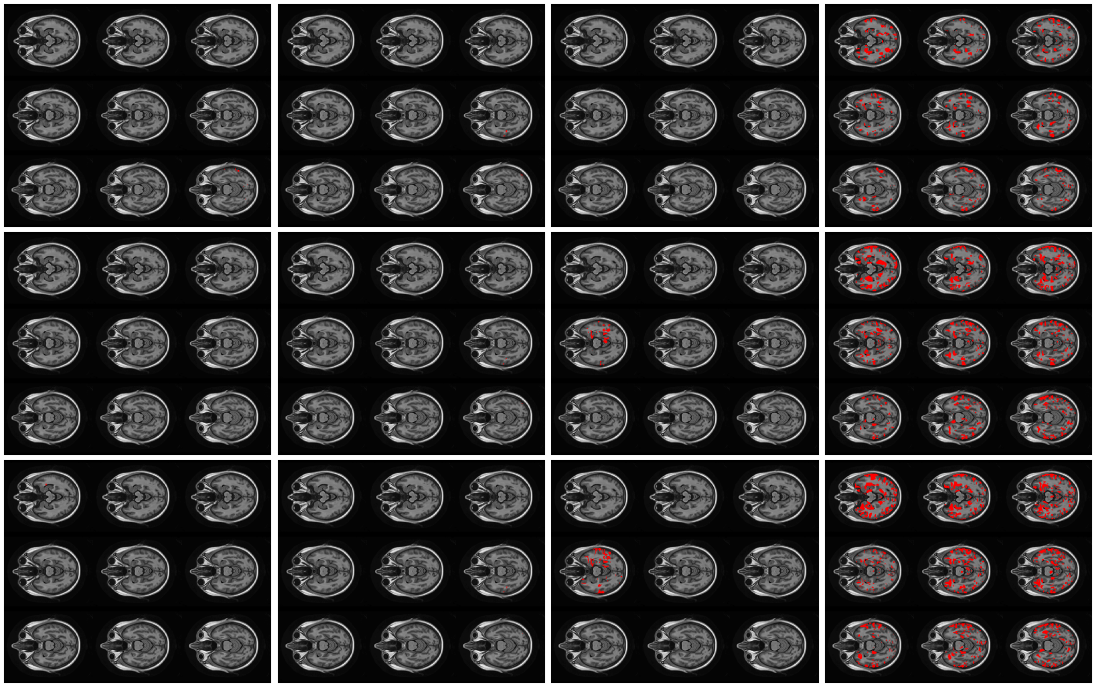

Figure 3.10: Each sub-panel corresponds to association maps of the 9 axial slices. Columns 1-4 correspond to maps under the projected Lasso, group Lasso, projected group bridge and projected group Lasso methods respectively. The top, middle, and bottom rows correspond to maps at baseline, month 6 and month 12 respectively.

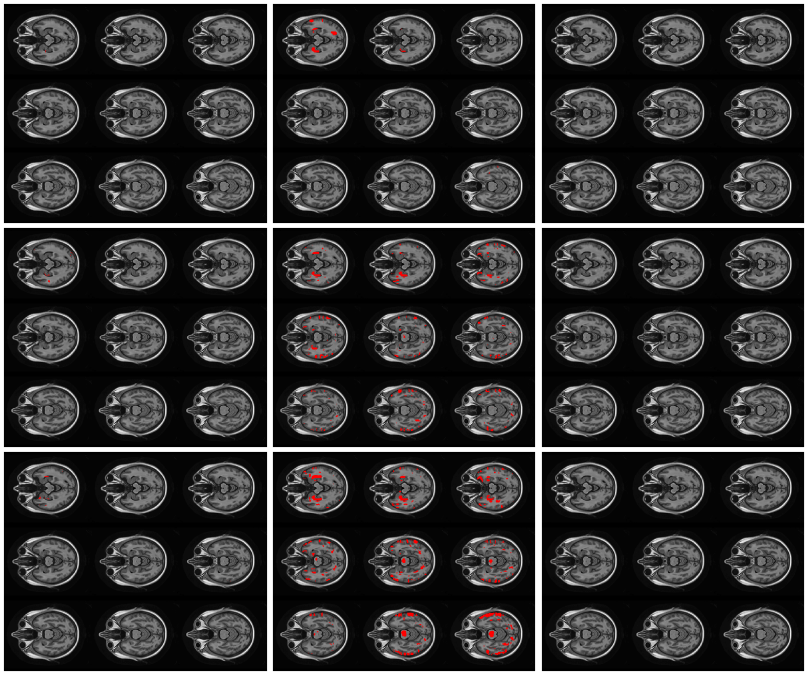

Figure 3.11: Association maps from WNET, WPCR and group bridge Methods listed in columns. The panels (each depicting 9 axial slices) from top to bottom correspond to baseline, month 6 and month 12 respectively.

	Left hemisphere			Right hemisphere		
Region name	baseline	month6	month12	baseline	month6	month12
Hippocampus	105	191	273	281	481	263
Amygdala	44	92	109	80	48	99
ParaHippocampal	6	107	382	193	210	357
Calcarine	184	169	305	48	227	377
Cuneus	4	5	4	0	2	49
Lingual	267	631	545	67	277	269
Occipital_Sup	53	87	93	0	10	15
Occipital_Mid	111	453	322	185	311	402
Occipital_Inf	27	64	29	60	105	97
Fusiform	177	388	337	315	293	352
Temporal_Pole_Sup	94	215	382	190	311	599
Temporal_Mid	739	1219	1532	241	862	979
Temporal_Pole_Mid	62	130	126	124	271	507
Temporal_Inf	708	784	864	824	1165	1084

Table 3.6: Brain Region Analysis of Associated Voxels from Projected Group Lasso Method

It is also clear that all methods except the group lasso based approaches have inferior predictive performance, which highlight the advantage of pooling information across longitudinal images under group lasso. In addition, the projected group lasso approach has significantly improved prediction performance compared to the group lasso without noise correction for the vast majority of the 2-D slices. The prediction performance for the group bridge appears inferior compared to projected group lasso, which is potentially due to the lack of optimization bound guarantees under the projected gradient descent algorithm under the group bridge. Another potential explanation is that there is a large number of homogeneous signals across longitudinal visits, which is better addressed via the group lasso penalty compared to the group bridge penalty.

we included here in Table 3.7 the calculated correlation between predicted and observed outcomes from all the methods we have compared. The results are averaged across the random splits for each 2-D slice analysis at the three visits: baseline, month 6 and month 12. Compared to the prediction MSE results provided in the main manuscript, the correlation results actually illustrate an even clearer advantage of our projected group lasso method in terms of prediction performance.

We have implemented an additional 3-D analysis for the ADNI dataset. A $32 \times 32 \times 32$ area was used for the 3-D analysis that included subcortical areas such as the hippocampus, amygdala and putamen in the right hemisphere. The same group of individuals were utilized in the 3-D analysis as in the analysis with 2-D slices, and the analysis was performed in a similar fashion to the 2-D analysis. The prediction performance is summarized in Table 3.8. Consistent with the 2-D analysis results, the best prediction performance was observed under the proposed projected group lasso method, with the group lasso approach without noise correction having the second best performance. The WNET and WPCR methods had the worst prediction performance, which is consistent with earlier results. This additional 3-D analysis provides supporting evidence of the prowess of our method for higher dimensional applications, and complements the existing 2-D analysis results presented in the main manuscript.

Finally, we also fit a convolutional neural network (CNN) with standard architecture for

the 2-D slices. However, with the limited samples available from the ADNI study and in the presence of noisy images, the CNN model demonstrated poor prediction power that was incomparable with our proposed methods (Table 3.9). The CNN architecture, with two convolutional layers, two max pooling layers and two dense layers, is displayed in Figure 3.12. The CNN models are implemented in R with the `keras` package. We note that the performance of CNN models might improve with more sophisticated architectures - however with the limited sample size, we do not expect the improvements to be substantial, and do not expect the CNN models to outperform the proposed approaches.

		s131	s130	s129	s128	s127	s126	s125	s124	s123
Baseline	WNET	0.45	0.41	0.43	0.34	0.25	0.33	0.35	0.25	0.38
	WPCR	0.34	0.30	0.32	0.27	0.30	0.27	0.26	0.29	0.32
	p_lasso	0.63	0.66	0.47	0.61	0.52	0.48	0.43	0.47	0.49
	glasso	0.48	0.48	0.44	0.51	0.47	0.48	0.36	0.47	0.46
	gbridge	0.16	0.07	0.05	0.21	0.14	0.24	0.09	0.09	0.10
	p_glasso	0.62	0.65	0.63	0.62	0.58	0.60	0.60	0.56	0.55
	p_gbridge	0.49	0.53	0.51	0.49	0.48	0.49	0.47	0.39	0.32
Month 6	WNET	0.33	0.35	0.38	0.42	0.46	0.35	0.35	0.34	0.39
	WPCR	0.37	0.41	0.38	0.35	0.36	0.37	0.33	0.30	0.31
	p_lasso	0.59	0.52	0.45	0.52	0.59	0.50	0.53	0.53	0.50
	glasso	0.53	0.53	0.53	0.55	0.55	0.59	0.48	0.58	0.57
	gbridge	0.16	0.09	0.01	0.21	0.14	0.27	0.12	0.12	0.17
	p_glasso	0.65	0.62	0.60	0.61	0.60	0.63	0.64	0.60	0.57
	p_gbridge	0.57	0.52	0.53	0.55	0.55	0.56	0.53	0.45	0.45
Month 12	WNET	0.31	0.43	0.44	0.46	0.40	0.39	0.33	0.42	0.46
	WPCR	0.28	0.33	0.38	0.37	0.36	0.30	0.29	0.34	0.41
	p_lasso	0.45	0.57	0.63	0.52	0.57	0.58	0.53	0.55	0.54
	glasso	0.58	0.55	0.56	0.61	0.55	0.62	0.53	0.59	0.60
	gbridge	0.16	0.09	0.09	0.20	0.12	0.21	0.21	0.20	0.19
	p_glasso	0.61	0.64	0.65	0.67	0.62	0.62	0.65	0.64	0.61
	p_gbridge	0.52	0.55	0.59	0.59	0.55	0.55	0.54	0.49	0.46

Table 3.7: Summary of Correlation between Observed and Predicted Outcome based on 2-D analysis of ADNI data.

3.6 Discussion

In this paper, we have proposed a novel approach for joint estimation of multiple scalar-on-image regression models involving noisy high-dimensional images. Although there is a rich literature on functional data analysis, the development for scalar-on-image regression

	Baseline	Month 6	Month 12
WNET	1.031	1.043	1.012
WPCR	2.221	1.392	1.496
p_lasso	0.987	0.983	1.020
glasso	0.955	0.928	0.938
gbridge	1.012	0.991	1.046
p_glasso	0.943	0.923	0.920
p_gbridge	1.000	1.000	1.000

Table 3.8: Prediction Results (PMSE) for ADNI 3-D Analysis

	slice 131	slice 130	slice 129	slice 128	slice 127	slice 126	slice 125	slice 124	slice 123
Baseline	1.59	1.60	1.62	1.59	1.61	1.74	1.63	1.72	1.46
Month 6	1.30	1.26	1.38	1.30	1.26	1.34	1.52	1.39	1.47
Month 12	1.09	1.04	1.10	1.07	1.10	1.10	1.01	1.08	1.05

Table 3.9: Summary of Prediction MSE of CNN Model on ADNI 2-D Slices

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 124, 124, 16)      416
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 62, 62, 16)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 58, 58, 32)        12832
_________________________________________________________________
max_pooling2d_1 (MaxPooling2D) (None, 29, 29, 32)      0
_________________________________________________________________
flatten (Flatten)            (None, 26912)             0
_________________________________________________________________
dense (Dense)                (None, 64)                1722432
_________________________________________________________________
dense_1 (Dense)              (None, 32)                2080
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 33
=================================================================
Total params: 1,737,793
Trainable params: 1,737,793
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.12: CNN Architecture Summary

methods is fairly recent, and existing methods in literature haven't addressed the question of mis-specification resulting from noisy images. Hence, the proposed methods are one of the first to address these issues via a novel M-estimation approach involving convex and non-convex group penalties that account for functional data mis-specifications. The implementation of the proposed methods are done via computationally efficient algorithms that are slightly slower than existing functional linear models that don't account for measurement error, but is still scalable to high-dimensional brain images. The approach requires one to compute the noise covariance matrix that can be estimated from a validation dataset in ADNI analysis, and is largely robust to mis-specifications in the noise covariance. While we were able to establish optimization convergence results for convex grouped penalties, the corresponding results for non-convex grouped penalties are still an open problem with very limited prior literature (Fan et al., 2014), and will be addressed in future work. The application of our proposed methods on the analysis of the ADNI T1-weighted MRI data provides a concrete example of the advantages of integrative learning via grouped penalties in multi-task learning over cross-sectional studies. Future work will include extending the proposed approach to other types of images, e.g. PET images and RAVENS maps.

3.7 Appendices

3.7.1 Discrete Wavelet Transform in 3-D

Similar to the 2-D setting, the functional predictor $X_{mi}(\mathbf{v})$ can be decomposed by a set of orthonormal wavelet bases as:

$$X_{mi}(\mathbf{v}) = \sum_{k,l,h=0}^{2^{j_0}-1} c^0_{mi,j_0,\{k,l,h\}} \phi_{j_0,\{k,l,h\}}(\mathbf{v}) + \sum_{j=j_0}^{J} \sum_{k,l,h=0}^{2^j-1} \sum_{q=1}^{7} c^q_{mi,j,\{k,l,h\}} \psi^q_{j,\{k,l,h\}}(\mathbf{v}) \quad (3.7.1)$$

with j_0 as the primary level of decomposition and J as the maximum level of decomposition. $\{\phi_{j_0,\{k,l,h\}}, k,l,h = 1,\ldots,2^{j_0}-1\}$ and $\{\psi^q_{j,\{k,l,h\}}, j = j_0,\cdots,J, k,l,h = 0,\ldots,2^j-1, q = 1,\cdots,7\}$ denote the wavelets that are pairwise orthonormal. These wavelets can also be

expressed by product of one-dimensional father and mother wavelets (Mallat, 1999) as

$$\begin{aligned}
\phi_{j_0,\{k,l,h\}}(\mathbf{v}) &= \phi_{j_0,k}(v_1)\phi_{j_0,l}(v_2)\phi_{j_0,h}(v_3),\ \psi^1_{j,\{k,l,h\}}(\mathbf{v}) = \phi_{j,k}(v_1)\phi_{j,l}(v_2)\psi_{j,h}(v_3),\\
\psi^2_{j,\{k,l,h\}}(\mathbf{v}) &= \phi_{j,k}(v_1)\psi_{j,l}(v_2)\phi_{j,h}(v_3),\ \psi^3_{j,\{k,l,h\}}(\mathbf{v}) = \phi_{j,k}(v_1)\psi_{j,l}(v_2)\psi_{j,h}(v_3),\\
\psi^4_{j,\{k,l,h\}}(\mathbf{v}) &= \psi_{j,k}(v_1)\phi_{j,l}(v_2)\phi_{j,h}(v_3),\ \psi^5_{j,\{k,l,h\}}(\mathbf{v}) = \psi_{j,k}(v_1)\phi_{j,l}(v_2)\psi_{j,h}(v_3)\\
\psi^6_{j,\{k,l,h\}}(\mathbf{v}) &= \psi_{j,k}(v_1)\psi_{j,l}(v_2)\phi_{j,h}(v_3),\ \psi^7_{j,\{k,l,h\}}(\mathbf{v}) = \psi_{j,k}(v_1)\psi_{j,l}(v_2)\psi_{j,h}(v_3),
\end{aligned}$$

where $\mathbf{v} = (v_1, v_2, v_3)$ and $\phi_{j,\cdot}(\cdot), \psi_{j,\cdot}(\cdot)$ are the one-dimensional father and mother wavelets of level j. The wavelet coefficients in (3.7.1) can be calculated by

$$c^0_{mi,j_0,\{k,l,h\}} = \langle X_{mi}, \phi_{j_0,\{k,l,h\}}\rangle \text{ and } c^q_{mi,j,\{k,l,h\}} = \langle X_{mi}, \psi^q_{j,\{k,l,h\}}\rangle.$$

In practice the 3-D functional data is only observed at discrete locations, and we pad zero values around the original functional data to increase the dimension to the nearest high power of 2 which we denote as p_0. Then the maximum level $J = \log_2(p_0) - 1$ and $p = p_0^3$, and the length of the coefficient vector $\mathbf{c}_{mi}$ in (3.7.1) is p.

We use wavelet expansions to represent the corresponding functional regression coefficients truncated at level J as:

$$\beta_m(\mathbf{v}) = \sum_{k,l,h=0}^{2^{j_0}-1} a_{m,j_0,\{k,l,h\}}\phi_{j_0,\{k,l,h\}}(\mathbf{v}) + \sum_{j=j_0}^{J}\sum_{k,l,h=0}^{2^j-1}\sum_{q=1}^{7} d^q_{m,j,\{k,l,h\}}\psi^q_{j,\{k,l,h\}}(\mathbf{v}), \qquad (3.7.2)$$

where $a_{m,j_0,\{k,l,h\}} = \langle \beta_m, \phi_{j_0,\{k,l,h\}}\rangle$ and $d^q_{m,j,\{k,l,h\}} = \langle \beta_m, \psi^q_{j,\{k,l,h\}}\rangle$. Expression (3.7.2) uses the same wavelets to decompose the functional regression coefficient as in the decomposition of the functional predictor in (3.7.1). Combining (3.7.1)-(3.7.2), one can rewrite the regression working model as:

$$\begin{aligned}
y_{mi} &= \beta_{m0} + \sum_{k,l,h=0}^{2^{j_0}-1} c^0_{mi,j_0,\{k,l,h\}}a_{m,j_0,\{k,l,h\}} + \sum_{j=j_0}^{J}\sum_{k,l,h=0}^{2^j-1}\sum_{q=1}^{7} c^q_{mi,j,\{k,l,h\}}d^q_{m,j,\{k,l,h\}} + \epsilon_{mi}\\
&= \beta_{m0} + \boldsymbol{c}^T_{mi}\boldsymbol{\eta}_m + \epsilon_{mi},\ j = 1,\ldots,p,\ i = 1,\ldots,n_m,\ m = 1,\ldots,M,
\end{aligned} \qquad (3.7.3)$$

This is in the same format as model (4) in the manuscript. Then the theoretical results would still hold for the 3-D setting.

3.7.2 KKT Condition

We first give a lemma in preparation to prove Theorem 2.1. Suppose the following Karush-Kuhn-Tucker (KKT) conditions (Ekeland and Temam, 1999) hold.

$$\tilde{\boldsymbol{C}}_I^T \boldsymbol{y} - \tilde{\boldsymbol{C}}_I^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}}) = n\lambda_n \nabla_\rho(\hat{\boldsymbol{\eta}}_I) \tag{3.7.4}$$

$$\tilde{\boldsymbol{C}}_{II}^T \boldsymbol{y} - \tilde{\boldsymbol{C}}_{II}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}}) = n\lambda_n \partial_\rho(\hat{\boldsymbol{\eta}}_{II}) \tag{3.7.5}$$

$$\lambda_{\min}(\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I) > n\lambda_n \kappa(\rho, \hat{\boldsymbol{\eta}}_I) \tag{3.7.6}$$

where $\hat{\boldsymbol{\eta}}_I = \{\hat{\eta}_{mj} | \hat{\eta}_{mj} \neq 0, \hat{\boldsymbol{\eta}}_{(j)} \neq \mathbf{0}\}$ and $\hat{\boldsymbol{\eta}}_{II} = \{\hat{\eta}_{mj} | \hat{\eta}_{mj} = 0, \hat{\boldsymbol{\eta}}_{(j)} \neq \mathbf{0}\}$, $\tilde{\boldsymbol{C}}_I$ is the submatrix of $\tilde{\boldsymbol{C}}$ formed by columns in $I = \{(m,j) | \hat{\eta}_{mj} \in \hat{\boldsymbol{\eta}}_I\}$, $\tilde{\boldsymbol{C}}_{II}$ is the submatrix of $\tilde{\boldsymbol{C}}$ formed by columns in $II = \{(m,j) | \hat{\eta}_{mj} \in \hat{\boldsymbol{\eta}}_{II}\}$, $\kappa(\rho, \hat{\boldsymbol{\eta}}_I) = \max_{\{j | \hat{\boldsymbol{\eta}}_{(j)} \neq \mathbf{0}\}} \frac{1}{4} \|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-3/2}$ is the local concavity of $\rho(\cdot)$, $\nabla_\rho(\cdot)$ and $\partial_\rho(\cdot)$ are the gradient and one subgradient of $\rho(\cdot)$, and $\lambda_{min}(A)$ denotes the minimum eigen value for the matrix A.

Lemma 3.7.1. *$\hat{\boldsymbol{\eta}} \in \mathbb{R}^{Mp}$ is a strict local maximizer of (4) if the KKT conditions (3.7.4)-(3.7.6) hold, and $\nabla_\rho(\cdot)$ and $\partial_\rho(\cdot)$ satisfy*

$$\nabla_\rho(\hat{\eta}_{mj}) = \frac{1}{2}\mathrm{sgn}(\hat{\eta}_{mj}) \|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2} \quad for \quad \hat{\eta}_{mj} \in \hat{\boldsymbol{\eta}}_I$$
$$\partial_\rho(\hat{\eta}_{mj}) \in (-\frac{1}{2}\|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2}, \frac{1}{2}\|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2}) \quad for \quad \hat{\eta}_{mj} \in \hat{\boldsymbol{\eta}}_{II}.$$

On the other hand, if $\hat{\boldsymbol{\eta}}$ is a local maximizer of (4), then it must satisfy (3.7.4) - (3.7.6) with $>$ replaced by $\geq$ in (3.7.6) and $\partial_\rho(\hat{\eta}_{mj}) \in [-\frac{1}{2}\|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2}, \frac{1}{2}\|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2}]$.

Proof. We follow the proof in Fan and Lv (2011) and Li et al. (2014). We will first derive the necessary condition. In view of (4), up to an affine transformation, the log-likelihood

can be expressed in matrix form as

$$\ell(\boldsymbol{\eta}) = \boldsymbol{y}^T\tilde{\boldsymbol{C}}\boldsymbol{\eta} - \frac{1}{2}\boldsymbol{\eta}^T\tilde{\boldsymbol{C}}^T\tilde{\boldsymbol{C}}\boldsymbol{\eta}$$

Then we have

$$\nabla\ell(\boldsymbol{\eta}) = \tilde{\boldsymbol{C}}^T\boldsymbol{y} - \tilde{\boldsymbol{C}}^T\tilde{\boldsymbol{C}}\boldsymbol{\eta}, \quad \nabla^2\ell(\boldsymbol{\eta}) = -\tilde{\boldsymbol{C}}^T\tilde{\boldsymbol{C}}$$

If $\hat{\boldsymbol{\eta}}$ is a local maximizer of the penalized problem (4), then by the classical Karush-Kuhn-Tucker (KKT) condition, there exists gradient $\nabla_\rho(\hat{\boldsymbol{\eta}})$ and sub-gradient $\partial_\rho(\hat{\boldsymbol{\eta}})$ such that

$$\begin{aligned}
\tilde{\boldsymbol{C}}_I^T\boldsymbol{y} - \tilde{\boldsymbol{C}}_I^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}}) - n\lambda_n\nabla_\rho(\hat{\boldsymbol{\eta}}_I) &= \boldsymbol{0} \\
\tilde{\boldsymbol{C}}_{II}^T\boldsymbol{y} - \tilde{\boldsymbol{C}}_{II}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}}) - n\lambda_n\partial_\rho(\hat{\boldsymbol{\eta}}_{II}) &= \boldsymbol{0} \\
\tilde{\boldsymbol{C}}_{III}^T\boldsymbol{y} - \tilde{\boldsymbol{C}}_{III}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}}) - n\lambda_n\partial_\rho(\hat{\boldsymbol{\eta}}_{III}) &= \boldsymbol{0}
\end{aligned}$$

where

$$\begin{aligned}
&\nabla_\rho(\hat{\eta}_{mj}) = \frac{1}{2}\mathrm{sgn}(\hat{\eta}_{mj})\|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2} \quad for \quad \hat{\eta}_{mj} \in \hat{\boldsymbol{\eta}}_I \\
&\partial_\rho(\hat{\eta}_{mj}) \in [-\frac{1}{2}\|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2}, \frac{1}{2}\|\hat{\boldsymbol{\eta}}_{(j)}\|_1^{-1/2}] \quad for \quad \hat{\eta}_{mj} \in \hat{\boldsymbol{\eta}}_{II} \\
&\partial_\rho(\hat{\eta}_{mj}) \in (-\infty, +\infty) \quad for \quad \hat{\eta}_{mj} \in \hat{\boldsymbol{\eta}}_{III}
\end{aligned}$$

Here $\hat{\boldsymbol{\eta}}_I$ and $\hat{\boldsymbol{\eta}}_{II}$ are defined in Lemma 1 and $\hat{\boldsymbol{\eta}}_{III} = \{\hat{\eta}_{mj}|\hat{\eta}_{mj} = 0, \hat{\boldsymbol{\eta}}_{(j)} = \boldsymbol{0}\}$. It is easy to see that the third condition always holds. Also we note that $\hat{\boldsymbol{\eta}}_I$ is a local maximizer of (4) constrained on the subspace $\mathcal{S}_1 = \{\boldsymbol{\eta} \in \mathbb{R}^{Mp} : \boldsymbol{\eta}_{II\cup III} = \boldsymbol{0}\}$ which has dimension card(I). Thus it follows from the second order condition that

$$\lambda_{\min}(\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I) \geq n\lambda_n\kappa(\rho, \hat{\boldsymbol{\eta}}_I)$$

where $\kappa(\rho, \hat{\boldsymbol{\eta}}_I)$ is as defined in Lemma 1.

Next we will show the sufficient condition. We denote the penalized likelihood in (4) as $Q_n(\boldsymbol{\eta})$. Firstly we constrain the penalized problem on the subspace $\mathcal{S}_1$. It follows from conditions (3.7.4) and (3.7.6) that $\hat{\boldsymbol{\eta}}$ is the unique maximizer of $Q_n(\boldsymbol{\eta})$ in a neighborhood $\mathcal{N}_1 \subset \mathcal{S}_1$ centered at $\hat{\boldsymbol{\eta}}$.

Then we show that there exists a L_1 neighborhood $\mathcal{N}_2$ in the subspace $\mathcal{S}_2$ of dimension card$(I \cup II)$ such that $\mathcal{S}_1 \subset \mathcal{S}_2 \subset \mathbb{R}^{Mp}$, and that $\hat{\boldsymbol{\eta}}$ is the unique maximizer of $Q_n(\boldsymbol{\eta})$ constrained on $\mathcal{S}_2$. To show this, we take a sufficiently small ball $\mathcal{N}_2$ in $\mathcal{S}_2$ centered at $\hat{\boldsymbol{\eta}}$ such that $\mathcal{N}_2 \cap \mathcal{S}_1 \subset \mathcal{N}_1$. We then need to show that $Q_n(\hat{\boldsymbol{\eta}}) > Q_n(\boldsymbol{\phi}_1)$ for any $\boldsymbol{\phi}_1 \in \mathcal{N}_2 \backslash \mathcal{N}_1$. Let $\boldsymbol{\phi}_2$ be the projection of $\boldsymbol{\phi}_1$ onto the subspace $\mathcal{S}_1$. Then we have $\boldsymbol{\phi}_2 \in \mathcal{N}_1$ which implies that $Q_n(\hat{\boldsymbol{\eta}}) > Q_n(\boldsymbol{\phi}_2)$ if $\boldsymbol{\phi}_2 \neq \hat{\boldsymbol{\eta}}$. Thus it suffices to show that $Q_n(\boldsymbol{\phi}_2) > Q_n(\boldsymbol{\phi}_1)$.

By the mean-value theorem, we have

$$Q_n(\boldsymbol{\phi}_1) - Q_n(\boldsymbol{\phi}_2) = [\nabla Q_n(\boldsymbol{\phi}_0)]^T(\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2)$$

where $\boldsymbol{\phi}_0$ lies on the line segment joining $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$. Note that the coordinates of $\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2$ are zero for indices in I and $\text{sgn}(\phi_{0,mj}) = \text{sgn}(\phi_{1,mj})$ for indices in II where $\phi_{0,mj}$ and $\phi_{1,mj}$ are the (m, j)th coordinate of $\boldsymbol{\phi}_0$ and $\boldsymbol{\phi}_1$ respectively. Thus we have

$$\begin{aligned}
[\nabla Q_n(\boldsymbol{\phi}_0)]^T(\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2) &= [\tilde{\boldsymbol{C}}_{II}^T(\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\phi}_0)]^T\boldsymbol{\phi}_{1,II} - n\lambda_n \sum_{(m,j)\in II} \nabla_\rho(\phi_{0,mj})\phi_{1,mj} \\
&= [\tilde{\boldsymbol{C}}_{II}^T(\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\phi}_0)]^T\boldsymbol{\phi}_{1,II} - n\lambda_n \sum_{(m,j)\in II} \nabla_\rho(|\phi_{0,mj}|)|\phi_{1,mj}| \\
&= [\tilde{\boldsymbol{C}}_{II}^T(\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\phi}_0)]^T\boldsymbol{\phi}_{1,II} - n\lambda_n \sum_{(m,j)\in II} 1/2(\sum_{m=1}^{M} |\phi_{0,mj}|)^{-1/2}|\phi_{1,mj}|
\end{aligned} \tag{3.7.7}$$

By continuity of $\nabla_\rho(\cdot)$ and (3.7.5), there exists $\delta > 0$ such that for any $\boldsymbol{\phi}$ in an L_1-ball in $\mathcal{S}_2$ centered at $\hat{\boldsymbol{\eta}}$ with radius δ, we have

$$(n\lambda_n)^{-1}\|\tilde{\boldsymbol{C}}_{II}^T(\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\phi})\|_\infty < 1/2(\|\hat{\boldsymbol{\eta}}_{(j)}\|_1 + \delta)^{-1/2}$$

Let $\mathcal{N}_2$ be that ball. Then we also have

$$\sum_{m=1}^{M} |\phi_{0,mj}| \leq \sum_{m=1}^{M} |\phi_{0,mj} - \hat{\eta}_{mj}| + \sum_{m=1}^{M} |\hat{\eta}_{mj}| \leq \delta + \|\hat{\boldsymbol{\eta}}_{(j)}\|_1$$

Thus we know the term (3.7.7) is strictly less than

$$n\lambda_n[1/2(\|\hat{\boldsymbol{\eta}}_{(j)}\|_1+\delta)^{-1/2}]\|\boldsymbol{\phi}_{1,II}\|_1 - n\lambda_n[1/2(\|\hat{\boldsymbol{\eta}}_{(j)}\|_1+\delta)^{-1/2}]\|\boldsymbol{\phi}_{1,II}\|_1 = 0$$

This concludes that $Q_n(\boldsymbol{\phi}_1) < Q_n(\boldsymbol{\phi}_2)$ and shows that there exists a neighborhood $\mathcal{N}_2$ centered at $\hat{\boldsymbol{\eta}}$ in the subspace $\mathcal{S}_2$ such that $\hat{\boldsymbol{\eta}}$ is the unique maximizer of $Q_n(\boldsymbol{\eta})$ constrained on $\mathcal{S}_2$. Noting that the third KKT condition always hold, it is easy to see that $\hat{\boldsymbol{\eta}}$ is indeed a local maximizer in $\mathbb{R}^{Mp}$, which completes the proof. □

3.7.3 Proof of Theorem 3.2.1

Proof. Let $\boldsymbol{\xi} = (\xi_{11}, \xi_{12}, \cdots, \xi_{Mp})^T = \tilde{\boldsymbol{C}}^T\boldsymbol{y} - \tilde{\boldsymbol{C}}^T(\tilde{\boldsymbol{C}}\boldsymbol{\eta}^{\mathbf{0}})$. Consider events

$$\mathcal{E}_1 = \left\{\|\boldsymbol{\xi}_I\|_\infty \le \sigma\sqrt{2n\log n}\right\} \quad \text{and}$$
$$\mathcal{E}_2 = \left\{\|\boldsymbol{\xi}_{II\cup III}\|_\infty \le \sigma n^{1-\alpha_p}\sqrt{2\log n}\right\}$$

where $\sigma = \max\{\sigma_1, \cdots, \sigma_M\}$.

Based on the true model, we have that $\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\eta}^0 = \boldsymbol{\epsilon} + \boldsymbol{A}$ where $\boldsymbol{A} = (\boldsymbol{A}_1^T, \cdots, \boldsymbol{A}_M^T)^T$, $\boldsymbol{A}_m = (\boldsymbol{a}_{m1}, \cdots, \boldsymbol{a}_{mn})^T = (\int X_{m1}(\mathbf{v})\boldsymbol{e}_m^0(\mathbf{v})\mathrm{d}\mathbf{v}, \cdots, \int X_{mn}(\mathbf{v})\boldsymbol{e}_m^0(\mathbf{v})\mathrm{d}\mathbf{v})^{\mathrm{T}}$ for $m = 1, \cdots, M$, and that $\boldsymbol{\epsilon} = (\epsilon_{11}, \cdots, \epsilon_{Mn})^T, \epsilon_{mi} \sim \mathrm{N}(0, \sigma_m^2)$. These together imply that $\xi_{mj} = \tilde{\boldsymbol{C}}_{mj}^T(\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\eta}^0) \sim \mathrm{N}(\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m, \sigma_{mj}^2)$ where $\sigma_{mj}^2 \le \sigma^2\|\tilde{\boldsymbol{C}}_{mj}\|_2^2$. We assume after standardization that $\|\tilde{\boldsymbol{C}}_{mj}\|_2 = \sqrt{n}$, then it follows from Bonferroni's inequality and tail probability of normal

distribution that

$$
\begin{aligned}
P(\mathcal{E}_1 \cap \mathcal{E}_2) &= 1 - P(\mathcal{E}_1^c \cup \mathcal{E}_2^c) \\
&\geq 1 - P(\mathcal{E}_1^c) - P(\mathcal{E}_2^c) \\
&= 1 - P\Big(\cup_{(m,j)\in I} |\xi_{mj}| \geq \sigma\sqrt{2n\log n}\Big) \\
&\quad - P\Big(\cup_{(m,j)\in II\cup III} |\xi_{mj}| \geq \sigma n^{1-\alpha_p}\sqrt{2\log n}\Big) \\
&\geq 1 - \sum_{(m,j)\in I} P\Big(|\xi_{mj}| \geq \sigma\sqrt{2n\log n}\Big) \\
&\quad - \sum_{(m,j)\in II\cup III} P\Big(|\xi_{mj}| \geq \sigma n^{1-\alpha_p}\sqrt{2\log n}\Big) \\
&\geq 1 - \sum_{(m,j)\in I} P\Big(|\xi_{mj} - \tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m| \geq \sigma\sqrt{2n\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\Big) \\
&\quad - \sum_{(m,j)\in II\cup III} P\Big(|\xi_{mj} - \tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m| \geq \sigma n^{1-\alpha_p}\sqrt{2\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\Big) \\
&= 1 - \sum_{(m,j)\in I} 2\exp\Big\{ - \frac{\big(\sigma\sqrt{2n\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\big)^2}{2\sigma_{mj}^2}\Big\} \\
&\quad - \sum_{(m,j)\in II\cup III} 2\exp\Big\{ - \frac{\big(\sigma n^{1-\alpha_p}\sqrt{2\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\big)^2}{2\sigma_{mj}^2}\Big\} \\
&\geq 1 - \sum_{(m,j)\in I} 2\exp\Big\{ - \frac{\big(\sigma\sqrt{2n\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\big)^2}{2\sigma^2\|\tilde{\boldsymbol{C}}_{mj}\|_2^2}\Big\} \\
&\quad - \sum_{(m,j)\in II\cup III} 2\exp\Big\{ - \frac{\big(\sigma n^{1-\alpha_p}\sqrt{2\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\big)^2}{2\sigma^2\|\tilde{\boldsymbol{C}}_{mj}\|_2^2}\Big\}
\end{aligned}
$$

We know that $|\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m| \leq \|\tilde{\boldsymbol{C}}_{mj}\|_2\|\boldsymbol{A}_m\|_2 = \sqrt{n}O(\sqrt{n}p^{-1/2}) = O(np^{-1/2})$. Thus the term $\exp\Big\{ - \frac{\big(\sigma\sqrt{2n\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\big)^2}{2\sigma^2\|\tilde{\boldsymbol{C}}_{mj}\|_2^2}\Big\} \approx \exp\Big\{ - \Big[\log n + (\frac{C'^2}{2\sigma^2})(\frac{n}{p}) - \frac{\sqrt{2}C'}{\sigma}\sqrt{\frac{n\log n}{p}}\Big]\Big\}$. From our assumptions on the order of n and p, the latter two terms will be negligible compared to the first term of $\log n$ with sufficiently large n. Then this quantity would be approximated by n^{-1}. Similarly, $\exp\Big\{ - \frac{\big(\sigma n^{1-\alpha_p}\sqrt{2\log n} - |\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{A}_m|\big)^2}{2\sigma^2\|\tilde{\boldsymbol{C}}_{mj}\|_2^2}\Big\}$ can be approximated by $e^{-n^{1-2\alpha_p}\log n}$ with sufficiently large n. Combining together all these results, we have $P(\mathcal{E}_1 \cap \mathcal{E}_2) \geq 1 - 2[sn^{-1} + (Mp - s)e^{-n^{1-2\alpha_p}\log n}]$ with sufficiently large n.

Under the event $\mathcal{E}_1 \cap \mathcal{E}_2$, we will show that there exists a solution to (4) that achieve the

weak oracle properties in (a) and (b).

Step 1: We prove that with sufficiently large n, there exists a solution to (3.7.4) in the L_∞ ball

$$\mathcal{N} = \{\boldsymbol{\delta} \in \mathbb{R}^s : \|\boldsymbol{\delta} - \boldsymbol{\eta}_I^0\|_\infty = n^{-\gamma} \log n\}$$

Let $\boldsymbol{\theta} = n\lambda_n \nabla_\rho(\boldsymbol{\delta})$. Then for any $(m, j) \in I$, we have

$$\begin{aligned}
|\theta_{mj}| &= n\lambda_n 2^{-1} \mathrm{sgn}(\delta_{mj}) \|\boldsymbol{\delta}_{(j)}\|_1^{-1/2} \\
&\leq 2^{-1} n\lambda_n / (\textstyle\sum_{m=1}^M |\delta_{mj}|)^{1/2} \\
&\leq 2^{-1} n\lambda_n / (\textstyle\sum_{m=1}^M |\eta_{mj}^0| - \sum_{m=1}^M |\delta_{mj} - \eta_{mj}^0|)^{1/2} \\
&\leq 2^{-1} n\lambda_n / (\textstyle\sum_{m=1}^M |\eta_{mj}^0| - \sum_{m=1}^M 1/2|\eta_{mj}^0|)^{1/2} \\
&\leq n\lambda_n / (\sqrt{2} l) \\
&\leq n\lambda_n (2Md)^{-1/2}
\end{aligned} \tag{3.7.8}$$

as we konw under condition (C2) for sufficiently large n, $1/2|\eta_{mj}^0| \geq d \geq n^{-\gamma} \log n \geq |\delta_{mj} - \eta_{mj}^0|$. Thus it holds that

$$\|\boldsymbol{\theta}\|_\infty \leq n\lambda_n (2Md)^{-1/2}$$

Then given that event $\mathcal{E}_1$ holds, we have

$$\|\boldsymbol{\xi}_I - \boldsymbol{\theta}\|_\infty \leq \|\boldsymbol{\xi}_I\|_\infty + \|\boldsymbol{\theta}\|_\infty \leq \sigma\sqrt{2n \log n} + n\lambda_n (2Md)^{-1/2} \tag{3.7.9}$$

Now define

$$\boldsymbol{\Psi}(\boldsymbol{\delta}) = (\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I)^{-1} \Big[\tilde{\boldsymbol{C}}_I^T (\tilde{\boldsymbol{C}}_I \boldsymbol{\delta} - \tilde{\boldsymbol{C}}_I \boldsymbol{\eta}_I^0) - (\boldsymbol{\xi}_I - \boldsymbol{\theta})\Big]$$

We note that $\boldsymbol{\Psi}(\boldsymbol{\delta}) = 0$ is equivalent to (3.7.4). We need to show that former has a solution in the L_∞ ball $\mathcal{N}$. Let $\boldsymbol{u} = -(\tilde{\boldsymbol{C}}_I^T \tilde{\boldsymbol{C}}_I)^{-1}(\boldsymbol{\xi}_I - \boldsymbol{\theta})$, then $\boldsymbol{\Psi}(\boldsymbol{\delta}) = (\boldsymbol{\delta} - \boldsymbol{\eta}_I^0) + \boldsymbol{u}$. It follows from

equation (3.7.9), condition (C3) and the requirement $\lambda_n b_s = o(n^{-\alpha_d/2-\gamma}\log n)$ that

$$\begin{aligned}\|\boldsymbol{u}\|_\infty &\le \|(\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I)^{-1}\|_\infty\|\boldsymbol{\xi}_I-\boldsymbol{\theta}\|_\infty \\ &\le \mathcal{O}(b_s n^{-1})\Big[\sigma\sqrt{2n\log n}+n\lambda_n(2Md)^{-1/2}\Big] \\ &= \mathcal{O}(b_s n^{-1/2}\sqrt{\log n}+b_s\lambda_n d^{-1/2}) \\ &= o(n^{-\gamma}\log n)\end{aligned} \tag{3.7.10}$$

Then for sufficiently large n, if $(\boldsymbol{\delta}-\boldsymbol{\eta}_I^0)_{mj} = n^{-\gamma}\log n$, we have

$$\boldsymbol{\Psi}_{mj}(\boldsymbol{\delta}) \ge n^{-\gamma}\log n - \|\boldsymbol{u}\|_\infty \ge 0$$

and if $(\boldsymbol{\delta}-\boldsymbol{\eta}_I^0)_{mj} = -n^{-\gamma}\log n$, we have

$$\boldsymbol{\Psi}_{mj}(\boldsymbol{\delta}) \le -n^{-\gamma}\log n + \|\boldsymbol{u}\|_\infty \le 0$$

Thus by the continuity of the vector-valued function $\boldsymbol{\Psi}(\boldsymbol{\delta})$ and applying Miranda's existence theorem (Vrahatis, 1989), we know that $\boldsymbol{\Psi}(\boldsymbol{\delta})=0$ has a solution $\hat{\boldsymbol{\eta}}_I$ in $\mathcal{N}$. Thus we have shown that (3.7.4) indeed has a solution in the L_∞ ball $\mathcal{N}$.

Step 2: We verify that $\hat{\boldsymbol{\eta}} = (\hat{\boldsymbol{\eta}}_I^T, \boldsymbol{0}^T)^T \in \mathbb{R}^{Mp}$ satisfies (3.7.5) for the choice of λ_n. Actually (3.7.5) requires for any $(m,j)\in II$ that

$$|\tilde{\boldsymbol{C}}_{mj}^T\boldsymbol{y} - \tilde{\boldsymbol{C}}_{mj}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}}^0)| < 1/2n\lambda_n\Big(\sum_{m':(m',j)\in I}|\hat{\eta}_{m'j}|\Big)^{-1/2}$$

We know from condition (C2) that for sufficiently large n, $d \ge n^{-\gamma}\log n$. Thus by definition, we have $\operatorname{sgn}(\hat{\eta}_{m'j}) = \operatorname{sgn}(\eta^0_{m'j})$ for $(m',j)\in I$ and in addition

$$\begin{aligned}\sum_{m':(m',j)\in I}|\hat{\eta}_{m'j}| &\le \sum_{m':(m',j)\in I}|\hat{\eta}_{m'j}-\eta^0_{m'j}| + |\eta^0_{m'j}| \\ &\le 2\sum_{m':(m',j)\in I}|\eta^0_{m'j}| \\ &\le 2L^2\end{aligned}$$

Thus it suffices to show that

$$\|\tilde{\boldsymbol{C}}_{II}^T\boldsymbol{y} - \tilde{\boldsymbol{C}}_{II}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}})\|_\infty < n\lambda_n/(2\sqrt{2}L)$$

Note that

$$\tilde{\boldsymbol{C}}_{II}^T\boldsymbol{y} - \tilde{\boldsymbol{C}}_{II}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}}) = \tilde{\boldsymbol{C}}_{II}^T(\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\eta}^0) - \tilde{\boldsymbol{C}}_{II}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}} - \tilde{\boldsymbol{C}}\boldsymbol{\eta}^0)$$

Given that event $\mathcal{E}_2$ holds and $\alpha_\lambda < \alpha_p$, we have

$$(n\lambda_n)^{-1}\|\tilde{\boldsymbol{C}}_{II}^T(\boldsymbol{y} - \tilde{\boldsymbol{C}}\boldsymbol{\eta}^0)\|_\infty \leq (n\lambda_n)^{-1}\sigma n^{1-\alpha_p}\sqrt{2\log n} = (n\lambda_n)^{-1}O(n^{1-\alpha_p}\sqrt{\log n}) = o(1) \tag{3.7.11}$$

And since $\hat{\boldsymbol{\eta}}_I$ solves $\boldsymbol{\Psi}(\boldsymbol{\delta}) = 0$, we have $\hat{\boldsymbol{\eta}}_I - \boldsymbol{\eta}_I^0 = -\boldsymbol{u} = (\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I)^{-1}(\boldsymbol{\xi}_I - \boldsymbol{\theta})$. We also have equation (3.7.8) implying that $\|\boldsymbol{\theta}\|_\infty \leq n\lambda_n/(\sqrt{2}l)$. Then by condition (C4) for sufficiently large n, and given $\alpha_\lambda < \alpha_p < 1/2$, we have

$$\begin{aligned}
(n\lambda_n)^{-1}\|\tilde{\boldsymbol{C}}_{II}^T(\tilde{\boldsymbol{C}}\hat{\boldsymbol{\eta}} - \tilde{\boldsymbol{C}}\boldsymbol{\eta}^0)\|_\infty &= (n\lambda_n)^{-1}\|\tilde{\boldsymbol{C}}_{II}^T\tilde{\boldsymbol{C}}_I(\hat{\boldsymbol{\eta}}_I - \boldsymbol{\eta}_I^0)\|_\infty \\
&= (n\lambda_n)^{-1}\|\tilde{\boldsymbol{C}}_{II}^T\tilde{\boldsymbol{C}}_I(\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I)^{-1}(\boldsymbol{\xi}_I - \boldsymbol{\theta})\|_\infty \\
&\leq (n\lambda_n)^{-1}\|\tilde{\boldsymbol{C}}_{II}^T\tilde{\boldsymbol{C}}_I(\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I)^{-1}\|_\infty\|\boldsymbol{\xi}_I\|_\infty \\
&\quad + (n\lambda_n)^{-1}\|\tilde{\boldsymbol{C}}_{II}^T\tilde{\boldsymbol{C}}_I(\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I)^{-1}\|_\infty\|\boldsymbol{\theta}\|_\infty \\
&\leq \|\tilde{\boldsymbol{C}}_{II}^T\tilde{\boldsymbol{C}}_I(\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I)^{-1}\|_\infty[\sigma n^{-1/2}\lambda_n^{-1}\sqrt{2\log n} + 1/(\sqrt{2}l)] \\
&\leq 1/(2\sqrt{2}L)
\end{aligned} \tag{3.7.12}$$

which completes the proof of step 2.

And finally, it is easy to see that (3.7.6) holds for sufficiently large n given that $\lambda_n\kappa_0 = o(\tau_0)$ and $\tau_0 = \lambda_{\min}(n^{-1}\tilde{\boldsymbol{C}}_I^T\tilde{\boldsymbol{C}}_I)$. Thus we have shown that under the event $\mathcal{E}_1 \cap \mathcal{E}_2$, $\hat{\boldsymbol{\eta}}$ is a local maximizer of (4) such that $\|\hat{\boldsymbol{\eta}}_I - \boldsymbol{\eta}_I\|_\infty \leq n^{-\gamma}\log n$ and $\hat{\boldsymbol{\eta}}_{II\cup III} = \boldsymbol{0}$. □

3.7.4 Proof of Corollary 3.2.1

Proof. The local maximizer of (4) is obtained with the standardized design matrix $\tilde{\boldsymbol{C}}$. We first connect it with the wavelet coefficients of the regression using design matrix in its original scale. We define $\boldsymbol{D}$ to be a diagonal matrix with $\|\boldsymbol{c}_{mj}\|_2$ as the mj-th diagonal element and $\boldsymbol{D}_m$ to be the corresponding m-th block of $\boldsymbol{D}$.

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{\eta} + \boldsymbol{A} + \boldsymbol{\epsilon} = \boldsymbol{C}(\sqrt{Mn}\boldsymbol{D}^{-1})(\boldsymbol{D}/\sqrt{Mn})\boldsymbol{\eta} + \boldsymbol{A} + \boldsymbol{\epsilon} = \tilde{\boldsymbol{C}}\tilde{\boldsymbol{\eta}} + \boldsymbol{A} + \boldsymbol{\epsilon}$$

where $\tilde{\boldsymbol{C}} = \boldsymbol{C}(\sqrt{Mn}\boldsymbol{D}^{-1})$. Correspondingly, $\tilde{\boldsymbol{\eta}} = (\boldsymbol{D}/\sqrt{Mn})\boldsymbol{\eta}$ is the vector of wavelet coefficients satisfying the finite error bounds in Theorem 2.1. Thus it is easy to see that $\|\hat{\tilde{\boldsymbol{\eta}}} - \tilde{\boldsymbol{\eta}}^0\|_1 = \|\hat{\tilde{\boldsymbol{\eta}}}_I - \tilde{\boldsymbol{\eta}}^0_I\|_1 + \|\hat{\tilde{\boldsymbol{\eta}}}_{II\cup III} - \tilde{\boldsymbol{\eta}}^0_{II\cup III}\|_1 = \|\hat{\tilde{\boldsymbol{\eta}}}_I - \tilde{\boldsymbol{\eta}}^0_I\|_1 + 0 \le sn^{-\gamma}\log n$. We define $B^+_m(\mathbf{v}) = \mathbf{e}_m \otimes B(\mathbf{v})$ where $\mathbf{e}_m = (0,\cdots,0,1,0,\cdots,0)^T$ is the standard basis vector of length M where the m-th location has value 1 and $\otimes$ denotes the Kronecker product. Then

$$\begin{aligned}
|\hat{\boldsymbol{\beta}}_m(\mathbf{v}) - \boldsymbol{\beta}^0_m(\mathbf{v})| &= |B^+_m(\mathbf{v})^T\hat{\boldsymbol{\eta}} - B^+_m(\mathbf{v})^T\boldsymbol{\eta}^0 - \boldsymbol{e}^0_m(\mathbf{v})| \\
&= |B^+_m(\mathbf{v})^T\sqrt{Mn}\boldsymbol{D}^{-1}(\hat{\tilde{\boldsymbol{\eta}}} - \tilde{\boldsymbol{\eta}}^0) - \boldsymbol{e}^0_m(\mathbf{v})| \\
&\le \|B^+_m(\mathbf{v})^T\sqrt{Mn}\boldsymbol{D}^{-1}\|_\infty\|\hat{\tilde{\boldsymbol{\eta}}} - \tilde{\boldsymbol{\eta}}^0\|_1 + |\boldsymbol{e}^0_m(\mathbf{v})| \\
&\le \tau_m(\mathbf{v})sn^{-\gamma}\log n + O(p^{-1/2})
\end{aligned}$$

since

$$\begin{aligned}
\|B^+_m(\mathbf{v})^T\sqrt{Mn}\boldsymbol{D}^{-1}\|_\infty &= \|\sqrt{Mn}(\mathbf{e}_m \otimes B(\mathbf{v}))^T\boldsymbol{D}^{-1}\|_\infty \\
&= \sqrt{Mn}\|B(\mathbf{v})^T\boldsymbol{D}_m^{-1}\|_\infty \\
&= \max_{j\in\{1,\cdots,p\}} \frac{|b_j(\mathbf{v})|}{\sqrt{\frac{1}{Mn}\|\boldsymbol{c}_{mj}\|_2^2}} := \tau_m(\mathbf{v})
\end{aligned}$$

For the mean functions, we have

$$\begin{aligned}\left|\int \boldsymbol{X}_{mi}(\mathbf{v})\hat{\boldsymbol{\beta}}_m(\mathbf{v})\mathrm{d}\mathbf{v} - \int \boldsymbol{X}_{mi}(\mathbf{v})\boldsymbol{\beta}_m^0(\mathbf{v})\mathrm{d}\mathbf{v}\right| &= |\boldsymbol{c}_{mi}\hat{\boldsymbol{\eta}} - \boldsymbol{c}_{mi}\boldsymbol{\eta}^0 - \boldsymbol{a}_{mi}| \\ &= |\boldsymbol{c}_{mi}\sqrt{Mn}\boldsymbol{D}^{-1}(\hat{\tilde{\boldsymbol{\eta}}} - \tilde{\boldsymbol{\eta}}^0) - \boldsymbol{a}_{mi}| \\ &\leq \sqrt{Mn}\|\boldsymbol{c}_{mi}\boldsymbol{D}^{-1}\|_\infty\|\hat{\tilde{\boldsymbol{\eta}}} - \tilde{\boldsymbol{\eta}}^0\|_1 + |\boldsymbol{a}_{mi}| \\ &\leq \iota_m s n^{-\gamma}\log n + O(p^{-1/2})\end{aligned}$$

where $\iota_m = \max_{i\in\{1,\cdots,n\},j\in\{1,\cdots,p\}} \frac{|c_{mi,j}|}{\sqrt{\frac{1}{Mn}\|\boldsymbol{c}_{mj}\|_2^2}}$. □

3.7.5 Proof of Lemma 3.3.1

Proof. We have for the outcome $\boldsymbol{y}_m = \boldsymbol{C}_m\boldsymbol{\eta}_m^0 + \boldsymbol{A}_m + \boldsymbol{\epsilon}_m$. Thus

$$\begin{aligned}\left\|\hat{\boldsymbol{\gamma}}_m - \left(B^T\Sigma_m^x B\right)\boldsymbol{\eta}_m^0\right\|_\infty &= \left\|\frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{y}_m - \left(B^T\boldsymbol{\Sigma}_m^x B\right)\boldsymbol{\eta}_m^0\right\|_\infty \\ &= \left\|\frac{1}{n}\boldsymbol{W}_m^T(\boldsymbol{C}_m\boldsymbol{\eta}_m^0 + \boldsymbol{A}_m + \boldsymbol{\epsilon}_m) - \left(B^T\boldsymbol{\Sigma}_m^x B\right)\boldsymbol{\eta}_m^0\right\|_\infty \\ &\leq \left\|\frac{\boldsymbol{W}_m^T\boldsymbol{\epsilon}_m}{n}\right\|_\infty + \left\|\frac{\boldsymbol{W}_m^T\boldsymbol{A}_m}{n}\right\|_\infty + \left\|\left\{B^T\boldsymbol{\Sigma}_m^x B - \frac{\boldsymbol{W}_m^T\boldsymbol{C}_m}{n}\right\}\boldsymbol{\eta}_m^0\right\|_\infty\end{aligned}$$

Then using the results from Lemma 14 of Loh and Wainwright (2012) we further have for the first term

$$\mathsf{P}\left(\left\|\frac{\boldsymbol{W}_m^T\boldsymbol{\epsilon}_m}{n}\right\|_\infty \geq c_0\sigma\sigma_m\sqrt{\frac{\log p}{n}}\right) \leq c_1\exp\{-c_2\log p\}$$

and for the third term

$$\mathsf{P}\left(\left\|B^T\boldsymbol{\Sigma}_m^x B - \frac{\boldsymbol{W}_m^T\boldsymbol{C}_m}{n}\right\|_\infty \geq c_0\sigma\sigma_x\sqrt{\frac{\log p}{n}}\right) \leq c_1\exp\{-c_2\log p\}$$

which implies that

$$\mathsf{P}\left(\left\|\left\{B^T\boldsymbol{\Sigma}_m^x B - \frac{\boldsymbol{W}_m^T\boldsymbol{C}_m}{n}\right\}\boldsymbol{\eta}_m^0\right\|_\infty \geq c_0\sigma\sigma_x\|\boldsymbol{\eta}_m^0\|_1\sqrt{\frac{\log p}{n}}\right) \leq c_1\exp\{-c_2\log p\}$$

While for the second term, we have by the union bound and by the sub-Gaussian distributed feature of $\boldsymbol{W}_m = (\boldsymbol{w}_{m1}, \cdots, \boldsymbol{w}_{mn})^T$ as well as $\sup_{\mathbf{v}\in\mathcal{V}} |e_m^0(\mathbf{v})| = O(p^{-1/2})$ that

$$P\Big(\Big\|\frac{\boldsymbol{W}_m^T\boldsymbol{A}_m}{n}\Big\|_\infty > c_0' p^{-1/2}\sigma\sqrt{\frac{\log p}{n}}\Big) \le \sum_{j=1}^{p} P\Big(\Big|\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{w}_{mi,j}^T\boldsymbol{a}_{mi}\Big| > c_0' p^{-1/2}\sigma\sqrt{\frac{\log p}{n}}\Big)$$
$$\le c_1\exp(-c_2\log p)$$

Putting together these results leads us to bound the first deviation condition with parameter $\phi = \max_{m\in\{1,\cdots,M\}}\{c_0\sigma(\sigma_m + \sigma\|\boldsymbol{\eta}_m^0\|_1)\} + c_0'\sigma p^{-1/2}$. For the second deviation condition, we have directly from Lemma 14 of Loh and Wainwright (2012) that

$$\begin{aligned}
&\mathsf{P}\Big(\Big\|\Big(\hat{\boldsymbol{\Gamma}}_m - B^T\boldsymbol{\Sigma}_m^x B\Big)\boldsymbol{\eta}_m^0\Big\|_\infty \ge c_0\sigma^2\|\boldsymbol{\eta}_m^0\|_1\sqrt{\frac{\log p}{n}}\Big)\\
&= \mathsf{P}\Big(\Big\|\Big\{\frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{W}_m - B^T\Big(\boldsymbol{\Sigma}_u + \boldsymbol{\Sigma}_m^x\Big)B\Big\}\boldsymbol{\eta}_m^0\Big\|_\infty \ge c_0\sigma^2\|\boldsymbol{\eta}_m^0\|_1\sqrt{\frac{\log p}{n}}\Big)\\
&\le c_1\exp\{-c_2\log p\}
\end{aligned}$$

Thus the second deviation condition can also be bounded by the same parameter ϕ.

In addition to presenting the deviation condition in two separate expressions, we can actually have only one deviation condition as $\|\hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\|_\infty \le \phi\sqrt{\frac{\log p}{n}}$, which will also be used in proving Theorems 3.1 and 3.2. We know that

$$\begin{aligned}
\|\hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\|_\infty &= \Big\|\frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{y}_m - \Big(\frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{W}_m - B^T\boldsymbol{\Sigma}_u B\Big)\boldsymbol{\eta}_m^0\Big\|_\infty\\
&= \Big\|\frac{1}{n}\boldsymbol{W}_m^T(\boldsymbol{C}_m\boldsymbol{\eta}_m^0 + \boldsymbol{A}_m + \boldsymbol{\epsilon}_m) - \Big(\frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{W}_m - B^T\boldsymbol{\Sigma}_u B\Big)\boldsymbol{\eta}_m^0\Big\|_\infty\\
&\le \Big\|\frac{\boldsymbol{W}_m^T\boldsymbol{\epsilon}_m}{n}\Big\|_\infty + \Big\|\frac{\boldsymbol{W}_m^T\boldsymbol{A}_m}{n}\Big\|_\infty + \Big\|\Big\{B^T\boldsymbol{\Sigma}_u B - \frac{\boldsymbol{W}_m^T(\boldsymbol{W}_m - \boldsymbol{C}_m)}{n}\Big\}\boldsymbol{\eta}_m^0\Big\|_\infty
\end{aligned}$$

We have already discussed the first and second terms. For the third term, similarly from Lemma 14 of Loh and Wainwright (2012) we have that

$$\mathsf{P}\Big(\Big\|\Big\{B^T\boldsymbol{\Sigma}_u B - \frac{\boldsymbol{W}_m^T(\boldsymbol{W}_m - \boldsymbol{C}_m)}{n}\Big\}\boldsymbol{\eta}_m^0\Big\|_\infty \ge c_0\sigma\sigma_u\|\boldsymbol{\eta}_m^0\|_1\sqrt{\frac{\log p}{n}}\Big) \le c_1\exp\{-c_2\log p\}$$

Thus we know the same parameter ϕ will still work to bound this single deviation condition.

□

3.7.6 Proof of Theorem 3.3.1

Proof. We prove the following results conditional on the deviation condition and the lower-RE condition, which have been shown to hold with probability at least $1-c_1\exp\{-c_2\log p\}$ from Lemma 3.1 in the manuscript and Lemma 2 in the Supplementary Materials. Denote the loss function as $\mathcal{L}(\boldsymbol{\eta}) = \sum_{m=1}^{M}\left\{\frac{1}{2}\boldsymbol{\eta}_m^T\hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m - \langle\hat{\boldsymbol{\gamma}}_m,\boldsymbol{\eta}_m\rangle\right\} + \lambda_n\rho(\boldsymbol{\eta})$ where $\rho(\boldsymbol{\eta}) = \sum_{j=1}^{p}\Big(\sum_{m=1}^{M}|\eta_{mj}|\Big)^{1/2}$. With the assumption $R \geq \rho(\boldsymbol{\eta}^0)$ we are guaranteed that $\boldsymbol{\eta}^0$ is feasible and by definition $\mathcal{L}(\hat{\boldsymbol{\eta}}) \leq \mathcal{L}(\boldsymbol{\eta}^0)$. Defining $\hat{\boldsymbol{\nu}} = \hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0$, through some algebra we obtain the equivalent inequality

$$\sum_{m=1}^{M}\frac{1}{2}\hat{\boldsymbol{\nu}}_m^T\hat{\boldsymbol{\Gamma}}_m\hat{\boldsymbol{\nu}}_m \leq \sum_{m=1}^{M}\langle\hat{\boldsymbol{\nu}}_m,\hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\rangle + \lambda_n\Big\{\sum_{j=1}^{p}\Big(\sum_{m=1}^{M}|\eta_{mj}^0|\Big)^{1/2} - \sum_{j=1}^{p}\Big(\sum_{m=1}^{M}|\hat{\eta}_{mj}|\Big)^{1/2}\Big\} \tag{3.7.13}$$

Note that assuming the deviation condition holds, then for the first term on RHS

$$\begin{aligned}\sum_{m=1}^{M}\langle\hat{\boldsymbol{\nu}}_m,\hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\rangle &\leq \sum_{m=1}^{M}\|\hat{\boldsymbol{\nu}}_m\|_1\|\hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\|_\infty \leq \phi\sqrt{\frac{\log p}{n}}\Big(\sum_{m=1}^{M}\|\hat{\boldsymbol{\nu}}_m\|_1\Big)\\ &= \phi\sqrt{\frac{\log p}{n}}\|\hat{\boldsymbol{\nu}}\|_1 = \phi\sqrt{\frac{\log p}{n}}(\|\hat{\boldsymbol{\nu}}_S\|_1 + \|\hat{\boldsymbol{\nu}}_{S^C}\|_1)\end{aligned} \tag{3.7.14}$$

Next we will establish an upper bound for the second term on RHS $\sum_{j=1}^{p}\Big(\sum_{m=1}^{M}|\eta_{mj}^0|\Big)^{1/2} - \sum_{j=1}^{p}\Big(\sum_{m=1}^{M}|\hat{\eta}_{mj}|\Big)^{1/2}$. Note that for $j \in \{j : \|\boldsymbol{\eta}_{(j)}^0\|_1 = 0\}$,

$$\begin{aligned}\sum_{j\in\{j:\|\boldsymbol{\eta}_{(j)}^0\|_1=0\}}\Big\{\Big(\sum_{m=1}^{M}|\eta_{mj}^0|\Big)^{1/2} - \Big(\sum_{m=1}^{M}|\hat{\eta}_{mj}|\Big)^{1/2}\Big\} &= \sum_{j\in\{j:\|\boldsymbol{\eta}_{(j)}^0\|_1=0\}} -\Big(\sum_{m=1}^{M}|\hat{\nu}_{mj}|\Big)^{1/2}\\ &\leq -\Big(\sum_{j\in\{j:\|\boldsymbol{\eta}_{(j)}^0\|_1=0\}}\sum_{m=1}^{M}|\hat{\nu}_{mj}|\Big)^{1/2}\\ &= -(\|\hat{\boldsymbol{\nu}}_{S^C}\|_1)^{1/2}\\ &\leq -R^{-1}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1\end{aligned} \tag{3.7.15}$$

We use the inequality $\sum_{i=1}^{I} a_i^{1/2} \geq (\sum_{i=1}^{I} a_i)^{1/2}$ for $a_1, \cdots, a_I \geq 0$ and

$$\begin{aligned}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1^{1/2} &\leq \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1=0\}} \Big(\sum_{m=1}^{M} |\hat{\nu}_{mj}|\Big)^{1/2} = \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1=0\}} \Big(\sum_{m=1}^{M} |\hat{\eta}_{mj}|\Big)^{1/2} \\ &\leq \sum_{j=1}^{p} \Big(\sum_{m=1}^{M} |\hat{\eta}_{mj}|\Big)^{1/2} \leq R\end{aligned}$$

in establishing the inequalities above. While for $j \in \{j : \|\boldsymbol{\eta}^0_{(j)}\|_1 > 0\}$, we will use the following inequality: for $a > 0$, $b > 0$,

$$(a^{1/2} - b^{1/2})^2 = (a-b)^2/(a^{1/2} + b^{1/2})^2 \leq (a-b)^2/(a+b)$$

or

$$a^{1/2} - b^{1/2} \leq \sqrt{(a-b)^2/(a+b)} = |a-b|/\sqrt{a+b}$$

Plugging in $a = \sum_{m=1}^{M} |\eta^0_{mj}|$ and $b = \sum_{m=1}^{M} |\hat{\eta}_{mj}|$, we have for $j \in \{j : \|\boldsymbol{\eta}^0_{(j)}\|_1 > 0\}$,

$$\begin{aligned}&\sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \left\{ \Big(\sum_{m=1}^{M} |\eta^0_{mj}|\Big)^{1/2} - \Big(\sum_{m=1}^{M} |\hat{\eta}_{mj}|\Big)^{1/2} \right\} \\ &\leq \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \left\{ \frac{|\sum_{m=1}^{M} |\eta^0_{mj}| - \sum_{m=1}^{M} |\hat{\eta}_{mj}||}{\sqrt{\sum_{m=1}^{M} |\eta^0_{mj}| + \sum_{m=1}^{M} |\hat{\eta}_{mj}|}} \right\} \\ &\leq \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \left\{ \frac{\sum_{m=1}^{M} ||\eta^0_{mj}| - |\hat{\eta}_{mj}||}{\sqrt{\sum_{m=1}^{M} |\eta^0_{mj}| + \sum_{m=1}^{M} |\hat{\eta}_{mj}|}} \right\} \\ &\leq \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \left\{ \frac{\sum_{m=1}^{M} |\hat{\nu}_{mj}|}{\sqrt{\sum_{m=1}^{M} |\eta^0_{mj}| + \sum_{m=1}^{M} |\hat{\eta}_{mj}|}} \right\} \\ &\leq \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \left\{ \frac{\sum_{m=1}^{M} |\hat{\nu}_{mj}|}{\sqrt{\sum_{m=1}^{M} |\eta^0_{mj}|}} \right\} \\ &\leq l^{-1} \|\hat{\boldsymbol{\nu}}_S\|_1\end{aligned} \tag{3.7.16}$$

Then by combing results from (3.7.14), (3.7.15), and (3.7.16), we obtain an upper bound for the RHS of (3.7.13) as

$$\phi\sqrt{\frac{\log p}{n}}(\|\hat{\boldsymbol{\nu}}_S\|_1 + \|\hat{\boldsymbol{\nu}}_{S^C}\|_1) + \lambda_n(l^{-1}\|\hat{\boldsymbol{\nu}}_S\|_1 - R^{-1}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1) \tag{3.7.17}$$

Our choice of λ_n guarantees that the term (3.7.17) is at most $\frac{3l^{-1}\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_S\|_1 - \frac{R^{-1}\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1$. And it is easy to see the LHS of (3.7.13) is non-negative, so we have $\frac{3l^{-1}\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_S\|_1 - \frac{R^{-1}\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1 \geq 0$, or $\|\hat{\boldsymbol{\nu}}_{S^C}\|_1 \leq 3l^{-1}R\|\hat{\boldsymbol{\nu}}_S\|_1$. Consequently, we have the inequality that

$$\|\hat{\boldsymbol{\nu}}\|_1 = \|\hat{\boldsymbol{\nu}}_S\|_1 + \|\hat{\boldsymbol{\nu}}_{S^C}\|_1 \leq \big(1 + 3l^{-1}R\big)\|\hat{\boldsymbol{\nu}}_S\|_1 \leq h_1\sqrt{Mk}\|\hat{\boldsymbol{\nu}}\|_2 \tag{3.7.18}$$

Now from the lower-RE condition for $\hat{\boldsymbol{\Gamma}}_m$'s and the assumption that $\alpha_1/\tau(n,p) \geq 2h^2Mk$, and also using (3.7.18) we have

$$\begin{aligned}
\sum_{m=1}^{M}\hat{\boldsymbol{\nu}}_m^T\hat{\boldsymbol{\Gamma}}_m\hat{\boldsymbol{\nu}}_m &\geq \sum_{m=1}^{M}\Big(\alpha_1\|\hat{\boldsymbol{\nu}}_m\|_2^2 - \tau(n,p)\|\hat{\boldsymbol{\nu}}_m\|_1^2\Big) \\
&= \alpha_1\|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p)\Big(\sum_{m=1}^{M}\|\hat{\boldsymbol{\nu}}_m\|_1^2\Big) \\
&\geq \alpha_1\|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p)\Big(\sum_{m=1}^{M}\|\hat{\boldsymbol{\nu}}_m\|_1\Big)^2 \\
&= \alpha_1\|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p)\|\hat{\boldsymbol{\nu}}\|_1^2 \\
&\geq \alpha_1\|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p)h_1^2Mk\|\hat{\boldsymbol{\nu}}\|_2^2 \\
&\geq \frac{\alpha_1}{2}\|\hat{\boldsymbol{\nu}}\|_2^2
\end{aligned} \tag{3.7.19}$$

Then combining (3.7.13), (3.7.18), (3.7.19) and the upper bound in (3.7.17) we have

$$\begin{aligned}
\frac{\alpha_1}{4}\|\hat{\boldsymbol{\nu}}\|_2^2 &\leq \phi\sqrt{\frac{\log p}{n}}\|\hat{\boldsymbol{\nu}}\|_1 + \lambda_n l^{-1}\|\hat{\boldsymbol{\nu}}\|_1 \\
&\leq 2\max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n l^{-1}\right\}\|\hat{\boldsymbol{\nu}}\|_1 \\
&\leq 2h_1\sqrt{Mk}\max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n l^{-1}\right\}\|\hat{\boldsymbol{\nu}}\|_2
\end{aligned} \tag{3.7.20}$$

which leads to the error bound in L_2 norm:

$$\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2 = \|\hat{\boldsymbol{\nu}}\|_2 \leq \frac{8h_1\sqrt{Mk}}{\alpha_1} \max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n l^{-1}\right\}$$

By applying (3.7.18) again we have the error bound in L_1 norm:

$$\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_1 = \|\hat{\boldsymbol{\nu}}\|_1 \leq h_1\sqrt{Mk}\|\hat{\boldsymbol{\nu}}\|_2 \leq \frac{8h_1^2 Mk}{\alpha_1} \max\left\{\phi\sqrt{\frac{\log p}{n}}, \lambda_n l^{-1}\right\}$$

□

3.7.7 Proof of Corollary 3.3.1

Proof. When the restriction changes to $\|\boldsymbol{\eta}\|_1 \leq R^2$, the only difference in the proof procedure involves the equation (3.7.15). Now we have for

$$(\|\hat{\boldsymbol{\nu}}_{S^C}\|_1)^{1/2} = \left(\sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1=0\}}\sum_{m=1}^{M}|\hat{\nu}_{mj}|\right)^{1/2} = \left(\sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1=0\}}\sum_{m=1}^{M}|\hat{\eta}_{mj}|\right)^{1/2} \leq \|\hat{\boldsymbol{\eta}}\|_1^{1/2} \leq R$$

Based on this inequality, the conclusion in (3.7.15) still goes through. Thus the results in Theorem 3.1 still hold in the current setting. □

3.7.8 Proof of Theorem 3.3.2

Proof. We prove the following results conditional on the deviation condition and the lower-RE condition, which have been shown to hold with probability at least $1 - c_1\exp\{-c_2\log p\}$ from Lemma 3.1 in the manuscript and Lemma 2 in the Supplementary Materials. Denote the loss function as $\mathcal{L}(\boldsymbol{\eta}) = \sum_{m=1}^{M}\left\{\frac{1}{2}\boldsymbol{\eta}_m^T\hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m - \langle\hat{\boldsymbol{\gamma}}_m, \boldsymbol{\eta}_m\rangle\right\} + \lambda_n\rho(\boldsymbol{\eta})$ where $\rho(\boldsymbol{\eta}) = \sum_{j=1}^{p}\left(\sum_{m=1}^{M}|\eta_{mj}|^q\right)^{1/q}$ with $q > 1$. With the assumption $R \geq \rho(\boldsymbol{\eta}^0)$ we are guaranteed that $\boldsymbol{\eta}^0$ is feasible and by definition $\mathcal{L}(\hat{\boldsymbol{\eta}}) \leq \mathcal{L}(\boldsymbol{\eta}^0)$. Defining $\hat{\boldsymbol{\nu}} = \hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0$, through some

algebra we obtain the equivalent inequality

$$\sum_{m=1}^{M} \frac{1}{2} \hat{\boldsymbol{\nu}}_m^T \hat{\boldsymbol{\Gamma}}_m \hat{\boldsymbol{\nu}}_m \leq \sum_{m=1}^{M} \langle \hat{\boldsymbol{\nu}}_m, \hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m \boldsymbol{\eta}_m^0 \rangle + \lambda_n \Big\{ \sum_{j=1}^{p} \Big(\sum_{m=1}^{M} |\eta_{mj}^0|^q \Big)^{1/q} - \sum_{j=1}^{p} \Big(\sum_{m=1}^{M} |\hat{\eta}_{mj}|^q \Big)^{1/q} \Big\} \tag{3.7.21}$$

Note that assuming the deviation condition holds, then for the first term on RHS

$$\begin{aligned} \sum_{m=1}^{M} \langle \hat{\boldsymbol{\nu}}_m, \hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m \boldsymbol{\eta}_m^0 \rangle &\leq \sum_{m=1}^{M} \|\hat{\boldsymbol{\nu}}_m\|_1 \|\hat{\boldsymbol{\gamma}}_m - \hat{\boldsymbol{\Gamma}}_m \boldsymbol{\eta}_m^0\|_\infty \\ &\leq \phi \sqrt{\frac{\log p}{n}} \Big(\sum_{m=1}^{M} \|\hat{\boldsymbol{\nu}}_m\|_1 \Big) \\ &= \phi \sqrt{\frac{\log p}{n}} \|\hat{\boldsymbol{\nu}}\|_1 \\ &= \phi \sqrt{\frac{\log p}{n}} (\|\hat{\boldsymbol{\nu}}_S\|_1 + \|\hat{\boldsymbol{\nu}}_{S^C}\|_1) \end{aligned} \tag{3.7.22}$$

Next we will establish an upper bound for the second term on RHS

$$\sum_{j=1}^{p} \Big(\sum_{m=1}^{M} |\eta_{mj}^0|^q \Big)^{1/q} - \sum_{j=1}^{p} \Big(\sum_{m=1}^{M} |\hat{\eta}_{mj}|^q \Big)^{1/q}.$$

Note that for $j \in \{j : \|\boldsymbol{\eta}_{(j)}^0\|_1 = 0\}$,

$$\begin{aligned} &\sum_{j \in \{j : \|\boldsymbol{\eta}_{(j)}^0\|_1 = 0\}} \Big\{ \Big(\sum_{m=1}^{M} |\eta_{mj}^0|^q \Big)^{1/q} - \Big(\sum_{m=1}^{M} |\hat{\eta}_{mj}|^q \Big)^{1/q} \Big\} \\ = &\sum_{j \in \{j : \|\boldsymbol{\eta}_{(j)}^0\|_1 = 0\}} - \Big(\sum_{m=1}^{M} |\hat{\nu}_{mj}|^q \Big)^{1/q} \\ \leq &- \sum_{j \in \{j : \|\boldsymbol{\eta}_{(j)}^0\|_1 = 0\}} M^{-(q-1)/q} \Big(\sum_{m=1}^{M} |\hat{\nu}_{mj}| \Big) \\ = &-M^{-(q-1)/q} \|\hat{\boldsymbol{\nu}}_{S^C}\|_1 \end{aligned} \tag{3.7.23}$$

where we use the Hölder's inequality that

$$\sum_{m=1}^{M} |\hat{\nu}_{mj}| \leq \Big(\sum_{m=1}^{M} |\hat{\nu}_{mj}|^q \Big)^{1/q} \Big(\sum_{m=1}^{M} 1^{q/(q-1)} \Big)^{(q-1)/q}.$$

While for $j \in \{j : \|\boldsymbol{\eta}^0_{(j)}\|_1 > 0\}$,

$$\begin{aligned}
&\sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \Big\{ \Big(\sum_{m=1}^{M} |\eta^0_{mj}|^q \Big)^{1/q} - \Big(\sum_{m=1}^{M} |\hat{\eta}_{mj}|^q \Big)^{1/q} \Big\} \\
&= \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \Big\{ \Big(\sum_{m=1}^{M} |\eta^0_{mj}|^q \Big)^{1/q} - \Big(\sum_{m=1}^{M} |\eta^0_{mj} + \hat{\nu}_{mj}|^q \Big)^{1/q} \Big\} \\
&\leq \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \Big\{ \Big(\sum_{m=1}^{M} |\eta^0_{mj}|^q \Big)^{1/q} - \Big(\sum_{m=1}^{M} |\eta^0_{mj}|^q \Big)^{1/q} + \Big(\sum_{m=1}^{M} |\hat{\nu}_{mj}|^q \Big)^{1/q} \Big\} \\
&= \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \Big(\sum_{m=1}^{M} |\hat{\nu}_{mj}|^q \Big)^{1/q} \\
&\leq \sum_{j\in\{j:\|\boldsymbol{\eta}^0_{(j)}\|_1>0\}} \Big(\sum_{m=1}^{M} |\hat{\nu}_{mj}| \Big) \\
&= \|\hat{\boldsymbol{\nu}}_S\|_1
\end{aligned} \tag{3.7.24}$$

Then by combing results from (3.7.22), (3.7.23), and (3.7.24), we obtain an upper bound for the RHS of (3.7.21) as

$$\phi\sqrt{\frac{\log p}{n}}(\|\hat{\boldsymbol{\nu}}_S\|_1 + \|\hat{\boldsymbol{\nu}}_{S^C}\|_1) + \lambda_n(\|\hat{\boldsymbol{\nu}}_S\|_1 - M^{-(q-1)/q}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1) \tag{3.7.25}$$

Our choice of λ_n guarantees that the term (3.7.25) is at most $\frac{3\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_S\|_1 - \frac{M^{-(q-1)/q}\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1$. And it is easy to see the LHS of (3.7.21) is non-negative, so we have

$$\frac{3\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_S\|_1 - \frac{M^{-(q-1)/q}\lambda_n}{2}\|\hat{\boldsymbol{\nu}}_{S^C}\|_1 \geq 0, \text{ or } \|\hat{\boldsymbol{\nu}}_{S^C}\|_1 \leq 3M^{(q-1)/q}\|\hat{\boldsymbol{\nu}}_S\|_1.$$

Consequently, we have the inequality that

$$\|\hat{\boldsymbol{\nu}}\|_1 = \|\hat{\boldsymbol{\nu}}_S\|_1 + \|\hat{\boldsymbol{\nu}}_{S^C}\|_1 \leq \big(1 + 3M^{(q-1)/q}\big)\|\hat{\boldsymbol{\nu}}_S\|_1 \leq h_2\sqrt{Mk}\|\hat{\boldsymbol{\nu}}\|_2 \tag{3.7.26}$$

Now from the lower-RE condition for $\hat{\mathbf{\Gamma}}_m$'s and the assumption that $\alpha_1/\tau(n,p) \geq 2h_2^2 Mk$, and also using (3.7.26) we have

$$\begin{aligned}
\sum_{m=1}^{M} \hat{\boldsymbol{\nu}}_m^T \hat{\mathbf{\Gamma}}_m \hat{\boldsymbol{\nu}}_m &\geq \sum_{m=1}^{M} \Big(\alpha_1 \|\hat{\boldsymbol{\nu}}_m\|_2^2 - \tau(n,p) \|\hat{\boldsymbol{\nu}}_m\|_1^2 \Big) \\
&= \alpha_1 \|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p) \Big(\sum_{m=1}^{M} \|\hat{\boldsymbol{\nu}}_m\|_1^2 \Big) \\
&\geq \alpha_1 \|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p) \Big(\sum_{m=1}^{M} \|\hat{\boldsymbol{\nu}}_m\|_1 \Big)^2 \\
&= \alpha_1 \|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p) \|\hat{\boldsymbol{\nu}}\|_1^2 \\
&\geq \alpha_1 \|\hat{\boldsymbol{\nu}}\|_2^2 - \tau(n,p) h_2^2 Mk \|\hat{\boldsymbol{\nu}}\|_2^2 \\
&\geq \frac{\alpha_1}{2} \|\hat{\boldsymbol{\nu}}\|_2^2
\end{aligned} \tag{3.7.27}$$

Then combining (3.7.21), (3.7.26), (3.7.27) and the upper bound in (3.7.25) we have

$$\begin{aligned}
\frac{\alpha_1}{4} \|\hat{\boldsymbol{\nu}}\|_2^2 &\leq \phi \sqrt{\frac{\log p}{n}} \|\hat{\boldsymbol{\nu}}\|_1 + \lambda_n \|\hat{\boldsymbol{\nu}}\|_1 \\
&\leq 2 \max \left\{ \phi \sqrt{\frac{\log p}{n}}, \lambda_n \right\} \|\hat{\boldsymbol{\nu}}\|_1 \\
&\leq 2 h_2 \sqrt{Mk} \max \left\{ \phi \sqrt{\frac{\log p}{n}}, \lambda_n \right\} \|\hat{\boldsymbol{\nu}}\|_2
\end{aligned} \tag{3.7.28}$$

which leads to the error bound in L_2 norm:

$$\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2 = \|\hat{\boldsymbol{\nu}}\|_2 \leq \frac{8 h_2 \sqrt{Mk}}{\alpha_1} \max \left\{ \phi \sqrt{\frac{\log p}{n}}, \lambda_n \right\}$$

By applying (3.7.26) again we have the error bound in L_1 norm:

$$\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_1 = \|\hat{\boldsymbol{\nu}}\|_1 \leq h_2 \sqrt{Mk} \|\hat{\boldsymbol{\nu}}\|_2 \leq \frac{8 h_2^2 Mk}{\alpha_1} \max \left\{ \phi \sqrt{\frac{\log p}{n}}, \lambda_n \right\}$$

□

3.7.9 Proof of Theorem 3.3.3

Proof. We prove the following results conditional on the lower- and upper-RE conditions, which have been shown to hold with probability at least $1 - c_1 \exp\{-c_2 \log p\}$ from Lemma 2 in the Supplementary Materials. We utilize Theorem 2 in Agarwal et al. (2012) and its remarks to prove this theorem. First we need to show the L_q norm ($q > 1$) is decomposable as defined in Definition 3 of Agarwal et al. (2012) and find the subspace compatibility constant as defined in Definition 4 of Agarwal et al. (2012).

For any subset S of $\{1, \cdots, p\}$, define the subspace $\mathcal{M}(S) = \Big\{\boldsymbol{\eta} \in \mathbb{R}^{Mp} \Big| \|\boldsymbol{\eta}_{(j)}\|_1 = 0, \forall j \notin S\Big\}$ and its orthogonal complement $\mathcal{M}^{\perp}(S) = \Big\{\boldsymbol{\eta} \in \mathbb{R}^{Mp} \Big| \|\boldsymbol{\eta}_{(j)}\|_1 = 0, \forall j \in S\Big\}$ where $\boldsymbol{\eta}_{(j)} = (\eta_{1j}, \cdots, \eta_{Mj})^T$. Then for any pairs of vectors $\boldsymbol{\eta}_1 \in \mathcal{M}(S)$ and $\boldsymbol{\eta}_2 \in \mathcal{M}^{\perp}(S)$, we have

$$\begin{aligned}
\rho(\boldsymbol{\eta}_1 + \boldsymbol{\eta}_2) &= \sum_{j=1}^{p} \Big(\sum_{m=1}^{M} |\eta_{1,mj} + \eta_{2,mj}|^q \Big)^{1/q} \\
&= \sum_{j \in S} \Big(|\eta_{1,mj} + 0|^q \Big)^{1/q} + \sum_{j \notin S} \Big(|0 + \eta_{2,mj}|^q \Big)^{1/q} \\
&= \sum_{j \in S} \Big(|\eta_{1,mj}|^q \Big)^{1/q} + \sum_{j \notin S} \Big(|\eta_{2,mj}|^q \Big)^{1/q} \\
&= \rho(\boldsymbol{\eta}_1) + \rho(\boldsymbol{\eta}_2)
\end{aligned}$$

This has shown the decomposability of the L_q norm. We also need to determine the subspace compatibility constant for it. For $\boldsymbol{u} \in \mathcal{M}(S)/0$ and subset S with cardinality k, using Cauchy-Schwarz inequality we have

$$\begin{aligned}
\rho(\boldsymbol{u}) = \sum_{j=1}^{p} \Big(\sum_{m=1}^{M} |u_{mj}|^q \Big)^{1/q} = \sum_{j \in S} \Big(\sum_{m=1}^{M} |u_{mj}|^q \Big)^{1/q} \leq \sum_{j \in S} \Big(\sum_{m=1}^{M} |u_{mj}| \Big) \\
\leq \sqrt{\Big(\sum_{j \in S} \sum_{m=1}^{M} |u_{mj}|^2 \Big)} \sqrt{Mk} = \sqrt{Mk} \|\boldsymbol{u}\|_2
\end{aligned}$$

Thus the subspace compatibiity constant $\Psi(\mathcal{M}) := \sup_{\boldsymbol{u} \in \mathcal{M}(S)/0} \frac{\rho(\boldsymbol{u})}{\|\boldsymbol{u}\|_2} = \sqrt{Mk}$.

We also need to show that the lower- and upper-RE conditions imply the restricted strong convexity (RSC) and restricted smoothness (RSM) conditions with $L_{1,q}$ norm. First from

Hölder's inequality, we have

$$\begin{aligned}\sum_{m=1}^{M}\Big(\sum_{j=1}^{p}|\theta_{mj}|\Big)^2 &\le \Big(\sum_{m=1}^{M}\sum_{j=1}^{p}|\theta_{mj}|\Big)^2 = \Big(\sum_{j=1}^{p}\sum_{m=1}^{M}|\theta_{mj}|\Big)^2 \\ &\le \Big[\sum_{j=1}^{p}M^{(q-1)/q}\Big(\sum_{m=1}^{M}|\theta_{mj}|^q\Big)^{1/q}\Big]^2\end{aligned} \tag{3.7.29}$$

Thus for $\boldsymbol{\theta} = (\boldsymbol{\theta}_1^T, \cdots, \boldsymbol{\theta}_M^T)^T \in \mathbb{R}^{Mp}$ where $\boldsymbol{\theta}_m = (\theta_{m1}, \cdots, \theta_{mp})^T$ for $m = 1, \cdots, M$, from lower-RE condition, we have

$$\sum_{m=1}^{M}\boldsymbol{\theta}_m^T\hat{\boldsymbol{\Gamma}}_m\boldsymbol{\theta}_m \ge \alpha_1\Big(\sum_{m=1}^{M}\|\boldsymbol{\theta}_m\|_2^2\Big) - \tau(n,p)\Big(\sum_{m=1}^{M}\|\boldsymbol{\theta}_m\|_1^2\Big).$$

Combined with (3.7.29), we have

$$\begin{aligned}\sum_{m=1}^{M}\boldsymbol{\theta}_m^T\hat{\boldsymbol{\Gamma}}_m\boldsymbol{\theta}_m &\ge \alpha_1\Big(\sum_{m=1}^{M}\|\boldsymbol{\theta}_m\|_2^2\Big) - \tau(n,p)\Big[\sum_{j=1}^{p}M^{(q-1)/q}\Big(\sum_{m=1}^{M}|\theta_{mj}|^q\Big)^{1/q}\Big]^2 \\ &= \alpha_1\|\boldsymbol{\theta}\|_2^2 - \tau(n,p)M^{2(q-1)/q}\rho^2(\boldsymbol{\theta})\end{aligned}$$

Compared with the RSC condition in Agarwal et al. (2012), it suffices to have $\gamma_l = 2\alpha_1$ and $\tau_l = \tau(n,p)M^{2(q-1)/q}$. Similarly having $\gamma_u = 2\alpha_2$ and $\tau_u = \tau(n,p)M^{2(q-1)/q}$ gives us the RSM condition.

One last step is to find the contraction coefficient and the tolerance parameter. The compound contraction coefficient is defined as $\kappa := \left\{1 - \frac{\bar{\gamma}_l}{8\alpha_2} + \frac{64k\tau M^{(3q-2)/q}}{\bar{\gamma}_l}\right\}\xi$ where $\bar{\gamma}_l := 2\alpha_1 - 64k\tau M^{(3q-2)/q}$ and $\xi := \frac{\bar{\gamma}_l}{2\bar{\gamma}_l - \gamma_l}$. From assumption in our Theorem 3.2, $\bar{\gamma}_l > 0$. Then given that $\tau \asymp \frac{\log p}{n}$ and that $n \gtrsim Mk\log p$, we have $\xi = \mathcal{O}(1)$ and $\kappa \in (0,1)$. The compound tolerance parameter is defined as $\epsilon^2 := 8\xi\omega\Big[6\sqrt{Mk}\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2 + 8\rho\big(\Pi_{\mathcal{M}^\perp}(\boldsymbol{\eta}^0)\big)\Big]^2$ where $\omega := \Big(\frac{\bar{\gamma}_l}{4\alpha_2} + \frac{256k\tau M^{(3q-2)/q}}{\bar{\gamma}_l} + 10\Big)\tau M^{2(q-1)/q}$. Again given that $\tau \asymp \frac{\log p}{n}$ and that $n \gtrsim Mk\log p$, we know that $\omega = \mathcal{O}(\frac{\log p}{n})$. Also as $\boldsymbol{\eta}^0$ is feasible with the restriction, $\rho\big(\Pi_{\mathcal{M}^\perp}(\boldsymbol{\eta}^0)\big) = 0$. Thus the compound tolerance parameter $\epsilon^2 = \mathcal{O}(\frac{\log p}{n})k\|\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}^0\|_2^2$.

Finally by applying Theorem 2 in Agarwal et al. (2012) and its remarks, we have for any

$t \geq 2[\log(1/\kappa)]^{-1} \log \frac{\mathcal{L}(\boldsymbol{\eta}^*)-\mathcal{L}(\hat{\boldsymbol{\eta}})}{\delta^2} + \log_2 \log_2(\frac{R\lambda_n}{\delta^2})(1 + \frac{\log 2}{\log(1/\kappa)})$ and $\delta^2 \geq (1-\kappa)^{-1}\epsilon^2$,

$$\|\boldsymbol{\eta}^{(t)} - \hat{\boldsymbol{\eta}}\|_2^2 \leq \frac{2\lambda_n^2 + 16\tau M^{2(q-1)/q}}{\bar{\gamma}_l \lambda_n^2}\delta^2 + \frac{144\tau k M^{(3q-2)/q}}{\bar{\gamma}_l} = \mathcal{O}(1)\delta^2 + o(1)$$

Thus putting together the results for ϵ^2 and δ^2 leads us to the conclusion. □

3.7.10 Proof of Corollary 3.3.2

Proof. The first estimator of $\hat{\boldsymbol{\Sigma}}_u$ has expression $\hat{\boldsymbol{\Sigma}}_u = \frac{1}{n_0} U_0^T U_0$ where U_0 has n_0 independently sub-Gaussian distributed rows of the measurement errors. The second estimator has expression

$$\hat{\boldsymbol{\Sigma}}_u = \frac{1}{n^*(M-1)} \sum_{i=1}^{n^*} \sum_{m=1}^{M} (\boldsymbol{z}_{mi} - \bar{\boldsymbol{z}}_{\cdot i})(\boldsymbol{z}_{mi} - \bar{\boldsymbol{z}}_{\cdot i})^T$$

under the assumptions that $\boldsymbol{z}_{mi} = \boldsymbol{x}_i + \boldsymbol{u}_{mi}$ and $\boldsymbol{u}_{mi} \overset{i.i.d.}{\sim} N(0, \boldsymbol{\Sigma}_u)$. Under this set-up, we have

$$\hat{\boldsymbol{\Sigma}}_u = \frac{1}{n^*(M-1)} \sum_{i=1}^{n^*} \sum_{m=1}^{M} (\boldsymbol{u}_{mi} - \bar{\boldsymbol{u}}_{\cdot i})(\boldsymbol{u}_{mi} - \bar{\boldsymbol{u}}_{\cdot i})^T = (1/n^*) \sum_{i=1}^{n^*} S_i,$$

where $\bar{\boldsymbol{u}}_{\cdot i} = \sum_{m=1}^{M} \boldsymbol{u}_{mi}/M$ and $n^*(M-1)\hat{\boldsymbol{\Sigma}}_u \sim Wishart(\boldsymbol{\Sigma}_u, n^*(M-1))$. Thus the second estimator has the same probability distribution as $\frac{1}{n^*(M-1)}\tilde{\boldsymbol{U}}^T\tilde{\boldsymbol{U}}$ where $\tilde{\boldsymbol{U}}$ has rows that are independently distributed as $N\big(0, \boldsymbol{\Sigma}_u\big)$. If we can prove the probability results for $\frac{1}{n^*(M-1)}\tilde{\boldsymbol{U}}^T\tilde{\boldsymbol{U}}$, then the results hold with same probabilities for the second estimator of $\hat{\boldsymbol{\Sigma}}_u$. In the following parts of the proof, we use $\hat{\boldsymbol{\Sigma}}_u$ to denote both the first estimator and $\frac{1}{n^*(M-1)}\tilde{\boldsymbol{U}}^T\tilde{\boldsymbol{U}}$. The arguments go through for both expressions in the same manner.

Define $\tilde{\boldsymbol{\Gamma}}_m = \frac{1}{n}\boldsymbol{W}_m^T\boldsymbol{W}_m - B^T\hat{\boldsymbol{\Sigma}}_u B$. We first show that the deviation condition still holds with high probability. We know that

$$\begin{aligned}\|\hat{\gamma}_m - \tilde{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\|_\infty &\leq \|\hat{\gamma}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\|_\infty + \|(\tilde{\boldsymbol{\Gamma}}_m - \hat{\boldsymbol{\Gamma}}_m)\boldsymbol{\eta}_m^0\|_\infty \\ &= \|\hat{\gamma}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\|_\infty + \|B^T(\hat{\boldsymbol{\Sigma}}_u - \boldsymbol{\Sigma}_u)B\boldsymbol{\eta}_m^0\|_\infty\end{aligned}$$

where $\|\hat{\gamma}_m - \hat{\boldsymbol{\Gamma}}_m\boldsymbol{\eta}_m^0\|_\infty$ has been shown to be bounded in Lemma 3.1 and then it suffices to show that $\|B^T(\hat{\boldsymbol{\Sigma}}_u - \boldsymbol{\Sigma}_u)B\boldsymbol{\eta}_m^0\|_\infty$ can be bounded as well. Actually applying Lemma 14

from Loh and Wainwright (2012) and also noting that $n_0 > n$, $n^*(M-1) > n$, we would obtain

$$\begin{aligned}\mathsf{P}\Big(\|B^T(\hat{\boldsymbol{\Sigma}}_u - \boldsymbol{\Sigma}_u)B\boldsymbol{\eta}_m^0\|_\infty \geq c_0\sigma_u^2\|\boldsymbol{\eta}_m^0\|_1\sqrt{\frac{\log p}{n^{**}}}\Big) &\leq \mathsf{P}\Big(\|B^T(\hat{\boldsymbol{\Sigma}}_u - \boldsymbol{\Sigma}_u)B\|_\infty \geq c_0\sigma_u^2\sqrt{\frac{\log p}{n}}\Big) \\ &\leq c_1\exp\{-c_2\log p\}\end{aligned}$$

where n^{**} refers to either n_0 or $n^*(M-1)$ in the above equation depending on the context. This is exactly what we need to show the deviation condition. Next we turn to the lower- and upper-RE conditions. Similarly we can split the terms as

$$|\boldsymbol{\theta}^T(\tilde{\boldsymbol{\Gamma}}_m - B^T\boldsymbol{\Sigma}_m^x B)\boldsymbol{\theta}| \leq |\boldsymbol{\theta}^T(\hat{\boldsymbol{\Gamma}}_m - B^T\boldsymbol{\Sigma}_m^x B)\boldsymbol{\theta}| + |\boldsymbol{\theta}^T B^T(\hat{\boldsymbol{\Sigma}}_u - \boldsymbol{\Sigma}_u)B\boldsymbol{\theta}|$$

Based on the proof of Lemma 2 in the Supplementary Materials, we can guarantee the same parameters for the lower- and upper-RE conditions by ensuring the following two inequalities:

$$\sup_{\boldsymbol{\theta}\in\mathsf{K}(2s)}|\boldsymbol{\theta}^T(\hat{\boldsymbol{\Gamma}}_m - B^T\boldsymbol{\Sigma}_m^x B)\boldsymbol{\theta}| \leq \frac{1}{108}\delta_{\min}$$

and

$$\sup_{\boldsymbol{\theta}\in\mathsf{K}(2s)}|\boldsymbol{\theta}^T B^T(\hat{\boldsymbol{\Sigma}}_u - \boldsymbol{\Sigma}_u)B\boldsymbol{\theta}| \leq \frac{1}{108}\delta_{\min}$$

hold with high probability. The first inequality has already been investigated in the proof of Lemma 2. While for the second inequality, by applying Lemma 15 of Loh and Wainwright (2012), we have for some positive constant c'' that

$$\begin{aligned}\mathsf{P}\Big(\sup_{\boldsymbol{\theta}\in\mathsf{K}(2s)}|\boldsymbol{\theta}^T B^T(\hat{\boldsymbol{\Sigma}}_u - \boldsymbol{\Sigma}_u)B\boldsymbol{\theta}| \geq \delta_{\min}/108\Big) &\leq 2\exp\Big(-c''n\min\{\frac{\delta_{\min}^2}{\sigma_u^4}, \frac{\delta_{\min}}{\sigma_u^2}\} + 2s\log p\Big) \\ &\leq 2\exp\Big(-c''n\min\{\frac{\delta_{\min}^2}{\sigma^4}, \frac{\delta_{\min}}{\sigma^2}\} + 2s\log p\Big)\end{aligned}$$

With the choice of s in the proof of Lemma 2, we see the probability statement still holds for the lower- and upper-RE conditions. □

3.7.11 Proof of Lemma 3.3.2

Proof. Using Lemma 12 from Loh and Wainwright (2012) and letting $\delta = \frac{1}{54}\delta_{\min}$, $\mathbf{\Gamma} = \hat{\mathbf{\Gamma}}_m - B^T\mathbf{\Sigma}_m^x B$, we have the bound

$$|\boldsymbol{\theta}^T(\hat{\mathbf{\Gamma}}_m - B^T\mathbf{\Sigma}_m^x B)\boldsymbol{\theta}| \leq \frac{1}{2}\delta_{\min}\big(\|\boldsymbol{\theta}\|_2^2 + \frac{1}{s}\|\boldsymbol{\theta}\|_1^2\big)$$

Then we have

$$\begin{aligned}
\boldsymbol{\theta}^T\hat{\mathbf{\Gamma}}_m\boldsymbol{\theta} &\geq \boldsymbol{\theta}^T(B^T\mathbf{\Sigma}_m^x B)\boldsymbol{\theta} - \frac{1}{2}\delta_{\min}\big(\|\boldsymbol{\theta}\|_2^2 + \frac{1}{s}\|\boldsymbol{\theta}\|_1^2\big) \\
&\geq \lambda_{\min}(B^T\mathbf{\Sigma}_m^x B)\|\boldsymbol{\theta}\|_2^2 - \frac{1}{2}\delta_{\min}\big(\|\boldsymbol{\theta}\|_2^2 + \frac{1}{s}\|\boldsymbol{\theta}\|_1^2\big) \\
&\geq \frac{1}{2}\delta_{\min}\|\boldsymbol{\theta}\|_2^2 - \frac{1}{2s}\delta_{\min}\|\boldsymbol{\theta}\|_1^2
\end{aligned}$$

and also

$$\begin{aligned}
\boldsymbol{\theta}^T\hat{\mathbf{\Gamma}}_m\boldsymbol{\theta} &\leq \boldsymbol{\theta}^T(B^T\mathbf{\Sigma}_m^x B)\boldsymbol{\theta} + \frac{1}{2}\delta_{\min}\big(\|\boldsymbol{\theta}\|_2^2 + \frac{1}{s}\|\boldsymbol{\theta}\|_1^2\big) \\
&\leq \lambda_{\max}(B^T\mathbf{\Sigma}_m^x B)\|\boldsymbol{\theta}\|_2^2 + \frac{1}{2}\delta_{\min}\big(\|\boldsymbol{\theta}\|_2^2 + \frac{1}{s}\|\boldsymbol{\theta}\|_1^2\big) \\
&\leq \frac{3}{2}\delta_{\max}\|\boldsymbol{\theta}\|_2^2 + \frac{1}{2s}\delta_{\min}\|\boldsymbol{\theta}\|_1^2
\end{aligned}$$

Thus it remains to show that

$$\sup_{\boldsymbol{\theta}\in\mathsf{K}(2s)} |\boldsymbol{\theta}^T(\hat{\mathbf{\Gamma}}_m - B^T\mathbf{\Sigma}_m^x B)\boldsymbol{\theta}| \leq \frac{1}{54}\delta_{\min}$$

with high probability for certain $s \geq 1$.

We denote $D_m(s) := \sup_{\boldsymbol{\theta}\in\mathsf{K}(2s)} |\boldsymbol{\theta}^T(\hat{\mathbf{\Gamma}}_m - B^T\mathbf{\Sigma}_m^x B)\boldsymbol{\theta}|$. We know that $\boldsymbol{W}_m$ is a sub-Gaussian matrix with parameters $\big(B^T(\mathbf{\Sigma}_m^x + \mathbf{\Sigma}_u)B, \sigma^2\big)$. Thus using the results from Lemma 15 of Loh and Wainwright (2012), we have for some positive constant c'

$$\mathsf{P}(D_m(s) \geq \delta_{\min}/54) \leq 2\exp\Big(-c'n\min\{\frac{\delta_{\min}^2}{\sigma^4}, \frac{\delta_{\min}}{\sigma^2}\} + 2s\log p\Big)$$

Now we choose $s := \frac{1}{c}\frac{n}{\log p}\min\big\{\frac{\delta_{\min}^2}{\sigma^4}, 1\big\}$. It is guaranteed that for sufficiently small c, we

have $s \geq 1$ and

$$\mathsf{P}(D_m(s) \geq \delta_{\min}/54) \leq 2\exp\Big(- c_2 n \min\big\{\frac{\delta_{\min}^2}{\sigma^4}, 1\big\}\Big)$$

This concludes the proof for lower- and upper-RE conditions. □

Chapter 4

A Unified Sparse Learning Framework for Lipschitz Loss Functions

4.1 Introduction

High-dimensional data has emerged in various research fields such as human genetics, neuroimaging, and microbiome studies. When the number of features in the data becomes larger than the sample size or even increases exponentially with the sample size , the traditional regression models would fail to provide an estimation for the regression coefficients, and the theoretical large sample results would not apply. In order to accommodate the ultra high number of features in the regression framework, a series of penalized methods have been proposed. These methods assume that there are only a small set of features contributing to the outcome variable, thus the regression coefficients only include very few nonzero elements. The number of truly nonzero regression coefficients is usually assumed to be small, and not increase with the total number of features and the sample size. Famous example of the sparse learning methods include Lasso with convex L_1 penalty (Tibshirani, 1996) and the closely related Dantzig selector (Bickel et al., 2009), and the non-convex type

of methods such as the smoothly clipped absolute deviation (SCAD) (Fan and Li, 2001) and the minimax concave penalty (MCP) (Zhang, 2010).

For these penalized methods, the sparse assumption on the regression coefficients have enabled establishment of nice theoretical results. Agarwal et al. (2010) studied the convex optimization with norm-based penalties including Lasso and derived the finite sample statistical and optimization error bounds for the estimators. Fan and Lv (2011) investigated the performance of specific nonconvex penalties including SCAD and MCP in ultrahigh dimensions and showed them to possess the oracle property under mild assumptions, in the context of generalized linear models. In addition to modeling continuous outcomes, binary classification is also of vital importance in high-dimensional data analysis. Logistic loss and hinge loss of the support vector machine (SVM) are among the most commonly used loss functions in sparse learning of classification problem. Peng et al. (2016) derived the finite sample statistical error bounds for L_1-norm SVM and further showed the oracle property of non-convex penalized SVM when using their L_1-norm SVM as initial value. There are also developments that applied Lasso and the Dantzig selector to the generalized linear models such as that in the original Lasso paper (Tibshirani, 1996) and the generalized Dantzig selector (GDS) in James and Radchenko (2009).

A more recent line of work has extended penalized sparse learning approaches to the case of high dimensional covariates with measurement errors. Due to various reasons like technical limitations and experimental design, measurement errors are prevalent in the real applications (Raser and O'shea, 2005; Liu, 2016). If we ignore the problem and continue using the contaminated features in the learning process, we have the risk of admitting many false positive signals (Sørensen et al., 2015) and attenuating the estimated coefficients to the null (Carroll and Stefanski, 1994). Typically in high-dimensional settings, one uses corrected versions of the objective functions in order to tackle the measurement error in covariates that require availability of additional validation samples in order to compute moments of the noise distribution. See for example, recent work by Loh et al. (2012); Rosenbaum and Tsybakov (2013); Datta et al. (2017) and more recent work involving grouped penalties in the presence of noisy imaging features (Ma and Kundu, 2021). However, these existing

methods may not be suitable in scenarios where replicated samples are limited, or simply unavailable. Another important limitation is that these existing approaches have primarily been developed for linear models, but there are very limited (if any) approaches for dealing with high-dimensional covariates with measurement error for a wider class of loss functions involving classification or quantile regression that are commonly encountered in practice.

In contrast to the methods that rely on knowing the noise distribution, there are some more recent work on accommodating measurement errors in linear models without requirement the knowledge of noise distributions or the requirement of validation samples. Examples include the matrix uncertainty selector (MUS) (Rosenbaum and Tsybakov, 2010) that is motivated by the Dantzig selector, sparse total least squares (Zhu et al., 2011), and the orthogonal matching pursuit (Chen and Caramanis, 2013). Of particular note in this line of work, is the seminal work by Rosenbaum and Tsybakov (2010) where the authors proposed an MUS estimator that depends on the construction of high confidence set that is guaranteed to contain the true parameters and then define a sparse estimator belonging to this confidence set that satisfies certain properties. Later Sørensen et al. (2018) proposed the generalized MUS (GMUS) which extended the MUS to accommodate generalized linear models (GLM). However, this heuristic approach did not provide any asymptotic or finite sample theory on the statistical errors. In general, while the above set of approaches are useful, they are not directly applicable to loss functions lying outside of the GLM family, such as the hinge loss, that is routinely used in classification problems.

Given the importance of classification and quantile regression problems in literature that go beyond linear regression settings, it is extremely important to develop scalable and theoretically guaranteed estimators for these class of problems. Moreover in order to make these methods scalable and readily applicable to a wide class of problems, it is desirable to avoid requiring additional validation samples for computing moments of the noise distributions. While there is extensive literature on linear regression approaches with measurement error, and some work on measurement error models in GLM frameworks, there is a paucity of relevant measurement error models in high dimensions for more general loss functions that go beyond these settings.

In order to bridge these gaps in the literature, we propose in this chapter a unified sparse learning framework for the class of Lipschitz loss functions in the ultra high-dimensional setting with built-in strategy for measurement errors. This class of loss functions include widely used class of loss functions such as logistic loss, hinge loss for binary classification, quantile regression loss, and various smoothed versions of these losses, and was investigated in Dedieu (2019) without considering the scenario of high-dimensional covariates with measurement errors. Motivated by the line of work by Rosenbaum and Tsybakov (2010) and the considerations mentioned earlier, we define a sparse estimator in terms of L_1 norm on a confidence set based on the gradient of the empirical loss function where the true regression coefficients reside in with high probability. With mild assumptions that are also proved to hold with high probability, we derive the finite sample statistical error bounds and sign consistency results for the proposed estimator. We develop a Newton-Raphson type algorithm with linear programming to handle the computation of the sparse estimator. We also conduct extensive simulation experiments and illustrate the superior performance of our proposed estimator over other competing methods in a variety of settings.

We make several significantly novel contributions in this article. (i) develop an unifying penalized regression approach for high-dimensional covariates with measurement error under a broad class of loss functions; (ii) derive theoretical guarantees for the proposed estimators; (iii) develop an efficient computational algorithm for implementation; (iv) evaluate the numerical performance of the approach for certain special classes of models such as the SVM that is perhaps one of the most widely used classifiers. To our knowledge, we are one of the first to propose a penalized SVM approach for high-dimensional covariates subject to measurement error.

We give the outline of the rest of the chapter here. In Section 4.2, we define our model framework and present the theoretical results. In Section 4.3, we develop the computational algorithm and provide advice on parameter tuning and initialization. In Section 4.4, we compare our proposed method to other competing methods over a series of simulation scenarios. In Section 4.5, we apply the methods to a real dataset to classify children with attention deficit hyperactivity disorder (ADHD) and children as neurotypical controls and

find the most related functional connectivities. Finally in Section 4.6, we include the details of proving the theoretical results.

4.2 Proposed Method with Lipschitz Losses

We consider a dataset with independent samples $\{(\boldsymbol{x}_i, y_i) \in \mathbb{R}^p \times \mathcal{Y}, i = 1, \cdots, n\}$, where the relationship between the outcome $y \in \mathcal{Y}$ and covariates $\boldsymbol{x}$ are defined by a pre-specified class of loss functions denoted by $f(\cdot, y)$. As elaborated below, the class of general loss functions considered in this article can encompass both binary classification and quantile regression problems, which corresponds to y being discrete or continuous respectively. With a fixed loss function $f(\cdot, y)$, we define the coefficient $\boldsymbol{\beta}^*$ as the one that minimizes the theoretical version of the empirical loss $\mathcal{L}(\boldsymbol{\beta}) = (1/n)\sum_{i=1}^{n} f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}\rangle; y_i)$ as $\boldsymbol{\beta}^* = \text{argmin}_{\boldsymbol{\beta}}\{E(\mathcal{L}(\boldsymbol{\beta}))\} = \text{argmin}_{\boldsymbol{\beta}}\{E(f(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle; y))\}$ noted that the expectation is over the joint distribution of $\boldsymbol{x}$ and y. Our goal is to study the theoretical and empirical properties under such loss functions under high-dimensional settings, and for cases encompassing measurement error on the observed covariates, that is often encountered in practice. The high-dimensional settings considered in our work corresponds to ultra-high dimensions where the number of covariates p is allowed to grow exponentially with the sample size n such that $\log(p)/n$ goes towards zero with increasing n and p. Such settings are typically encountered in our motivating neuromaging applications, where the number of voxels in a brain image can be orders of magnitudes higher than the sample size, or brain network-based analysis where the number of candidate edges in the network grow quadratically with the number of brain regions and vastly exceed the sample size. In such high dimensional settings, we typically assume that the true coefficient $\boldsymbol{\beta}^*$ to be sparse, which is routinely done in literature. This implies that the number of non-zero elements in $\boldsymbol{\beta}^*$ (denoted as k) is usually much smaller compared to p and n. Under this set-up our goal is to find a sparse estimator $\hat{\boldsymbol{\beta}}$ that is close to the true coefficient $\boldsymbol{\beta}^*$ for a general class of loss functions that satisfy certain reasonable conditions including Lipschitz continuity.

The class of loss functions considered in our paper is assumed to be Lipschitz continuous

that admits first and second-order derivatives as elaborated below. As illustrated in the sequel, this class of loss functions involve several commonly used losses in literature including logistic regression, smooth hinge loss, and quantile regression.

Definition 4.2.1. *A non-negative, convex loss function $f(\cdot, y)$ is L-Lipschitz continuous if*

$$|f(t_1, y) - f(t_2, y)| \leq L|t_1 - t_2|, \forall t_1, t_2,$$

and there exists first-order derivative function $\partial f(\cdot, y)$ such that

$$f(t_2, y) - f(t_1, y) \geq \partial f(t_1, y)(t_2 - t_1), \forall t_1, t_2.$$

In addition, the loss function is twice-differentiable and admits a second-order derivative function $\partial^2 f(\cdot, y)$.

We note that the derivatives are with respect to the first argument of the loss function. Below we present three example loss functions that satisfy the definition above.

1. Logistic regression: The logistic loss satisfies Definition 4.2.1 with $L = 1$, where $\mathcal{Y} = \{0, 1\}$ with $\log(P(y_i = 1|\boldsymbol{x} = \boldsymbol{x}_i)) - \log(P(y_i = 0|\boldsymbol{x} = \boldsymbol{x}_i)) = \langle \boldsymbol{x}_i, \boldsymbol{\beta} \rangle$. The loss function takes the form $f(t; y) = -yt + \log\{1 + e^t\}$ with first-order derivative $\partial f(t; y) = -y + e^t/(1 + e^t)$ and second-order derivative $\partial^2 f(t; y) = e^t/(1 + e^t)^2$, so that $|\partial f(t; y)| < 1$ and $\partial^2 f(t; y) > 0$.

2. Smooth hinge loss: The smooth hinge loss is an adaptive version of the original hinge loss for support vector machine (SVM) and has the advantage of avoiding discontinuity issues encountered in the original hinge loss and admitting first and second-order derivatives due to convexity - see Luo et al. (2021) for more details. Smooth hinge losses have been used in literature for classification problems involving text and document classification (Chang et al., 2008; Hong et al., 2019), and those involving disease and socio-economic status (Lee and Mangasarian, 2001; Wang et al., 2020). We use the form of the smooth hinge loss function as $f(t; y) = \frac{1}{2}(1 - yt) + \frac{1}{2}\sqrt{(1 - yt)^2 + \sigma^2}$, where $\mathcal{Y} = \{-1, 1\}$, $\sigma > 0$. The smooth hinge loss tends to the original hinge loss as $\sigma \to 0$, as evident from Figure 1 in Luo et al. (2021). The smooth hinge loss also satisfies Definition 4.2.1 with $L = 1$

as $|\partial f(t;y)| < 1$ and $\partial^2 f(t;y) > 0$. This can be seen from the first-order derivative that has the expression, $\partial f(t;y) = -\frac{1}{2}y[1 + (1-yt)/\sqrt{(1-yt)^2+\sigma^2}]$, while the second-order derivative is $\partial^2 f(t;y) = \frac{1}{2}\sigma^2/[(1-yt)^2+\sigma^2]^{3/2}$.

3. Quantile regression: We consider the smoothed version of the quantile loss, named conquer loss, which is twice-differentiable and globally convex, which was proposed by He et al. (2021). Here, $\mathcal{Y} = \mathbb{R}$ and the loss function takes the form $f(t;y) = l_h(y-t)$ with $l_h(u) = (\rho_\tau * K_h)(u) = \int_{-\infty}^{\infty} \rho_\tau(v)K_h(v-u)\mathrm{d}v$ where $*$ denotes the convolution operator. Here $\rho_\tau(u) = u\{\tau - \mathbb{1}(u<0)\}$ is the check function, while $K(\cdot)$ represents a kernel function that integrates to one. Several examples of such kernel function have been given in Section 2 of He et al. (2021). The corresponding first- and second-order derivatives of the conquer loss are $\partial f(t;y) = \mathcal{K}_h(t-y) - \tau$ and $\partial^2 f(t;y) = K_h(y-t)$, where $K_h(u) = h^{-1}K(u/h)$, $\mathcal{K}_h(u) = \mathcal{K}(u/h)$ and $\mathcal{K}(u) = \int_{-\infty}^{u} K(v)\mathrm{d}v$. We can see that both $\mathcal{K}_h(t-y)$ and τ are bounded between 0 and 1, thus the first-order derivative is bounded with $|\partial f(t;y)| \leq 1$. In addition, the non-negative kernel function ensures the second-order derivative to be non-negative. Therefore, we know that the conquer loss for the quantile regression satisfies our Definition 4.2.1 with $L = 1$.

While the above three examples of loss functions provide some concrete settings for the proposed approach, we note that our methodology is generally applicable to more general loss functions that satisfy Definition 4.2.1. Starting from a setting that involves covariates without measurement error that has been the main thrust in literature (Dedieu, 2019), we subsequently generalize the proposed method to the case of covariates with additive measurement error. Our treatment of high-dimensional noisy covariates in classification and quantile regression problems is one of the first such results in literature to our knowledge, and is a novel contribution of independent interest. In addition, the proposed approach has several practical advantages including the ability to avoid computing a noise covariance matrix, which is often encountered in regression problems involving high-dimensional features observed with measurement error (Loh et al., 2012; Ma and Kundu, 2021), but can slow down computations in high-dimensional applications and requires replicated datasets. These, and additional aspects of the proposed methodology and theory are elaborated be-

low.

4.2.1 Estimation with Noiseless Predictors

In this section, we propose an estimator for the Lipschitz continuous loss function when we have the uncontaminated predictor $\boldsymbol{x}_i$'s available. Our treatment assumes that the noiseless predictors $\boldsymbol{x}_i$'s are independently distributed as multivariate normal with mean $\mathbf{0}$ and covariance Σ, which is routinely assumed in literature. We denote the gradient of the empirical loss $\mathcal{L}(\boldsymbol{\beta})$ as $S(\boldsymbol{\beta})$ where

$$S(\boldsymbol{\beta}) = (1/n)\sum_{i=1}^{n} \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}\rangle; y_i)\boldsymbol{x}_i.$$

From the definition of the true coefficient $\boldsymbol{\beta}^*$ we know that the expectation of its gradient function is equal to $\mathbf{0}$, which is to say that $E\{S(\boldsymbol{\beta}^*)\} = \mathbf{0}$. Therefore intuitively we expect the gradient $S(\boldsymbol{\beta}^*)$ to be bounded under a certain threshold with probability tending to 1. This motivates us to define the following confidence set $\mathbb{C}$ based on the gradient function: $\mathbb{C} = \{\boldsymbol{\beta} \in \mathbb{R}^p : \|S(\boldsymbol{\beta})\|_\infty \leq \lambda\}$, where λ acts as a pre-determined threshold and $||\cdot||_\infty$ denotes the supremum norm. A suitable choice of λ should ensure that the true coefficient $\boldsymbol{\beta}^*$ lies in this confidence set $\mathbb{C}$ with high probability. This is confirmed in the following Lemma 4.2.1.

Lemma 4.2.1. *Let* $\lambda = \sqrt{\frac{\phi \log p}{n}}$ *and* $\sigma_x^2 = \|\Sigma\|_{op}$ *denotes the spectral norm of matrix* Σ*, then* $\boldsymbol{\beta}^* \in \mathbb{C}$ *with probability at least* $[1 - 2p^{1-(\phi/2L^2\sigma_x^2)}]$.

The parameter ϕ in Lemma 4.2.1 refers to a constant that can be selected on a finite scale. Under our high-dimensional settings where $\log(p)/n$ becomes negligible for increasing n and p, Lemma 4.2.1 suggests that the confidence set only contains those $\boldsymbol{\beta}$ values that encourage the gradient $S(\boldsymbol{\beta})$ to have increasingly tighter bounds in terms of the $||\cdot||_\infty$ norm. Hence, the set of admissible solutions for $\boldsymbol{\beta}$'s belonging to the confidence set $\mathbb{C}$ have a bounded empirical gradient, which is expected to mimic the behavior of the true coefficient $\boldsymbol{\beta}^*$ that satisfies $E\{S(\boldsymbol{\beta}^*)\} = \mathbf{0}$ with high probability. In addition to the proposed estimator belonging to $\mathbb{C}$, another key consideration is sparsity. In other words, it is desirable to select

an estimator with the highest sparsity level, as measured by L_1 norm (represented as $||\cdot||_1$), to prevent the solution from admitting excessive false positive signals and to replicate the behavior of the true coefficient $\boldsymbol{\beta}^*$ that is assumed to be sparse. Following this argument, we define our estimator as:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \in \mathbb{C}}{\operatorname{argmin}} \|\boldsymbol{\beta}\|_1. \tag{4.2.1}$$

Remark 1: The proposed estimator in (4.2.1) can be viewed as a generalization of the Dantzig selector (Candes and Tao, 2007) for linear regression to the case with Lipschitz continuous losses. On closer inspection of the the constraint of the Dantzig selector, we can see that it is exactly the gradient function of the least squares loss of a linear regression model.

Remark 2: The proposed estimator in (4.2.1) is fundamentally different from related work based on Lipschitz continuous loss functions (Dedieu, 2019). These existing methods were focused on minimizing the empirical loss functions under sparse estimators for the coefficients $\boldsymbol{\beta}$ using suitable penalties. In contrast, our treatment using confidence sets provides a more intuitive approach based on the behavior of gradients, and is more closely aligned with the related methods in Rosenbaum and Tsybakov (2010, 2013).

We denote the difference between our estimator and the true coefficient as $\hat{\boldsymbol{h}} = \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*$. It turns out that the difference $\hat{\boldsymbol{h}}$ can be proved to lie in a cone set $\mathcal{H}$, as detailed in the following Lemma 4.2.2. The cone set condition ensures that the norm of the estimated coefficients corresponding to truly zero effects ($\beta_j^* = 0$) is bounded above by the difference between the estimated and true non-zero coefficients. When the number of truly non-zero features (k) is small (in comparison to p), and assuming that all true coefficients are bounded, this upper bound is a tractable quantity that is bounded for all estimated coefficients belonging to the confidence set $\mathbb{C}$. Hence the following result on the cone set ensures that the number of false positives is bounded, and it will form the basis of our main error bound result in the sequel (Theorem 4.2.1). This concept of cone set is also crucial in defining the restricted strong convexity condition following Lemma 4.2.2. Usually when the

number of total predictors p far exceeds the sample size n, there is no guarantee of strong convexity on all directions as the Hessian matrix is singular. However, if we are able to define the strong convexity on some restricted directions that are concerned to our problem, which is the cone set $\mathcal{H}$ for our case, then we can still derive nice theoretical property on our estimator. More discussions can be found in Negahban et al. (2012).

Lemma 4.2.2. *Assume $\boldsymbol{\beta}^*$ is k-sparse and lies in the confidence set $\mathbb{C}$. Define set $s = \{j \in \{1, \cdots, n\} : \beta_j^* \neq 0\}$ and complement set $s^c = \{j \in \{1, \cdots, n\} : \beta_j^* = 0\}$, then $\boldsymbol{h} = \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^* \in \mathcal{H} = \{\boldsymbol{h} \in \mathbb{R}^p : \|\boldsymbol{h}_{s^c}\|_1 \leq \|\boldsymbol{h}_s\|_1\}$.*

While Lemma 4.2.2 provides some understanding of the behavior of the estimated coefficients in context of the true coefficients $\boldsymbol{\beta}^*$, it is not sufficient to characterize the overall error bounds on its own. In order to guarantee that the estimator $\hat{\boldsymbol{\beta}}$ lies in the neighborhood of the true coefficient $\boldsymbol{\beta}^*$, we also need the following restricted strong convexity (RSC) condition to hold with high probability on the cone set defined in the above Lemma. This condition is needed to ensure that the loss function is not too flat in the restricted cone set such that a closeness in the values of the loss function corresponding to the true and estimated coefficients translates to tight error bounds. With the theoretical results from Raskutti et al. (2010), we can prove this condition to hold with high probability under certain assumptions, as detailed in Lemma 4.2.3 in the sequel. First we formulate the RSC condition below, in the absence of measurement errors.

Restricted strong convexity condition without measurement errors: There exists $\tau > 0$ such that for $\boldsymbol{h} \in \mathcal{H}$ we have $\mathcal{L}(\boldsymbol{\beta}^* + \boldsymbol{h}) - \mathcal{L}(\boldsymbol{\beta}^*) - \langle S(\boldsymbol{\beta}^*), \boldsymbol{h}\rangle \geq \tau \|\boldsymbol{h}\|_2^2$.

Definition 4.2.2. *(square root of a matrix) we say $\Sigma^{1/2} = \sqrt{\Sigma}$ is the square root of a positive semidefinite matrix Σ if: (i) $\Sigma^{1/2}$ is positive semidefinite; (ii) $\Sigma^{1/2}\Sigma^{1/2} = \Sigma$.*

Lemma 4.2.3. *Assume that $\min_{i, \boldsymbol{h} \in \mathcal{H}} |\partial^2 f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \boldsymbol{h}\rangle; y)| \geq M_1 > 0$ for $i = 1, \ldots, n$, and for $Var(\mathbf{x}) = \Sigma$ we define $\kappa_1 = (1/4)\lambda_{\min}(\Sigma^{1/2})$ and $\kappa_2 = 9\sqrt{\max_{j=1,\cdots,p} \Sigma_{jj}}$. If that $n > 4(\kappa_2/\kappa_1)^2 k \log p$, then the RSC condition holds with $\tau = \frac{M_1}{2}\left(\kappa_1 - 2\kappa_2\sqrt{\frac{k \log p}{n}}\right)^2$ with probability at least $(1 - c_1 \exp(-c_2 n))$ for positive constants c_1 and c_2.*

Lemma 4.2.3 states that as long as the number of covariates increases at a rate such that

$\log(p)/n$ is bounded, the RSC condition will hold. We note that the bound on $\log(p)/n$ depends on the ratio κ_2/κ_1 that is related to the condition number encountered in random matrix theory literature (Edelman, 1988), as well as the true sparsity level k.

Remark 3: With the reference to the example loss functions after Definition 4.2.1, the assumption $\min_{i,\boldsymbol{h}\in\mathcal{H}} |\partial^2 f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \boldsymbol{h}\rangle; y)| \geq M_1 > 0$ can be translated into the requirement that the inner product $|\langle \boldsymbol{x}_i, \hat{\boldsymbol{\beta}}\rangle| = |\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \boldsymbol{h}\rangle| < \infty$ for $\boldsymbol{h} \in \mathcal{H}$. We can see that $|\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \boldsymbol{h}\rangle| \leq (\max_j\{x_{ij}\})\|\boldsymbol{\beta}^* + \boldsymbol{h}\|_1$ and $\|\boldsymbol{\beta}^* + \boldsymbol{h}\|_1 = \|\boldsymbol{\beta}^*_s + \boldsymbol{h}_s\|_1 + \|\boldsymbol{\beta}^*_{s^c} + \boldsymbol{h}_{s^c}\|_1 \leq \|\boldsymbol{\beta}^*_s\|_1 + \|\boldsymbol{h}_s\|_1 + \|\boldsymbol{h}_{s^c}\|_1 \leq \|\boldsymbol{\beta}^*_s\|_1 + 2\|\boldsymbol{h}_s\|_1$. As we have assumed $\boldsymbol{x}_i$ to follow multivariate normal distribution, then $\max_j\{x_{ij}\}$ would have a finite amplitude with high probability converging to 1. Moreover, both $\boldsymbol{\beta}^*_s$ and $\boldsymbol{h}_s$ are vector of length k, where k is assumed to be fixed and not growing with n and p, thus their L_1 norm would also have finite amplitude. Combining these arguments, the assumption in Lemma 4.2.3 should stand with high probability converging to 1.

Given the RSC condition and the definition of cone sets, we are now in a position to state our main error bound results. Theorem 4.2.1 provides explicit non-asymptotic error bounds between $\hat{\boldsymbol{\beta}}$ and $\boldsymbol{\beta}^*$ under L_2 and L_1 norms.

Theorem 4.2.1. *Assume $\boldsymbol{\beta}^*$ is k-sparse, then with probability at least $[1 - 2p^{1-(\phi/2L^2\sigma_x^2)} - c_1\exp(-c_2 n)]$ for positive constants c_1 and c_2, the difference between the estimator $\hat{\boldsymbol{\beta}}$ and the true coefficient $\boldsymbol{\beta}^*$ can be bounded in terms of L_1 and L_2 norms as: $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2 \leq \frac{4}{\tau}\sqrt{\frac{\phi k \log p}{n}}$, and $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_1 \leq \frac{8k}{\tau}\sqrt{\frac{\phi \log p}{n}}$*

Remark 4: from Theorem 4.2.1, we can see that as long as $k^2 \log p = o(n)$ where o denotes the small-o notation, then the L_1 and L_2 error bounds would vanish to zero as n and p increase to infinity. This requires that the number of truly nonzero coefficients remains relatively small, while the total number of coefficients p is allowed to increase with n exponentially.

Although Theorem 4.2.1 is able to explicitly establish non-asymptotic error bounds, it is not immediately clear how the proposed approach performs in terms of feature selection. The next result in Theorem 4.2.2 establishes the sign consistency for a thresholded version of the

estimator $\hat{\boldsymbol{\beta}}$, which illustrates that the thresholded estimator is able to accurately identify the truly non-zero coefficients as long they are not exceedingly small. Taken together, the results in Theorems 2.1 and 2.2 establish desirable results on error bounds and sign consistency properties for a general class of Lipschitz continuous loss functions.

Theorem 4.2.2. *(**Sign Consistency**) Let* $\tau_1 = \frac{8k}{\tau}\sqrt{\frac{\phi \log p}{n}}$ *and define a thresholded version of estimator as* $\tilde{\beta}_j = \hat{\beta}_j \mathbf{1}\{|\hat{\beta}_j| > \tau_1\},\ j = 1, \cdots, p$. *If the error bounds in Theorem 4.2.1 hold and we have* $\min_{j \in s} |\beta_j^*| > 2\tau_1$, *then* $sign\tilde{\beta}_j = sign\beta_j^*$.

4.2.2 Estimation with Noisy Predictors

In the case that the observed predictors are contaminated with random noise, usual estimation procedures that do not account for noise lead to unsatisfactory results and inconsistent estimates (Loh et al., 2012; Sørensen et al., 2018; Ma and Kundu, 2021). Based on the above and given the fact that the true coefficient is not guaranteed to be within the confidence set $\mathbb{C}$ defined in Section 2 with high probability in the presence of noise, it is clear that the proposed approach in Section 2 needs to be generalized to accommodate noisy predictors. Also see Section 7 in Rosenbaum and Tsybakov (2010) for more detailed explanations. We illustrate in this section, that a modified confidence set can be constructed, which is guaranteed to contain the true coefficient with high probability even in the presence of measurement error for the covariates.

In this section, we still assume the true predictor $\boldsymbol{x}_i$'s follow independent normal distributions with mean $\mathbf{0}$ and covariance matrix Σ, which we will denote as Σ_x in the following discussion to distinguish from the covariance of the measurement errors. However, the true covariates are unobserved, and instead one observes a contaminated version of the predictor as $\boldsymbol{w}_i$'s where $\boldsymbol{w}_i = \boldsymbol{x}_i + \boldsymbol{u}_i,\ i = 1, \cdots, n$. The $\boldsymbol{u}_i$'s are the measurement errors, which are assumed to independently follow multivariate normal distribution with mean $\mathbf{0}$ and covariance Σ_u. The $\boldsymbol{x}_i$'s and $\boldsymbol{u}_i$'s are assumed to be independent from each other on all indices. The above constructions for the measurement error is routinely assumed in literature involving linear regression models (Rosenbaum and Tsybakov, 2010;

Loh et al., 2012; Ma and Kundu, 2021). Based on the new design matrix, the corresponding empirical loss and gradient functions would be $\mathcal{L}_w(\boldsymbol{\beta}) = (1/n)\sum_{i=1}^n f(\langle \boldsymbol{w}_i, \boldsymbol{\beta}\rangle; y_i)$ and $S_w(\boldsymbol{\beta}) = (1/n)\sum_{i=1}^n \partial f(\langle \boldsymbol{w}_i, \boldsymbol{\beta}\rangle; y_i)\boldsymbol{w}_i$. We make the following assumptions.

(A1) $n\sigma_u^2 = O(1)$ where $\sigma_u^2 = \|\Sigma_u\|_{op}$, and $||\cdot||_{op}$ denotes the operator norm and O is the big-O notation.

(A2) Without loss of generality, we will assume the design matrix $W = (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_n)^T$ has been standardized such that all the diagonal elements in $\frac{W^TW}{n}$ are equal to 1.

(A3) The true coefficient $\boldsymbol{\beta}^*$ satisfies $\boldsymbol{\beta}^* = \text{argmin}\{E(f(\langle \boldsymbol{x}, \boldsymbol{\beta}\rangle; y))\}$.

We note that it is not unreasonable to have assumption *(A1)* for practical applications. In medical imaging analysis, as we collect the image data at finer and finer resolutions, which means more voxels and larger p, we expect the scale of measurement errors to go down. This implies that σ_u^2 is on a negative order of p. Typically the number of samples we collect on is far less compared to the voxel numbers ($p >> n$). Thus the assumption $n\sigma_u^2 = O(1)$ or even $n\sigma_u^2 = o(1)$ (with o denoting the small-o notation) will be applicable in a general sense. Assumption *(A2)* is a common argument in the literature (Fan and Lv, 2011; Rosenbaum and Tsybakov, 2010). Finally, Assumption *(A3)* captures the relationship between the true coefficient $\boldsymbol{\beta}^*$ and the outcome $y \in \mathcal{Y}$ via the loss function that depends on the true (noiseless) predictors $\boldsymbol{x}$.

Similar to the MU-selector (Rosenbaum and Tsybakov, 2010) for linear regression models, we enlarge the feasible band for $\|S_w(\boldsymbol{\beta})\|_\infty$ by allowing the boundary of the feasible set to adapt to the L_1 norm of the coefficients. The modified confidence set $\mathbb{C}^w$ is defined as: $\mathbb{C}^w = \{\boldsymbol{\beta} \in \mathbb{R}^p : \|S_w(\boldsymbol{\beta})\|_\infty \leq \lambda + \gamma\|\boldsymbol{\beta}\|_1\}$. Following the same spirit of the last section, we define our estimator to be the sparsest solution within the confidence set: $\hat{\boldsymbol{\beta}}^w = \text{argmin}_{\boldsymbol{\beta}\in\mathbb{C}^w} \|\boldsymbol{\beta}\|_1$. Similar to Rosenbaum and Tsybakov (2010), we can define an optimization problem as Lasso type analog of this estimator as: $\min_{\boldsymbol{\beta}}\{\mathcal{L}_w(\boldsymbol{\beta}) + \lambda_1\|\boldsymbol{\beta}\|_1 + \lambda_2\|\boldsymbol{\beta}\|_1^2\}$.

Once again, with properly chosen parameters λ and γ, we can show that the true coefficient $\boldsymbol{\beta}^*$ lie in $\mathbb{C}^w$ with high probability, as in the following Lemma 4.2.4. Also, the difference between $\hat{\boldsymbol{\beta}}^w$ and $\boldsymbol{\beta}^*$ lies in the cone set $\mathcal{H}$ in the same manner. Thus with a modified

RSC condition when the measurement errors are present, holding with high probability, we can obtain the error bounds and sign consistency results similar to those in Theorems 4.2.1-4.2.2, as shown in the following Lemma 4.2.5 and Theorems 4.2.3 - 4.2.4.

Lemma 4.2.4. *Let* $\lambda = \sqrt{\frac{\phi_1 \log p}{n}}$ *and* $\gamma = M_2\sqrt{\frac{\phi_2 \log n}{n}}$ *where* $M_2 = \max|\partial^2 f(\cdot; y)|$, *then* $\boldsymbol{\beta}^* \in \mathbb{C}^w$ *with probability at least* $\{1 - 2p^{1-[\phi_1/2L^2(\sigma_x^2+\sigma_u^2)]} - 2n^{(1-\phi_2/2n\sigma_u^2)}\}$.

Remark 5: in general, we can find a finite upper bound of the second-order derivative M_2 for the class of Lipschitz continuous losses we consider. Take the logistic loss for example, it is easy to see that $|\partial^2 f(\cdot; y)| = e^t/(1+e^t)^2 \leq 1/4$. As for the smooth hinge loss, we also have $|\partial^2 f(\cdot; y)| = \frac{1}{2}\sigma^2/[(1-yt)^2+\sigma^2]^{3/2} \leq 1/(2\sigma)$.

Restricted strong convexity condition with measurement errors

There exists $\tau_w > 0$ such that for $\boldsymbol{h} \in \mathcal{H}$ we have

$$\mathcal{L}_w(\boldsymbol{\beta}^* + \boldsymbol{h}) - \mathcal{L}_w(\boldsymbol{\beta}^*) - \langle S_w(\boldsymbol{\beta}^*), \boldsymbol{h}\rangle \geq \tau_w \|\boldsymbol{h}\|_2^2.$$

Lemma 4.2.5. *We denote* $\Sigma^w = \Sigma_x + \Sigma_u$, $\kappa_1^w = (1/4)\lambda_{\min}(\sqrt{\Sigma^w})$ *and* $\kappa_2^w = 9\sqrt{\max_{j=1,\cdots,p}\Sigma_{jj}^w}$. *Assume that for the samples we collect* $\min_{i,\boldsymbol{h}\in\mathcal{H}}|\partial^2 f(\langle \boldsymbol{w}_i, \boldsymbol{\beta}^* + \boldsymbol{h}\rangle; y)| \geq M_1^w > 0$. *If that* $n > 4(\kappa_2^w/\kappa_1^w)^2 k \log p$, *then the RSC condition holds with* $\tau_w = \frac{M_1^w}{2}\Big(\kappa_1^w - 2\kappa_2^w\sqrt{\frac{k\log p}{n}}\Big)^2$ *with probability at least* $(1 - c_1' \exp(-c_2' n))$ *for positive constants* c_1' *and* c_2'.

Remark 6: similar to the arguments after Lemma 4.2.3, the assumption in Lemma 4.2.4 that $\min_{i,\boldsymbol{h}\in\mathcal{H}}|\partial^2 f(\langle \boldsymbol{w}_i, \boldsymbol{\beta}^* + \boldsymbol{h}\rangle; y)| \geq M_1^w > 0$ should hold with high probability as $\boldsymbol{w}_i$'s also follow multivariate normal distribution.

Theorem 4.2.3. *Assume* $\boldsymbol{\beta}^*$ *is* k*-sparse, then with probability at least* $\{1-2p^{1-[\phi_1/2L^2(\sigma_x^2+\sigma_u^2)]} - 2n^{(1-\phi_2/2n\sigma_u^2)} - c_1' \exp(-c_2' n)\}$ *for positive constants* c_1' *and* c_2', *the difference between* $\hat{\boldsymbol{\beta}}^w$ *and* $\boldsymbol{\beta}^*$ *can be bounded in terms of* L_1 *and* L_2 *norms:*

$$\|\hat{\boldsymbol{\beta}}^w - \boldsymbol{\beta}^*\|_2 \leq \frac{4\sqrt{k}}{\tau_w}\Big(\sqrt{\frac{\phi_1 \log p}{n}} + M_2\sqrt{\frac{\phi_2 \log n}{n}}\|\boldsymbol{\beta}^*\|_1\Big)$$

$$\|\hat{\boldsymbol{\beta}}^w - \boldsymbol{\beta}^*\|_1 \leq \frac{8k}{\tau_w}\Big(\sqrt{\frac{\phi_1 \log p}{n}} + M_2\sqrt{\frac{\phi_2 \log n}{n}}\|\boldsymbol{\beta}^*\|_1\Big)$$

Theorem 4.2.4. ***(Sign Consistency)*** *Let* $\tau_1^w = \frac{8k}{\tau}\Big(\sqrt{\frac{\phi_1 \log p}{n}} + aM_2\sqrt{\frac{\phi_2 \log n}{n}}\Big)$ *where* $\|\beta^*\|_1 \leq a$ *and define a thresholded version of estimator as*

$$\tilde{\beta}_j^w = \hat{\beta}_j^w \mathbf{1}\{|\hat{\beta}_j^w| > \tau_1^w\},\ j = 1, \cdots, p$$

If the error bounds in Theorem 4.2.3 hold and $\min_{j\in s}|\beta_j^*| > 2\tau_1^w$*, then* $sign\tilde{\beta}_j^w = sign\beta_j^*$*.*

4.3 Computations

4.3.1 Computational Algorithms

We propose to utilize Newton-Raphson type of method with first-order approximation to the gradient function at each iteration. We assume $\boldsymbol{\beta}^{(m)}$ to be our estimate at the m-th iteration. Then through first-order Taylor expansion, we can obtain an approximation of the gradient function around $\boldsymbol{\beta}^{(m)}$ as

$$S(\boldsymbol{\beta}) \approx S(\boldsymbol{\beta}^{(m)}) + \frac{\partial S(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(m)}}(\boldsymbol{\beta} - \boldsymbol{\beta}^{(m)}) = \Sigma^{(m)}\boldsymbol{\beta} + \nu^{(m)}$$

where $\Sigma^{(m)} = \frac{\partial S(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(m)}} = \frac{1}{n}\sum_{i=1}^n [\partial^2 f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^{(m)}\rangle; y_i)]\boldsymbol{x}_i\boldsymbol{x}_i^T$ and $\nu^{(m)} = S(\boldsymbol{\beta}^{(m)}) - \Sigma^{(m)}\boldsymbol{\beta}^{(m)}$. Using this approximation, the computation turns into the problem of minimizing $\|\boldsymbol{\beta}\|_1$ for $\boldsymbol{\beta} \in \{\boldsymbol{\beta} \in \mathbb{R}^p : \|\Sigma^{(m)}\boldsymbol{\beta} + \nu^{(m)}\|_\infty \leq \lambda\}$, which translates into a linear programming that can be solved by standard software. We formalize the linear programming problem here:

$$\begin{aligned} &\min_{\boldsymbol{b}^+, \boldsymbol{b}^-}\ \mathbf{1}_p^T(\boldsymbol{b}^+ + \boldsymbol{b}^-) \\ &\textit{such that } \boldsymbol{b}^+ \geq \mathbf{0},\ \boldsymbol{b}^- \geq \mathbf{0}, \\ &\qquad \Sigma^{(m)}(\boldsymbol{b}^+ - \boldsymbol{b}^-) \leq \lambda\mathbf{1}_p - \nu^{(m)}, \\ &\qquad -\Sigma^{(m)}(\boldsymbol{b}^+ - \boldsymbol{b}^-) \leq \lambda\mathbf{1}_p + \nu^{(m)}. \end{aligned}$$

where $\mathbf{1}_p$ denotes the vector of length p with entries all equal to 1. We obtain the estimation of $\boldsymbol{\beta}$ at iteration $m+1$ as $\boldsymbol{\beta}^{(m+1)} = \hat{\boldsymbol{b}}^+ - \hat{\boldsymbol{b}}^-$. Then we just repeat this process until convergence.

As for the computation with noisy predictors, we need the approximation of the modified score function $S_w(\boldsymbol{\beta})$ around $\boldsymbol{\beta}^{(m)}$ such that

$$S_w(\boldsymbol{\beta}) \approx S_w(\boldsymbol{\beta}^{(m)}) + \left.\frac{\partial S_w(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(m)}} (\boldsymbol{\beta} - \boldsymbol{\beta}^{(m)}) = \Sigma_w^{(m)}\boldsymbol{\beta} + \nu_w^{(m)}$$

where $\Sigma_w^{(m)} = \left.\frac{\partial S_w(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(m)}} = \frac{1}{n}\sum_{i=1}^n [\partial^2 f(\langle \boldsymbol{w}_i, \boldsymbol{\beta}^{(m)}\rangle; y_i)]\boldsymbol{w}_i\boldsymbol{w}_i^T$ and $\nu_w^{(m)} = S_w(\boldsymbol{\beta}^{(m)}) - \Sigma_w^{(m)}\boldsymbol{\beta}^{(m)}$. And the computation turns into the problem of minimizing $\|\boldsymbol{\beta}\|_1$ for $\boldsymbol{\beta} \in \{\boldsymbol{\beta} \in \mathbb{R}^p : \|\Sigma_w^{(m)}\boldsymbol{\beta} + \nu_w^{(m)}\|_\infty \le \lambda + \gamma\|\boldsymbol{\beta}\|_1\}$, which can also be formalized as a linear programming problem:

$$\begin{aligned}
&\min_{\boldsymbol{b}^+, \boldsymbol{b}^-} \ \mathbf{1}_p^T(\boldsymbol{b}^+ + \boldsymbol{b}^-) \\
&\textit{such that } \boldsymbol{b}^+ \ge \mathbf{0},\ \boldsymbol{b}^- \ge \mathbf{0}, \\
&\qquad (\Sigma_w^{(m)} - \gamma J_p)\boldsymbol{b}^+ - (\Sigma_w^{(m)} + \gamma J_p)\boldsymbol{b}^- \le \lambda\mathbf{1}_p - \nu_w^{(m)}, \\
&\qquad -(\Sigma_w^{(m)} + \gamma J_p)\boldsymbol{b}^+ + (\Sigma_w^{(m)} - \gamma J_p)\boldsymbol{b}^- \le \lambda\mathbf{1}_p + \nu_w^{(m)}.
\end{aligned}$$

where J_p denotes the p by p square matrix with entries all equal to 1.

4.3.2 Parameter Tuning and Initialization

It is critical to tune the model parameters in order for the algorithm to perform optimally in practice. For the parameters λ and γ involved in the confidence set, we can conduct a cross validation (CV) procedure on a selected grid of candidate values. The evaluation criterion for the CV procedure can either be the mis-classification rate on the testing samples, or the deviance for likelihood-based loss functions, such as the logistic loss.

There is one specific parameter, σ^2, for the smooth hinge loss function. When σ^2 is equal to 0, the smooth hinge loss is exactly the original hinge loss. As σ^2 increases, the corresponding smooth hinge loss function gets further away from the original hinge loss. If σ^2 becomes too

large, it would dominant the loss function and cause it to lose distinguishing power. On the other hand, if σ^2 gets too close to 0 and resembles the original hinge loss, it would be very unstable when the inner product of $\boldsymbol{x}$ and $\boldsymbol{\beta}$ is around zero, which can lead to ill condition for the linear programming algorithm in the computation. Given all these considerations, we want to pick a value for σ^2 which is neither too large nor too small. We choose σ^2 equal to 1, 2 and 4 in our empirical experiments, which provide stable estimations.

A warm start for the computational algorithm is also critical here for our methods. In our empirical experiments, we have found that the algorithm may not converge to a fixed point but rather jump between two positions if we start the algorithm from a random point, such as a vector of all zeros. A good starting point is not difficult to obtain for the binary classification problem. We can use the estimate from the lasso regression, or the generalized Dantzig selector.

4.4 Simulations

We examine the empirical performance of our proposed method in several different scenarios of binary classification problem that is subject to measurement errors, and also compare with other competing methods including logistic regression with L_1-norm penalty ('L1logistic'), L_1-norm SVM, GDS and GMUS. These competing methods are realized through `R` packages including `glmnet` (Friedman et al., 2010), `penalizedSVM` (Becker et al., 2009) and `hdme` (Sørensen et al., 2018). We denote our proposed methods as matrix uncertainty classifier (MUC). We test MUC with logistic loss ('MUC.logistic') and with smooth hinge loss at different σ^2 settings of 1 and 4 ('MUC.smhinge1', 'MUC.smhinge4').

We consider three data generation schemes, which are presented in details below.

- Scheme 1: $P(y = 1) = P(y = 0) = 0.5$, $\boldsymbol{x}|(y = 1) \sim \mathrm{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, $\boldsymbol{x}|(y = 0) \sim \mathrm{N}_p(-\boldsymbol{\mu}, \boldsymbol{\Sigma})$, $\boldsymbol{\mu} = (0.1, 0.2, 0.3, 0.4, 0.5, 0, \cdots, 0)^T \in \mathbb{R}^p$, $\boldsymbol{\Sigma} = (\sigma_{ij})$ where $\sigma_{ij} = 1$ for $i = j$, $\sigma_{ij} = -0.2$ for $1 \leq i \neq j \leq 5$ and $\sigma_{ij} = 0$ otherwise. Set $\boldsymbol{\beta}^0 = (1.39, 1.47, 1.56, 1.65, 1.74, 0, \cdots, 0)^T \in \mathbb{R}^p$ such that the Bayes rule of $\mathrm{sign}(\boldsymbol{x}^T\boldsymbol{\beta}^0)$

has Bayes error 6.3%.

- Scheme 2: $\boldsymbol{x} \sim \mathrm{N}_p(\mathbf{0}_p, \boldsymbol{\Sigma})$, $\boldsymbol{\Sigma} = (\sigma_{ij})$ where $\sigma_{ij} = 0.4^{|i-j|}$ for $1 \leq i, j \leq p$, $\boldsymbol{\beta}^0 = (1.1, 1.1, 1.1, 1.1, 1.1, 0, \cdots, 0)^T \in \mathbb{R}^p$, $P(y = 1|\boldsymbol{x}) = \Phi(\boldsymbol{x}^T\boldsymbol{\beta}^0)$ where $\Phi(\cdot)$ denotes the cumulative density function of centered t-distribution with degree of freedom 2.

- Scheme 3: $\boldsymbol{x} \sim \mathrm{N}_p(\mathbf{0}_p, \boldsymbol{\Sigma})$, $\boldsymbol{\Sigma} = (\sigma_{ij})$ where $\sigma_{ij} = 0.4^{|i-j|}$ for $1 \leq i, j \leq p$, $\boldsymbol{\beta}^0 = (1.1, 1.1, 1.1, 1.1, 1.1, 0, \cdots, 0)^T \in \mathbb{R}^p$, $P(y = 1|\boldsymbol{x}) = [1 + \exp(-\boldsymbol{x}^T\boldsymbol{\beta}^0)]^{-1}$.

Scheme 1 and 2 are adapted from experiments in Peng et al. (2016). Scheme 1 resembles the linear discriminate analysis (LDA). Scheme 2 is based on a robit regression framework (Liu, 2004), while Scheme 3 is in a typical logistic regression setting. In all three schemes, the predictors are designed to be correlated to certain degree. We also generate the error-prone predictors as $\boldsymbol{w} = \boldsymbol{x} + \boldsymbol{u}$ where $\boldsymbol{u} \sim \mathrm{N}_p(\mathbf{0}_p, \sigma_u^2\mathrm{I}_p)$. All methods use the error-prone predictors in the estimation procedure instead of the true predictors $\boldsymbol{x}$'s. We set the training and testing sample sizes both at $n = 100$ and the total number of predictors at $p = 1000$. We consider four levels of standard deviation σ_u for the measurement errors at 0.3, 0.4, 0.5 and 0.6.

The performance of the competing methods are evaluated by variable selection accuracy measured by number of false negatives ('β.FN') and number of false positives ('β.FP') and estimation error in L_1 norm ('β.L1err') on the regression coefficients, as well as the Matthews correlation coefficient ('y.MCC'), classification accuracy ('y.Accu'), sensitivity ('y.SE'), specificity ('y.SP'), precision ('y.Prec'), recall ('y.Recall'), F1 score ('y.F1') on the testing samples.

Tables 4.1 through 4.3 summarize the results from our proposed methods and all the competing methods evaluated by the performance metrics. In general, the methods ignoring the measurement errors including 'L1logistic', L_1-norm SVM and GDS have worse performance compared to the methods with noise correction strategy including GMUS and our MUC methods. The difference is most obvious on the number of false positives and L_1-norm error in the estimated regression coefficients, which has been supported by the fact that there is risk of admitting many false positives when ignoring measurement errors in the es-

timation (Sørensen et al., 2015). On the other hand, among the methods with strategy on noise correction, our MUC method with smooth hinge loss has generally better performance compared to GMUS method, and even advantage over MUC method with logistic loss in cases including Scheme 3 where the data generation follows logistic regression. When the data is generated with a different framework, like LDA in Scheme 1 or robit regression in Scheme 2, then the MUC with smooth hinge loss is shown to be more flexible and able to provide better performance.

Table 4.1: Results from Simulation Scheme 1 with LDA data generation

	β.FN	β.FP	β.L1err	y.MCC	y.Accu	y.SE	y.SP	y.Prec	y.Recall	y.F1
	$\sigma_u = 0.3$									
L1logistic	1	15	8.651	**0.476**	0.734	0.713	0.753	0.751	0.713	0.722
L1SVM	1	34	9.464	0.368	0.684	0.673	0.691	0.694	0.673	0.678
GDS	2	9	7.576	0.467	0.728	0.690	**0.764**	**0.758**	0.690	0.708
GMUS	2	7	**7.478**	0.449	0.716	0.681	0.750	0.747	0.681	0.685
MUC.logistic	2	2	7.513	0.465	0.733	0.728	0.737	0.734	0.728	0.729
MUC.smhinge1	2	2	7.526	**0.476**	**0.739**	**0.731**	0.745	0.741	**0.731**	**0.734**
MUC.smhinge4	2	2	7.527	0.457	0.729	0.726	0.732	0.731	0.726	0.727
	$\sigma_u = 0.4$									
L1logistic	2	8	8.033	0.435	0.713	0.719	0.708	0.712	0.719	0.706
L1SVM	2	36	9.665	0.305	0.649	0.634	0.669	0.650	0.634	0.636
GDS	2	7	7.604	0.428	0.707	0.713	0.702	0.710	0.713	0.697
GMUS	2	5	**7.463**	**0.441**	0.711	0.716	0.705	**0.720**	0.716	0.700
MUC.logistic	3	0	7.567	0.401	0.701	0.706	0.696	0.698	0.706	0.698
MUC.smhinge1	2	2	7.511	0.433	0.716	**0.722**	0.712	0.707	**0.722**	**0.712**
MUC.smhinge4	2	1	**7.463**	0.433	**0.717**	0.706	**0.728**	0.713	0.706	0.708
	$\sigma_u = 0.5$									
L1logistic	2	18.0	9.205	**0.409**	**0.700**	0.703	**0.697**	**0.694**	0.703	0.686
L1SVM	2	33.5	9.737	0.290	0.644	0.662	0.628	0.626	0.662	0.638
GDS	2	10.5	7.812	0.376	0.675	0.721	0.639	0.678	0.721	0.687
GMUS	2	4.0	7.663	0.384	0.678	**0.724**	0.644	0.682	**0.724**	0.683
MUC.logistic	3	1.5	7.641	0.394	0.699	0.710	0.685	0.681	0.710	**0.691**
MUC.smhinge1	3	2.0	7.897	0.390	0.694	0.699	0.693	0.677	0.699	0.685
MUC.smhinge4	3	0.5	**7.631**	0.389	0.692	0.713	0.677	0.675	0.713	0.689
	$\sigma_u = 0.6$									
L1logistic	2	16	9.133	0.324	0.657	0.653	0.660	0.671	0.653	0.647
L1SVM	2	37	9.900	0.241	0.621	0.602	0.636	0.631	0.602	0.611
GDS	2	11	8.039	0.351	0.672	0.638	**0.697**	**0.703**	0.638	0.653
GMUS	2	7	7.772	0.335	0.661	0.618	0.692	0.700	0.618	0.635
MUC.logistic	3	2	7.770	0.346	0.673	0.680	0.666	0.678	0.680	0.675
MUC.smhinge1	3	2	**7.734**	0.359	0.679	0.674	0.685	0.683	0.674	0.676
MUC.smhinge4	3	1	7.860	**0.372**	**0.685**	**0.687**	0.685	0.689	**0.687**	**0.686**

Table 4.2: Results from Simulation Scheme 2 with robit data generation

	β.FN	β.FP	β.L1err	y.MCC	y.Accu	y.SE	y.SP	y.Prec	y.Recall	y.F1
	$\sigma_u = 0.3$									
L1logistic	0	6	5.137	0.536	0.763	**0.781**	0.749	0.767	**0.781**	0.766
L1SVM	1	33	7.055	0.409	0.704	0.710	0.699	0.705	0.710	0.704
GDS	0	6	4.863	0.539	0.762	0.770	0.760	**0.780**	0.770	0.762
GMUS	1	3	4.638	**0.545**	0.763	0.779	0.752	0.779	0.779	0.767
MUC.logistic	1	1	4.802	0.531	0.765	0.764	**0.768**	0.770	0.764	0.765
MUC.smhinge1	1	1	4.764	0.527	0.763	0.762	0.766	0.768	0.762	0.762
MUC.smhinge4	1	1	**4.507**	0.540	**0.769**	0.774	0.767	0.772	0.774	**0.770**
	$\sigma_u = 0.4$									
L1logistic	0	3.0	5.179	0.502	0.744	0.728	**0.764**	0.758	0.728	0.730
L1SVM	1	33.0	7.105	0.331	0.665	0.659	0.671	0.659	0.659	0.655
GDS	1	3.0	4.747	0.520	0.752	0.747	0.762	0.765	0.747	0.742
GMUS	1	0.5	4.736	0.512	0.745	0.732	0.763	**0.769**	0.732	0.731
MUC.logistic	2	0.0	4.862	0.508	0.754	0.767	0.741	0.740	0.767	0.752
MUC.smhinge1	1	0.0	4.766	**0.522**	**0.760**	**0.775**	0.747	0.746	**0.775**	**0.759**
MUC.smhinge4	2	0.0	**4.594**	0.504	0.752	0.765	0.739	0.737	0.765	0.750
	$\sigma_u = 0.5$									
L1logistic	1	3.0	5.296	0.485	0.735	0.756	0.715	0.741	0.756	0.736
L1SVM	2	34.0	7.296	0.331	0.664	0.674	0.654	0.667	0.674	0.664
GDS	1	5.5	5.018	0.493	0.741	**0.764**	0.717	0.744	**0.764**	0.744
GMUS	1	2.5	4.884	**0.497**	0.742	0.761	0.721	**0.747**	0.761	0.743
MUC.logistic	2	0.0	4.977	0.493	**0.746**	0.759	0.733	0.745	0.759	**0.750**
MUC.smhinge1	2	0.0	4.839	0.491	**0.746**	0.753	**0.737**	0.745	0.753	0.747
MUC.smhinge4	2	0.0	**4.740**	0.488	0.744	0.754	0.734	0.743	0.754	0.746
	$\sigma_u = 0.6$									
L1logistic	1	5	5.361	**0.426**	**0.710**	0.668	0.740	0.735	0.668	0.684
L1SVM	2	34	7.390	0.278	0.639	0.627	0.648	0.637	0.627	0.627
GDS	1	10	5.545	0.413	0.706	0.648	**0.751**	0.735	0.648	0.686
GMUS	2	1	5.148	0.390	0.690	0.615	0.750	**0.744**	0.615	0.658
MUC.logistic	2	0	5.230	0.409	0.705	0.706	0.702	0.697	0.706	0.699
MUC.smhinge1	2	1	5.139	0.418	0.709	**0.719**	0.697	0.698	**0.719**	**0.707**
MUC.smhinge4	2	1	**5.067**	0.406	0.704	0.699	0.706	0.698	0.699	0.696

Table 4.3: Results from Simulation Scheme 3 with logistic data generation

	β.FN	β.FP	β.L1err	y.MCC	y.Accu	y.SE	y.SP	y.Prec	y.Recall	y.F1
	$\sigma_u = 0.3$									
L1logistic	1	10	5.722	0.558	0.774	0.775	0.774	**0.775**	0.775	0.767
L1SVM	1	34	7.283	0.387	0.690	0.703	0.680	0.682	0.703	0.686
GDS	1	5	4.928	**0.566**	0.775	**0.787**	0.764	**0.775**	**0.787**	0.769
GMUS	1	2	4.770	0.556	0.767	0.777	0.759	**0.775**	0.777	0.759
MUC.logistic	2	1	4.857	0.557	**0.778**	0.780	**0.778**	0.769	0.780	**0.773**
MUC.smhinge1	2	0	4.874	0.554	0.776	0.781	0.774	0.767	0.781	0.772
MUC.smhinge4	2	0	**4.624**	0.554	0.776	0.778	0.776	0.768	0.778	0.771
	$\sigma_u = 0.4$									
L1logistic	1.0	3.0	4.992	0.534	0.764	0.751	0.776	0.775	0.751	0.755
L1SVM	1.0	33.0	7.186	0.356	0.678	0.665	0.689	0.678	0.665	0.668
GDS	1.0	7.0	4.972	**0.541**	**0.768**	0.756	**0.777**	**0.777**	0.756	0.758
GMUS	1.0	1.0	4.781	0.526	0.758	0.742	0.773	0.776	0.742	0.746
MUC.logistic	1.5	0.0	4.846	0.515	0.757	0.762	0.754	0.753	0.762	0.755
MUC.smhinge1	2.0	0.5	4.800	0.508	0.754	0.758	0.751	0.750	0.758	0.752
MUC.smhinge4	2.0	0.0	**4.541**	0.530	0.765	**0.769**	0.762	0.761	**0.769**	**0.763**
	$\sigma_u = 0.5$									
L1logistic	1	11	6.009	0.443	0.718	0.705	0.726	0.736	0.705	0.707
L1SVM	2	34	7.370	0.320	0.660	0.642	0.677	0.670	0.642	0.652
GDS	1	9	5.289	0.467	0.729	0.717	0.734	0.748	0.717	0.717
GMUS	1	3	5.056	0.447	0.715	0.701	0.723	0.745	0.701	0.696
MUC.logistic	2	0	5.055	0.463	0.732	0.722	0.740	0.741	0.722	0.730
MUC.smhinge1	2	1	4.901	0.472	0.736	**0.735**	0.736	0.742	**0.735**	0.736
MUC.smhinge4	2	1	**4.872**	**0.482**	**0.741**	0.728	**0.753**	**0.752**	0.728	**0.738**
	$\sigma_u = 0.6$									
L1logistic	1	6	5.566	0.446	0.714	0.728	0.705	0.712	0.728	0.705
L1SVM	1	36	7.343	0.343	0.668	0.689	0.650	0.655	0.689	0.665
GDS	1	7	5.246	**0.452**	0.718	**0.736**	0.704	**0.715**	**0.736**	0.712
GMUS	2	2	5.029	0.419	0.699	0.718	0.683	0.711	0.718	0.688
MUC.logistic	2	0	5.155	0.432	0.716	0.710	0.722	0.705	0.710	0.705
MUC.smhinge1	2	1	5.042	0.441	0.720	0.725	0.716	0.705	0.725	**0.713**
MUC.smhinge4	2	1	**5.019**	0.444	**0.721**	0.714	**0.729**	0.711	0.714	0.710

4.5 Real Data Application

We test the classification performance of our proposed methods and the competing methods using the training and validation datasets from the Connectomics in NeuroImaging Transfer Learning Challenge (CNI-TLC) (Schirmer et al., 2021). Their training dataset includes 100 children diagnosed with attention deficit hyperactivity disorder (ADHD) and another 100 children as neurotypical controls. The validation dataset includes additional 20 children with ADHD and 20 children as controls. We combine these 240 samples and randomly split them into two parts: first part with 216 children's data to train the models and obtain estimated coefficients, second part with the rest 24 children's data as testing samples to evaluate the classification accuracy of the models.

For each child in the training and validation datasets, resting-state functional magnetic resonance imaging (rs-fMRI) time series are provided in the Craddock200 parcellation with 200 regions of interests (ROIs) (Craddock et al., 2012). We first compute the correlation matrix of the rs-fMRI time series for each child. Then the partial correlation matrix is obtained by finding a sparse inverse of the correlation matrix at a certain density level. The sparse inverse matrix is calculated using `R` package `QUIC` with specified ρ parameter (Hsieh et al., 2011). Finally, the vectorized lower diagonal part of the partial correlation matrix, or we call it the edge set, is fed into the models as predictors after removing the edges whose standard deviation across the training dataset is below 0.03.

Table 4.4 summarizes the misclassification rates on the testing samples, where the lowest rate is obtained with the proposed 'MUC.logistic' method and followed closely by the 'MUC.smhinge4' method. Table 4.4 also includes the number of selected edges from all methods. All four edges selected by the 'MUC.smhinge4' as well as the GMUS method are included in the eight selected edges of the 'MUC.logistic' method, and these eight edges are included in the selections of the 'MUC.smhinge1' method. In addition, all of the 15 edges selected by 'MUC.smhinge1' are also selected by the GDS method and the L1SVM method. By observing the extremely large number of selected edges by L1SVM and GDS methods, it is reasonable to assume that they may have admitted excessive false positive edges. Finally,

we note that the 'L1logistic' method has the second highest misclassification rate, and its selection of edges is quite different from all other methods, indicating a bad performance.

Table 4.4: Summary on the CNI-TLC Analysis

	Misclassification rate	*Number of selected edges*
L1logistic	0.375	90
L1SVM	0.375	127
GDS	0.458	79
GMUS	0.292	4
MUC.logistic	0.167	8
MUC.smhinge1	0.250	15
MUC.smhinge4	0.208	4

We further provide the four edges selected by all our proposed methods here in Table 4.5. The atypical functional connectivity from these edges in the ADHD population have been hinted in the literature (Cherkasova and Hechtman, 2009; Vance et al., 2007; Rolls et al., 2021; Fair et al., 2010).

Table 4.5: Four Selected Edges with Region Location Information

Edge information	*Related region information*
Edge between region 32 & 140	Region 32: Temporal pole: middle temporal gyrus, right Region 140: Middle temporal gyrus, right
Edge between region 38 & 132	Region 38: Middle frontal gyrus, right Region 132: Superior parietal gyrus, right
Edge between region 44 & 195	Region 44: Calcarine, right Region 195: Lingual gyrus, right
Edge between region 140 & 166	Region 140: Middle temporal gyrus, right Region 166: Angular gyrus, right

4.6 Appendices

In this section, we will provide the detailed proofs of the theoretical results.

4.6.1 Proof of Lemma 4.2.1

Proof. From the definition of the confidence set $\mathbb{C}$, we know it is equivalent to prove that $P(\|S(\boldsymbol{\beta}^*)\|_\infty > \lambda) \leq 2p^{1-(\phi/2L^2\sigma_x^2)}$. By the union bound, we have that

$$P(\|S(\boldsymbol{\beta}^*)\|_\infty > \lambda) \leq \sum_{j=1}^{p} P\bigg(\frac{1}{n}\bigg|\sum_{i=1}^{n} \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) x_{ij}\bigg| > \lambda\bigg)$$

As $\boldsymbol{x}_i$ follows multivariate normal with mean $\mathbf{0}$ and covariance Σ, we know that x_{ij} is sub-Gaussian with parameter σ_x^2 such that $P(|x_{ij}| > t) \leq 2\exp(-\frac{t^2}{2\sigma_x^2})$. Also from the assumption that loss function $f(\cdot, y)$ is L-Lipschitz continuous, we have that $|\partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i)| \leq L$. Thus

$$P\Big(|\partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) x_{ij}| > \lambda\Big) \leq P\Big(|x_{ij}| > \frac{\lambda}{L}\Big) \leq 2\exp\Big(-\frac{\lambda^2}{2L^2\sigma_x^2}\Big)$$

By the definition of $\boldsymbol{\beta}^*$, we know that $E\{\partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) x_{ij}\} = 0$. Combining these two results we know that $\partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) x_{ij}$ is sub-Gaussian with parameter $L^2\sigma_x^2$. Then with the independence of $\partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) x_{ij}$ for $i = 1, \cdots, n$, we have that

$$P\bigg(\frac{1}{n}\bigg|\sum_{i=1}^{n} \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) x_{ij}\bigg| > \lambda\bigg) \leq 2\exp\Big(-\frac{\lambda^2}{2L^2\sigma_x^2}\Big)$$

Then by the union bound result and that $\lambda = \sqrt{\frac{\phi \log p}{n}}$, we have that

$$P(\|S(\boldsymbol{\beta}^*)\|_\infty > \lambda) \leq 2p\exp\Big(-\frac{\lambda^2}{2L^2\sigma_x^2}\Big) = 2p^{1-(\phi/2L^2\sigma_x^2)}$$

□

4.6.2 Proof of Lemma 4.2.2

Proof. As we assume that $\boldsymbol{\beta}^*$ lies in the confidence set $\mathbb{C}$, then by the definition of $\hat{\boldsymbol{\beta}}$ we have that $\|\hat{\boldsymbol{\beta}}\|_1 \leq \|\boldsymbol{\beta}^*\|_1$. Also we know that $\|\hat{\boldsymbol{\beta}}\|_1 = \|\hat{\boldsymbol{\beta}}_s\|_1 + \|\hat{\boldsymbol{\beta}}_{s^c}\|_1$ and that $\|\boldsymbol{\beta}^*\|_1 = \|\boldsymbol{\beta}^*_s\|_1$. Thus we have $\|\hat{\boldsymbol{\beta}}_s\|_1 + \|\hat{\boldsymbol{\beta}}_{s^c}\|_1 \leq \|\boldsymbol{\beta}^*_s\|_1$ which means that $\|\hat{\boldsymbol{\beta}}_{s^c}\|_1 \leq \|\boldsymbol{\beta}^*_s\|_1 - \|\hat{\boldsymbol{\beta}}_s\|_1$.

From the definition $\hat{\boldsymbol{h}} = \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*$, we know that $\|\hat{\boldsymbol{h}}_{s^c}\|_1 = \|\hat{\boldsymbol{\beta}}_{s^c}\|_1$, and by the triangle

inequality that $\|\hat{\boldsymbol{h}}_s\|_1 = \|\hat{\boldsymbol{\beta}}_s - \boldsymbol{\beta}^*_s\|_1 \geq \|\boldsymbol{\beta}^*_s\|_1 - \|\hat{\boldsymbol{\beta}}_s\|_1$. Thus comparing with the result that $\|\hat{\boldsymbol{\beta}}_{s^c}\|_1 \leq \|\boldsymbol{\beta}^*_s\|_1 - \|\hat{\boldsymbol{\beta}}_s\|_1$, we have $\|\hat{\boldsymbol{h}}_{s^c}\|_1 \leq \|\hat{\boldsymbol{h}}_s\|_1$. □

4.6.3 Proof of Lemma 4.2.3

Proof. By second-order Taylor expansion, we know that

$$\mathcal{L}(\boldsymbol{\beta}^* + \boldsymbol{h}) - \mathcal{L}(\boldsymbol{\beta}^*) - \langle S(\boldsymbol{\beta}^*), \boldsymbol{h}\rangle = \frac{1}{2n}\sum_{i=1}^{n} \partial^2 f\big(\langle \boldsymbol{\beta}^*, \boldsymbol{x}_i\rangle + \nu\langle \boldsymbol{h}, \boldsymbol{x}_i\rangle\big)\langle \boldsymbol{h}, \boldsymbol{x}_i\rangle^2, \quad \nu \in [0,1]$$

With the assumption that $\min_{i,\boldsymbol{h}\in\mathcal{H}} |\partial^2 f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \boldsymbol{h}\rangle; y)| \geq M_1 > 0$, and we also know that $\nu\boldsymbol{h} \in \mathcal{H}$ given $\boldsymbol{h} \in \mathcal{H}$, thus we have

$$\mathcal{L}(\boldsymbol{\beta}^* + \boldsymbol{h}) - \mathcal{L}(\boldsymbol{\beta}^*) - \langle S(\boldsymbol{\beta}^*), \boldsymbol{h}\rangle \geq \frac{M_1}{2}\Big[\frac{1}{n}\sum_{i=1}^{n}\langle \boldsymbol{h}, \boldsymbol{x}_i\rangle^2\Big] = \frac{M_1}{2}\Big(\frac{\|X\boldsymbol{h}\|_2}{\sqrt{n}}\Big)^2$$

where $X = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n)^T$. Based on Theorem 1 of Raskutti et al. (2010),

$$\frac{\|X\boldsymbol{h}\|_2}{\sqrt{n}} \geq \kappa_1\|\boldsymbol{h}\|_2 - \kappa_2\sqrt{\frac{\log p}{n}}\|\boldsymbol{h}\|_1$$

with probability at least $(1 - c_1\exp(-c_2 n))$. Then for $\boldsymbol{h} \in \mathcal{H}$, $\|\hat{\boldsymbol{h}}\|_1 = \|\hat{\boldsymbol{h}}_s\|_1 + \|\hat{\boldsymbol{h}}_{s^c}\|_1 \leq 2\|\hat{\boldsymbol{h}}_s\|_1 \leq 2\sqrt{k}\|\hat{\boldsymbol{h}}_s\|_2 = 2\sqrt{k}\|\hat{\boldsymbol{h}}\|_2$ by the triangle inequality. Thus $\frac{\|X\boldsymbol{h}\|_2}{\sqrt{n}} \geq \Big(\kappa_1 - 2\kappa_2\sqrt{\frac{k\log p}{n}}\Big)\|\boldsymbol{h}\|_2$ for $\boldsymbol{h} \in \mathcal{H}$. Combining with the results above will lead us to the statement. □

4.6.4 Proof of Theorem 4.2.1

Proof. As the loss function is convex and admits first-order derivative, we have that

$$f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \hat{\boldsymbol{h}}\rangle; y_i) - f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) \geq \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \hat{\boldsymbol{h}}\rangle; y_i)\langle \boldsymbol{x}_i, \hat{\boldsymbol{h}}\rangle = \langle \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \hat{\boldsymbol{h}}\rangle; y_i)\boldsymbol{x}_i, \hat{\boldsymbol{h}}\rangle$$

Then we have that

$$\begin{aligned}\mathcal{L}(\boldsymbol{\beta}^* + \hat{\boldsymbol{h}}) - \mathcal{L}(\boldsymbol{\beta}^*) &= \frac{1}{n}\sum_{i=1}^{n} f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \hat{\boldsymbol{h}}\rangle; y_i) - \frac{1}{n}\sum_{i=1}^{n} f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^*\rangle; y_i) \\ &\geq \frac{1}{n}\sum_{i=1}^{n} \langle \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* + \hat{\boldsymbol{h}}\rangle; y_i)\boldsymbol{x}_i, \hat{\boldsymbol{h}}\rangle = \langle S(\boldsymbol{\beta}^* + \hat{\boldsymbol{h}}), \hat{\boldsymbol{h}}\rangle\end{aligned}$$

Then if we assume that $\boldsymbol{\beta}^* \in \mathbb{C}$ and that $\hat{\boldsymbol{h}}$ satisfies the RSC condition, which is with probability at least $[1 - 2p^{1-(\phi/2L^2\sigma_x^2)} - c_1\exp(-c_2 n)]$, then we have that

$$\begin{aligned}\tau\|\hat{\boldsymbol{h}}\|_2^2 &\leq \langle S(\boldsymbol{\beta}^* + \hat{\boldsymbol{h}}), \hat{\boldsymbol{h}}\rangle - \langle S(\boldsymbol{\beta}^*), \hat{\boldsymbol{h}}\rangle \\ &\leq \|S(\boldsymbol{\beta}^* + \hat{\boldsymbol{h}}) - S(\boldsymbol{\beta}^*)\|_\infty \|\hat{\boldsymbol{h}}\|_1 \\ &\leq (\|S(\boldsymbol{\beta}^* + \hat{\boldsymbol{h}})\|_\infty + \|S(\boldsymbol{\beta}^*)\|_\infty)\|\hat{\boldsymbol{h}}\|_1 \qquad (as\ \boldsymbol{\beta}^*, \hat{\boldsymbol{\beta}} \in \mathbb{C}) \\ &\leq 2\lambda\|\hat{\boldsymbol{h}}\|_1\end{aligned}$$

From Lemma 4.2.2 and the triangle inequality, we also have that $\|\hat{\boldsymbol{h}}\|_1 = \|\hat{\boldsymbol{h}}_s\|_1 + \|\hat{\boldsymbol{h}}_{s^c}\|_1 \leq 2\|\hat{\boldsymbol{h}}_s\|_1 \leq 2\sqrt{k}\|\hat{\boldsymbol{h}}_s\|_2 = 2\sqrt{k}\|\hat{\boldsymbol{h}}\|_2$. Combined with the results above, we have that $\tau\|\hat{\boldsymbol{h}}\|_2^2 \leq 2\lambda\|\hat{\boldsymbol{h}}\|_1 \leq 4\sqrt{k}\lambda\|\hat{\boldsymbol{h}}\|_2$. Then we are led to the result that

$$\|\hat{\boldsymbol{h}}\|_2 \leq \frac{4\sqrt{k}\lambda}{\tau} = \frac{4}{\tau}\sqrt{\frac{\phi k \log p}{n}}$$

Applying $\|\hat{\boldsymbol{h}}\|_1 \leq 2\sqrt{k}\|\hat{\boldsymbol{h}}\|_2$ again, we are led to the result that

$$\|\hat{\boldsymbol{h}}\|_1 \leq \frac{8k}{\tau}\sqrt{\frac{\phi \log p}{n}}$$

□

4.6.5 Proof of Theorem 4.2.2

Proof. For $j \in s^c$, we have $\beta_j^* = 0$. Then $|\hat{\beta}_j| = |\hat{\beta}_j - \beta_j^*| < \|\hat{\boldsymbol{h}}\|_1 \leq \tau_1$. Thus $\tilde{\beta}_j = 0$ and we have $sign\tilde{\beta}_j = sign\beta_j^* = 0$.

For $j \in s$, we also have $|\hat{\beta}_j - \beta_j^*| < \|\hat{\boldsymbol{h}}\|_1 \leq \tau_1$. Combined with that $|\beta_j^*| > 2\tau_1$, then we also

have $sign\tilde{\beta}_j = sign\beta_j^*$. □

4.6.6 Proof of Lemma 4.2.4

Proof. By first-order Taylor expansion, we have

$$\partial f(\langle \boldsymbol{w}_i, \boldsymbol{\beta}^* \rangle; y_i) = \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* \rangle; y_i) + \partial^2 f(\langle \boldsymbol{w}_i^*, \boldsymbol{\beta}^* \rangle; y_i) \langle \boldsymbol{u}_i, \boldsymbol{\beta} \rangle$$

where $\boldsymbol{w}_i^* = \boldsymbol{x}_i + \kappa_i \boldsymbol{u}_i$, $\kappa_i \in [0, 1]$. Then we can rewrite $S_w(\boldsymbol{\beta}^*)$ as

$$\begin{aligned} S_w(\boldsymbol{\beta}^*) &= (1/n) \sum_{i=1}^{n} \Big[\partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* \rangle; y_i) + \partial^2 f(\langle \boldsymbol{w}_i^*, \boldsymbol{\beta}^* \rangle; y_i) \langle \boldsymbol{u}_i, \boldsymbol{\beta}^* \rangle \Big] \boldsymbol{w}_i \\ &= (1/n) \sum_{i=1}^{n} \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* \rangle; y_i) \boldsymbol{w}_i + (1/n) \sum_{i=1}^{n} \partial^2 f(\langle \boldsymbol{w}_i^*, \boldsymbol{\beta}^* \rangle; y_i) \langle \boldsymbol{u}_i, \boldsymbol{\beta}^* \rangle \boldsymbol{w}_i \end{aligned}$$

As $\boldsymbol{w}_i$ follows multivariate normal distribution with mean $\mathbf{0}$ and covariance $(\Sigma_x + \Sigma_u)$ with $\|\Sigma_x + \Sigma_u\|_{op} \leq \sigma_x^2 + \sigma_u^2$. Then using similar arguments to those in Lemma 4.2.1, we can have that $P(\|(1/n) \sum_{i=1}^{n} \partial f(\langle \boldsymbol{x}_i, \boldsymbol{\beta}^* \rangle; y_i) \boldsymbol{w}_i\|_\infty > \lambda) \leq 2p^{1-[\phi_1/2L^2(\sigma_x^2+\sigma_u^2)]}$.

We denote $U = (\boldsymbol{u}_1, \cdots, \boldsymbol{u}_n)^T$. Then

$$\Big\| (1/n) \sum_{i=1}^{n} \partial^2 f(\langle \boldsymbol{w}_i^*, \boldsymbol{\beta}^* \rangle; y_i) \langle \boldsymbol{u}_i, \boldsymbol{\beta}^* \rangle \boldsymbol{w}_i \Big\|_\infty \leq M_2 \Big\| \frac{1}{n} \sum_{i=1}^{n} \langle \boldsymbol{u}_i, \boldsymbol{\beta}^* \rangle \boldsymbol{w}_i \Big\|_\infty = M_2 \Big\| \frac{1}{n} W^T U \boldsymbol{\beta}^* \Big\|_\infty$$

Note that by Cauchy-Schwartz inequality and that W has been standardized such that $\Big\| \boldsymbol{w}_{(j)} \Big\|_2 = \sqrt{n}$ for $j = 1, \cdots, p$

$$\begin{aligned} \Big\| \frac{1}{n} W^T U \boldsymbol{\beta}^* \Big\|_\infty &= \frac{1}{n} \max_{1 \leq j \leq p} \Big| \boldsymbol{w}_{(j)}^T U \boldsymbol{\beta}^* \Big| \\ &\leq \frac{1}{n} \|U \boldsymbol{\beta}^*\|_2 \max_{1 \leq j \leq p} \Big\| \boldsymbol{w}_{(j)} \Big\|_2 \\ &= \frac{1}{\sqrt{n}} \|U \boldsymbol{\beta}^*\|_2 \\ &\leq \|U \boldsymbol{\beta}^*\|_\infty \\ &\leq \|U\|_\infty \|\boldsymbol{\beta}^*\|_1 \end{aligned}$$

From the distribution assumption on $\boldsymbol{u}_i$'s and by the union bound, we have that

$$P\left(\|U\|_\infty > \sqrt{\frac{\phi_2 \log n}{n}}\right) \leq \sum_{i=1}^{n} P\left(\|\boldsymbol{u}_i\|_\infty > \sqrt{\frac{\phi_2 \log n}{n}}\right) \leq 2n \exp\left(-\frac{\phi_2 \log n}{2n\sigma_u^2}\right) = 2n^{(1-\phi_2/2n\sigma_u^2)}$$

Putting together the results above, we have $\boldsymbol{\beta}^*$ satisfies $\|S_w(\boldsymbol{\beta}^*)\|_\infty \leq \sqrt{\frac{\phi_1 \log p}{n}} + M_2\sqrt{\frac{\phi_2 \log n}{n}}\|\boldsymbol{\beta}^*\|_1$ with probability at least $\{1 - 2p^{1-[\phi_1/2L^2(\sigma_x^2+\sigma_u^2)]} - 2n^{(1-\phi_2/2n\sigma_u^2)}\}$. □

4.6.7 Proof of Lemma 4.2.5

Proof is similar to that of Lemma 4.2.3 by replacing $\boldsymbol{x}_i$'s with $\boldsymbol{w}_i$'s.

4.6.8 Proof of Theorem 4.2.3

Proof is similar to that of Theorem 4.2.1.

4.6.9 Proof of Theorem 4.2.4

Proof is similar to that of Theorem 4.2.2.

Bibliography

Agarwal, A., Negahban, S., and Wainwright, M. J. (2010). Fast global convergence rates of gradient methods for high-dimensional statistical recovery. Advances in Neural Information Processing Systems, 23.

Agarwal, A., Negahban, S., and Wainwright, M. J. (2012). Fast global convergence of gradient methods for high-dimensional statistical recovery. Ann. Stat., 40(5):2452–2482.

Akoglu, L., Tong, H., and Koutra, D. (2015). Graph based anomaly detection and description: a survey. Data mining and knowledge discovery, 29(3):626–688.

Avants, B. B., Epstein, C. L., Grossman, M., and Gee, J. C. (2008). Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. Med Image Anal, 12(1):26–41.

Avants, B. B., Tustison, N. J., Song, G., et al. (2011). A reproducible evaluation of ants similarity metric performance in brain image registration. Neuroimage, 54(3):2033–2044.

Bassett, D. S. and Bullmore, E. (2006). Small-world brain networks. The neuroscientist, 12(6):512–523.

Becker, N., Werft, W., Toedt, G., Lichter, P., and Benner, A. (2009). penalizedsvm: a r-package for feature selection svm classification. Bioinformatics, 25(13):1711–1712.

Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. Neural computation, 15(6):1373–1396.

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

Breheny, P. and Huang, J. (2015). Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. Stat. Comput., 25:173–187.

Bullmore, E. and Sporns, O. (2009). Complex brain networks: graph theoretical analysis of structural and functional systems. Nature reviews neuroscience, 10(3):186.

Candes, E. and Tao, T. (2007). The dantzig selector: Statistical estimation when p is much larger than n. The annals of Statistics, 35(6):2313–2351.

Carroll, R. and Stefanski, L. (1994). Measurement error, instrumental variables and corrections for attenuation with applications to meta-analyses. Stat. Med., 13(12):1265–1282.

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2010). The horseshoe estimator for sparse signals. Biometrika, 97(2):465–480.

Chang, C., Kundu, S., and Long, Q. (2018). Scalable bayesian variable selection for structured high-dimensional data. Biometrics, 74(4):1372–1382.

Chang, K.-W., Hsieh, C.-J., and Lin, C.-J. (2008). Coordinate descent method for large-scale l2-loss linear support vector machines. Journal of Machine Learning Research, 9(7).

Chen, Y. and Caramanis, C. (2013). Noisy and missing data regression: Distribution-oblivious support recovery. In International Conference on Machine Learning, pages 383–391. PMLR.

Cherkasova, M. V. and Hechtman, L. (2009). Neuroimaging in attention-deficit hyperactivity disorder: beyond the frontostriatal circuitry. The Canadian Journal of Psychiatry, 54(10):651–664.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. Journal of the American statistical association, 74(368):829–836.

Cole, M. W., Reynolds, J. R., Power, J. D., Repovs, G., Anticevic, A., and Braver, T. S. (2013). Multi-task connectivity reveals flexible hubs for adaptive task control. Nature neuroscience, 16(9):1348.

Connor, K. M. and Davidson, J. R. (2003). Development of a new resilience scale: The connor-davidson resilience scale (cd-risc). Depression and anxiety, 18(2):76–82.

Cook, J. R. and Stefanski, L. A. (1994). Simulation-extrapolation estimation in parametric measurement error models. J Am Stat Assoc, 89(428):1314–1328.

Craddock, R. C., Holtzheimer III, P. E., Hu, X. P., and Mayberg, H. S. (2009). Disease state prediction from resting state functional connectivity. Magnetic Resonance in Medicine: An Official Journal of the Int Soc. for Magnetic Resonance in Medicine, 62(6):1619–1628.

Craddock, R. C., James, G. A., Holtzheimer III, P. E., Hu, X. P., and Mayberg, H. S. (2012). A whole brain fmri atlas generated via spatially constrained spectral clustering. Human brain mapping, 33(8):1914–1928.

Crambes, C., Kneip, A., Sarda, P., et al. (2009). Smoothing splines estimators for functional linear regression. Ann. Stat., 37(1):35–72.

Csardi, G., Nepusz, T., et al. (2006). The igraph software package for complex network research. InterJournal, complex systems, 1695(5):1–9.

Cui, P., Wang, X., Pei, J., and Zhu, W. (2018). A survey on network embedding. IEEE Transactions on Knowledge and Data Engineering, 31(5):833–852.

Cui, Z., Henrickson, K., Ke, R., and Wang, Y. (2019). Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. IEEE Transactions on Intelligent Transportation Systems, 21(11):4883–4894.

Datta, A., Zou, H., et al. (2017). Cocolasso for high-dimensional error-in-variables regression. Ann. Stat., 45(6):2400–2426.

Dedieu, A. (2019). Sparse (group) learning with lipschitz loss functions: a unified analysis. arXiv preprint arXiv:1910.08880.

Du, Y., Fu, Z., and Calhoun, V. D. (2018). Classification and prediction of brain disorders using functional connectivity: promising but challenging. Frontiers in neuroscience, 12:525.

Duchi, J., Shalev-Shwartz, S., et al. (2008). Efficient projections onto the l 1-ball for learning in high dimensions. In Proc. 25th Int. Conf. on ML, pages 272–279.

Durante, D., Dunson, D. B., and Vogelstein, J. T. (2017). Nonparametric bayes modeling of populations of networks. Journal of the American Statistical Association, 112(520):1516–1530.

Edelman, A. (1988). Eigenvalues and condition numbers of random matrices. SIAM journal on matrix analysis and applications, 9(4):543–560.

Ekeland, I. and Temam, R. (1999). Convex analysis and variational problems. SIAM.

Fair, D. A., Posner, J., Nagel, B. J., Bathula, D., Dias, T. G. C., Mills, K. L., Blythe, M. S., Giwa, A., Schmitt, C. F., and Nigg, J. T. (2010). Atypical default network connectivity in youth with attention-deficit/hyperactivity disorder. Biological psychiatry, 68(12):1084–1091.

Falconer, E., Bryant, R., Felmingham, K. L., Kemp, A. H., Gordon, E., Peduto, A., Olivieri, G., and Williams, L. M. (2008). The neural networks of inhibitory control in posttraumatic stress disorder. Journal of psychiatry & neuroscience: JPN, 33(5):413.

Falkai, P., Schmitt, A., and Andreasen, N. (2018). Forty years of structural brain imaging in mental disorders: is it clinically useful or not? Dialogues in clinical neuroscience, 20(3):179.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. Journal of the American statistical Association, 96(456):1348–1360.

Fan, J. and Lv, J. (2011). Nonconcave penalized likelihood with np-dimensionality. IEEE Trans. Inf. Theory, 57(8):5467–5484.

Fan, J., Xue, L., and Zou, H. (2014). Strong oracle optimality of folded concave penalized estimation. Ann. Stat., 42(3):819.

Feng, L., Bi, X., and Zhang, H. (2021). Brain regions identified as being associated with verbal reasoning through the use of imaging regression via internal variation. J Am Stat Assoc, 116(533):144–158.

Feng, X., Li, T., Song, X., and Zhu, H. (2019). Bayesian scalar on image regression with nonignorable nonresponse. J Am Stat Assoc, pages 1–24.

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., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. Journal of statistical software, 33(1):1.

Geweke, J. (1992). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. In IN BAYESIAN STATISTICS. Citeseer.

Goldsmith, J., Wand, M. P., and Crainiceanu, C. M. (2011). Functional regression via variational bayes. Electron. J. Statist., 5:572–602.

Gramacy, R. B. (2018). monomvn: Estimation for Multivariate Normal and Student-t Data with Monotone Missingness. R package version 1.9-8.

Guha, S. and Rodriguez, A. (2020). Bayesian regression with undirected network predictors with an application to brain connectome data. Journal of the American Statistical Association, pages 1–13.

Hallquist, M. N. and Hillary, F. G. (2018). Graph theory approaches to functional network organization in brain disorders: A critique for a brave new small-world. Network Neuroscience, 3(1):1–26.

He, X., Pan, X., Tan, K. M., and Zhou, W.-X. (2021). Smoothed quantile regression with large-scale inference. Journal of Econometrics.

Higgins, I. A., Kundu, S., Choi, K. S., Mayberg, H. S., and Guo, Y. (2019). A difference degree test for comparing brain networks. Human brain mapping, 40(15):4518–4536.

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

Hoff, P. D. (2005). Bilinear mixed-effects models for dyadic data. Journal of the american Statistical association, 100(469):286–295.

Hoff, P. D., Raftery, A. E., and Handcock, M. S. (2002). Latent space approaches to social network analysis. Journal of the american Statistical association, 97(460):1090–1098.

Hong, B., Zhang, W., Liu, W., Ye, J., Cai, D., He, X., and Wang, J. (2019). Scaling up sparse support vector machines by simultaneous feature and sample reduction. J. Mach. Learn. Res., 20:121–1.

Hsieh, C.-J., Dhillon, I., Ravikumar, P., and Sustik, M. (2011). Sparse inverse covariance matrix estimation using quadratic approximation. Advances in neural information processing systems, 24.

Huang, J., Ma, S., Xie, H., and Zhang, C.-H. (2009). A group bridge approach for variable selection. Biometrika, 96(2):339–355.

James, G. M. (2002). Generalized linear models with functional predictors. J. R. Stat. Soc. Ser. B Methodol., 64(3):411–432.

James, G. M. and Radchenko, P. (2009). A generalized dantzig selector with shrinkage tuning. Biometrika, 96(2):323–337.

Jiang, W. et al. (2007). Bayesian variable selection for high dimensional generalized linear models: convergence rates of the fitted densities. The Annals of Statistics, 35(4):1487–1511.

Kraemer, G., Reichstein, M., and D., M. M. (2018). dimRed and coRanking—unifying dimensionality reduction in r. The R Journal, 10(1):342–358. coRanking version 0.2.2.

Kundu, S. and Dunson, D. B. (2014). Latent factor models for density estimation. Biometrika, 101(3):641–654.

Kundu, S., Lukemire, J., Wang, Y., and Guo, Y. (2019a). A novel joint brain network analysis using longitudinal alzheimer's disease data. Sci. Rep., 9(1):1–18.

Kundu, S., Mallick, B. K., Baladandayuthapani, V., et al. (2019b). Efficient bayesian regularization for graphical model selection. Bayesian Analysis, 14(2):449–476.

Lahat, D., Adali, T., and Jutten, C. (2015). Multimodal data fusion: an overview of methods, challenges, and prospects. Proc. IEEE, 103(9):1449–1477.

Lanius, R., Bluhm, R., Lanius, U., and Pain, C. (2006). A review of neuroimaging studies in ptsd: heterogeneity of response to symptom provocation. Journal of psychiatric research, 40(8):709–729.

Lazar, N. (2008). The statistical analysis of functional MRI data. Springer.

Lee, Y.-J. and Mangasarian, O. L. (2001). Ssvm: A smooth support vector machine for classification. Computational optimization and Applications, 20(1):5–22.

Li, Q., Wang, S., Huang, C.-C., Yu, M., and Shao, J. (2014). Meta-analysis based variable selection for gene expression data. Biometrics, 70(4):872–880.

Li, Y., Yu, R., Shahabi, C., and Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. arXiv preprint arXiv:1707.01926.

Liben-Nowell, D. and Kleinberg, J. (2007). The link-prediction problem for social networks. Journal of the American society for information science and technology, 58(7):1019–1031.

Liu, C. (2004). Robit regression: a simple robust alternative to logistic and probit regression. Applied Bayesian Modeling and Casual Inference from Incomplete-Data Perspectives, pages 227–238.

Liu, T. T. (2016). Noise contributions to the fmri signal: An overview. NeuroImage, 143:141–151.

Loh, P.-L. and Wainwright, M. J. (2012). High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. Ann. Stat., 40(3):1637–1664.

Loh, P.-L., Wainwright, M. J., et al. (2012). High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. The Annals of Statistics, 40(3):1637–1664.

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

Lukemire, J., Kundu, S., Pagnoni, G., and Guo, Y. (2020). Bayesian joint modeling of multiple brain functional networks. Journal of the American Statistical Association, pages 1–13.

Luo, J., Qiao, H., and Zhang, B. (2021). Learning with smooth hinge losses. arXiv preprint arXiv:2103.00233.

Lv, J., Fan, Y., et al. (2009). A unified approach to model selection and sparse recovery using regularized least squares. Ann. Stat., 37(6A):3498–3528.

Ma, X. and Kundu, S. (2021). Multi-task learning with high-dimensional noisy images. arXiv preprint arXiv:2103.03370.

Mallat, S. (1999). A wavelet tour of signal processing (3rd edition). Elsevier.

Meng, L. and Xiang, J. (2018). Brain network analysis and classification based on convolutional neural network. Frontiers in computational neuroscience, 12:95.

Monti, F., Boscaini, D., Masci, J., Rodola, E., Svoboda, J., and Bronstein, M. M. (2017). Geometric deep learning on graphs and manifolds using mixture model cnns. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5115–5124.

Morris, J. S. (2015). Functional regression. Annu Rev Stat Appl., 2:321–359.

Nardi, Y., Rinaldo, A., et al. (2008). On the asymptotic properties of the group lasso estimator for linear models. Electron. J. Stat., 2:605–633.

Nason, G. (2008). Wavelet methods in statistics with R. Springer Science & Business Media.

Negahban, S. N., Ravikumar, P., et al. (2012). A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. Stat. Sci., 27(4):538–557.

Nowicki, K. and Snijders, T. A. B. (2001). Estimation and prediction for stochastic block-structures. Journal of the American statistical association, 96(455):1077–1087.

Page, G., Bhattacharya, A., and Dunson, D. (2013). Classification via bayesian nonparametric learning of affine subspaces. Journal of the American Statistical Association, 108(501):187–201.

Peng, B., Wang, L., and Wu, Y. (2016). An error bound for l1-norm support vector machine coefficients in ultra-high dimension. The Journal of Machine Learning Research, 17(1):8279–8304.

Polson, N. G., Scott, J. G., and Windle, J. (2013). Bayesian inference for logistic models using pólya–gamma latent variables. Journal of the American statistical Association, 108(504):1339–1349.

Power, J. D., Cohen, A. L., Nelson, S. M., Wig, G. S., Barnes, K. A., Church, J. A., Vogel, A. C., Laumann, T. O., Miezin, F. M., Schlaggar, B. L., et al. (2011). Functional network organization of the human brain. Neuron, 72(4):665–678.

Ramsay, J. O. and Dalzell, C. (1991). Some tools for functional data analysis. J. R. Stat. Soc. Ser. B Methodol., 53(3):539–561.

Ramsay, J. O. and Silverman, B. W. (2005). Functional Data Analysis. Springer, New York.

Raser, J. M. and O'shea, E. K. (2005). Noise in gene expression: origins, consequences, and control. Science, 309(5743):2010–2013.

Raskutti, G., Wainwright, M. J., and Yu, B. (2010). Restricted eigenvalue properties for correlated gaussian designs. The Journal of Machine Learning Research, 11:2241–2259.

Rasmussen, C. E. and Williams, C. K. I. (2006). Gaussian processes for machine learning. MIT Press Cambridge, MA.

Reiss, P. T., Huo, L., Zhao, Y., Kelly, C., and Ogden, R. T. (2015). Wavelet-domain regression and predictive inference in psychiatric neuroimaging. Ann Appl Stat, 9(2):1076.

Relión, J. D. A., Kessler, D., Levina, E., and Taylor, S. F. (2019). Network classification with applications to brain connectomics. The annals of applied statistics, 13(3):1648.

Robert, C. P. (2015). The metropolis-hastings algorithm. arXiv preprint arXiv:1504.01896.

Rolls, E. T., Cheng, W., and Feng, J. (2021). Brain dynamics: the temporal variability of connectivity, and differences in schizophrenia and adhd. Translational psychiatry, 11(1):1–11.

Rolls, E. T., Huang, C.-C., Lin, C.-P., Feng, J., and Joliot, M. (2020). Automated anatomical labelling atlas 3. Neuroimage, 206:116189.

Rosenbaum, M. and Tsybakov, A. B. (2010). Sparse recovery under matrix uncertainty. The Annals of Statistics, 38(5):2620–2651.

Rosenbaum, M. and Tsybakov, A. B. (2013). Improved matrix uncertainty selector. In From Probability to Statistics and Back: High-Dimensional Models and Processes–A Festschrift in Honor of Jon A. Wellner, pages 276–290. Institute of Mathematical Statistics.

Savitsky, T., Vannucci, M., and Sha, N. (2011). Variable selection for nonparametric gaussian process priors: Models and computational strategies. Statistical science: a review journal of the Institute of Mathematical Statistics, 26(1):130.

Scher, C. D., Stein, M. B., Asmundson, G. J., McCreary, D. R., and Forde, D. R. (2001). The childhood trauma questionnaire in a community sample: psychometric properties and normative data. Journal of traumatic stress, 14(4):843–857.

Schirmer, M. D., Venkataraman, A., Rekik, I., Kim, M., Mostofsky, S. H., Nebel, M. B., Rosch, K., Seymour, K., Crocetti, D., Irzan, H., et al. (2021). Neuropsychiatric disease

classification using functional connectomics-results of the connectomics in neuroimaging transfer learning challenge. Medical image analysis, 70:101972.

Sørensen, Ø., Frigessi, A., and Thoresen, M. (2015). Measurement error in lasso: Impact and likelihood bias correction. Stat. Sin., pages 809–829.

Sørensen, Ø., Hellton, K. H., Frigessi, A., and Thoresen, M. (2018). Covariate selection in high-dimensional generalized linear models with measurement error. Journal of Computational and Graphical Statistics, 27(4):739–749.

Sprang, G. (1997). The traumatic experiences inventory (tei): A test of psychometric properties. Journal of Psychopathology and Behavioral Assessment, 19(3):257–271.

Sra, S. (2011). Fast projections onto l 1, q-norm balls for grouped feature selection. In Joint Eur Confer on Mach Learn and Knowl Discovery in Databases, pages 305–317. Springer.

Sripada, R. K., Garfinkel, S. N., and Liberzon, I. (2013). Avoidant symptoms in ptsd predict fear circuit activation during multimodal fear extinction. Frontiers in human neuroscience, 7:672.

Stevens, J. S., Jovanovic, T., Fani, N., Ely, T. D., Glover, E. M., Bradley, B., and Ressler, K. J. (2013). Disrupted amygdala-prefrontal functional connectivity in civilian women with posttraumatic stress disorder. Journal of psychiatric research, 47(10):1469–1478.

Tang, L. and Song, P. X. (2016). Fused lasso approach in regression coefficients clustering: learning parameter heterogeneity in data integration. J Mach Learn Res, 17(1):3915–3937.

Tenenbaum, J. B., De Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. science, 290(5500):2319–2323.

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

Tustison, N. J., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P. A., and Gee, J. C. (2010). N4itk: improved n3 bias correction. IEEE Trans Med Imaging, 29(6):1310–1320.

Tustison, N. J., Cook, P. A., Klein, A., Song, G., Das, S. R., Duda, J. T., Kandel, B. M., van Strien, N., Stone, J. R., Gee, J. C., et al. (2014). Large-scale evaluation of ants and freesurfer cortical thickness measurements. Neuroimage, 99:166–179.

Tustison, N. J., Holbrook, A. J., Avants, B. B., et al. (2019). Longitudinal mapping of cortical thickness measurements: An alzheimer's disease neuroimaging initiative-based evaluation study. J. Alzheimer's Dis., 71(1):165–183.

Vaishali, S., Rao, K., and Rao, G. S. (2015). A review on noise reduction methods for brain mri images. In 2015 Int Conf on Signal Proc and Comm Eng Sys, pages 363–365. IEEE.

Van De Geer, S. A., Bühlmann, P., et al. (2009). On the conditions used to prove oracle results for the lasso. Electron. J. Stat., 3:1360–1392.

Vance, A., Silk, T., Casey, M., Rinehart, N. J., Bradshaw, J. L., Bellgrove, M. A., and Cunnington, R. (2007). Right parietal dysfunction in children with attention deficit hyperactivity disorder, combined type: a functional mri study. Molecular psychiatry, 12(9):826–832.

Vrahatis, M. N. (1989). A short proof and a generalization of miranda's existence theorem. Proc. Am. Math. Soc., 107(3):701–703.

Walker, J. S. (2008). A primer on wavelets and their scientific applications. CRC press.

Wang, G., Zhang, G., Choi, K.-S., Lam, K.-M., and Lu, J. (2020). Output based transfer learning with least squares support vector machine and its application in bladder cancer prognosis. Neurocomputing, 387:279–292.

Wang, X., Nan, B., Zhu, J., and Koeppe, R. (2014). Regularized 3d functional regression for brain image data via haar wavelets. Ann Appl Stat, 8(2):1045.

Wang, X., Zhu, H., and Initiative, A. D. N. (2017). Generalized scalar-on-image regression models via total variation. J Am Stat Assoc, 112(519):1156–1168.

Weaver, C., Xiao, L., and Lindquist, M. A. (2021). Single-index models with functional connectivity network predictors. Biostatistics.

Weiner, M. W. and Veitch, D. P. (2015). Introduction to special issue: overview of alzheimer's disease neuroimaging initiative. Alzheimers. Dement., 11(7):730–733.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y. (2020). A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems, 32(1):4–24.

Yang, Y., Dunson, D. B., et al. (2016). Bayesian manifold regression. The Annals of Statistics, 44(2):876–905.

Yetkin, F. Z., Rosenberg, R. N., et al. (2006). Fmri of working memory in patients with mild cognitive impairment and probable alzheimer's disease. Eur. Rad., pages 193–206.

Zhan, X. and Yu, R. (2015). A window into the brain: advances in psychiatric fmri. BioMed research international, 2015.

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

Zhang, S., Zhou, D., Yildirim, M. Y., Alcorn, S., He, J., Davulcu, H., and Tong, H. (2017). Hidden: hierarchical dense subgraph detection with application to financial fraud detection. In Proceedings of the 2017 SIAM International Conference on Data Mining, pages 570–578. SIAM.

Zhang, Y. and Yang, Q. (2018). An overview of multi-task learning. N. Sci. Rev, 5(1):30–43.

Zhao, P. and Yu, B. (2006). On model selection consistency of lasso. J. Mach. Learn. Res., 7(Nov):2541–2563.

Zhou, D., Zhang, S., Yildirim, M. Y., Alcorn, S., Tong, H., Davulcu, H., and He, J. (2017). A local algorithm for structure-preserving graph cut. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 655–664.

Zhou, N. and Zhu, J. (2010). Group variable selection via a hierarchical lasso and its oracle property. Stat. Its Interface, 3(4):557–574.

Zhu, H., Leus, G., and Giannakis, G. B. (2011). Sparsity-cognizant total least-squares for perturbed compressive sampling. IEEE Transactions on Signal Processing, 59(5):2002–2016.

Ziemer, W. P. (2012). Weakly differentiable functions: Sobolev spaces and functions of bounded variation, volume 120. Springer Science & Business Media.

Zipunnikov, V., Caffo, B., Yousem, D. M., Davatzikos, C., Schwartz, B. S., and Crainiceanu, C. (2011). Functional principal component model for high-dimensional brain imaging. NeuroImage, 58(3):772–784.

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.