Learning Structure in High-Dimensional Data with Applications to Neuroimaging

by

Daniel Allan Kessler

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Statistics)
in the University of Michigan
2023

Doctoral Committee:

Professor Elizaveta Levina, Chair
Assistant Professor Snigdha Panigrahi
Professor Chandra Sripada
Professor Ji Zhu

Daniel Allan Kessler
kesslerd@umich.edu
ORCID iD: 0000-0003-2052-025X

DEDICATION

This dissertation is dedicated to the memory of Takanori Watanabe, PhD.

ACKNOWLEDGEMENTS

I am deeply grateful to my advisor, Dr. Liza Levina, without whom I likely would not have pursued a PhD in statistics. While there are many features of the Michigan Statistics Department that I treasure, in my mind nothing comes close to the fact that Liza is in this department. I wish to also thank my committee members with whom I have had the opportunity to work closely, some for many years. Dr. Ji Zhu has provided invaluable regular feedback on my research projects at weekly group meetings and has also been an excellent source of professional advice and friendship. Dr. Snigdha Panigrahi has been a fantastic instructor and coauthor and is single-handedly responsible for getting me excited about selective inference. Dr. Chandra Sripada was instrumental in developing my skills as a researcher, my knowledge of neuroscience and psychiatry, and has contributed immensely to my ability to write and communicate my ideas. I am also grateful to the members of Chandra's research group, especially Mike Angstadt and Saige Rutherford, who were extraordinarily helpful in facilitating access to valuable data.

I feel very fortunate to have pursued my graduate training in such an amazingly nurturing and friendly department, and I am especially glad of everyone who has helped to make the Student Council a valuable part of department life. I'm also grateful to my many friends in the department, with particular gratitude to Michael Law for his friendship and help.

I am grateful to my extended family near and far. Since I was a child, you have helped to foster my curiosity and modeled how to approach others with respect and kindness. I am thankful to my enormously supportive parents who have always encouraged and enabled me to pursue my interests without pressure or expectation, and to my brother and his family whom I feel so fortunate to have lived near during my graduate training.

I owe an extraordinary debt of gratitude to my partner, Sara, without whom I doubt I would have made it more than a year into my PhD program. Your constant encouragement, understanding, and unwavering support have made all the difference. Thank you, too, for adding Isla to our family.

I was fortunate to be supported in my work by NSF-DMS 1646108 and a Rackham Predoctoral Fellowship from the University of Michigan.

Data were provided in part by the Human Connectome Project, WU-Minn Consortium (Principal Investigators: David Van Essen and Kamil Ugurbil; 1U54MH091657) funded by

the 16 NIH Institutes and Centers that support the NIH Blueprint for Neuroscience Research; and by the McDonnell Center for Systems Neuroscience at Washington University.

Data used in the preparation of this work were obtained from the Adolescent Brain Cognitive Development (ABCD) Study (https://abcdstudy.org), held in the NIMH Data Archive (NDA). This is a multisite, longitudinal study designed to recruit more than 10,000 children age 9-10 and follow them over 10 years into early adulthood. The ABCD Study® is supported by the National Institutes of Health and additional federal partners under award numbers U01DA041048, U01DA050989, U01DA051016, U01DA041022, U01DA051018, U01DA051037, U01DA050987, U01DA041174, U01DA041106, U01DA041117, U01DA041028, U01DA041134, U01DA050988, U01DA051039, U01DA041156, U01DA041025, U01DA041120, U01DA051038, U01DA041148, U01DA041093, U01DA041089, U24DA041123, U24DA041147. A full list of supporters is available at https://abcdstudy.org/federal-partners.html. A listing of participating sites and a complete listing of the study investigators can be found at https://abcdstudy.org/consortium_members/. ABCD consortium investigators designed and implemented the study and/or provided data but did not necessarily participate in the analysis or writing of this report. This manuscript reflects the views of the authors and may not reflect the opinions or views of the NIH or ABCD consortium investigators. The ABCD data repository grows and changes over time. The ABCD data used in this report came from NDA Study 721, 10.15154/1504041, which can be found at https://nda.nih.gov/study.html?id=721. The specific NDA study associated with this report is NDA Study 1364, 10.15154/1523385.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

LIST OF APPENDICES

ABSTRACT

The scale of modern datasets, with more and more variables measured on more and more observations, presents many statistical challenges, but also opportunities to discover and exploit the rich structure that is often present in the data. In neuroimaging studies, multiple kinds of brain imaging are conducted on the same participant, with each modality of imaging having its own further structure, and many associated phenotypic measurements taken on the participants. Understanding the complicated and noisy underlying relationships between all of these measurements holds promise for scientific and treatment breakthroughs in the long term, and requires sophisticated methods designed to uncover this structure. This thesis presents three projects on learning structure in high-dimensional datasets motivated by applications in neuroimaging.

The first project considers the setting where many networks are observed on a common node set: each observation comprises edge weights, covariates observed at each node, and a response. In our neuroimaging application, the edge weights correspond to functional connectivity between brain regions, node covariates encode task activations at each brain region, and performance on a behavioral task is the response. The goal is to use the edge weights and node covariates to predict the response and to identify a parsimonious and interpretable set of predictive features. We propose an approach that uses feature groups defined according to a community structure believed to exist in the network (naturally occurring in neuroimaging applications). We propose two schemes for forming feature groups where each group incorporates both edge weights and node covariates, and derive optimization algorithms for both using an overlapping group LASSO penalty. Empirical results on synthetic data show that our method, relative to competing approaches, has similar or improved prediction error along with superior support recovery, enabling a more interpretable and potentially a more accurate understanding of the underlying process. We also apply the method to neuroimaging data.

The second project focuses on inference for structure learned using Canonical Correlation Analysis (CCA). CCA is a method for analyzing a sample of pairs of random vectors; it learns a sequence of paired linear transformations of the original variables that are maximally correlated within pairs while uncorrelated across pairs. CCA outputs both canonical

correlations as well as the canonical directions which define the transformations. While inference for canonical correlations is well developed, conducting inference for canonical directions is more challenging and not well-studied, but is key to interpretability. We propose a computational bootstrap method for inference on CCA direction (combootcca). We conduct thorough simulation studies that range from simple and well-controlled to complex but realistic and validate the statistical properties of combootcca while comparing it to several competitors. We also apply the combootcca method to a brain imaging dataset and discover linked patterns in brain connectivity and behavioral scores.

The third project proposes a new method for matrix CCA (matcca), which works with pairs of random matrices rather than pairs of random vectors, motivated by a neuroimaging application where the brain imaging data takes the form of a high-dimensional covariance matrix. Our matcca method uses a nuclear norm penalty that encourages the canonical directions associated with the matrix-variate data to have low rank structure when arranged into a matrix. Results from both synthetic and neuroimaging data show that matcca is very effective at recovering low rank signals even in noisy cases with few observations.

CHAPTER 1

Introduction

Modern scientific datasets are growing ever larger and larger in the big data era. While much statistical theory has been developed for the asymptotic regime wherein the number of observations N and/or the number of predictors p grow, the reality is that this growth is generally not smooth or uniform in p: expanding the number of features used during data collection is not tantamount to drawing balls from an urn. Instead, new features are often acquired in blocks (e.g., adding microarray data to a study adds many new features all of a similar type), and they often have partial overlap with old features. As a result, while modern datasets have intimidating scale, they also come with rich structure, which, if carefully exploited, can help us to find signal even from relatively few noisy observations. Moreover, aligning statistical analyses to structure allows us to take advantage of ongoing advances in scientific understanding of complex phenomena. In some sense, we wish to partially automate the "step" of the analysis wherein the statistician presents the results to the scientist and asks if what they have found seems plausible.

This kind of structure is certainly present in neuroimaging studies, which serves as the motivating application throughout this dissertation, where multiple kinds of brain imaging are typically acquired on the same participants. This multimodal data offers non-redundant views into the brain (Uludağ and Roebroeck, 2014), and while it is of course possible to analyze each view separately, we require sophisticated methods in order to discover and exploit the potentially rich structure that exists within individual modalities as well as those that links these different views.

In Chapter 2, we assume the presence of structure based on a neuroscientifically plausible hypothesis, and we use this structure to guide feature selection. Of note, Chapter 2 is adapted from Kessler et al. (2022) and reflects joint work with both the advisor and a former postdoc. In Chapter 3, we make minimal assumptions about the form of the structure and instead use Canonical Correlation Analysis (CCA), a classic statistical tool, in order to discover structure that relates one random vector to another (and vice versa). However, because CCA will essentially *always* find something apparently interesting, we develop and

validate a computational bootstrap approach in order to subject our discovered structure to inference. In Chapter 4, in order to make more direct use of the structure in our data, we extend CCA from the vector-variate to the matrix-variate setting. In so doing, we are able to use matrix-specific penalties in order to guide our discoveries.

Although the three projects in this dissertation all involve the use of structure in large, complex datasets encountered in the course of human neuroimaging research, each has a distinct statistical goal related to different common themes in statistics. In Chapter 2, the goal is interpretable *prediction*: we want to predict a score or label after observing a weighted network and associated node covariates; in Chapter 3, the goal is *inference*: after performing CCA, we want to conduct inference on our canonical directions; in Chapter 4, the goal is *estimation*: we want to exploit low-rank structure in matrix-variate data in order to obtain better estimates of the canonical directions when the sample is small or the signal is weak. Finally, in Chapter 5, we conclude by summarizing our contributions and then outlining several new lines of work related to the projects presented.

CHAPTER 2

Predicting Responses from Weighted Networks with Node Covariates in an Application to Neuroimaging

2.1 Introduction

Predicting a response such as a psychiatric disease status from brain scans of a given individual is an increasingly common task in human neuroimaging; see Calhoun et al. (2017); Arbabshirani et al. (2017); Burgos and Colliot (2020) for some reviews. While earlier work in neuroimaging, especially in functional magnetic resonance imaging (fMRI), focused on brain activation in response to particular tasks, the use of "resting state" imaging has become increasingly popular as a means of characterizing brain patterns and understanding differences across individuals and populations. Neuroimaging studies increasingly aim to predict individual phenotypes from the resting state functional connectivity (Khosla et al., 2019). In addition to functional connectivity measurements, many of these studies also acquire spatially-localized brain characteristics (e.g., activation in response to a cognitive task) on the same participants. The simultaneous use of multiple modalities obtained from brain imaging, e.g., both connectivity and activation during a task, offers an opportunity for better prediction and deeper understanding in how various characteristics of the brain affect the phenotype (Calhoun and Sui, 2016).

This task is an instance of a general statistical problem: modeling or predicting a response y as a function of one or more network-valued predictors. In general, a network contains information about connections (edges) between units of observation (nodes), and may also have additional information on the nodes available (node covariates). In the neuroimaging application above, the nodes are locations in the brain, and the edges correspond to connectivity between these locations. Typical neuroimaging networks are undirected, and edge weights represent the strength of connectivity, though this can vary with imaging modality.

The nodes can be "labeled" by mapping every participant's brain onto a common anatomical atlas (e.g., Power et al., 2011), also known as a parcellation, with the result that the networks can be aligned with each other and observed on a common node set. Node covariates may take the form of measurements such as activation in response to a cognitive task or gray matter volume, and the response y is a participant-level variable such as a cognitive task score or disease status.

In human neuroimaging, available parcellations of the brain often provide not only a common atlas for nodes but also a partition into "brain systems," which we may think of as network communities. These communities may be based on domain knowledge or result from a prior data analysis involving some type of community detection algorithm (Yeo et al., 2011; Power et al., 2011). Obtaining results at the level of these brain systems rather than individual nodes aids interpretability and comparisons across studies, and helps to balance power and spatial specificity (Noble et al., 2022).

Currently popular methods in the field, such as connectome predictive modeling (Shen et al., 2017) and Brain Basis Set modeling (Sripada et al., 2019), tend to vectorize edge weights and use them as a "bag of features" (Chung et al., 2021) to feed into conventional supervised learning algorithms. These approaches do not account for community structure that may be present and offer only ad hoc interpretation at that level of resolution. Further, these methods typically do not accommodate node covariates. Domain-agnostic approaches for variable selection, such as the LASSO (Tibshirani, 1996), similarly disregard the network structure of the data, but can still be used as a predictive performance benchmark for the new, more interpretable methods we aim to develop here.

Our goal in this work is to develop methods that can predict the response accurately from edge weights and node covariates while providing interpretation at the level of network communities. We call such methods "network-aware," in contrast to the "bag of features" methods discussed above. We do this by imposing structured penalties that reflect communities, proposing two different grouping schemes depending on the mechanism believed to be involved, and deriving an efficient algorithm based on overlapping group LASSO to obtain interpretable group-sparse solutions. We call this method NetCov, for prediction from *net*works with node *cov*ariates. This approach is in contrast to methods that first aggregate connectivity among communities (e.g., Yu et al., 2019), which may miss more nuanced patterns when the signal within a community is heterogeneous.

Some examples of previous uses of a group LASSO penalty in neuroimaging include Shimizu et al. (2015), which used both group and sparse group LASSO (SGL) with voxels grouped based on brain region in the classification of depression using task fMRI. Relión et al. (2019) proposed a prediction framework that uses an SGL penalty where each node of the

network has all of its incident edge weights grouped together (resulting in overlapping groups) and applied this to a Schizophrenia dataset. Richie-Halford et al. (2021) also proposed an SGL-based method intended for use with diffusion-weighted magnetic resonance imaging (dMRI) where voxel-level features are grouped based on tissue tract. To the best of our knowledge, NetCov is the first method to (a) construct groups that include both edge weights and node covariates, thus naturally spanning multiple imaging modalities, and (b) leverage community information in the construction of groups.

While our goal is to predict a response y given an observed network, a related line of work approaches the converse problem and aims to characterize networks given an observed response. For example, Tang et al. (2017) proposed a method for testing the hypothesis that two networks are drawn from the same distribution and applied it to neural connectome graphs. Ginestet et al. (2017) obtained a central limit theorem for networks which enabled the construction of Wald-like hypothesis tests for samples of networks analogous to classical one- and two-sample tests and then illustrated the method on functional connectivity data. Xia et al. (2020) introduced a "Multi-scale network regression" model in which edge weights are predicted using phenotypes in a penalized model. Another more recent instance of work that assesses how networks change given a response is Kim et al. (2023), which modeled edge weights using a mixed-effects framework with a network-aware variance structure.

The remainder of the paper is organized as follows. In Section 2.2, we propose our model and describe the two feature grouping schemes. The fitting algorithm is presented in Section 2.3. Numerical experiments assessing our approach and comparing it to other methods are presented in Section 2.4. We then apply the approach to data from a large human neuroimaging study in Section 2.5, and conclude with a discussion of limitations and future work in Section 2.6.

2.2 The NetCov Model and Network-Aware Penalties

We start by fixing notation. Let N be the number of observations (e.g., participants, in the neuroimaging context), and let the data collected for each participant $i = 1, \ldots, N$ be the triple $\left(A^{(i)}, X^{(i)}, y^{(i)}\right)$. Here $A^{(i)}$ is the $n \times n$ signed and weighted adjacency matrix associated with the i-th observation, on a common set of nodes labeled $1, \ldots, n$, with $A^{(i)}_{kl}$ representing the weight of the edge from node k to node l for participant i. The matrix $X^{(i)} \in \mathbb{R}^{n \times d}$ contains node covariates for participant i, with the k-th row corresponding to the covariates of node k. The response variable $y^{(i)}$ for participant i may be real-valued or categorical. Finally, let $c : \{1, \ldots, n\} \to \{1, \ldots, K\}$ be the map that assigns each node k, $k = 1, \ldots, n$ to one of K possible communities. We assume this map is known (or learned previously) and

will use it to construct feature groups.

Importantly, the response $y^{(i)}$, edge weights $A^{(i)}$, and node covariates $X^{(i)}$ are all observed, and while we expect there may be correlations among them, they are not inferred one from the other. The class of matrices $A^{(i)}$ may be further restricted based on the application at hand. For instance, in all our examples, the networks are undirected, and so the $A^{(i)}$ are symmetric matrices.

2.2.1 Prediction Model

We use a standard generalized linear model for relating the response y to the predictors (A, X), since we aim for interpretable parameters. We assume that conditional on (A, X), y follows a distribution amenable to generalized linear modeling (McCullagh and Nelder, 1998) and satisfies

$$\mathbb{E}[y] = g^{-1}\left\{\mu + \text{Trace}\left(\beta_A^\intercal A\right) + \text{Trace}\left(\beta_X^\intercal X\right)\right\},$$

where g^{-1} is the inverse of the link function g, and $\mu \in \mathbb{R}$, $\beta_A \in \mathbb{R}^{n\times n}$, and $\beta_X \in \mathbb{R}^{n\times d}$ are (unknown) coefficients. The choice of g will depend on the setting: for continuous y we may use the identity link function, and for binary y letting $g^{-1}(t) = \frac{1}{1+\exp(-t)}$ yields a logistic regression model. Note that if the networks have no self-loops or are undirected, many of the entries of A are either redundant or of no interest, and we can remove these terms from the model by constraining the corresponding entries of β_A to also be zero.

It will often be notationally convenient to use $\boldsymbol{Z}^{(i)} \in \mathbb{R}^p$ to denote an appropriately vectorized version of $\left(A^{(i)}, X^{(i)}\right)$ and to let $\boldsymbol{\beta} \in \mathbb{R}^p$, be an analogously vectorized version of (β_A, β_X). The dimension p depends on the number of non-redundant entries; for instance, if the networks are undirected with no self-loops, then $p = n(n-1)/2 + dn$. If we write $Z \in \mathbb{R}^{N\times p}$ for the matrix with $\boldsymbol{Z}^{(i)}$ as the i-th row, we can write

$$\mathbb{E}[\boldsymbol{y}] = g^{-1}\left(\mu + Z\boldsymbol{\beta}\right),$$

which substantially simplifies subsequent derivations.

2.2.2 Feature Groups

Feature groups are at the core of our method and its goal to provide interpretable network-aware solutions. Suppose, without loss of generality, that the nodes are ordered such that community assignments are contiguous and non-decreasing, i.e., $i < j \implies c(i) \leq c(j)$. Recall that coefficients corresponding to the node covariates are held in the matrix $\beta_X \in$

$\mathbb{R}^{n\times d}$. Since each row corresponds to coefficients associated with a distinct node, and each node is assigned to a community, we can partition the predictors in $X^{(i)}$ into K "blocks," where each block comprises the covariates associated with the nodes in a specific community. Let $G_X^k = \{i : c(i) = k\}$ denote a given block. Recall that coefficients corresponding to edge weights are held in the matrix $\beta_A \in \mathbb{R}^{n\times n}$. We can partition β_A into "cells," where each cell corresponds to coefficients associated with edges linking nodes belonging to a specific pair of communities. If the network is directed, there will be K^2 cells, whereas if it is undirected there will be $K(K+1)/2$ cells. Let $G_A^{k,k'} = \{(i,j) : c(i) = k \text{ and } c(j) = k'\}$ denote a given cell. An example of this partitioning scheme is depicted in Fig. 2.1.

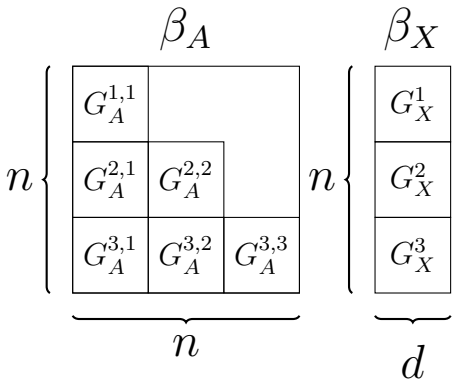

Figure 2.1: Feature groups for undirected networks with $K = 3$. Network cells are on the left and node blocks are on the right.

Blocks and cells form natural grouping units for nodal and edge covariates, respectively, and the question is how to combine them. We propose two different feature grouping schemes: Node-Based Groups (NBG) and Edge-Based Groups (EBG). These two schemes are motivated principally by the neuroscientific notion of "lesion network mapping" (Fox, 2018), and are schematically illustrated in Fig. 2.2. In the first scenario (NBG), an aberration caused by a disease, trauma, etc., affects a brain system, corresponding to say community k. This affects the block G_X^k and also affects its connectivity with other brain systems, so $G_A^{1,k}, G_A^{2,k}, \ldots, G_A^{K,k}$ are affected too. Formally, the K feature groups under NBG, $\mathcal{G} = \left\{G^1, G^2, \ldots, G^K\right\}$, are given by

$$G^k = G_X^k \cup \bigcup_{j=1}^{K} G_A^{j,k}.$$

In the second scenario (EBG), the aberration affects edges instead of nodes, disrupting connectivity between two systems, say k and k', affecting $G_A^{k,k'}$. This in turn affects covariates

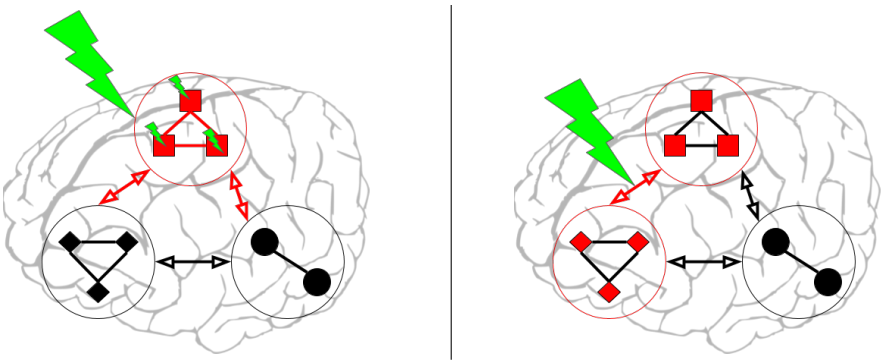

Figure 2.2: The neuroimaging motivation for grouping. Circled groups of nodes represent brain systems, lines represent connectivity between systems. Black is normal, red is abnormal, and a lightning bolt indicates a disease or injury. Left (NBG): a disease affects a system and therefore its connections to other systems also become abnormal. Right (EBG): a disease affects connectivity between two systems, and the systems themselves become abnormal.

within both systems, i.e., G_X^k and $G_X^{k'}$. Formally, the $K(K+1)/2$ groups under EBG, $\mathcal{G} = \left\{G^{k,k'} : 1 \leq k \leq k' \leq K\right\}$, are given by

$$G^{k,k'} = G_A^{k,k'} \cup G_X^k \cup G_X^{k'}.$$

To sum up, an NBG group corresponds to one community, including its nodes and all edges it is involved in, while an EBG group corresponds to a connection between two communities, including the nodes of both communities and edges between them. Note that under both NBG and EBG, each feature appears in at least one group, but the groups overlap. In either case, a given group comprises coefficients associated with both edge weights and node covariates. The stated definitions for undirected networks can be readily extended to directed settings. The corresponding groupings of the coefficients of $\boldsymbol{\beta}$ are illustrated, for $K = 3$, in Fig. 2.3 for NBG and Fig. 2.4 for EBG.

2.3 An Algorithm for Fitting NetCov

Recall that $Z \in \mathbb{R}^{N \times p}$ is a design matrix with row vectors corresponding to the appropriate vectorization of $\left(A^{(i)}, X^{(i)}\right)$ (where the dimension p will vary depending on the number of nodes, node covariates, whether the network is directed, etc), $\mu \in \mathbb{R}$ is an intercept, and $\boldsymbol{\beta} \in \mathbb{R}^p$ gives regression coefficients corresponding to the columns of Z. We write $\boldsymbol{\beta}_G$ to denote the subvector of $\boldsymbol{\beta}$ containing coefficients corresponding to features in group G. Similarly, Z_G denotes the submatrix formed by the columns of the design matrix Z corresponding to group G.

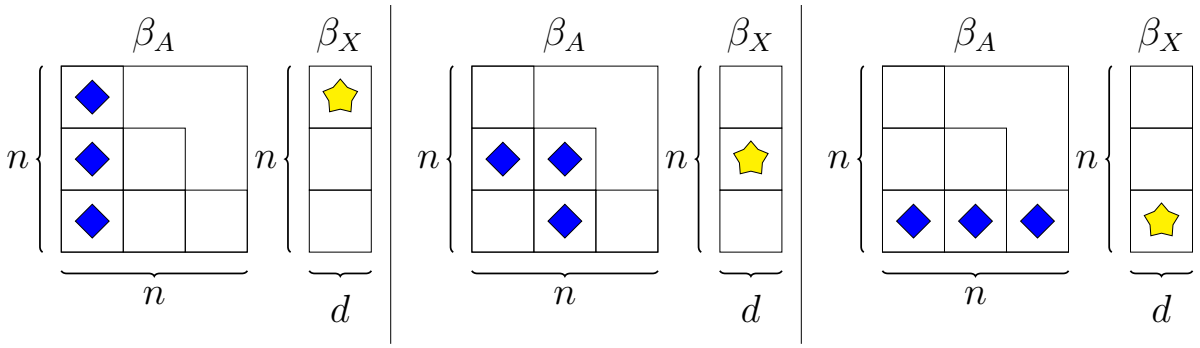

Figure 2.3: The NBG feature groups for $K = 3$. The panels, from left to right, show features associated with communities 1, 2, and 3. Yellow stars correspond to node covariates, and blue diamonds to edge weights.

2.3.1 The Objective Function

We fit NetCov following a standard approach of minimizing a penalized loss function,

$$Q(\mu, \boldsymbol{\beta} \mid Z, \boldsymbol{y}) = \frac{1}{n} L(\mu, \boldsymbol{\beta} \mid Z, \boldsymbol{y}) + \Omega(\boldsymbol{\beta}), \tag{2.1}$$

where L is a loss function measuring the fit to training data and Ω is a regularization penalty, which we use to encourage group sparsity. We use deviance as the loss function, which depends on the assumed distribution of the response. For example, for linear models (i.e., with the identity link function), we use the least squares loss function given by

$$\frac{1}{2} \sum_{i=1}^{N} \left(y^{(i)} - \mu - \boldsymbol{\beta}^\intercal \boldsymbol{Z}^{(i)} \right)^2.$$

For a binary response $y \in \{0, 1\}$, we use the logistic regression loss

$$-2 \sum_{i=1}^{N} \left[y^{(i)} \left(\mu + \boldsymbol{\beta}^\intercal \boldsymbol{Z}^{(i)} \right) - \log \left\{ 1 + \exp \left(\mu + \boldsymbol{\beta}^\intercal \boldsymbol{Z}^{(i)} \right) \right\} \right].$$

We assume that each predictor has been standardized to have mean 0 and variance 1, and a continuous response y is also standardized to have mean 0 and variance 1. The means and variances are learned from the training data only and are then used to normalize the test data, which, as a result, may not have exactly mean 0 and variance 1. There is another important and less trivial standardization step, discussed further in Section 2.3.2.

Recall that $\mathcal{G}$ denotes the collection of groups, where $G \in \mathcal{G}$ is a set of indices for a given group of variables, and the structure of G is determined by the mechanism we are looking

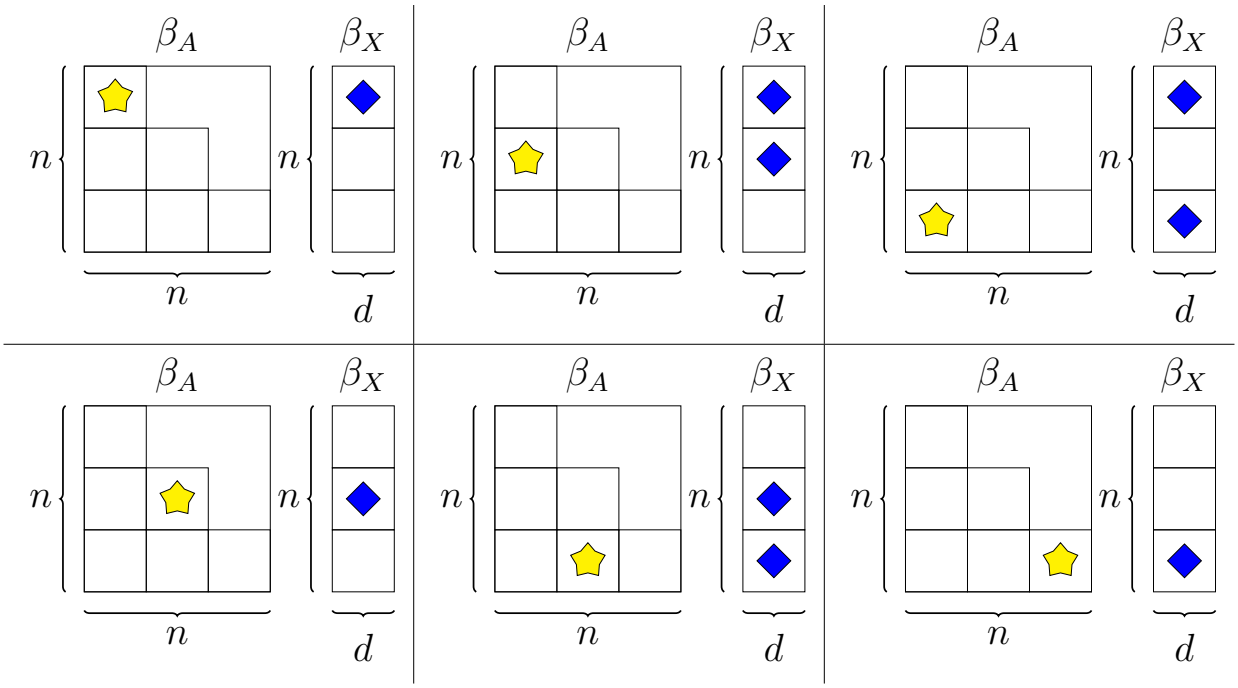

Figure 2.4: The EBG feature groups for $K = 3$. Each panel shows one group corresponding to a connection between communities k_1 and k_2, with $1 \leq k_1 \leq k_2 \leq 3$. Yellow stars correspond to node covariates, and blue diamonds to edge weights.

to find. We assume that the coefficients $\boldsymbol{\beta}$ are group-sparse, i.e., that only a small subset of feature groups are "active," that is, contain non-zero coefficients. The union of all active groups gives the active set of coefficients. This assumption is both pragmatic and principled: the number of the edge features is $O(n^2)$ which for all but very small networks puts us in a high-dimensional regime where some regularization is necessary. At the same time, selecting features in groups yields interpretability at the level of communities (brain systems, in our application), rather than at the level of individual edges and/or nodes. This is a desirable property for many applications, and especially so for neuroimaging, where domain knowledge generally exists at a resolution more compatible with large communities than specific brain coordinates, and where individual measurements are noisy and unreliable.

To select features in groups, we will employ the group LASSO penalty, which under appropriate conditions enjoys similar or superior theoretical properties relative to the standard LASSO (Nardi and Rinaldo, 2008; Huang and Zhang, 2010). The group penalty we use is of the form introduced by Yuan and Lin (2006),

$$\Omega(\boldsymbol{\beta}) = \lambda \sum_{G \in \mathcal{G}} \sqrt{|G|} \|\boldsymbol{\beta}_G\|_2, \tag{2.2}$$

where λ is a tuning parameter and $|G|$ is the cardinality of group G. Because the 2-norm is non-differentiable at $\mathbf{0}$, this penalty yields a loosely analogous geometry to the LASSO with critical points at solutions that are group-sparse, whereas the classical LASSO (Tibshirani, 1996) does not encourage group-sparse solutions.

In our setting, there is an additional complication: our feature groups overlap, as can be easily seen in Figs. 2.3 and 2.4. Optimization of an objective with penalty Eq. (2.2) has an undesirable property for our application: the critical points correspond to solutions where "inactive" groups have all coefficients set to 0. This means that we have a set of inactive groups $\mathcal{G}_I \subseteq \mathcal{G}$ such that $\forall G \in \mathcal{G}_I, \boldsymbol{\beta}_G = \mathbf{0}$, and a given feature can only be active if *all* of the groups in which it participates are active. In some circumstances, this may be a desirable property, for example as in Relión et al. (2019). However, the neuroimaging setting we consider calls for exactly the opposite: related nodes and edges are affected jointly. This problem is discussed in Jacob et al. (2009); Obozinski et al. (2011), and we follow their approach of duplicating variables to render the groups non-overlapping, then using the standard group LASSO formulation on the expanded variable set. That is, we construct, via concatenation, $Z^\star = \Big[Z_G : G \in \mathcal{G}\Big]$ and $\boldsymbol{\beta}^\star = \Big[\boldsymbol{\beta}_G^\intercal : G \in \mathcal{G}\Big]^\intercal$, and then minimize $Q(\mu, \boldsymbol{\beta}^\star \mid Z^\star, \boldsymbol{y})$. For the purpose of visualization and interpretation, we can map an estimate of $\boldsymbol{\beta}^\star$ back to the dimension of $\boldsymbol{\beta}$ through summation: e.g., suppose that the i-th coordinate of $\boldsymbol{\beta}$ appears in two groups and thus corresponds to coordinates j and j' of $\boldsymbol{\beta}^\star$; then we can find $(\boldsymbol{\beta})_i = (\boldsymbol{\beta}^\star)_j + (\boldsymbol{\beta}^\star)_{j'}$.

2.3.2 Standardizing within Groups

It is standard practice when fitting the LASSO to standardize the columns of the design matrix Z to have mean 0 and variance 1, since otherwise their coefficients are not on the same scale and cannot be sensibly combined into one penalty. For the group LASSO, Simon and Tibshirani (2012) argue that the appropriate normalization involves not only centering and rescaling columns, but orthonormalizing the columns corresponding to each group, Z_G. As discussed further in Breheny and Huang (2015), this orthonormalization yields a more straightforward and efficient algorithm, and is tantamount to penalizing the contribution of each group to the linear predictor. This approach is called the "groupwise prediction penalty" in Bühlmann and van de Geer (2011).

This can be accomplished in practice by computing the SVD of each $Z_G^\star = U_G \Sigma_G V_G^T$, where we limit the decomposition to singular vectors corresponding to nonzero singular values. We then construct a new design matrix comprising orthonormalized groups as $\tilde{Z}^\star = \Big[U_G : G \in \mathcal{G}\Big]$, and we use this quantity when minimizing Eq. (2.1) in $\tilde{\boldsymbol{\beta}}^\star$. Note that groups

of less than full column rank will have their penalty in Eq. (2.2) scaled based on the number of columns of their corresponding U_G, i.e., the rank of Z_G. After obtaining an optimal $\tilde{\boldsymbol{\beta}}^\star$, it is possible to invert both the orthonormalization and variable duplication to arrive at a solution that is parameterized by $\boldsymbol{\beta}$, which we use for both prediction and interpretation. See Breheny and Huang (2015); Zeng and Breheny (2016) for more details.

2.3.3 Implementation and Parameter Tuning

Efficient algorithms for solving the non-overlapping standardized group LASSO penalty in the context of both linear and logistic regression are presented in Breheny and Huang (2015) and available in the `grpreg` package in `R` (R Core Team, 2023). To use this approach in the overlapping case, we use the `grpregOverlap` (Zeng and Breheny, 2016) package: it manages variable duplication and depends heavily on `grpreg`. Unfortunately, as of this writing, this package is no longer available from CRAN, but it is available from Github at https://github.com/YaohuiZeng/grpregOverlap; this version incorporates a number of improvements and fixes that we contributed in the course of our present work.

In practice, the tuning parameter λ needs to be learned from the data. Following standard practice, we use cross-validation on training data to choose λ, for both NetCov and the regular LASSO, which we use as a baseline comparison. We adopt an approach based on the default settings for `glmnet` (Friedman et al., 2010b). First, we identify the data-driven quantity $\lambda_{\max}$, the smallest value of λ for which the selected model is fully sparse. Then, we set $\lambda_{\min} = 0.05\lambda_{\max}$ and create a logarithmically-spaced grid of candidate values for λ between $\lambda_{\min}$ and $\lambda_{\max}$. We then conduct ten-fold cross-validation at each of the values along this grid, and compute the average out-of-sample deviances. Let $\lambda^\star$ be the value of λ from the grid that minimizes the mean deviance. We then set $\hat{\lambda}$ to the largest value in the grid that has the out-of-sample deviance within one standard error of that corresponding to $\lambda^\star$. Finally, we refit the model to the full training set with $\lambda = \hat{\lambda}$.

2.4 Numerical Experiments

We conduct numerical experiments in a variety of settings to assess the performance of our procedure and to compare with competing strategies. Generating simulated data involves generating or specifying the covariates $A^{(i)}$, $X^{(i)}$, specifying the coefficients $\boldsymbol{\beta}$, and drawing the response $\boldsymbol{y}$ from an appropriate model. In our first set of simulations in Section 2.4.1 the design matrix is synthetic, which allows us to vary more parameters and explore their influence on performance. In Sections 2.4.2 and 2.4.3, we fix the design to correspond to

covariates from a human neuroimaging study.

To set coefficients $\boldsymbol{\beta}$, we vary (i) NBG vs EBG group structure, (ii) number of active feature groups (either 1 or 5), and (iii) the magnitude of active features (i.e., controlling the signal-to-noise ratio [SNR]). We simulate all these scenarios for both continuous and binary responses. For simplicity, we only consider undirected networks with no self-loops and keep the number of nodal covariates $d = 1$. However, all results can be readily extended beyond these settings.

All simulations include both a training set used for both parameter tuning and model fitting and a test set used to assess out-of-sample performance. While there are 8 unique combinations of group structure, number of active groups, and response (continuous versus binary), we smoothly vary the SNR across 20 levels; this leads to 160 unique settings for each experiment. We repeat each experiment 10 times for each setting and average the results. Because we conduct a total of nearly 10,000 experiments; we conduct our simulations on a high performance computing system with the extremely useful `R` package `batchtools` (Lang et al., 2017).

As a baseline comparison to our method, we include regular LASSO as implemented in the `glmnet` (Friedman et al., 2010b) package. For both NetCov and LASSO, the tuning parameter is chosen by cross-validation as described in Section 2.3.3. For the simulations, we do not include other potential competing methods developed specifically for neuroimaging, such as Brain Basis Sets (Sripada et al., 2019) because they do not perform feature selection.

Since our primary goal is interpretation obtained from variable selection, we look at support recovery as a measure of performance, computing both recall and precision for $\boldsymbol{\beta}$, where recall is defined as

$$\frac{\text{TP}}{\text{TP} + \text{FN}},$$

and precision is defined as

$$\frac{\text{TP}}{\text{TP} + \text{FP}},$$

where TP denotes the number of true nonzero coefficients that are in the estimated support, FP denotes the number of true zero coefficients that are in the estimated support, and FN denotes the number of true nonzero coefficients that are not in the estimated support.

We also assess out-of-sample prediction performance using out-of-sample classification accuracy for binary responses and using the correlation between our predictions $\hat{\boldsymbol{y}}$ and the observed values in the test set, for the ease of comparison with the neuroimaging literature, which uses this measure (e.g., Sripada et al. (2020); Hsu et al. (2018)). We plot these metrics

against SNR for linear models and against Bayes Error (BE) for logistic models, given by

$$\text{SNR} = \sigma^{-2} \operatorname{Var}_{\boldsymbol{Z}} (\boldsymbol{Z}\boldsymbol{\beta}) \tag{2.3}$$

and

$$\text{BE} = E_{\boldsymbol{Z}} \left[\min \left(\text{logit}^{-1} (\boldsymbol{Z}\boldsymbol{\beta}), 1 - \text{logit}^{-1} (\boldsymbol{Z}\boldsymbol{\beta}) \right) \right]. \tag{2.4}$$

In all of our simulations with continuous responses, we set the error variance $\sigma^2 = 1$. When working with semi-synthetic data, where the design is based on real data, as discussed in Section 2.4.2, we compute the expectation and variance with respect to the empirical distribution of the training data.

2.4.1 Experiment I: Fully Synthetic Data

In this experiment, we generate fully synthetic designs $\left(A^{(i)}, X^{(i)}\right)_{i=1}^{N}$. We set the number of observations $N = 1000$ and the number of communities $K = 10$, with 5 nodes each for a total of $n = 50$ nodes. These are on the order of what is expected in neuroimaging settings, albeit on the low end, to accommodate running a large number of simulations. All unique entries of $A^{(i)}$ and $X^{(i)}$ are drawn independently from a standard normal distribution.

We specify $\boldsymbol{\beta}$ by first fixing its support and then setting all nonzero entries to the same constant α, which will be varied to control SNR. We form feature groups according to either the NBG or EBG scheme as described in Section 2.2.2. We then select either one or five groups to be active and take their union to be the active feature set. For NBG, the selected groups are $\left\{G^{(1)}\right\}$ or $\left\{G^{(1)}, G^{(2)}, G^{(3)}, G^{(4)}, G^{(5)}\right\}$. For EBG, the selected groups are $\left\{G^{(1,1)}\right\}$ or $\left\{G^{(1,1)}, G^{(3,1)}, G^{(3,2)}, G^{(4,4)}, G^{(6,5)}\right\}$. For EBG, we chose these five to include both diagonal and off-diagonal cells and to vary the amount of overlap. We always set the intercept $\mu = 0$. Finally, we draw each $y^{(i)}$ according to either a linear model $y^{(i)} = \boldsymbol{Z}^{(i)}\boldsymbol{\beta} + \epsilon^{(i)}$, where each $\epsilon^{(i)}$ is an independent standard normal, or a logistic model, where each $y^{(i)}$ is an independent Bernoulli random variable with success probability $\text{logit}^{-1}\left(\boldsymbol{Z}^{(i)}\boldsymbol{\beta}\right)$. For each realization, the training and the test set have the same design matrix Z and $\boldsymbol{\beta}$, and differ only in their responses $\boldsymbol{y}$, which are drawn independently. NetCov is fit with the true group structure, since we treat it as known.

Results in Fig. 2.5 show that, as expected, support recovery generally improves with increasing SNR, and NetCov yields superior support recovery, especially on recall in low and intermediate SNR regimes. NetCov is also generally superior to the LASSO on precision, although in high SNR settings the pattern is sometimes decreasing. While this at first may seem surprising, this is a result of the parameter tuning process. Both NetCov and

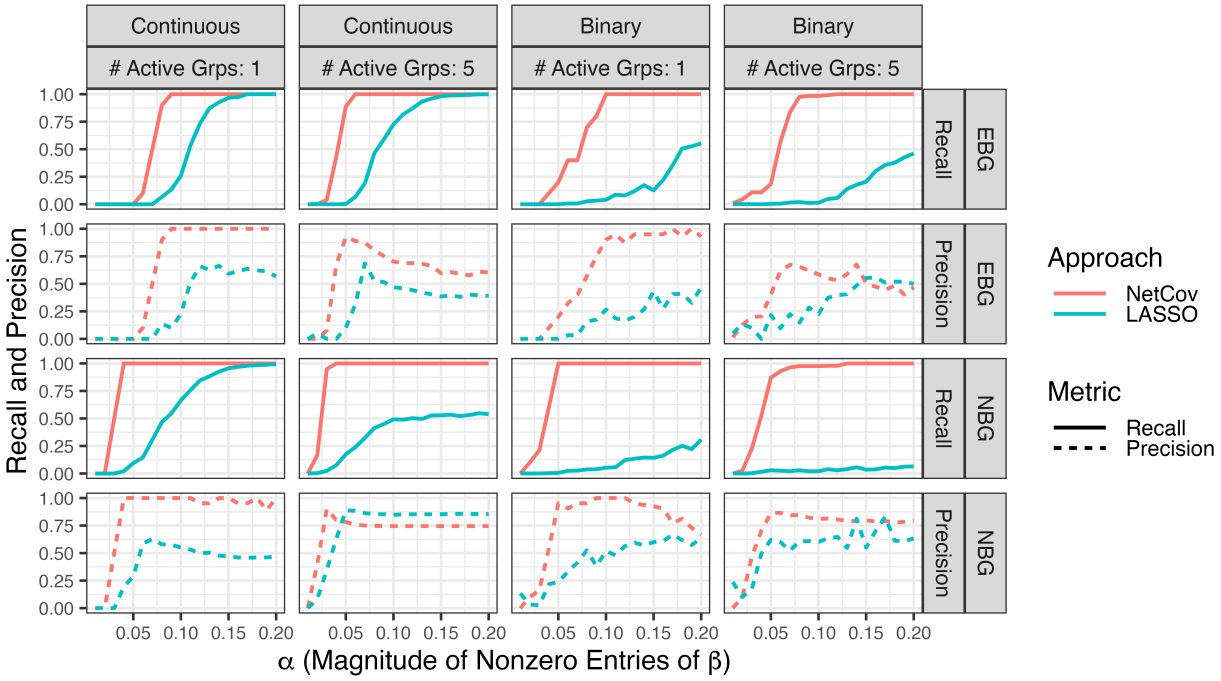

Figure 2.5: Support recovery in Experiment I: recall and precision as a function of nonzero coefficient magnitude α for NetCov (red) and LASSO (blue). Each of the four columns corresponds to either continuous or binary response and either 1 or 5 active groups. Each of the four rows corresponds to either EBG or NBG and either support recovery or precision for $\boldsymbol{\beta}$.

LASSO incur bias due to the use of penalization. There is a tendency for cross-validation to choose a small value of λ in order to reduce this bias, but this comes at the cost of selecting inactive features which harms precision (but not recall). In Appendix A.1, the receiver operating characteristic curves that characterize behavior along the entire λ path show that NetCov:EBG generally dominates the LASSO. See also Wang et al. (2020a) for a discussion of circumstances in which growing signal strength does not yield improved support recovery, chiefly due to effect size heterogeneity.

Out-of-sample prediction performance in the continuous and binary cases is shown in Fig. 2.6. Consistent with the improved support recovery, out-of-sample prediction for NetCov is generally superior to the LASSO. This suggests that when the additional structure imposed by NetCov is present in the data, NetCov will yield not only superior support recovery and interpretability relative to the LASSO, but also improved out-of-sample prediction performance. This improvement comes at a cost of trading off additional flexibility of the LASSO, which is an advantage in less structured models.

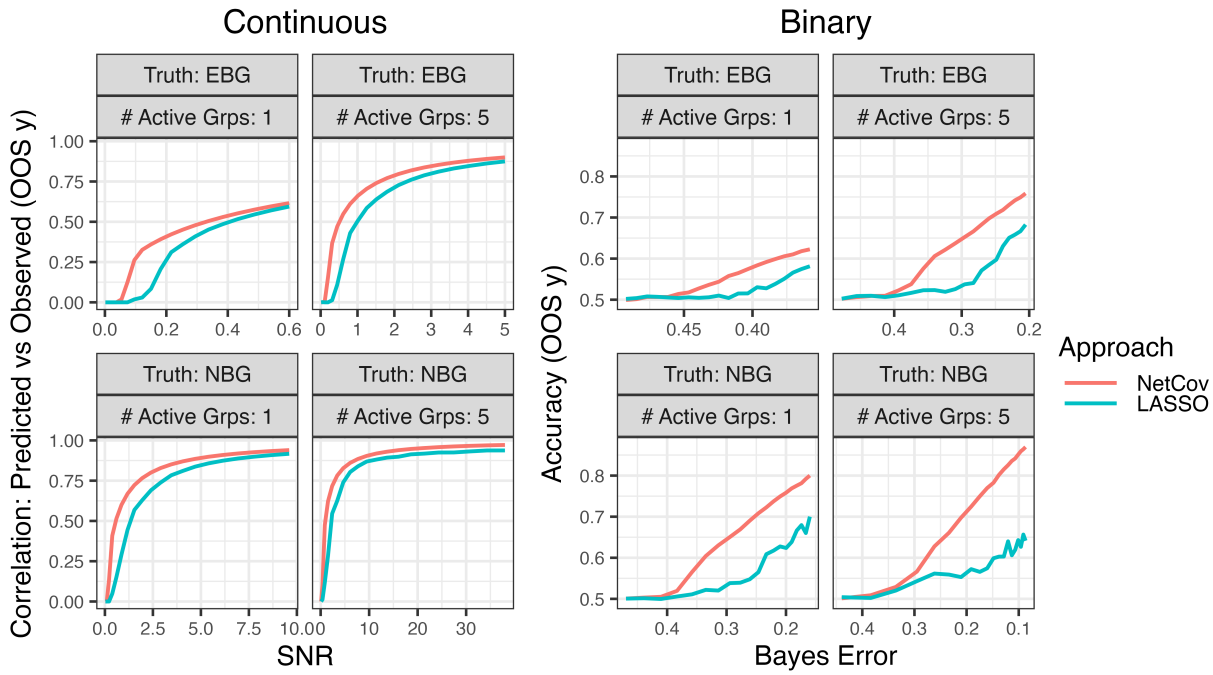

Figure 2.6: Out-of-sample prediction performance in Experiment I as a function of problem difficulty (SNR for continuous response and Bayes error for binary) for NetCov (red) and LASSO (blue). Note the horizontal scale is different in every panel

2.4.2 Experiment II: Semi-Synthetic Data

In this experiment, we use data from the neuroimaging application as the design matrix; see Section 2.5 for more details regarding this dataset. We have 785 observations in the training set and 96 observations in the test set. Each network in the sample has 236 nodes, and each node is assigned to one of 13 communities. The names of these communities and the number of nodes in each are given in Table 2.1. In addition to signed, weighted edges in the adjacency matrices, we also have a single continuous covariate associated with each node. The edge weights represent functional connectivity in resting state fMRI and the covariate is measured during a working memory task-based fMRI session; see Section 2.5 for details. Like in Experiment I, these networks are undirected without self-loops.

In order to keep the intercept at 0 as in Experiment I, we center the columns of the training design matrix and subtract these column means from the test design matrix. In all other respects, this experiment is identical to Experiment I in Section 2.4.1, where we specify the support of $\boldsymbol{\beta}$ to involve either one or five groups under either the EBG of NBG schemes. This semi-synthetic experiment involves real-world data with unknown dependence structure for A and X, but we draw responses $\boldsymbol{y}$ from our model with known $\boldsymbol{\beta}$, which allows

Brain System Name	Number of Nodes
Sensomotor Hand	30
Sensomotor Mouth	5
Cingulo-Opercular Task Control	14
Auditory	13
Default Mode	58
Memory	5
Visual	31
Frontoparietal Task Control	25
Salience	18
Subcortical	13
Ventral Attention	9
Dorsal Attention	11
Cerebellar	4

Table 2.1: Systems (communities) in the Power parcellation and number of nodes in each (Power et al., 2011).

us to assess support recovery and other performance metrics.

In Fig. 2.7 we see the results for support recovery of $\boldsymbol{\beta}$. Somewhat surprisingly, NetCov does not show a consistent improvement over LASSO, and while at least for EBG, recall increases appreciably with growing signal strength, precision remains poor. Performance of NBG is strikingly poor. This pattern is present in both the continuous and binary response cases. We believe that this is due to the presence of very large groups (especially for NBG), which are in turn due to the cardinality of communities as presented in Table 2.1. The failure of NetCov to perform accurate support recovery limits its competitiveness for prediction in both the continuous and binary cases is depicted in Fig. 2.8. To overcome these challenges, we modify our parcellation to avoid the problems described above when we conduct Experiment III, described below in Section 2.4.3.

2.4.3 Experiment III: Semi-Synthetic Data with Smaller Communities

As seen in Experiment II above, the parcellation from our application is problematic for NetCov, especially for NBG. This is because most of the feature groups have more predictors than there are observations. For NBG, this is true for *all* groups—the NBG group based on the smallest community, with 4 nodes, has $\binom{4}{2} + 4 \times 232 + 4 = 938$ predictors, and $N_{\text{train}} = 785$—and the orthonormalization discussed in Section 2.3.2 results in all of the feature groups being functionally identical. For EBG, there are 11 groups with more predictors

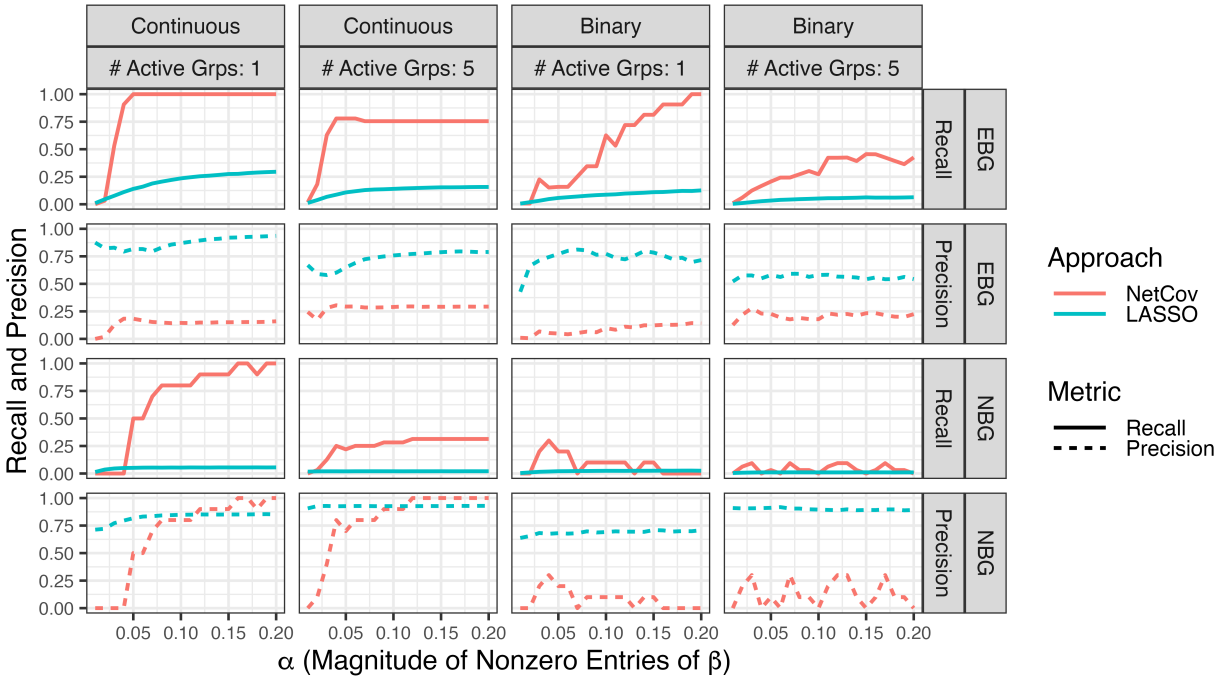

Figure 2.7: Support recovery in Experiment II: recall and precision as a function of nonzero coefficient magnitude α for NetCov (red) and LASSO (blue). Each of the four columns corresponds to either continuous or binary response and either 1 or 5 active groups. Each of the four rows corresponds to either EBG or NBG and either support recovery or precision for $\boldsymbol{\beta}$.

than observations, and so selecting any of these is problematic. The covariance structure within the groups is nontrivial, and this is problematic for some of our larger groups because of the way the penalty Eq. (2.2) accounts for group sizes.

As a simple remedy, in this experiment we randomly break up the large communities into smaller pieces until we arrive at a parcellation that has 50 communities where 15 of the communities have 4 nodes, 34 communities have 5 nodes, and a single community has 6 nodes. This modification does not change the overall covariance structure of Z, but it does change the intra-group covariance structure and also puts all groups on roughly equal footing in terms of size. We repeat the procedure described in Section 2.4.2 but with these new community assignments.

Support recovery with this new parcellation is depicted in Fig. 2.9. As we see, this new parcellation generally restores the enhanced performance of NetCov for EBG, especially for recall at low signal strength regimes. NetCov's EBG variant generally outperforms or is comparable to the LASSO with respect to out-of-sample performance for both continuous and binary responses as seen in Fig. 2.10. NBG remains limited, unless the sample size

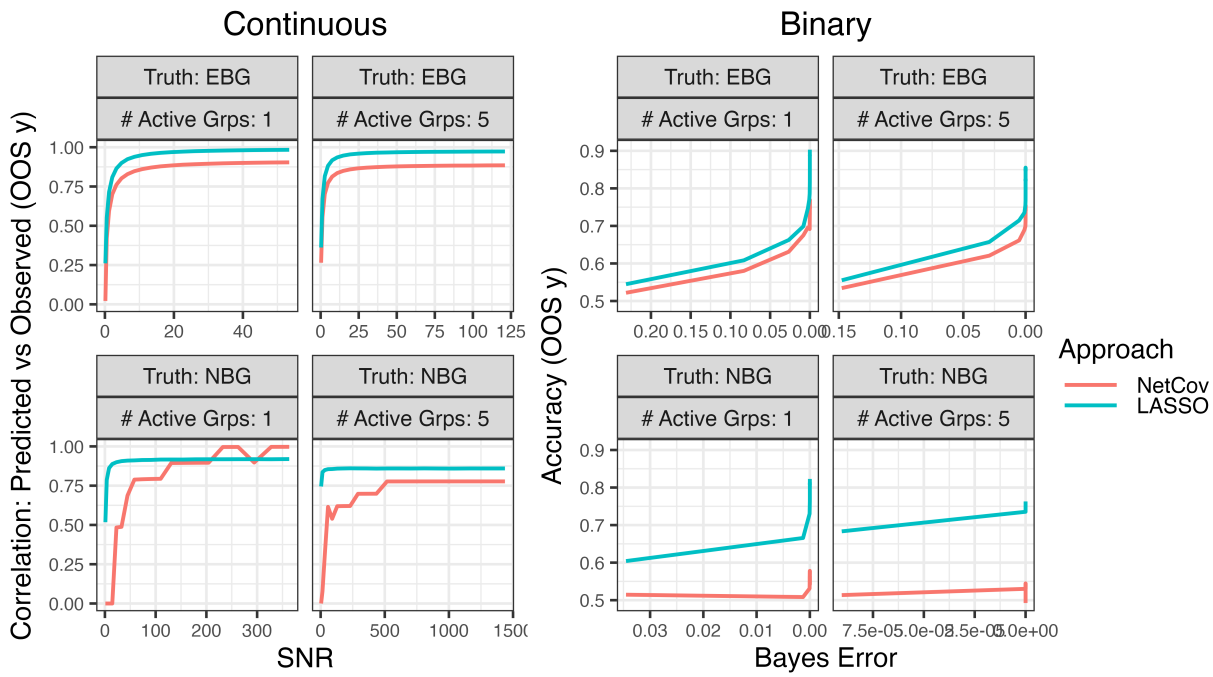

Figure 2.8: Out-of-sample prediction performance in Experiment II of NetCov (red) and LASSO (blue), as a function of problem difficulty (SNR for continuous response; Bayes error for binary). Note that the horizontal scale is different in every panel.

is very large relative to the number of features or we add another penalty to encourage within-group sparsity.

2.5 Application to Neuroimaging Data

We demonstrate the utility of our method by applying it to a subset of data from the Human Connectome Project (HCP; Van Essen et al., 2013) obtained and processed by the lab of our collaborator (see Acknowledgments). In brief, each participant contributes an observation $\left(A^{(i)}, X^{(i)}, y^{(i)}\right)$ which comprises functional connectivity from resting state data, activation during a working memory task, and a variety of behavioral measures, respectively. There are 881 participants that have complete data for all the measures we consider. We partition these into a training set of size 785 participants and a test set of size 96. This particular training/test split corresponds to the partitions used in Sripada et al. (2019), which was constructed to avoid any twins or sets of familially related individuals appearing in *both* the training and test sets. We describe each component of the observations in more detail below.

For $A^{(i)}$ and $X^{(i)}$, spatial locations of nodes as well as their community assignments were defined according to the "Power parcellation" (Power et al., 2011), mentioned earlier in the

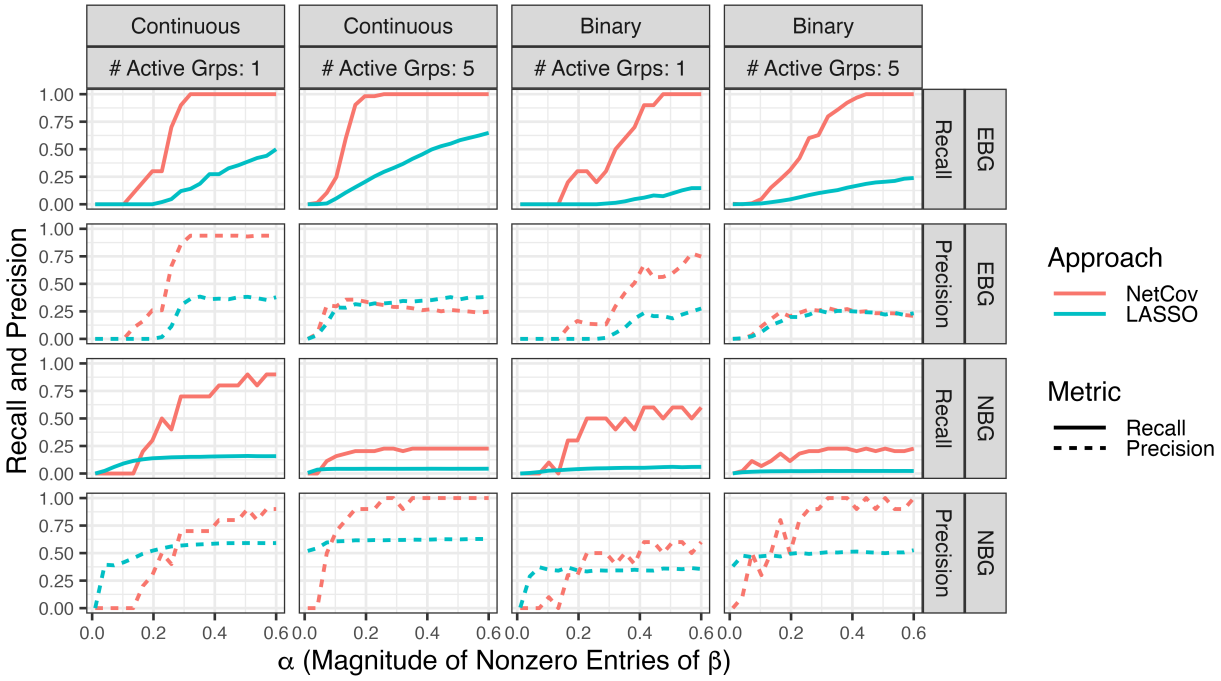

Figure 2.9: Support recovery (as measured by recall and precision) in Experiment III by NetCov (red) and LASSO (blue) as a function of nonzero coefficient magnitude α. Each of the four columns corresponds to either continuous or binary response and either 1 or 5 active groups. Each of the four rows corresponds to either EBG or NBG and either support recovery or precision for $\boldsymbol{\beta}$.

text. This yields 264 nodes assigned to 14 communities. Since our grouping scheme assumes that the nodes in a given system are meaningfully related, we removed all nodes that were assigned the "unknown" community label. This left 236 nodes divided into 13 communities. The putative brain systems corresponding to the communities, and the number of nodes in each, are given in Table 2.1.

The connectivity data is a subset of that used in Sripada et al. (2019), which describes the processing pipeline in detail. In brief, resting state fMRI data was obtained for each participant in 4 different sessions (two back-to-back sessions per day across two days), from which connectivity measures $A^{(i)}$ are extracted. During a resting state fMRI scanning session, the participant's brain activity is indirectly measured at many thousands of voxels in the brain while they lie passively in the scanner. Each of the 236 nodes corresponds to a set of voxels. As part of a comprehensive preprocessing pipeline, average time courses are extracted for all voxels in the same node. Each entry of $A^{(i)}$ is taken to be the correlation between the average time series at two of these nodes, Fisher transformed to the real line, for the i-th participant.

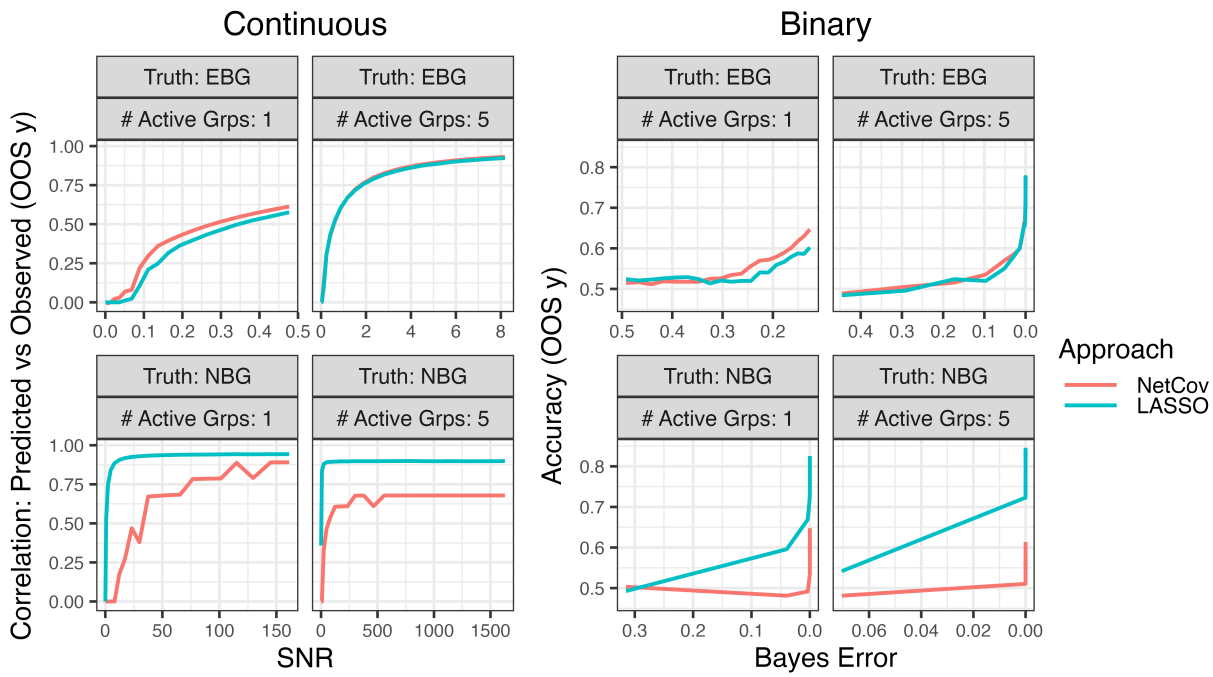

Figure 2.10: Out-of-sample prediction performance in Experiment III by NetCov (red) and LASSO (blue), as a function of problem difficulty (SNR for continuous response and Bayes error for binary). Note the horizontal scale is different in each panel.

We form $X^{(i)}$ by obtaining a single node covariate ($d = 1$) for each of the 236 nodes from brain activity during the "N-back" task (Barch et al., 2013), which is designed to measure working memory. This data is a superset[1] of that used in Panigrahi et al. (2023a), which describes the data further. During the "N-back" task, participants view a sequence of images that are presented in blocks. Each block corresponds to a condition. In the 0-back condition, participants are asked to judge whether each presented item is the same as what they saw at the beginning of the block. During the 2-back condition, participants indicate whether each item is the same as what they saw two trials previous. Of course, the 2-back condition is more demanding with respect to working memory. In an attempt to isolate brain activity specific to working memory, activation during the 0-back condition is subtracted from activation during the 2-back condition. This removes activity common to both conditions (e.g., visual processing, motor activity to push a button, etc.). This 2-back minus 0-back contrast was computed by our collaborator using in-house processing scripts that use SPM12. These contrasts were initially computed at the voxel level, but averaged values were extracted for each of the 236 nodes using the MarsBar utility (Brett et al., 2002).

For $y^{(i)}$, we separately consider various phenotypes provided by our collaborator. Several

[1]In their analysis, Panigrahi et al. (2023a) used only data from the 785 participants in our training set.

of the responses reflect performance during tasks from the NIH Toolbox (Hodes et al., 2013), namely the (i) pattern comparison processing speed test (Processing Speed), (ii) flanker inhibitory control and attention test (Flanker), and (iii) list sorting working memory test (List Sorting), as well as (iv) performance on the Penn Progressive Matrices task (PMAT) (Bilker et al., 2012). Other responses capture the five facets of the NEO personality assessment (McCrae and Costa, 2004): (v) openness to experience (NEO: O), (vi) conscientiousness (NEO: C), (vii) extraversion (NEO: E), (viii) agreeableness (NEO: A), and (ix) neuroticism (NEO: N). We also considered (x) accuracy on the "N-back" task described above (Working Memory). Finally, we considered (xi) a measure of general cognitive ability (GCA) obtained from factor analysis described in Sripada et al. (2020). All candidate responses are continuous, so we use a linear model for the response, i.e.,

$$y^{(i)} = \boldsymbol{Z}^{(i)}\boldsymbol{\beta} + \epsilon^{(i)}.$$

In addition to covariates of interest, there are several nuisance covariates. These are age (conventionally represented by a linear and a quadratic term), handedness, gender, brain size, which multiband reconstruction algorithm was used, and movement of the head during resting state scan ("meanFD") along with its square. We control for these by first fitting a regression model to the training data that includes only the nuisance covariates and predicts the brain features and phenotypes. Using the coefficients learned in the training data, we then subtract the nuisance-predicted values from both the training and test data and use this corrected data for all downstream tasks.

While we can assess out-of-sample performance on the test set, we cannot assess support recovery directly as we did in our simulation studies, since the true $\boldsymbol{\beta}$ is unknown. As in the simulations, we assess performance by computing the correlation coefficient between predicted and observed responses on the test data, in order to facilitate comparisons with the neuroimaging literature, which frequently uses this measure.

We compare our approach with both the conventional LASSO as well as connectome predictive modeling (CPM). CPM is a popular and relatively simple technique for predicting scalar responses using brain connectivity data. In brief (see Shen et al., 2017, for a more detailed explanation), it is a three-step procedure that involves marginal feature selection, feature aggregation, and then estimation of a regression model. The first step consists of correlating screening between edge weights of the connectome and the response y. Next, all edges that pass screening are aggregated into two summary measures, by summing together those that are positively correlated with the response, and those that are negatively correlated. In the last step, a simple regression model is fit with these two summary measures as

predictors. While there are a number of variations (involving, e.g., robust regression), we opt for this simple pipeline and use $p < 0.01$ as the threshold for feature selection. One noteworthy limitation of CPM, in contrast to NetCov, is that it operates only on edge weights and does not make use of node covariates. For NetCov and LASSO, we use the same approach as in Section 2.4, including cross-validation on the training data to select $\hat{\lambda}$.

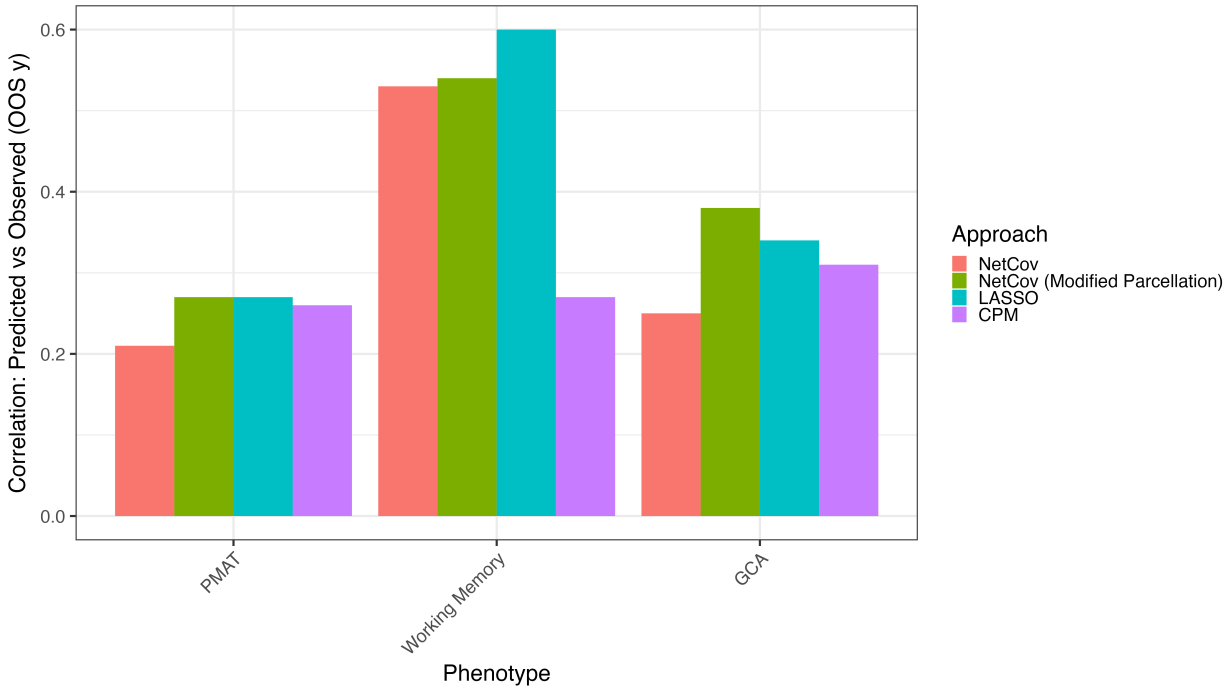

Figure 2.11: Out-of-sample correlation for selected phenotypes in application to human neuroimaging data.

We show results for selected phenotypes in Fig. 2.11. Tuning of λ yields poor performance for other phenotypes, and we present these results in Appendix A.2. For the original Power parcellation community assignments, NetCov is reasonably competitive with, although typically slightly worse than, the LASSO. When we use the modified Power parcellation community assignments as described in Section 2.4.3, this small difference disappears. Good predictive performance of the LASSO is expected since it imposes less structure on the estimated coefficients, at the cost of less interpretability.

We present the estimated coefficients for β_A and β_X associated with GCA in Fig. 2.12 and present results for other phenotypes in Appendix A.2 for both NetCov (with EBG grouping) and the LASSO. The edges used by CPM for GCA are given in Fig. 2.13. These figures illustrate the degree to which NetCov yields more interpretable solutions, implicating only a small number of brain systems through both edge weights and node covariates. In contrast, non-zero LASSO coefficients are scattered across brain systems and moreover appear not

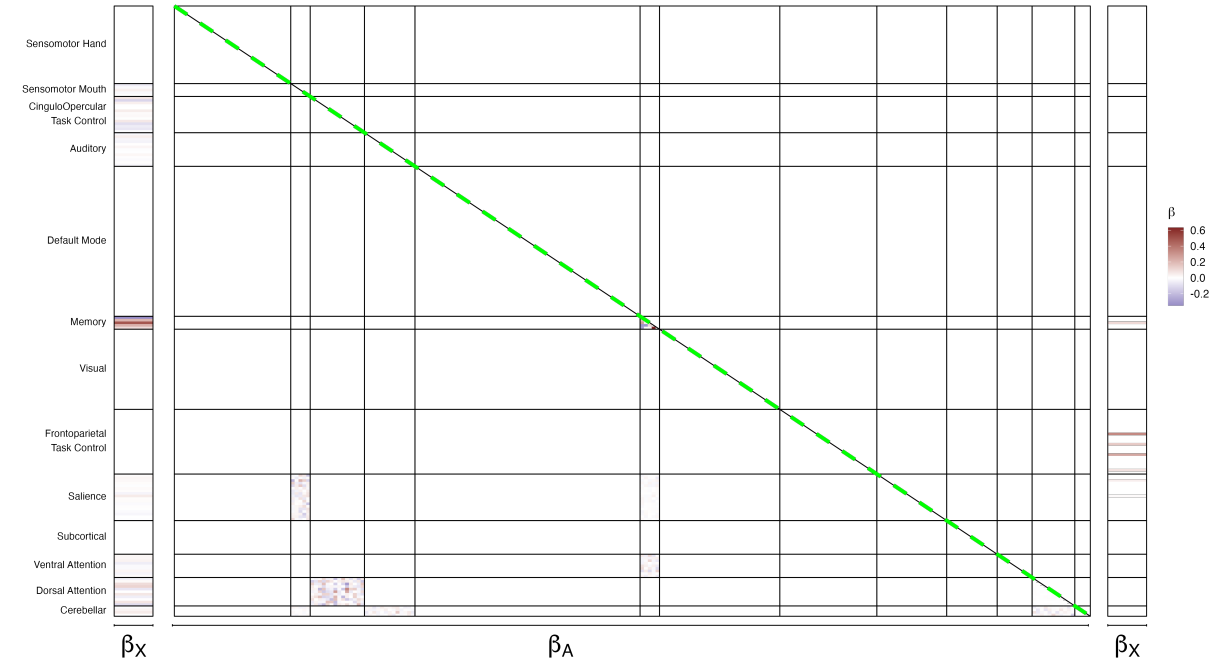

Figure 2.12: Visualization of $\boldsymbol{\beta}$ coefficients with GCA as response. Coefficients from NetCov with EBG are presented at left (β_X) and on the lower triangle (β_A). Coefficients from LASSO are presented at right (β_X) and on the upper triangle (β_A). Solid lines depict boundaries of the Power parcellation.

to include any edges. Of particular note, some of the systems that NetCov selects (e.g., Cingulo-Opercular Task Control and Dorsal Attention), are consistent with previous reports in the literature that studied the neural basis of general intelligence, closely related to our GCA factor (Duncan et al., 2000; Tong et al., 2022).

2.6 Summary and Discussion

We have introduced NetCov, a method for prediction from samples of weighted networks and node covariates, which offers a novel way to discover relevant brain systems when both edge and node covariates are present. We proposed two approaches, node-based and edge-based (NBG and EBG), depending on what we believe is central to the mechanism of the underlying model, and implemented both through constructing appropriate group penalties. As we saw in simulations, when the assumptions underlying these grouping schemes are met, NetCov yields both superior predictive performance and better support recovery. In our application to human neuroimaging data, NetCov offers comparable predictive performance to the LASSO but dramatically enhances interpretability of findings.

There are some settings that present challenges for NetCov. As demonstrated in Experiment II in Section 2.4.2, large groups with more features than observations can be problematic, especially for NBG, and as the number of nodes n grows, the size of NBG groups can grow as fast as $O(n^2)$. EBG suffers less from this and this issue can be addressed by the use of smaller communities, but constructing them of course needs to be application-specific. This challenge, faced by the group LASSO generally, is discussed in Bühlmann and van de Geer (2011, page 250, see citations within), where one proposed remedy is to use the smoothed group LASSO instead. An alternative approach which may preserve interpretability while overcoming these limitations, is the use of "bi-level" selection methods (Huang et al., 2009; Breheny and Huang, 2009; Breheny, 2015) or the sparse group LASSO (Friedman et al., 2010a; Cai et al., 2019). These methods select both feature groups and smaller subsets of features within each group; examples of this approach include Relión et al. (2019); Richie-Halford et al. (2021). Another alternative is the ridged group LASSO (Simon and Tibshirani, 2012), which performs selection at the group level and scales the penalty applied to standardized groups on their "effective degrees of freedom."

Future work in the neuroimaging setting includes application of this method to other datasets, especially where the outcome is a binary indicator of a degenerative disease process, since this is one of the chief motivations for both the NBG and EBG schemes. In addition, there are numerous alternative candidates for node covariates, including structural measures like gray matter volume or surface measures like cortical thickness. Finally, there are also other candidate edge weights, including those obtained from diffusion weighted imaging (DWI) which is believed to capture anatomical, rather than functional, features of brain connectivity. If multiple types of edge weights were present, both NBG and EBG could be extended to accommodate their simultaneous use.

While the selection of interpretable feature groups is a useful step on the path to a better understanding of complex phenomena, NetCov does not directly provide inference. Although data splitting can be employed, wherein a subset of features is selected in training data, and then an appropriately restricted model is fit in test data with conventional inferential tests, recent results from post-selection inference for the group LASSO enable inferential tests at the level of feature groups (Yang et al., 2016) as well as individual coefficients within feature groups (Panigrahi et al., 2023a). These methods could be developed for NetCov to perform inference in addition to variable selection.

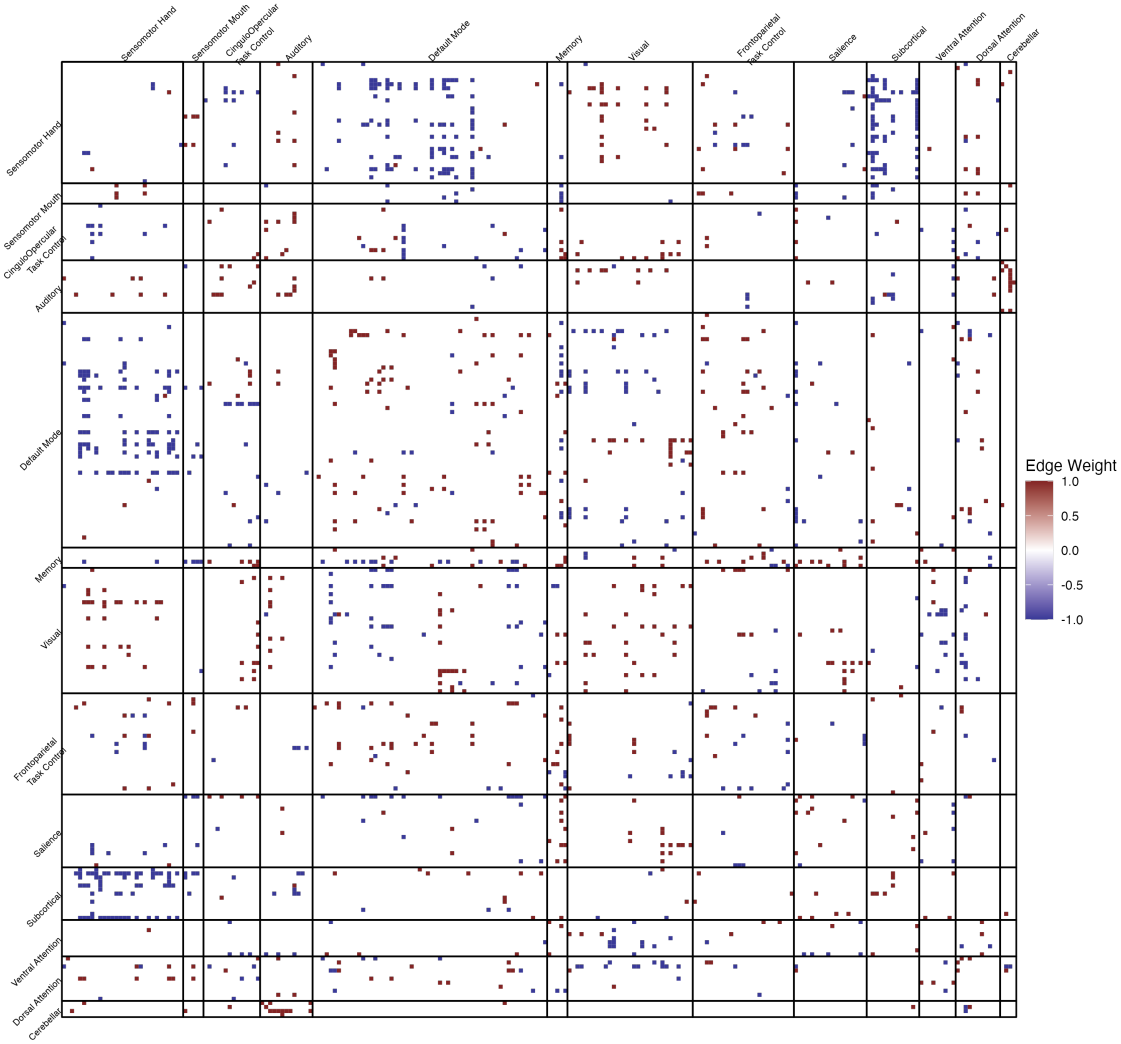

Figure 2.13: Edges selected by CPM, colored by the sign of their association with GCA. Solid lines depict boundaries of the Power parcellation.

CHAPTER 3

Computational Inference for Directions in Canonical Correlation Analysis

3.1 Introduction

Canonical Correlation Analysis (CCA) is a classical technique (Hotelling, 1935) for identifying linear relationships among two sets of variables. Informally, it learns a linear transformation for each set of variables such that the transformed variables are maximally correlated with one another. CCA has seen a recent resurgence of interest with the growing popularity of multi-modal datasets, which have two (or more) sets of variables collected on the same individual. For example, brain imaging studies often capture various brain-related metrics as well as phenotypic and behavioral measures on the same individuals, and it is natural to ask how these are related; see, e.g., Wang et al. (2020b) for a review of CCA in neuroscience applications. In many of these applications, at least one of the two datasets is high-dimensional, i.e., with more variables than observations. In those settings, data reduction can be applied upstream (e.g., with principal components analysis (PCA) as in Smith et al., 2015; Goyal et al., 2022) to render the problem low-dimensional, or regularized forms of CCA which seek sparsity can be used (Witten et al., 2009; Xia et al., 2018).

In order to build intuition for CCA, it is useful to view it as a generalization of regression. Consider the classical regression model

$$Y = X\beta + \epsilon,$$

where $Y \in \mathbb{R}^N, X \in \mathbb{R}^{N\times p}, \epsilon \in \mathbb{R}^N$. We can obtain an estimator $\hat{\beta}$ by solving an optimization problem, with the classic least squares estimator given by

$$\hat{\beta}_{\text{OLS}} = \underset{\beta}{\text{argmin}} \|Y - X\beta\|_2^2. \tag{3.1}$$

However, we can obtain a closely related estimator by solving a different optimization problem:

$$\hat{\beta}_{\text{CCA}} = \underset{\beta}{\text{argmax}} \, \text{Corr}\left(X\beta, Y\right), \tag{3.2}$$
$$\text{s.t.} \;\; \text{Var}\left(X\beta\right) = 1,$$

where the constraint serves only to make the solution unique by requiring that the predictions have unit variance. The estimators obtained from (3.1) and (3.2) will satisfy $\hat{\beta}_{\text{OLS}} \propto \hat{\beta}_{\text{CCA}}$, i.e., we will have found the *direction* in $\mathbb{R}^p$ of our regression coefficients. Now suppose we measure q different responses for each observation, so that $Y \in \mathbb{R}^{N \times q}$ is now a matrix rather than a vector, with $q > 1$. A natural analogue to the problem in (3.2) is the following optimization problem:

$$\left(\hat{\beta}_{\text{CCA}}, \hat{\gamma}_{\text{CCA}}\right) = \underset{\beta,\gamma}{\text{argmax}} \, \text{Corr}\left(X\beta, Y\gamma\right), \tag{3.3}$$
$$\text{s.t.} \;\; \text{Var}\left(X\beta\right) = \text{Var}\left(Y\gamma\right) = 1,$$

where again the constraint serves to make the solution unique up to sign flipping. The solution to this problem gives exactly the two sets of canonical directions defined by CCA.

Despite its long history in the statistical literature, CCA is generally deployed as an exploratory tool, without a readily available set of tools for inference. Indeed, in a recent review of CCA aimed at neuroscientists (Wang et al., 2020b), CCA is categorized as a method focused on estimation (in contrast to prediction or inference); the authors go on to emphasize that if inference does occur, it is often constrained to testing a global null corresponding to no correlation between the datasets. While exploratory analysis is useful, there is growing appreciation in applications that the discoveries of CCA analyses may be illusory (Dinga et al., 2019). The development of valid inferential tools for this setting is vital in order to appropriately characterize uncertainty so that truly interesting phenomena may be distinguished from optimistic over-fitting.

A natural starting point is to assess whether the estimated correlation among the canonical variates is significantly different from zero. While parametric tests for this hypothesis have been developed (For a review, see §11.3.6 of Muirhead, 1982), in practice they are sensitive to violations of assumptions (Winkler et al., 2020), and so non-parametric approaches have become more popular, e.g., as used in Witten et al. (2009). While the use of permutation-based procedures to assess the canonical correlations is quite common in the neuroimaging literature, recent work (Winkler et al., 2020) shows that a simple permutation procedure

yields inflated Type I for all canonical correlations beyond the first; they also introduce a more nuanced permutation procedure that addresses this issue and also allows for the incorporation of nuisance covariates. Inference for the canonical correlations in the high dimensional setting remains an active research area, too (McKeague and Zhang, 2022).

If the hypothesis of zero canonical correlations can be rejected, the next natural question to ask is which variables have significant coefficients in the canonical directions; as in regression, one may only be interested in inference on individual coefficients if the F-test rejects the global null of all coefficients being zero. However, inference on canonical directions remains an elusive goal in CCA. For example, Rosa et al. (2015) applied sparse, non-negative CCA to pairs of brain images obtained via arterial spin labeling under different pharmacological challenges, and performed a permutation-based procedure to assess the significance of the canonical correlations, but the authors acknowledge that they are unable to perform inference at the level of individual features. Indeed, a recent review of CCA intended for neuroscientists (Zhuang et al., 2020) concludes by acknowledging that, at the time of writing, inferential tools are only available for the canonical correlations rather than the canonical directions, and that the development of inference for directions at the level of individual features would benefit future neuroscience research.

The importance of developing these tools is highlighted in a recent paper that studies the stability of CCA (Helmer et al., 2020). In this work, the authors consider the sampling error of the estimated CCA directions, but rather than considering individual coordinates, they focus on the angle between an estimated direction and the true direction. While stability in this sense is important if a canonical direction is to be interpreted holistically, it does not afford inference at the level of individual coefficients. Indeed, there may be cases where an overall canonical direction is "unstable," but a small number of coordinates of interest can be reliably differentiated from 0. The authors provide guidance for what sort of stability can be expected as the ratio of samples to features varies. While Helmer et al. (2020)'s approach is generally numerical and relies on the generation of synthetic data, recent theoretical developments in Bykhovskaya and Gorin (2023) echo these empirical findings, characterizing limiting angles between true and estimated canonical variates in terms of these ratios.

In the absence of rigorous statistical tests, practitioners have developed ad hoc methods to characterize uncertainty about canonical directions. These assessments generally involve some form of resampling, such as bootstrap or permutation tests, and the statistical properties of these approaches have not been well studied. These procedures do not not always take the form of hypothesis testing. For example, Alnæs et al. (2020); Linke et al. (2021) performed CCA to obtain canonical directions, but subjected the resultant directions to ICA to aid interpretability (Miller et al., 2016). They then used a resampling procedure in order

to assess the stability of their final results, but they do not perform inference for individual coordinates of the canonical directions. In other cases, variants of the bootstrap are used in order to construct confidence intervals for individual coordinates of the directions (Xia et al., 2018). One shortcoming of these approaches is that their statistical properties (e.g., control of Type I error) are not well-studied.

In this work, we help to fill this gap and provide concrete guidance regarding statistical inference for canonical directions. We propose a method, combootcca, and provide evidence for its validity. We also review several other approaches based on our review of the literature and compare their performance to combootcca empirically in terms of coverage, statistical validity (control of Type I error), and power (control of Type II error), in a variety of simulation studies that range from simple but carefully controlled to complex but realistic. We then illustrate our recommended methodology in an application to neuroimaging data.

3.1.1 CCA: Population Model and Estimation

Let $x \in \mathbb{R}^p$ and $y \in \mathbb{R}^q$ be random vectors with covariances Σ_x and Σ_y, respectively, and let $\Sigma_{xy} = \mathrm{Cov}(x, y)$ denote their cross-covariance. Informally, the initial goal of CCA is to identify a pair of vectors $\beta \in \mathbb{R}^p, \gamma \in \mathbb{R}^q$ such that $x^\intercal \beta$ is maximally correlated with $y^\intercal \gamma$. In order to fix the scale of β and γ, we require that $\beta^\intercal \Sigma_x \beta = \gamma^\intercal \Sigma_y \gamma = 1$, i.e., the transformed variables have unit variance. We can then proceed to find additional pairs of vectors subject to an orthogonality constraint. Formally, CCA involves solving the following sequence of optimization problems for $k = 1, 2, \ldots, K$, where K is the rank of Σ_{xy}:

$$\begin{aligned} (\beta_k, \gamma_k) &= \underset{(b_k, g_k)}{\mathrm{argmax}}\, b_k^\intercal \Sigma_{xy} g_k, \\ \text{s.t. } b_k^\intercal \Sigma_x b_l &= g_k^\intercal \Sigma_y g_l = \mathbb{I}(k = l). \end{aligned} \tag{3.4}$$

We shall refer to $\beta_1, \beta_2, \ldots, \beta_K$ and $\gamma_1, \gamma_2, \ldots, \gamma_K$ as the *canonical directions* associated with x and y, respectively. It is often convenient to gather the canonical directions into a matrix, writing

$$\begin{aligned} B &= \begin{bmatrix} \beta_1 & \beta_2 & \ldots & \beta_K \end{bmatrix} \in \mathbb{R}^{p \times K}, \\ \Gamma &= \begin{bmatrix} \gamma_1 & \gamma_2 & \ldots & \gamma_K \end{bmatrix} \in \mathbb{R}^{q \times K}, \end{aligned}$$

which permits us to rewrite the constraint in (3.4) as $B^\intercal \Sigma_x B = \Gamma^\intercal \Sigma_y \Gamma = I_K$. We refer to the transformed variables as the *canonical variates* and denote them by

$$c = \begin{bmatrix} c_1 & c_2 & \dots c_K \end{bmatrix}^\intercal = \begin{bmatrix} x^\intercal \beta_1 & x^\intercal \beta_2 & \dots & x^\intercal \beta_K \end{bmatrix}^\intercal \in \mathbb{R}^K,$$
$$d = \begin{bmatrix} d_1 & d_2 & \dots d_K \end{bmatrix}^\intercal = \begin{bmatrix} y^\intercal \gamma_1 & y^\intercal \gamma_2 & \dots & y^\intercal \gamma_K \end{bmatrix}^\intercal \in \mathbb{R}^K.$$

The correlations between the canonical variates are the *canonical correlations* and are denoted by

$$\rho = \begin{bmatrix} \rho_1 & \rho_2 & \dots & \rho_K \end{bmatrix}^\intercal \in [0,1]^K, \quad R = \text{diag}(\rho),$$

where $R \in \mathbb{R}^{K \times K}$ is a diagonal matrix with $R_{k,k} = \rho_k$.

If the population cross-covariance Σ_{xy} is known and the covariances Σ_x and Σ_y are known and non-singular, then all of the components of the CCA solution can be obtained as follows as presented in Muirhead (1982). First, perform the singular value decomposition

$$\Sigma_x^{-1/2} \Sigma_{xy} \Sigma_y^{-1/2} = USV^\intercal.$$

The diagonal entries of S are the canonical correlations, i.e., $R = S$. The canonical directions can be obtained as $B = \Sigma_x^{-1/2} U$ and $\Gamma = \Sigma_y^{-1/2} V$. This formulation will be especially convenient for our numerical studies discussed in Section 3.3, as for a fixed generative covariance structure we can recover the true CCA solution.

In practice, we observe the data matrices $X \in \mathbb{R}^{N \times p}$ and $Y \in \mathbb{R}^{N \times q}$, where N is the number of observations, from which we construct CCA estimators $\hat{B}, \hat{\Gamma}$, and $\hat{\rho}$. Without loss of generality, suppose that both X and Y have been column-centered. There are many options available for estimating the covariance matrices (Fan et al., 2016); classical CCA is based on the empirical covariances, which are the maximum likelihood estimators. Replacing the covariances Σ_x, Σ_y, and Σ_{xy} with their estimated counterparts $\hat{\Sigma}_x, \hat{\Sigma}_y$, and $\hat{\Sigma}_{xy}$ an estimated CCA solution can be obtained using the SVD approach described above, but a more popular approach (also used by R's `cancor` function) is due to Björck and Golub (1973) and is briefly summarized in Golub and Van Loan (2013, p. 331). In our notation, it first performs (thin) QR decompositions

$$X = Q_X R_X, \; Y = Q_Y R_Y,$$

followed by the SVD decomposition

$$Q_X^\intercal Q_Y = USV^\intercal.$$

Object	Dimension	Description
x	p	Random Vector
y	q	Random Vector
X	$N \times p$	Data matrix
Y	$N \times q$	Data matrix
ρ	K	Canonical Correlations (vector)
R	$K \times K$	Canonical Correlations (diagonal matrix)
B	$p \times K$	Canonical Directions
Γ	$q \times K$	Canonical Directions

Table 3.1: CCA notation.

The estimated canonical correlations are given by the diagonal of S, and the estimated canonical directions can be obtained as $\hat{B} = R_X^{-1}U$ and $\hat{\Gamma} = R_Y^{-1}V$. Because of the special form of R_X and R_Y, it is not necessary to explicitly invert them, and the canonical directions can instead be found by back-solving. However, we must note that this approach satisfies a related, but distinct, constraint from that presented in (3.4): the resulting canonical variates are empirically uncorrelated with one another, but rather than having unit variance, they have unit norm, which we can see by observing

$$\|X\hat{\beta}_k\|_2^2 = \|Q_X R_X \hat{\beta}_k\|_2^2 = \|Q_X R_X R_X^{-1} U e_k\|_2^2 = e_k^\intercal U^\intercal Q_X^\intercal Q_X U e_k = 1,$$

with an analogous result for $\hat{\Gamma}$. Recalling that we assume X and Y are column-centered, the empirical variance of these canonical variates will be $(N-1)^{-1}$. This is problematic for inference on the canonical directions, but can be readily remedied by multiplying $\hat{B}$ and $\hat{\Gamma}$ by $\sqrt{N-1}$, which will set the canonical variates' empirical variance to 1 and put the estimated canonical directions on a scale free of N.

3.1.2 Inverting the CCA Model

Given the population covariance matrix and assuming that both Σ_x and Σ_y are non-singular, there is a straightforward mapping from the covariance Σ to the CCA solution R, B, Γ as given in Section 3.1.1. However, in some settings (for example, Simulation III discussed in Section 3.3.4), it is more convenient to directly specify the CCA parameters (R, B, Γ). Unfortunately, these parameters do not typically uniquely identify Σ, but given a set of CCA parameters R, B, Γ, one can define a covariance matrix Σ to match them. Assume that both B and Γ have full column rank and that the diagonal of R is a descending sequence of unique and strictly positive canonical correlations. Without loss of generality, assume that $p \geq q = K$. Recall that the covariances Σ_x and Σ_y satisfy $B^\intercal \Sigma_x B = \Gamma^\intercal \Sigma_y \Gamma = I_K$. Since Γ

is a square matrix of full column rank, we can just solve for $\Sigma_y = (\Gamma^{-1})^\intercal \Gamma^{-1}$. If $p = q$, we can do the same for B, but if $p > q$, then B is a rectangular matrix with full column rank, so let B^+ denote the Moore-Penrose inverse of B, satisfying $B^+B = I_K$. Then we can find a solution $\Sigma_x = (B^+)^\intercal B^+$. However, Σ_x is a $p \times p$ square matrix of rank at most $K < p$. This rank deficiency may be undesirable, and so in simulation studies we remedy this by inflating the trailing eigenvalues of Σ_x to make it full rank. Specifically, we replace the $(K+1)$th through pth eigenvalues of Σ_x by linearly interpolating between the Kth eigenvalue and 0 (without including the endpoints). This Σ_x has full rank and will still satisfy $B^\intercal \Sigma_x B = I_K$. Following Chen et al. (2013), we can take $\Sigma_{xy} = \Sigma_x B R \Gamma^\intercal \Sigma_y$ and $\Sigma_{yx} = \Sigma_{xy}^\intercal$. Finally, we can assemble Σ as

$$\Sigma = \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_y \end{bmatrix}.$$

3.1.3 Inference for CCA

In practice, inference for both the correlations ρ and the directions B and Γ may be of interest. Inferential tools for the correlations ρ are relatively better developed, especially when the number of observations N exceeds both p and q. Hotelling's classic 1936 paper discusses two different tests for "complete independence" which are equivalent to testing $\rho_1 = 0$ (and therefore all subsequent canonical correlations as well). Anderson's 2003 textbook, in Section 12.4.1, reviews approaches to inference both on ρ globally and on its individual elements. Inference for ρ in the high-dimensional case is an area of current research, see for example McKeague and Zhang (2022).

Permutation tests are also a popular approach for testing ρ, applied by Witten et al. (2009) and appearing in subsequent applied work, e.g., Alnæs et al. (2020). Recently, Winkler et al. (2020) carefully studied the use of permutation tests for the canonical correlations, noting the practical difficulties of parametric inference, demonstrating shortcomings of permutation tests as typically applied and proposing a remedy.

Inference for the canonical directions, however, has received less recent attention in the statistical literature. Anderson (1999) reviews several decades of work on the limiting distributions of the estimated canonical directions, finds faults with all past results, and derives the limiting distribution of $\hat{B}$ and $\hat{\Gamma}$ when x and y are jointly normal *and* $p = q$. The latter is an especially significant limitation for the neuroimaging applications we have in mind.

Laha et al. (2021) developed asymptotically exact inference for the first canonical direction and its associated correlation in the high-dimensional setting under the assumption of sparsity. Their approach is based on a debiasing argument that yields an asymptotically normal distribution. However, their method is unable to provide results for any canonical

directions except for the first, and moreover it is not clear that their method is applicable in low-dimensional settings with moderate N. We will compare our proposal to this approach in simulations.

Recently, the permutation method of Winkler et al. (2020), originally proposed for canonical correlations, was extended to perform inference on the canonical loadings (Zhang et al., 2022). The canonical loadings are related to the canonical directions but are not the same: they are the correlations between the original variables and the canonical variates. This approach optionally obtains a permutation distribution corresponding to the maximal absolute value over all coordinates for each direction, resulting in "built-in" family-wise error control, if desired, achieved by comparing the empirical values to the distribution of the maxima. However, this may be more stringent control than necessary and result in the loss of power. In addition, it only tests whether a given coordinate is nonzero and does not provide confidence intervals, which limits interpretability of this inference approach.

3.2 Bootstrap Inference for CCA

In the absence of analytical tools for inference, practitioners of CCA often apply bootstrap to characterize the uncertainty in CCA estimates, especially the canonical directions. While at a high level applying bootstrap sounds straightforward, it has been implemented in a variety of ad hoc ways in practice, without any clarity about the relative merits of different approaches. Next, we discuss two important aspects of designing a reliable bootstrap algorithm for CCA and compare multiple alternatives for each. In Section 3.3.5, we will demonstrate the empirical consequences of these choices, and show that they can have substantial impacts on the statistical properties of the procedure. Our newly proposed algorithm for bootstrapping CCA, informed by these results, is stated formally at the end of this section. We call it combootcca (COMputational BOOTstrap for CCA).

3.2.1 Alignment of Bootstrap Replicates

One significant hurdle when using resampling-based methods with CCA is the issue of alignment. There is a fundamental sign ambiguity in CCA, just like with any estimated direction, since $\text{Corr}\left(X^\intercal\beta, Y^\intercal\gamma\right) = \text{Corr}\left(X^\intercal\left(-\beta\right), Y^\intercal\left(-\gamma\right)\right)$. When considering a single CCA solution, this ambiguity is of little consequence, but when multiple bootstrap realizations are drawn, there is a need to "align" the estimates obtained from the resampled data so that they can be meaningfully combined and compared to the estimate obtained from original data. In addition to sign ambiguity, the canonical directions may change order, especially if the as-

sociated canonical correlations are not well separated, or be rotated in some way. Without alignment, all of these can substantially inflate our estimates of variance, leading to conservative inference with little power. On the other hand, a "strict" alignment strategy may lead to underestimating variance and result in invalid inference, so a balance is needed.

The general goal of alignment is to learn and apply, for each resampled estimate $\left(\tilde{R}^\star, \tilde{B}^\star, \tilde{\Gamma}^\star\right)$, a transformation f to obtain an aligned version $\left(\hat{R}^\star, \hat{B}^\star, \hat{\Gamma}^\star\right)$. We will learn f by comparing $\left(\tilde{R}^\star, \tilde{B}^\star, \tilde{\Gamma}^\star\right)$ to $\left(\hat{R}, \hat{B}, \hat{\Gamma}\right)$, where the latter is a "reference" solution which typically corresponds to the values estimated on the full data. While the canonical correlations are invariant to the scale of the predictors, the canonical directions are not. Because our original variables may be on different scales which may unduly influence alignment, prior to learning f we multiply[1] the rows of $\hat{B}$, $\hat{\Gamma}$, $\hat{B}^\star$, and $\hat{\Gamma}^\star$ by the standard deviations of their corresponding samples, which transforms the canonical direction matrices to what they would have been if all variables had been standardized prior to CCA and prevents coordinates corresponding to variables with low empirical variance from dominating the alignment. We consider several possible alignment strategies, described below. The empirical results on the significant impact of alignment on statistical validity will be presented in Section 3.3.5.

Identity (no alignment). This alignment "strategy" is included for baseline assessment of the need for alignment. Here f is the identity operator, and given that it does not even correct for the sign ambiguity, we expect it to perform poorly.

Sign Flip. This alignment strategy deals with sign ambiguity by flipping the signs of canonical directions and has been used in practice (e.g., in McIntosh, 2021; Nakua et al., 2023). That is, the transformation f right-multiplies matrices B and Γ by a signature matrix H (i.e., a diagonal matrix with entries in ± 1). To decide which directions need to be flipped, we construct the similarity matrix G_B by calculating pairwise cosine similarity between the columns of the matrices $\hat{B}$ and $\tilde{B}^\star$ (after they have been standardized as described above), where the cosine similarity between a vector u and v is $u^\intercal v \left\|u\right\|_2^{-1} \left\|v\right\|_2^{-1}$. We similarly obtain G_Γ as the pairwise cosine similarities between the columns of the standardized matrices $\hat{\Gamma}$ and $\tilde{\Gamma}^\star$. Finally, we average them together and obtain $G = \frac{1}{2}\left(G_B + G_\Gamma\right)$. We then construct H by setting $H_{k,k} = \operatorname{sign}\left(G_{k,k}\right)$, i.e., if the averaged cosine similarity is negative, we flip the sign, and if it is positive, we do not. The aligned solution is given by $\left(\hat{R}^\star, \hat{B}^\star, \hat{\Gamma}^\star\right) =$

[1] At first glance, division may seem more appropriate, but if we increase the variance associated with say the first coordinate of x, then its associated canonical direction coordinate must *decrease* to offset this, so the remedy is indeed to *multiply* the canonical directions (which is then tantamount to having divided the variables in the first place).

$\left(\tilde{R}^\star, \tilde{B}^\star H, \tilde{\Gamma}^\star H\right)$.

Assignment via Weighted Hungarian Algorithm. This alignment strategy is allowed to both change the ordering as well as the associated signs of the directions. Thus, we find a transformation matrix T that can be written as the product of a permutation matrix P and a signature matrix H. To the best of our knowledge, this approach is novel and has not been considered in the literature. We treat alignment as an assignment problem, where the task is to optimally assign the columns of $\left(\tilde{R}^\star, \tilde{B}^\star, \tilde{\Gamma}^\star\right)$ to the columns of $\left(\hat{R}, \hat{B}, \hat{\Gamma}\right)$ while allowing sign flipping. After adjusting for scaling as described above, we construct the similarity matrix G based on the cosine similarity in the same manner that we did for the "Sign Flip" alignment strategy. In order to incorporate information about the (empirical) canonical correlations into the alignment strategy, we weight the matrix of cosine similarities by the square roots of the canonical correlations. That is, we construct $G_{\text{w}} = (\hat{R})^{\frac{1}{2}} G (\tilde{R}^\star)^{\frac{1}{2}}$. We then take the entry-wise absolute value to obtain $G_{\text{wPos}} = \text{abs}\,(G_{\text{w}})$. Then, we find a permutation matrix P that maximizes $\text{trace}\,(G_{\text{wPos}} P)$ by using the Hungarian Algorithm (Kuhn, 1955) as implemented in the `R` package `RcppHungarian` (Silverman, 2022). We then apply this permutation to the original matrix and extract the signs of the diagonal entries as $\text{diag}\,(H) = \text{sign}\,(\text{diag}\,(G_{\text{w}} P))$. Our transformation can then be written as $T = PH$, and our aligned solution for this bootstrap realization is $\left(\hat{R}^\star, \hat{B}^\star, \hat{\Gamma}^\star\right) = \left(\tilde{R}^\star P, \tilde{B}^\star T, \tilde{\Gamma}^\star T\right)$. This is the approach that is used in combootcca, our recommended approach that is described in Section 3.2.3, and it is an integral part of Algorithm 1.

Rotation via Procrustes. This alignment strategy involves finding orthogonal matrices that can be applied to the directions of $\tilde{B}^\star$ and $\tilde{\Gamma}^\star$, respectively. Since CCA is symmetric, we learn separate transformations for $\tilde{B}^\star$ and $\tilde{\Gamma}^\star$, although in principle one could learn the transformation for $\tilde{B}^\star$ and apply it to both $\tilde{B}^\star$ and $\tilde{\Gamma}^\star$, or vice versa. Formally, we find

$$T_B = \underset{Q : Q^\intercal Q = I}{\text{argmin}} \left\| \hat{B} - \tilde{B}^\star Q \right\|_F .$$

The solution to this problem is well-known (Schönemann, 1966), and a concise treatment is given in Golub and Van Loan (2013, p. 328): take the singular value decomposition $\left(\tilde{B}^\star\right)^\intercal \hat{B} = USV^\intercal$, and the optimal solution is given by $T_B = UV^\intercal$; an analogous approach can be used to find T_Γ. Once we have obtained the orthogonal matrices T_B and T_Γ, we align through right multiplication, i.e., taking $\left(\hat{B}^\star, \hat{\Gamma}^\star\right) = \left(\tilde{B}^\star T_B, \tilde{\Gamma}^\star T_\Gamma\right)$. At first glance, this may suggest that it will serve to rotate the canonical directions, but this will generally not be the case. While it is generally true that *left* multiplication by an orthogonal matrix will

apply a rotation to the columns of a given matrix, *right* multiplication by an orthogonal matrix will not generally perform a rotation of the columns. Instead, it will apply a rotation to the rows. In the special case where the matrix being transformed is itself orthogonal, then this will also effect a rotation of the columns, but in CCA, the canonical directions are generally not orthogonal matrices. Recall that the population quantities and related estimates satisfy $B^\intercal \Sigma_x B = \Gamma^\intercal \Sigma_y \Gamma = I_K$, i.e., they have orthonormal columns with respect to the inner product induced by Σ_x and Σ_y, respectively. B (or Γ) will be orthogonal in the usual sense only in the special case that Σ_x (or Σ_y) is equal to I. One additional consequence is that the matrix of canonical correlations will generally no longer be diagonal, i.e., $\tilde{R}^\star T_B$ or $\tilde{R}^\star T_\Gamma$ may have non-zero entries that are not on the diagonal. There are examples in the literature where it seems a Procrustes alignment is used. For example, Xia et al. (2018) describes a matching procedure and cites Mišić et al. (2016), which in turn refers to McIntosh and Lobaugh (2004), which proposes a Procrustes alignment. Notably, Xia et al. (2018) uses a version of sparse CCA (Witten et al., 2009) which assumes that the covariances Σ_x and Σ_y are identity, in which case the associated directions are orthogonal.

The preceding strategies have been described in an order that proceeds from the least to most strict alignment. In general, if we are too "gentle" with our alignment, we will sacrifice power, as much of our apparent variability will simply be due to ambiguities that arise due to poor alignment. On the other hand, if we are too strict, we will sacrifice control of Type I error as we will underestimate variance. A balance must be struck, and as we shall see later, the weighted Hungarian approach appears to do just this.

3.2.2 Constructing Confidence Intervals from the Bootstrap Distribution

There are multiple options available for how to construct confidence intervals from (aligned) bootstrap replicates. One option is the so-called "normal bootstrap." In this approach, one assumes that the distribution of the estimator is normally distributed and centered at its true value. Then a confidence interval centered at the estimate can be constructed using the variance of the bootstrap replicates and the quantiles of the normal distribution. This approach was used to construct confidence intervals for the coefficients in Nakua et al. (2023); Mišić et al. (2016); Kebets et al. (2019) (the latter two analyses used Partial Least Squares [PLS], a method closely related to CCA). It is also the approach offered by the `RGCCA` package (Girka et al., 2023) (but which does not offer options regarding alignment strategy).

An alternative is the so-called "percentile" bootstrap (Efron and Tibshirani, 1993), which directly uses quantiles of the empirical distribution of the bootstrapped replicates in order to

construct a confidence interval, without relying on normality. As a consequence, this interval is not guaranteed to be centered at the original estimate. As we shall see, this approach will generally perform better as it does not explicitly assume that the initial estimator is an unbiased proxy for the truth, nor does it make strong assumptions about the normality of the sampling distribution. Although a fairly common approach in general, it does not seem popular in CCA applications.

3.2.3 The Combootcca Algorithm

Based on the careful investigation of the different options discussed above and empirical comparisons between them in Section 3.3.5, we present our final algorithm for a computational bootstrap approach to inference for CCA directions (combootcca). To the best of our knowledge, this is a novel algorithm which has not been previously considered in the literature for inference on CCA directions. We present this approach in Algorithm 1. In brief, given data matrices $X \in \mathbb{R}^{N \times p}$ and $Y \in \mathbb{R}^{N \times q}$, we first fit the CCA model using the approach described in Section 3.1.1 and obtain estimates $\left(\hat{R}, \hat{B}, \hat{\Gamma}\right)$. These quantities will subsequently be used as a "reference solution" for alignment. Then we draw bootstrap samples from the rows of the data matrices (with replacement), to obtain $X^\star$ and $Y^\star$, and the corresponding bootstrapped CCA estimates $\left(\tilde{R}^\star, \tilde{B}^\star, \tilde{\Gamma}^\star\right)$. Then, we align the solutions using the weighted Hungarian strategy described above to obtain $\left(\hat{R}^\star, \hat{B}^\star, \hat{\Gamma}^\star\right) = \left(\tilde{R}^\star P, \tilde{B}^\star T, \tilde{\Gamma}^\star T\right)$. We record the values, and we repeat the procedure `nBoots` times (we use 1×10^4 repetitions for all results presented below), each time drawing a new sample with replacement $(X^\star, Y^\star)$. To obtain $1 - \alpha$ level confidence intervals for $(\beta_i)_j$ or $(\gamma_i)_j$, we find the empirical $\frac{\alpha}{2}$ and $1 - \frac{\alpha}{2}$ quantiles of the bootstrapped estimates at that coordinate and set these as the end points of our confidence intervals. We use the `boot` package (Davison and Hinkley, 1997; Canty and Ripley, 2022) along with custom alignment code to carry out the above procedure.

3.3 Empirical Results on Synthetic Data

In this section, we apply combootcca and several alternative methods described in Section 3.3.1 below to three different simulation settings, with data drawn from different generative models. Since in simulations we have the ground truth available, we will compare the different confidence intervals on coverage, length, and rejection rate if used to test the hypothesis of a parameter being equal to zero (rejected if zero is not in the interval). The confidence level is fixed at 95% in all cases.

All of our synthetic data will be drawn from a multivariate normal distribution with

Algorithm 1 The combootcca algorithm for confidence intervals for CCA directions.

Require: $X \in \mathbb{R}^{N \times p}, Y \in \mathbb{R}^{N \times q}, \alpha \in (0,1), \texttt{nBoots} \in \mathbb{N}$

$K \leftarrow \textsc{Minimum}(p, q)$

$\left(\hat{R}, \hat{B}, \hat{\Gamma}\right) = \mathrm{CCA}(X, Y)$

$\hat{B}^\star \leftarrow 0^{p \times K \times \texttt{nBoots}}$

$\hat{\Gamma}^\star \leftarrow 0^{q \times K \times \texttt{nBoots}}$

for $k \leftarrow 1, \texttt{nBoots}$ **do**

$(X^\star, Y^\star) \leftarrow \textsc{Resample with Replacement}(X, Y)$

$\left(\tilde{R}^\star, \tilde{B}^\star, \tilde{\Gamma}^\star\right) \leftarrow \mathrm{CCA}(X^\star, Y^\star)$

$G_B \leftarrow \textsc{CosineSim}(\hat{B}, \tilde{B}^\star)$ ▷ Compute column-wise cosine similarity

$G_\Gamma \leftarrow \textsc{CosineSim}(\hat{\Gamma}, \tilde{\Gamma}^\star)$

$G \leftarrow \frac{1}{2}\left(G_B + G_\Gamma\right)$ ▷ Average cosine similarity for B and Γ

$G_{\mathrm{w}} \leftarrow (\hat{R})^{\frac{1}{2}} G (\tilde{R}^\star)^{\frac{1}{2}}$ ▷ Weight by canonical correlations

$G_{\mathrm{wPos}} \leftarrow \mathrm{abs}\left(G_{\mathrm{w}}\right)$ ▷ Take entry-wise absolute value

$P \leftarrow \textsc{Hungarian}(G_{\mathrm{wPos}})$ ▷ Permutation P maximizes $\mathrm{trace}\left(G_{\mathrm{wPos}} P\right)$

$H \leftarrow \textsc{SignDiag}(G_{\mathrm{w}} P)$ ▷ H reflects any negative cosine similarities

$\hat{B}^\star[:, :, k] \leftarrow \tilde{B}^\star P H$

$\hat{\Gamma}^\star[:, :, k] \leftarrow \tilde{\Gamma}^\star P H$

end for

$\hat{B}^\star_{\mathrm{Lower}} \leftarrow 0^{p \times K}$

$\hat{B}^\star_{\mathrm{Upper}} \leftarrow 0^{p \times K}$

for $i \leftarrow 1, p$ **do**

for $j \leftarrow 1, K$ **do**

$\hat{B}^\star_{\mathrm{Lower}}[i, j] \leftarrow \textsc{Quantile}(\frac{\alpha}{2}, \hat{B}^\star[i, j, :])$

$\hat{B}^\star_{\mathrm{Upper}}[i, j] \leftarrow \textsc{Quantile}(1 - \frac{\alpha}{2}, \hat{B}^\star[i, j, :])$

end for

end for

$\hat{\Gamma}^\star_{\mathrm{Lower}} \leftarrow 0^{q \times K}$

$\hat{\Gamma}^\star_{\mathrm{Upper}} \leftarrow 0^{q \times K}$

for $i \leftarrow 1, q$ **do**

for $j \leftarrow 1, K$ **do**

$\hat{\Gamma}^\star_{\mathrm{Lower}}[i, j] \leftarrow \textsc{Quantile}(\frac{\alpha}{2}, \hat{\Gamma}^\star[i, j, :])$

$\hat{\Gamma}^\star_{\mathrm{Upper}}[i, j] \leftarrow \textsc{Quantile}(1 - \frac{\alpha}{2}, \hat{\Gamma}^\star[i, j, :])$

end for

end for

return $\hat{B}^\star_{\mathrm{Lower}}, \hat{B}^\star_{\mathrm{Upper}}, \hat{\Gamma}^\star_{\mathrm{Lower}}, \hat{\Gamma}^\star_{\mathrm{Upper}}$

means $\mu_x = 0_p$ and $\mu_y = 0_q$, and the (joint) covariance Σ given by

$$\Sigma = \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_y \end{bmatrix}.$$

When determining coverage, we have to confront the fundamental sign ambiguity of CCA. Suppose the true canonical correlations ρ are all distinct and that we know the "true" canonical directions (B, Γ). Any pair $(BH, \Gamma H)$, where H is a signature matrix, *also* qualify as the "true" canonical directions. This ambiguity can lead to low coverage if not accounted for, so when evaluating coverage, we maximize it over all such signature matrices H (sign flips). The sign ambiguity is of no consequence when evaluating rejection rate, since that depends only on whether the interval contains zero. Note that we do not allow for reordering of directions when considering coverage, length, or rejection decisions.

3.3.1 Alternative Confidence Intervals for Canonical Directions

We next present several methods for obtaining confidence intervals corresponding to the elements of the canonical directions.

Asymptotic confidence intervals. Anderson (1999) derived the asymptotic distribution of the canonical directions in the case where x and y are jointly multivariate normal, $p = q$ and $\rho_1 > \rho_2 > \ldots > \rho_p > 0$. We shall later see empirically this result does not generalize to the setting $p \neq q$. The key result is a limiting normal distribution for the entries of $\hat{B}$ and $\hat{\Gamma}$ which can be used to construct asymptotic confidence intervals for each entry. For $\hat{B}$, this takes the form of

$$\sqrt{n}\left(\hat{B}_{ij} - B_{ij}\right) \xrightarrow{d} \mathcal{N}\left(0, \sigma^2_{B_{ij}}\right),$$

where

$$\sigma^2_{B_{ij}} = \frac{1}{2}B^2_{ij} + \left(1 - \rho^2_j\right) \sum_{\substack{k=1 \\ k \neq j}}^{p} \frac{\rho^2_k + \rho^2_j - 2\rho^2_k\rho^2_j}{\left(\rho^2_j - \rho^2_k\right)^2} B^2_{ik},$$

and analogous results can be obtained for the elements of $\hat{\Gamma}$. In practice, estimates have to be substituted for parameters in the expression for variance in order to obtain asymptotic confidence intervals. Extending Anderson's results to the case $p \neq q$ is non-trivial and is outside the scope of this work.

Regression-based confidence intervals. Given the intimate connection between regression and CCA discussed in Section 3.1, it is natural to consider adapting tools from regression

to provide inference for CCA. Recall the estimation strategy discussed in Section 3.1.1. Suppose that we have already obtained the estimated directions $\hat{\Gamma}$, and want to obtain estimates of the directions $\hat{B}$ using regression. First, suppose without loss of generality that all columns of data matrices below have been centered. Let

$$\tilde{\beta}_k = \underset{\beta}{\operatorname{argmin}} \|Y\hat{\gamma}_k - X\beta\|_2^2 .$$

This is the ordinary least squares estimator, which as we noted in Section 3.1 maximizes the correlation between $Y\hat{\gamma}_k$ and $X\beta$, and thus is a solution to the (unconstrained) CCA problem. In order to satisfy the constraint that $\tilde{\beta}_k^\intercal \hat{\Sigma}_x \tilde{\beta} = 1$, we can simply rescale and obtain $\hat{\beta}_k = \tilde{\beta}_k \left(\tilde{\beta}_k^\intercal \hat{\Sigma}_x \tilde{\beta}\right)^{-1/2}$, which will coincide with the solution we would have obtained with regular CCA; similar results can be obtained for $\hat{\Gamma}$ by symmetry. This least squares formulation of CCA was noted in Gao et al. (2017). Because we will depend upon the distribution of the regression-based estimators, we need them to be independent of the estimated directions which we treat fixed. To accomplish this, we propose to use this approach with sample splitting. First, partition the observations into two disjoint sets X_1, Y_1 and X_2, Y_2 and perform CCA on (X_1, Y_1) to obtain estimated directions $\hat{B}_1, \hat{\Gamma}_1$. Next, in the held out-data fix $\hat{\Gamma} = \hat{\Gamma}_1$ and use the above procedure to obtain $\hat{B}_2$, then fix $\hat{B} = \hat{B}_1$ and use the above procedure to obtain $\hat{\Gamma}_2$. Because they were obtained using regression, conditional on $\hat{\Gamma}_1$, the entries of $\hat{B}_2$ will each follow a (scaled) t-distribution, which we use to construct confidence intervals, e.g.,

$$\left(\hat{\beta}_k\right)_i \pm t_{N/2-q}^{(\alpha/2)} \hat{\sigma}_k \left(X_2^\intercal X_2\right)_{i,i}^{-1} ,$$

where $t_{N/2-q}^{(\alpha/2)}$ is the $\alpha/2$th quantile of the t distribution with $N/2 - q$ degrees of freedom, and $\hat{\sigma}_k$ is the square root of the estimated error variance in the kth regression model after rescaling. By symmetry, analogous confidence intervals can be obtained for Γ.

Debiased (Sparse) CCA Recent work by Laha et al. (2021) introduces a method for obtaining asymptotically exact inference for canonical directions in a high-dimensional regime. It works with sparse CCA, although to make a direct comparison in our setting we have to take the regularization parameter $\lambda = 0$, which may be outside the scope of their theoretical results. This approach relies on the characterization of the first canonical directions as the unique maximizers (modulo sign flipping) of a smooth function and uses a one-step correction to de-bias the (regularized) estimators; this de-biasing step is carefully accounted for in obtaining a limiting distribution. One limitation of this approach, however, is that it only provides results for the leading canonical directions (β_1 and γ_1),

and does not offer any inference for subsequent canonical directions. We use the function `give_SCCA` with default settings, available from the authors' package on GitHub at https://github.com/nilanjanalaha/de.bias.CCA, to obtain estimates of the variances and use these to construct confidence intervals. This function requires that we provide an estimate of the canonical directions: in their example, they use the result of the sparse CCA method of Mai and Zhang (2019), but since we do not require sparsity we simply use the (rescaled) estimates from `R`'s `cancor` function.

3.3.2 Simulation I: Synthetic Data with One Canonical Correlation

In our first simulation study, we consider the setting where there is a single non-zero canonical correlation, i.e., $K = 1$. We vary $(p, q) \in \{(10, 10), (100, 10)\}$, and $\rho_1 \in \{0.9, 0.5, 0.2\}$. In line with the simulation studies of Laha et al. (2021), we construct a sparse precision matrix for both x and y and then invert it to obtain (dense) covariance matrices Σ_x, Σ_y. The sparse precision matrix takes the initial form $\Omega_{i,j} = 1_{\{i=j\}} + 0.5 \cdot 1_{\{|i-j|=1\}} + 0.4 \cdot 1_{\{|i-j|=2\}}$. We then apply a modification to Ω to make specification of the canonical directions in Simulation II simpler. Specifically, for the Ω associated with x we place 0's everywhere but the diagonal in the floor (p) and floor $(p) + 1$ rows and columns, and we make an analogous modification of the Ω associated with y. This has the effect of breaking the marginal dependence between the first half of the coordinates and the latter half of the coordinates, and without this it is difficult to specify subsequent canonical directions (as we do in Simulation II, see Section 3.3.3) without running afoul of the orthogonality constraints. In Appendix B.1, we also show results when identity covariance matrices are used instead, and we obtain results similar to those presented below.

We consider both a "dense" and a "sparse" regime for the canonical directions. In the dense regime, the canonical directions are proportional to

$$\check{\beta}_1 = \begin{bmatrix} \mathbf{1}_{p/2}^\intercal & \mathbf{0}_{p/2}^\intercal \end{bmatrix}^\intercal$$
$$\check{\gamma}_1 = \begin{bmatrix} \mathbf{1}_{q/2}^\intercal & \mathbf{0}_{q/2}^\intercal \end{bmatrix}^\intercal,$$

whereas in the sparse regime they are proportional to

$$\check{\beta}_1 = \begin{bmatrix} \mathbf{1}_{2}^\intercal & \mathbf{0}_{p-2}^\intercal \end{bmatrix}^\intercal$$
$$\check{\gamma}_1 = \begin{bmatrix} \mathbf{1}_{2}^\intercal & \mathbf{0}_{q-2}^\intercal \end{bmatrix}^\intercal,$$

i.e., in the dense regime the first half of the coordinates are nonzero, whereas in the sparse regime only the first two coordinates are nonzero. In both cases, we then normalize to obtain $\beta_1 = \left(\check{\beta}_1^\intercal \Sigma_x \check{\beta}_1\right)^{-1/2} \check{\beta}_1$ and $\gamma_1 = \left(\check{\gamma}_1^\intercal \Sigma_y \check{\gamma}_1\right)^{-1/2} \check{\gamma}_1$. We fix $N = 1000$. In line with Chen et al. (2013), we construct the cross-covariance as $\Sigma_{xy} = \rho_1 \Sigma_x \beta_1 \gamma_1^\intercal \Sigma_y$. We draw 1000 replicates for each setting for each of the methods, except for that of Laha et al. (2021), which is appreciably slower than the others, where we instead use 100 replications. Since there are only two possible values for coordinates in our setup, for simplicity we examine statistical properties associated with the confidence intervals at just two coordinates for each vector: the last coordinate ($(\beta_1)_p$ and $(\gamma_1)_q$) which is always zero, and the first coordinate ($(\beta_1)_1$ and $(\gamma_1)_1$), which is always non-zero.

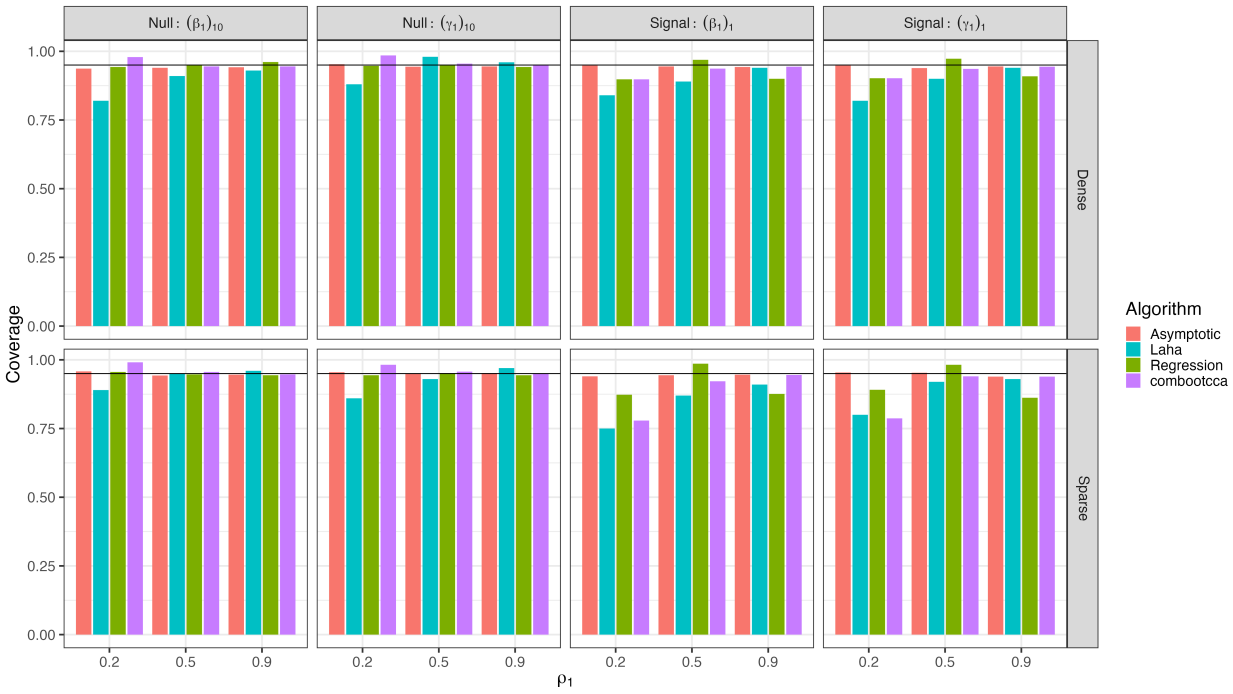

Figure 3.1: Coverage rates in simulation I for $p = q = 10$. The horizontal line indicates nominal 95% coverage.

We plot coverage and lengths of intervals at these representative coordinates for $p = q = 10$ in Figures 3.1 and 3.2, while coverage and lengths for $p = 100, q = 10$ are depicted in Figures 3.3 and 3.4, respectively. With balanced dimensions $p = q = 10$, the methods generally perform well, although all but the asymptotic approach fall short in their coverage of signals when ρ_1 is small. Notably, the asymptotic, regression, and combootcca methods attain nominal coverage of null coordinates, which is tantamount to valid control of Type I error, whereas the method of Laha et al. (2021) does not achieve Type I error control when $\rho_1 = 0.2$. The good performance of the asymptotic approach was expected since the $p = q$

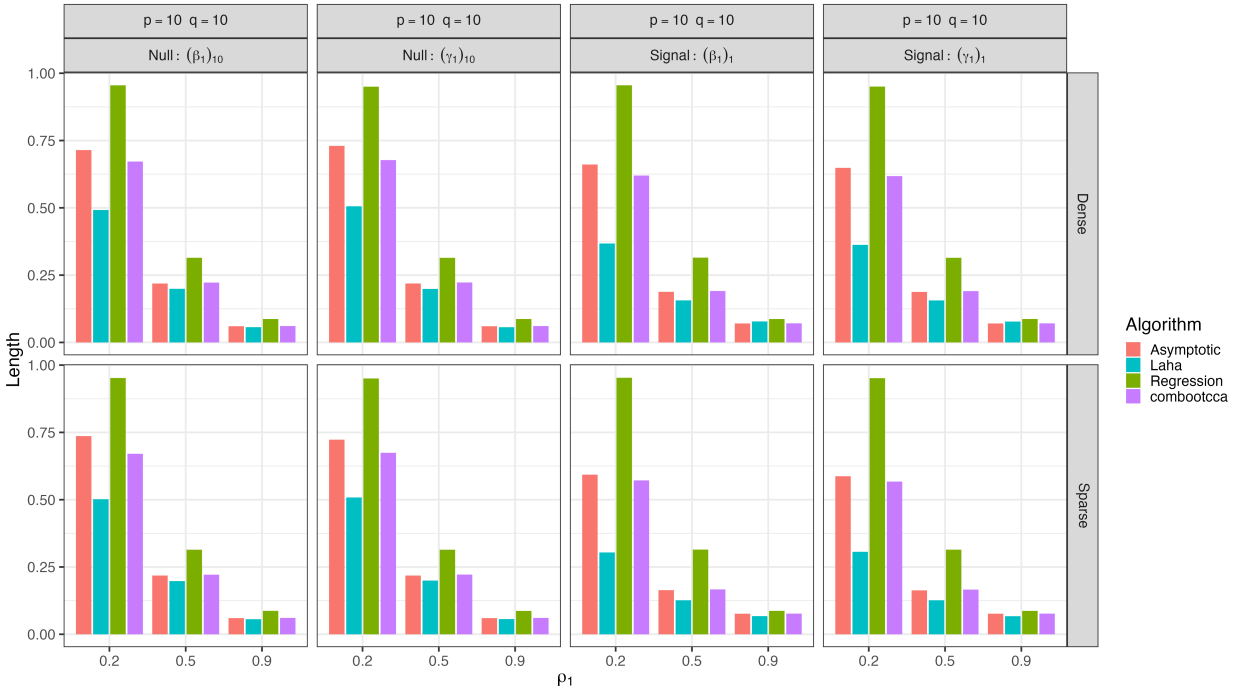

Figure 3.2: Lengths of confidence intervals in simulation I for $p = q = 10$.

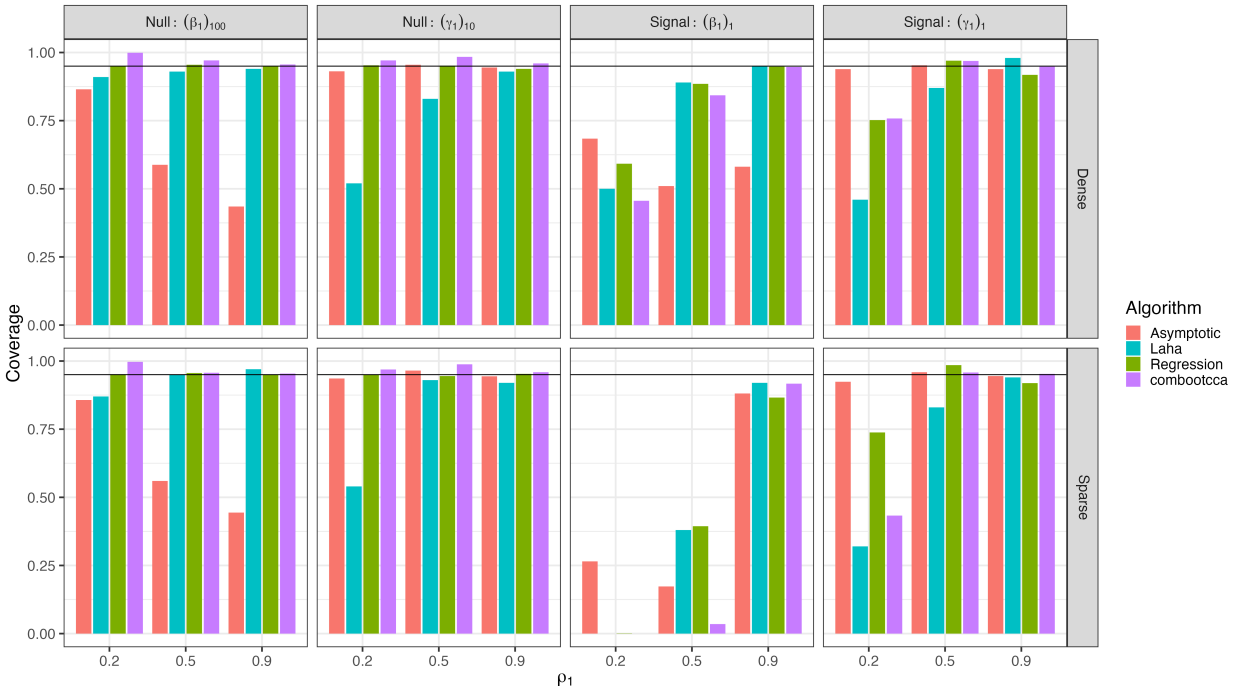

Figure 3.3: Coverage rates in simulation I for $p = 100, q = 10$. The horizontal line indicates nominal 95% coverage.

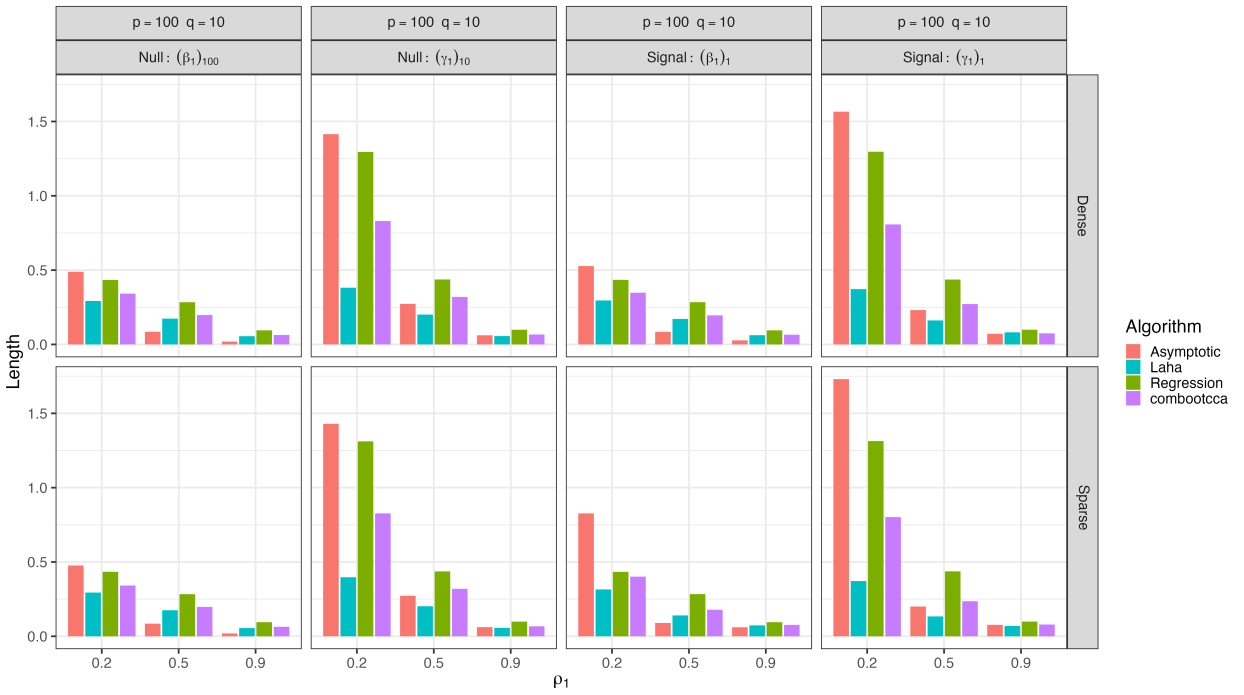

Figure 3.4: Lengths of confidence intervals in simulation I for $p = 100, q = 10$.

regime satisfies its assumptions. The pattern for the high-dimensional case where $p = 100$ and $q = 10$ is generally similar but with some differences. Since $p \neq q$, the theoretical justification for the asymptotic approach fails, and indeed the coverage for the high-dimensional β_1 is poor. This provides empirical evidence that the theoretical results developed in Anderson (1999) are indeed not applicable beyond the setting they were obtained for. Interestingly, the asymptotic method does appear to provide generally nominal coverage tests for the entries of the low-dimensional γ_1, The method of Laha et al. (2021) continues to struggle with coverage of null coordinates except when $\rho = 0.9$; interestingly this is worse for coverage of the low-dimensional γ_1. The sub-nominal coverage of combootcca for signals when ρ_1 is small is exacerbated, especially in the sparse regime and for β_1.

In general, it appears that achieving nominal coverage for a non-null coordinate is generally more challenging than for a null coordinate, and that this difficulty is greater when the canonical correlation ρ_1 is smaller. A heuristic explanation for this is as follows. While $\hat{\beta}_1$ is a consistent estimator for β_1, it is not necessarily unbiased. Because $\hat{\beta}_1$ must satisfy $\hat{\beta}_1^\intercal \hat{\Sigma}_x \hat{\beta}_1 = 1$, this is approximately a constraint on the norm of $\hat{\beta}_1$. Thus, if the direction is perfectly estimated, the leading coordinates of $\hat{\beta}_1$ will approach their true value, but if the direction is misestimated (as is more likely with small ρ_1 and larger dimensions), then there will be mass in other coordinates which will shrink the true non-zero coordinates towards

0, and our confidence intervals will reflect this. Moreover, we expect that this bias will be exacerbated in the sparse regime, when most coordinates are in fact zero.

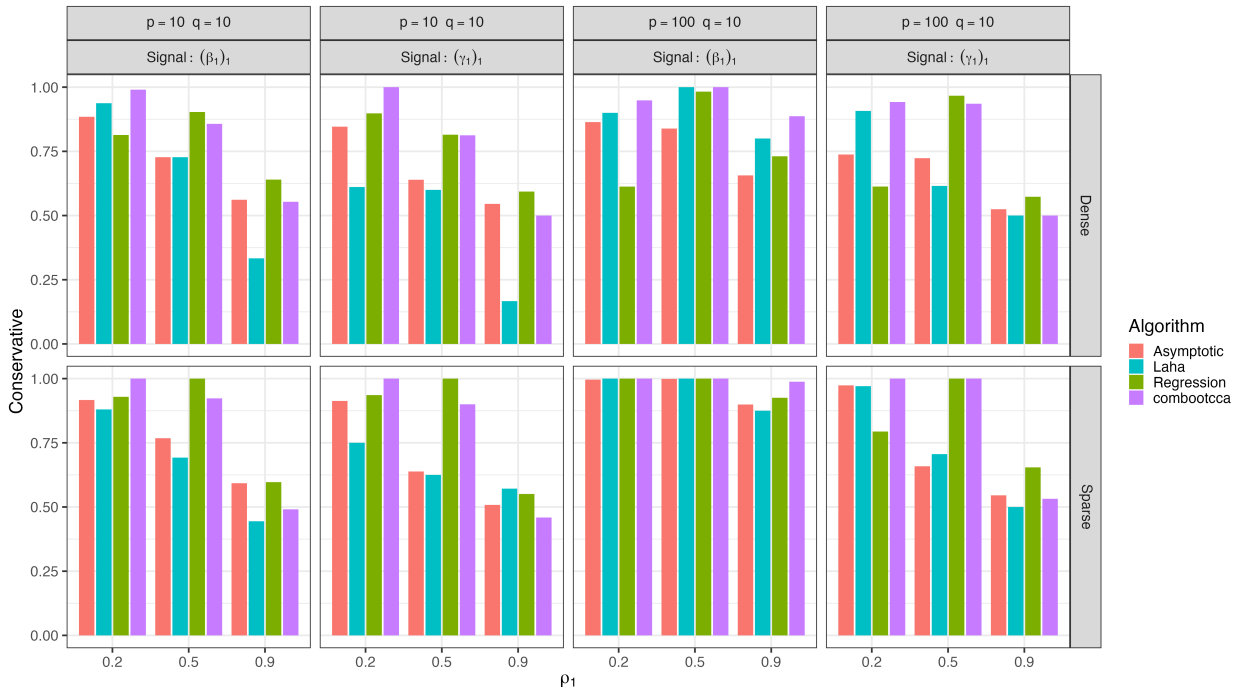

Figure 3.5: Bias in simulation I: the proportion of confidence intervals that failed to cover non-null signals that are "conservative" (the true value is greater in magnitude than any value in the confidence interval).

In order to investigate this, we examined confidence intervals that failed to cover a non-zero signal, checking whether the (absolute value of) the upper bound of the interval was less than (the absolute value of) the truth. If so, we considered that confidence interval "conservative." Figure 3.5 depicts the proportion of non-covering intervals that were conservative, and indeed shows that when ρ_1 is small, and especially in the sparse regime, the intervals from all methods are generally conservative, meaning that when they fail to cover the true non-zero value, it is likely because the estimate was shrunk towards zero.

Even confidence intervals that fail to achieve nominal coverage can lead to correct inference when the question is whether a given coordinate is equal to zero. Type I error for this hypothesis test is simply 1 minus coverage of at null coordinates (already depicted in Figures 3.1 and 3.3), and in Figure 3.6, we show the power of the test, i.e., the proportion of times confidence intervals for non-zero signals do not contain 0. Here we see that combootcca is generally the most powerful method among the three that achieve nominal control of Type I error.

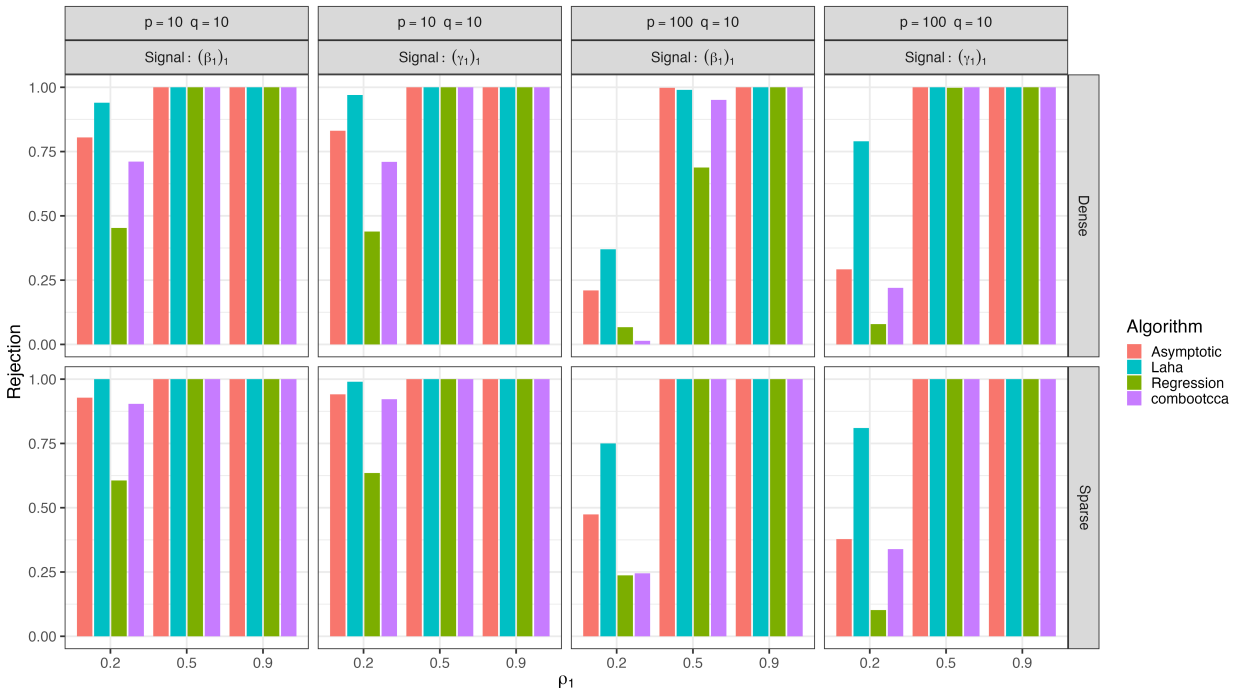

Figure 3.6: Power (correct rejection rates) in simulation I.

3.3.3 Simulation II: Synthetic Data with Two Canonical Correlations

Simulation II is similar to Simulation I in all respects except that we fix $\rho_1 = 0.9$ and introduce a second nonzero canonical correlation. We again consider both a dense and sparse regime, where the first canonical directions are the same as in Section 3.3.2, and the second canonical directions in the dense regime are proportional to

$$\check{\beta}_1 = \begin{bmatrix} \mathbf{0}_{p/2}^\intercal & \mathbf{1}_{p/2}^\intercal \end{bmatrix}^\intercal$$
$$\check{\gamma}_1 = \begin{bmatrix} \mathbf{0}_{q/2}^\intercal & \mathbf{1}_{q/2}^\intercal \end{bmatrix}^\intercal,$$

and in the sparse regime to

$$\check{\beta}_1 = \begin{bmatrix} \mathbf{0}_{2}^\intercal & \mathbf{1}_{p-2}^\intercal \end{bmatrix}^\intercal$$
$$\check{\gamma}_1 = \begin{bmatrix} \mathbf{0}_{2}^\intercal & \mathbf{1}_{q-2}^\intercal \end{bmatrix}^\intercal.$$

Canonical directions are subsequently normalized with respect to their associated covariances. Thanks to the structure of the covariance, these new directions satisfy $\beta_1^\intercal \Sigma_x \beta_2 = \gamma_1^\intercal \Sigma_y \gamma_2 = 0$. We fix $\rho_1 = 0.9$ and vary $\rho_2 \in \{0.8, 0.5, 0.2\}$. As in Simulation I, we examine

statistical properties associated with the confidence intervals for the first canonical directions at the last coordinates $(\beta_1)_p$ and $(\gamma_1)_q$ (always zero) and at the first coordinates $(\beta_1)_1$ and $(\gamma_1)_1$ (always non-zero). We also consider statistical properties for the second canonical directions at the first coordinates $(\beta_2)_1$ and $(\gamma_2)_1$ (always zero) and and the last coordinates $(\beta_2)_p$ and $(\gamma_2)_q$ (always non-zero). The method of Laha et al. (2021) is only applicable to the first canonical directions; it gives no results for the second canonical direction. As with Simulation I, in Appendix B.2 we repeated this experiment with identity covariances for Σ_x and Σ_y, and we found a generally similar pattern of results to those presented below.

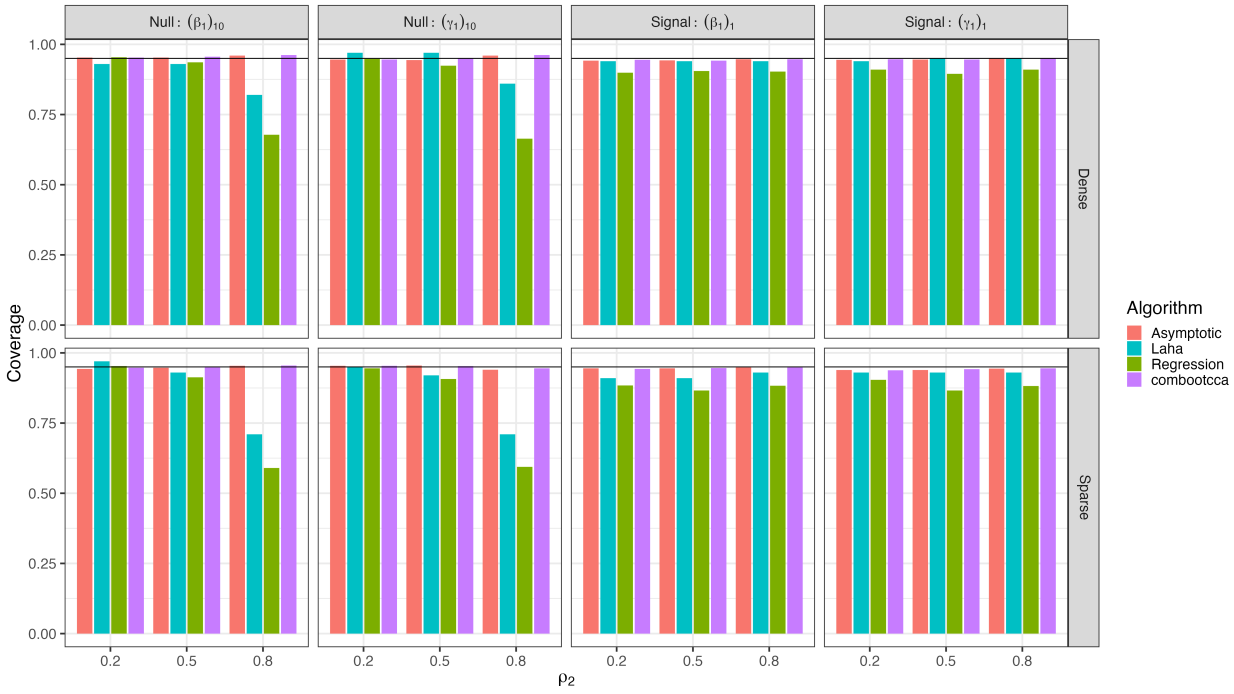

Figure 3.7: Coverage rates for first canonical directions in simulation II for $p = q = 10$. The horizontal line indicates nominal 95% coverage.

We first evaluate how coverage for the first canonical direction (with ρ_1 fixed at 0.9) varies when the strength of the second canonical correlation varies. Coverage is shown in Figure 3.7 for $p = q = 10$ and in Figure 3.8 for $p = 100, q = 10$, with corresponding lengths shown in Figures 3.9 and 3.10. Results are generally similar to Simulation I when ρ_1 was set to 0.9, although both the method of Laha et al. (2021) and regression-based method have poor coverage of null coordinates (i.e., inflated Type I error) when ρ_2 is large, which suggests that a narrower gap between the canonical correlations is especially detrimental for this method. For regression, we conjecture that in this setting, the initial estimate of the canonical directions is more likely to be inaccurate, and because the regression method effectively performs inference conditional on this value, it provides "valid" inference but for

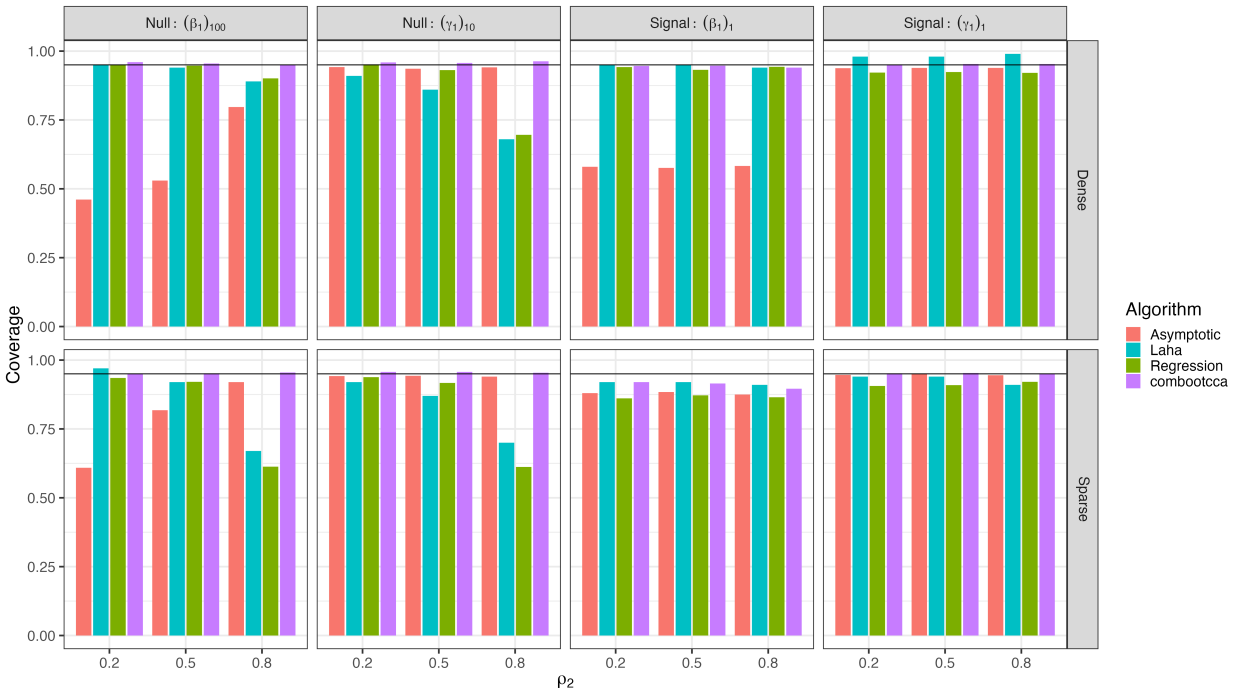

Figure 3.8: Coverage rates for first canonical directions in simulation II for $p = 100, q = 10$. The horizontal line indicates nominal 95% coverage.

the wrong quantity. Examining the lengths makes clear that the combootcca and asymptotic methods are appropriately sensitive to this small gap in the canonical correlations and make their confidence intervals wider than the other methods.

We repeat the investigation of bias in confidence intervals described in Section 3.3.2 and arrive at a similar conclusion: sparse signals yield more conservative (biased) intervals, which can harm coverage while retaining good power. These results are depicted in Figures 3.11 and 3.12. All methods have very good power for the first canonical direction, which is unsurprising as it is associated with a large canonical correlation ($\rho_1 = 0.9$).

Next, we examine coverage of the second canonical direction as ρ_2 varies. Coverage is depicted in Figure 3.13 for $p = q = 10$ and in Figure 3.14 for $p = 100, q = 10$, with associated lengths depicted in Figures 3.15 and 3.16. Recall that the method of Laha et al. (2021) is only applicable for the *first* canonical directions, so it offers no results here. The only method with nominal coverage of null coordinates (and thus valid control of Type I error) is combootcca; the asymptotic approach works for $p = q$ but fails when $p \neq q$. When the signal is dense, combootcca has nominal coverage of signal coordinates when $p = q$, and when $p \neq q$ it approaches nominal coverage for signal coordinates except when $\rho_2 = 0.2$. When the signal is sparse and $p \neq q$, combootcca again suffers from poor coverage of signal coordinates, and we see in Figure 3.17 that this again reflects overly conservative (in magnitude) confidence

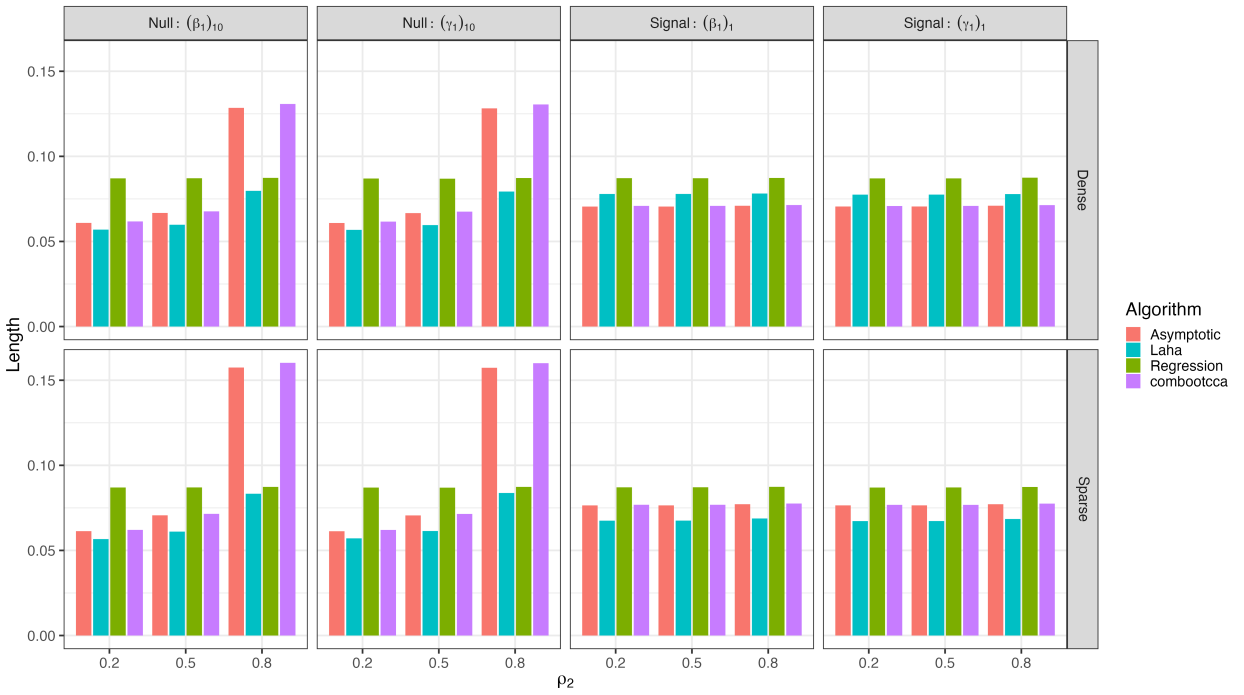

Figure 3.9: Lengths of confidence intervals for first canonical directions in simulation II for $p = q = 10$.

intervals, but that combootcca still has non-trivial power as depicted in Figure 3.18.

3.3.4 Simulation III: Data-Based Simulation

While Simulation I and II were useful for studying the behavior of various methods in simple settings, Simulation III offers a more realistic setting. Rather than specifying Σ_x, Σ_y along with ρ, B, and Γ a priori, we instead specify them based on our neuroimaging data set as described in Section 3.4. Specifically, we take the estimated covariance $\hat{\Sigma}$ for our processed and cleaned data, i.e., the empirical covariances of X_2 and Y_2, as well as their cross-covariance. Using the empirical covariance as the ground truth, we solve the population version of CCA as described in Section 3.1.1, and we arrive at corresponding values for ρ, B, and Γ. However, the directions are fully dense with variable magnitudes, and so modifications are necessary in order to carefully study the empirical statistical properties of confidence intervals. Specifically, we modify the last coordinate of one of $\beta_1, \beta_2, \gamma_1$, or γ_2. We set it to take one of the following three values: (i) 0 (in which case it corresponds to a true null), (ii) the mean of the absolute values of the other entries of the direction, (iii) the max of the absolute values of the other entries of the direction. In both cases (ii) and (iii), we preserve the sign of the coordinate. After this modification, we need to reconstruct the generative

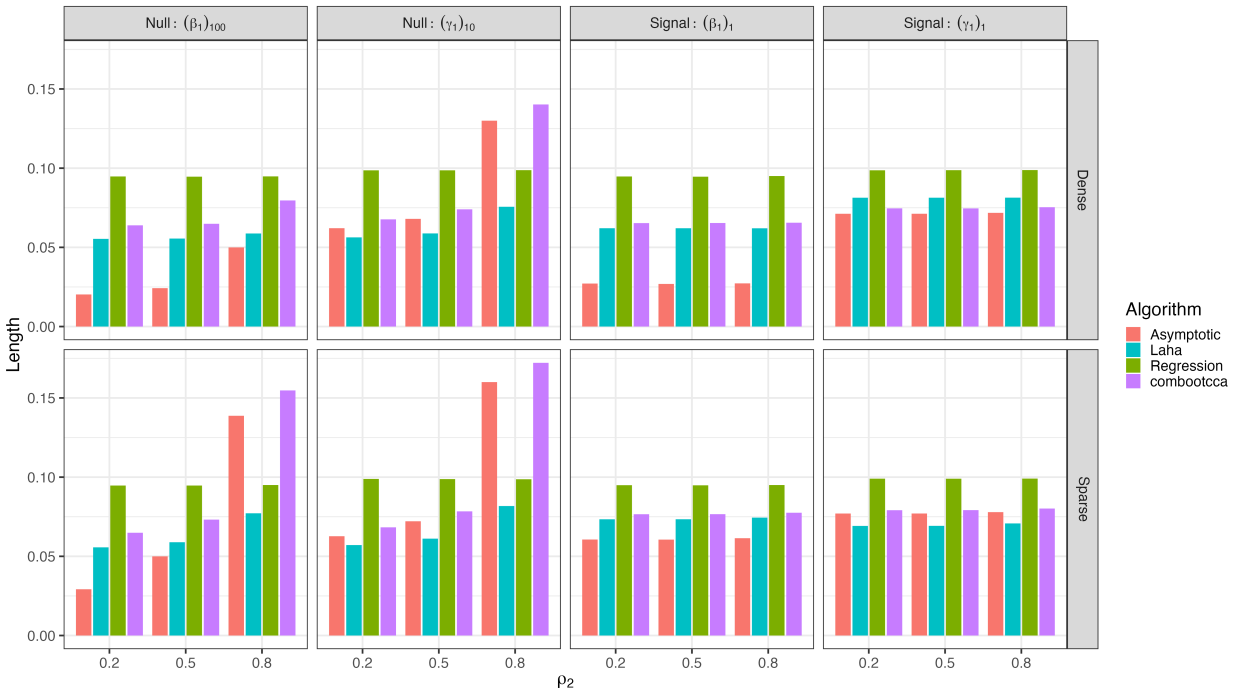

Figure 3.10: Lengths of confidence intervals for first canonical directions in simulation II for $p = 100, q = 10$.

covariance Σ, since it will no longer correspond to this modified solution. We invert the CCA model as described in Section 3.1.2 to recover a Σ that fits the desired CCA solution, and then we generate data from it. Our choice of Σ requires that we leave (p, q) fixed at $(250, 11)$ (to correspond to the application), and we similarly set $N = 2969$ to correspond to the application.

We present coverage for all methods in Figure 3.19 with associated lengths in Figure 3.20. When considering only the first direction, only combootcca attains nominal coverage in all settings, while the asymptotic method does so for the low-dimensional γ_1. The method of Laha et al. (2021) has coverage close to (but short of) nominal for β_1, but poor coverage for γ_1, and offers no results for the second directions β_2 and γ_2. When examining results for the second canonical directions β_2 and γ_2, combootcca has nominal (or near nominal) coverage for null and moderate signal, but falls short of nominal coverage when the signal is set to its maximum. This is consistent with our earlier simulation studies where confidence intervals for coordinates with large values are conservatively biased, which can be confirmed in Figure 3.21. In Figure 3.22, we show the power for each method. Although not the most powerful method, combootcca is unique in that it had nominal Type I error control, but still enjoys non-trivial power, especially in the presence of coordinates with large values.

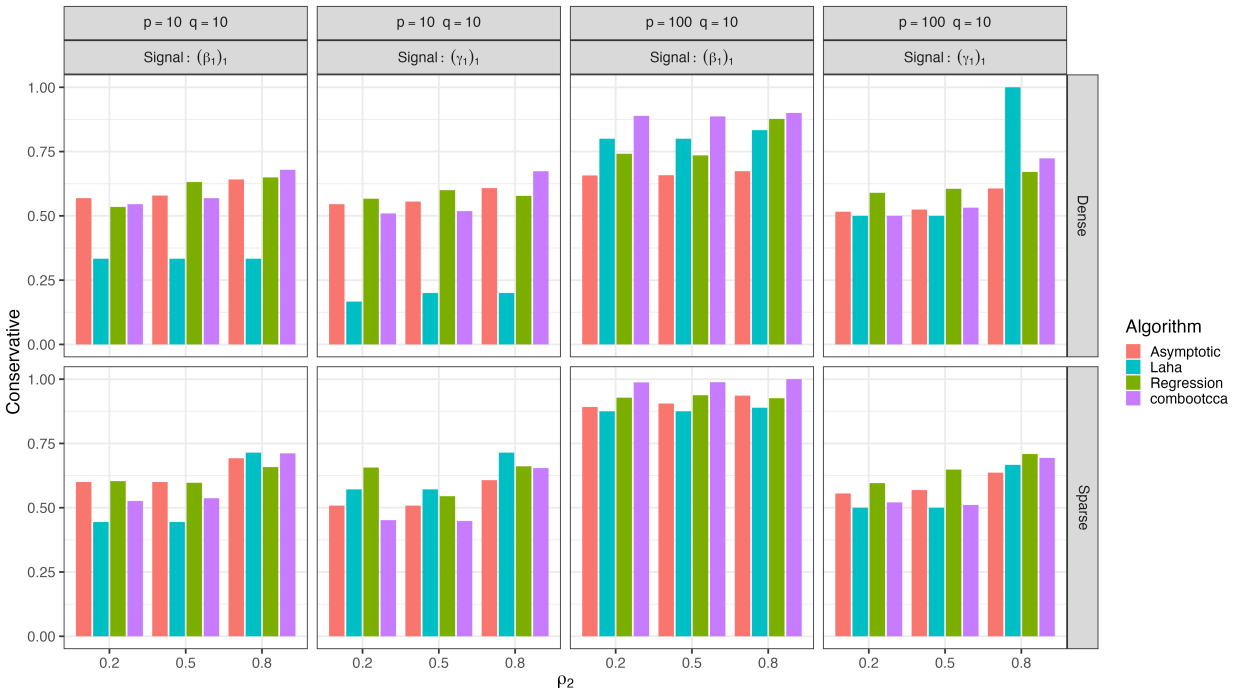

Figure 3.11: Bias in simulation II for first canonical directions: the proportion of confidence intervals that failed to cover non-null signals that are "conservative" (the true value is greater in magnitude than any value in the confidence interval).

3.3.5 Comparison of Bootstrap Strategies

In Section 3.2, we outlined different alignment strategies for the bootstrap as well as two approaches to constructing confidence intervals. Here, we provide empirical evidence to justify our choices in the combootcca method, namely, performing alignment with a weighted Hungarian algorithm and using percentile bootstrap.

Figure 3.23 shows coverage rates in the setting of Simulation I (see Section 3.3.2) for all four alignment strategies considered as well as the two different types of confidence intervals. We can easily see that the bootstrap that uses the normal approximation (in rows 1 and 3) generally fails to achieve nominal coverage when ρ is small, regardless of alignment strategy. This is a particularly noteworthy observation, given the popularity of this type of bootstrap in the applied literature. Coverage is generally better for the percentile-based bootstrap, although as we have seen in the preceding sections, its coverage is poor when the signal is sparse and ρ_1 is small, however this generally reflects a conservative bias in the intervals that still leaves non-trivial power. Considering alignment strategies, the Procrustes-based alignment fails to achieve nominal coverage in many settings for both the normal-approximation and percentile bootstrap; this is consistent with our characterization of it as an overly ag-

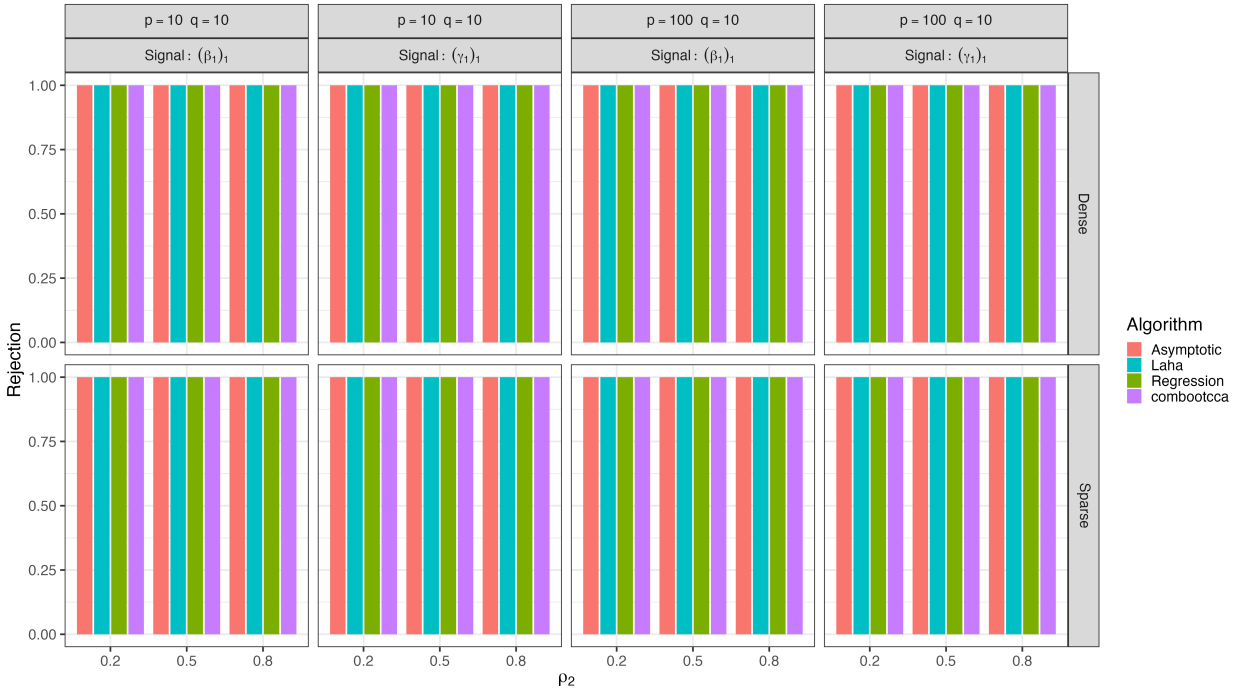

Figure 3.12: Power (correct rejection rates) for first canonical directions in simulation II.

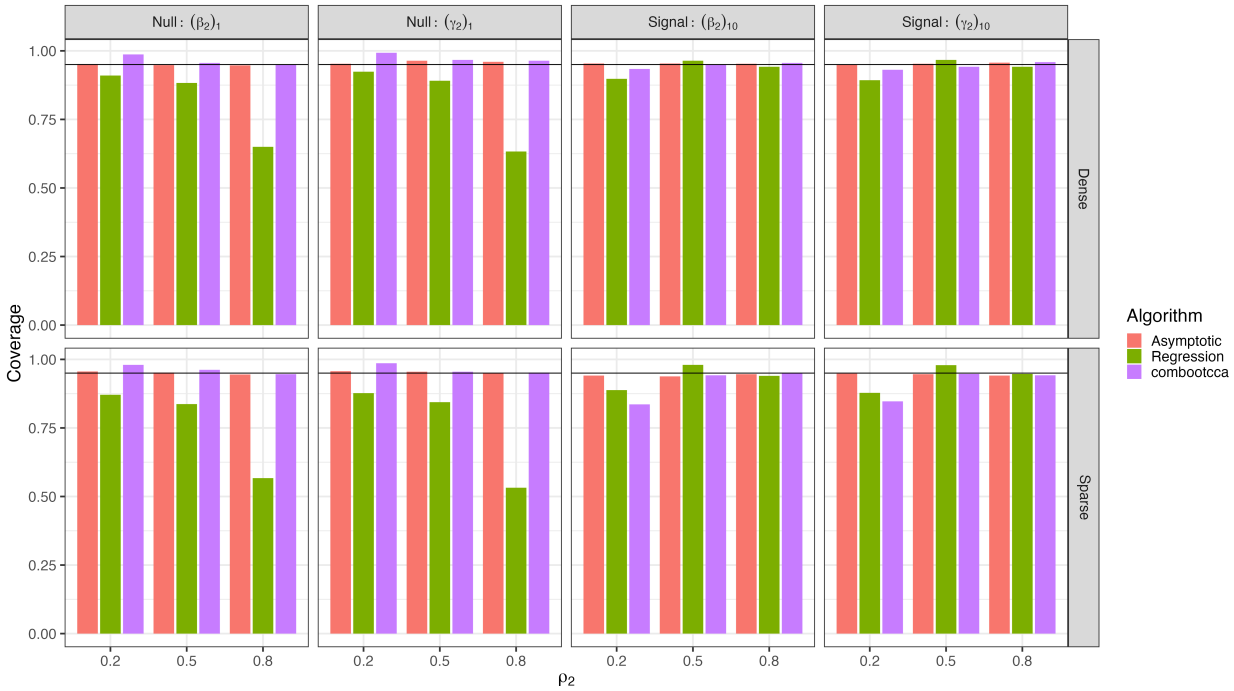

Figure 3.13: Coverage rates for second canonical directions in simulation II for $p = q = 10$. The horizontal line indicates nominal 95% coverage.

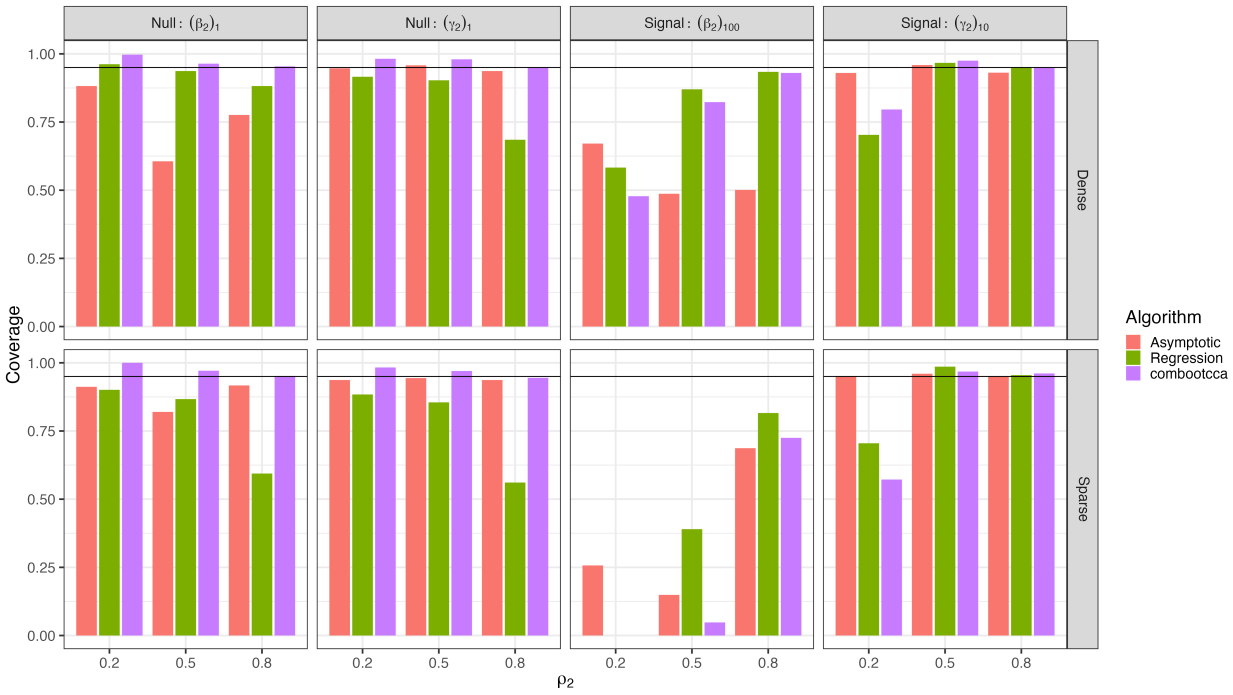

Figure 3.14: Coverage rates for second canonical directions in simulation II for $p = 100, q = 10$. The horizontal line indicates nominal 95% coverage.

gressive alignment strategy. In order to choose between the remaining alignment strategies, we consider power as presented in Figure 3.24. We have already eliminated the normal-approximation bootstrap from consideration due to its poor coverage, and when using the percentile-based bootstrap we see that the weighted Hungarian alignment has the best power.

In order to verify that this recommendation is not specific to our toy simulation study, we also study the choice of alignment strategy and type of confidence interval in the more realistic setting of Simulation III (see Section 3.3.4). In Figure 3.25, we can again see that the normal approximation is problematic for coverage, and that Procrustes-based alignment can yield very poor coverage, especially for the smaller dimensional γ_1 and γ_2. Turning to power as depicted in Figure 3.26, the weighted Hungarian alignment coupled with percentile-based bootstraps remains the winning combination.

3.4 Application to ABCD Dataset

We apply our methods to data taken from the ABCD study (Casey et al., 2018) processed by the lab of our collaborator, Dr. Chandra Sripada. We work with an initial corpus of 5937 individuals who have complete data on (i) usable resting state fMRI data, (ii) behavioral performance scores on 11 tasks, and (iii) nuisance covariates. This same subset of the ABCD

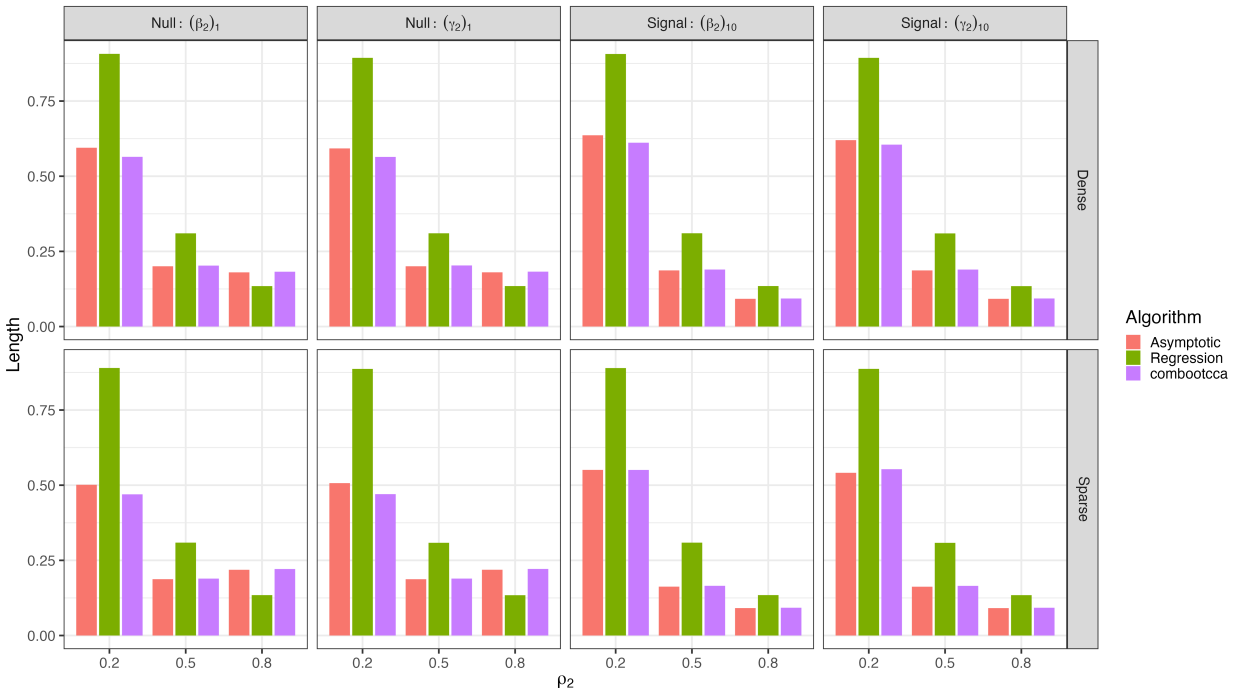

Figure 3.15: Lengths of confidence intervals for second canonical directions in simulation II for $p = q = 10$.

data was analyzed in Sripada et al. (2021), which describes the data and processing in more detail. Interestingly, our analysis which aims to tie behavioral tests to biological measurements, is very much in the spirit of Hotelling's seminal which introduced CCA (Hotelling, 1936), which suggested on its first page that with CCA, "the scores on a number of mental tests may be compared with physical measurements on the same persons."

The initial data matrix $\tilde{X} \in \mathbb{R}^{5937 \times \binom{418}{2}}$ holds functional connectivity data. Each row corresponds to a vectorized correlation matrix taken pairwise over time between 418 parcels in the brain according to the parcellation of Gordon et al. (2016). The initial data matrix $\tilde{Y} \in \mathbb{R}^{5937 \times 11}$ holds behavior scores for the same participants on a corpus of 11 tasks taken from the neurocognition assessment from the ABCD study and described in more detail in Luciana et al. (2018). In brief, seven of the tasks are taken from the NIH Toolbox (Hodes et al., 2013): (i) Picture Vocabulary (Vocabulary), (ii) Oral Reading Recognition (Reading), (iii) Pattern Comparison Processing Speed (Processing Speed), (iv) List Sorting Working Memory (Working Memory), (v) Picture Sequence Memory (Episodic Memory). (vi) Flanker Inhibitory Control & Attention (Flanker), and (vii) Dimensional Change Card Sort (Card Sort). From the Rey Auditory Verbal Learning Test, we use performance in both the (viii) Short Delay (Memory: Short Delay) and (ix) Long Delay (Memory: Long Delay) conditions. Finally, we also used performance in the following tasks: (x) Matrix Reasoning,

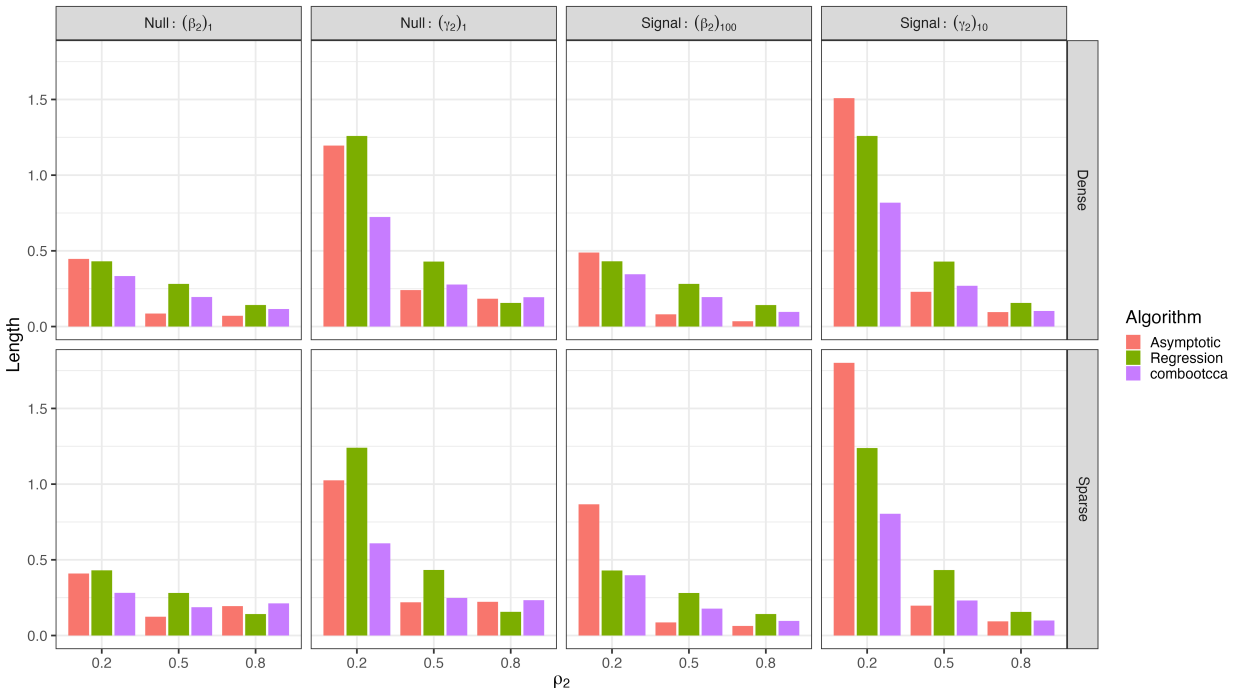

Figure 3.16: Lengths of confidence intervals for second canonical directions in simulation II for $p = 100, q = 10$.

and (xi) Little Man Task (Spatial Rotation).

We corrected for six nuisance covariates for each participant, namely age, age^2, sex, meanFD, meanFD^2, and race/ethnicity. MeanFD is a summary measure of how much the participant moved their head during the resting state scanning session. After adding an intercept column of ones and dummy-coding categorical nuisance covariates, we obtained the nuisance matrix $W \in \mathbb{R}^{5937\times 10}$. Before performing CCA, we first remove variation associated with the nuisance covariates and then reduce the dimension of the neuroimaging data. We randomly partitioned our data into two roughly equally-sized sets,

$$\tilde{X} = \begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \end{bmatrix}, \quad \tilde{Y} = \begin{bmatrix} \tilde{Y}_1 \\ \tilde{Y}_2 \end{bmatrix}, \quad W = \begin{bmatrix} W_1 \\ W_2 \end{bmatrix}.$$

Using the training data $(\tilde{X}_1, \tilde{Y}_1, \tilde{W}_1)$, we regress the variables of interest on the nuisance covariates, obtaining the coefficients

$$\hat{A}_X = (W_1^\intercal W_1)^{-1} W_1^\intercal X_1, \quad \hat{A}_Y = (W_1^\intercal W_1)^{-1} W_1^\intercal Y_1.$$

We then remove the contributions of nuisance covariates using the coefficients learned in the

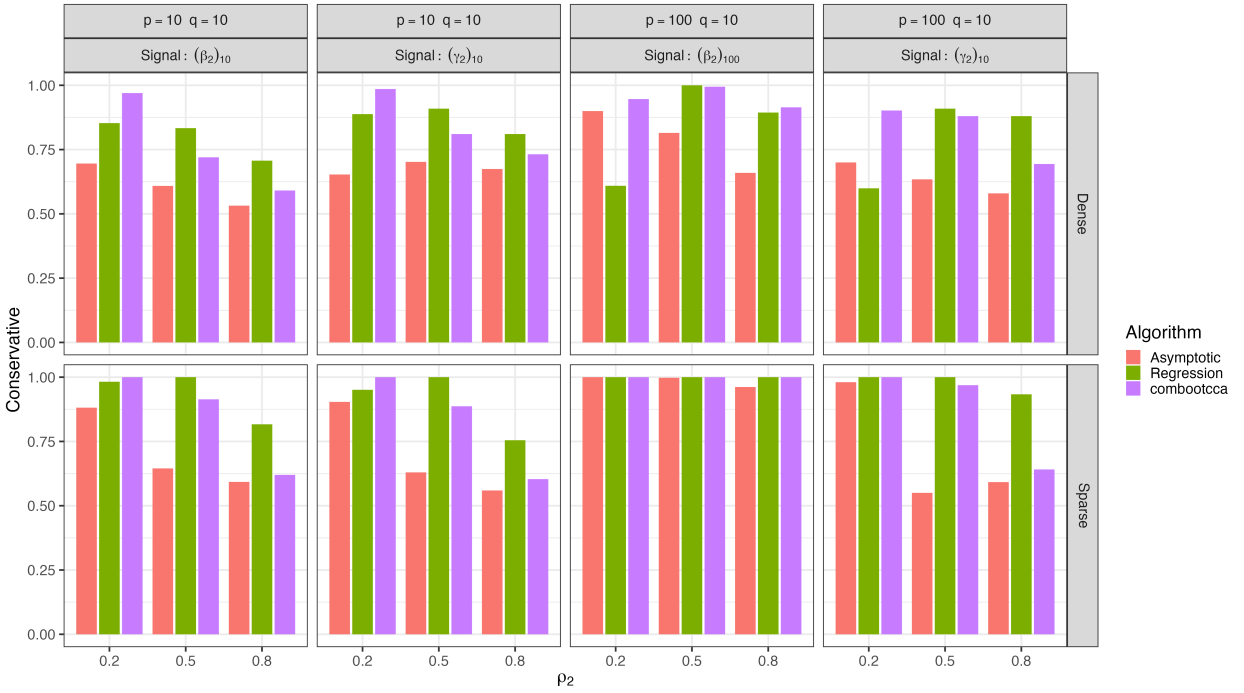

Figure 3.17: Bias in simulation II for second canonical directions: the proportion of confidence intervals that failed to cover non-null signals that are "conservative" (the true value is greater in magnitude than any value in the confidence interval).

training data by setting

$$\check{X}_1 = X_1 - W_1\hat{A}_X, \qquad \check{X}_2 = X_2 - W_2\hat{A}_X,$$
$$\check{Y}_1 = Y_1 - W_1\hat{A}_Y, \qquad \check{Y}_2 = Y_2 - W_2\hat{A}_Y.$$

Next, we reduced the dimension of the neuroimaging data matrix $\check{X}$ using PCA. This is a common processing step upstream of CCA in the neuroimaging literature (see e.g., Helmer et al. (2020); Fernandez-Cabello et al. (2022)). We learned the PCA transformation on the training data: because $\check{X}_1$ was already column-centered (since an intercept was included in the nuisance matrix W_1), we performed PCA via SVD and decomposed $\check{X}_1 = USV^\intercal$. Based on input from our collaborators, we retained the leading 250 principal components and truncated V accordingly. Then, we projected the held-out, but nuisance-corrected neuroimaging data onto this basis with $X_2 = \check{X}_2 V_{[:,1:250]}$. The phenotypic data is already low-dimensional, so we simply set $Y_2 = \check{Y}_2$. As a final preprocessing step, we standardized the columns of X_2 and Y_2 to have mean 0 and variance 1.

We then perform CCA on X_2 and Y_2. We use R's `cancor` function (which uses QR decomposition internally), but we rescale the canonical directions as discussed in Section 3.1.1. In

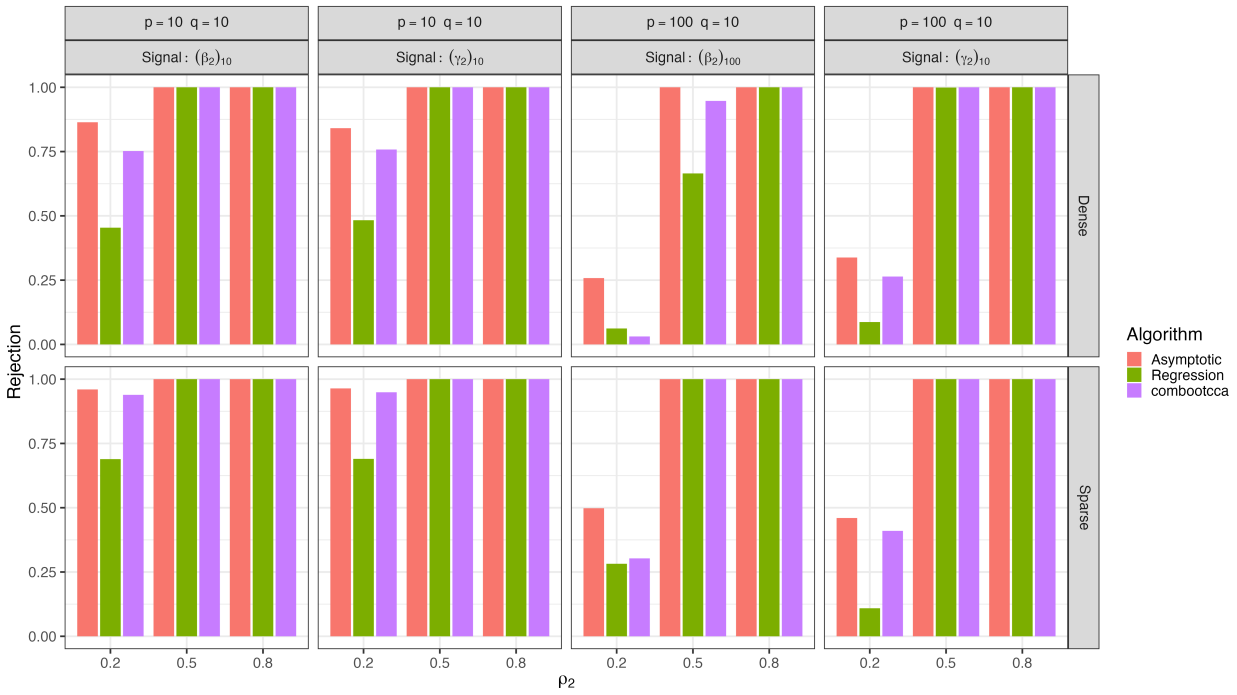

Figure 3.18: Power (correct rejection rates) for second canonical directions in simulation II.

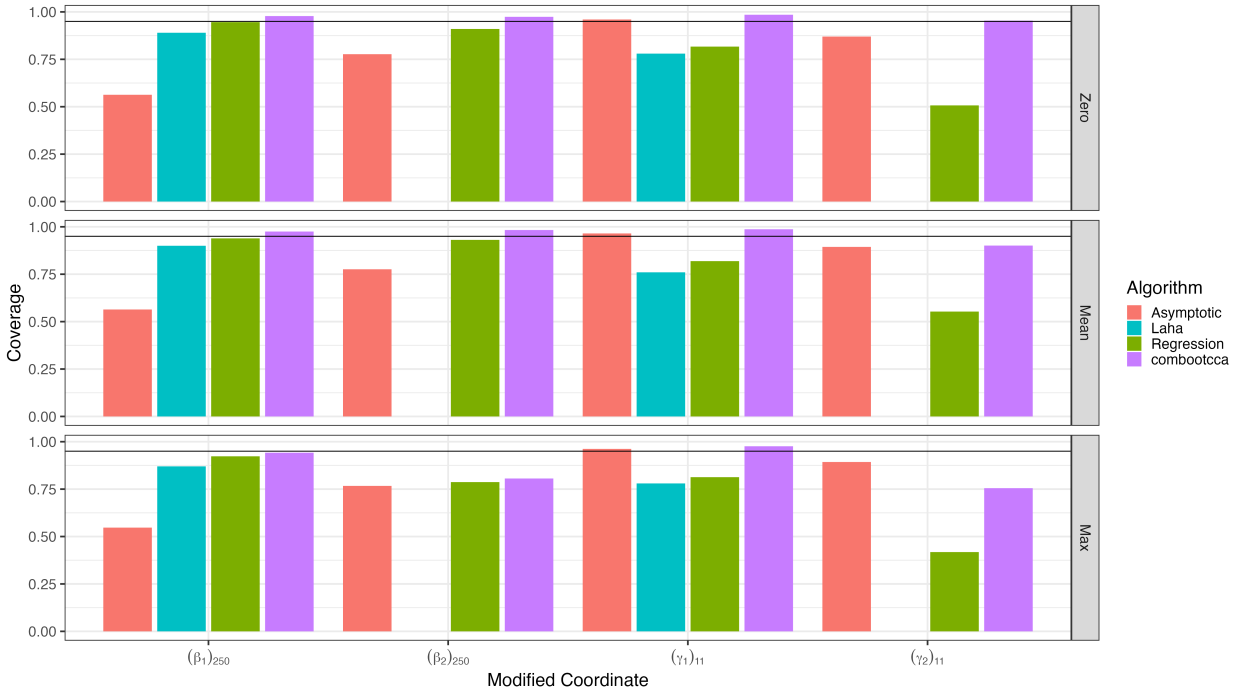

Figure 3.19: Coverage rates in simulation III.

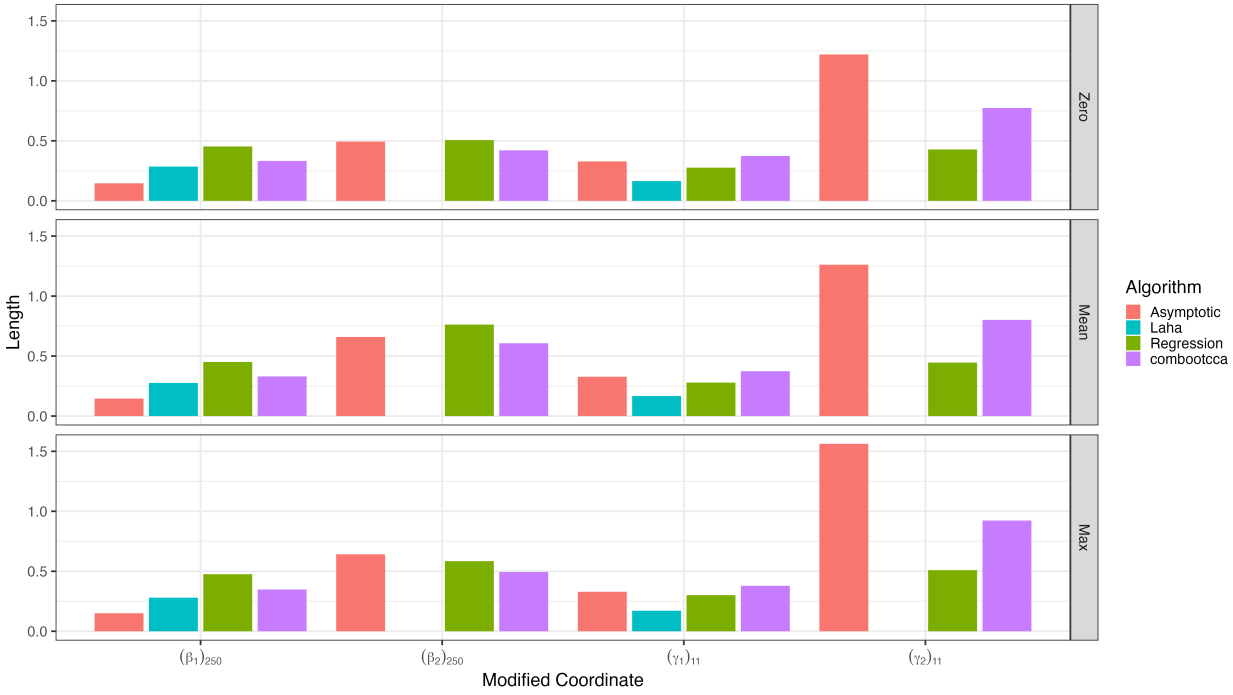

Figure 3.20: Lengths of confidence intervals in simulation III.

Figure 3.27 we plot the canonical correlations. Although inference for ρ is not the main focus of the present manuscript, for completeness we report the results of inference on ρ obtained from the `yacca` package's `F.test.cca` function (Butts, 2022). As depicted in Figure 3.27, we find that three canonical correlations are significantly nonzero at $\alpha = 0.05$, and we will therefore restrict our attention to inference on first three canonical directions. Notably, the first canonical correlation is well-separated from the rest, but subsequent canonical correlations are not.

We perform inference on the canonical directions using the combootcca method. Figure 3.28 shows point estimates and associated confidence intervals for the first three directions of Γ. Here, interpretability is aided by the data standardization, as all of the variables are on the same scale. The intervals for the first direction are markedly shorter than for the subsequent directions; this is consistent with greater uncertainty due to the poor separation of canonical correlations beyond the first. While there is a fundamental sign ambiguity in CCA, the relative signs of the directions can be meaningfully interpreted. For example, in the first direction, the confidence intervals for Vocabulary, Reading, Working Memory, and Matrix Reasoning do not include 0 and all have the same sign. The fact that these four tasks appear in a single canonical direction is noteworthy: the Vocabulary and Reading tests are classic hallmarks of "crystallized intelligence," whereas Working Memory and Matrix Reasoning are

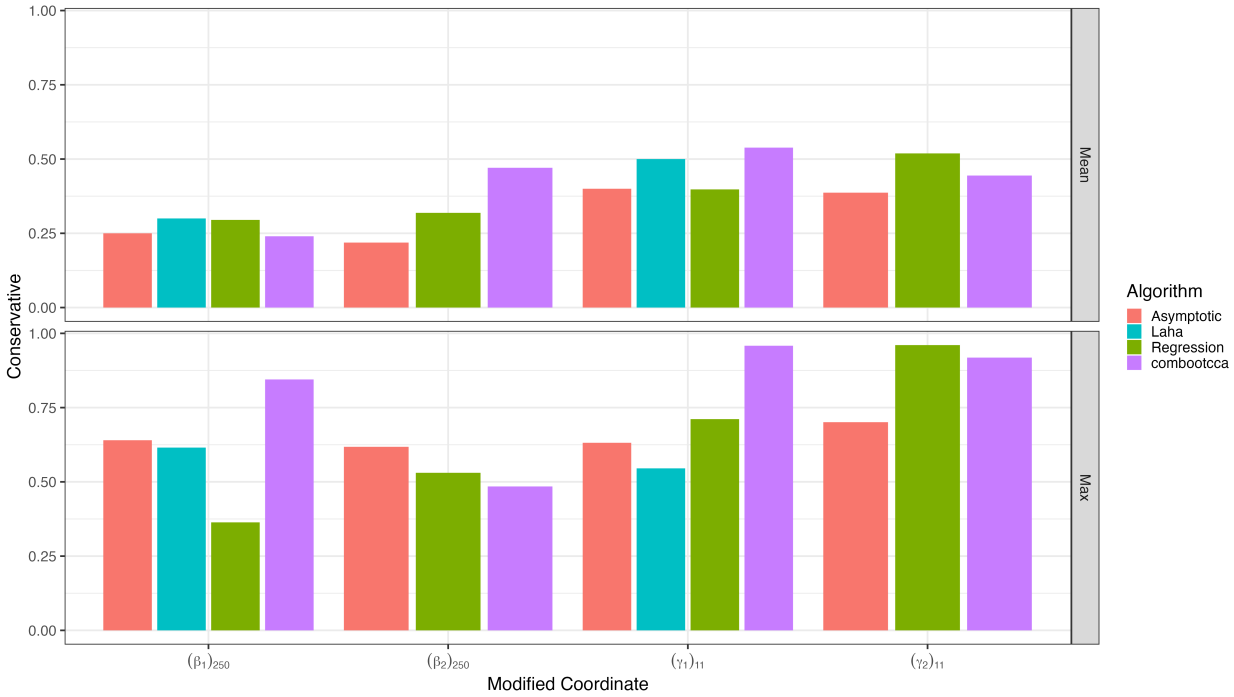

Figure 3.21: Bias in simulation III. The proportion of confidence intervals that failed to cover non-null signals that are "conservative" (the true value is greater in magnitude than any value in the confidence interval).

considered strong indicators of "fluid intelligence," and there is disagreement as to whether these are really two distinct capacities or if they reflect a single general ability. Indeed, Panigrahi et al. (2023b) deployed a multi-task learning approach with the same data but restricted their behavioral measures to these four tasks in an investigation of the shared versus distinct neural bases for these two types of cognitive abilities. Our results can also be compared with those obtained from an independent CCA analysis of the ABCD data in Goyal et al. (2022), where their post-hoc analysis of their second canonical variate implicates many of the same tasks that we identified.

Figure 3.29 shows confidence intervals for the first three directions of B. Notably, the first direction has a number of coordinates that are significantly nonzero, whereas there is just one for the second direction and none for the third direction. This parallels our findings in Simulation III (Section 3.3.4), wherein we saw that combootcca had non-trivial power for the leading direction of B but little power for the second direction, perhaps due to the poor separation of the second canonical correlation from subsequent canonical correlations. Recall that the coordinates of B correspond to PCA scores from the higher-dimensional brain imaging data: while they are not directly interpretable, it is possible to invert the data reduction step. If we wish to recover the canonical directions in the original feature

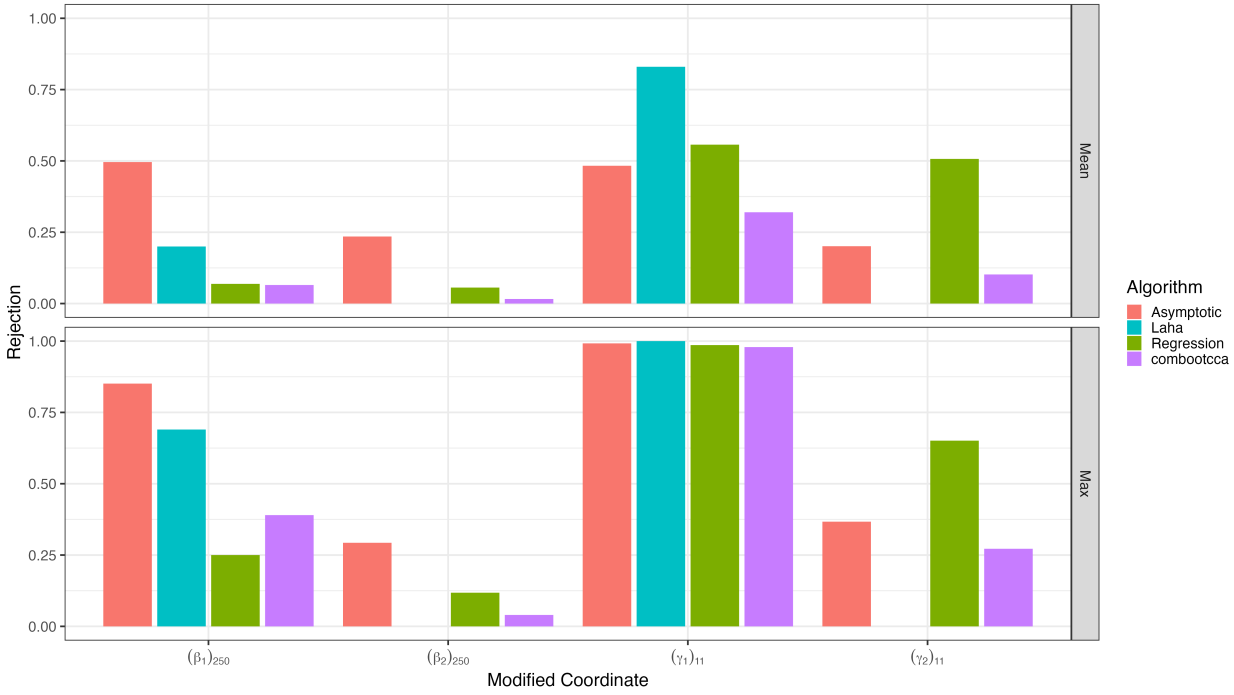

Figure 3.22: Power (correct rejection rates) in simulation III.

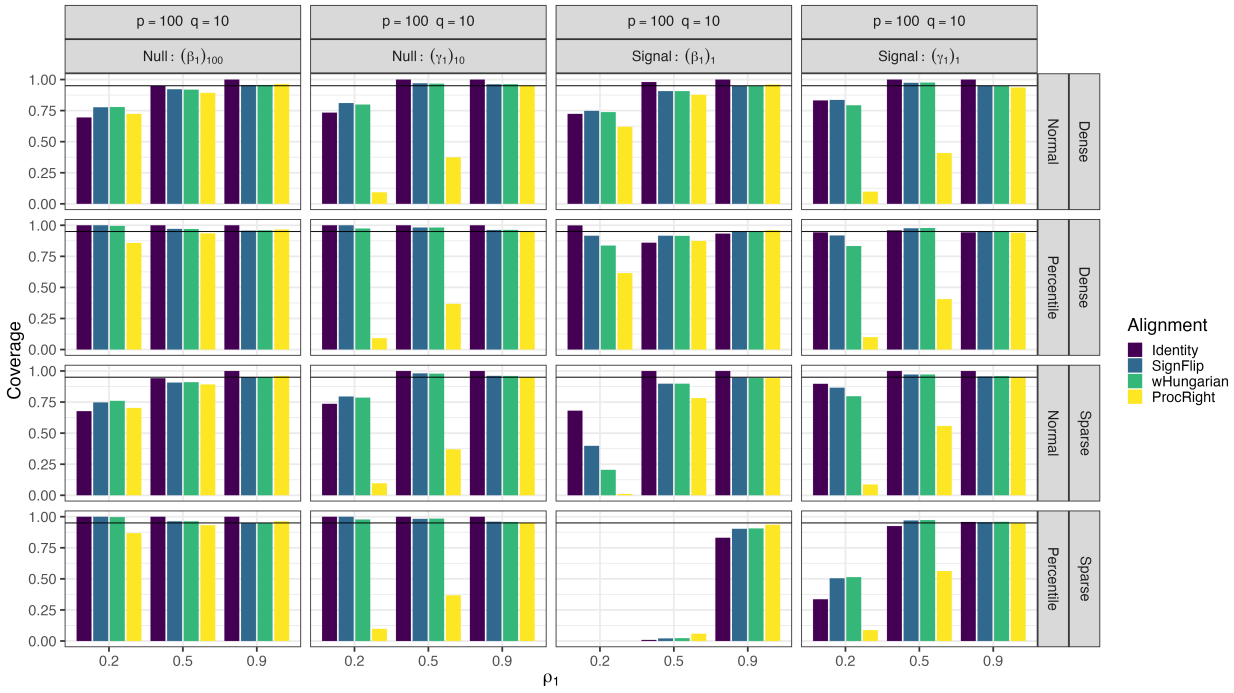

Figure 3.23: Comparison of coverage rates for different types of bootstraps and alignment strategies in the setting of simulation I.

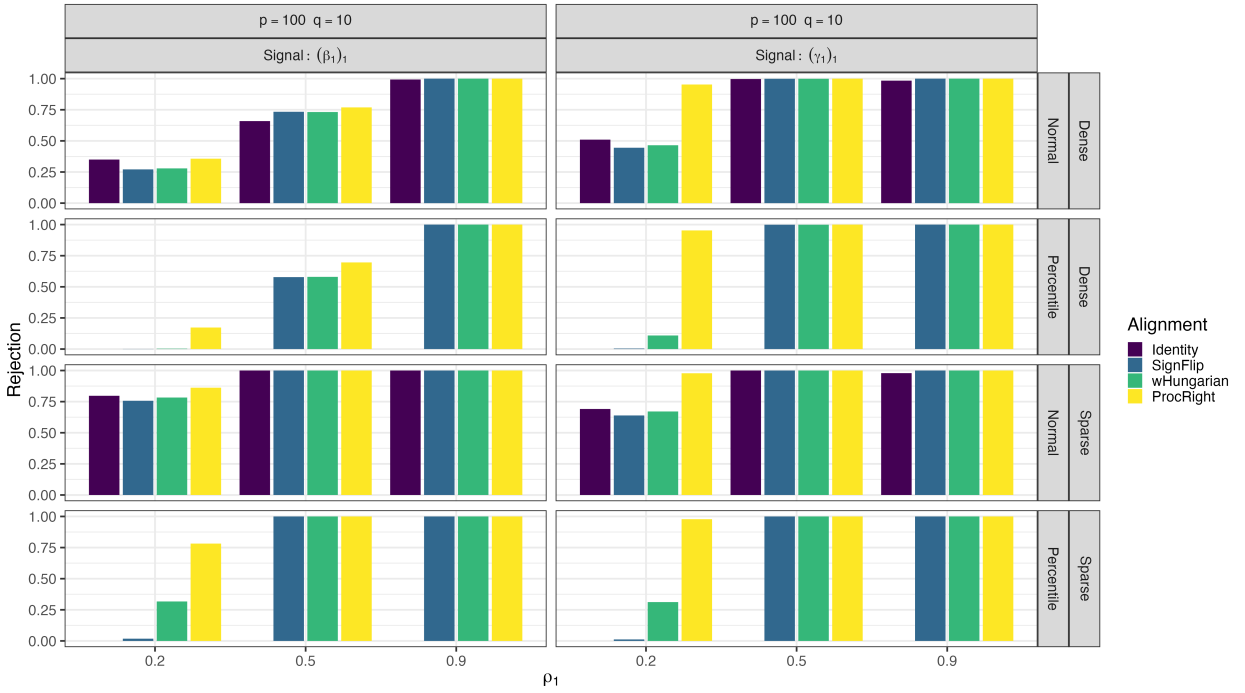

Figure 3.24: Comparison of power (correct rejection rates) for different types of bootstraps and alignment strategies in the setting of simulation I.

space, we can compute $\check{B} = V_{[1:250]} D^{-1/2} \hat{B}$, where D is a diagonal matrix with the empirical variances of X_2 (the adjustment by D is necessary to invert the standardization we applied as a preprocessing step). Alternately, we can set to 0 any entries of $\hat{B}$ whose corresponding confidence intervals included 0, and construct an analogous quantity with this thresholded version of $\hat{B}$. These reconstructed directions can be reshaped into matrix form and organized according to the assignment of parcels to putative brain systems in the parcellation of Gordon et al. (2016). Figures 3.30, 3.31, and 3.32 depict $\check{\beta}_1$, $\check{\beta}_2$, and $\check{\beta}_3$, respectively, whereas Figures 3.33 and 3.34 show the analogous quantities obtained after thresholding $\hat{\beta}_1$ and $\hat{\beta}_2$ (there are no significantly non-zero coordinates in $\hat{\beta}_3$, so we do not depict it). Of particular interest given the tasks associated with the first direction, we note that when examining the brain features associated with $\hat{\beta}_1$ in Figures 3.30 and 3.33, there qualitatively appears to be far more mass in edges linking the Default, FrontoParietal, Dorsal Attention, Salience, Cingulo-Opercular, Cingulo-Parietal, and Ventral Attention systems. These systems were situated by Margulies et al. (2016) near the beginning of a gradient that transitions from transmodal to sensory cortices. This pattern, coupled with the behavioral variables implicated in γ_1, suggest that our leading pair of canonical directions may indeed reflect shared structure in both brain and behavior that undergird important and general cognitive ability.

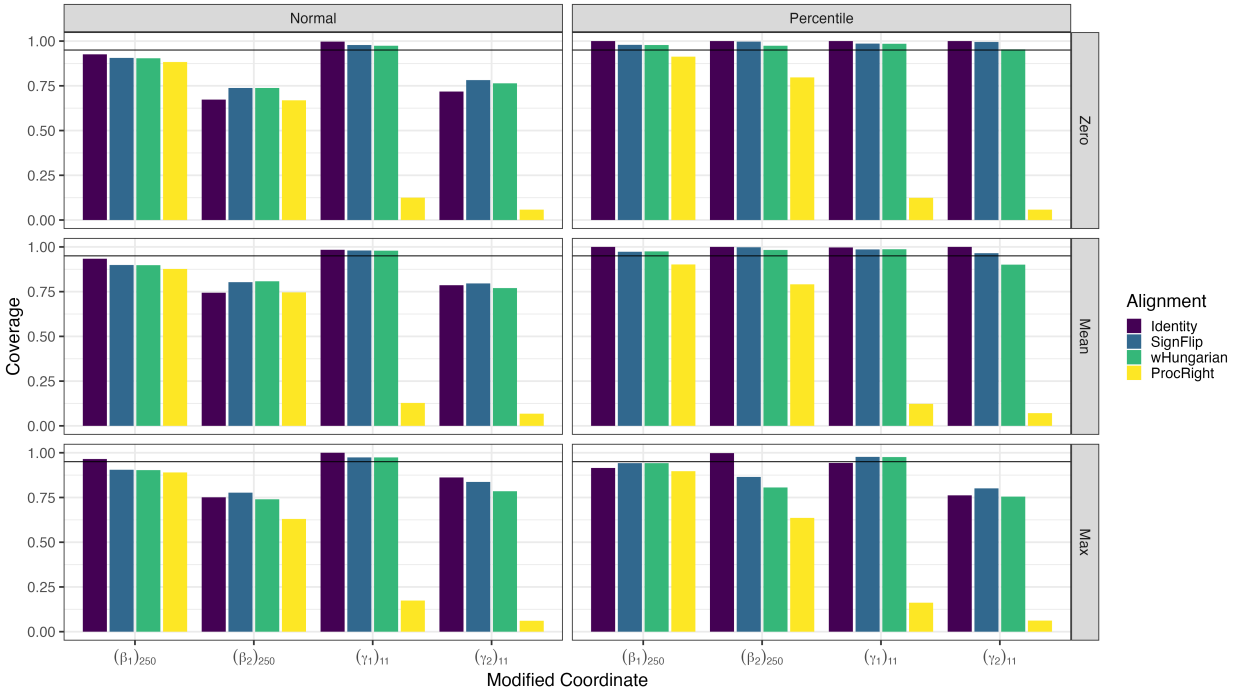

Figure 3.25: Comparison of coverage rates for different types of bootstraps and alignment strategies in the setting of simulation III.

3.5 Discussion

In this work, we have considered the problem of inference for the directions obtained from CCA. While much statistical work has focused on inference for the canonical correlations, the canonical directions in the classical setting have received less treatment, except for Anderson (1999)'s treatment of the $p = q$ fixed, $N \to \infty$ case. In the absence of clear guidance from the statistical literature, practitioners often either ignore this part of the analysis, or use a variety of heuristic resampling methods. Recent methodological work from applied groups (e.g., Helmer et al. (2020); McIntosh (2021)) has focused on characterizing the "stability" or "reproducibility" of the canonical directions. While this is informative, it is not the same as performing statistical inference on the canonical directions, and often involves arriving at global conclusions (e.g., "this vector is unstable") based on the angles between canonical directions under resampling as opposed to local inference (e.g., "this coordinate is significantly nonzero"). While this line of work has generally approached the issue numerically, very recent work in the statistical literature (Bykhovskaya and Gorin, 2023) has provided theoretical results in a similar vein in the setting where p, q, and N all grow together. That framework, too, still has a global rather than local focus: in the example application, the authors obtain an estimate of the angle between estimated canonical variates and true canonical variates,

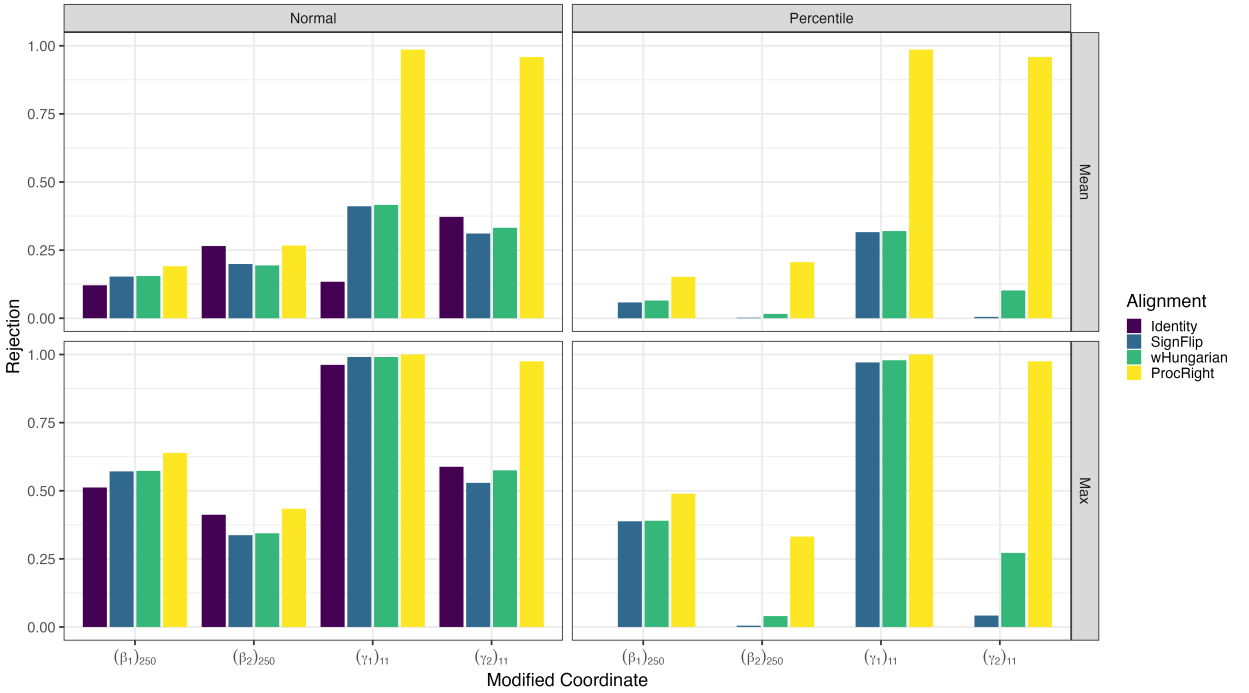

Figure 3.26: Comparison of power (correct rejection rates) for different types of bootstraps and alignment strategies in the setting of simulation III.

remark that it is low, and then present point estimates for the canonical directions. While these are useful studies, ultimately our goal is to go one step further and use the characterization of stability in order to deliver inference. While this is indeed what is attempted in practice, our carefully designed simulation studies, which range from tightly controlled to highly realistic settings, allow us to evaluate how well various methods perform with regard to statistical inference with known and non-trivial ground truth. In addition, in contrast to most applications, we carefully consider a number of design choices for bootstrap-based approaches and their relative consequences, merits, and pitfalls. This is especially useful since various applied papers often arrive at different procedures which may have consequences for coverage as well as both Type I and Type II error control. Moreover, our use of a data-based simulation study, wherein we assess the statistical properties of our procedures on synthetic data designed to closely mimic our eventual application with realistic levels of signal, is a useful example of how a statistical method can be evaluated prior to its application on a given data set.

Based on our simulation studies, we specifically recommend the use of percentile-based bootstraps with the weighted Hungarian algorithm alignment as performed in combootcca, which overall delivers the best combination of coverage, error control, and power. The recent approach of Laha et al. (2021), originally developed for sparse CCA, seems to be a

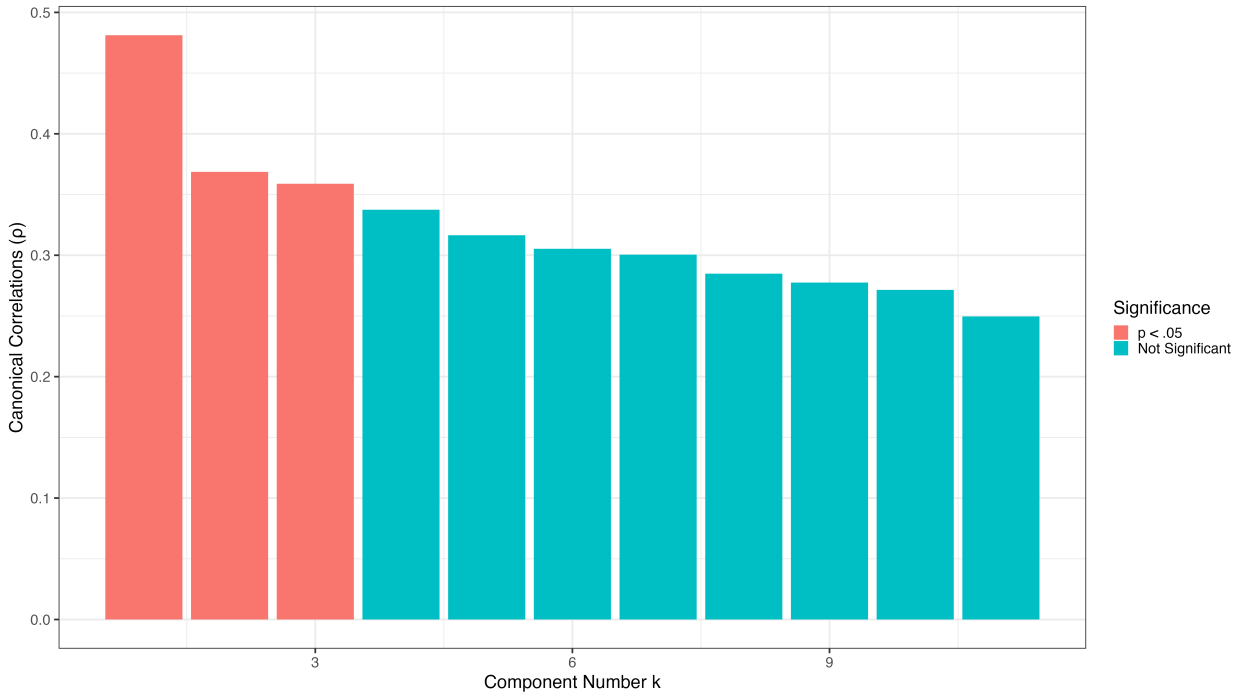

Figure 3.27: Results for ABCD: canonical correlations.

promising alternative, although its control of Type I error is not sufficiently consistent for us to recommend its use in low dimensional settings.

While combootcca has good Type I and II error control, the simulation studies show it can fail to achieve nominal coverage of coordinates with large magnitude, especially when the true direction is sparse. We conjectured that this was due to bias in the estimator which was exacerbated by the bootstrap, and we saw empirical evidence for this in Figures 3.5, 3.11, 3.17, and 3.21. It may be possible to mitigate this bias by estimating and correcting for it. This could be done with the use of a double bootstrap (Davison and Hinkley, 1997, p. 103) or related procedures such as bias-corrected and accelerated intervals (BC_a; Efron and Tibshirani, 1993, p. 184), although this of course comes at additional computational cost and requires further study.

One noteworthy limitation of the approaches we have presented is that we do not correct for multiple comparisons. While this is common in the applied literature, where confidence intervals for the canonical directions are generally considered descriptive and not corrected (e.g., Mišić et al., 2016). this is an important caveat that should be kept in mind particularly with respect to our findings from the ABCD data. While control of the family-wise error rate could be achieved with Bonferroni correction, this may be rather conservative and decrease power, which in some settings is already limited. An alternative would be to

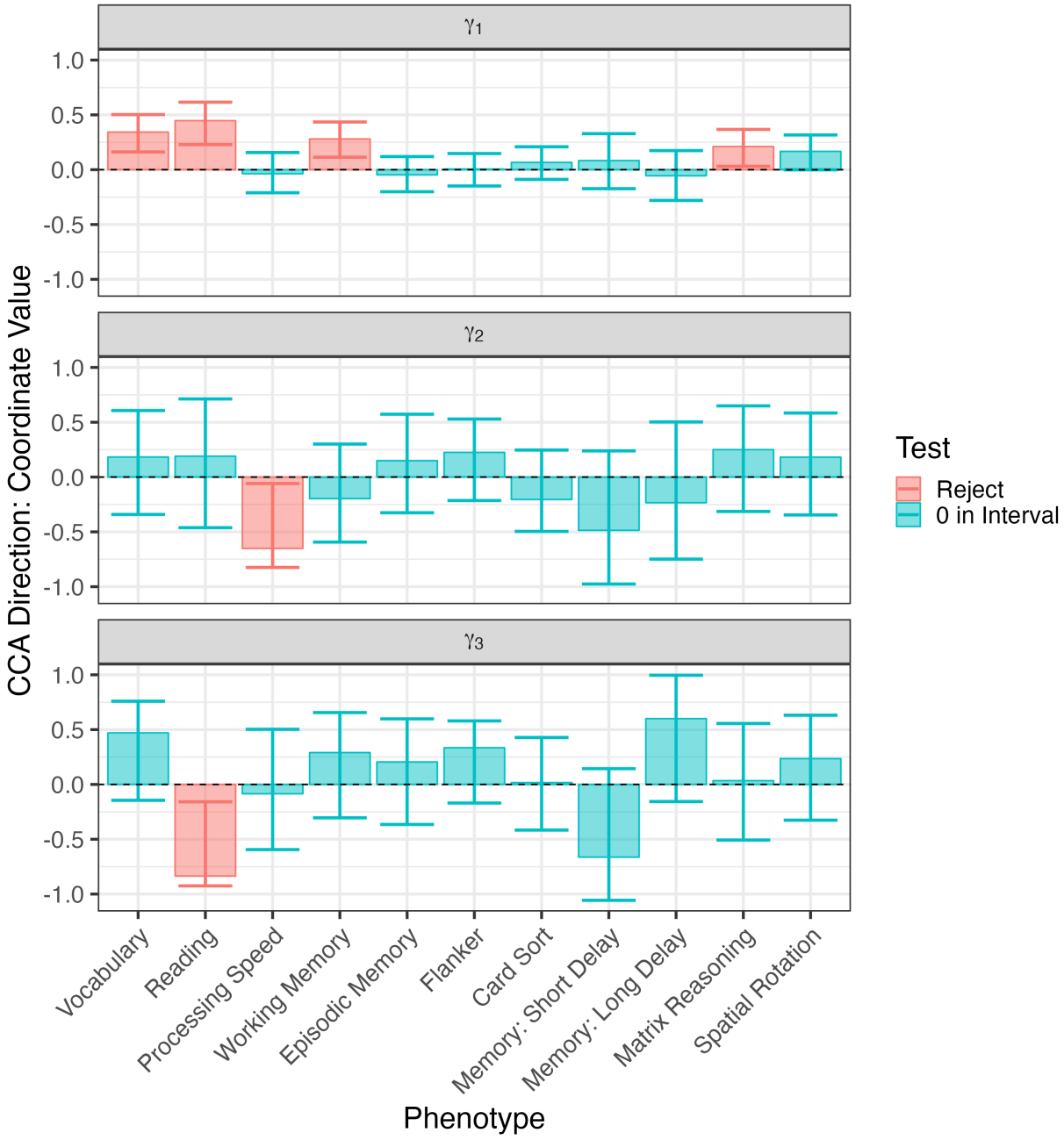

Figure 3.28: Results for ABCD: point estimates and confidence intervals for first three canonical directions of Γ.

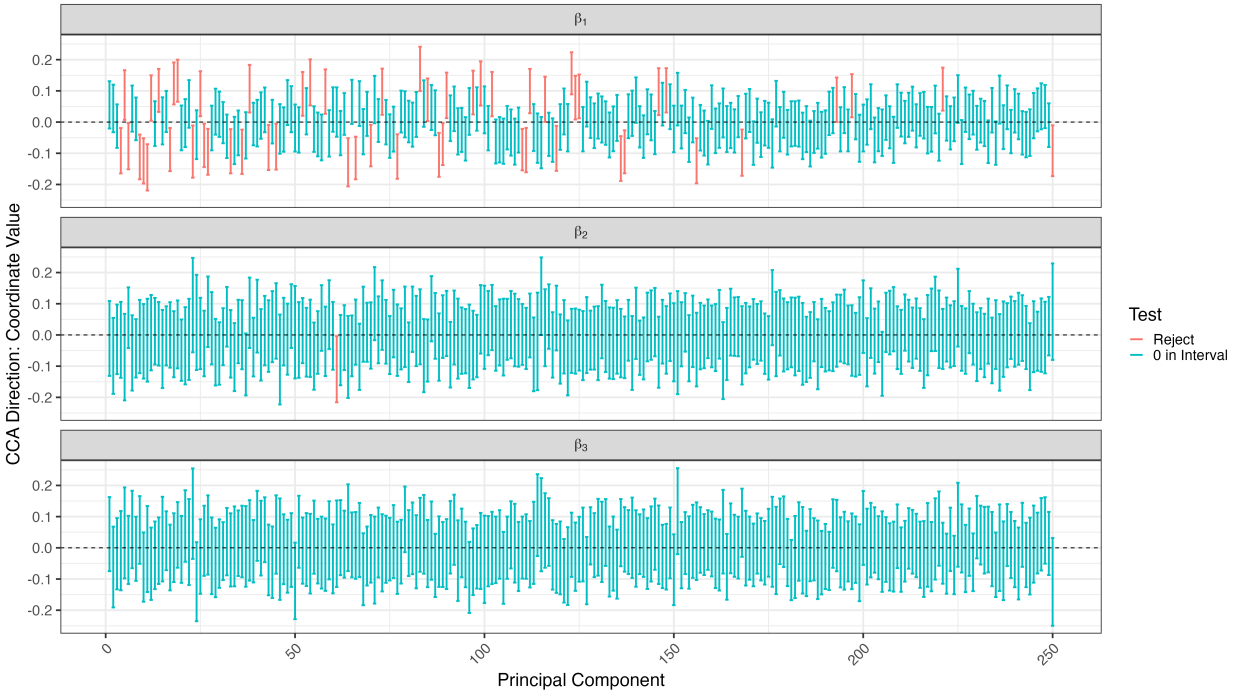

Figure 3.29: Results for ABCD: confidence intervals for first three canonical directions of B.

instead control the false discovery rate (FDR; Benjamini and Hochberg, 1995). While the classical procedure is valid for independent tests and those that are positively dependent (Benjamini and Yekutieli, 2001), it is not immediately clear what sort of dependence might exist among tests for the elements of B and Γ. However, because combootcca already uses a resampling-based procedure, it may be possible to estimate the dependence and to adjust accordingly, as in Yekutieli and Benjamini (1999). We are not selective in terms of which confidence intervals we report for a given direction; thus we should not suffer from decreased coverage rates due to selective reporting (Benjamini and Yekutieli, 2005), but future work may be needed to account for the potentially sequential nature of inference, wherein a subset of canonical directions are selected for further scrutiny based on hypothesis tests applied to the canonical correlations. This could be viewed as testing hypotheses on a tree, and so the approach of Bogomolov et al. (2021) may be applicable.

Future work includes additional theoretical analysis. Given the use of the singular value decomposition in CCA as discussed in Section 3.1.1, recent results in Agterberg et al. (2022) concerning the limiting distribution of entries of singular vectors may be of use for this analysis. In addition, our simulation studies suggest that when $p > q$ but $q \ll N$, the results of Anderson (1999) may still give the correct limiting distribution for $\hat{\Gamma}$. This appears to be in line with theoretical results in Fine (2003), which revisited the work of Anderson (1999)

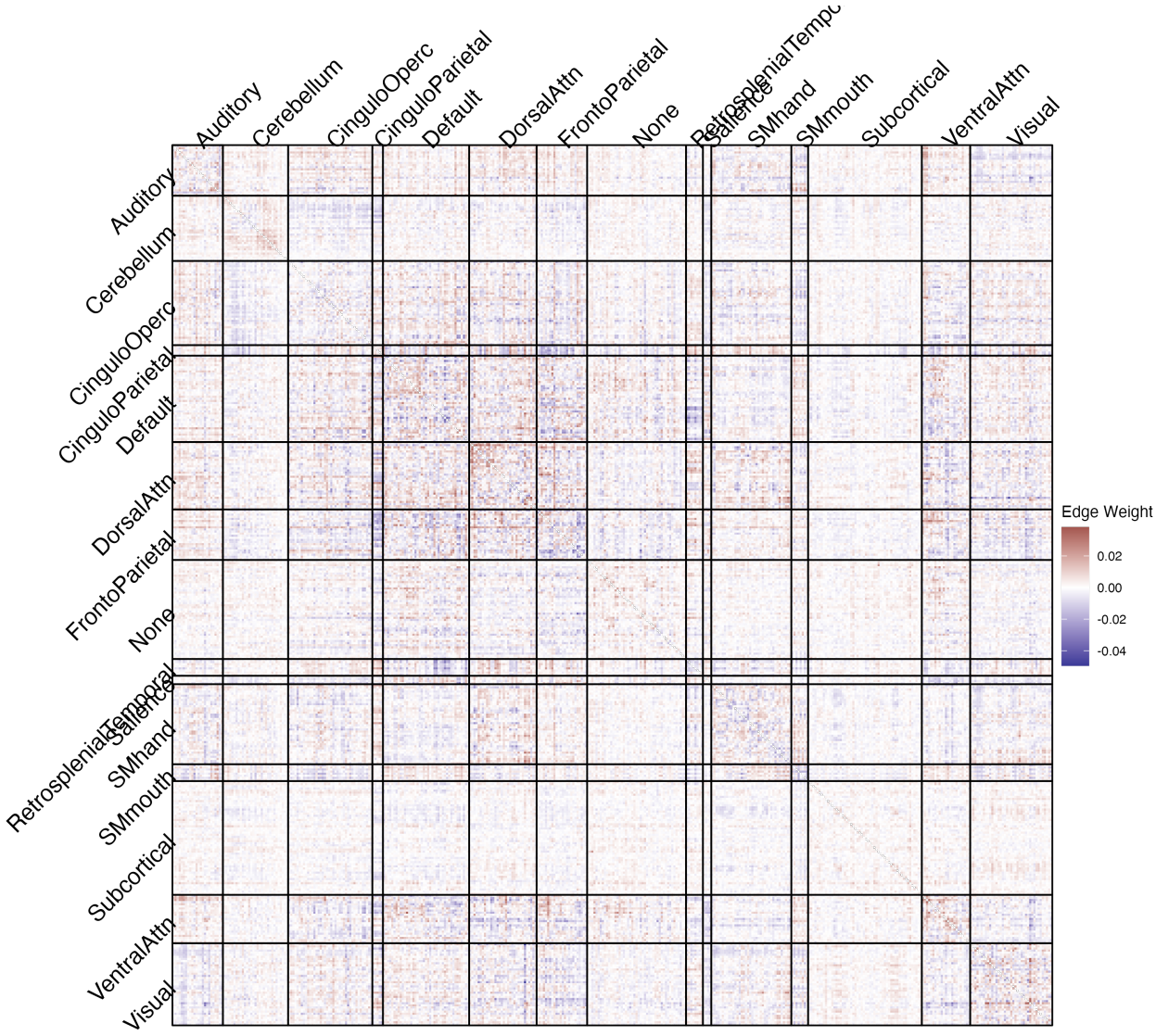

Figure 3.30: Brain connectivity features recovered from first canonical direction β_1.

from an operator- and tensor-focused perspective, but appears to offer results for only the lower dimensional direction. This approach to inference would be of particular utility: the coordinates of the lower-dimensional directions, which often correspond to phenotype, are often of primary interest, and asymptotic confidence intervals are vastly faster to compute than the bootstrap. Nonetheless, given the widespread use of resampling-based strategies in the applied literature, our contribution of combootcca along with demonstration of the pitfalls of related strategies is still valuable.

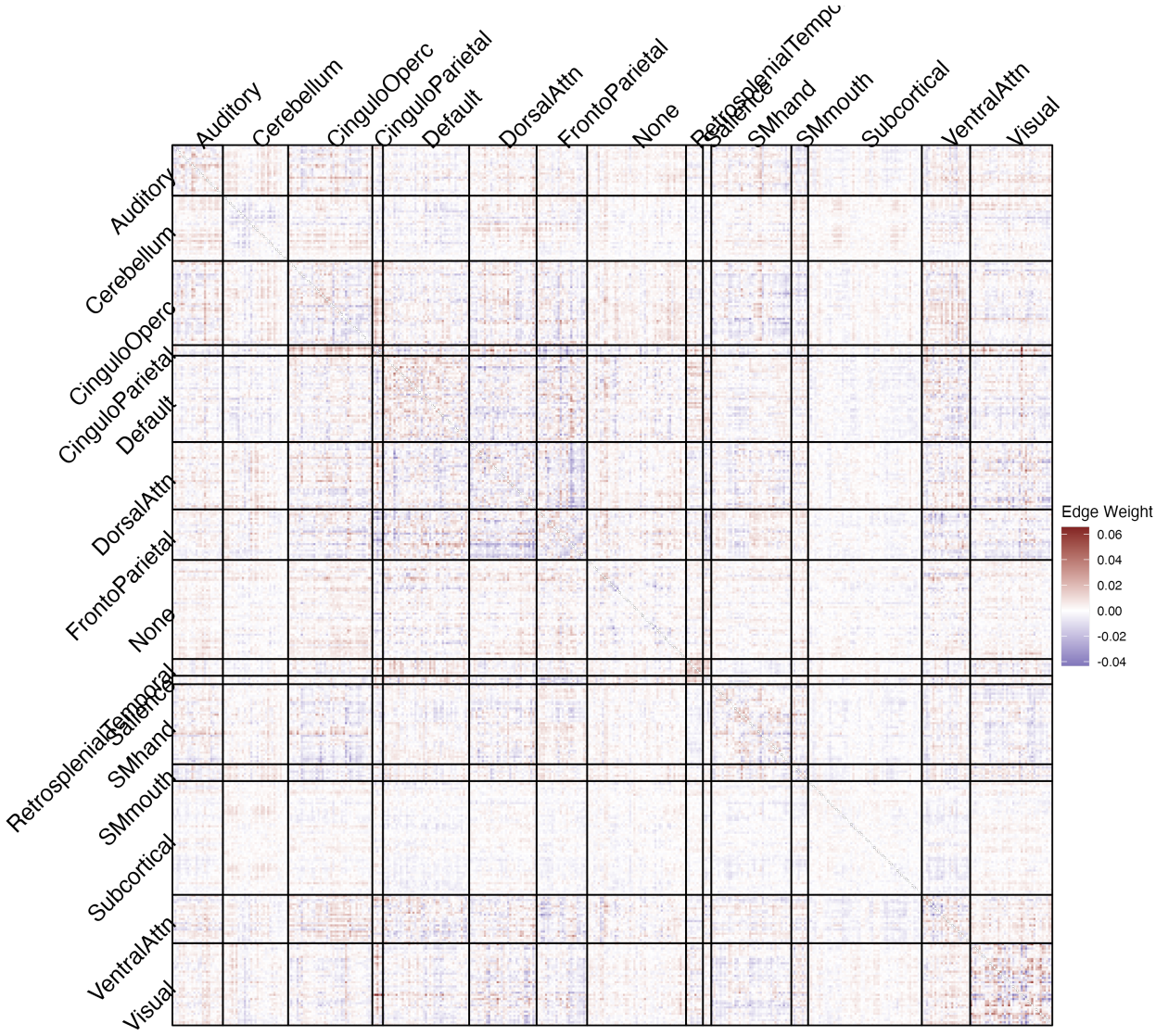

Figure 3.31: Brain connectivity features recovered from second canonical direction β_2.

Another direction for future work involves developing forms of matrix CCA, which would be more closely tailored to our neuroimaging application. Recall that X is the matrix of brain connectivity, but the structure of this matrix is ignored when it is vectorized and reduced using PCA as part of preprocessing. While this transformation is reversible for the purposes of visualization (e.g., Figure 3.33), our methods do not take advantage of this rich structure. There are thus opportunities to develop variants of CCA that involve structure-enforcing penalties (e.g., similar to that of Relión et al. (2019); Kessler et al. (2022)). In Chapter 4,

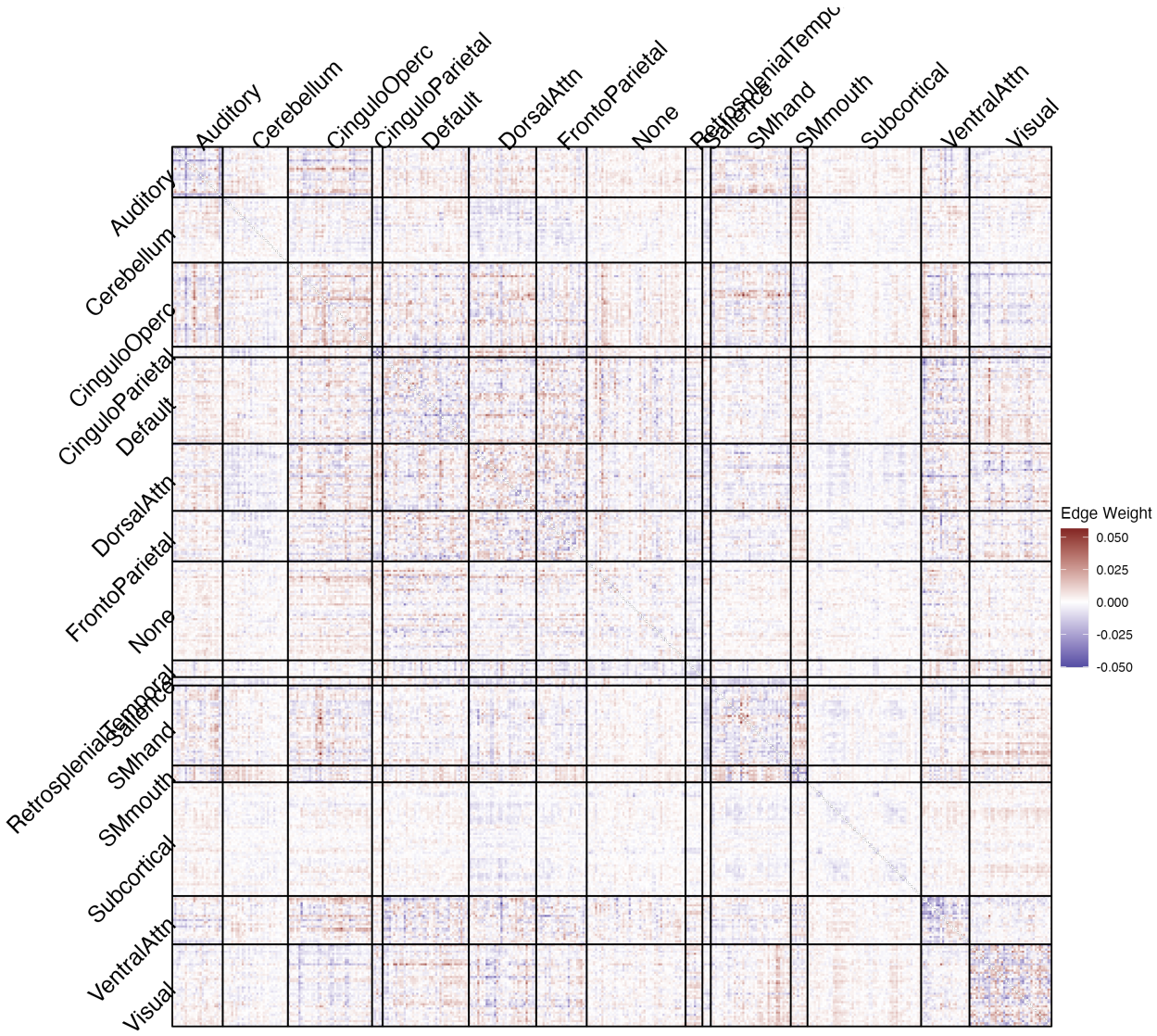

Figure 3.32: Brain connectivity features recovered from third canonical direction β_3.

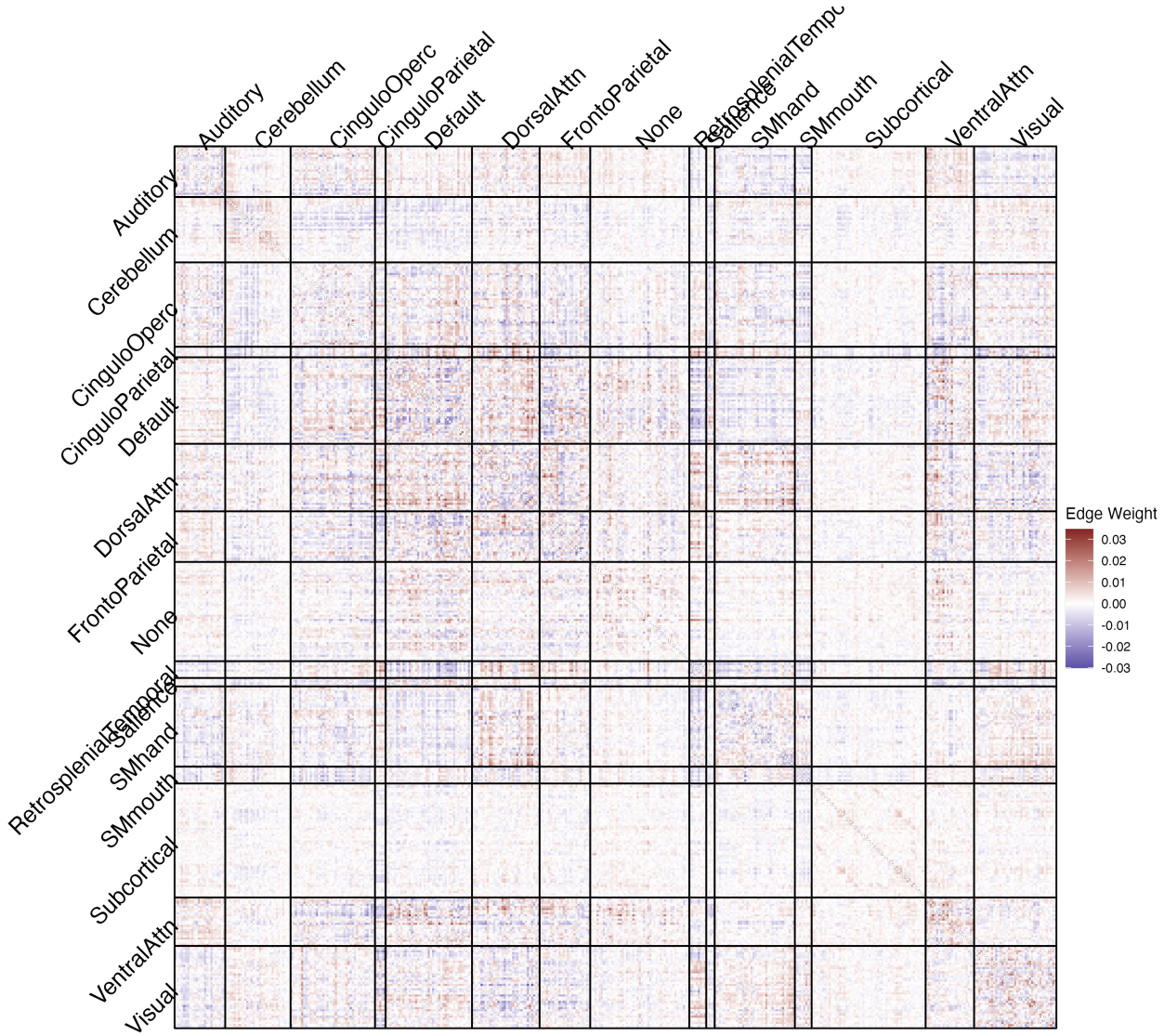

Figure 3.33: Brain connectivity features recovered from first canonical direction β_1 using only significantly nonzero coordinates.

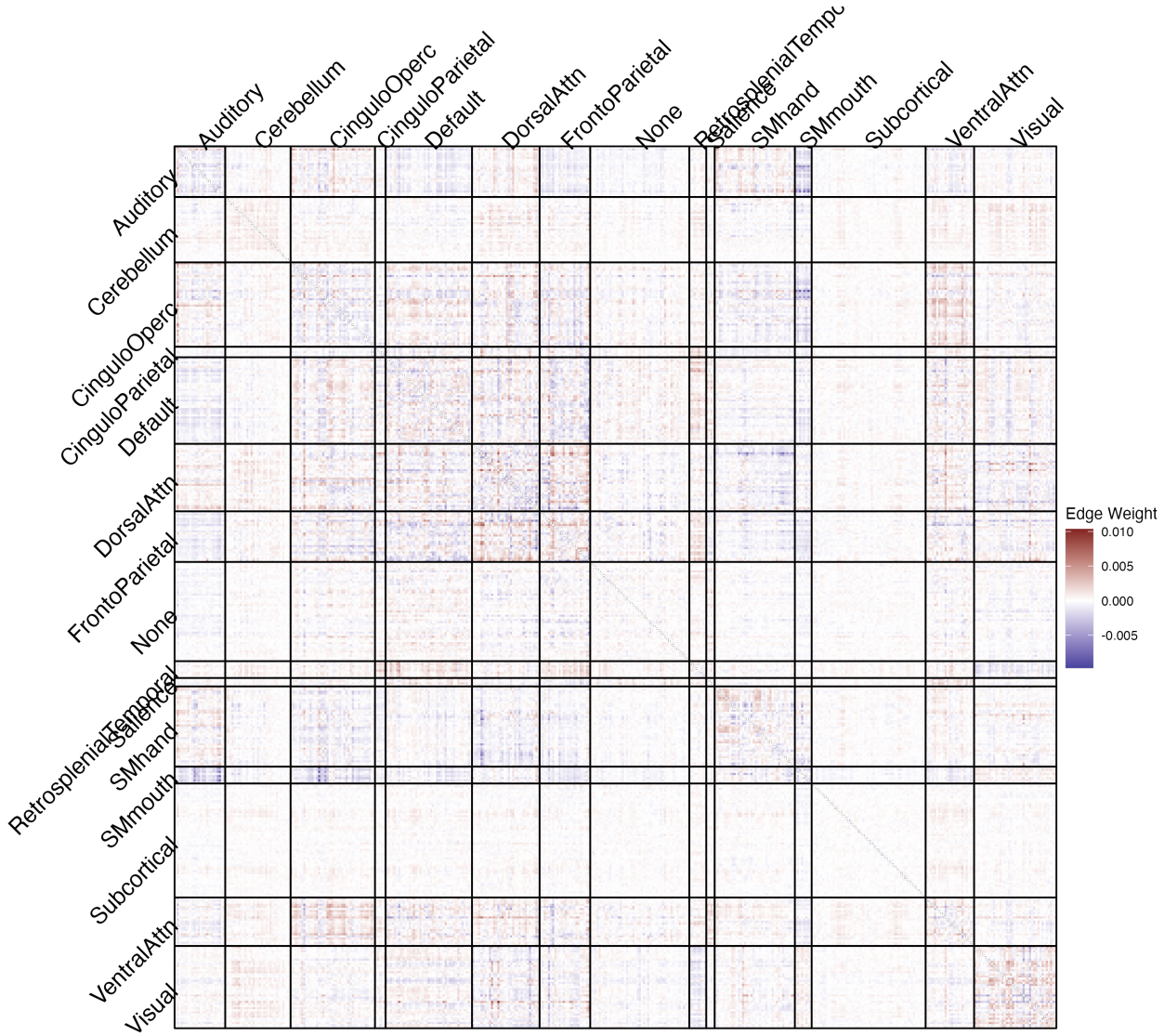

Figure 3.34: Brain connectivity features recovered from second canonical direction β_2 using only significantly nonzero coordinates.

we present work in this direction when seeking low rank structure.

CHAPTER 4

Matrix-Variate Canonical Correlation Analysis

4.1 Introduction

Canonical Correlation Analysis (CCA) is a classical technique (Hotelling, 1935) that has seen a surge of recent interest in both the applied (Wang et al., 2020b) and theoretical (Bykhovskaya and Gorin, 2023; Gao et al., 2017) literature. In brief, CCA is a method applicable to pairs of random vectors, say x and y: it identifies a pair of linear forms (one for each vector) such that when applied to the random vector, the newly transformed random variables, or "canonical variates," are maximally correlated with one another. Additional pairs of linear forms can then be obtained subject to orthogonality constraints.

Many modern applications involve high dimensional data. In the setting where the number of observations exceeds the dimension of either random vector, then the classic approach for estimating canonical directions is no longer applicable. This challenge has sparked interest in regularized forms of CCA, such as ridge CCA (Vinod, 1976) and sparse CCA (Witten et al., 2009). While these methods are generally applicable, more specialized methods can be developed for data with particular structure. One recent example is group regularized CCA (GRCAA; Tuzhilina et al., 2021); GRCCA is applicable when the coordinates of the random vectors can be organized into groups based on some *a priori* knowledge.

In this work, we are motivated by the application from Chapter 3 wherein our samples natively take the form of correlation matrices for each participant in a neuroimaging study. There, we vectorized these matrices and reduced the dimension using PCA, but here we aim to work with the matrices directly and to exploit this structure. More generally, consider the setting where one or both of the random vectors are replaced with random matrices, as may occur with image data or correlation matrices obtained from functional connectivity studies in neuroimaging as discussed above. In other words, we consider matrix-variate CCA. Let $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2}$ and $\mathcal{Y} \in \mathbb{R}^{q_1 \times q_2}$ be random *matrices*. We can apply CCA in this setting by

identifying each matrix with its vectorized analog and performing conventional CCA, i.e., by solving the objective function

$$\begin{aligned}\beta_1, \gamma_1 = \underset{b,g}{\operatorname{argmax}} \operatorname{Corr}\left(\operatorname{vec}(\mathcal{X})^\intercal \operatorname{vec}(b), \operatorname{vec}(\mathcal{Y}^\intercal \operatorname{vec}(g)\right) \\ \text{s.t. } \operatorname{Var}\left(\operatorname{vec}(\mathcal{X})^\intercal \operatorname{vec}(b)\right) = \operatorname{Var}\left(\operatorname{vec}(\mathcal{Y})^\intercal \operatorname{vec}(g)\right) = 1.\end{aligned}$$

where $\operatorname{vec}(\cdot)$ denotes the vectorization of its argument. While valid, this approach fails to exploit any matrix-specific structure that may be present in our original data.

Motivated by the application of CCA to image data, a literature has developed focused on two-dimensional CCA (2D-CCA) (Lee and Choi, 2007). In this setting, the images are represented by matrices, and the goal is to tailor the approach to the image while avoiding the explicit vectorization discussed above. The objective function for 2D-CCA is

$$\begin{aligned}\underset{\ell_x, r_x, \ell_y, r_y}{\operatorname{argmax}} \operatorname{cov}\left(\ell_x^\intercal \mathcal{X} r_x, \ell_y^\intercal \mathcal{Y} r_y, \right) \\ \text{s.t. } \operatorname{Var}\left(\ell_x^\intercal \mathcal{X} r_x\right) = \left(\ell_y^\intercal \mathcal{Y} r_y\right) = 1.\end{aligned}$$

In words, this approach learns "left" and "right" transforms for both $\mathcal{X}$ and $\mathcal{Y}$. Together, the left and right transformations reduce the matrix to a vector: applying the left transform returns a vector, and then applying the right transform returns a scalar. Lee and Choi (2007) propose an algorithm that alternates between finding left and right transforms respectively: when one is held fixed, the problem reduces to classic vector-variate CCA. For image data, this approach has vastly reduced computational cost relative to the vectorized approach, and it often recovers better canonical directions in the sense of achieving improved performance on downstream tasks, e.g., classification (Lee and Choi, 2007)

The canonical directions themselves are implicit in this formulation, but can be made explicit by observing

$$\begin{aligned}\ell_x^\intercal \mathcal{X} r_x &= \operatorname{tr}\left(\ell_x^\intercal \mathcal{X} r_x\right) \\ &= \operatorname{tr}\left(r_x \ell_x^\intercal \mathcal{X}\right) \\ &= \operatorname{tr}\left(\left(\ell_x r_x^\intercal\right)^\intercal \mathcal{X}\right) \\ &= \operatorname{vec}\left(\ell_x r_x^\intercal\right)^\intercal \operatorname{vec}\left(\mathcal{X}\right) \\ &= \operatorname{vec}\left(B\right)^\intercal \operatorname{vec}\left(\mathcal{X}\right) \\ &= \beta^\intercal \operatorname{vec}\left(\mathcal{X}\right),\end{aligned}$$

where $B = \ell_x r_x^\intercal$, and $\beta = \operatorname{vec}(B)$. Since B is the outer product of two vectors, it has

rank at most 1, and an analogous result follows by symmetry for Γ. This rank constraint on the (implicit) canonical directions does not seem to have been originally recognized in the 2D-CCA literature, although it was noted recently in Chen et al. (2021). However, this insight helps to explain why 2D-CCA is much more computationally efficient: there are $p_1 + p_2 + q_1 + q_2$ free parameters in 2D-CCA (corresponding to l_x, r_x, l_y, and r_y), whereas vectorized CCA has $p_1 \times p_2 \times q_1 \times q_2$ free parameters (since all of the entries of B and Γ are free). Moreover, if the interesting signal in the images is low rank, then this constraint may make it easier to recover the signal in otherwise high dimensional settings. However, there are some shortcomings to this approach. First, it does not readily generalize to other rank constraints, e.g., there is no immediate way to modify the algorithm to obtain implicit canonical directions of rank say 2. Indeed, if the left and right transforms are taken to be say matrices with 2 columns, then the "trick" no longer works: applying the left transform will then give nontrivial matrices rather than vectors, which precludes the application of the classic CCA approach that the 2D-CCA algorithm requires. Second, the alternating algorithm is mostly heuristic and thus can be sensitive to initialization.

In order to overcome these shortcomings, we propose a more general approach for matrix-variate CCA that we call matcca (MATrix CCA). Our objective can be written more explicitly as

$$\begin{aligned} \beta_1, \gamma_1 = \underset{b,g}{\operatorname{argmax}} & \operatorname{Corr}\left(\operatorname{tr}(\mathcal{X}^\intercal b), \operatorname{tr}(\mathcal{Y}^\intercal g)\right) \\ \text{s.t. } & \operatorname{Var}\left(\operatorname{tr}\left(\mathcal{Y}^\intercal g\right)\right) = \operatorname{Var}\left(\operatorname{tr}\left(\mathcal{X}^\intercal b\right)\right) = 1 \\ & \text{and } \operatorname{rank}(b) \leq K_{\mathcal{X}}, \operatorname{rank}(g) \leq K_{\mathcal{Y}}. \end{aligned}$$

If we take $K_x = K_y = 1$, then this coincides with the 2D-CCA objective, although writing it this way will help to suggest an alternative algorithmic approach. As we shall see later, we will not directly enforce this rank constraint but instead use its convex relaxation: the nuclear norm (Fazel et al., 2001).

Both Safayani et al. (2018) and Chen et al. (2021) study 2D-CCA and consider its extension to the tensor setting. However, it does not appear that their formulations allow for both more than 1 canonical direction *and* canonical directions with rank greater than 1. Indeed, when 2D-CCA was initially introduced, there is ambiguity as to how the solution actually corresponds to CCA. For example, when using the algorithm presented in Lee and Choi (2007), if d_1 and d_2 are both greater than 1, then each matrix will have a corresponding canonical variate *matrix* of size $d_1 \times d_2$. In conventional CCA, there is a pairwise correlation among the canonical variates, but it is not clear how this can be understood when the vector of canonical variates becomes instead a matrix.

Given the intimate connection between CCA and regression discussed in Section 3.1, it is natural to seek analogs of this approach in the regression literature. Regularized matrix-variate regression was proposed in Zhou and Li (2014). In their setting, $y^{(i)}$ is a scalar and they compare the solution of

$$\min_B n^{-1} \sum_{i=1}^{N} \left(y^{(i)} - \operatorname{tr}\left(B^\intercal \mathcal{X}^{(i)}\right)\right)^2 + \lambda \left\|\operatorname{vec}(B)\right\|_1, \tag{4.1}$$

with

$$\min_B n^{-1} \sum_{i=1}^{N} \left(y^{(i)} - \operatorname{tr}\left(B^\intercal \mathcal{X}^{(i)}\right)\right)^2 + \lambda \left\|B\right\|_\star, \tag{4.2}$$

where

$$\left\|\mathcal{X}\right\|_\star = \sum_{k=1}^{K} \sigma_k(\mathcal{X}),$$

and $\sigma_k(\mathcal{X})$ gives the kth singular value of the matrix $\mathcal{X}$. Equation (4.1) is tantamount to vectorizing the matrix $\mathcal{X}$ and applying LASSO regularization (Tibshirani, 1996), whereas Equation (4.2) involves a relaxation of a rank constraint which had previously been considered for matrix completion (Cai et al., 2010). Zhou and Li (2014) demonstrate in simulation studies that when the true signal is low rank, (4.2) has superior performance to (4.1). They also consider a family of closely related penalty functions, all of which operate on the vector of singular values, but in our work we will restrict our attention to the nuclear norm, which corresponds to the LASSO penalty applied to the singular values. The approach that we will propose in Section 4.2 can be understood as the CCA-generalization of the regularized matrix regression approach of Zhou and Li (2014).

There is precedent for the use of low rank promoting penalties in the neuroimaging literature. Brzyski et al. (2020) proposed penalizing scalar-on-matrix regression with both a LASSO (ℓ_1) and nuclear norm and applied this method to predict language test scores using functional brain connectivity networks. However, to the best of our knowledge the penalized CCA formulation that we present in Section 4.2 is novel in and of itself and also in its application to neuroimaging data.

In our methodological development, we will depend heavily on work by Mai and Zhang (2019), wherein they reformulate classical CCA as a constrained quadratic optimization problem, and then use this formulation to introduce penalization. Their approach was designed to recover sparse solutions for vector-variate CCA. Here, we review their approach in the vector case and then in Section 4.2, we show how we extend it to work with matrix-variate data. In their work, they show that in the low dimensional setting (i.e., both p and

q smaller than N) the CCA objective, i.e.,

$$\left(\hat{\beta}_k^{\text{CCA}}, \hat{\gamma}_k^{\text{CCA}}\right) = \underset{(b_k, g_k)}{\text{argmax}}\, g_k^\intercal \hat{\Sigma}_{YX} b_k,$$
$$\text{s.t. } b_k^\intercal \hat{\Sigma}_x b_l = g_k^\intercal \hat{\Sigma}_y g_l = \mathbb{I}(k = l)$$

can be rewritten as the solution of a constrained quadratic optimization problem

$$\left(\hat{\beta}_k^{\text{CCA}}, \hat{\gamma}_k^{\text{CCA}}\right) = \underset{(b_k, g_k)}{\text{argmin}} \left\{ \frac{1}{2n} \sum_{i=1}^{n} \left(Y_i^\intercal g_k - X_i^\intercal b_k\right)^2 + g_k^\intercal \left(\sum_{i<k} \hat{\rho}_l \hat{\Sigma}_{YY} \hat{\gamma}_l \cdot b_l^\intercal \hat{\Sigma}_{XX} \right) b_k \right\}$$
$$\text{s.t. } b_k^\intercal \hat{\Sigma}_{XX} b_k = g_k^\intercal \hat{\Sigma}_{YY} g_k = 1.$$

While there are still constraints that control the scaling of $\hat{\gamma}$ and $\hat{\beta}$, the orthogonality constraints have been absorbed into the second term. This new formulation is useful, as in the high dimensional case, we can add a penalty to encourage sparsity and obtain

$$\left(\hat{\beta}_k^{\text{CCA}}, \hat{\gamma}_k^{\text{CCA}}\right) = \underset{(g_k, b_k)}{\text{argmin}} \Bigg\{ \frac{1}{2n} \sum_{i=1}^{n} \left(Y_i^\intercal g_k - X_i^\intercal b_k\right)^2 + g_k^\intercal \left(\sum_{i<k} \hat{\rho}_l \hat{\Sigma}_{YY} \hat{\gamma}_l \cdot b_l^\intercal \hat{\Sigma}_{XX} \right) b_k$$
$$+ \lambda_X \left\| b_k \right\|_1 + \lambda_Y \left\| g_k \right\|_1 \Bigg\}$$
$$\text{s.t. } b_k^\intercal \hat{\Sigma}_x b_k = g_k^\intercal \hat{\Sigma}_y g_k = 1.$$

Mai and Zhang (2019) suggest solving this optimization problem using an alternating algorithm, i.e., fix g_k and optimize in β_k, then fix β_k and optimize in g_k, and so on, until convergence. They prove that for a fixed g_k, the optimal β_k is given by the solution to a LASSO penalized regression problem, and conversely that for a fixed β_k, the optimal g_k is given by the solution to a LASSO penalized regression problem.

4.2 Methods

In order to introduce nuclear norm penalization for matrix-CCA, we employ the framework of Mai and Zhang (2019). As the authors note, their approach can be generalized by replacing the LASSO penalties with another pair of penalties, say P_X and P_Y, in which case the

objective is

$$\begin{aligned}\left(\hat{\beta}_k^{\text{CCA}}, \hat{\gamma}_k^{\text{CCA}}\right) = \underset{(b_k, g_k)}{\text{argmin}} \Bigg\{ & \frac{1}{2n} \sum_{i=1}^{n} \left(Y_i^\intercal g_k - X_i^\intercal b_k\right)^2 + g_k^\intercal \left(\sum_{i<k} \hat{\rho}_l \hat{\Sigma}_{YY} \hat{\gamma}_l \cdot b_l^\intercal \hat{\Sigma}_{XX} \right) b_k \\ & + P_X\left(b_k\right) + P_Y\left(g_k\right) \Bigg\} \\ \text{s.t. } & b_k^\intercal \hat{\Sigma}_x b_k = g_k^\intercal \hat{\Sigma}_y g_k = 1.\end{aligned}$$

So long as the penalties are positively homogeneous (i.e., for any $C > 0$, $P_Y(Cw) = CP_Y(w)$ for arbitrary $w \in \mathbb{R}^q$ and $P_X(Cv) = CP_X(v)$ for arbitrary $v \in \mathbb{R}^p$), then the alternating updates can be given by the solution to penalized regression problems. While Mai and Zhang (2019) discuss a variety penalties to encourage structured sparsity with vector-variate data (e.g., group lasso (Yuan and Lin, 2006), fused lasso (Tibshirani et al., 2005)), we propose to instead consider penalties appropriate for matrix-variate data. In particular, for $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2}$, define

$$P_X(\mathcal{X}) = \lambda_X \left\| \mathcal{X} \right\|_\star,$$

and define P_Y analogously. As a norm, P_X is absolutely homogeneous, and thus is also positively homogeneous and satisfies the condition of Mai and Zhang (2019)'s Lemma 3; thus, our alternating updates can be given by the solution to a nuclear norm penalized regression problem. An alternating algorithm is then given by the following. Suppose that we have already recovered $k-1$ canonical pairs and associated canonical correlations given by

$$\hat{\Gamma}_{k-1} = \left(\hat{\gamma}_1, \ldots, \hat{\gamma}_{k-1}\right), \hat{B}_{k-1} = \left(\hat{\beta}_1, \ldots, \hat{\beta}_{k-1}\right), R_{k-1} = \text{diag}\left(\hat{\rho}_1, \ldots, \hat{\rho}_{k-1}\right).$$

First, we compute $\Omega_k = I_n - \frac{1}{n} Y \Gamma_{k-1} R_{k-1} B_{k-1} X^\intercal$. This matrix will serve to remove variation explained by the preceding canonical pairs. Next, initialize $\left\{\hat{\gamma}_k^{(0)}, \hat{\beta}_k^{(0)}\right\}$. The iterative updates then take the form

$$\begin{aligned}\tilde{Y}_k^{(m)} &= \Omega_k^\intercal Y \hat{\gamma}_k^{(m)} \\ \breve{\beta}_k &= \underset{b_k}{\text{argmin}} \left\{ \frac{1}{2n} \| \tilde{Y}_k^{(m)} - X b_k \|_2^2 + P_X\left(b_k\right) \right\} \\ \hat{\beta}_k^{(m)} &= \left\{ \breve{\beta}_k^{(m)\intercal} \hat{\Sigma}_{XX} \breve{\beta}_k^{(m)} \right\}^{-1/2} \cdot \breve{\beta}_k^{(m)} \\ \tilde{X}_k^{(m)} &= \Omega_k^\intercal X \hat{\beta}_k^{(m)} \\ \breve{\gamma}_k &= \underset{g_k}{\text{argmin}} \left\{ \frac{1}{2n} \| \tilde{X}_k^{(m)} - Y g_k \|_2^2 + P_X\left(g_k\right) \right\} \\ \hat{\gamma}_k^{(m)} &= \left\{ \breve{\gamma}_k^{(m)\intercal} \hat{\Sigma}_{YY} \breve{\gamma}_k^{(m)} \right\}^{-1/2} \cdot \breve{\gamma}_k^{(m)},\end{aligned} \tag{4.3}$$

after which m is incremented and we repeat until convergence, returning the estimators $\hat{\beta}_k, \hat{\gamma}_k$ for $k = 1, \ldots, K$, with K being specified by the user.

Because the algorithm is iterative and also has an outer loop over the canonical pairs, we may need to solve these inner problems many times. This computational burden is further compounded by the need to tune parameters with cross validation. When using the LASSO penalty, we can use the `glmnet` package (Friedman et al., 2010b) in `R`, which has been highly optimized to solve LASSO penalized regression very quickly, although because it is only used to solve an "inner" problem we cannot directly use its cross-validation functionality. In our setting, where $\mathcal{X}$ is a matrix and P_X is the nuclear norm penalty, we need to repeatedly solve the optimization problem

$$\theta^\star = \underset{\theta}{\operatorname{argmin}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(y^{(i)} - \operatorname{tr}\left(\operatorname{mat}(\theta)^\intercal X^{(i)} \right) \right)^2 \right\} + \lambda \left\| \operatorname{mat}(\theta) \right\|_\star , \tag{4.4}$$

where $\operatorname{mat}(\theta) \in \mathbb{R}^{p_1 \times p_2}$. Surprisingly, we were unable to find an `R` package that implements nuclear norm penalized trace regression, so we implement our own solver following the contractive Peaceman-Rachford splitting method (PRSM) (Eckstein and Bertsekas, 1992) adapted for nuclear norm penalized regression as described in Fan et al. (2017). Letting Y be the vector of $y^{(i)}$'s and constructing X by vectorizing each $\mathcal{X}^{(i)}$, transposing it, and then stacking them together, the updates are

$$\begin{aligned}
\theta_1^{k+1} &= \left(2X^\intercal X/N + \beta I\right)^{-1} \left(\beta \theta_2^{(k)} + \rho^{(k)} + 2X^\intercal Y/N \right) \\
\rho^{(k+1/2)} &= \rho^{(k)} - \alpha\beta \left(\theta_1^{(k+1)} - \theta_2^{(k)} \right) \\
\theta_2^{(k+1)} &= \operatorname{vec}\left(S_{\lambda/\beta}\left(\operatorname{mat}\left(\theta_1 - \rho^{(k+1/2)} \right) \right) \right) \\
\rho^{(k+1)} &= \rho^{(k+1/2)} - \alpha\beta \left(\theta_1^{(k+1)} - \theta_2^{(k+1)} \right).
\end{aligned} \tag{4.5}$$

We set the tuning parameters to $\alpha = 0.9$ and $\beta = 1$ in line with Fan et al. (2017). $S_{\lambda/\beta}(\cdot)$ is the singular value soft-thresholding function, i.e., if $Z \in \mathbb{R}^{m \times n}$ with rank r and singular value decomposition $Z = U\Lambda V^\intercal$, then

$$S_\tau(Z) = U \operatorname{diag}\left[(\lambda_1 - \tau)_+, (\lambda_2 - \tau)_+, \ldots, (\lambda_r - \tau)_+ \right] V^\intercal,$$

where $(z)_+ = \max(z, 0)$; in words, it applies the soft-thresholding operator to the singular values of its argument and then reconstructs the matrix. We terminate when $\left\| \theta_1^k - \theta_2^k \right\|_2$ is suitably small and return θ_2^k. Thus, (4.5) offers a way to solve the nuclear norm-penalized regression (NNR) problem in (4.4), and by plugging this in to the iterative penalized CCA

solver (4.3), we have our algorithm for matcca.

In order to improve the speed of our implementation, we note that the matrix inverse in the first line of (4.5) does not vary with k. Rather than explicitly computing the inverse, which may be unstable, we instead perform Cholesky factorization and obtain

$$2X^\intercal X/N + \beta I = LL^\intercal.$$

With these factors in hand, each time we need to find θ_1^{k+1}, we can instead evaluate

$$\begin{aligned}\theta_1^{k+1} &= (LL^\intercal)^{-1}\left(2X^\intercal X/N + \beta I = LL^\intercal\right) \\ &= (L^\intercal)^{-1} L^{-1}\left(2X^\intercal X/N + \beta I\right),\end{aligned}$$

Because L is lower triangular, the last line can be evaluated quickly by backsolving first against L, and then that result can be used to quickly forward solve against $L^\intercal$. This approach is much faster than solving the system of equations each time, and it is more numerically stable than finding the inverse explicitly. However, the cost (in both computational and memory complexity) of evaluating $X^\intercal X$ is high, since it has $O(p_1^2 p_2^2)$ entries, and with large matrices $\mathcal{X}$ this becomes intractable. As discussed in Section 4.5, future works includes algorithmic improvements to facilitate solving against this very large matrix without explicitly computing it.

An `R` package implementing the LASSO penalized SCCA approach of Mai and Zhang (2019) is available from the author's website. However, the software in its current state cannot be applied in our setting. First, it is hard-coded to use `glmnet` for its iterative updates, which precludes the use of other penalization schemes. In the course of the present work, we have extended this software to handle more arbitrary penalization strategies, and we have also fixed several bugs and made other improvements.

We tune the parameters associated with the penalty using cross-validation. When only a single canonical pair is sought, a natural choice would be the parameter that offers the best averaged canonical correlation on held out data when using the learned directions. However, when retrieving more than a single pair, this metric doesn't take into account performance of canonical directions beyond the first. While in theory one could tune the parameters to different values for each direction, this can very quickly explode in terms of computational complexity, so we require that all directions have the same tuning parameter, although distinct parameters can be chosen for x and y, which may especially be appropriate if they are different types of objects (e.g., x is a vectorized matrix whereas y is a simple vector). In order to consider canonical correlations beyond the first, when performing cross-validation we learn the directions on training data and then evaluate all the empirical correlations in

hold-out data. From this we obtain a matrix of correlations: we take its entry-wise square (so that the entries are non-negative and can be interpreted like the R^2 of a regression) and then take its trace to sum across canonical directions. We choose the values of the tuning parameters that maximizes this metric.

4.3 Numerical Results

We consider the performance of our method in a variety of synthetic settings, where we consider matrix-scalar, matrix-vector, and matrix-matrix CCA. In each case, for the part of the optimization problem involving a matrix, we use our algorithm with either the LASSO penalty, implemented in `glmnet` (Friedman et al., 2010b), or with the nuclear norm penalized regression (NNR) approach as described in Section 4.2. We tune all parameters using 5-fold cross-validation using the procedure described earlier. We then fit CCA to the entire data using the selected parameter.

For notational simplicity in this section we describe everything in terms of vector CCA, but be aware that many of these "vectors" will, in some of our simulations, be flattened matrices. In each of our simulations, we draw our data from a factor model based on Bach and Jordan (2005). This involves first drawing N-many replicates of $z \sim \mathcal{N}_K(0, I_K)$ (K will vary across the simulations), which corresponds to the latent signal shared by x and y. Then we gather these into the matrix

$$Z = \begin{bmatrix} z_1^\intercal \\ z_2^\intercal \\ \cdots \\ z_N^\intercal \end{bmatrix}.$$

The matrices $W_x \in \mathbb{R}^{p \times K}$ and $W_y \in \mathbb{R}^{q \times K}$ govern how this shared signal manifests in our data, and conditioning on Z we draw data matrices

$$X = ZW_x^\intercal + E_X$$
$$Y = ZW_y^\intercal + E_Y,$$

where $E_X \in \mathbb{R}^{N \times p}$ and $E_Y \in \mathbb{R}^{N \times q}$ are matrices with entry-wise independent normal random variables. As a proxy for signal-to-noise ratio (SNR), we vary the standard deviation of the entries of E_X and E_Y in our experiments.

In this simulation setup, the kth column of W_x is proportional to the true canonical direction β_k, and the kth column of W_y is proportional to the true canonical direction γ_k. Accordingly, in each simulation we assess our methods by computing the absolute value of

the cosine similarity between the kth estimated direction and the kth column of either W_x or W_y, where the cosine similarity of two vectors u and v is defined as $u^\intercal v \left\|u\right\|_2^{-1} \left\|v\right\|_2^{-1}$. We repeat each simulation setting 100 times for each SNR level, and we summarize the cosine similarity with the mean and plot this and its 95% confidence interval (based on the empirical standard errors) in each simulation study. In all of our simulation studies, we fix $N = 100$ whereas W_x always has 400 rows (and thus X has 400 columns). Thus, regularization is necessary in order to obtain a solution.

4.3.1 Scalar-Matrix CCA

In our first simulation, $\mathcal{X} \in \mathbb{R}^{20\times 20}$ is a random matrix, whereas $y \in \mathbb{R}^1$ is a random scalar. This is essentially a regression problem, but because regression can be understood as a special case of CCA where $K = 1$, we can still deploy the tools that we developed to work in the CCA setting.

W_x comprises a single column, but this column corresponds to a vectorized matrix. We construct the corresponding matrix $\text{mat}\left(W_x\right) \in \mathbb{R}^{20\times 20}$, with $\text{mat}\left(W_x\right)_{[1:5,1:5]} = 1$ and 0's elsewhere. This corresponds to a signal that is both sparse, as only 25 of the 400 entries are nonzero, and low rank, as it can be written as the outer product of two simple vectors of 1's and 0's. Since y is a scalar, W_y is trivially fixed at 1. While we will use both LASSO and NNR to penalize the updates associated with $\mathcal{X}$, we impose no penalties on the updates associated with y since its solution is trivial. We then draw Z, X, and Y in the manner described earlier. Since the estimated direction associated γ is trivial (as it is ± 1), we only consider the cosine similarity of $\hat{\beta}_1$ with W_x, which we plot in Figure 4.1 as a function of the standard deviation of the noise. Unsurprisingly, both LASSO and NNR perform better in lower noise settings, although NNR consistently out-performs LASSO across the noise range until both methods essentially revert to random guessing in the presence of high noise. This performance is especially noteworthy, since the structure of the signal is both sparse (which is good for LASSO) and low rank (which is good for NNR).

4.3.2 Vector-Matrix CCA

In this simulation, $\mathcal{X} \in \mathbb{R}^{20\times 20}$ is again a random matrix, but now $y \in \mathbb{R}^2$ is a random *vector*, and $K = 2$. We consider two different sub-settings in this simulation study. In the first,

$$W_y = \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix},$$

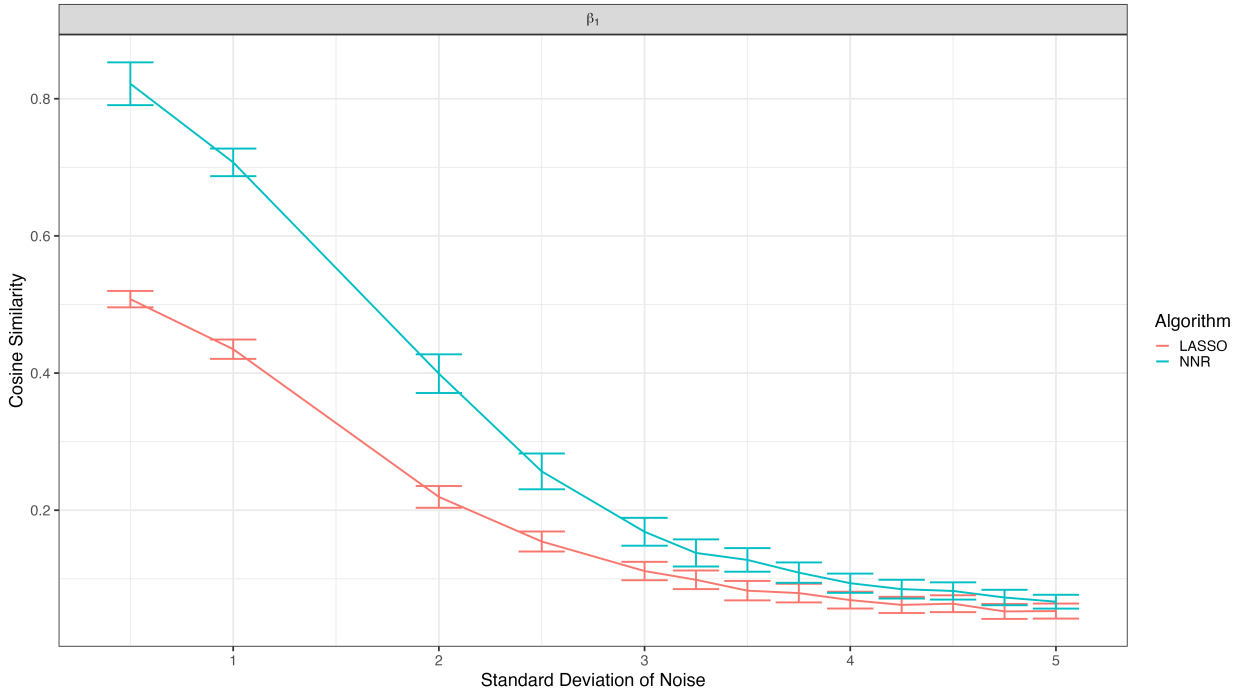

Figure 4.1: Cosine similarity for matrix-scalar setting.

in which case each of the coordinates of y are merely proxies for the two latent variables. In order to keep the canonical directions distinct, however, we make the second coordinate a weaker proxy for the second latent variable. In the second setting, we let

$$W_y = \begin{bmatrix} 1 & 0.5 \\ 1 & -0.5 \end{bmatrix},$$

in which case the coordinates of y correspond to orthogonal linear functions of the latent vector. While we will use both LASSO and NNR to penalize the updates associated with $\mathcal{X}$, we impose no penalties on the updates associated with Y since it is a low-dimensional vector.

The signal in $\mathcal{X}$ is the same in both cases. Since $K = 2$, W_x now has two columns. We set the first column the same as in the first simulation study, i.e., with $\text{mat}\left((W_x)_1\right)_{[1:5,1:5]} = 1$ and 0's elsewhere, while we set the second column as $\text{mat}\left((W_x)_2\right)_{[15:20,15:20]} = 0.5$ and 0's elsewhere, The first signal is a sparse, low-rank signal at the top left of the matrix, whereas the second is a sparse, low-rank signal at the bottom right of the matrix, and as was the case with W_y, we attenuate the signal associated with the second direction to allow for separation.

In Figure 4.2, we show the cosine similarities for the estimates $\hat{\beta}_1, \hat{\beta}_2, \hat{\gamma}_1$, and $\hat{\gamma}_2$ when W_y is diagonal (the first case); Figure 4.3 shows analogous results for the case where W_y

involves mixing the latent variables in each coordinate of y. As expected, estimation quality decreases with growing noise magnitude. However, regardless of the structure of W_y, NNR consistently out-performs LASSO for estimating both directions β_1 and β_2, although both methods do better on β_1 than β_2 which is reasonable as β_1 is a stronger signal. There is not an appreciable difference between the two methods when it comes to estimating γ_1 or γ_2 although γ_2 is typically harder.

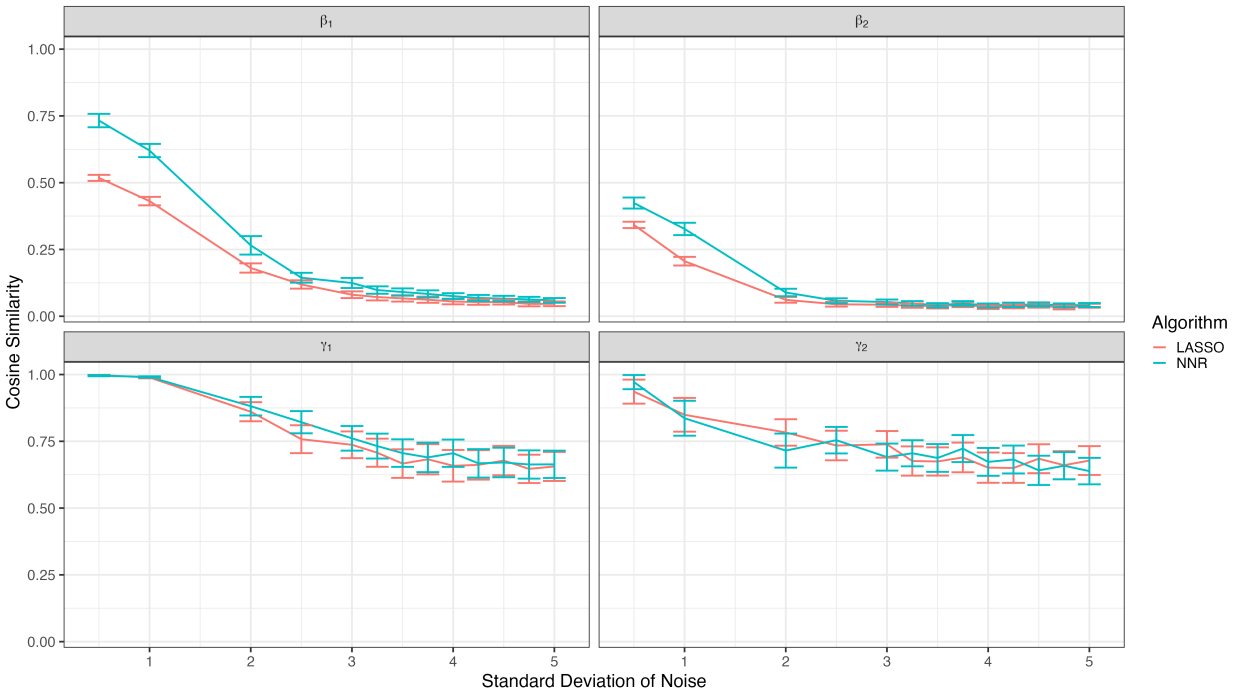

Figure 4.2: Cosine similarity for matrix-vector setting with diagonal W_y.

4.3.3 Matrix-Matrix CCA

In this setting, both $\mathcal{X} \in \mathbb{R}^{20\times 20}$ and $\mathcal{Y} \in \mathbb{R}^{20\times 20}$ are random matrices. We keep $K = 2$, and construct $W_{\mathcal{X}}$ the same as we did in the previous simulation, i.e., the first signal is a block at the top left of the matrix and the second signal is a block at the bottom right of the matrix. We construct $W_{\mathcal{Y}}$, which has 2 columns, similarly, with with $\text{mat}\left((W_{\mathcal{Y}})_1\right)_{[1:5,15:20]} = 1$ and 0's elsewhere, while we set the second column as $\text{mat}\left((W_{\mathcal{Y}})_2\right)_{[15:20,1:5]} = 0.5$ and 0's elsewhere. In other words, $W_{\mathcal{Y}}$ has its first signal in the top right of the matrix and its second signal in a block at the bottom left. Figure 4.4 shows the cosine similarities for the estimates of $\beta_1, \beta_2, \gamma_1$, and γ_2. We expected this setting to be the most challenging, since both $\mathcal{X}$ and $\mathcal{Y}$ have effectively 400 coordinates each, but to our surprise nuclear norm regularized CCA actually performs *best* in this setting, with cosine similarity approaching 1 when noise is

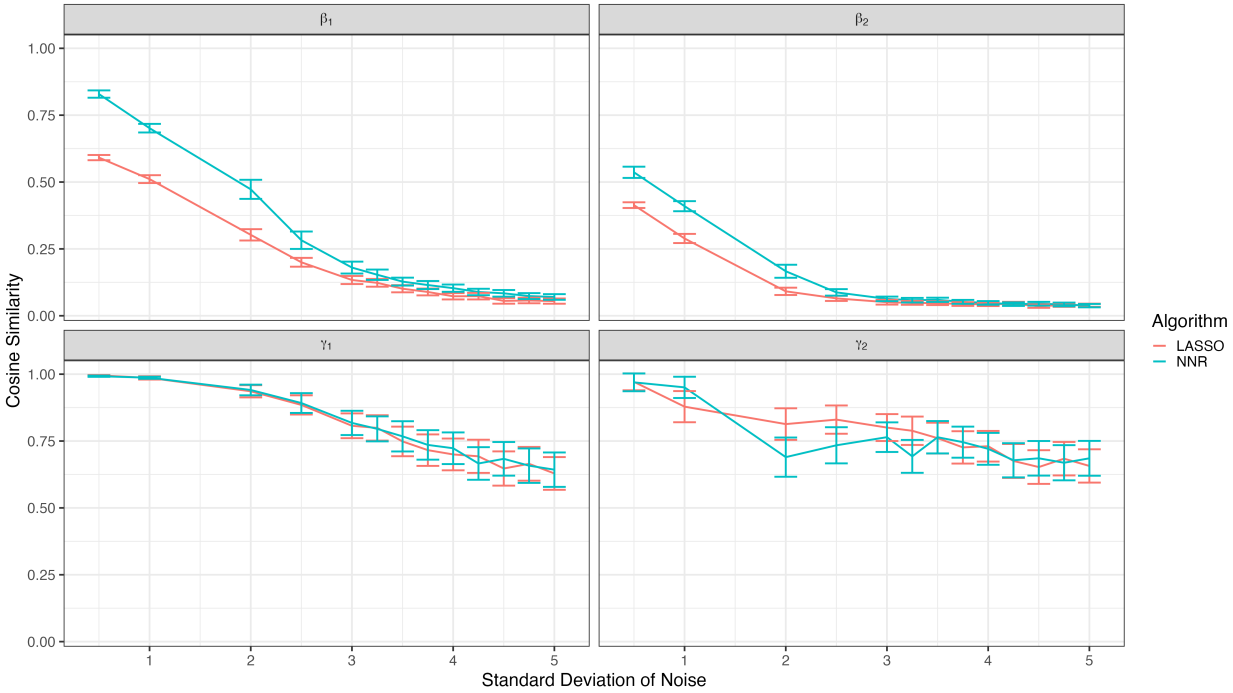

Figure 4.3: Cosine similarity for matrix-vector setting with mixing W_y.

low for both the first and second canonical directions. The signal is still reasonably sparse, and so when the noise is weak LASSO regularized CCA still has decent performance, but nuclear norm regularized CCA appears to markedly out-perform it, especially for the second canonical directions β_2 and γ_2, where LASSO regularized CCA's performance tapers off quickly with growing noise.

4.4 Application to Neuroimaging Data

We apply matcca to a subset of data taken from the ABCD study (Casey et al., 2018) that was graciously processed by our collaborator, Dr. Chandra Sripada, and his research group. This data is closely related to that discussed in Section 3.4, although without the data reduction step. For clarity of presentation, we briefly recap the salient details of this data, highlighting the distinctions in its use here relative to that in Section 3.4. We start with a dataset comprising 5937 participants with complete data, i.e., (i) good quality resting state fMRI data, (ii) behavioral scores for 11 tasks, and (iii) nuisance covariates. From the resting state fMRI data, functional connectivity matrices were obtained, which reflect the correlation over time of various regions of interest (ROIs) in the brain, where these regions are defined according to the parcellation of Gordon et al. (2016). There are 418 ROIs in this parcellation,

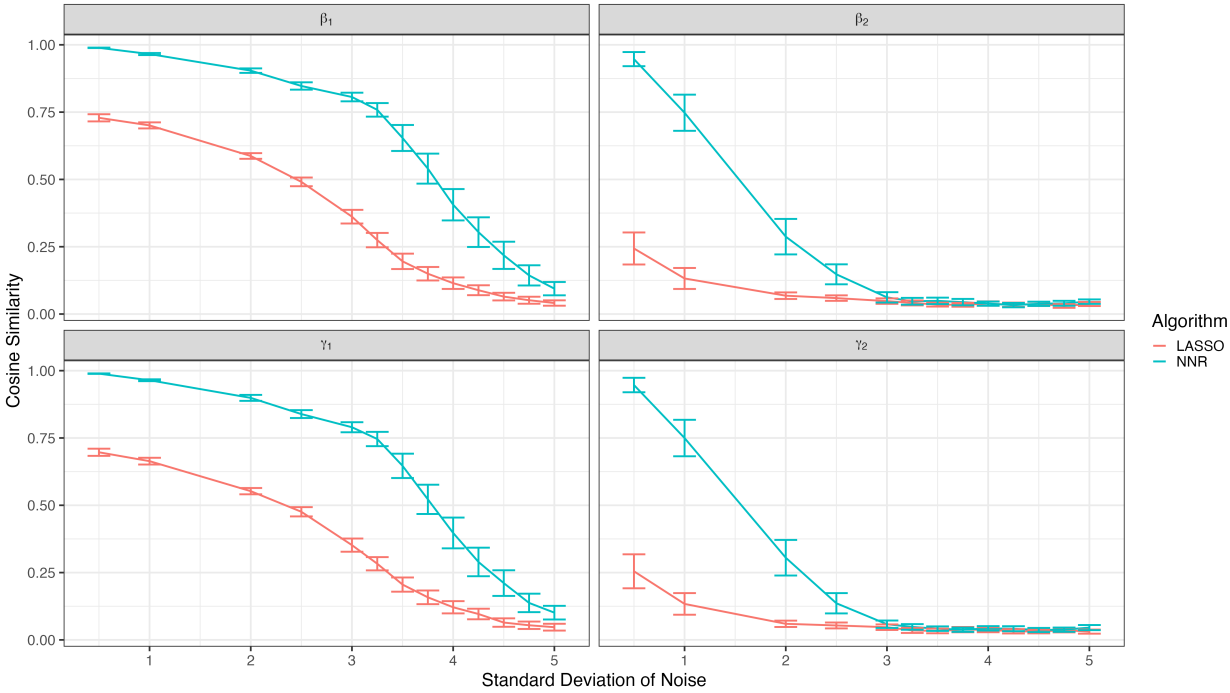

Figure 4.4: Cosine similarity for matrix-matrix setting.

in which case each $\mathcal{X}^{(i)} \in \mathbb{R}^{418\times 418}$. Unfortunately, our attempts to apply matcca to this data were unsuccessful due to the size of the matrices involved (but see discussion in Section 4.5 regarding ongoing work to address this). To overcome this, we instead restrict our attention to a smaller sub-block of this matrix. The parcellation of Gordon et al. (2016) assigns each node to one of several distinct "brain systems," and we extract the submatrix correspond to connections among ROIs labeled as belonging to the FrontoParietal system. There are 24 nodes in this system, so each $\mathcal{X}^{(i)} \in \mathbb{R}^{24\times 24}$, which is more in line with the size of the data we considered in our simulation studies. This is still computationally demanding and parameter tuning required an overnight run on a powerful server.

For y, we take participant performance on 11 behavioral tasks from the ABCD study's neurocognition assessment (Luciana et al., 2018). This includes seven tasks from the NIH Toolbox (Hodes et al., 2013): (i) Picture Vocabulary (Vocabulary), (ii) Oral Reading Recognition (Reading), (iii) Pattern Comparison Processing Speed (Processing Speed), (iv) List Sorting Working Memory (Working Memory), (v) Picture Sequence Memory (Episodic Memory). (vi) Flanker Inhibitory Control & Attention (Flanker), and (vii) Dimensional Change Card Sort (Card Sort); the two conditions of the Rey Auditory Verbal Learning Test, (viii) Short Delay (Memory: Short Delay) and (ix) Long Delay (Memory: Long Delay); and performance on the (x) Matrix Reasoning Task, and (xi) Little Man Task (Spatial Rotation).

In order to avoid nuisance covariates from driving shared variation in $\mathcal{X}$ and y, we use regression to remove unwanted variation. In order to avoid leakage, we partition our data into two sets, fit nuisance regression models on the first set, and then remove these effects from the second set. The nuisance covariates we consider are (i) participant age, (ii) the square of age, (iii) participant sex, (iv) meanFD, (v) the square of meanFD, (vi) race/ethnicity, where meanFD is a summary measure of how much the participant moved their head during the resting state scanning session. We then use this second set of nuisance-corrected data for all subsequent analytic tasks.

We further split this second set of data into a training and a test set, each of which now comprise roughly one fourth of our original sample. We will use this training data to learn canonical directions and then, at the very end, evaluate correlation in the held-out data when using the learned directions. Before we can apply our penalized CCA methods with both LASSO regularization and nuclear norm regularization, we need to have some notion of a useful range of penalization tuning parameters from which to select using cross validation. To arrive at an initial guess, we first fit conventional (unpenalized) CCA to the training data. We project y onto its first estimated canonical direction $\hat{\gamma}_1$ and use this an approximation of the first canonical variate. We then use `glmnet` to fit a LASSO penalized regression that predicts this canonical variate using the functional connectivity data. In the process, `glmnet` constructs a grid of sensible λ values. We retain the smallest and largest values and linearly interpolate between them to obtain a grid of 10 candidate values that we will use when tuning matcca as well as LASSO penalized λ. Using this grid, we then use cross-validation separately with both the LASSO and nuclear norm penalized forms of CCA to select the optimal value of λ based on 5-fold cross-validation. For both models, we initialize the directions using the SVD of the empirical cross-covariance, and we recover two canonical pairs. Because each $y^{(i)} \in \mathbb{R}^{11}$ is already low-dimensional, we apply no regularization to the canonical directions $\hat{\gamma}_1$ or $\hat{\gamma}_2$ associated with y. We then fit both LASSO and nuclear norm penalized CCA to the training set, where each method uses the value of the tuning parameter selected by cross-validation, again using SVD-based initialization and recovering two canonical pairs.

We depict the canonical directions associated with y in Figure 4.5; we can observe that both LASSO and nuclear norm penalized CCA have recovered very similar (although not numerically identical) directions associated with the phenotypes. Moreover, even though we did not encourage sparsity, some of the coefficients in the first direction are very small, whereas they have large values in the second direction (this is especially true of the Episodic Memory task). Interestingly, these canonical directions bear some resemblance to those recovered in Section 3.4, although there are clear differences. While these two analyses rely

on data from the same source, this is still notable as the present analysis uses a relatively small slice of the correlation matrix. We visualize the two canonical directions associated with $\mathcal{X}$ in Figures 4.6. The nuclear norm regularized CCA has recovered canonical directions that are symmetric, although as a consequence of its regularization it has put non-zero mass on the diagonal (which carries no predictive signal since it has a constant value in the data). The directions recovered by the LASSO, however, are not symmetric: this is likely a consequence of instability in the selection path given the very poor conditioning induced by having the same variables present twice (due to the symmetry of the connectivity submatrix). We assess the correlation of the canonical directions both in the training data and on the held-out data. These correlations are given in Table 4.1. While out-of-sample performance is modest, the correlations are nonetheless significantly different from 0 with $p < .05$ when evaluated with a t-test as in `cor.test`. Note that because the correlations are being considered in held-out data, this inferential approach is reasonable, whereas testing the in-sample performance in this manner would not be appropriate.

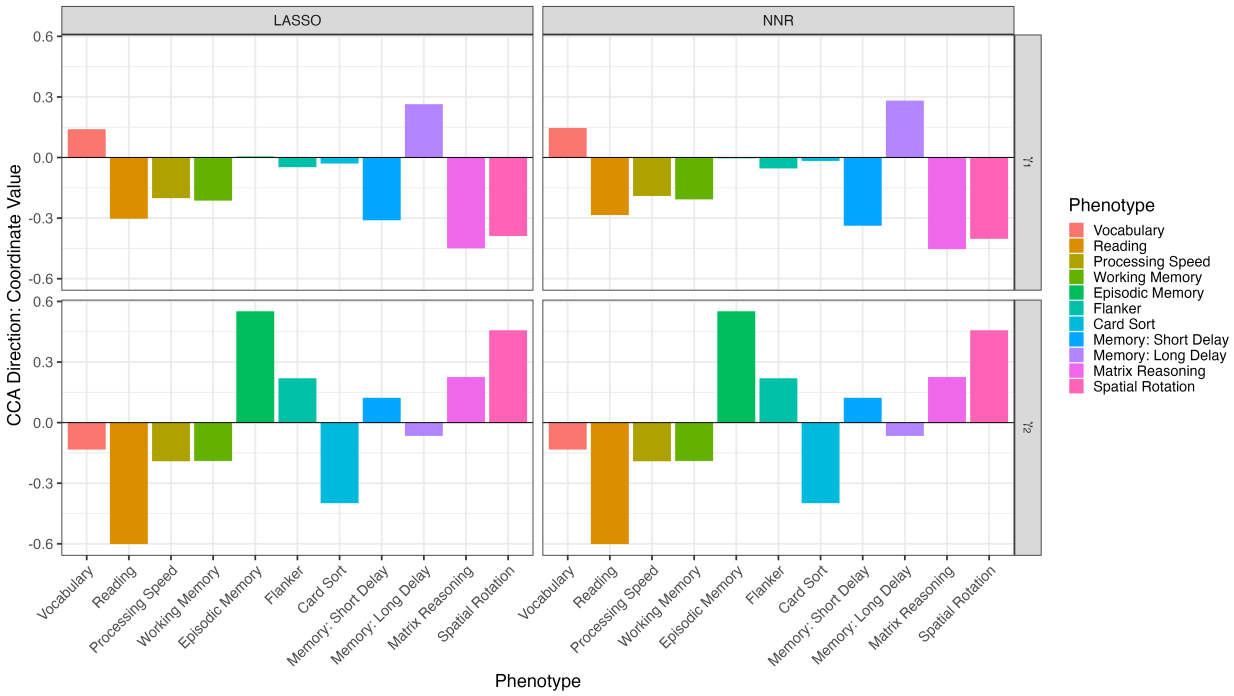

Figure 4.5: First and second estimated canonical directions associated with y in ABCD data for LASSO and nuclear norm regularized CCA.

We may observe that the general pattern of results for both LASSO and nuclear norm regularized CCA are quite similar. This appears to be a consequence of the cross-validation procedure choosing a very small value of λ for both methods. In Figures 4.7 and 4.8, we plot the singular values associated with the directions for $\mathcal{X}$. We can observe that the

LASSO	In-Sample	
	β_1	β_2
γ_1	0.447	0.017
γ_2	0.033	0.444

LASSO	Out-Of-Sample	
	β_1	β_2
γ_1	0.079	0.002
γ_2	-0.019	0.067

Nuclear Norm	In-Sample	
	β_1	β_2
γ_1	0.445	0.017
γ_2	0.0029	0.444

Nuclear Norm	Out-Of-Sample	
	β_1	β_2
γ_1	0.078	0.003
γ_2	-0.020	0.066

Table 4.1: Correlation of canonical variates in training and test data using LASSO and nuclear norm regularized CCA.

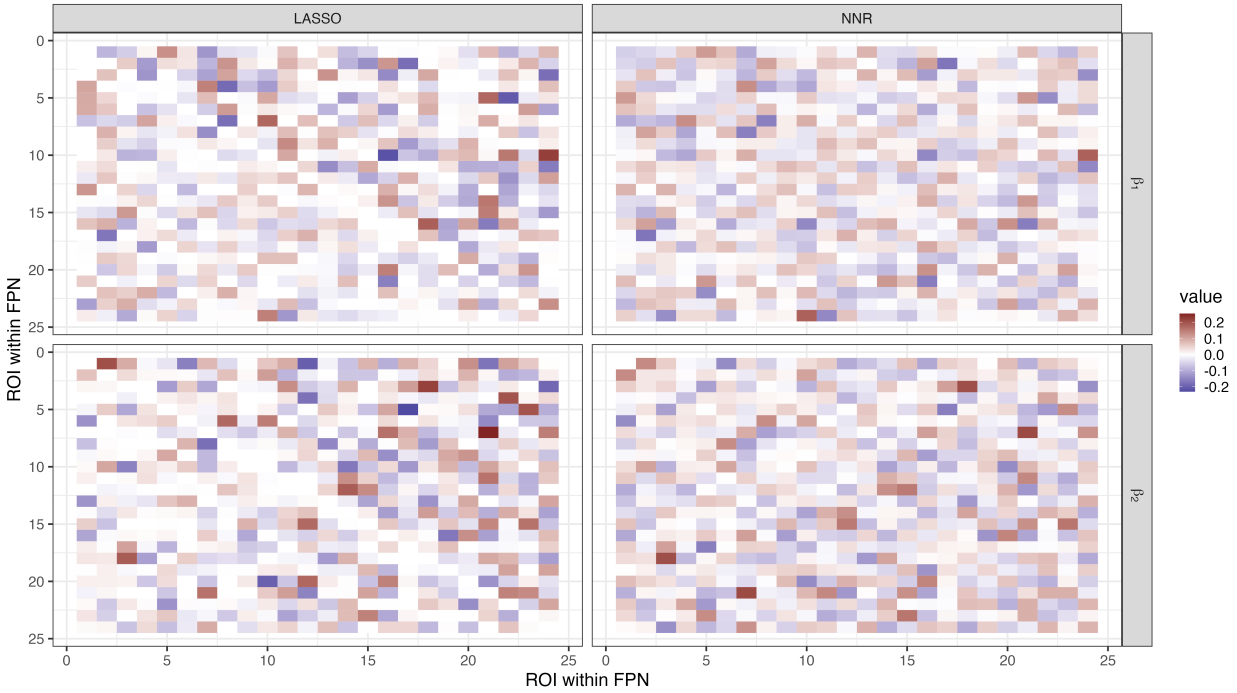

Figure 4.6: First and second estimated canonical directions associated with $\mathcal{X}$ in ABCD data for LASSO and nuclear norm regularized CCA.

nuclear norm penalty only truncates a few of the trailing singular values, which results in a matrix that is still approximately full rank. In this setting, the dimension of our problem, even when vectorized, is appreciably smaller than the number of observations, and so it is not terribly surprising that the optimal approach is to employ very little regularization and instead provide estimates that are quite close to what we would obtain with no penalization. Since the N is large in this sample, it underscores the importance of further optimizing our algorithm so that it can be run on the entire connectivity matrix, in which case regularization will be vital as we will be in a high-dimensional setting.

4.5 Discussion

In this work we introduced a matrix-variate formulation of CCA. Our method, matcca, exploits matrix-variate data by seeking low-rank structure in the canonical directions (when they are considered as matrices). Using simulation studies, we demonstrated that this approach can effectively recover shared low-rank signal in a variety of settings even when the number of observations is relatively low and the noise is non-trivial. Of particular note, we demonstrated that our method out-performed LASSO penalized CCA, even though in our simulation settings the shared signal is low rank *and* sparse. However, this is not the

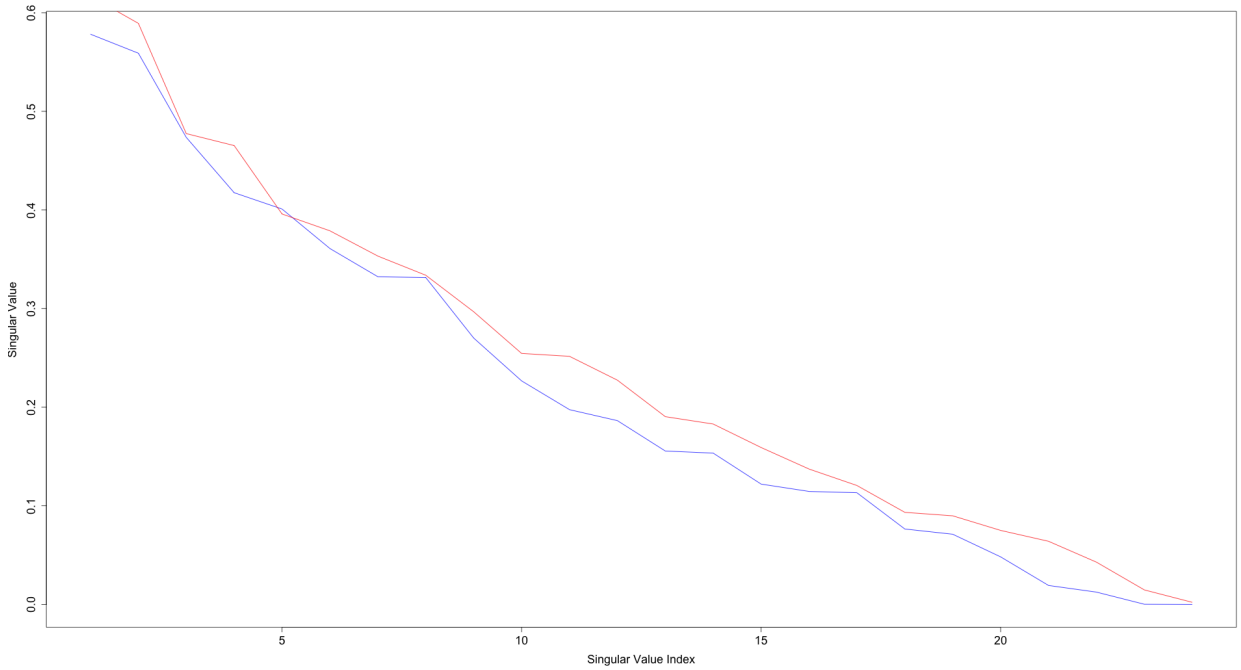

Figure 4.7: Singular values of first estimated canonical direction associated with $\mathcal{X}$ in ABCD Data for LASSO and nuclear norm regularized CCA.

only kind of structure that we might seek with matrix-variate data. In the neuroimaging setting where participant-specific networks are extracted, previous work has sought different kinds of structure, e.g., row/column sparse in Relión et al. (2019), community-block sparse in Kessler et al. (2022), low rank and sparse in Brzyski et al. (2020). Our framework can be readily extended to incorporate penalties corresponding to this structure, but successful implementation will depend heavily on the efficiency with which penalized regression can be fit. Since optimization problems involving the penalty must be solved many, many times (possibly hundreds of thousands of times), it is vital that these routines are fast, and so approximate solutions or further relaxations may be necessary.

Our matrix-variate CCA formulation is really a special case of an even more general formulation of CCA that can accommodate tensors and other increasingly complex objects. Formally, let X, Y be random *objects* in two possibly distinct finite-dimensional inner product spaces V and W, with inner products denoted by $\langle \cdot, \cdot \rangle_V$ and $\langle \cdot, \cdot \rangle_W$. We can then write the CCA objective for finding the first canonical pair as

$$\begin{aligned}(\beta_1, \gamma_1) = \underset{b,g}{\operatorname{argmax}} \operatorname{Corr}\left(\langle X, b\rangle_V, \langle Y, g\rangle_W\right) \\ \text{s.t. } \operatorname{Var}\langle X, b\rangle_V = \operatorname{Var}\langle Y, g\rangle_W = 1.\end{aligned}$$

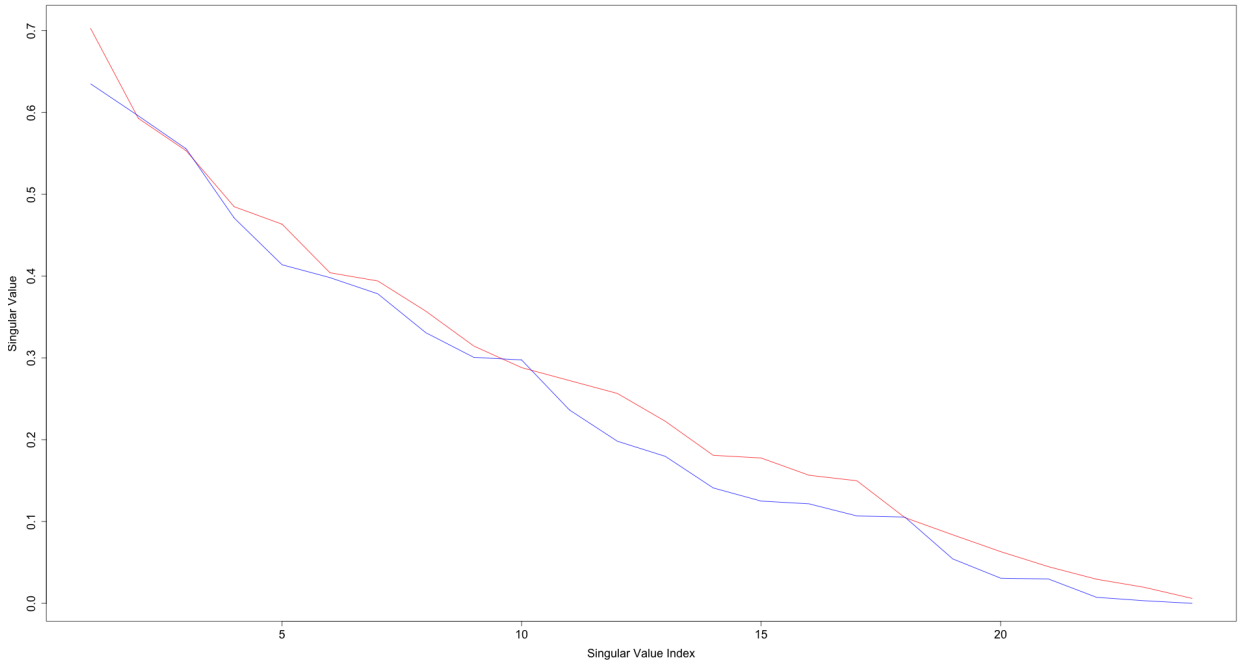

Figure 4.8: Singular values of second estimated canonical direction associated with $\mathcal{X}$ in ABCD data for LASSO and nuclear norm regularized CCA.

Subsequent canonical pairs can be found by maximizing the same objective subject to an induced orthogonality requirement, i.e.,

$$\beta_2 : \text{Corr}\left(\langle X, b_1\rangle_V, \langle X, b_2\rangle_V\right) = 0$$
$$\gamma_2 : \text{Corr}\left(\langle Y, g_1\rangle_W, \langle Y, g_2\rangle_W\right) = 0,$$

and so on for later pairs. When $V = \mathbb{R}^p$ and $W = \mathbb{R}^q$ with the usual Euclidean inner product, then we recover classical CCA. When $V = \mathbb{R}^{p_1 \times p_2}$ and $W = \mathbb{R}^{q_1 \times q_2}$ with the Frobenius inner product, then we obtain matrix-variate CCA. This formulation, coupled with Mai and Zhang (2019)'s reformulation of CCA as a constrained quadratic optimization problem with optional penalties, provides a powerful and flexible framework for developing CCA for complex types of data wherein penalties tailored to the structure and complexity of the data can be exploited. While at first glance this may appear to simply be a proposal for the expanded use of kernel CCA (Akaho, 2007), this framework is distinct in that it aims to make the directions *explicit* and to enable them to be penalized.

Future work in this area includes improvements to the efficiency of our implementation with a particular eye to the nuclear norm penalized regression approach. As currently implemented, its complexity precludes its application to matrices with a large number of rows

or columns. Improvements to initialization may also help to reduce the number of iterations necessary to converge. As mentioned, we initialize our canonical directions based on the SVD of the empirical cross-covariance, but this is a decomposition that treats the data as vectors. While this is a fair choice when comparing matcca to LASSO regularized CCA, an initialization strategy that takes into account the structure of the data may provide a better starting point. One particular option is to initialize using a variant of SVD architected for collections of images, such as the Population Value Decomposition method of Crainiceanu et al. (2011).

One shortcoming of our approach to penalized CCA is that the resultant canonical variates are no longer uncorrelated (or equivalently, the canonical directions are no longer orthogonal with respect to the inner product induced by the covariances). This challenge is not specific to matcca but affects LASSO regularized CCA when implemented using the approach of Mai and Zhang (2019). This can be addressed by the introduction of an additional tuning parameter which will penalize non-orthogonality, allowing the analyst to trade-off (i) maximizing correlation, (ii) satisfying the penalty, and (iii) maintaining orthogonality of the canonical variates.

CHAPTER 5

Conclusion and Future Directions

Motivated by the opportunity to leverage structure in order to learn about the brain, this dissertation has presented three distinct projects with different statistical goals. First, in Chapter 2, we used neuroscientifically-plausible structure to guide learning in a prediction task. We proposed NetCov as a method to predict a label or score using network edge weights and node covariates, and showed that it enjoyed attractive support recovery properties relative to competitors in simulations and that in an application, it was competitive with other approaches while offering superior interpretability. Then, in Chapter 3, we focused on inference for discovered structure where we introduced combootcca: a resampling-based approach for inference on the canonical directions in CCA. In simulation studies, we showed that combootcca's unique alignment strategy coupled with percentile-based bootstrapped confidence intervals offers the best statistical properties of all methods considered. Finally, in Chapter 4, we pursued estimation and developed matcca as a matrix-variate extension of CCA in order to exploit low-rank structure. In simulation studies, matcca was very effective at recovering low-rank signals in noisy settings with relatively few observations, and moreover it out-performed the LASSO even though the signal was sparse.

These projects reflect just a few of the opportunities to exploit structure in statistical learning. Below, we outline several future directions inspired by this work.

Selective-Inference for Structural Learning One of the challenges with exploiting structure in statistical learning is how to retain statistical validity in downstream tasks. Although it does not appear in this dissertation, we were inspired by the overlapping group LASSO problem at the heart of NetCov in Chapter 2 to tackle post-selective inference for the group LASSO, and we contributed to Panigrahi et al. (2023a). Related selective inference problems emerge from Chapter 3: it is possible to reformulate the regression approach to CCA as inference for a selected target (which may deviate from the "true" canonical direction), and it may be possible to show that in some satisfactorily conditional sense, it exhibits appropriate statistical properties. Another problem is the sequential and nested

nature of the inferential procedure. For example, in Section 3.4, we conducted testing on the canonical correlations, rejected H_0 for the first three, and then performed inference on only these canonical directions. In the extreme case where one screens potentially thousands of canonical directions, finds a subset that seem to exhibit signal, and then performs follow-up analyses, we can imagine that the subsequent inference will not be valid without further adjustment. The low rank structure that we sought in Chapter 4 poses a particularly interesting challenge for selective-inference. The "conditional" approach to selective inference that we deployed in Panigrahi et al. (2023a) hinges on the characterization of a tractable selection event with non-trivial probability. When the "selection event" corresponds to the identification of a subspace, as is the case with low rank approximations, we are confronted with a continuous selection space, and so conditional approaches are not immediately applicable. These are all open problems that we hope to pursue in future work.

Object-Oriented CCA The matcca approach is just one instance of a more general approach to CCA that we outline in Section 4.5. In the spirit of Marron and Dryden (2021), through thoughtful consideration of appropriate inner product spaces, we can conceivably develop CCA extensions for a variety of data types. Although (weighted) networks were the motivation for our development of matrix-variate CCA in the first place, other inner products may be more sensible, especially for more complex networks as may be obtained from event data. While this approach has close ties to kernel CCA (Akaho, 2007), the framework we deployed in Chapter 4 is distinct in that we want to characterize and understand the directions, and in particular penalize them to exploit various types of structure. Matrix-variate CCA with the nuclear norm penalty is then just one instance of a broader class of "Object Oriented" CCA methods which could be tailored for different types of data. Of course, each application will require careful consideration and development of appropriate penalties.

Multi-Scale Inference Another type of structure inherent in many datasets, including neuroimaging, is the resolution. Although not in this thesis, we have contributed to work (Kim et al., 2023) where we are able to test hypotheses about differential brain connectivity at the level of the mean of a network cell and also at the level of individual edges. A natural extension of that work in the spirit of what we have presented here would be to develop an omnibus test that is sensitive not just to homogeneous changes in the mean, but to heterogeneous changes, too. In the predictive setting, this could also take the form of bi-level feature selection, which we discuss in Section 2.6

APPENDIX A

Appendix for Chapter 2

A.1 ROC Curves

There is what might at first seem to be a surprising phenomenon present in our numerical experiments most visible in the "5 active groups" panels of Fig. 2.5: as the magnitude of non-zero entries of β grows, recall climbs steadily toward 1, but precision climbs and then falls. As discussed in the text, this behavior is a consequence of parameter tuning which selects the value of λ that minimizes prediction risk rather than support recovery. To further understand this, we conduct a small simulation study where we explore the behavior of the EBG variant of NetCov and the LASSO in this setting. We obtain three realizations of data where β follows the EBG grouping scheme, there are 5 active groups, and where we set α (the magnitude of active entries) to 0.01, 0.04, and 0.2, respectively; all other simulation parameters were fixed per the regime of Experiment I as described in Section 2.4.1. We fit both the LASSO and the NetCov method with the EBG grouping scheme to each of these realizations along a λ path as described in Section 2.3.3 (note that the values of λ along these paths for NetCov: EBG and LASSO will be distinct). For a fixed value $\tilde{\lambda}$ along this path, we can obtain an estimated active set for both the LASSO and NetCov: EBG. By comparing the estimated active set to the true active set, we compute both the True Positive Rate (True Positives divided by the sum of True and False Positives) and the False Positive Rate (False Positives divided by the sum of True and False Positives). We connect these points to obtain six receiver operating characteristic curves for the three different levels of signal strength times the two different models. This is depicted in Fig. A.1. Fixing α and comparing the two fitting methods, we see that the curve corresponding to NetCov: EBG is generally above or equal to the LASSO curve. The apparently poor precision of the NetCov: EBG method in the high SNR regime (as may be especially evidence with 5 active groups as seen in Fig. 2.5) may seem odd given its apparently "perfect" ROC curve, but cross validation chooses $\hat{\lambda}$ somewhere along the curve where the True Positive Rate is 1, but

where the False Positive Rate is nonzero. In other words, there exists a value of λ that would give perfect support recovery, but cross validation does not choose it. This appears to be a symptom of tuning λ to minimize prediction error, and this leads to a too-small value of $\hat{\lambda}$ in order to avoid bias encountered from shrinking large predictors but this comes at the cost of selecting several inactive predictors.

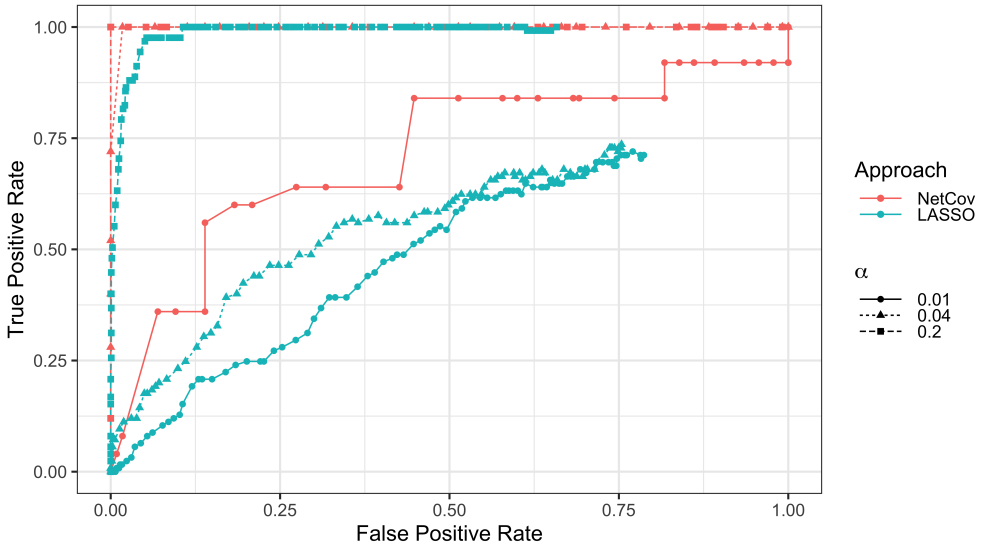

Figure A.1: False positive and true positive rates along the λ path. We depict receiver operating characteristic curves for the LASSO and NetCov: EBG model at varying levels of signal intensity. Data is drawn according to the setting of Experiment I as described in Section 2.4.1 with the EBG grouping scheme and 5 active groups.

A.2 Additional Neuroimaging Results

In the main text, for brevity we presented out-of-sample correlation for a subset of the phenotypes in Fig. 2.11. In Fig. A.2, we present the out-of-sample correlations for all phenotypes.

We present the estimated coefficients for β_A and β_X from both NetCov (EBG grouping) and LASSO associated with PMAT and Working Memory in Figs. A.3 and A.4. In Figs. A.5 and A.6, we depict edges selected by CPM for PMAT and Working Memory.

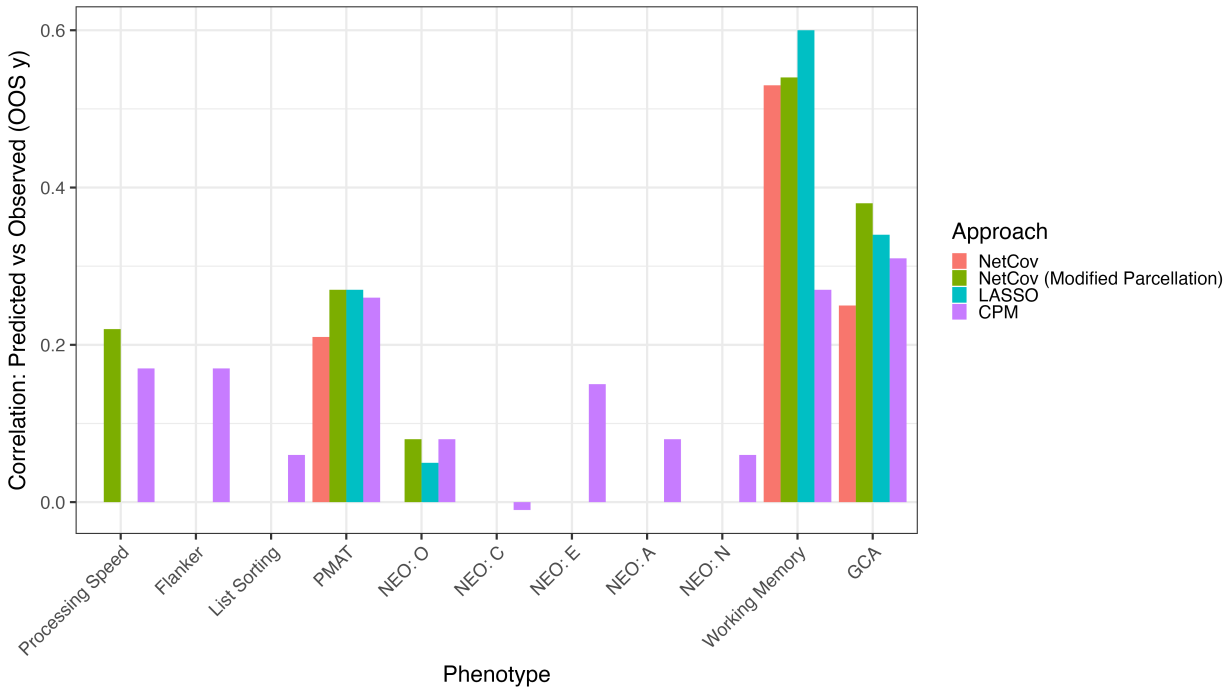

Figure A.2: Out-of-sample correlation for all phenotypes in application to human neuroimaging data.

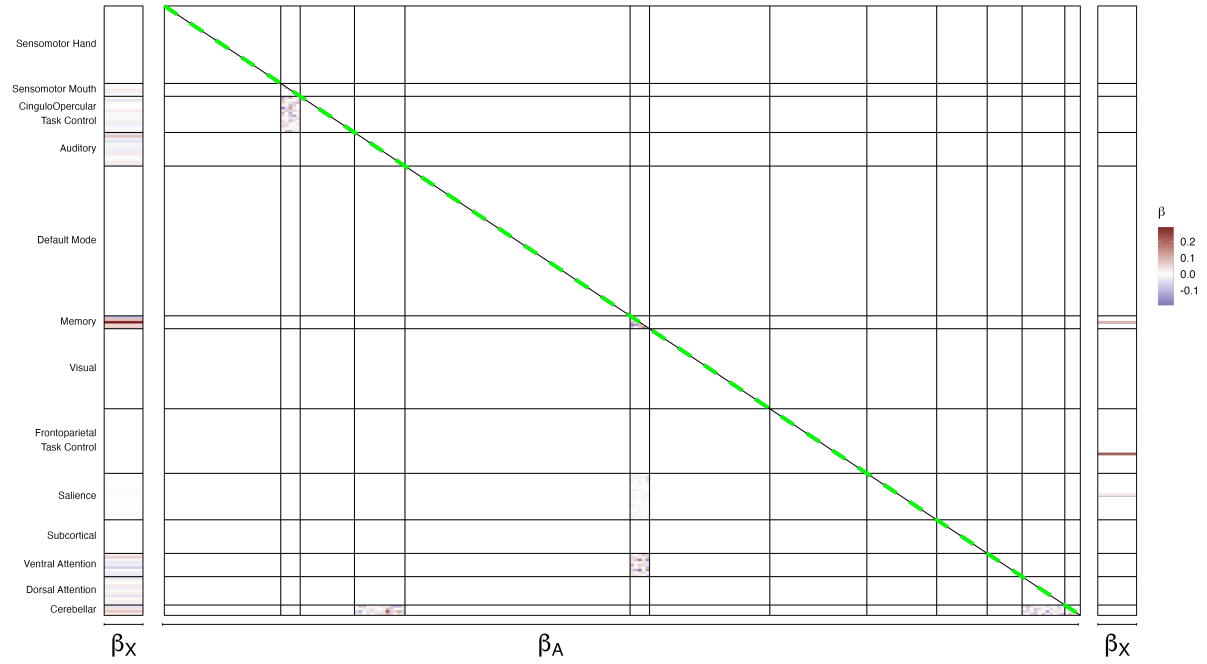

Figure A.3: Visualization of $\boldsymbol{\beta}$ coefficients with PMAT as response. Coefficients from NetCov with EBG are presented at left (β_X) and on the lower triangle (β_A). Coefficients from LASSO are presented at right (β_X) and on the upper triangle (β_A). Solid lines depict boundaries of the Power parcellation.

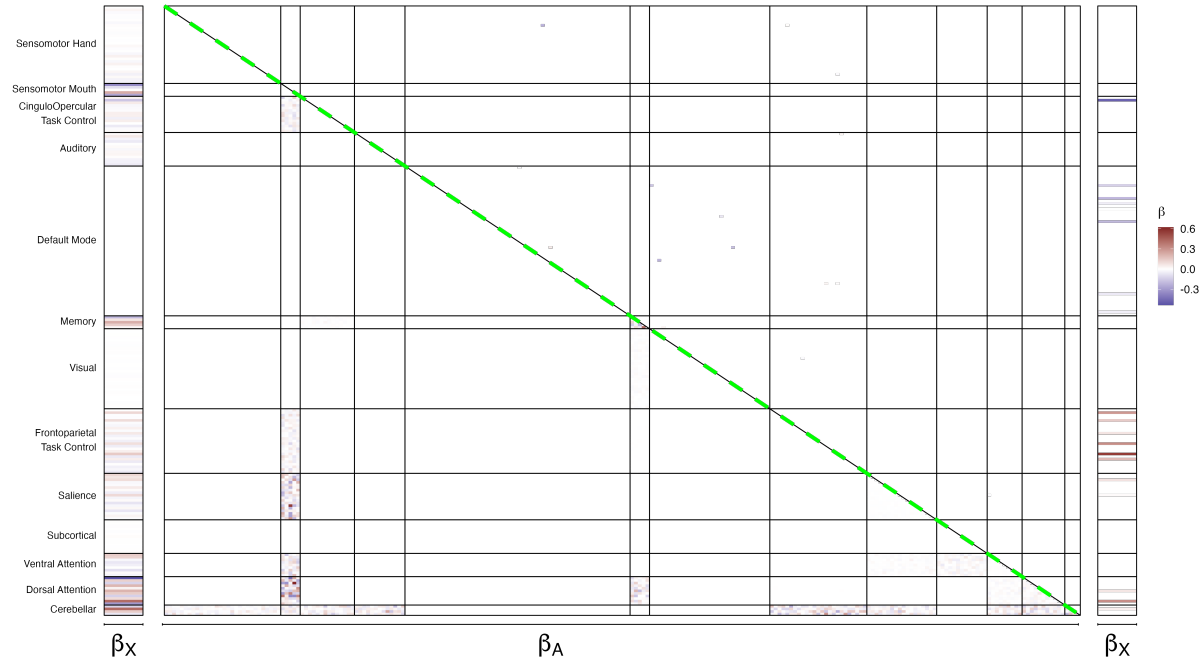

Figure A.4: Visualization of $\boldsymbol{\beta}$ coefficients with Working Memory as response. Coefficients from NetCov with EBG are presented at left (β_X) and on the lower triangle (β_A). Coefficients from LASSO are presented at right (β_X) and on the upper triangle (β_A). Solid lines depict boundaries of the Power parcellation.

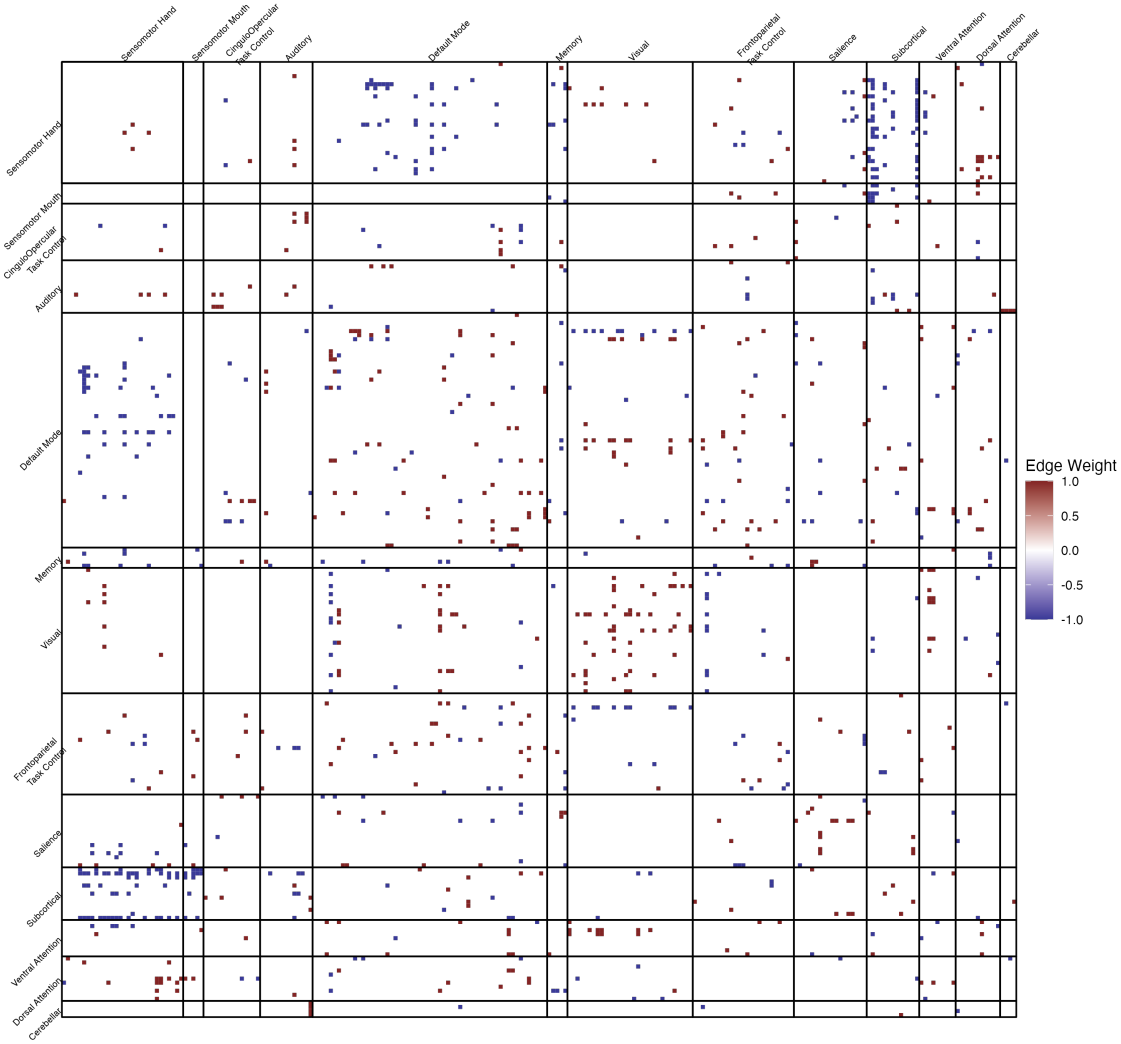

Figure A.5: Edges selected by CPM, colored by the sign of their association with PMAT. Solid lines depict boundaries of the Power parcellation.

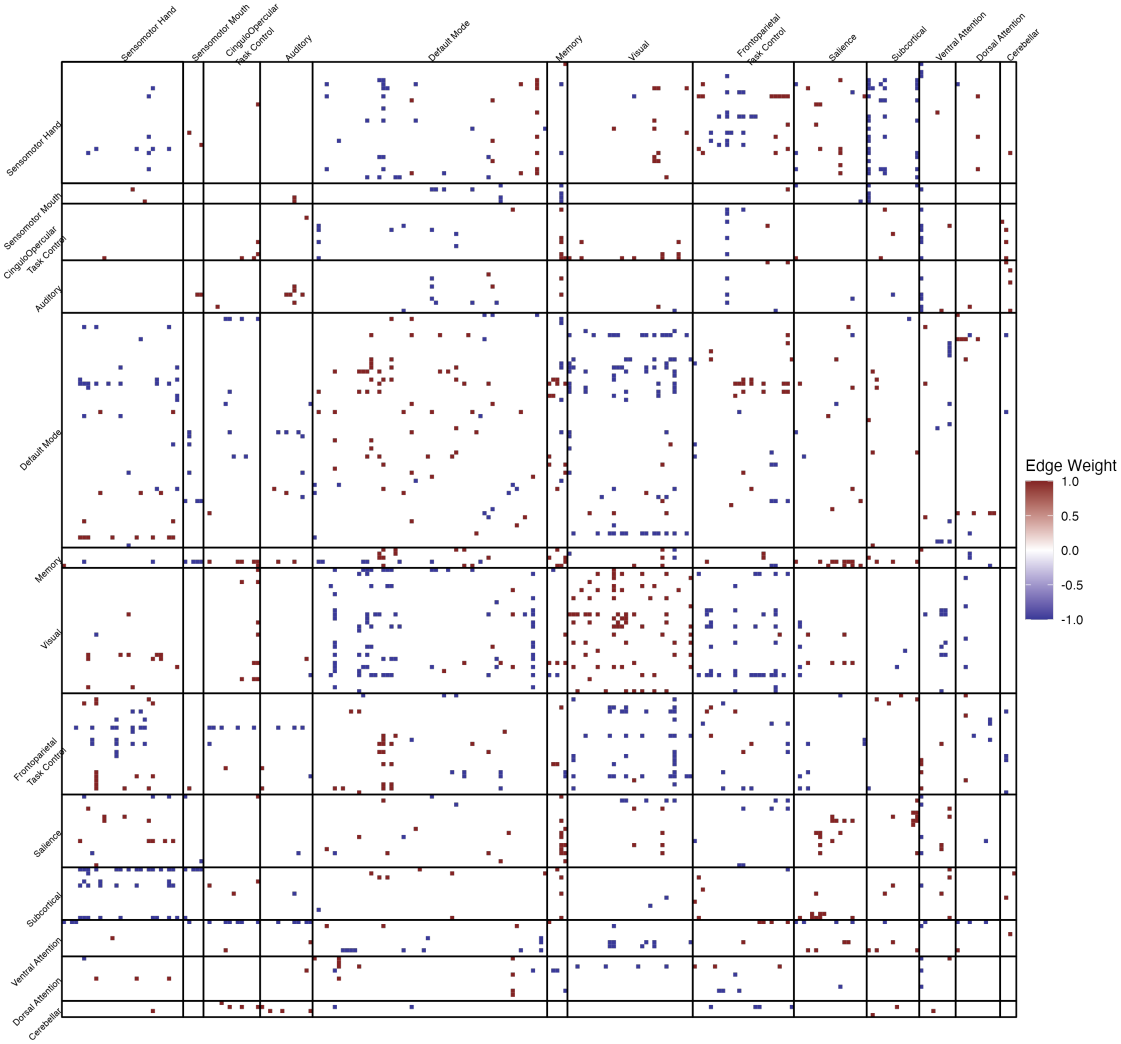

Figure A.6: Edges selected by CPM, colored by the sign of their association with Working Memory. Solid lines depict boundaries of the Power parcellation.

APPENDIX B

Appendix for Chapter 3

B.1 Simulation I with Identity Covariances

We repeat the procedures for Simulations I (described in Section 3.3.2), but with identity covariance matrices for both x and y, i.e., $\Sigma_x = I_p$ and $\Sigma_y = I_q$. Results analogous to those in Section 3.3.2, which constructed and inverted sparse precision matrices to define Σ_x and Σ_y, are given below in Figures B.1 through B.6. In general, the results are similar, although the challenge of covering non-zero coordinates due to conservative bias appears moderately attenuated.

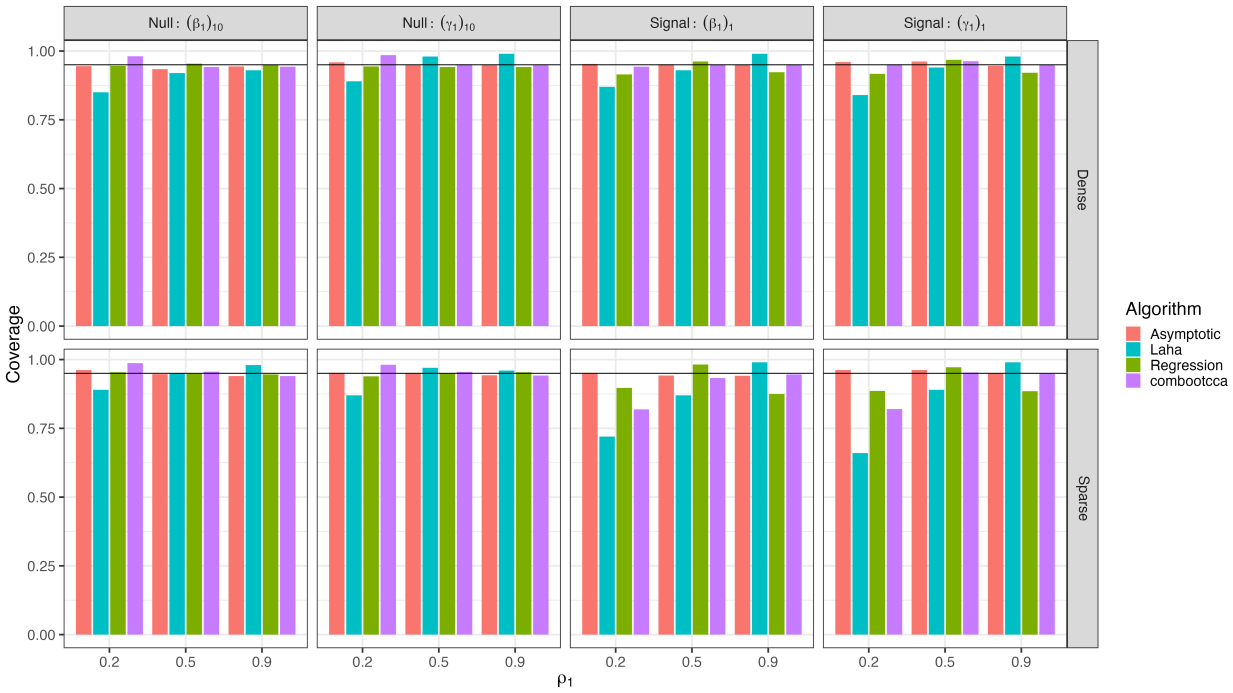

Figure B.1: Coverage rates in simulation I for $p = q = 10$. The horizontal line indicates nominal 95% coverage.

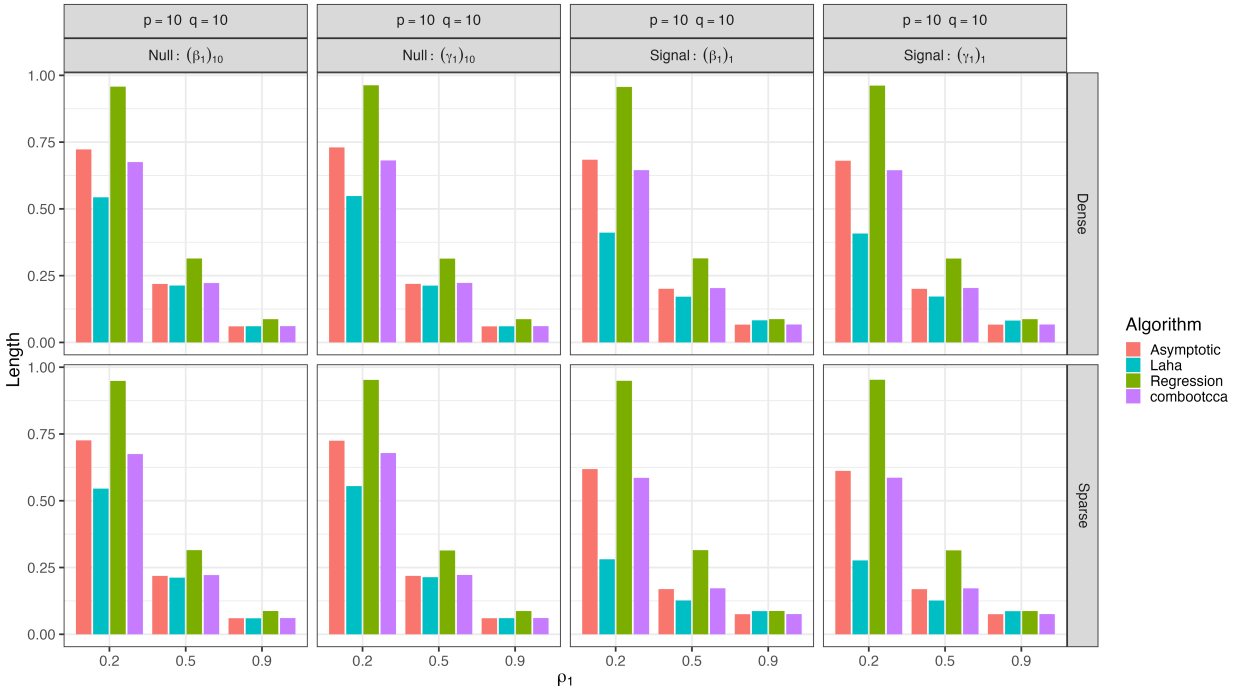

Figure B.2: Lengths of confidence intervals in simulation I for $p = q = 10$.

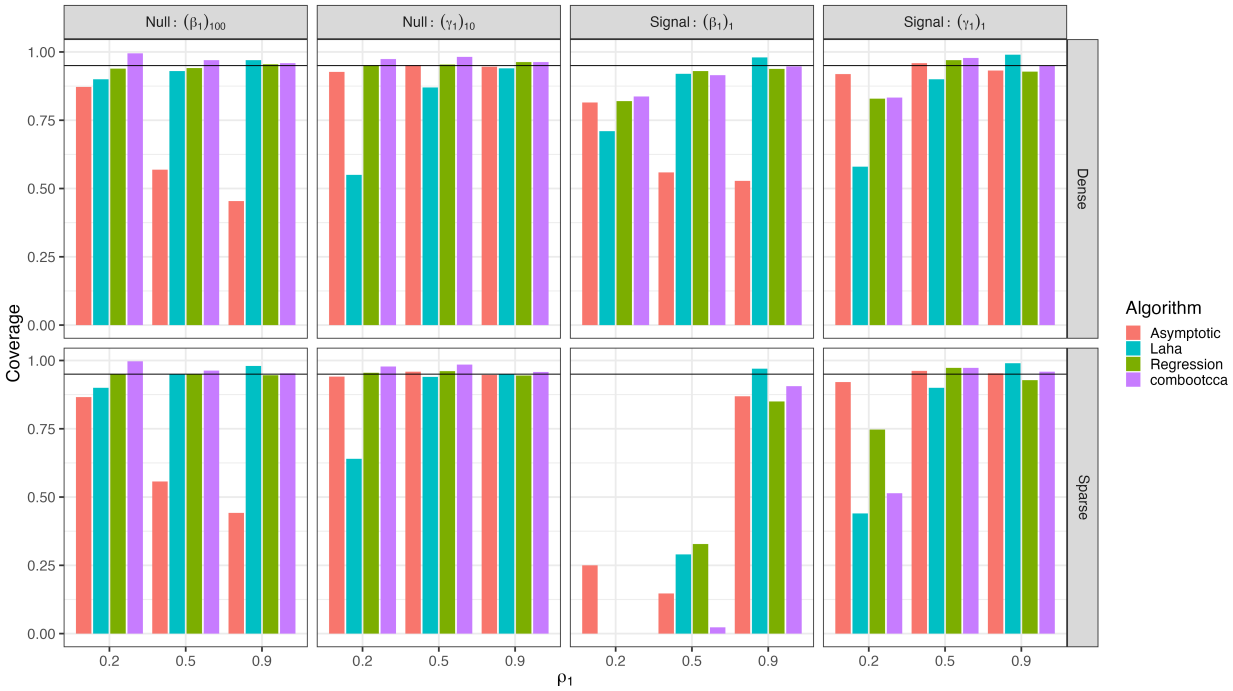

Figure B.3: Coverage rates in simulation I for $p = 100, q = 10$. The horizontal line indicates nominal 95% coverage.

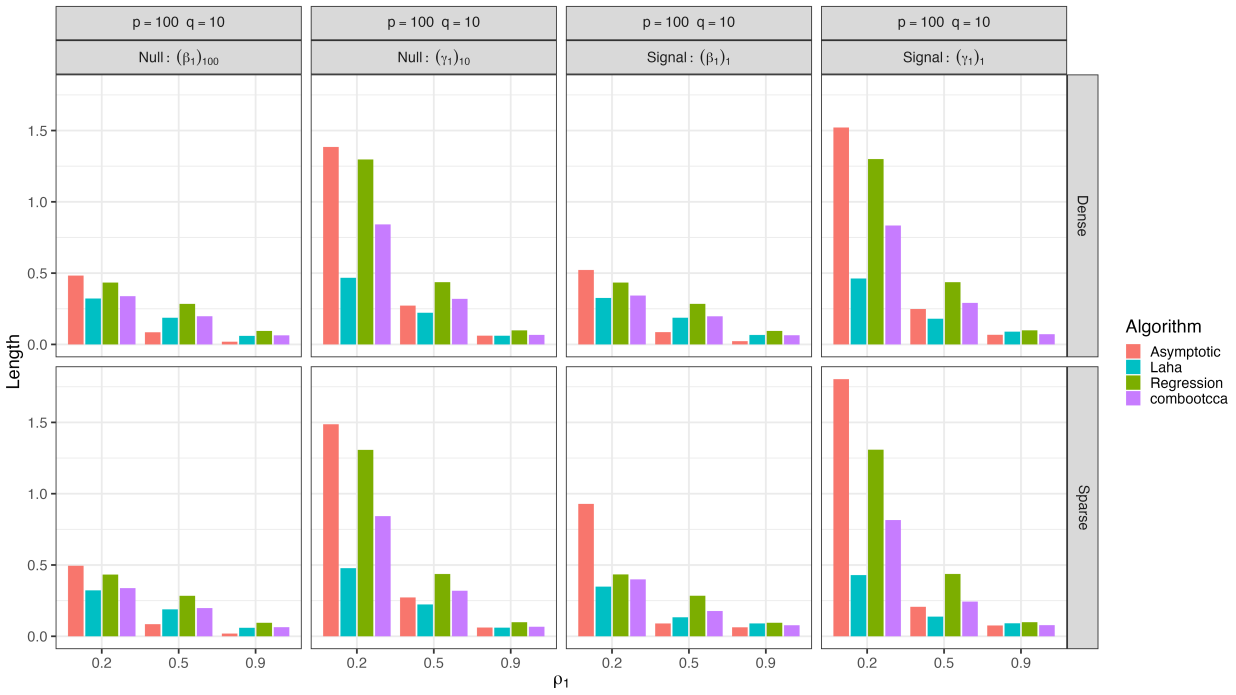

Figure B.4: Lengths of confidence intervals in simulation I for $p = 100, q = 10$.

B.2 Simulation II with Identity Covariances

We repeat the procedures for Simulations I (described in Section 3.3.3), but with identity covariance matrices for both x and y, i.e., $\Sigma_x = I_p$ and $\Sigma_y = I_q$. Results analogous to those in Section 3.3.3, which constructed and inverted sparse precision matrices to define Σ_x and Σ_y, are given below in Figures B.7 through B.18. As was the case when we repeated Simulation I with identity covariance matrices in Section B.1, the pattern of results are quite similar, and again we observe that under-coverage of non-zero coordinates appears less severe.

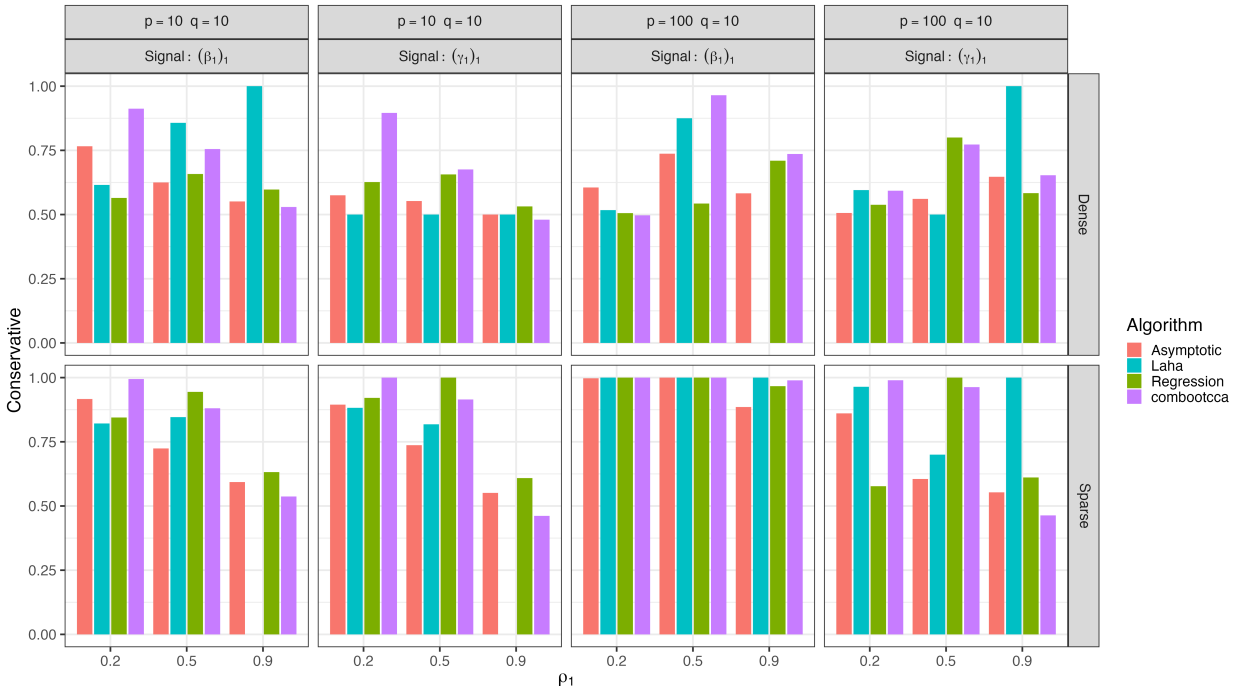

Figure B.5: Bias in simulation I: the proportion of confidence intervals that failed to cover non-null signals that are "conservative" (the true value is greater in magnitude than any value in the confidence interval).

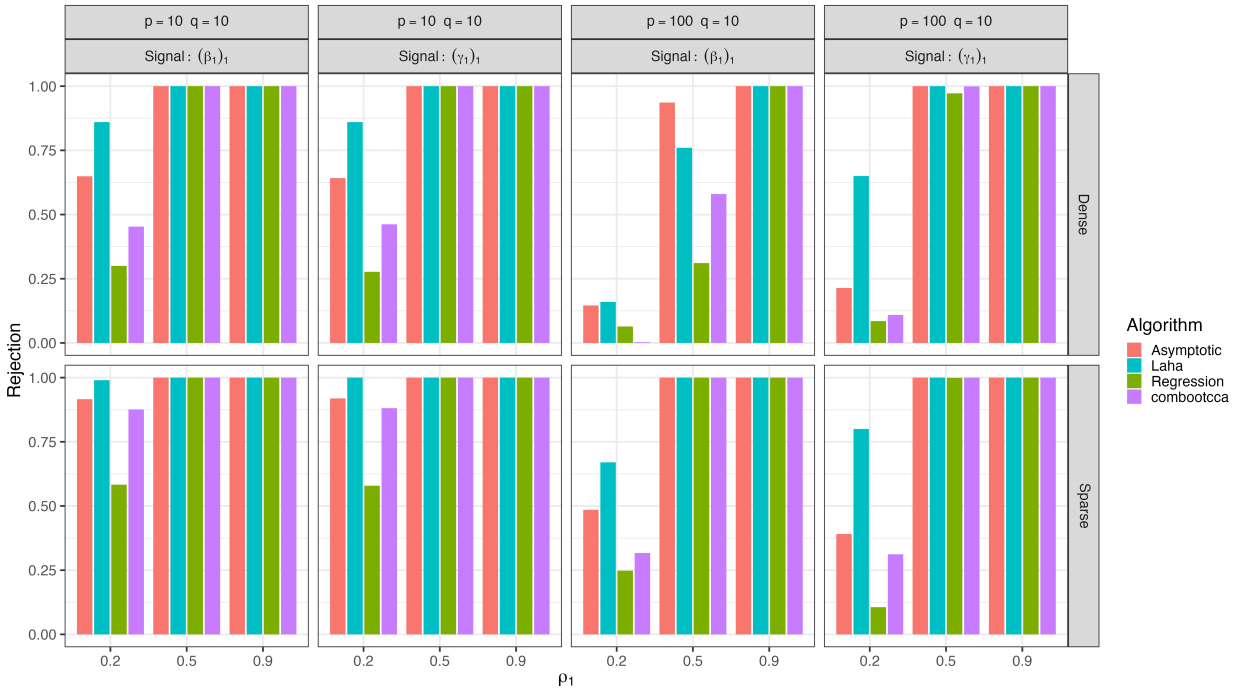

Figure B.6: Power (correct rejection rates) in simulation I.

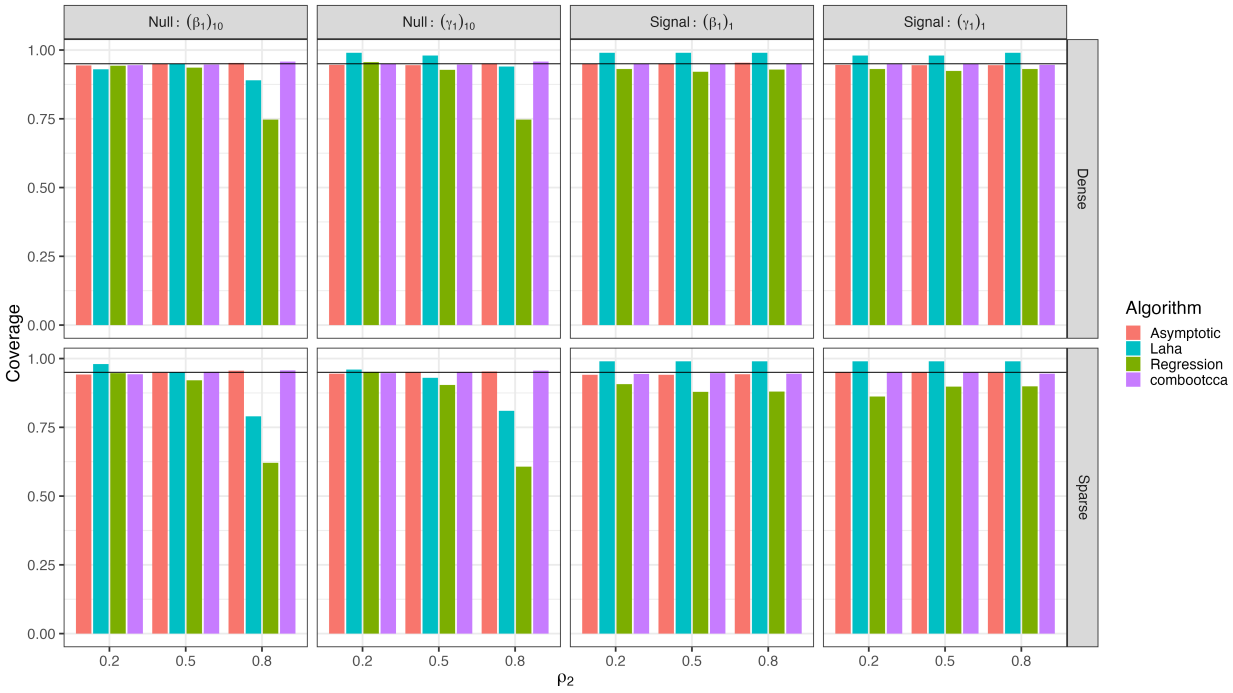

Figure B.7: Coverage rates for first canonical directions in simulation II for $p = q = 10$. The horizontal line indicates nominal 95% coverage.

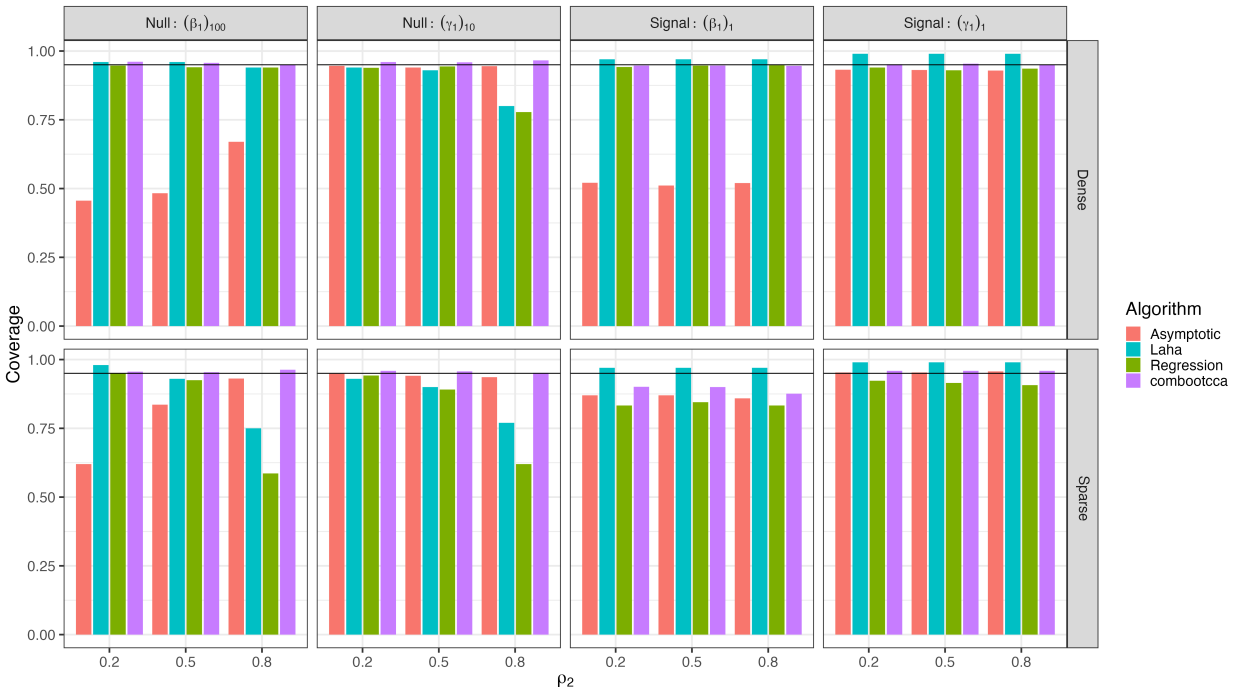

Figure B.8: Coverage rates for first canonical directions in simulation II for $p = 100, q = 10$. The horizontal line indicates nominal 95% coverage.

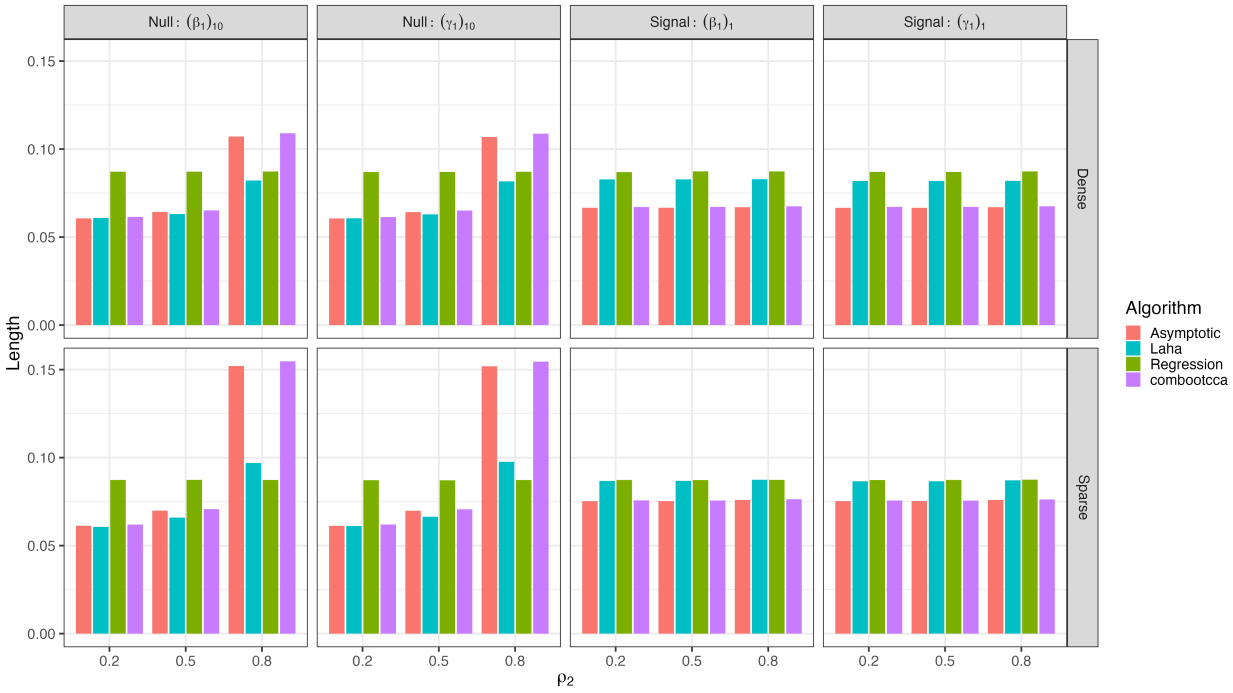

Figure B.9: Lengths of confidence intervals for first canonical directions in simulation II for $p = q = 10$.

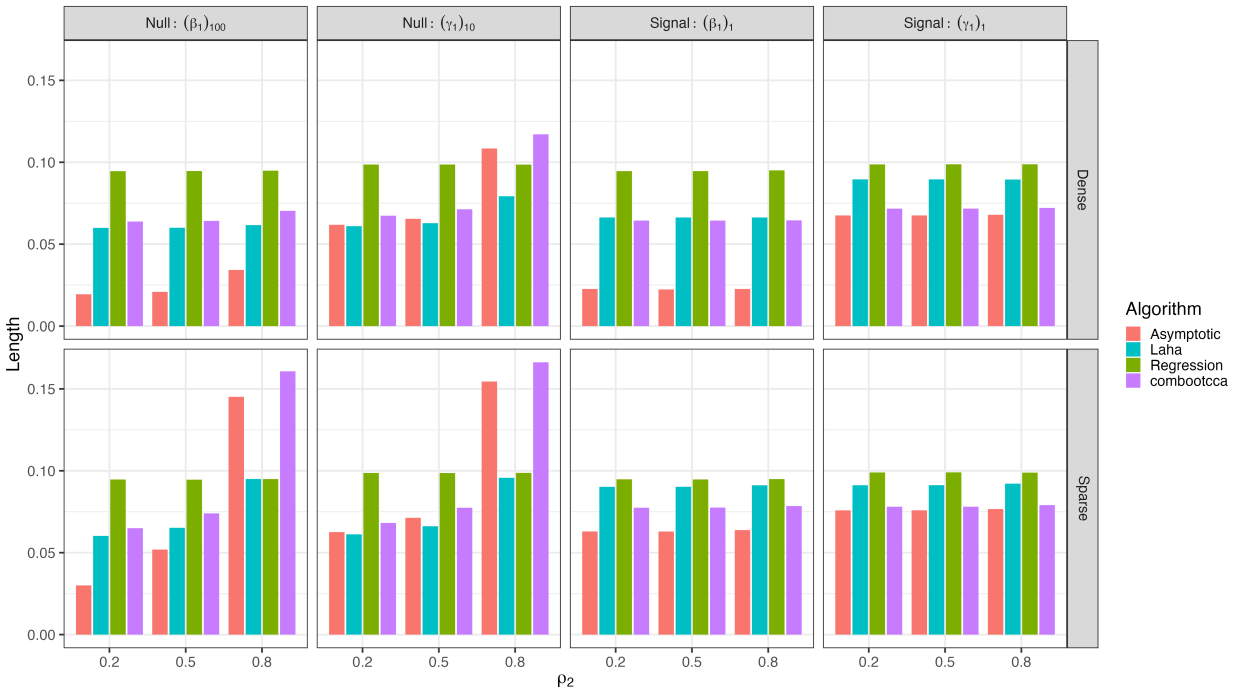

Figure B.10: Lengths of confidence intervals for first canonical directions in simulation II for $p = 100, q = 10$.

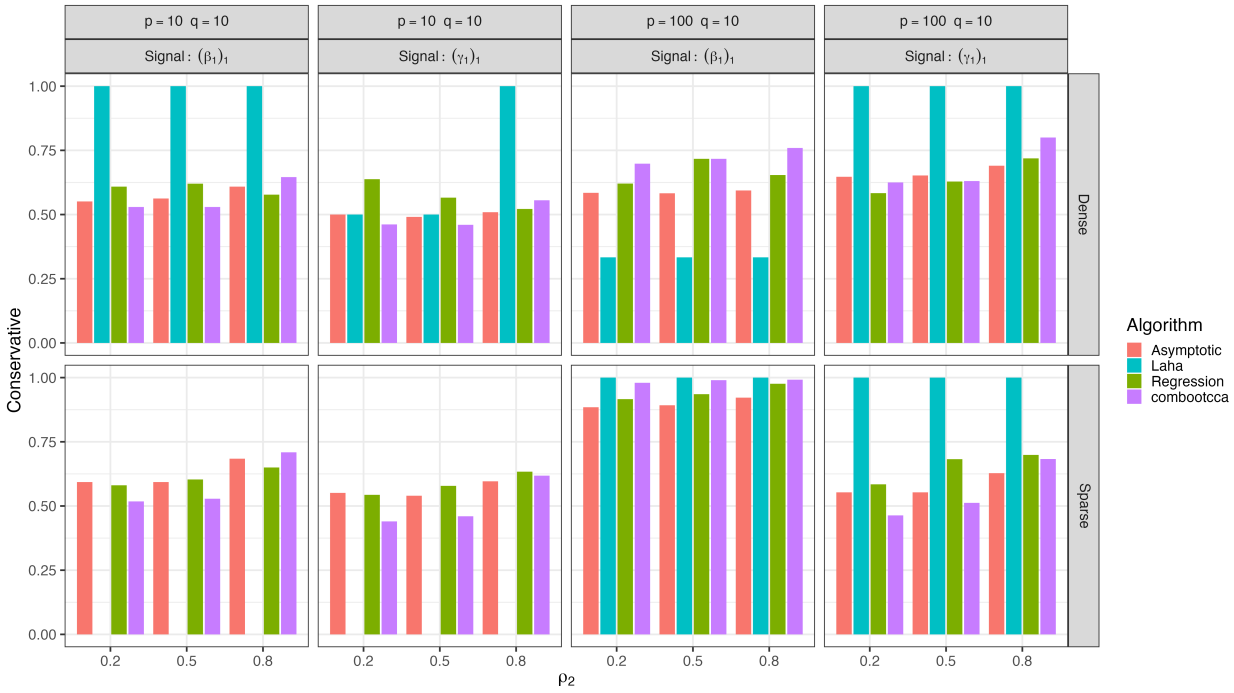

Figure B.11: Bias in simulation II for first canonical directions: the proportion of confidence intervals that failed to cover non-null signals that are "conservative" (the true value is greater in magnitude than any value in the confidence interval).

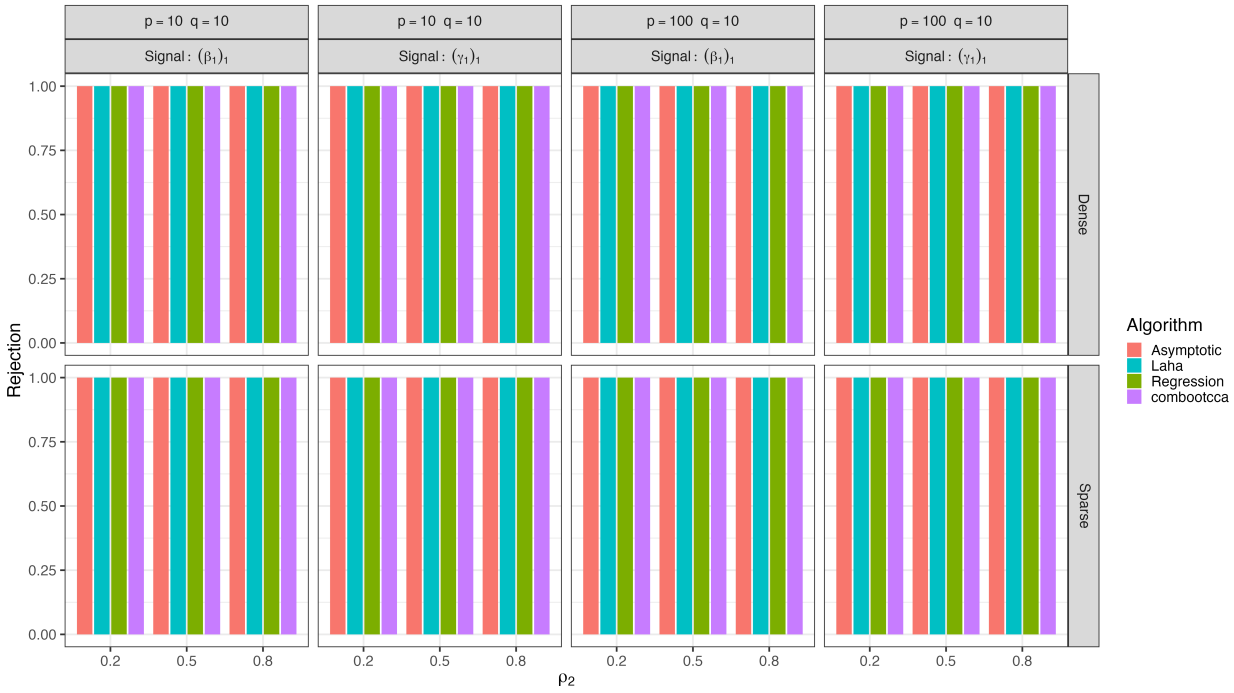

Figure B.12: Power (correct rejection rates) for first canonical directions in simulation II.

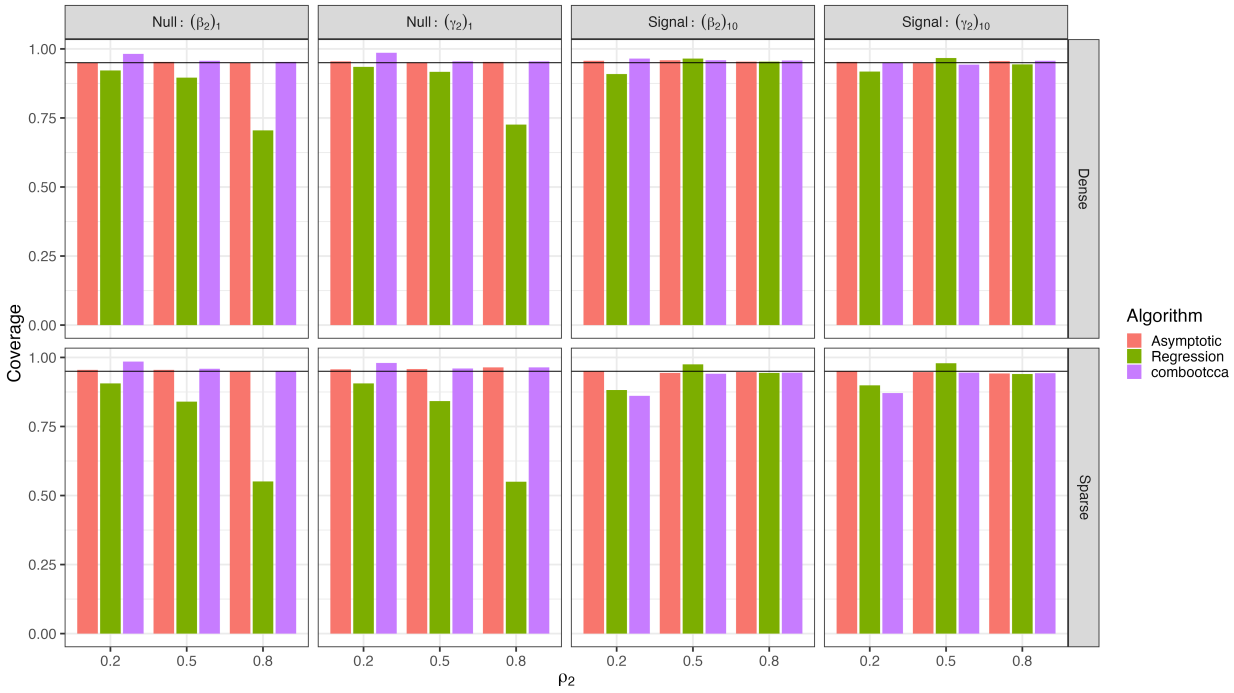

Figure B.13: Coverage rates for second canonical directions in simulation II for $p = q = 10$. The horizontal line indicates nominal 95% coverage.

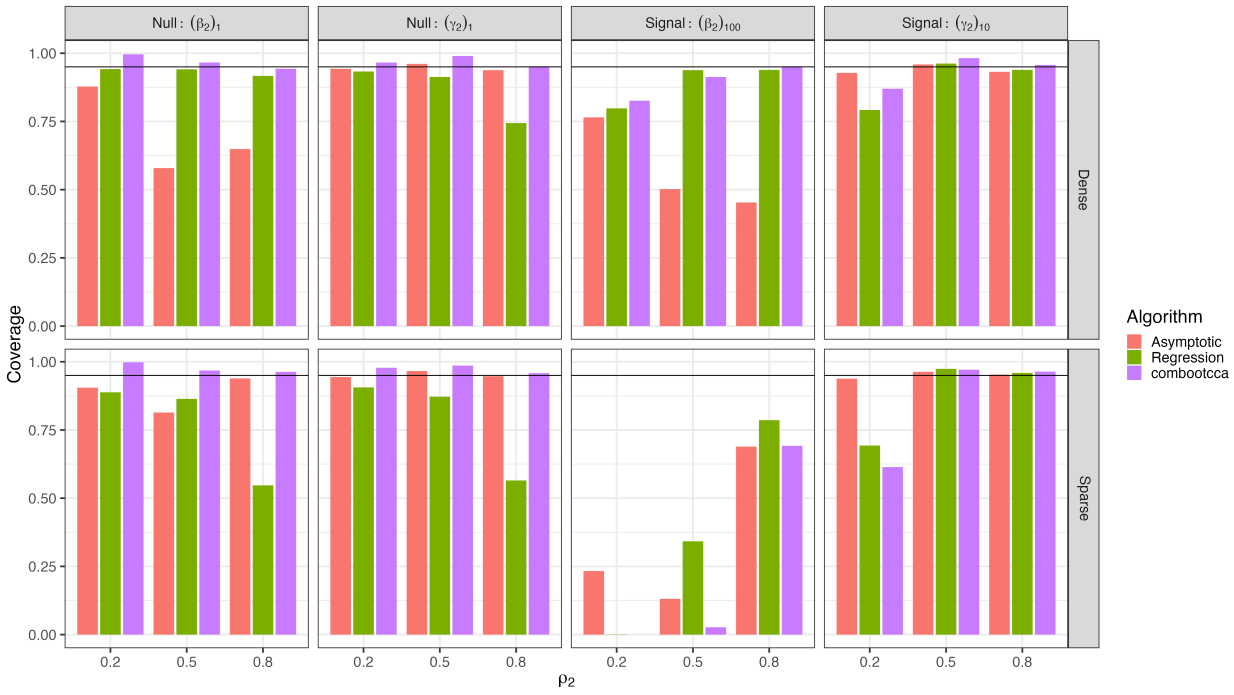

Figure B.14: Coverage rates for second canonical directions in simulation II for $p = 100, q = 10$. The horizontal line indicates nominal 95% coverage.

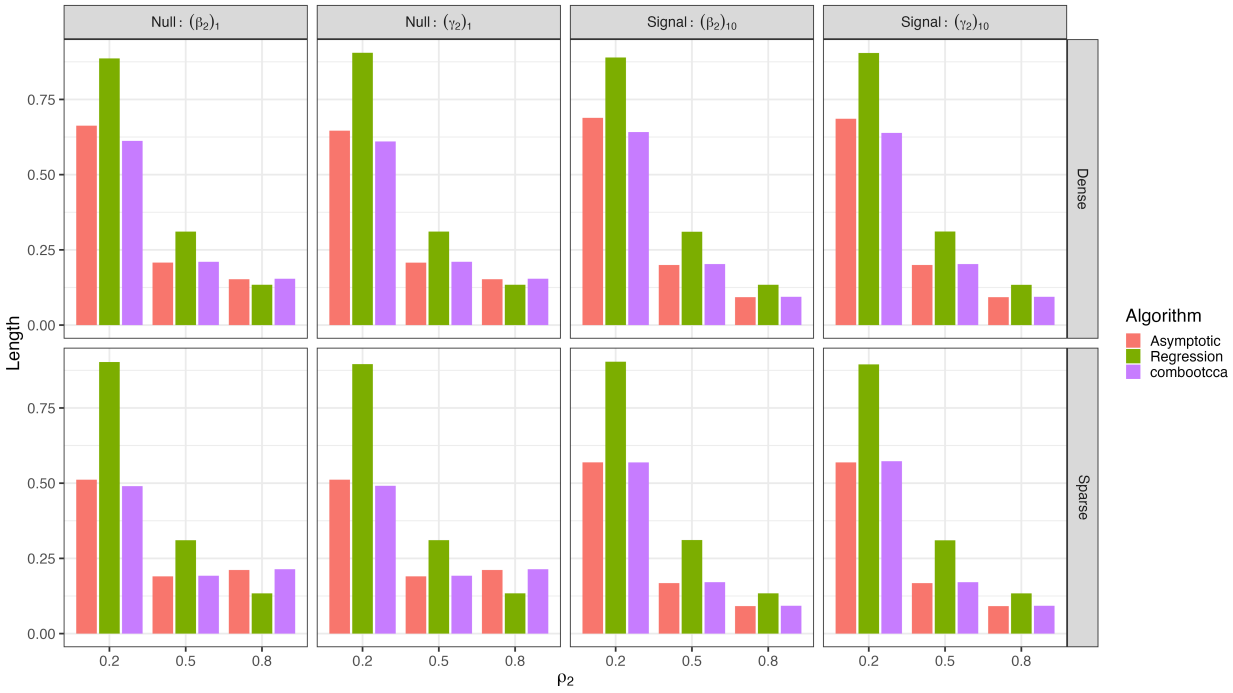

Figure B.15: Lengths of confidence intervals for second canonical directions in simulation II for $p = q = 10$.

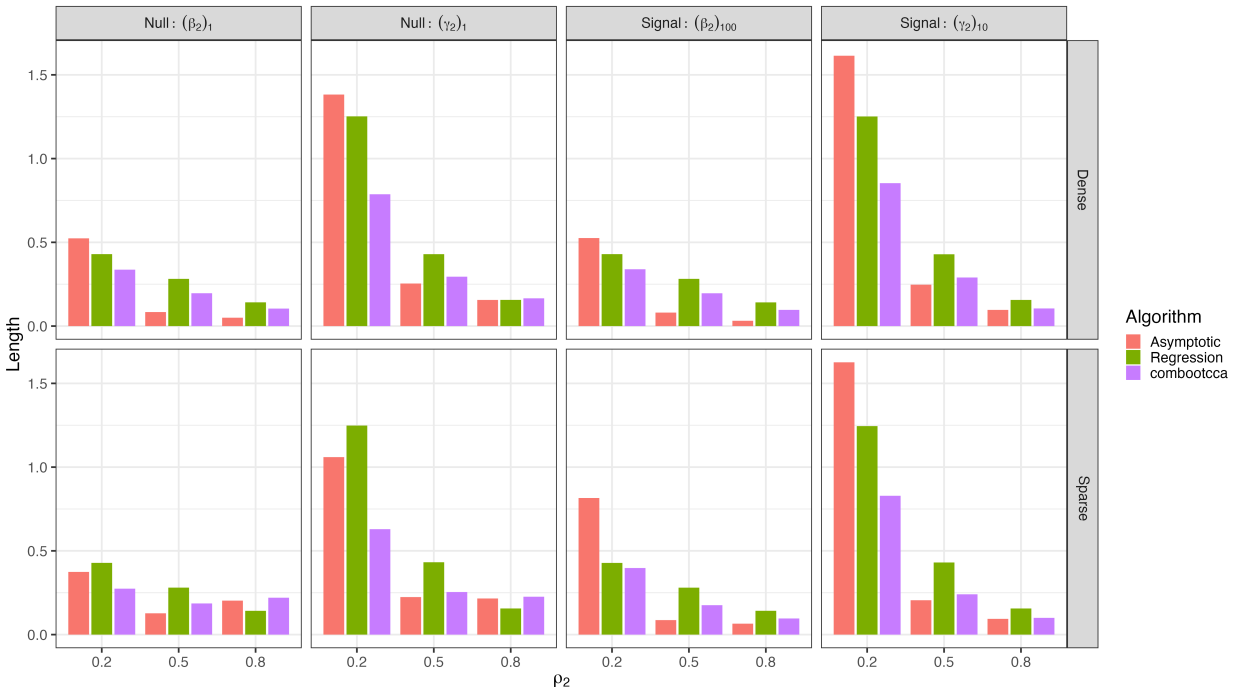

Figure B.16: Lengths of confidence intervals for second canonical directions in simulation II for $p = 100, q = 10$.

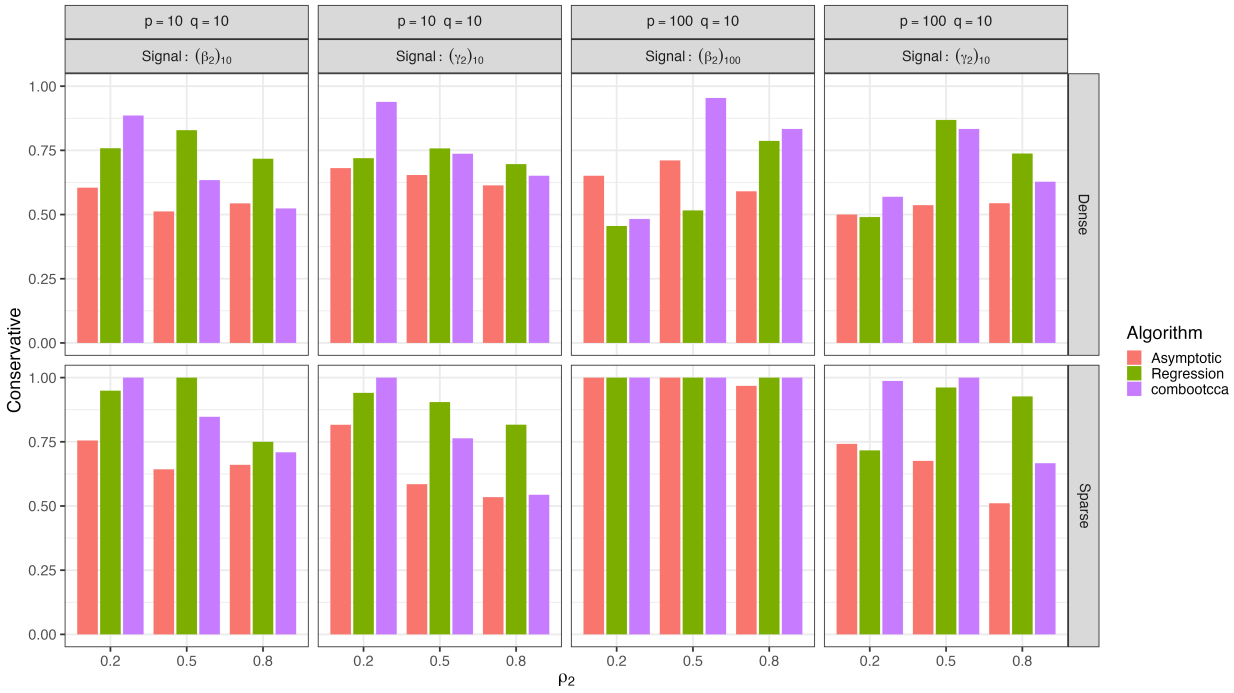

Figure B.17: Bias in simulation II for second canonical directions: the proportion of confidence intervals that failed to cover non-null signals that are "conservative" (the true value is greater in magnitude than any value in the confidence interval).

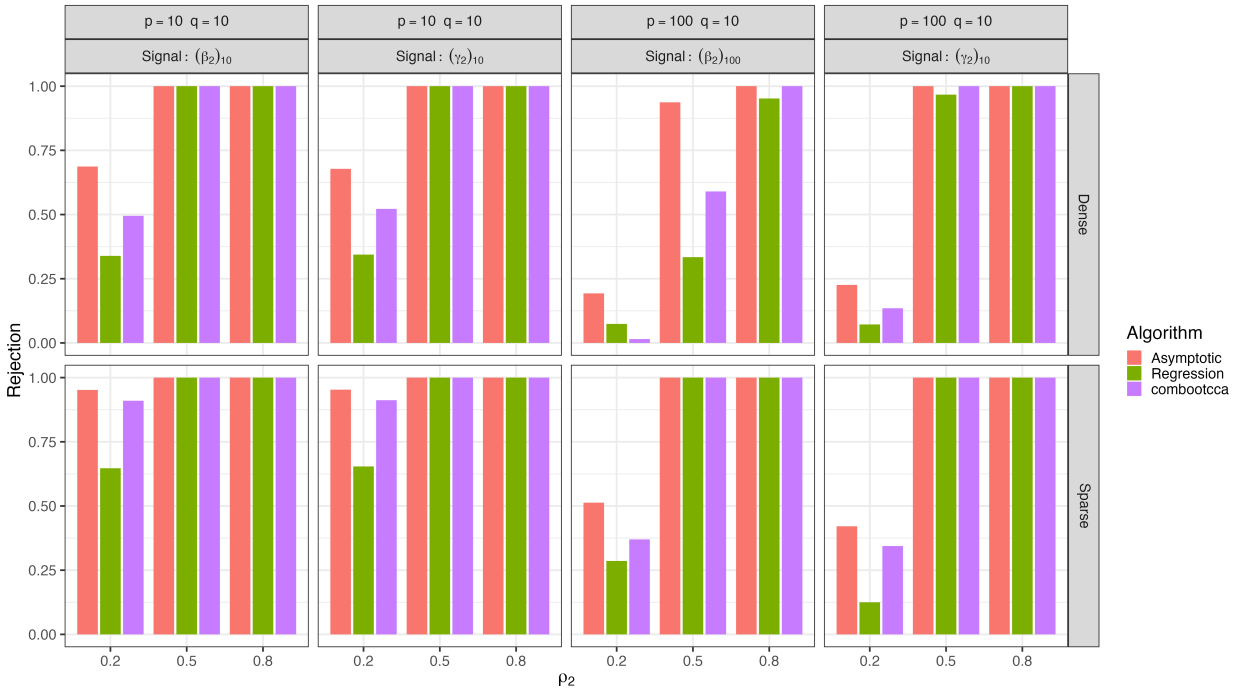

Figure B.18: Power (correct rejection rates) for second canonical directions in simulation II.

BIBLIOGRAPHY

Agterberg, J., Lubberts, Z., and Priebe, C. E. (2022). Entrywise Estimation of Singular Vectors of Low-Rank Matrices With Heteroskedasticity and Dependence. *IEEE Transactions on Information Theory*, 68(7):4618–4650.

Akaho, S. (2007). A kernel method for canonical correlation analysis. arXiv:cs/0609071.

Alnæs, D., Kaufmann, T., Marquand, A. F., Smith, S. M., and Westlye, L. T. (2020). Patterns of sociocognitive stratification and perinatal risk in the child brain. *Proceedings of the National Academy of Sciences*, 117(22):12419–12427.

Anderson, T. W. (1999). Asymptotic Theory for Canonical Correlation Analysis. *Journal of Multivariate Analysis*, 70(1):1–29.

Anderson, T. W. (2003). *An Introduction to Multivariate Statistical Analysis*. Wiley Series in Probability and Statistics. Wiley-Interscience, Hoboken, N.J, 3rd ed edition.

Arbabshirani, M. R., Plis, S., Sui, J., and Calhoun, V. D. (2017). Single subject prediction of brain disorders in neuroimaging: Promises and pitfalls. *NeuroImage*, 145:137–165. Individual Subject Prediction.

Bach, F. R. and Jordan, M. I. (2005). A Probabilistic Interpretation of Canonical Correlation Analysis. Technical report, University of California, Berkeley.

Barch, D. M., Burgess, G. C., Harms, M. P., Petersen, S. E., Schlaggar, B. L., Corbetta, M., Glasser, M. F., Curtiss, S., Dixit, S., Feldt, C., Nolan, D., Bryant, E., Hartley, T., Footer, O., Bjork, J. M., Poldrack, R., Smith, S., Johansen-Berg, H., Snyder, A. Z., and Van Essen, D. C. (2013). Function in the human connectome: Task-fMRI and individual differences in behavior. *NeuroImage*, 80:169–189.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300.

Benjamini, Y. and Yekutieli, D. (2001). The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics*, 29(4):1165–1188.

Benjamini, Y. and Yekutieli, D. (2005). False Discovery Rate–Adjusted Multiple Confidence Intervals for Selected Parameters. *Journal of the American Statistical Association*, 100(469):71–81.

Bilker, W. B., Hansen, J. A., Brensinger, C. M., Richard, J., Gur, R. E., and Gur, R. C. (2012). Development of abbreviated nine-item forms of the raven's standard progressive matrices test. *Assessment*, 19(3):354–369. PMID: 22605785.

Björck, Å. and Golub, G. H. (1973). Numerical Methods for Computing Angles Between Linear Subspaces. *Mathematics of Computation*, 27(123):579–594.

Bogomolov, M., Peterson, C. B., Benjamini, Y., and Sabatti, C. (2021). Hypotheses on a tree: New error rates and testing strategies. *Biometrika*, 108(3):575–590.

Breheny, P. (2015). The group exponential lasso for bi-level variable selection. *Biometrics*, 71(3):731–740.

Breheny, P. and Huang, J. (2009). Penalized methods for bi-level variable selection. *Statistics and Its Interface*, 2(3):369–380.

Breheny, P. and Huang, J. (2015). Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and Computing*, 25(2):173–187.

Brett, M., Anton, J.-L., Valabregue, R., and Poline, J.-B. (2002). Region of interest analysis using an SPM toolbox. In *Presented at the 8th International Conference on Functional Mapping of the Human Brain*. Abstract Available in NeuroImage, Vol 16, No 2.

Brzyski, D., Hu, X., Goni, J., Ances, B., Randolph, T. W., and Harezlak, J. (2020). A sparsity inducing nuclear-norm estimator (SpINNEr) for matrix-variate regression in brain connectivity analysis. *arXiv:2001.11548 [cs, math, stat]*.

Burgos, N. and Colliot, O. (2020). Machine learning for classification and prediction of brain diseases: Recent advances and upcoming challenges. *Current Opinion in Neurology*, 33(4):439–450.

Butts, C. T. (2022). *Yacca: Yet Another Canonical Correlation Analysis Package*.

Bykhovskaya, A. and Gorin, V. (2023). High-dimensional canonical correlation analysis.

Bühlmann, P. and van de Geer, S. (2011). *Statistics for High-Dimensional Data*. Springer Series in Statistics. Springer Berlin Heidelberg.

Cai, J.-F., Candès, E. J., and Shen, Z. (2010). A Singular Value Thresholding Algorithm for Matrix Completion. *SIAM Journal on Optimization*, 20(4):1956–1982.

Cai, T. T., Zhang, A., and Zhou, Y. (2019). Sparse Group Lasso: Optimal Sample Complexity, Convergence Rate, and Statistical Inference. *arXiv:1909.09851*.

Calhoun, V. D., Lawrie, S. M., Mourao-Miranda, J., and Stephan, K. E. (2017). Prediction of individual differences from neuroimaging data. *NeuroImage*, 145:135–136. Individual Subject Prediction.

Calhoun, V. D. and Sui, J. (2016). Multimodal Fusion of Brain Imaging Data: A Key to Finding the Missing Link(s) in Complex Mental Illness. *Biological Psychiatry: Cognitive Neuroscience and Neuroimaging*, 1(3):230–244.

Canty, A. and Ripley, B. D. (2022). *Boot: Bootstrap R (S-plus) Functions*.

Casey, B. J., Cannonier, T., Conley, M. I., Cohen, A. O., Barch, D. M., Heitzeg, M. M., Soules, M. E., Teslovich, T., Dellarco, D. V., Garavan, H., Orr, C. A., Wager, T. D., Banich, M. T., Speer, N. K., Sutherland, M. T., Riedel, M. C., Dick, A. S., Bjork, J. M., Thomas, K. M., Chaarani, B., Mejia, M. H., Hagler, D. J., Daniela Cornejo, M., Sicat, C. S., Harms, M. P., Dosenbach, N. U. F., Rosenberg, M., Earl, E., Bartsch, H., Watts, R., Polimeni, J. R., Kuperman, J. M., Fair, D. A., and Dale, A. M. (2018). The Adolescent Brain Cognitive Development (ABCD) study: Imaging acquisition across 21 sites. *Developmental Cognitive Neuroscience*, 32:43–54.

Chen, M., Gao, C., Ren, Z., and Zhou, H. H. (2013). Sparse CCA via precision adjusted iterative thresholding.

Chen, Y.-L., Kolar, M., and Tsay, R. S. (2021). Tensor Canonical Correlation Analysis With Convergence and Statistical Guarantees. *Journal of Computational and Graphical Statistics*, 30(3):728–744.

Chung, J., Bridgeford, E., Arroyo, J., Pedigo, B. D., Saad-Eldin, A., Gopalakrishnan, V., Xiang, L., Priebe, C. E., and Vogelstein, J. T. (2021). Statistical Connectomics. *Annual Review of Statistics and Its Application*, 8(1):463–492.

Crainiceanu, C. M., Caffo, B. S., Luo, S., Zipunnikov, V. M., and Punjabi, N. M. (2011). Population Value Decomposition, a Framework for the Analysis of Image Populations. *Journal of the American Statistical Association*, 106(495):775–790.

Davison, A. C. and Hinkley, D. V. (1997). *Bootstrap Methods and Their Applications*. Cambridge University Press, Cambridge.

Dinga, R., Schmaal, L., Penninx, B. W. J. H., van Tol, M. J., Veltman, D. J., van Velzen, L., Mennes, M., van der Wee, N. J. A., and Marquand, A. F. (2019). Evaluating the evidence for biotypes of depression: Methodological replication and extension of drysdale et al. (2017). *NeuroImage: Clinical*, 22:101796.

Duncan, J., Seitz, R. J., Kolodny, J., Bor, D., Herzog, H., Ahmed, A., Newell, F. N., and Emslie, H. (2000). A Neural Basis for General Intelligence. *Science*, 289(5478):457–460. Publisher: American Association for the Advancement of Science.

Eckstein, J. and Bertsekas, D. P. (1992). On the Douglas—Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1):293–318.

Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*. Springer US, Boston, MA.

Fan, J., Liao, Y., and Liu, H. (2016). An overview of the estimation of large covariance and precision matrices. *The Econometrics Journal*, 19(1):C1–C32.

Fan, J., Wang, W., and Zhu, Z. (2017). A shrinkage principle for heavy-tailed data: High-dimensional robust low-rank matrix recovery.

Fazel, M., Hindi, H., and Boyd, S. (2001). A rank minimization heuristic with application to minimum order system approximation. In *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, volume 6, pages 4734–4739 vol.6. ISSN: 0743-1619.

Fernandez-Cabello, S., Alnæs, D., van der Meer, D., Dahl, A., Holm, M., Kjelkenes, R., Maximov, I. I., Norbom, L. B., Pedersen, M. L., Voldsbekk, I., Andreassen, O. A., and Westlye, L. T. (2022). Associations between brain imaging and polygenic scores of mental health and educational attainment in children aged 9–11. *NeuroImage*, 263:119611.

Fine, J. (2003). Asymptotic study of canonical correlation analysis: From matrix and analytic approach to operator and tensor approach. *SORT*, 27(2):165–174.

Fox, M. D. (2018). Mapping Symptoms to Brain Networks with the Human Connectome. *New England Journal of Medicine*, 379(23):2237–2245.

Friedman, J., Hastie, T., and Tibshirani, R. (2010a). A note on the group lasso and a sparse group lasso. *arXiv:1001.0736*.

Friedman, J. H., Hastie, T., and Tibshirani, R. (2010b). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, 33(1):1–22.

Gao, C., Ma, Z., and Zhou, H. H. (2017). Sparse CCA: Adaptive estimation and computational barriers. *Annals of Statistics*, 45(5):2074–2101.

Ginestet, C. E., Li, J., Balachandran, P., Rosenberg, S., and Kolaczyk, E. D. (2017). Hypothesis testing for network data in functional neuroimaging. *The Annals of Applied Statistics*, 11(2):725–750.

Girka, F., Camenen, E., Peltier, C., Gloaguen, A., Guillemot, V., Le Brusquet, L., and Tenenhaus, A. (2023). *RGCCA: Regularized and Sparse Generalized Canonical Correlation Analysis for Multiblock Data*. R package version 3.0.2.

Golub, G. H. and Van Loan, C. F. (2013). *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, Baltimore, fourth edition edition.

Gordon, E. M., Laumann, T. O., Adeyemo, B., Huckins, J. F., Kelley, W. M., and Petersen, S. E. (2016). Generation and Evaluation of a Cortical Area Parcellation from Resting-State Correlations. *Cerebral Cortex*, 26(1):288–303.

Goyal, N., Moraczewski, D., Bandettini, P. A., Finn, E. S., and Thomas, A. G. (2022). The positive–negative mode link between brain connectivity, demographics and behaviour: A pre-registered replication of Smith et al. (2015). *Royal Society Open Science*, 9(2):201090.

Helmer, M., Warrington, S. D., Mohammadi-Nejad, A.-R., Ji, J. L., Howell, A., Rosand, B., Anticevic, A., Sotiropoulos, S. N., and Murray, J. D. (2020). On stability of canonical correlation analysis and partial least squares with application to brain-behavior associations. *bioRxiv*, page 2020.08.25.265546.

Hodes, R. J., Insel, T. R., Landis, S. C., and Research, O. b. o. t. N. B. f. N. (2013). The NIH Toolbox: Setting a standard for biomedical research. *Neurology*, 80(11 Supplement 3):S1–S1.

Hotelling, H. (1935). The most predictable criterion. *Journal of Educational Psychology*, 26(2):139–142.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika*, 28(3-4):321–377.

Hsu, W.-T., Rosenberg, M. D., Scheinost, D., Constable, R. T., and Chun, M. M. (2018). Resting-state functional connectivity predicts neuroticism and extraversion in novel individuals. *Social Cognitive and Affective Neuroscience*, 13(2):224–232.

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

Huang, J. and Zhang, T. (2010). The benefit of group sparsity. *The Annals of Statistics*, 38(4):1978–2004.

Jacob, L., Obozinski, G., and Vert, J.-P. (2009). Group Lasso with Overlap and Graph Lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 433–440. ACM.

Kebets, V., Holmes, A. J., Orban, C., Tang, S., Li, J., Sun, N., Kong, R., Poldrack, R. A., and Yeo, B. T. T. (2019). Somatosensory-Motor Dysconnectivity Spans Multiple Transdiagnostic Dimensions of Psychopathology. *Biological Psychiatry*, 86(10):779–791.

Kessler, D., Levin, K., and Levina, E. (2022). Prediction of Network Covariates Using Edge and Node Attributes.

Khosla, M., Jamison, K., Ngo, G. H., Kuceyeski, A., and Sabuncu, M. R. (2019). Machine learning in resting-state fMRI analysis. *Magnetic Resonance Imaging*, 64:101–121.

Kim, Y., Kessler, D., and Levina, E. (2023). Graph-aware Modeling of Brain Connectivity Networks. *The Annals of Applied Statistics*.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nav.3800020109.

Laha, Huey, N., Coull, B., and Mukherjee, R. (2021). On statistical inference with high dimensional sparse CCA. *arXiv:2109.11997 [math, stat]*.

Lang, M., Bischl, B., and Surmann, D. (2017). Batchtools: Tools for R to work on batch systems. *Journal of Open Source Software*, 2(10):135.

Lee, S. H. and Choi, S. (2007). Two-dimensional canonical correlation analysis. *IEEE Signal Processing Letters*, 14(10):735–738.

Linke, J. O., Abend, R., Kircanski, K., Clayton, M., Stavish, C., Benson, B. E., Brotman, M. A., Renaud, O., Smith, S. M., Nichols, T. E., Leibenluft, E., Winkler, A. M., and Pine, D. S. (2021). Shared and Anxiety-Specific Pediatric Psychopathology Dimensions Manifest Distributed Neural Correlates. *Biological Psychiatry*, 89(6):579–587.

Luciana, M., Bjork, J. M., Nagel, B. J., Barch, D. M., Gonzalez, R., Nixon, S. J., and Banich, M. T. (2018). Adolescent neurocognitive development and impacts of substance use: Overview of the adolescent brain cognitive development (ABCD) baseline neurocognition battery. *Developmental Cognitive Neuroscience*, 32:67–79.

Mai, Q. and Zhang, X. (2019). An iterative penalized least squares approach to sparse canonical correlation analysis. *Biometrics*, 75(3):734–744.

Margulies, D. S., Ghosh, S. S., Goulas, A., Falkiewicz, M., Huntenburg, J. M., Langs, G., Bezgin, G., Eickhoff, S. B., Castellanos, F. X., Petrides, M., Jefferies, E., and Smallwood, J. (2016). Situating the default-mode network along a principal gradient of macroscale cortical organization. *Proceedings of the National Academy of Sciences of the United States of America*, 113(44):12574–12579.

Marron, J. and Dryden, I. L. (2021). *Object Oriented Data Analysis*. Chapman and Hall/CRC, Boca Raton, 1 edition.

McCrae, R. R. and Costa, P. T. (2004). A contemplated revision of the neo five-factor inventory. *Personality and Individual Differences*, 36(3):587–596.

McCullagh, P. and Nelder, J. A. (1998). *Generalized Linear Models*. Number 37 in Monographs on Statistics and Applied Probability. Chapman & Hall/CRC, 2nd ed edition.

McIntosh, A. R. (2021). Comparison of Canonical Correlation and Partial Least Squares analyses of simulated and empirical data. *arXiv:2107.06867 [stat]*.

McIntosh, A. R. and Lobaugh, N. J. (2004). Partial least squares analysis of neuroimaging data: Applications and advances. *NeuroImage*, 23:S250–S263.

McKeague, I. W. and Zhang, X. (2022). Significance testing for canonical correlation analysis in high dimensions. *Biometrika*, 109(4):1067–1083.

Miller, K. L., Alfaro-Almagro, F., Bangerter, N. K., Thomas, D. L., Yacoub, E., Xu, J., Bartsch, A. J., Jbabdi, S., Sotiropoulos, S. N., Andersson, J. L. R., Griffanti, L., Douaud, G., Okell, T. W., Weale, P., Dragonu, I., Garratt, S., Hudson, S., Collins, R., Jenkinson, M., Matthews, P. M., and Smith, S. M. (2016). Multimodal population brain imaging in the UK Biobank prospective epidemiological study. *Nature Neuroscience*, 19(11):1523–1536.

Mišić, B., Betzel, R. F., de Reus, M. A., van den Heuvel, M. P., Berman, M. G., McIntosh, A. R., and Sporns, O. (2016). Network-level structure-function relationships in human neocortex. *Cerebral Cortex*, 26(7):3285–3296.

Muirhead, R. J., editor (1982). *Aspects of Multivariate Statistical Theory*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., Hoboken, NJ, USA.

Nakua, H., Yu, J.-C., Abdi, H., Hawco, C., Voineskos, A., Hill, S., Lai, M.-C., Wheeler, A. L., McIntosh, A. R., and Ameis, S. H. (2023). Comparing the stability and reproducibility of brain-behaviour relationships found using Canonical Correlation Analysis and Partial Least Squares within the ABCD Sample.

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

Noble, S., Mejia, A. F., Zalesky, A., and Scheinost, D. (2022). Improving power in functional magnetic resonance imaging by moving beyond cluster-level inference. *Proceedings of the National Academy of Sciences*, 119(32):e2203020119.

Obozinski, G., Jacob, L., and Vert, J.-P. (2011). Group Lasso with Overlaps: The Latent Group Lasso approach. *arXiv:1110.0413*.

Panigrahi, S., MacDonald, P. W., and Kessler, D. (2023a). Approximate post-selective inference for regression with the group lasso. *Journal of Machine Learning Research*, 24(79):1–49.

Panigrahi, S., Stewart, N., Sripada, C. S., and Levina, E. (2023b). Selective Inference for Sparse Multitask Regression with Applications in Neuroimaging.

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., and Petersen, S. E. (2011). Functional Network Organization of the Human Brain. *Neuron*, 72(4):665–678.

R Core Team (2023). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

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):1648–1677.

Richie-Halford, A., Yeatman, J. D., Simon, N., and Rokem, A. (2021). Multidimensional analysis and detection of informative features in human brain white matter. *PLOS Computational Biology*, 17(6):e1009136.

Rosa, M. J., Mehta, M. A., Pich, E. M., Risterucci, C., Zelaya, F., Reinders, A. A. T. S., Williams, S. C. R., Dazzan, P., Doyle, O. M., and Marquand, A. F. (2015). Estimating multivariate similarity between neuroimaging datasets with sparse canonical correlation analysis: An application to perfusion imaging. *Frontiers in Neuroscience*, 9.

Safayani, M., Ahmadi, S. H., Afrabandpey, H., and Mirzaei, A. (2018). An EM based probabilistic two-dimensional CCA with application to face recognition. *Applied Intelligence*, 48(3):755–770.

Schönemann, P. H. (1966). A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10.

Shen, X., Finn, E. S., Scheinost, D., Rosenberg, M. D., Chun, M. M., Papademetris, X., and Constable, R. T. (2017). Using connectome-based predictive modeling to predict individual behavior from brain connectivity. *Nature Protocols*, 12(3):506–518.

Shimizu, Y., Yoshimoto, J., Toki, S., Takamura, M., Yoshimura, S., Okamoto, Y., Yamawaki, S., and Doya, K. (2015). Toward Probabilistic Diagnosis and Understanding of Depression Based on Functional MRI Data Analysis with Logistic Group LASSO. *PLOS ONE*, 10(5):e0123524.

Silverman, J. (2022). *RcppHungarian: Solves Minimum Cost Bipartite Matching Problems*.

Simon, N. and Tibshirani, R. (2012). Standardization and the Group Lasso Penalty. *Statistica Sinica*, 22(3):983–1001.

Smith, S. M., Nichols, T. E., Vidaurre, D., Winkler, A. M., Behrens, T. E. J., Glasser, M. F., Ugurbil, K., Barch, D. M., Van Essen, D. C., and Miller, K. L. (2015). A positive-negative mode of population covariation links brain connectivity, demographics and behavior. *Nature Neuroscience*, 18(11):1565–1567.

Sripada, C., Angstadt, M., Rutherford, S., Kessler, D., Kim, Y., Yee, M., and Levina, E. (2019). Basic Units of Inter-Individual Variation in Resting State Connectomes. *Scientific Reports*, 9(1):1900.

Sripada, C., Angstadt, M., Rutherford, S., Taxali, A., and Shedden, K. (2020). Toward a "treadmill test" for cognition: Improved prediction of general cognitive ability from the task activated brain. *Human Brain Mapping*, 41(12):3186–3197.

Sripada, C., Angstadt, M., Taxali, A., Clark, D. A., Greathouse, T., Rutherford, S., Dickens, J. R., Shedden, K., Gard, A. M., Hyde, L. W., Weigard, A., and Heitzeg, M. (2021). Brain-wide functional connectivity patterns support general cognitive ability and mediate effects of socioeconomic status in youth. *Translational Psychiatry*, 11(1):1–8.

Tang, M., Athreya, A., Sussman, D. L., Lyzinski, V., Park, Y., and Priebe, C. E. (2017). A Semiparametric Two-Sample Hypothesis Testing Problem for Random Graphs. *Journal of Computational and Graphical Statistics*, 26(2):344–354.

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108.

Tong, X., Xie, H., Carlisle, N., Fonzo, G. A., Oathes, D. J., Jiang, J., and Zhang, Y. (2022). Transdiagnostic connectome signatures from resting-state fMRI predict individual-level intellectual capacity. *Translational Psychiatry*, 12(1):1–11. Number: 1 Publisher: Nature Publishing Group.

Tuzhilina, E., Tozzi, L., and Hastie, T. (2021). Canonical correlation analysis in high dimensions with structured regularization. *Statistical Modelling*, page 1471082X211041033.

Uludağ, K. and Roebroeck, A. (2014). General overview on the merits of multimodal neuroimaging data fusion. *NeuroImage*, 102:3–10.

Van Essen, D. C., Smith, S. M., Barch, D. M., Behrens, T. E. J., Yacoub, E., Ugurbil, K., and the WU-Minn HCP Consortium (2013). The WU-Minn Human Connectome Project: An overview. *NeuroImage*, 80:62–79.

Vinod, H. D. (1976). Canonical ridge and econometrics of joint production. *Journal of Econometrics*, 4(2):147–166.

Wang, H., Yang, Y., and Su, W. J. (2020a). The Price of Competition: Effect Size Heterogeneity Matters in High Dimensions. *arXiv:2007.00566*.

Wang, H.-T., Smallwood, J., Mourao-Miranda, J., Xia, C. H., Satterthwaite, T. D., Bassett, D. S., and Bzdok, D. (2020b). Finding the needle in a high-dimensional haystack: Canonical correlation analysis for neuroscientists. *NeuroImage*, 216:116745.

Winkler, A. M., Renaud, O., Smith, S. M., and Nichols, T. E. (2020). Permutation inference for canonical correlation analysis. *NeuroImage*, 220:117065.

Witten, D. M., Tibshirani, R., and Hastie, T. (2009). A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515–534.

Xia, C. H., Ma, Z., Ciric, R., Gu, S., Betzel, R. F., Kaczkurkin, A. N., Calkins, M. E., Cook, P. A., García de la Garza, A., Vandekar, S. N., Cui, Z., Moore, T. M., Roalf, D. R., Ruparel, K., Wolf, D. H., Davatzikos, C., Gur, R. C., Gur, R. E., Shinohara, R. T., Bassett, D. S., and Satterthwaite, T. D. (2018). Linked dimensions of psychopathology and connectivity in functional brain networks. *Nature Communications*, 9(1):3003.

Xia, C. H., Ma, Z., Cui, Z., Bzdok, D., Thirion, B., Bassett, D. S., Satterthwaite, T. D., Shinohara, R. T., and Witten, D. M. (2020). Multi-scale network regression for brain-phenotype associations. *Human Brain Mapping*.

Yang, F., Foygel Barber, R., Jain, P., and Lafferty, J. (2016). Selective inference for group-sparse linear models. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Yekutieli, D. and Benjamini, Y. (1999). Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics. *Journal of Statistical Planning and Inference*, 82(1–2):171–196.

Yeo, B. T. T., Krienen, F. M., Sepulcre, J., Sabuncu, M. R., Lashkari, D., Hollinshead, M., Roffman, J. L., Smoller, J. W., Zöllei, L., Polimeni, J. R., Fischl, B., Liu, H., and Buckner, R. L. (2011). The organization of the human cerebral cortex estimated by intrinsic functional connectivity. *Journal of Neurophysiology*, 106(3):1125–1165.

Yu, M., Linn, K. A., Shinohara, R. T., Oathes, D. J., Cook, P. A., Duprat, R., Moore, T. M., Oquendo, M. A., Phillips, M. L., McInnis, M., Fava, M., Trivedi, M. H., McGrath, P., Parsey, R., Weissman, M. M., and Sheline, Y. I. (2019). Childhood trauma history is linked to abnormal brain connectivity in major depression. *Proceedings of the National Academy of Sciences*, 116(17):8582–8590.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.

Zeng, Y. and Breheny, P. (2016). Overlapping Group Logistic Regression with Applications to Genetic Pathway Selection. *Cancer Informatics*, 15.

Zhang, W., Paul, S. E., Winkler, A., Bogdan, R., and Bijsterbosch, J. D. (2022). Shared brain and genetic architectures between mental health and physical activity. *Translational Psychiatry*, 12(1):1–12.

Zhou, H. and Li, L. (2014). Regularized Matrix Regression. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 76(2):463–483.

Zhuang, X., Yang, Z., and Cordes, D. (2020). A technical review of canonical correlation analysis for neuroscience applications. *Human Brain Mapping*, 41(13):3807–3833.