Distribution Agreement

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

Signature:

Yikai Wang	Date

Novel Statistical and Machine Learning Methods with Application to Brain Imaging Data

By

Yikai Wang

Doctor of Philosophy

Biostatistics

Ying Guo, Ph.D.
Advisor

Benjamin B. Risk, Ph.D.
Committee Member

Howard H. Chang, Ph.D.
Committee Member

Shella Keilholz, Ph.D.
Committee Member

Accepted:

Lisa A. Tedesco, Ph.D.
Dean of the James T. Laney School of Graduate Studies

Date

Novel Statistical and Machine Learning Methods with Application to Brain Imaging Data

By

Yikai Wang
Master of Science, Emory University, 2019
Master of Public Health, Emory University, 2015
Bachelor of Science, South China University of Technology, 2013

Advisor: Ying Guo, Ph.D.

An abstract of
A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Biostatistics
2020

Abstract

Novel Statistical and Machine Learning Methods with Application to Brain Imaging Data

By

Yikai Wang

Brain imaging has been a breakthrough technique for understanding the functionality and organization of the human brain, serving as the fundamental basis for neuroscientific research. My dissertation is focusing on developing statistical and machine learning methods for brain imaging data.

For the first topic, we propose a novel hierarchical independent component modeling framework for longitudinal fMRI study (L-ICA). Existing ICA methods are only applicable for cross-sectional study. In this topic, we provide the first formal statistical modeling framework extending ICA to longitudinal study. By incorporating subject-specific random effects and visit-specific covariate effects, L-ICA is able to provide more accurate estimates for brain networks and borrow information across repeated scans to increase statistical power in detecting the covariate effects. We develop a fully traceable EM algorithm and a subspace-based approximate EM algorithm which greatly reduce the computation time while retaining high accuracy. Simulation and real data results demonstrate the advantages of L-ICA.

For the second topic, we propose a novel blind signal separation (BSS) model for decomposing brain connectivity matrices. Existing BSS methods are mainly focusing on decomposing neural activity signals, instead of brain connectivities. In this topic, we propose a low-rank decomposition method with uniform sparsity (LOCUS) for brain network measures. LOCUS adopts a low-rank factorization in each latent signal for robust recovery, and also incorporates a novel penalization approach for sparsity control on latent sources. We propose a highly efficient algorithm for parameter estimation. Simulation and real data results show that LOCUS provides highly reproducible findings than existing approaches.

For the third topic, we propose a novel deep learning (DL) framework for brain network data. DL methods are often criticized for low interpretability and instability. By incorporating the existing brain subnetwork structure, we propose a DL framework with adaptively shaped graph convolutional layer (DLconv) for brain network. Specifically, the shape of convolutional filter is driven by brain subnetwork, and subnetwork-level features are propagated separately until the final layer. With the inherent structure in DLconv, we propose a robust training procedure by updating the subnetwork-specific parameters in parallel. Real data studies demonstrate the advantages of DLconv.

Novel Statistical and Machine Learning Methods with Application to Brain Imaging Data

By

Yikai Wang
Master of Science, Emory University, 2019
Master of Public Health, Emory University, 2015
Bachelor of Science, South China University of Technology, 2013

Advisor: Ying Guo, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Biostatistics
2020

Acknowledgement

I would like to express my deepest gratitude to my advisor Dr. Ying Guo for teaching me how to explore new research ideas, for encouraging me to discover my potentials, and for supporting me in every regrad during my studies at Emory. Her knowledge and insights in statistics, machine learning and brain imaging provide me the greatest gudiance along this incredible doctoral journey. Her generosity allow me to embrace abundant resources in my research. Her personality sets me a role model in my life. It is my fortune to have her as my P.h.D advisor.

I would also like to thank my dissertation committee members, Dr. Benjamin B. Risk, Dr. Howard H. Chang and Dr. Shella Keilholz for their thoughtful suggestions and invaluable feedback on this dissertation, which have significantly improved the quality of my dissertation research and presentation. I also want to thank Mr. Kirk Easley for his generously financial supports on my stipend for my P.h.D. study. Although I cannot list all names, I really appreciates all the help and support from BIOS faculties and staff, Emory colleagues and friends, who makes my P.h.D. experiences so rewarding and enjoyable.

Lastly, none of this work would have been possible without the encouragement of my family. Thank you to my parents, Bing Li and Zhaorong Wang, my grandparents, Delin Li and Ruifen Li, and my wife, Xiao Wang, for their unconditional love and support. I could not have been this far in this journey without them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Brain imaging analyses have shown great promise for understanding the functionality and organization of the human brain (Bullmore and Sporns, 2009; Deco et al., 2011; Satterthwaite, Wolf, Roalf, Ruparel, Erus, Vandekar, Gennatas, Elliott, Smith, Hakonarson et al., 2014; Bowman, 2014; Lee et al., 2013). Recent findings from the brain imaging studies provide great insights into the underlying mechanism of neurodevelopment, neural processing, mental disorders and neurological diseases. Moreover, brain imaging can also serve as the biomarker for doctors and researchers to make clinical decision or assist in drug development (Johnson et al., 2013; Poil et al., 2013; Lee et al., 2013). Currently, there are several commonly used brain imaging modalities in the neuroscientific field, such as functional magnetic resonance imaging (fMRI), diffusion tensor imaging (DTI), positron emission tomography (PET) and electroencephalography (EEG). These imaging modalities are all from different perspectives which captures certain neuro-characteristics reflecting brain functional or structural properties. However, these data modalities are usually high-dimensional, has complicated spatial or temporal structure and has relatively low signal-to-noise ratio (Bowman, 2014; Shi, 2016). These properties make it very challenging to analyze the brain imaging data for scientific discoveries and also limits our ability to interpret the findings from brain imaging study. Therefore, the goal of this dissertation is to develop novel statistical machine learning methods to facilitate the analysis of brain imaging data.

Currently, blind signal separation (BSS) techniques are widely used in brain imaging study (Calhoun et al., 2003, 2009). BSS is a class of unsupervised machine learning methods aiming at recovering the latent signals from the observed data. Specifically, one of the most famous BSS methods for brain imaging application is independent component analysis (ICA) which can decompose the multivariate signal into inde-

pendent non-Gaussian latent sources (Guo and Tang, 2013; Wang and Guo, 2019). ICA has achieved great success for brain imaging analysis especially for fMRI data decomposition (Calhoun, Adali, Pearlson and Pekar, 2001; Beckmann and Smith, 2004). Specifically, fMRI measures the brain blood oxygen level for anatomically distinct brain regions across temporal domain and is commonly used to study the brain functional properties. ICA can decompose the observed fMRI signals into linear combinations of latent spatial sources signals that are statistically as independent as possible, and these decomposed spatial latent sources are corresponding to brain functional networks (BFN). As a BSS approach, ICA has many advantages. First, it requires no prior information about the human brain and can automatically cluster whole brain voxels into functionally related brain regions. In the meanwhile, ICA learns the weights of each latent signals which is the temporal loadings of each BFN and are widely used for subsequential analysis which significantly reduces the dimensionality and has better interpretability for brain imaging study. Moreover, unlike other second-order methods like PCA, ICA utilizes higher-order statistics and the spatial independence and non-Gaussianity assumptions are well supported by the nature of fMRI activation patterns (Guo and Pagnoni, 2008).

Initially, ICA was proposed for single fMRI data decomposition task (Mckeown et al., 1998). To deal with the group level fMRI data decomposition problem, multiple group ICA methods are proposed (Beckmann and Smith, 2004; Calhoun, Adali, Pearlson and Pekar, 2001; Guo and Tang, 2013; Shi and Guo, 2016). One commonly used group ICA method is the temporal concatenation group ICA (TC-GICA) which stacks all fMRI data across the temporal domain and then decompose the concatenated group level data via ICA method. Another powerful group ICA framework is the hierarchical ICA model which adopts a hierarchical modeling framework in ICA decomposition and models the subject-level variability simultaneously in decomposition (Shi and Guo, 2016). However, these existing group ICA methods are all

focusing on cross-sectional study and are not suitable for longitudinal fMRI data analysis. In recent years, longitudinal studies have become increasingly popular in the neuroscience community. In such studies, the imaging data from same subject can be measured for multiple times across visits which provides great insights into the effects and causal relationships in investigating changes in brain networks related to disease progression, treatment or neurodevelopment (Wang and Guo, 2019). Therefore, in the first topic, I will be working on a novel longitudinal-ICA (L-ICA) framework for jointly decomposing the fMRI data from a longitudinal imaging study. L-ICA is a hierarchical model where the first-level of L-ICA decomposes a subject's fMRI data obtained at a visit into a linear mixture of subject/visit-specific spatial source signals or ICs, and these ICs are then modeled at the second-level of L-ICA in terms of population-level baseline source signals, visit effects, covariate effects, subject-specific random effects and subject/visit-specific random variability. L-ICA is able to account for within-subject correlations among repeated scans, provide more accurate estimates of changes in brain functional networks on both the population and individual-level, and increase statistical power in detecting covariate effects on networks. Furthermore, L-ICA provides model-based prediction for changes in brain networks related to disease progression, treatment or neurodevelopment.

Nowadays, the whole brain network analysis becomes very popular in the neuroscience community (Bullmore and Sporns, 2009; Kemmer et al., 2015; Wang et al., 2016; Kemmer et al., 2018). With the advancement in imaging techniques, researchers can study the whole human brain network structure from functional and structural perspectives (Bullmore and Sporns, 2009; Rubinov and Sporns, 2010). Functional brain network studies the functional dependence among anatomically separated brain regions to construct the brain functional connection (Greicius et al., 2003). Various methods have been proposed to estimate the functional connectivity (FC) from different perspectives. One commonly used FC measure in neuroscience is Pearson's

correlation among brain regions based on fMRI time series (Kemmer et al., 2015). The structural brain network refers to the white matter fiber bundles connecting neurons from spatially remote brain regions (Hagmann et al., 2003). The structural connectivity is usually measured through the probabilities tractography of DTI data which orientates axonal tracts between regions in the brain (Gong et al., 2008). These two types of brain network provide two different perspective for studying the human brain. However, as we discussed, the brain connectivity data is also very challenging to analyze because of its dimensionality and specially structure. The undirected network connectivity data is usually stored as a symmetric matrix. Moreover, given the fact that self-connection is not of interest for network science, the diagonal elements for these connectivity data are usually not defined. This increases the difficulties for existing decomposition methods to be applied for brain connectivity data. Amico et al. (2017) proposed a connectivity independence component analysis (connICA) framework for decomposing brain functional connectivity data where they vectorized the upper-triangle part of the subject-specific connectivity matrix and treated each edge as independent features for existing ICA method. Their work ignores the topological structure in the network and cannot ensure sparsity for selecting the significantly edges in decomposed network. Therefore, in the second topic, I will be working on a novel signal decomposition framework specifically designed for brain connectivity data signal separation problem. We propose to decompose the brain connectivity data into the product of a latent source signal matrix that characterizes underlying network traits and a mixing matrix that contains subject-specific loadings on the source signals. Specifically, in our model, we specify each latent source with a low-rank structure to save the number of parameters for a more robust estimation, and we further propose a novel penalization approach to ensure uniform sparsity in the extracted latent sources to only select biologically meaningful subnetwork connections.

Recently, there is trending evidence in Neuroimaging society supporting the usage of subjects' brain network as individual fingerprint to drive clinical decisions or provide early diagnosis of mental issues, (Kiefer et al., 2015; Wu et al., 2015; Liu et al., 2017). However, brain network based prediction is still facing many challenges in real applications. Due to the large feature space and limited sample size in brain network data, brain network based prediction methods can be highly unstable and lack of interpretability (Chung, 2018). More importantly, human brain network is an extremely complicated system which consists of multiple underlying brain neuro-circuits reflecting different levels of brain subnetwork structure (Amico et al., 2017; Smith et al., 2009; Amico and Goñi, 2018b). Different subnetworks can exhibit distinct neuro-patterns controlling specific functionalities in neuro-processing. This makes the brain network a non-standard, non-linear system to analyze, which further increases the difficulty of using traditional statistical or machine learning methods in brain network prediction tasks. Nowadays, deep learning methods are very popular in many fields because of its high predictive power in terms of dealing with non-linear problems, where deep learning methods usually refer to a class of machine learning methods by stacking multiple layers of neural network models. However, these deep learning methods are usually criticized for the low interpretability and often referred as black box methods (Heaven, 2019). These drawbacks become a very serious problem in brain network analysis especially for clinical usage. Therefore, in the third topic, I will be proposing a novel deep learning framework (DLconv) with interpretable parameters for brain network analysis by incorporating existing brain subnetwork information. Specifically, to deal with the brain network data, we propose to extend the standard convolutional filter by defining an adaptively shaped graph convolutional layer (Mconv) by customizing the filter's shape based on existing subnetwork's masks. Compared with a fully connected layer with same number of outputs, this specification - Mconv - saves a large number of parameters and can

flexibly capture the subnetwork information across the whole brain network. More importantly, the adaptively shaped filter can be mapped back to the brain network for interpretation. Furthermore, we propose to constrain the information flow within each subnetwork and only combine them at the final output layer, which is essentially an ensemble learning model across multiple brain subnetworks. Therefore, the proposed deep learning framework can achieve great interpretability and by making use of existing brain subnetwork structure, it can potentially achieve more robust performance than traditional deep learning methods.

1.2 Overview

Topic 1 is summarized in Chapter 2. Section 2.1 discuss the background and motivation for Chapter 2. Section 2.2 introduces the L-ICA framwork including data preprocessing, model specification, EM algorithm, the approximate EM algorithm and the inference procedure. Section 2.3 presents the simulation study and Section 2.4 includes the real data application of L-ICA framework for a longitudinal fMRI study. Topic 2 is summarized in Chapter 3. Section 3.1 is the introduction for Chatper 3. Section 3.2 introduces the proposed signal separation model including model specification and the novel penalization approach, model training algorithm and model's property. Section 3.3 presents the simulation study and section 3.4 presents the real data application for the Topic 2. Topic 3 is summarized in Chapter 4. Section 4.1 introduces the background and the motivation of Topic 3. Section 4.2 includes the model specification and model training strategies. Section 4.3 shows the real data experiments for Topic 3.

Chapter 2

Group-level fMRI data decomposition for longitudinal imaging study

2.1 Introduction

Brain functional network analysis has been widely used in neuroimaging studies to reveal organization architectures of human brain. In functional imaging studies, neural activity is often captured by a series of 3-D fMRI brain images where the observed data represent the combinations of signals generated from various brain functional networks. One of the major objectives of fMRI-based network analysis is to decompose the observed series of brain images to identify underlying networks and characterize their spatial patterns and temporal dynamics. Independent component analysis (ICA) is one of the most commonly used tools for this purpose. As a special case of blind source separation, ICA decomposes observed fMRI signals into linear combinations of latent spatial source signals that are statistically as independent as possible. These latent independent components correspond to various functional networks. The popularity of the ICA method is mainly due to the following reasons. As a multivariate approach, ICA can jointly model the relationships among multiple voxels and hence provide a tool for investigating whole brain connectivity. Unlike second-order statistical methods such as PCA, ICA takes into account higher-order statistics, and the spatial statistical independence assumption of ICA is well-supported by the sparse nature in typical fMRI activation patterns (Calhoun, Adali, Pearlson and Pekar, 2001; Beckmann and Smith, 2004). Furthermore, ICA is a fully data-driven approach that does not require a priori temporal or spatial models. This makes ICA an important tool for analyzing resting-state fMRI where there is no experimental paradigm (Beckmann et al., 2005).

The classical ICA model was first applied to neuroimaging studies for single subject fMRI data decomposition (Mckeown et al., 1998). Some extensions referred as group ICA (Calhoun, Adali, Pearlson and Pekar, 2001) have been proposed to decompose the multiple-subject fMRI data. One commonly used group ICA framework

is the temporal concatenation group ICA (TC-GICA) which stacks subjects' fMRI data in the temporal domain and then decompose the concatenated group data via ICA (Beckmann and Smith, 2005; Calhoun, Adali, Pearlson and Pekar, 2001; Guo and Pagnoni, 2008). The main limitation of TC-GICA is the assumption of the homogeneity in spatial distribution of the networks across subjects while studies have shown that functional networks can vary considerably due to subjects' clinical, biological and demographic characteristics (Zhao et al., 2007; Greicius et al., 2004, among others). To address this limitation, a hierarchical ICA framework has been proposed to directly account for between-subject differences in group ICA decomposition and further allows for modeling subjects' covariate effects in ICA (Guo and Tang, 2013; Shi and Guo, 2016; Lukemire et al., 2018). All the aforementioned ICA methods are developed for cross-sectional imaging studies where subjects are only scanned once during the study.

In recent years, longitudinal studies have become increasingly popular in the neuroscience community. In such studies, brain imaging such as fMRI scans from the same individual are acquired repeatedly at multiple time points including the baseline as well as follow-up visit times. Within-subject changes in brain images across different time points provides great insights into effects and causal relationships in investigating changes in brain networks related to disease progression, treatment or neurodevelopment. By taking the advantage of using each subject as his/her own control, longitudinal studies are well-known to have the potentials to provide more reliable and significant scientific findings than cross-sectional studies. Existing longitudinal imaging analysis often focus on modeling fMRI brain activation or structural MRI volumetric measures across time (Calhoun, Adali, McGinty, Pekar, Watson and Pearlson, 2001; Dettwiler et al., 2014; Lee et al., 2015). There has also been some work on longitudinal analysis of brain connectivity, which mainly involve modeling pairwise connectivity measures or network summary measures from a per-specified

network structure (Dai et al., 2017; Wu, Taki, Sato, Qi, Kawashima and Fukuda, 2013; Li et al., 2009). However, methods are lacking for conducting longitudinal ICA that jointly decompose the subjects' repeatedly measured fMRI data, extract the underlying brain functional networks and studying the longitudinal effects on brain networks.

Existing group ICA methods are not suitable for modeling repeated measured images in longitudinal studies. There are only a couple of ad-hoc strategies for longitudinal ICA decomposition. The first approach is to conduct ICA separately at each time point and then take the ICs extracted from different time points for secondary longitudinal analysis. This separate analysis approach has limited capacity to evaluate changes in functional networks across time because 1) independent components do not have a natural order, it is difficult to identify matching components across different time points, especially in resting-state fMRI. 2) ICA algorithms usually have random elements in that they may find different local minima across different runs (Himberg et al., 2004). This reduces the comparability of the ICs extracted separately at each visit. Another major drawback of the approach is that it ignores within-subject correlations among repeatedly measured data, which results in considerable loss of statistical power in testing covariate effects. The second ad-hoc approach is to adopt the TC-GICA framework by stacking all subjects' repeatedly measured images into a single group data matrix and performing ICA decomposition to extract common group spatial source signals. Then, subject/visit-specific IC maps are reconstructed via post-ICA analysis such as the dual regression. The longitudinal effects are then evaluated based on the reconstructed subject/visit-specific ICs. The limitations of the TC-GICA approach are that it ignores the between-subject variability in the ICA decomposition, does not take into account the random variabilities introduced in reconstructing subject/visit-specific IC maps and does not account for within-subject correlations among repeated scans in ICA decomposition. These limitations lead to

loss of accuracy and efficiency in estimating and testing covariate effects on brain networks in longitudinal studies.

In this topic, we propose a longitudinal ICA (L-ICA) model that incorporates subject-level random effects and the time-dependent covariate effects in ICA decomposition to investigate temporal changes in brain networks and their associations with subjects clinical or demographic covariates. The L-ICA is a hierarchical model where the first-level of L-ICA decomposes a subject's fMRI data obtained at a visit into a linear mixture of subject/visit-specific spatial source signals or ICs, and these ICs are then modeled at the second-level of L-ICA in terms of population-level baseline source signals, visit effects, covariate effects, subject-specific random effects and subject/visit-specific random variability. To the best of our knowledge, L-ICA is the first model-based extension of ICA for longitudinal imaging analysis. L-ICA is able to account for within-subject correlations among repeated scans, provide more accurate estimates of changes in brain functional networks on both the population- and individual-level, and increase statistical power in detecting covariate effects on networks. Furthermore, L-ICA provides model-based prediction for changes in brain networks related to disease progression, treatment or neurodevelopment.

For model estimation, we propose an exact EM algorithm which is fully traceable and simultaneously provides the estimation on population-level spatial maps and subject/visit-specific ICs. Furthermore, we propose a subspace-based approximate EM algorithm to provide more efficient computation. Results from the simulation studies and real data analysis show that the approximate EM algorithm significantly reduces the computation time while maintaining high estimation accuracy comparable to the exact EM. Moreover, we develop a statistical inference procedure for testing covariate effects in L-ICA, which demonstrates lower type I error and higher statistical power than the existing testing method based on TC-GICA. We apply the L-ICA method to investigating changes in functional networks in ADNI2 longi-

tudinal rs-fMRI study. Results from L-ICA showed differential temporal changing patterns between Alzheimer and control groups in relevant brain networks, which is not revealed by existing ICA methods.

This chapter is organized as follows. The methodology of L-ICA is presented in the section 2.2 which includes the L-ICA model specification, estimation via the exact EM algorithm and the approximate EM algorithm, and the inference procedure. In the section 2.3, results from the simulation study are presented. Section 2.4 is the real data application of ANDI2 study. Conclusion and discussion are in section 2.5.

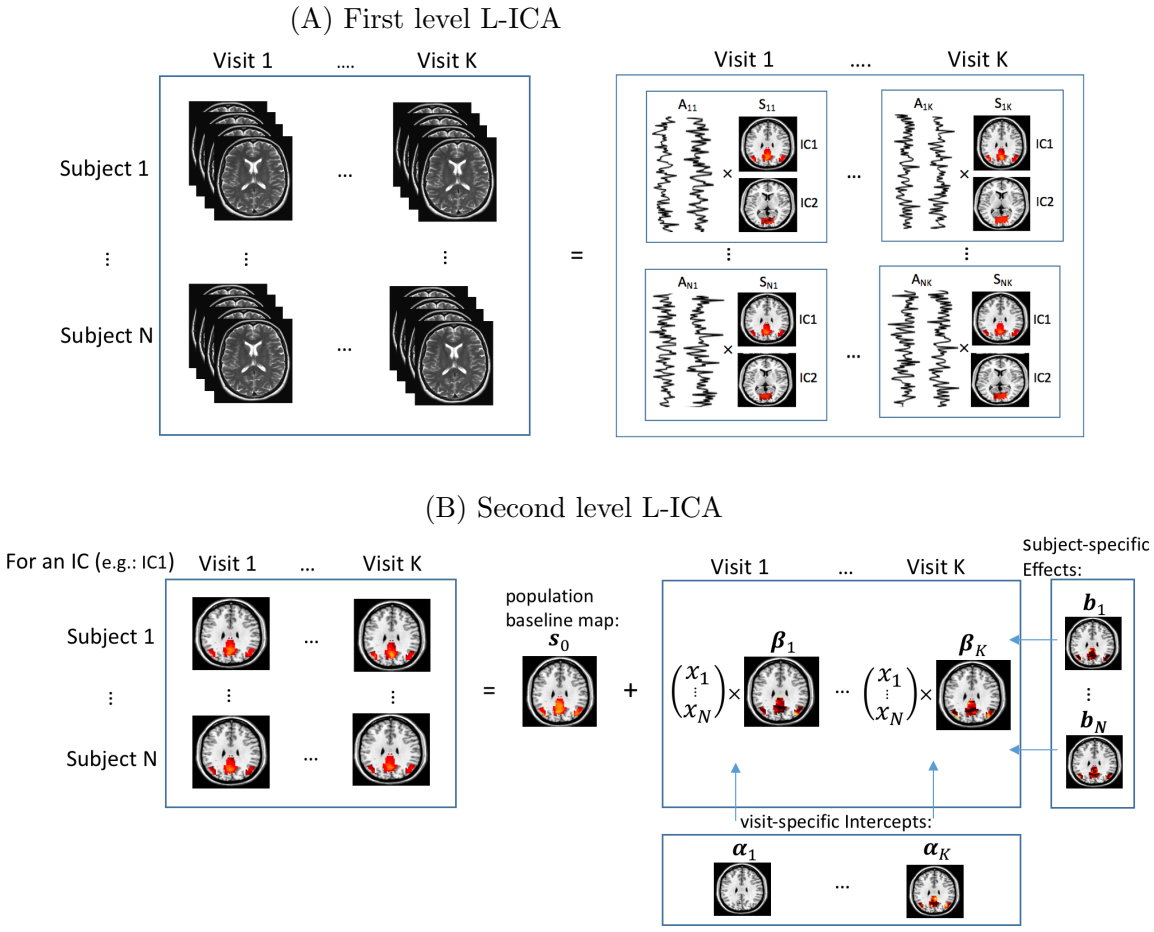

Figure 2.1: Schematic illustration of the hierarchical modeling framework of L-ICA. (A) the first level model of L-ICA with N subjects and K visits where each subject/visit-specific fMRI data is decomposed into q subject/visit-specific ICs, here $q = 2$ for illustration purpose. (B) the second level model of L-ICA for one specific IC where the subject/visit-specific ICs are modelled in terms of population-level source signals, subject specific random effects, visit effects and visit-specific covariate effects.

2.2 Methods

This section introduces the L-ICA framework, which includes the model specification, EM algorithms and the inference procedure. To set the notation, suppose that in a longitudinal fMRI study, there are N subjects and each of them has K visits during the study. At each visit, a series of T fMRI scans are acquired where each scan represents a 3D brain image containing V voxels. Let $\widetilde{\boldsymbol{Y}}_{ij} = [\widetilde{\boldsymbol{y}}_{ij}(1), ..., \widetilde{\boldsymbol{y}}_{ij}(V)]$ be the $T \times V$ fMRI data matrix for subject i ($i = 1, \ldots, N$) at visit j ($j = 1, \ldots, K$) where $\widetilde{\boldsymbol{y}}_{ij}(v) \in \mathbb{R}^T$ represents the centered blood-oxygen-level dependent (BOLD) signal series at voxel v ($v = 1, \ldots, V$). Prior to ICA, some preprocessing steps such as centering, dimension reduction and whitening of the observed data are usually performed to facilitate the subsequent ICA decomposition (Hyvärinen et al., 2001). Following a PPCA-based preprocessing procedure similar to that used in previous work (Beckmann and Smith, 2004; Shi and Guo, 2016; Guo and Tang, 2013), we perform the dimension reduction and whitening procedure on $\widetilde{\boldsymbol{Y}}_{ij}$ to obtain a $q \times V$ preprocessed data matrix $\boldsymbol{Y}_{ij}$ for subject i at visit j, where q is the number of independent components. Throughout the rest of this chapter, we will present the L-ICA model and methodologies based on the preprocessed data.

2.2.1 Longitudinal ICA model (L-ICA)

In this section, we propose a longitudinal ICA (L-ICA) model to jointly decompose repeated measured fMRI data acquired across multiple visits. The L-ICA is developed under a hierarchical modeling framework. We present a schematic illustration of the L-ICA in Figure 2.1. The first level of L-ICA decomposes the subject/visit-specific fMRI data into a product of subject/visit-specific spatial source signals and temporal mixing matrix. This allows capturing the variabilities of the functional networks across subjects and across visits. We also include a noise term in the first level ICA

model to account for residual variabilities in the fMRI data that are not explained by the extracted ICs, which is known as probabilistic ICA (Beckmann and Smith, 2004). Specifically, the first level of L-ICA is as follows,

$$\text{Level 1: } \boldsymbol{y}_{ij}(v) = \boldsymbol{A}_{ij}\boldsymbol{s}_{ij}(v) + \boldsymbol{e}_{ij}(v), \tag{2.1}$$

where $\boldsymbol{s}_{ij}(v) = [s_{ij}^{(1)}(v), ..., s_{ij}^{(q)}(v)]'$ is a $q \times 1$ vector with $s_{ij}^{(\ell)}(v)$ $(\ell = 1, \ldots, q)$ representing the spatial source signal of the ℓth IC (i.e., brain functional network) at voxel v for subject i at visit j, $\boldsymbol{A}_{ij}$ is the $q \times q$ mixing matrix for subject i at visit j, which is commonly assumed to be orthogonal given that $\boldsymbol{y}_{ij}(v)$ is whitened (Hyvärinen and Oja, 2000). $\boldsymbol{e}_{ij}(v)$ is a $q \times 1$ vector that represents the noise in the subject's data and $\boldsymbol{e}_{ij}(v) \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{E}_v)$ for $v = 1, ..., V$. Prior to ICA, preliminary analysis such as prewhitening (Beckmann and Smith, 2004) can be performed to remove correlations in the noise term and to standardize the variability across voxels (More details about prewhitening can be found in Appendix A). Therefore, following previous work (Hyvärinen et al., 2001; Beckmann and Smith, 2004, 2005; Guo and Pagnoni, 2008; Guo, 2011), we assume that the covariance for the noise term is isotropic across voxels, i.e. $\boldsymbol{E}_v = \sigma_0^2 \boldsymbol{I}_q$.

At the second-level of L-ICA, we further model subject/visit-specific spatial source signals $\boldsymbol{s}_{ij}(v)$ as a combination of the population-level source signals, subject-specific random effects, visit-specific covariate effects and subject/visit-specific random variations. That is,

$$\text{Level 2: } \boldsymbol{s}_{ij}(v) = \boldsymbol{s}_0(v) + \boldsymbol{b}_i(v) + \boldsymbol{\alpha}_j(v) + \boldsymbol{\beta}_j(v)'\boldsymbol{x}_i + \boldsymbol{\gamma}_{ij}(v), \tag{2.2}$$

where $\boldsymbol{s}_0(v) = [s_{01}(v), ..., s_{0q}(v)]'$ is the population-level spatial source signals. The q elements of $\boldsymbol{s}_0(v)$ are assumed to be independent and non-Gaussian. $\boldsymbol{b}_i(v)$ is the $q \times 1$ subject-specific random effects for q ICs where $\boldsymbol{b}_i(v) \sim N(\boldsymbol{0}, \boldsymbol{D})$ with

$\boldsymbol{D} = \text{diag}(\nu_1^2, ..., \nu_q^2)$. The subject-specific random effects help capture the within-subject correlations among the scans repeated acquired on the same subject at different visits, (Verbeke, 1997; Cheng et al., 2014; Gao, Ombao and Gillen, 2017). $\boldsymbol{\alpha}_j(v)$ is a $q \times 1$ visit effects parameter representing the population-level changes in spatial source signals from baseline to the jth visit. $\boldsymbol{x}_i = [x_{i1}, ..., x_{ip}]'$ is the $p \times 1$ subject-specific covariate vector which may contain a subject's clinical and demographic information such as disease group, gender, age, etc. $\boldsymbol{\beta}_j(v)$ is a $p \times q$ parameters matrix reflecting how subjects' covariates $\boldsymbol{x}_i$ modulate the subject/visit-specific brain networks. Finally, $\boldsymbol{\gamma}_{ij}(v)$ is a $q \times 1$ zero-mean Gaussian random vector, i.e. $\boldsymbol{\gamma}_i(v) \overset{\text{iid}}{\sim} \text{N}(\boldsymbol{0}, \tau^2 \boldsymbol{I}_q)$, capturing the residual random variability among subject/visit-specific brain networks after adjusting for the other effects in the model. In the Level 2 model, by including the subject-specific random effects, L-ICA is able to borrow information among the multiple visits within the same subject to obtain more accurate estimate of unique patterns in brain networks specific to the individual. L-ICA incorporates the visit-specific covariate effects to allow flexibly modeling time-varying covariate effects on subjects' brain networks in a longitudinal study .

2.2.2 Source signal distribution model

We specify mixtures of Gaussians (MoG) as our source distribution model for the population-level spatial source signals, $\boldsymbol{s}_0(v)$. MoG has been selected as the distribution for independent components in quite a few ICA analysis (Attias, 2000; Guo, 2011; Guo and Tang, 2013; Shi and Guo, 2016) because it has several desirable properties for modeling fMRI signals. Within each brain functional network, only a small percentage of locations in the brain are activated or deactivated whereas most brain areas exhibit background fluctuations (Biswal and Ulmer, 1999). MoG are well suited to model such mixed patterns. Furthermore, MoG can capture various types of non-Gaussian signals (Xu et al., 1997; Gao, Shahbaba and Ombao, 2017; Kostantinos,

2000; Gao et al., 2018) and also offer tractable likelihood-based estimation (McLachlan and Peel, 2004).

Specifically, for $\ell = 1, \ldots, q$ we assume that the spatial source signal $s_{0\ell}(v)$ follows a MoG distribution, i.e.

$$s_{0\ell}(v) \sim \text{MoG}(\boldsymbol{\pi}_\ell, \boldsymbol{\mu}_\ell, \boldsymbol{\sigma}_\ell^2), \tag{2.3}$$

where $\boldsymbol{\pi}_\ell = [\pi_{\ell,1}, ..., \pi_{\ell,m}]'$ with $\sum_{j=1}^m \pi_{\ell,j} = 1$ is the weight parameters in MoG, $\boldsymbol{\mu}_\ell = [\mu_{\ell,1}, ..., \mu_{\ell,m}]'$ and $\boldsymbol{\sigma}_\ell^2 = [\sigma_{\ell,1}^2, ..., \sigma_{\ell,m}^2]'$ are the mean and variance parameters of the Gaussian component distributions in the MoG; m is the number of Gaussian components in MoG. The probability density function of $\text{MoG}(\boldsymbol{\pi}_\ell, \boldsymbol{\mu}_\ell, \boldsymbol{\sigma}_\ell^2)$ is $\sum_{j=1}^m \pi_{\ell,j} g(s_{0\ell}(v); \mu_{\ell,j}, \sigma_{\ell,j}^2)$ where $g(\cdot)$ is the pdf of the Gaussian distribution. In fMRI applications, mixtures of two to three Gaussian components can be used to capture the distribution of fMRI spatial signals, with the different Gaussian components representing the background fluctuation and the negative or positive fMRI BOLD effects respectively (Beckmann and Smith, 2004; Guo and Pagnoni, 2008; Guo, 2011; Wang et al., 2013; Guo and Tang, 2013). Without loss of generality, we denote the first Gaussian component, i.e. $j = 1$, to be the background fluctuation state throughout the rest of this chapter. To facilitate derivations with the MoG model, we introduce a voxel-specific latent state variable $z_\ell(v)$ which represents which Gaussian component in MoG that voxel v belongs to. Specifically, $z_\ell(v)$ takes a value in $\{1, \ldots, m\}$ with probability $p[z_\ell(v) = j] = \pi_{\ell,j}$ $(j = 1, .., m)$. When $z_\ell(v) = j$, the vth voxel follows the jth Gaussian component distribution in MoG, i.e. $p(s_{0\ell}(v)|z_\ell(v) = j) = g(s_{0\ell}(v); \mu_{\ell,j}, \sigma_{\ell,j}^2)$.

2.2.3 Maximum likelihood estimation and the EM algorithm

The parameters in L-ICA model is estimated via maximum likelihood (ML) approach. Based on the hierarchical models in (2.1) and assuming the independence among

voxels, (2.2) and (2.3), the complete data log-likelihood for L-ICA model is,

$$l(\Theta; \mathcal{Y}, \mathcal{X}, \mathcal{S}, \mathcal{B}, \mathcal{Z}) = \sum_{v=1}^{V} l_v(\Theta; \mathcal{Y}, \mathcal{X}, \mathcal{S}, \mathcal{B}, \mathcal{Z}), \tag{2.4}$$

where $\mathcal{Y} = \{\boldsymbol{y}_{ij}(v) : i = 1, ..., N; j = 1, ..., K; v = 1, \ldots, V\}$ are the preprocessed longitudinal fMRI data across subjects, $\mathcal{X} = \{\boldsymbol{x}_i : i = 1, ..., N\}$ are subjects' covariates, $\mathcal{S} = \{\boldsymbol{s}_0(v), \boldsymbol{s}_{ij}(v) : i = 1, ..., N; j = 1, ..., T; v = 1, ..., V\}$ are the latent independent component spatial source signals, $\mathcal{B} = \{\boldsymbol{b}_i(v) : i = 1, ..., N; v = 1, ..., V\}$ are the subject-specific random effects and $\mathcal{Z} = \{\boldsymbol{z}(v) : v = 1, ..., V\}$ are the latent states for MoG source distribution model; the parameters in L-ICA are denoted by $\Theta = \{\{\boldsymbol{\alpha}_j(v)\}, \{\boldsymbol{\beta}_j(v)\}, \{\boldsymbol{A}_{ij}\}, \boldsymbol{E}, \boldsymbol{D}, \tau, \{\boldsymbol{\pi}_\ell\}, \{\boldsymbol{\mu}_\ell\}, \{\boldsymbol{\sigma}_\ell^2\} : i = 1, ..., N, j = 1, ..., K,$ $v = 1, ..., V, \ell = 1, ..., m\}$.

Since our likelihood function involves unobserved latent variables, we consider the expectation-maximization (EM) framework (Dempster et al., 1977) for finding the maximum likelihood estimates of parameters. The EM algorithm is an iterative algorithm that alternates between performing an expectation step (E-step) and a maximization step (M-step). In the E-step, we compute an expectation of the log-likelihood conditioning on the distribution of latent variables given the observed data $\mathcal{Y}$ and the current parameter estimates $\hat{\Theta}^{(k)}$. At the M-step, the updated maximum likelihood estimates of the parameters is computed by maximizing the expected log-likelihood found on the E-step. The parameter estimates found on the M-step are then used to begin another E-step, and the process is iterated until convergence, i.e. until the parameter estimates $\hat{\Theta}^{(k)}$ and $\hat{\Theta}^{(k+1)}$ in two consecutive iterations are considered sufficiently close. In the following, we present two EM algorithms for solving the L-ICA model. The first is an exact EM method that provides exact evaluation of the conditional expectation in the E-step. We then propose an approximation EM algorithm is computationally more efficient especially with large number of ICs.

2.2.3.1 The exact EM algorithm

We first develop an exact EM which has an explicit E-step and M-step to obtain ML estimates for the parameters in L-ICA.

E-step: In the E-step, given the estimated parameter $\hat{\Theta}^{(k)}$ from the last step, we evaluate the conditional expectation of the complete data log-likelihood as follows,

$$Q(\Theta|\hat{\Theta}^{(k)}) = \sum_{v=1}^{V} E_{\boldsymbol{L}(v)|\mathbf{y}(v),\hat{\Theta}^{(k)}} \left[l(\Theta; \mathcal{Y}, \mathcal{X}, \mathcal{S}, \mathcal{B}, \mathcal{Z}) \right], \tag{2.5}$$

where $\boldsymbol{L}(v) = [\boldsymbol{b}_1(v)', ..., \boldsymbol{b}_N(v)', \boldsymbol{s}_0(v)', \boldsymbol{s}_{11}(v)', ..., \boldsymbol{s}_{NK}(v)']'$ are the latent variables in L-ICA model which include the latent source signals on both the population and individual level and the subject-specific random effects. To calculate the conditional expectation, we need to derive the conditional distribution of $\boldsymbol{L}(v)$ given the observed data $\mathbf{y}(v)$, i.e. $p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \hat{\Theta}^{(k)})$. To facilitate this derivation, we take the following steps. First, we derive the distribution of $\boldsymbol{L}(v)$ given both the observed data $\mathbf{y}(v)$ and the latent states $\boldsymbol{z}(v)$, i.e. $p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \boldsymbol{z}(v), \hat{\Theta}^{(k)})$, which can be shown to be a multivariate Gaussian distribution. Next, we derive the conditional distribution of the latent states given the observed data, i.e. $p[\boldsymbol{z}(v) \mid \mathbf{y}(v), \hat{\Theta}^{(k)}]$, by applying the Bayes' Theorem. Finally, we obtain the conditional distribution of $\boldsymbol{L}(v)$ given $\mathbf{y}(v)$ by integrating out $\boldsymbol{z}(v)$, i.e.

$$p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \hat{\Theta}^{(k)}) = \sum_{\boldsymbol{z}(v) \in \mathcal{R}} p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \boldsymbol{z}(v), \hat{\Theta}^{(k)}) p[\boldsymbol{z}(v) \mid \mathbf{y}(v); \hat{\Theta}^{(k)}],$$

where $\mathcal{R}$ represents the set of all possible values of $\boldsymbol{z}(v)$, i.e., $\mathcal{R} = \{\boldsymbol{z}^r\}_{r=1}^{m^q}$ where $\boldsymbol{z}^r = [z_1^r, ..., z_q^r]'$ and $z_\ell^r \in \{1, ..., m\}$ for $\ell = 1, ...q$.

Following this procedure, we can derive explicit form for the conditional distribution for the latent variables and subsequently deriving the conditional expectation $Q(\Theta|\hat{\Theta}^{(k)})$ in (2.5).

M-step: In the M-step, the updated estimates are obtained by maximizing the expected log-likelihood function computed in the E-step, i.e.,

$$\hat{\Theta}^{(k+1)} = \underset{\Theta}{\operatorname{argmax}}\, Q(\Theta|\hat{\Theta}^{(k)}). \tag{2.6}$$

We have derived explicit solutions for all parameter updates (please see Appendix A for details).

The steps of the exact EM algorithm is summarized in Algorithm 1. The detailed derivations are presented in the Appendix A.

Algorithm 1 The Exact EM Algorithm

Initial values: Obtain an initial values $\hat{\Theta}^{(0)}$ based on existing group ICA software.
repeat
E-step:
1. Evaluate the conditional distribution of the latent variables $p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \hat{\Theta}^{(k)})$ using the proposed three-step approach:
1.a Evaluate the multivariate Gaussian $p[\boldsymbol{L}(v) \mid \mathbf{y}(v), \boldsymbol{z}(v), \hat{\Theta}^{(k)}]$;
1.b Evaluate $p[\boldsymbol{z}(v) \mid \mathbf{y}(v); \hat{\Theta}^{(k)}]$ via Bayes' Theorem
1.c integrate out the latent states $\boldsymbol{z}(v)$

$$p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \hat{\Theta}^{(k)}) = \sum_{\boldsymbol{z}(v)\in\mathcal{R}} p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \boldsymbol{z}(v), \hat{\Theta}^{(k)}) p[\boldsymbol{z}(v) \mid \mathbf{y}(v); \hat{\Theta}^{(k)}]$$

2. Estimate conditional expectation $Q(\Theta|\hat{\Theta}^{(k)})$ based on $p(\boldsymbol{L}(v) \mid \mathbf{y}(v), \hat{\Theta}^{(k)})$.
M-step:
Update parameters estimates

$$\hat{\Theta}^{(k+1)} = \underset{\Theta}{\operatorname{argmax}}\, Q(\Theta|\hat{\Theta}^{(k)}).$$

until convergence, i.e. $\frac{\|\hat{\Theta}^{(k+1)} - \hat{\Theta}^{(k)}\|}{\|\hat{\Theta}^{(k)}\|} < \epsilon$

After obtaining the ML estimates $\hat{\Theta}$, we estimate the baseline population- and subject/visit-specific source signals as well as their variability based on the mean and variance of their conditional distributions, i.e., $[\boldsymbol{s}_0(v) \mid \mathbf{y}(v); \hat{\Theta}]$ and $[\boldsymbol{s}_{ij}(v) \mid \mathbf{y}(v); \hat{\Theta}]$. These conditional moments are directly obtainable from the E-step of our algorithm upon convergence and no separate post-ICA steps are required. Based on

the estimated covariate effects $\{\hat{\boldsymbol{\beta}}(v)\}$, we can investigate how subjects' clinical and demographic characteristics affects their brain functional networks and their changes across visits. Furthermore, the L-ICA also provides model-based prediction of the brain functional networks for specific sub-populations at a given visit. For example, for a sub-population characterized by a covariates pattern $\boldsymbol{x}^*$, the predicted brain functional networks at the jth visit can be derived by plugging the ML parameter estimates into Level 2 of L-ICA, i.e.

$$\hat{\boldsymbol{s}}_j(v) = \hat{\boldsymbol{s}}_0(v) + \hat{\boldsymbol{\alpha}}_j(v) + \hat{\boldsymbol{\beta}}_j(v)'\boldsymbol{x}^*, \tag{2.7}$$

2.2.4 Subspace approximate EM algorithm

The exact EM algorithm requires $\mathcal{O}(m^q)$ operations at each voxel which is an exponential increase with regard to the number of the ICs extracted in L-ICA, which will be time consuming when q is large. The reason for needing $\mathcal{O}(m^q)$ operations is that, the exact EM evaluates the conditional distribution of the latent states $\boldsymbol{z}(v)$, i.e. $p[\boldsymbol{z}(v) \mid \mathbf{y}(v)]$, across the whole sample space $\mathcal{R}$ of $\boldsymbol{z}(v)$, which has a cardinality of m^q. To reduce the computation load, we develop a subspace-based approximate EM for L-ICA model. The motivation of the subspace EM is based on the observation from fMRI analysis that the density of $p[\boldsymbol{z}(v) \mid \mathbf{y}(v)]$ is mostly concentrated on a subspace $\mathcal{R}_s = \{\boldsymbol{z}^r \in \mathcal{R},\, s.t. \sum_\ell I(z_\ell^r \neq 1) \leq 1\}$. To help understand this subspace, recall that the latent state z_ℓ^r takes values in $(1, \ldots, m)$ with the first state, i.e. $z_\ell^r = 1$, corresponding to the background fluctuation while other states, i.e. $z_\ell^r \neq 1$, corresponding to either positive or negative signals at a voxel. Therefore, the subspace $\mathcal{R}_s$ corresponds to that a voxel has active signals in at most one of the q ICs. This approximation is reasonable when the source signals are sparse across ICs, i.e. $p(z_\ell \neq 1) \approx 0$ for $\ell = 1, \ldots, q$. Because given the statistical independence of the ICs, $p(z_{\ell^*} \neq 1 | z_\ell \neq 1) = p(z_{\ell^*} \neq 1) \approx 0$. That is given a voxel is activated in the ℓth IC,

the probability for it to be also activated in another IC ℓ^* is close to zero. In Shi and Guo (2016), we have provided theoretical proof that the density of the conditional distribution of the latent states is mostly concentrated in the subspace $\mathcal{R}_s$ when the source signals are sparse in each IC, which is the case with the fMRI spatial source signals which have been shown to be sparse across the brain for each network (Mckeown et al., 1998; Daubechies et al., 2009). It is noteworthy to mention that there are some network hubs in the brain that are active in multiple networks. The proposed subspace EM is still able to recover overlapping spatial signals across the ICs, hence capable of identifying brain regions that are involved in multiple functional networks (Shi and Guo, 2016). The subspace approximation only results in small attenuation on the estimated source signals in the overlapping region .

In the subspace EM algorithm, we follow the similar steps as in the exact EM algorithm presented in Algorithm 1. The main difference is that when evaluating and summing across the latent states $\boldsymbol{z}(v)$ in the E-step and M-step, we replace the whole sample space $\mathcal{R}$ with the proposed subspace $\mathcal{R}_s$ which only has carnality of $(m-1)q+1$. This means the subspace EM only requires $\mathcal{O}(mq)$ operations at each voxel which scales linearly with the number of ICs and is significantly faster than the exponential growth of the exact EM algorithm.

2.2.5 Statistical inference for testing covariate effects in L-ICA

In this session, we propose a statistical inference procedure for testing covariate effects in L-ICA to investigate whether the covariates have significant effects on brain functional networks and their changes across visits. Typically, statistical inference in maximum likelihood estimation is conducted by inverting the information matrix to estimate the variance-covariance matrix of ML estimates of the parameters. However, this standard approach is not feasible when modeling fMRI data with L-ICA because

the high dimensionality of the parameter space makes extremely challenge to obtain a reliable inversion of information matrix. To address this issue, we develop a computational efficient statistical inference procedure based on the connection between the L-ICA and multivariate linear models. The proposed inference procedure provides an efficient approach to estimate the variance-covariance matrix of the time-specific covariate effects at each voxel by directly using the output from our EM algorithms.

Specifically, let $\boldsymbol{y}_i(v)$ be the ith subjects longitudinal fMRI data which is a $qK \times 1$ vector obtained by stacking his/her data across visits, i.e. $\boldsymbol{y}_i(v) = [\boldsymbol{y}_{i1}(v)', ..., \boldsymbol{y}_{iK}(v)']'$. By collapsing the hierarchical models, we rewrite the L-ICA model in a non-hierarchical form which is similar to classical multivariate linear model, i.e.,

$$\boldsymbol{y}_i^*(v) = \boldsymbol{X}_i^* \boldsymbol{C}^*(v) + \boldsymbol{\zeta}_i(v), \tag{2.8}$$

where $\boldsymbol{y}_i^*(v) = \boldsymbol{A}_i' \boldsymbol{y}_i(v)$ is the response vector, $\boldsymbol{X}_i^*$ is the design matrix which includes the visit time and the covariates in L-ICA, $\boldsymbol{C}^*(v)$ is the parameter matrix which includes the effects parameters in L-ICA such as the visit effects $\boldsymbol{\alpha}$ and covariate effects $\boldsymbol{\beta}$, $\boldsymbol{\zeta}_i(v)$ is the zero-mean Gaussian random variation term which includes the subject-specific random effects and noise terms in L-ICA. Please see the Appendix A for details.

The model in (2.8) can be viewed a multivariate linear model. Based on linear model theory, a variance estimator for parameter estimates $\hat{\boldsymbol{C}}^*(v)$ can be derived as follows,

$$\mathrm{Var}[\hat{\boldsymbol{C}}^*(v)] = \left(\sum_{i=1}^{N} \boldsymbol{X}_i^{*,T} \boldsymbol{W}(v)^{-1} \boldsymbol{X}_i^* \right)^{-1}. \tag{2.9}$$

where $\boldsymbol{W}(v) = \mathrm{Var}(\boldsymbol{\zeta}_i(v))$ and can be estimated by plugging ML estimates obtained from the EM algorithm.

After deriving the variance estimator for the ML estimates of the parameters

in L-ICA, We can then conduct hypothesis testing on the covariate effects the brain networks and their changes across visits. Specifically, we first formulate the hypothesis in terms of linear combinations of the parameters in the L-ICA model, i.e. $H_0 : \boldsymbol{l}'\boldsymbol{C}^*(v) = \mathbf{0}$ vs. $H_1 : \boldsymbol{l}'\boldsymbol{C}^*(v) \neq \mathbf{0}$ where $\boldsymbol{l}$ is a vector of constant coefficients specified based on the hypothesis that we are testing on. We can then construct the test statistic as,

$$z(v) = \frac{\boldsymbol{l}'\hat{\boldsymbol{C}}^*(v)}{\sqrt{\boldsymbol{l}'\hat{\text{Var}}[\hat{\boldsymbol{C}}^*(v)]\boldsymbol{l}}}, \tag{2.10}$$

the test statistic $z(v)$ will then be compared against its null distribution to derive the p-value for testing the significance of the covariate effects at voxel v. Standard multiple testing correction procedures can be applied to control for family wise error rate (FWER) or the false discovery rate (FDR) when testing the covariate effects across voxels, (Genovese et al., 2002; Chumbley and Friston, 2009; Storey, 2011; Wang, Wu and Yu, 2017).

2.3 Simulation Study

We conducted three types of simulation studies to 1) evaluate the performance of the proposed L-ICA model as compared with the approach based on the existing TC-GICA framework, 2) to evaluate the performance of the proposed inference method for testing covariate effects on brain networks, and 3) to evaluate the performance of the proposed subspace-based EM algorithm as compared with the exact EM algorithm.

2.3.1 Simulation study I: performance of the L-ICA v.s. TC-GICA-based longitudinal analysis

In this simulation study, we evaluate the performance of the proposed L-ICA model versus the TC-GICA based approach for analyzing longitudinal fMRI. In the simulation, we considered three different sample sizes $N = 10, 20, 60$ and each subject has

three visits: baseline, visit 1 and visit 2 ($K = 3$). The simulated fMRI data were generated from 3 underlying ICs or source signals, i.e., $q = 3$, (see Figure 2.2 (A)). For each IC, we generated the source signals $\{\boldsymbol{s}_0(v)\}$ as a 3D spatial map with the dimension of $53 \times 63 \times 3$, which was based on three selected slices from a real fMRI imaging data. The source intensity at the activated region in the IC maps was generated from a Gaussian distribution with the mean of 4. The visit specific intercepts, i.e., $\boldsymbol{\alpha}_2(v)$ and $\boldsymbol{\alpha}_3(v)$, are set to be 2 and 3 respectively for the voxels within the activated IC regions and 0 for other voxels. We then generated a binary covariate for each subject as $x_i \overset{\text{iid}}{\sim} \text{Bernoulli}(0.5)$. The covariate effects at the jth visit, $\boldsymbol{\beta}_j(v)$, were specified using a 2D Gaussian process within the IC regions where the mean level of the covariate effects increased across the 3 visits. Additionally, we generated subject-specific random effects, i.e., $\boldsymbol{b}_i(v)$, from a zero-mean Gaussian distribution with the covariance matrix of $\boldsymbol{D} = \text{diag}(1.0^2, 1.1^2, 1.2^2)$. For the residual subject/visit-specific variability, i.e., $\boldsymbol{\gamma}_i(v)$, we considered two levels of variability: low ($\tau^2 = 0.5$) and high ($\tau^2 = 4$). The time series associated with each IC was generated from real fMRI time courses with the length of $T = 200$ and hence represented realistic fMRI temporal dynamics. We generated subject/visit-specific time sources that had similar frequency features but different phase patterns (Guo, 2011; Shi and Guo, 2016), which mimic temporal dynamics in resting-state fMRI. After simulating the spatial source signal and the temporal mixing matrices for the ICs, Gaussian background noise with a standard deviation of 1 (i.e. $\boldsymbol{E} = \boldsymbol{I}_q$) were added to generate observed fMRI data.

Following previous work (Beckmann and Smith, 2005; Guo and Pagnoni, 2008; Guo, 2011), we evaluate the performance of each method based on the correlations between the true ICs and estimated ICs in both temporal and spatial domains. We report the estimation accuracy for both the population-level as well as the subject/visit-specific source signals. To compare the performance in estimating the covariate effects, we report the mean square errors (MSEs) of $\hat{\boldsymbol{\beta}}(v)$ defined

by $\frac{1}{KV}\sum_{j=1}^{K}\sum_{v=1}^{V}\left\|\hat{\boldsymbol{\beta}}_j(v)-\boldsymbol{\beta}_j(v)\right\|_{\mathcal{F}}^2$ averaged across simulation runs. Here $\|\cdot\|_{\mathcal{F}}$ is the Frobenius norm for a matrix. Since ICA recovery is permutation invariant, the estimated ICs were matched to the true IC with which it has the highest spatial correlation. We present the simulation results in Table 2.1. The results show that L-ICA provides more accurate estimates for the source signals on both the population- and subject/visit-level, by demonstrating higher correlation with the true source signals. L-ICA also provides more accurate estimation of the covariate effects with smaller mean square errors (MSE). Moreover, compared with the TC-GICA, the L-ICA estimates of the source signals and covariate effects are more stable with consistently smaller standard deviations (SD) across simulation runs.

We also display the estimated population-level IC maps at baseline and the last visit , i.e. visit 2, based on both methods in Figure 2.2. The L-ICA shows better accuracy in recovering the true activation patterns in the ICs at both visits. The intensity of the source signals in the activated regions in each IC increases from baseline to the last visit in true IC maps. This increase in intensity is well captured by the L-ICA estimated IC maps but not obvious in the TC-GICA estimated IC maps. Furthermore, the estimated IC maps from the TC-GICA approach show "cross-talk" between the ICs. In Figure 2.2, we also present the true and estimated longitudinal trends of source signals for activated voxels in an IC. The L-ICA shows better performance than the TC-GICA approach in recovering the temporal changing patterns across voxels.

2.3.2 Simulation study II: performance of the proposed inference procedure for testing covariate effects

In this simulation study, we evaluate the performance of the methods in testing covariates effects on ICs. We simulated fMRI datasets with two source signals ($q = 2$), two visits ($K = 2$), one binary covariate and the sample size of $N = 40$. Since

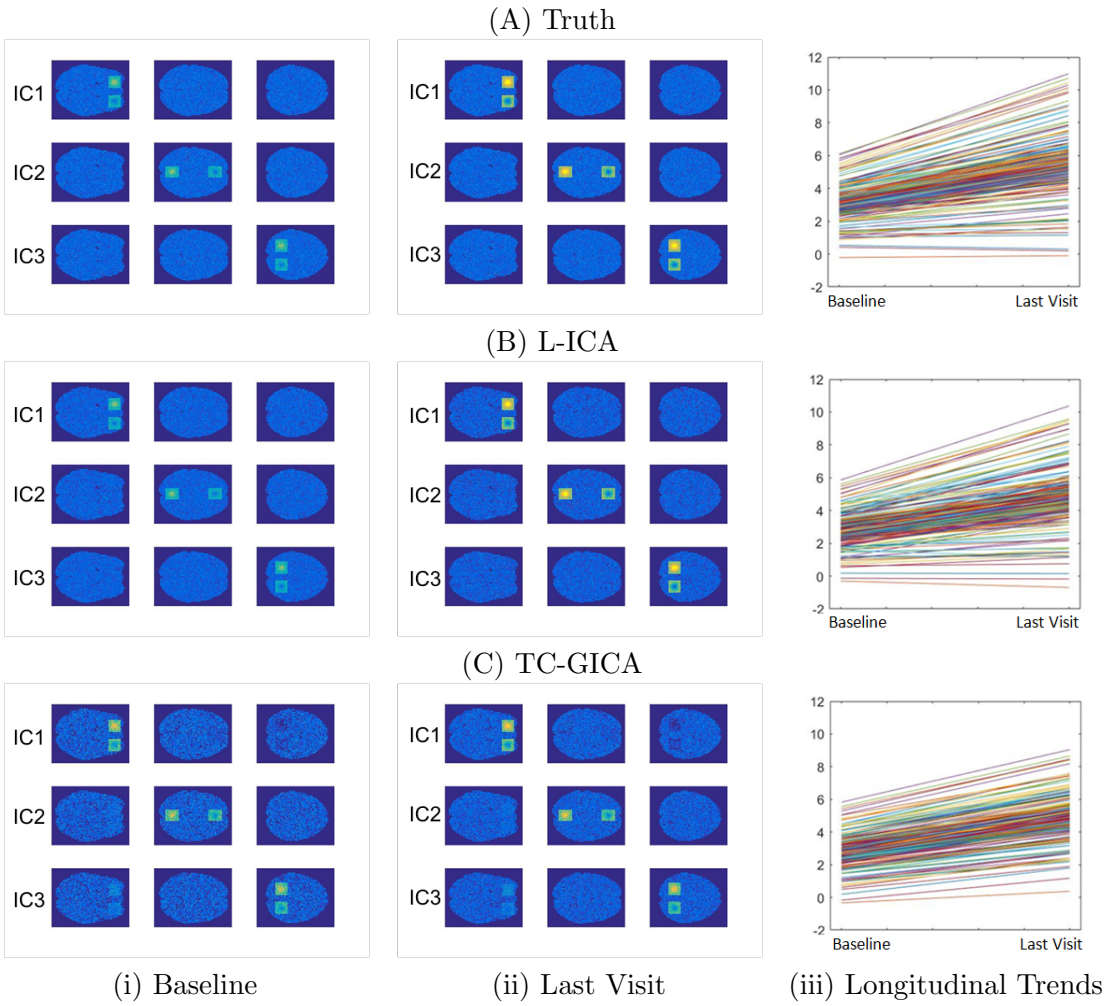

Figure 2.2: Comparison between the proposed L-ICA and the TC-GICA based approach for estimating the population-level IC maps at baseline and the last visit (N=20, low subject/visit-specific random variability): (A) truth, (B) L-ICA estimates and (C) estimates from TC-GICA. Column (i) represents the IC maps at baseline ; Column (ii) represents the IC maps at last visit; Column (iii) represents the longitudinal trends for activated voxels (where each line represents a voxel) in the first IC (IC1). Results show that L-ICA provides more accurate estimates than TC-GICA at each visit and more precisely captures the voxel-specific longitudinal trend.

Table 2.1: Simulation results for comparing L-ICA method against TC-GICA-based method with 100 simulation runs. Values presented are mean and standard deviation of correlations between the true and estimated: population-level spatial maps, subject/visit-specific spatial maps and subject/visit-specific time courses. The mean and standard deviation of the MSE of the covariate effects estimates are also provided.

Subj-Visit Var	Population-level spatial maps Corr.(SD)		Subject/Vist-specific spatial maps Corr.(SD)	
	L-ICA	TC-GICA	L-ICA	TC-GICA
Low				
N=10	0.929 (0.021)	0.853 (0.116)	0.979 (0.016)	0.942 (0.095)
N=20	0.959 (0.015)	0.889 (0.113)	0.981 (0.012)	0.937 (0.093)
N=60	0.984 (0.008)	0.940 (0.109)	0.999 (0.007)	0.951 (0.085)
High				
N=10	0.886 (0.053)	0.621 (0.213)	0.960 (0.044)	0.845 (0.152)
N=20	0.899 (0.042)	0.691 (0.187)	0.962 (0.034)	0.854 (0.141)
N=60	0.958 (0.011)	0.856 (0.162)	0.991 (0.019)	0.900 (0.099)
Subj-Visit Var.	Subject/Vist-specific time courses Corr.(SD)		Covariate Effects MSE(SD)	
	L-ICA	TC-GICA	L-ICA	TC-GICA
Low				
N=10	0.997 (0.004)	0.941 (0.076)	0.152 (0.009)	0.159 (0.068)
N=20	0.998 (0.003)	0.942 (0.075)	0.093 (0.006)	0.153 (0.063)
N=60	1.000 (0.001)	0.957 (0.063)	0.040 (0.000)	0.128 (0.039)
High				
N=10	0.987 (0.019)	0.884 (0.092)	0.253 (0.015)	0.273 (0.101)
N=20	0.990 (0.014)	0.885 (0.093)	0.187 (0.011)	0.239 (0.086)
N=60	0.992 (0.007)	0.910 (0.077)	0.098 (0.004)	0.192 (0.083)

we need a large number of simulation runs to estimate the type I error and power in the test, we generated source signal images with the dimension of 20×20 to facilitate computation. The covariate effects at baseline $\beta_1(v)$ are set to be 0 representing no difference at baseline and visit-specific covariate effects $\beta_2(v)$ took values in $\{0, 0.375, 0.5, 0.625, 0.75, 0.875, 1, 1.125, 1.25\}$ for the IC region and are set to 0 for background region.

We applied L-ICA method and TC-GICA method to the simulated datasets and

tested for covariate effects using both methods. We considered two type of hypothesis tests. The first one aims to test whether the covariate has an effect on the network source signals at a given visit, where the hypotheses are $H_0 : \beta_2(v) = 0$ versus $H_1 : \beta_2(v) \neq 0$ for the given IC. In the second test, we assess the whether the covariate's effect on the network vary across visits, or equivalently whether the covariate affect the longitudinal changes in the network across visits, where the hypotheses are $H_0 : \beta_1(v) = \beta_2(v)$ versus $H_0 : \beta_1(v) \neq \beta_2(v)$. These two type of tests are the most commonly conducted in longitudinal studies. For L-ICA, hypothesis tests were conducted using the test proposed in section 2.2.5. For TC-GICA based approach, covariate effects were tested by performing post-ICA longitudinal analysis of the dual-regression reconstructed subject/visit-specific IC maps. We estimated the Type-I error rate with the empirical probabilities of not rejecting H_0 at voxels where H_0 is true. We estimated the power of the tests with the empirical probabilities of rejecting H_0 at voxels where H_1 is true.

We report the Type-I error rates and the statistical power for detecting covariate effects based on 1000 simulation runs in Figure 2.3. The panel (A) in Figure 2.3 presents the Type I error rates where the diagonal line represents the nominal level for the type I error corresponding to various significance levels. The proposed L-ICA test demonstrates lower type-I error rates which are closer to the nominal level as compared with the TC-GICA method. For the power analysis presented in panel (B), the L-ICA have much higher statistical power in detecting covariate effects than the TC-GICA method. Overall, these results indicate that L-ICA provides more reliable and powerful statistical tests for assessing covariate effects on the functional networks.

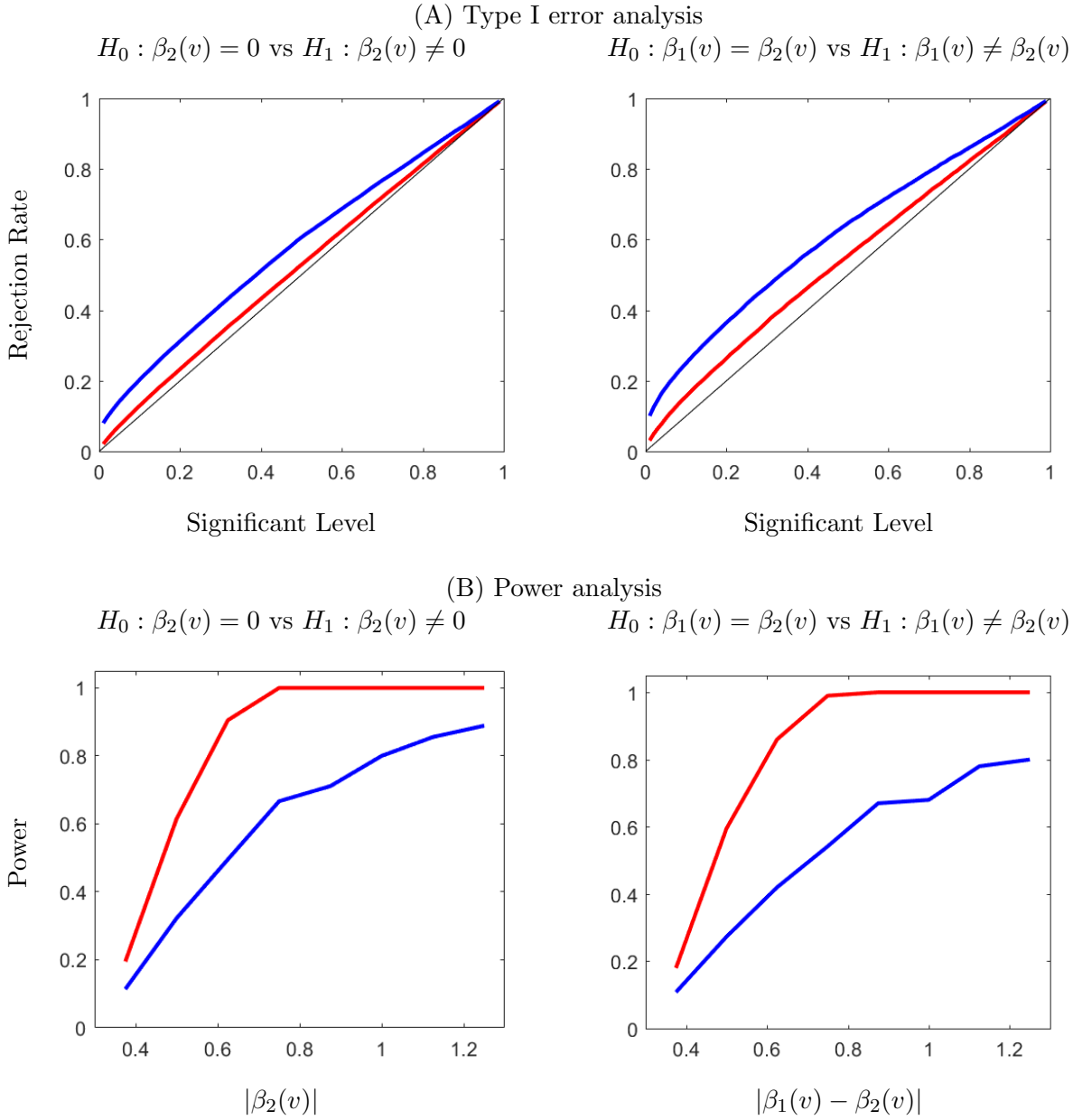

Figure 2.3: Simulation results for testing covariate effects based on 1000 runs with sample size $N = 40$ using the proposed L-ICA method (red) and the TC-GICA (blue) based method. We considered two types of hypothesis tests: testing the time-specific covariate effect at a given visit (the 2nd visit), i.e. $H_0 : \beta_2(v) = 0$ (the left column), and testing the time-varying longitudinal covariate effects between the 1st and 2nd visit, i.e. $H_0 : \beta_1(v) = \beta_2(v)$ (the right column). Panel (A) and (B) presents the type I error rates and the statistical power, respectively. The results show that the L-ICA method demonstrates lower type I error and higher statistical power as compared with the TC-GICA based method.

2.3.3 Simulation study III: performance of the subspace EM algorithm for LICA

In this section, we examined the performance of the subspace approximate EM algorithm as compared with the exact EM algorithm for the L-ICA model. We simulated data for ten subjects and considered three different number of ICs: $q = 3, 5, 10$. We summarize the results based on the two EM algorithms in Table 2.2. Results show that the accuracy of the subspace EM is comparable to that of the exact EM. The major advantage of the subspace EM is that it was much faster than the exact EM. This advantage becomes more clear with the increase of the number of ICs. For $q = 10$, the subspace-based EM only uses about 2% computation time of the exact EM.

Table 2.2: Simulation results for comparing subspace EM against exact EM based on 50 simulation runs. Values presented are mean and standard deviation of the computational/iteration time (in second), the mean and standard deviation of correlations between the true and estimated: baseline population-level spatial maps and subject/visit-specific time courses, the mean and standard deviation of the MSE of the covariates estimates. The stopping criteria is based on the correlation between true and estimated subject/visit-specific spatial maps to reach 0.99 for $q = 3, 5$ and 0.90 for $q = 10$.

	Iteration time (SD)		Baseline population-level spatial maps Corr.(SD)	
# of IC	Exact EM	Subspace EM	Exact EM	Subspace EM
q=3	98.77(2.53)	55.26(0.85)	0.963(0.001)	0.962(0.001)
q=5	387.08 (5.61)	89.42(4.51)	0.962(0.005)	0.961(0.004)
q=10	11254.67(9.01)	187.82(6.31)	0.913(0.010)	0.907(0.009)
	Subject/Visit-specific time courses Corr.(SD)		Covariate Effects MSE(SD)	
# of IC	Exact EM	Subspace EM	Exact EM	Subspace EM
q=3	0.998(0.003)	0.998(0.003)	0.083(0.009)	0.081(0.009)
q=5	0.996(0.004)	0.995(0.003)	0.083(0.011)	0.085(0.010)
q=10	0.989(0.010)	0.986(0.007)	0.097(0.023)	0.102(0.021)

2.4 Application to longitudindal rs-fMRI data from ADNI2 study

2.4.1 Rs-fMRI acquisition and description

We applied the proposed L-ICA method to the longitudinal rsfMRI data from the Alzheimer's Disease Neuroimaging Initiative 2 (ADNI2) study. One of the main purposes of the ADNI2 project is to examine changes in neuroimaging with the progression of mild cognitive impairment (MCI) and Alzheimer's Disease(AD). Data used in our analysis were downloaded from ADNI website (http://www.adni.loni.usc.edu) and included longitudinal rs-fMRI images that were collected at baseline screening, 1 year and 2 year for four disease groups, i.e. Alzheimer's Disease (AD), late mild cognition impairment (LMCI), early mild cognition impairment (EMCI) and control (CN). A T1-weighted high-resolution anatomical image (MPRAGE) and a series of resting state functional images were acquired with 3.0 Tesla MRI scanner (Philips Systems) during longitudinal visits. The rs-fMRI scans were acquired with 140 volumnes, TR/TE = 3000/30 ms, flip angle of 80 and effective voxel resolution of 3.3x3.3x3.3 mm. More details can be found at ADNI website (`http://www.adni.loni.usc.edu`). Quality control was performed on the fMRI images both by following the Mayo clinic quality control documentation (version 02-02-2015) and by visual examination. After the quality control, 51 subjects were included for the following ICA analysis. Among these subjects, 6 are diagnosed with AD, 17 are diagnosed with EMCI, 12 are diagnosed with LMCI and 16 are normal controls (CN) at baseline. For gender, there are 2 (33.3%) males for AD, 10 (58.8%) males for EMCI, 7 (58.3%) males for LMCI and 8 (50.0%) males for CN. The mean (SD) of age for each group is 80.3 (4.5) for AD, 72.8 (6.2) for EMCI, 70.0 (7.1) for LMCI and 74.8 (4.7) for CN. Based on F tests, there is no significant between-group difference in gender (p-value = 0.734) but signif-

icant difference in age across the groups (p-value = 0.008). We included both gender and age as covariates in the following L-ICA modeling to control for any potential confounding effects.

2.4.2 Rs-fMRI preprocessing

Skull stripping was conducted on the T1 images to remove extra-cranial material. The first 4 volumes of the fMRI were removed to stablize the signal, leaving 136 volumes for subsequent prepossessing. We registered each subject's anatomical image to the 8th volume of the slice-time-corrected functional image and then the subjects' images were normalized to MNI standard brain space. Spatial smoothing with a 6mm FWHM Gaussian kernel and motion corrections were applied to the function images. A validated confound regression approach (Satterthwaite, Wolf, Roalf, Ruparel, Erus, Vandekar, Gennatas, Elliott, Smith, Hakonarson et al., 2014; Wang et al., 2016; Kemmer et al., 2015) was performed on each subject's rs-fMRI time series data to remove the potential confounding factors including motion parameters, global effects, white matter (WM) and cerebrospinal fluid (CSF) signals. Furthermore, motion-related spike regressors were included to bound the observed displacement and the functional time series data were band-pass filtered to retain frequencies between 0.01 and 0.1 Hz which is the relevant range for rs-fMRI. Lastly, we performed the prior-ICA preprocessing steps including centering, dimension reduction and whitening as described in section 2.2.

2.4.3 L-ICA model specification for ADNI2 study

We applied the L-ICA for modeling the preprocessed baseline, 1 year and 2 year rs-fMRI data from ADNI2 study to examine the longitudinal pattern in brain networks among AD, LMCI, EMCI and CN subjects. We decomposed data into 14 ICs. The first level of L-ICA decompose subjects' longitudinal fMRI data as the product of

subject/visit-specific mixing matrix and spatial source signals as specified in equation (2.1). In the second level model of the L-ICA, we included three binary indicators representing subjects' membership in the four disease groups (with the CN as the reference group) as our primary covariates of interest. We also included subjects' gender and baseline age as covariates to adjust for any potential confounding effects. Specifically, The second level for lth IC was specified as

$$s_{ij}^{(l)}(v) = s_0^{(l)}(v) + b_i^{(l)}(v) + \alpha_j^{(l)}(v) + \left(\beta_{j1}^{(l)}(v), ..., \beta_{j5}^{(l)}(v)\right) \begin{pmatrix} x_i^{AD} \\ x_i^{LMCI} \\ x_i^{EMCI} \\ x_i^{Age} \\ x_i^{Gender} \end{pmatrix} + \gamma_{ij}^{(l)}(v),$$

where $x_i^{AD} = 1$ if subject i is in the AD group and 0 otherwise, and x_i^{LMCI} and x_i^{EMCI} are defined similarly. $\beta_{j1}^{(l)}(v)$, $\beta_{j2}^{(l)}(v)$ and $\beta_{j3}^{(l)}(v)$ represent the contrast between AD, LMCI and EMCI vs. CN, respectively, at the jth visit. We estimated the parameters in the L-ICA model using the subspace-based EM algorithm implemented by in-house MATLAB programs. To ensure the validity of the results from EM, we initialized the EM algorithm with 20 different initial values and the results were highly consistent.

2.4.4 Longitudinal changes in brain networks for ADNI2 study based on L-ICA

Among the extracted ICs from L-ICA, we identified components that correspond to well-established brain functional networks (Smith et al., 2009) such as the default mode network (DMN), medial visual network, occipital visual network and frontoparietal left network, which are visualized in Figures 2.4, 2.5, 2.6, 2.7. In Figure 2.4, we present the L-ICA model-based estimates of the DMN for the four disease groups at the three visits. The subpopulation maps were estimated at the mean baseline

age (73.7 year old) and averaged between the two genders to control for confounding effects. They were thresholded based on the estimated intensity of the source signals. To provide better visualization of the changing patterns across voxels in the DMN, we also present in Figures 2.8 the model-based estimates of longitudinal trends of source signals for voxels in the two subregions of DMN, i.e. the posterior cingulate cortex (PCC) and the lateral parietal cortex (LPC). Figure 2.4 and Figure 2.8 shows that the four disease groups demonstrated different temporal changing patterns in the DMN source signals across the visits. Results show that the AD and LMCI patients generally have more significant changes in the DMN network across the 3 visits as compared with the EMCI and CN subjects. We also found that the longitudinal changes in the network may not necessarily follow a linear pattern and are different between the PCC and LPC regions of the DMN. Another finding from Figure 2.8 is that the AD group demonstrate larger variations across voxels within the network as compared with the other groups.

We also present the estimated subpopulation IC maps and the voxel-level longitudinal trends for other networks of interest (Figures 2.5, 2.6, 2.7, 2.9). Similar as the DMN, we found that the AD and LMCI patients generally have more significant changes in these networks across the 3 visits as compared with the EMCI and CN subjects, the longitudinal changes are not necessarily linear across time, and that the AD group demonstrates the larger within-network heterogeneity as compared with the other groups.

We then applied the proposed inference procedure to formally test the between-group differences at each visit while controlling for potential confounding effects from age and gender. We considered the differences between AD and CN group to demonstrate network changes in clinically diagnosed Alzheimer patients as compared with normal controls. We also considered the differences between the two MCI groups to investigate the heterogeneity between the early and late MCI stages. We then

conducted tests to examine longitudinal changes from baseline to year 2 within disease group. For comparison, we applied the TC-GICA based method to examine the group differences and longitudinal differences. We illustrate results for the DMN for demonstration purpose. First, we note subject-level variance for DMN is 0.128 and the second level variance is 0.762 which gives a within-subject correlation at 0.144 for DMN.

Figure 2.10 and Figure 2.11 present the between-group test results for AD vs. CN and LMCI vs. EMCI, respectively. The proposed L-ICA detected significant between-group differences at each visit. Furthermore, the test results from L-ICA indicate that the between-group differences tend to increase across time with group differences observed at increasingly more spatial locations in the network. In comparison, the TC-GICA based approach identified few differences between the groups. Figure 2.12 represents the differences between baseline and following visits based on L-ICA. It shows that AD has more longitudinal changes compared with other groups. Specifically, Figure 2.13 presents the results for testing the changes from baseline screening to year 2 for AD group. Results from L-ICA show that the AD group demonstrated noticeable longitudinal changes in DMN, which are consistent with findings reported in previous work (Dai et al., 2017). In comparison, the TC-GICA approach identified very little longitudinal changes in DMN among the AD patients. As in the simulation studies, the results from the real data analysis show that the L-ICA method has higher statistical power in detecting group differences and longitudinal changes. Based on a reviewer's suggestion, we also conduct additional analyses to evaluate the robustness of the between-group test results based on the L-ICA. Our findings indicate the test results from L-ICA are fairly robust (Please refer to Appendix A for details).

2.5 Discussion

In this chapter, we proposed a longitudinal ICA model (L-ICA) to formally quantify time-evolving patterns in brain function networks. In the L-ICA model, we incorporated subject-specific random effects to capture the varbilities across subjects and also borrow information across visits within the same subject to improve the model efficiency. Furthermore, to capture the possible non-linear changing effects in brain functional networks, L-ICA incorporates visit-specific covariate effects which can flexibly capture time-varying effects from subjects' demographic, clinical and biological variables. The proposed L-ICA has demonstrated lower type I error and higher statistical power in detecting covariate effects on brain networks and their changes across time.

We develop a maximum likelihood estimation method via EM algorithms for L-ICA model. Based on results from the EM, L-ICA model can simultaneously estimate population and subject/visit-specific brain functional networks. We show that L-ICA's model-based estimates of brain functional networks are more accurate on both population- and individual level. Furthermore, we proposed a computationally efficient subspace based EM algorithm. Simulation study showed that the approximate EM dramatically improves computational efficiency while achieving similar accuracy in model estimation. Matlab functions for implementing the L-ICA model will be added to an Matlab toolbox "HINT: Hierarchical Independent Component Analysis Toolbox" (Lukemire et al., 2018) which is publicly available and updated on NITRC (NeuroImaging Tools and Resources Collaboratory) and the website of Center for Biomedical Imaging Statistics (CBIS) at Emory University.

Some potential extensions to L-ICA is to incorporate more general model specification such as functional data analysis for more flexible modeling of longitudinal effects. Another potential extension to L-ICA is to incorporate spatial dependence in

modeling the covariate effects in the ICA which can help improve the accuracy and efficiency in effects estimation. Furthermore, as a reviewer points out, given the computational cost of the MoG source distribution, one may consider alternative source distributions such as fixed or binary prior densities (Hyvärinen et al., 2001), which is worth investigating in future work.

Acknowledgements

We thank Dr. Tian Dai for helping with downloading and preprocessing ADNI2 study data. Research reported in this publication was supported by the National Institute of Mental Health of the National Institutes of Health under Award Number R01MH105561 and R01MH079448 and by the National Center for Advancing Translational Sciences of the National Institutes of Health under Award number UL1TR002378. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

ADNI is funded by the National Institute on Aging, the National Institute of Biomedical Imaging and Bioengineering, and through generous contributions from the following: AbbVie, Alzheimer's Association; Alzheimer's Drug Discovery Foundation; Araclon Biotech; BioClinica, Inc.; Biogen; Bristol-Myers Squibb Company; CereSpir, Inc.; Cogstate; Eisai Inc.; Elan Pharmaceuticals, Inc.; Eli Lilly and Company; EuroImmun; F. Hoffmann-La Roche Ltd and its affiliated company Genentech, Inc.; Fujirebio; GE Healthcare; IXICO Ltd.; Janssen Alzheimer Immunotherapy Research & Development, LLC.; Johnson & Johnson Pharmaceutical Research & Development LLC.; Lumosity; Lundbeck; Merck & Co., Inc.; Meso Scale Diagnostics, LLC.; NeuroRx Research; Neurotrack Technologies; Novartis Pharmaceuticals Corporation; Pfizer Inc.; Piramal Imaging; Servier; Takeda Pharmaceutical Company; and Transition Therapeutics. The Canadian Institutes of Health Research is providing funds to support ADNI clinical sites in Canada. Private sector contributions are facilitated

by the Foundation for the National Institutes of Health (www.fnih.org). The grantee organization is the Northern California Institute for Research and Education, and the study is coordinated by the Alzheimer's Therapeutic Research Institute at the University of Southern California. ADNI data are disseminated by the Laboratory for Neuro Imaging at the University of Southern California.

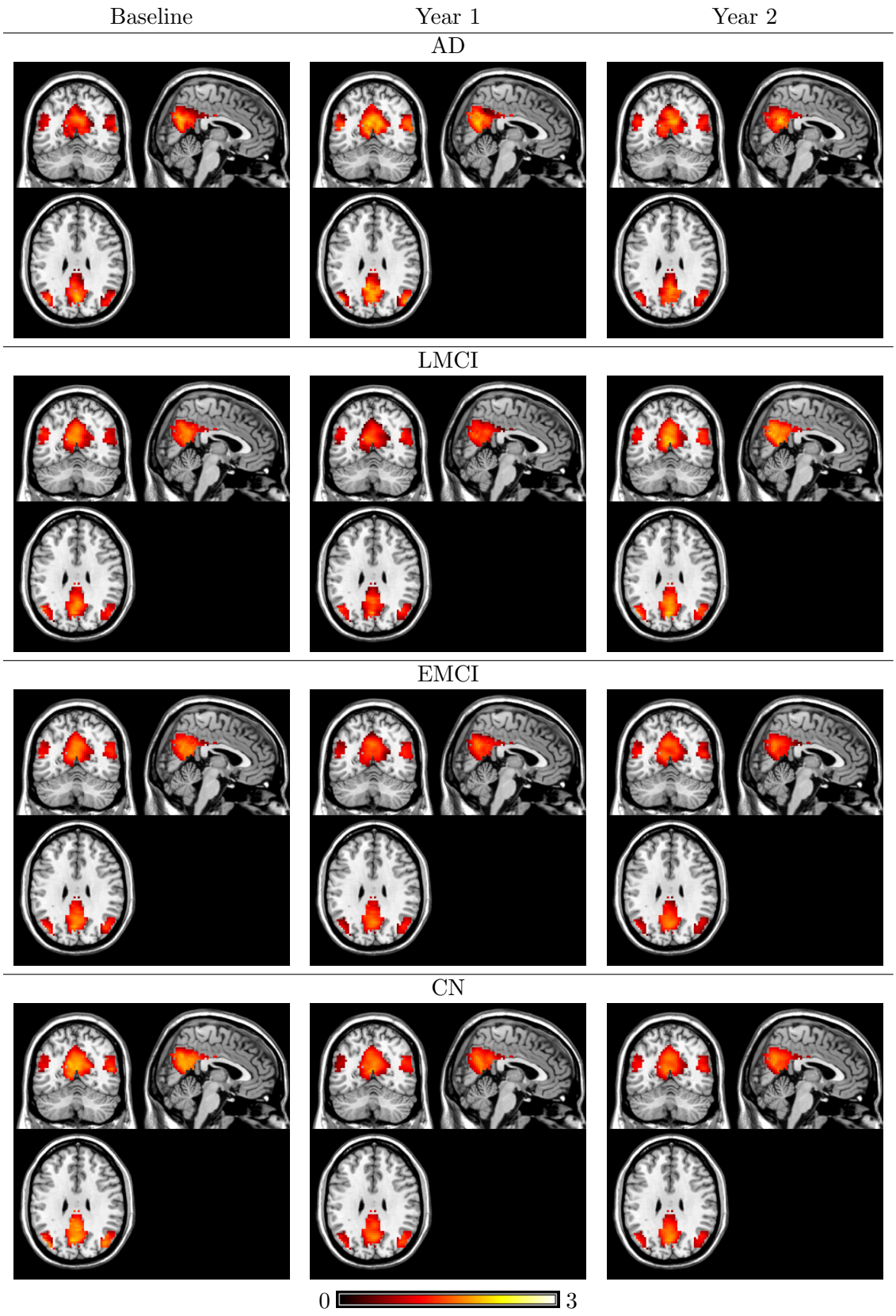

Figure 2.4: L-ICA estimates of subpopulation spatial source signal maps for the DMN for the four disease group across the visits, with the mean baseline age (73.7 year old) and are averaged between genders. All IC maps are thresholded based on the source signal intensity level.

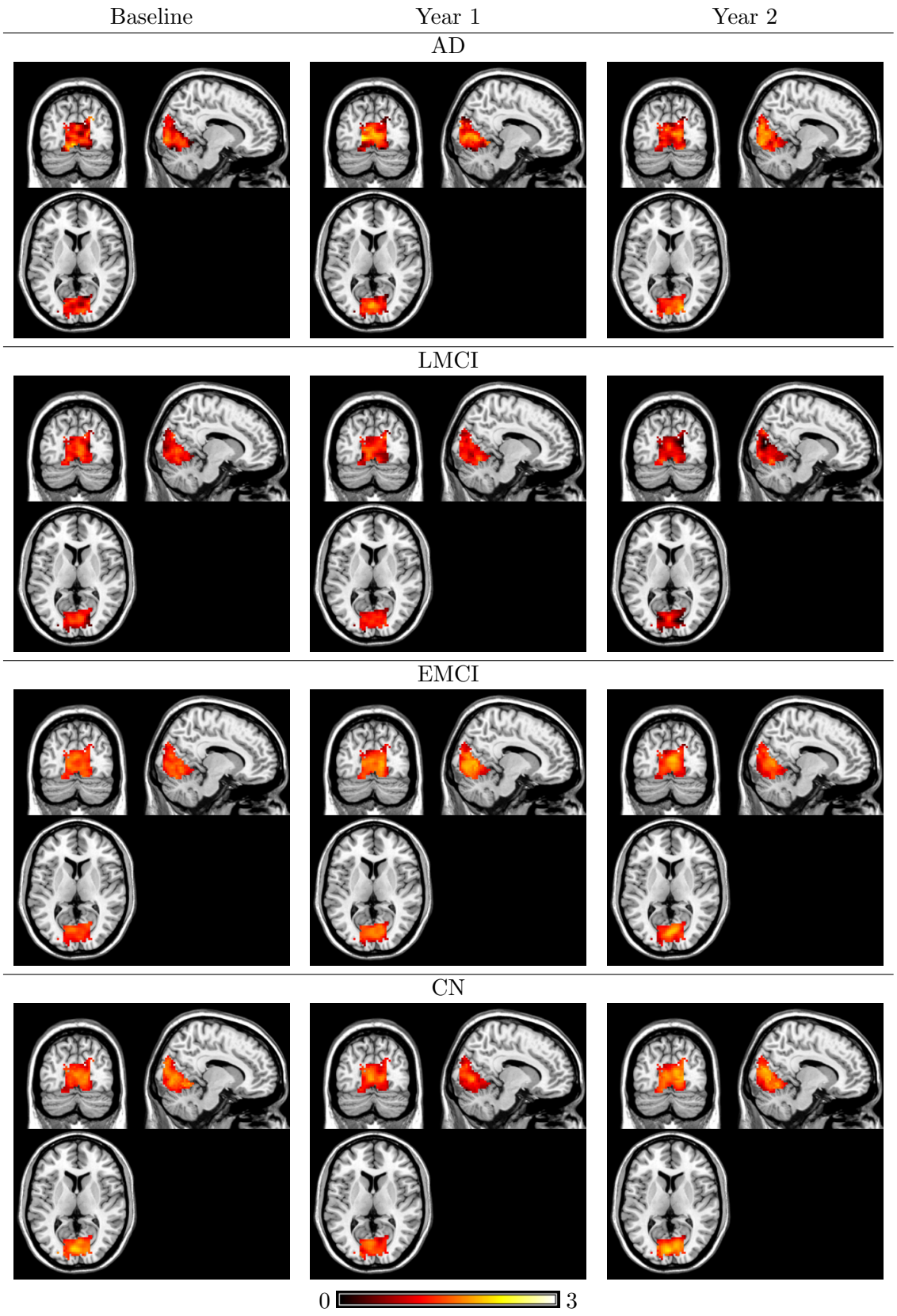

Figure 2.5: L-ICA estimates of subpopulation spatial source signal maps for the medial visual network for the four disease group across the visits, with the mean baseline age (73.7 year old) and are averaged between genders. All IC maps are thresholded based on the source signal intensity level.

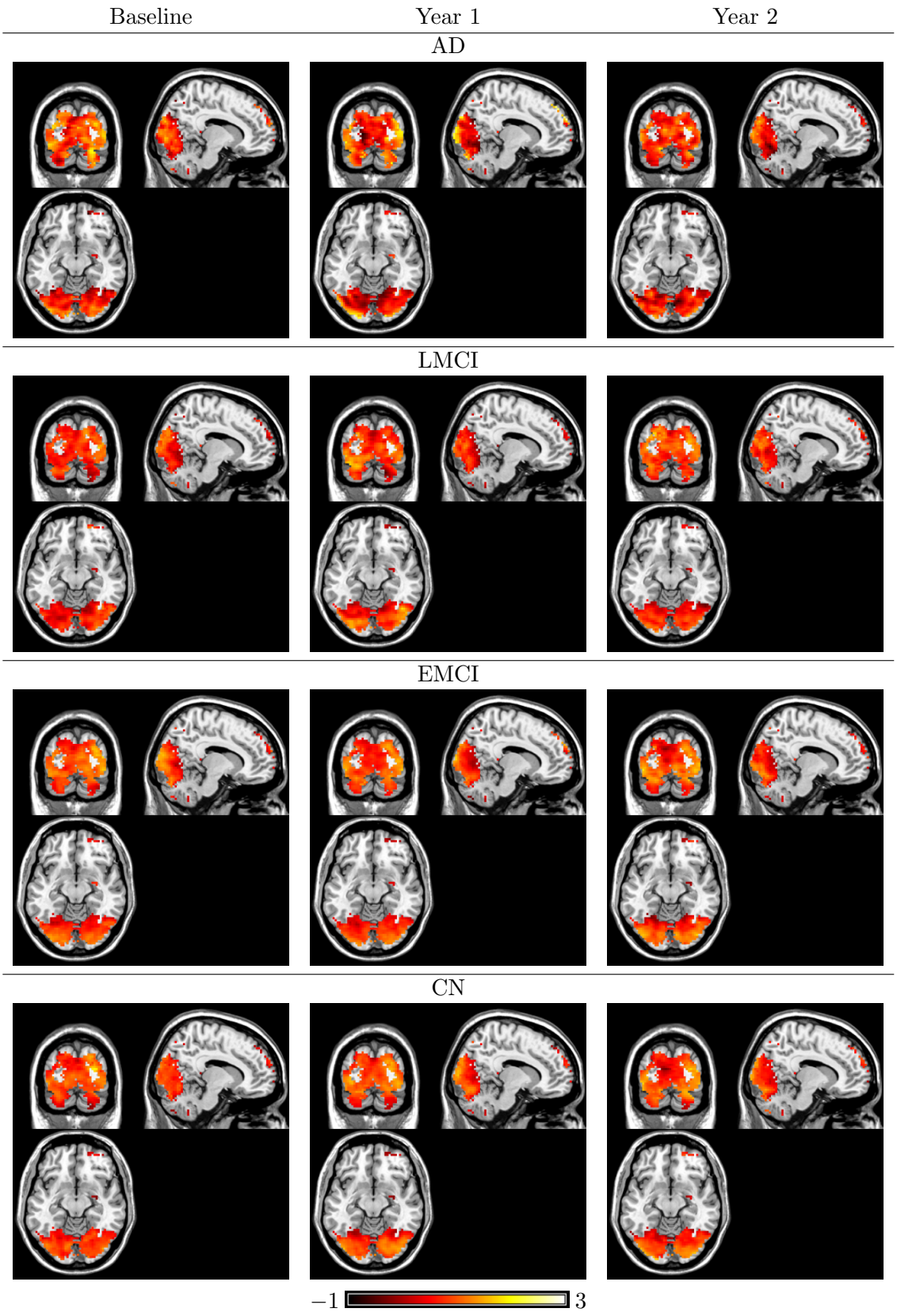

Figure 2.6: L-ICA estimates of subpopulation spatial source signal maps for the occipital visual network for the four disease group across the visits, with the mean baseline age (73.7 year old) and are averaged between genders. All IC maps are thresholded based on the source signal intensity level.

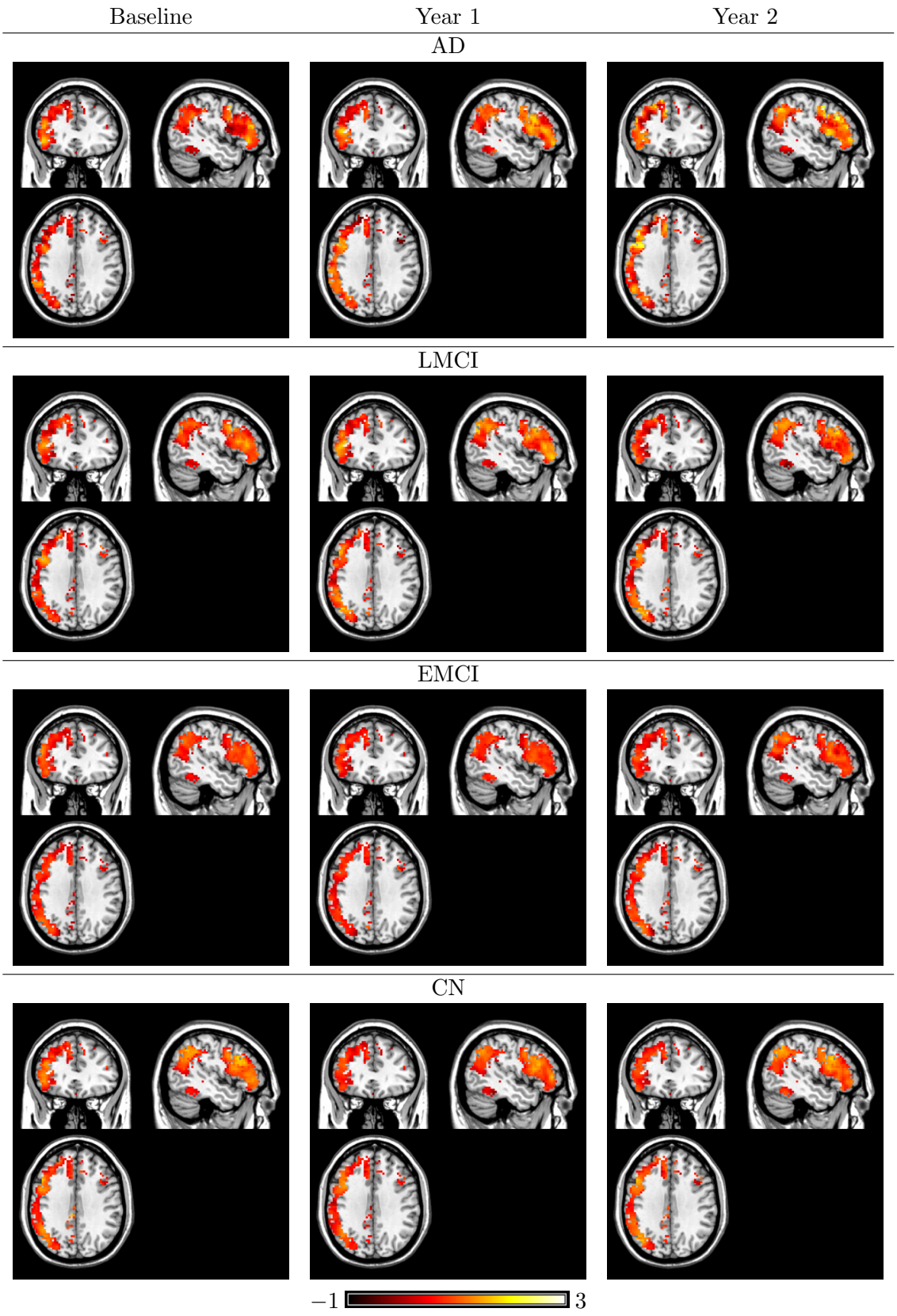

Figure 2.7: L-ICA estimates of subpopulation spatial source signal maps for the FPL for the four disease group across the visits, with the mean baseline age (73.7 year old) and are averaged between genders. All IC maps are thresholded based on the source signal intensity level.

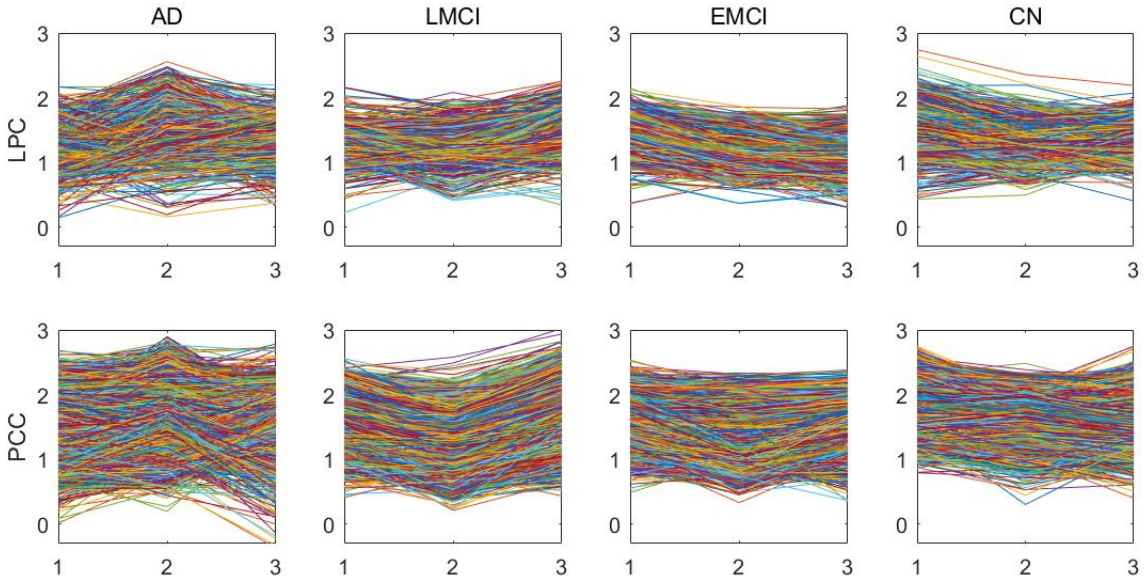

Figure 2.8: L-ICA estimates of longitudinal trends for voxels in the DMN network for each disease group in ADNI2 study. Results show that AD and late MCI (LMCI) patients generally have more changes across visits and that AD group has higher within-network variations than the other disease groups at each visit.

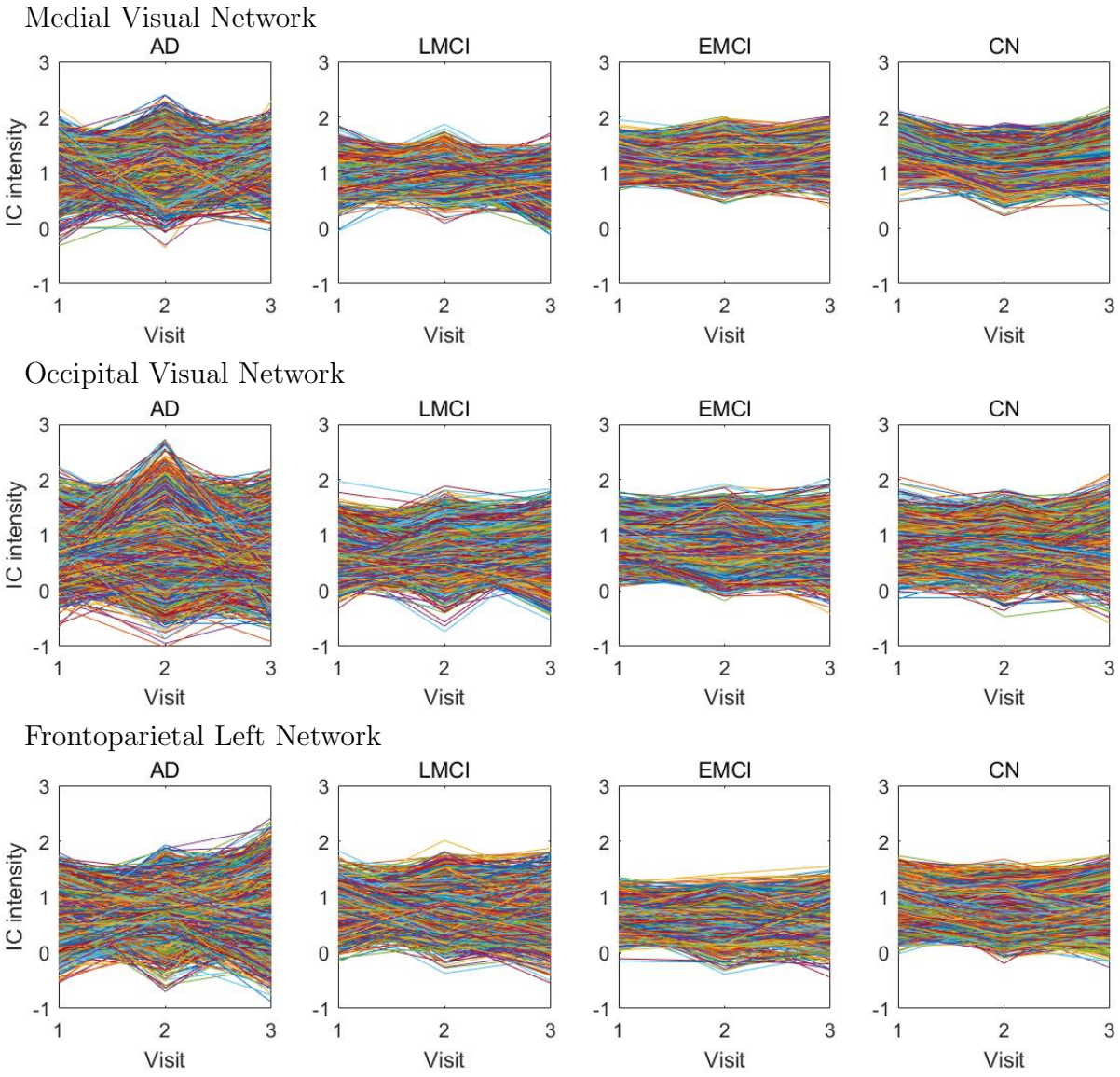

Figure 2.9: L-ICA estimates of longitudinal trends for voxels in FPL and visual networks for each disease group in ADNI2 study. Results show that AD and LMCI patients generally have more changes across visits and that AD group has higher within-network variations than the other disease groups at each visit.

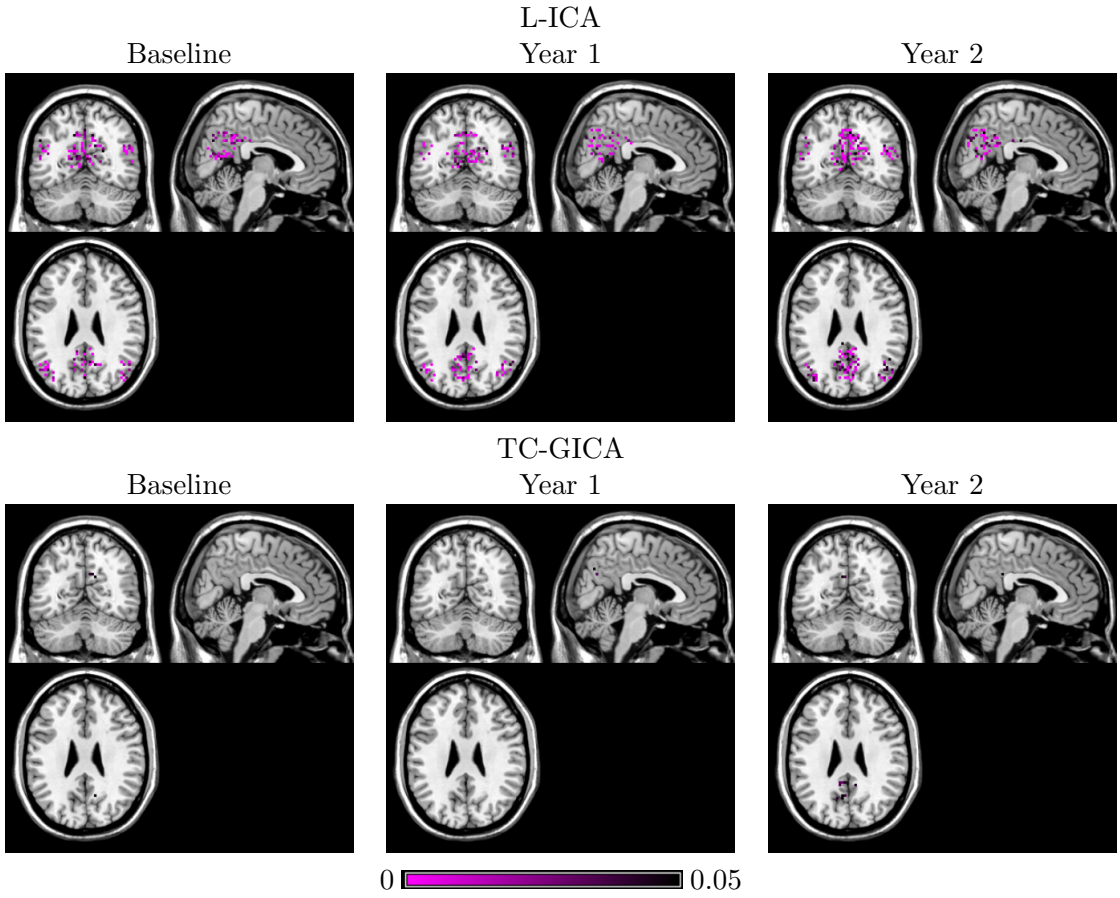

Figure 2.10: p-values for testing group differences in DMN between AD and CN subjects at each visit. The first row shows the test results based on L-ICA and the second row shows the results from the TC-GICA based approach.

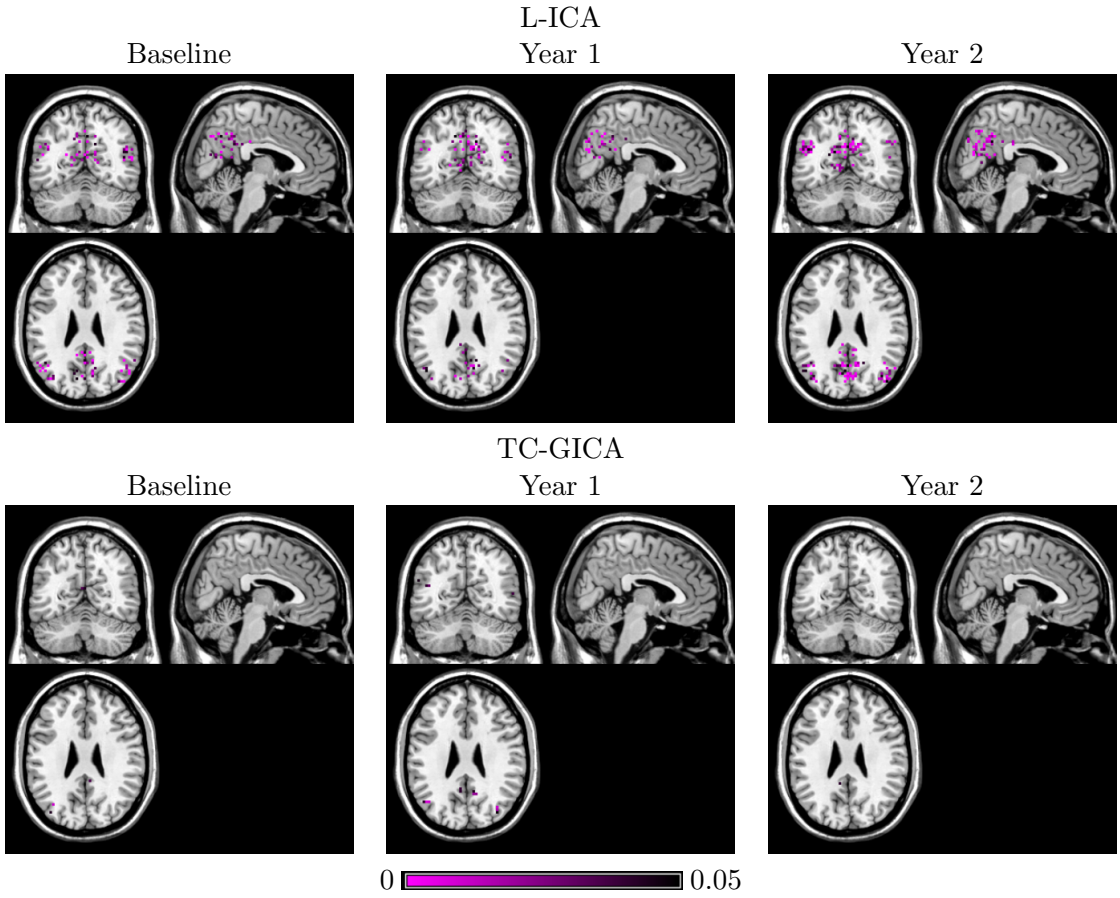

Figure 2.11: p-values, thresholded at 0.05, for testing group differences in DMN between EMCI and LMCI subjects at each visit. L-ICA finds between-group differences in DMN at each visit while TC-GICA detects little group differences.

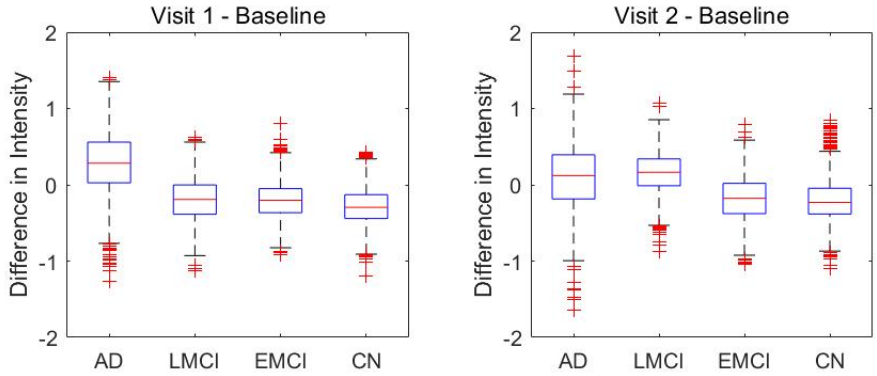

Figure 2.12: Longitudinal changes from baseline and later visits in DMN within AD, LMCI, EMCI and CN groups. The first column shows the comparison between year 1 versus baseline and the second column shows the comparison between year 2 versus baseline, where the value represents the longitudinal differences in source signal intensity for DMN voxels, i.e. $\hat{\boldsymbol{s}}_j(v) - \hat{\boldsymbol{s}}_0(v)$.

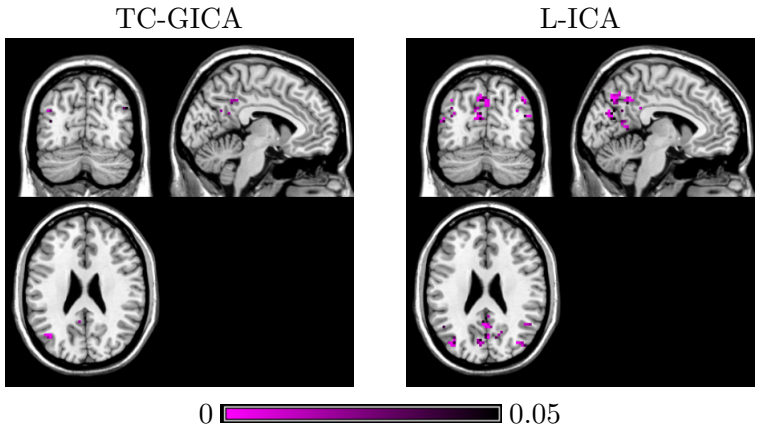

Figure 2.13: p-values, thresholded at 0.05, for longitudinal changes between baseline and year 2 for the default mode network (DMN) among the AD group. L-ICA finds longitudinal changes in major regions of DMN among AD patients while TC-GICA detects little changes in DMN among these patients.

Chapter 3

Novel signal decomposition method for brain connectivity data

3.1 Introduction

In this paper, we present a blind source separation method for decomposing network measures to identify underlying source signals characterizing independent network traits. Our work is motivated by brain network research in neuroimaging studies. In recent years, network-oriented analyses have become an important research field in neuroscience for understanding brain organization and its involvement in neurodevelopment and mental disorders (Bullmore and Sporns, 2009; Deco et al., 2011; Satterthwaite, Wolf, Roalf, Ruparel, Erus, Vandekar, Gennatas, Elliott, Smith, Hakonarson et al., 2014; Kemmer et al., 2015; Wang et al., 2016; Wang and Guo, 2019). In neuroimaging studies, network measures are derived from functional or structural brain imaging to reflect functional or structural connections among a set of nodes or brain regions. The network measures are typically encoded as symmetric matrices where the entries represent brain connectivity between pairs of nodes or regions in the brain. For example, derived from functional magnetic resonance imaging (fMRI) or EEG, functional connectivity (FC) measures the dependence between temporal dynamics in neural processing of spatially disjoint brain regions (Biswal et al., 1995; Lang et al., 2012; Kemmer et al., 2018). The commonly used FC measures include the Pearson correlation matrix or partial correlation matrix (Smith et al., 2011; Church et al., 2008; Seeley et al., 2009; Wang et al., 2016). These connectivity measures contain important information about the structure of brain organizations and could potentially serve as the individual neural fingerprint that guides important clinical decisions (Finn et al., 2015; Amico and Goñi, 2018b).

There are several major challenges in network analysis in neuroimaging studies. First, to fully capture the whole brain organization, connectivity matrices are usually high dimensional (Chung, 2018). For example, the voxel-level brain network based on fMRI data contains tens of thousands of nodes and nearly half a billion edges.

More commonly, atlas-based brain networks are constructed based on a brain atlas or node system such as the automated anatomical labeling (AAL) atlas (Tzourio-Mazoyer et al., 2002) and the more recent Power's node system (Power et al., 2011) and Glassier's atlas (Glasser et al., 2013). Although the number of nodes are reduced dramatically in these atlas-based networks, there are still hundreds of nodes and hundreds of thousands of edges for each single subject (Chung, 2018; Solo et al., 2018; Wu, Stramaglia, Chen, Liao and Marinazzo, 2013; Wang et al., 2016). This ultra-high dimensionality makes it challenging to analyze the brain network for scientific discoveries. Secondly, brain connectome is a complex organization encompassing a range of underlying subnetwork structures (i.e. default mode network). The observed connectivity matrices, measuring the overall connectivity patterns across the brain, represent aggregated information from various underlying connectivity systems or network structures. This makes it hard to interpret the findings from whole brain network analysis. Thirdly, given the large number of edges present in the brain networks, there is a high possibility of spurious findings across the network in neuroscientific research.

3.1.1 Our Proposal

In this paper, we propose a novel low-rank structure with uniform sparsity (LOCUS) method as a fully data-driven source separation method to decompose the network measures. LOCUS decomposes subjects' connectivity matrices into the product of a latent source signal matrix that characterizes underlying network traits and a mixing matrix that contains subject-specific loadings on the source signals. To achieve more efficient and accurate source separation for connectivity matrices, LOCUS adopts a low rank factorization structure for each of the latent source signals. This low rank structure results in a dramatic decrease in the number of parameters and hence leads to more robust recovery of underlying connectivity traits. Furthermore, we propose a

novel penalization approach to achieve sparsity in the extracted latent sources. This novel sparsity regularization method help remove false positive connections when extracting the connectivity traits.

3.1.2 Related Works and Our Contributions

Currently, the most popular blind source separation (BSS) methods for decomposing neuroimaging data is independent component analysis (ICA). In neuroscientific research, ICA is widely used to decompose the brain imaging data to reduce the dimensionality, increase signal to noise ratio and facilitate the process of data analysis, which has achieved great success in many applications (Beckmann and Smith, 2004; Guo and Tang, 2013; Wang and Guo, 2019; Guo, 2011; Contreras et al., 2017). However, existing ICA applications have mainly focused on decomposing observed neural activity signals such as the blood oxygen level-dependent (BOLD) series from fMRI or the electrodes signal series from EEG. As for the brain network data, the diagonal element of a connectivity matrix is undefined meaning that the connectivity data for each subject is actually stored as a lower triangle matrix without diagonal part (Amico et al., 2017). Therefore, traditional matrix decomposition approaches such as ICA are not well defined for this problem.

Currently, existing methods for decomposing the high-dimensional connectivity matrices usually follow two strategies. The first strategy is that instead of directly analyzing the functional connectivity matrix, researchers built models based on the intermediate step. For example, many existing works focus on modeling subject-level covariance matrix instead of Pearson's correlation or modeling precision matrix instead of partial correlation. In this case, the diagonal part is well defined and statistical models can be built based on existing standard matrix distribution, (Franks and Hoff, 2016; Cook and Forzani, 2008; Xia and Li, 2017; Virta et al., 2017). The second widely used strategy is based on vectorization which first converts the upper-

triangle part of subject-level FC matrices into a vector and stacks these vectors across subjects to form a group-level FC matrix for decomposition (Amico et al., 2017; Amico and Goñi, 2018a,b). Specifically, Amico et al. (2017) proposed a connectivity independent component analysis framework (connICA) which treated all subject-specific connectivities as independent features and utilized existing ICA algorithm to decompose the group-level FC matrix into group-level independent non-Gaussian sources and subject-specific loadings.

There is very limited development of ICA for decomposing connectivity measures derived from imaging. The only existing method connICA (Amico et al., 2017) directly applies an existing ICA algorithm on vectorized functional connectivity matrices without taking into account the special structures of connectivity matrices or sparisity control, potentially leading to inaccurate and noisy estimates. The proposed LOCUS is one of the first statistically principled BSS methods for brain connectivity measures and has several major innovative contributions: 1) LOCUS achieves more efficient and accurate source separation for connectivity matrices by assuming a low rank factorization structure for the extracted latent connectivity components/traits. This approach is well motivated by the observation that the brain connectivity matrices often has structures such as block-diagonal or banded structure (Amico et al., 2017) that can be efficiently captured with a low rank factorization (Zhou et al., 2013). The low rank structure results in a dramatic decrease in the number of parameters and hence leads to more accurate recovery of underlying connectivity traits. 2) LOCUS uses a novel sparsity regularization method to help remove false positive connections when extracting the connectivity traits, thus alleviating the issue of searching for optimal threshold in the existing connICA method. 3) Compared to the vectorization approach such as connICA, the low-rank structure provides another major advantage for LOCUS by generating results directly interpretable regarding the underlying network structures

To demonstrate the advantage of LOCUS over existing methods, we presents some findings based on the real data. Figure 3.1 shows the averaged covariance and correlation matrices across 515 healthy subjects in PNC study (more details about PNC study is in section 3.4). The high variability of each node, i.e. diagonal part in covariance matrix, limits our ability to investigate the between-node connectivities. Moreover, the varibility level for each brain node can also be altered significantly due to the image acquisition techniques or different preprocessing approaches. Therefore, the first strategy, discussed above, is usually not desirable for brain network analysis. Furthermore, Figure 3.2 shows 4 estimated latent sources based on connICA framework (with threshold). Because connICA cannot ensure sparsity, we further applied a threshold to remove weak signals. One important finding is that the underlying sources for FC matrices have a very clear low-rank structure. It also shows that the nearby edges, i.e. edges connected to the same node or edges within same functional module, tend to have high similarity within each sources. These findings motivates a novel decomposition method taking advantage of these properties of FC matrices.

Although connICA provides many important findings, the underlying ICA algorithm has relatively poor performance based on previous studies (Guo, 2011; Guo and Tang, 2013) and cannot ensure sparsity for the latent source signals. Moreover, it follows the vectorization strategy and treats each edge independently which ignores the similarity for the nearby edges. Furthermore, the latent sources of the real data tend to have a low-rank property indicating that treating each edge separately can be over-parameterized which could lead to noisy estimates and further limit model's reproducibility.

In the existing literature, the most popular class of modeling network-value data is mainly focusing on adjacency matrix and is trying to learn a latent space representation for each node (Hoff et al., 2002; Hoff, 2008; Wang, Durante, Jung and Dunson, 2017; Durante et al., 2017). In the most recent work, Durante et al. (2017) proposed

(A) Covariance (B) Correlation

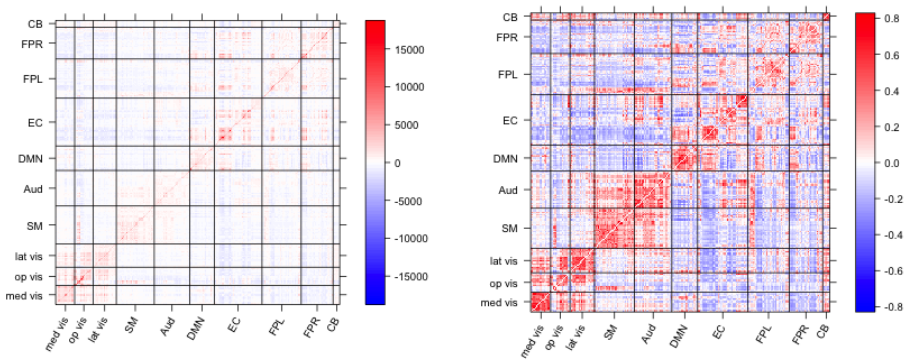

Figure 3.1: Preliminary findings based on PNC Study: (A) and (B) represents the averaged covariance matrix and Pearson correlation matrix across 515 healthy subjects in PNC study.

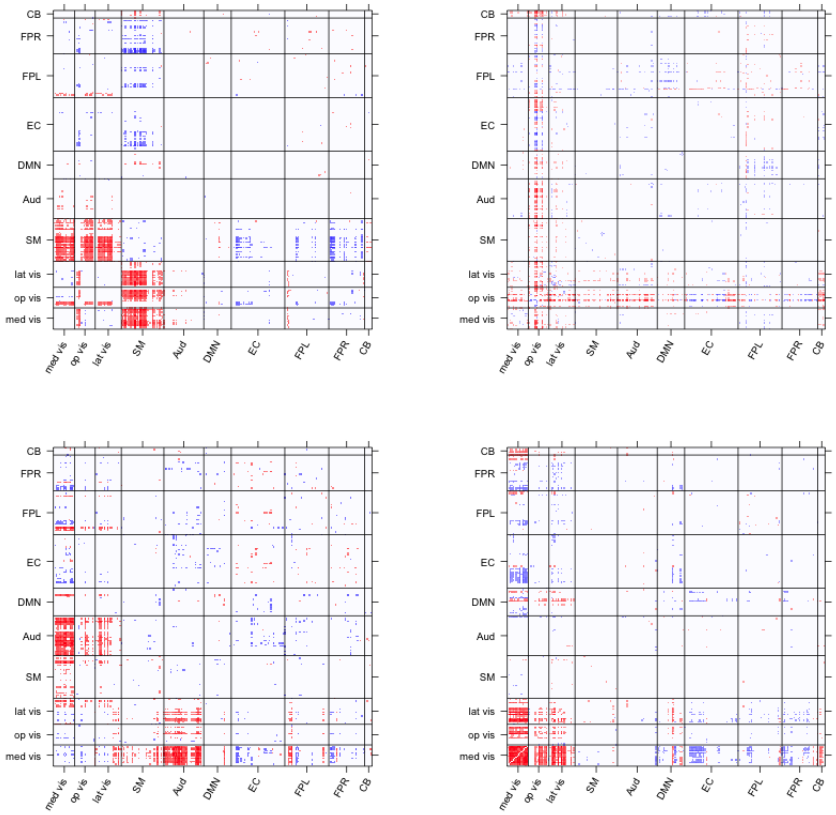

Figure 3.2: Four estimated latent sources based on connICA and PNC study, where each source is further threshold to ensure sparsity.

a nonparameteric Bayesian mixture model (BMM) for adjacency matrix of brain network, where a network is modeled from a mixture of Bernoulli random varaibles, and in each Bernoulli component all the nodes are mapped into a component-specific coordinate system with the common intercept term. The probability of being connected between two nodes in each component term is represented as the similarity between the two corresponding node coordinates, and the dimension of the coordinate space in each component controls the low-rankness for the latent source and as well as the number of parameters in the model. As demonstrated by Figure 3.2, brain latent sources (sub-network) are also of low-rank structures, which are well-supported by the latent space methods like BMM. Therefore, in LOCUS, we consider a similar specification for each latent sources with a source-specific scale-free node coordinate system. Modeling each node as coordinate vector provides a way to share the information across different edges connected to the same node and also can significantly save the number of parameters in the model.

But different from BMM and other existing works, LOCUS directly analyzes brain connectivity measures instead of binary edges. Most importantly, instead of modeling the whole brain network, LOCUS decomposes a brain network into the loadings of different latent sources and models each latent source simultaneously using a symmetric low-rank structure, which in nature is a BSS model and has distinct goal and application. Here we want to further point out that the latent source in LOCUS is distinctively different from latent component in BMM from Durante et al. (2017). First, latent sources in LOCUS are shared across all subjects while latent component in BMM is subject-specific. Second, different component share the same variables (i.e. intercept) across different components, which is different from LOCUS where each source are required to have distinct latent coordinate system to ensure independence. Third, in the scale-free coordinate system of LOCUS, the weights for each source not only control the importance for each coordinate, but can also contain both positive or

negative values, which is also different from the all-positive setting in Durante et al. (2017). We note that our setting can further increase the flexibility for each latent source to capture more complicated hidden patterns.

Currently, there are two major approaches of penalization for low rank factorizations (Raskutti and Yuan, 2015): element-wise penalty on each latent coordinate (Zhou et al., 2013; Sun and Li, 2017; Li et al., 2018) and rank-based penalty on entire latent source (Chen et al., 2013; Rabusseau and Kadri, 2016; Fan et al., 2017). As pointed out in Sun and Li (2017), rank-based penalty cannot ensure the sparsity and hence is not the ideal approach for our model. For the element-wise penalty, although it can ensure the sparsity for each latent coordinate, the latent sources constructed from these latent coordinates may not achieve the desired sparse pattern. As shown in the previous studies and our simulation study, element-wise penalty tends to give the structured noise which can be hardly identified from the true signals (Zhou et al., 2013; Zhou and Li, 2014). Moreover, since the fundamental objective of incorporating penalty is to ensure an uniform sparseness for constructed latent sources to facilitate the finding of true signal, we provide a much more intuitive way of penalization to achieve this goal compared with all these existing approaches. Specifically, instead of penalizing on each node coordinate or the rank, we directly penalize the inner product between node vectors and the weights to ensure the global sparsity for constructed sources. We point out that this is same as angle-based penalization if we focus on one node coordinates and condition all others, which in nature is trying to increase the orthogonality between each pair of the node coordinates adjusted by weights. This novel way of penalization provides a better way of controlling the sparsity globally and therefore we named it as uniform penalization approach.

As a summary, in this paper, we proposed a novel connectivity matrix decomposition method using symmetric low-rank structure with uniform sparsity (LOCUS). To the best of our knowledge, this is the first formal signal separation approach specifi-

cally designed for decomposing brain connectivity matrices, and the proposed penalty approach is at a different perspective from the existing methods. We also developed an efficient node-moving algorithm for estimating the unknown parameters in the proposed model with penalty, and we further conducted extensive simulation studies to evaluate the performance of the proposed model and node-moving algorithm. We also shown that our algorithm can break down the original non-convex optimization problem into a series of sub-convex problems which can be solved easily. We applied our proposed method to real data and found biologically meaningful results.

The overall structure of the paper is organized as follows. Section 3.2 introduces the methodology including model specification, the underlying Bayesian model, algorithm for estimation. Section 3.3 is for the simulation study. Section 3.4 is the real data application. Section 3.5 is the discussion and conclusion.

3.2 Methodology

3.2.1 Notation and Problem Statement

For $h > 1$, we define vector norm $\|\boldsymbol{x}\|_h = (\sum_i |x_i|^h)^{\frac{1}{h}}$, and we denote the Frobenius norm of a matrix by $\|\boldsymbol{X}\|_F = (\sum_{ij} X_{ij}^2)^{1/2}$. We also denote $\mathcal{L}$ to be a map from a square matrix to it's upper-trianlge part, i.e. $\mathcal{L}(A) = [A(1,2), A(1,3), ..., A(V-1,V)]' \in \mathcal{R}^p$ for any $A \in \mathcal{R}^{V\times V}$, where $p = V(V-1)/2$

Since self-relationship in the network is not of interest, for the undirected network all the information of a connectivity matrix are stored in its upper-triangle part. Therefore, we denote $\mathcal{L}_V(\mathcal{R}) = \{\boldsymbol{W} = \{W(u,v)\}_{1\le v<u\le V}, W(u,v) \in \mathcal{R}\}$ to be the space of all connectivity matrices for undirected network with V nodes. Assume that we have N observed connectivity matrices $\boldsymbol{Y}_1$,...,$\boldsymbol{Y}_N \in \mathcal{L}_V(\mathcal{R})$. Denote $p = V(V-1)/2$ to be the number of pairs of nodes in each $\boldsymbol{Y}_i$. For simplicity, we treat each $\boldsymbol{Y}_i$ as a column vector of length p and denote $\boldsymbol{Y} \in \mathcal{R}^{N\times p}$ to be the big data matrix containing

all connectivity matrices with $\boldsymbol{Y} = [\boldsymbol{Y}_1, ..., \boldsymbol{Y}_N]^T$.

The goal of signal separation is to decompose $\boldsymbol{Y}$ into a product of sources and loading, i.e. $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{E}$, where $\boldsymbol{S} = [\boldsymbol{S}_1, ..., \boldsymbol{S}_q]^T \in \mathcal{R}^{q \times p}$ is the source matrix / basis matrix, $\boldsymbol{A} = \{a_{il}\} \in \mathcal{R}^{N \times q}$ is the loading matrix with ith row representing the loading for ith subject, and $q \leq min(p, N)$. For example, based on $\boldsymbol{S}$ and $\boldsymbol{A}$, $\boldsymbol{Y}_i$ can be approximated by $a_{i1}\boldsymbol{S}_1 + ... + a_{iq}\boldsymbol{S}_q$. Moreover, existing signal separation methods all follow this framework but have different assumptions or constraints on $\boldsymbol{S}$ or $\boldsymbol{A}$:

- ICA has constraints on sources but leave loading matrix unconstrained, where it assumes that $\boldsymbol{S}_l$'s are independent and non-Gaussian distributed;
- Dictionary learning puts sparsity constraints on loading matrix $\boldsymbol{A}$ to achieve a sparse representation of the data but usually leave the basis matrix unconstrained.

As discussed in the introduction, the number of parameters, i.e. edges in the network, is massively larger than the sample size N even when number of nodes V is small. For power's node system, $V = 264$ which has 264*263/2 = 34716 edges, and the commonly used number of latent sources is around 30 which further gives 30*34716 parameters in the latent source matrix. It is clearly that the sample size for existing Neuroimaging studies cannot handle this many unknown parameters. Therefore, model with appropriate dimension reduction technique will be necessary, and the model also need to be flexible enough to capture the inherent network properties and topological structures varied across different latent sources. It is clear now that existing methods, such as connICA, which failed to consider these issues, are expected to have poor performance in connectivity matrices decomposition problem. In fact, the difference between network data and unstructured data is that the nearby edges in the network can share common properties driven by network configuration. Moreover, for the model robustness and interpretability, sparsity constraints is necessary and

also requires careful specification to keep the network properties.

In the next section, we will present our model which adopts a highly flexible dimension reduction approach to capture the source-specific network properties by borrowing information across nearby edges in the network. We will also propose a novel but intuitive sparsity constraints specifically designed for our decomposition model which can precisely control the sparsity level while preserve the network properties.

3.2.2 Locus - Proposed Method

3.2.2.1 Low-rank Decomposition Method

In this section, we will introduce the LOCUS framework for decomposing the connectivity matrices. Motivated by the findings of real data and the discussion above, we consider to model each latent sources based on a source-specific low-rank structure which can reduce the parameter space while preserves the network property. The first level model is same as existing framework as

$$\boldsymbol{Y}_i = a_{i1}\boldsymbol{S}_1 + ... + a_{iq}\boldsymbol{S}_q + \boldsymbol{e}_i, \tag{3.1}$$

where $\boldsymbol{e}_i$ is the noise term with mean as zero and a_{il} is the loading scalar for subject i at source l. Each latent sources $\boldsymbol{S}_l$ belongs to the same space as $\boldsymbol{Y}_i$ in $\mathcal{L}_V(\mathcal{R})$, where the diagonal is undefined and all information is stored in lower-triangle part. To flexibly model each latent source and reduce the dimension appropriately while preserving the network property, we link each $\boldsymbol{S}_l$ with a latent symmetric low-rank decomposition via $\mathcal{L}$ transformation:

$$\boldsymbol{S}_l = \mathcal{L}(\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_l'), \tag{3.2}$$

where $\boldsymbol{X}_l = [\boldsymbol{X}_l(1), .., \boldsymbol{X}_l(V)]^T \in \mathcal{R}^{V \times R_l}$ with $\boldsymbol{X}_l(v)$ representing the latent coordinates of dimension R_l for node v in lth latent source, and $\boldsymbol{D}_l = diag(\lambda_{l,1}, .., \lambda_{l,R_l})$ is a diagonal matrix containing the weight (both magnitude and direction) for each coordinate, and R_l is the dimension of the latent node-level coordinate system for lth latent source. Specifically, $\lambda_{l,r} \in \mathcal{R}$ can be either positive or negative making the constructed source matrix (before $\mathcal{L}$ transformation) can be either positive semi-definite or not.

Actually, this specification has an appealing interpretation. The weighted dot product setting in (3.2) is to learn a source-specific latent coordinate system for all nodes, where the similarity between each pair of nodes is used to represent the connectivity level in the latent source. In our case, the similarity between node v and u for lth source is defined as $S_l(u, v) = \boldsymbol{X}_l(u)'\boldsymbol{D}_l\boldsymbol{X}_l(v) = \sum_{r=1}^{R_l} \lambda_{l,r} X_{l,r}(u) X_{l,r}(v)$, which is a linear kernel adjusted by the source-specific weight. Here, $\lambda_{l,r}$ controls the importance of r coordinate, where the closer is $\lambda_{l,r}$ to zero, the lower the contribution of pathway r to $\boldsymbol{S}_l$. The coordinate $X_{l,r}(v)$ might represent the activity of node v within pathway r. For example, for the pathway with $\lambda_{l,r} > 0$, nodes with coordinates in the same direction, i.e. both positive or negative, will have stronger connectivity in $\boldsymbol{S}_l$. In summary, model settings in (3.2) are learning q latent coordinate spaces for each latent source.

We note that the factorization (3.2) utilized a similar dot product characterization of edge connectivity for a single network (Hoff et al., 2002; Hoff, 2008; Wang, Durante, Jung and Dunson, 2017; Durante et al., 2017). As shown in Durante et al. (2017), for modeling a single newtork, the representation (3.2) provides a more general characterization of interconnection structures and network properties than stochastic block model (Airoldi et al., 2008; Nowicki and Snijders, 2001) and latent distance model (Hoff et al., 2002). Durante et al. (2017) proposed a mixture model for populations of network adjacency matrices where they utilized multiple components to capture the

heterogeneity in network data. Specifically, their model specified the same low-rank structure for each component while forcing all weights to be positive and all components sharing the same dimension, which is more restrictive compared with our setting.

As shown in Figure 3.2, the network property and topological structures can vary substantially across different latent sources. Therefore, we allow the dimension R_l to be different across each latent source which controls the maximum information one latent source can hold, and by allowing the source-specific weights $\lambda_{l,r}$ to contain both positive and negative values, the latent symmetric low-rank structure can be more representative and flexible. Since the scale of the decomposition is not identifiable, for computational robustness, we constrain that each column of $\boldsymbol{X}_l$ has a L2 norm of 1.

So far, compared with connICA approach, our setting has two advantages. First, the symmetric low-rank structure can considerably reduce the dimensionality from $qV(V-1)/2$ to $V\sum_l R_l$ parameters, which can significantly improve the model robustness and reproducibility. Second, different from modeling each edge separately, through the latent coordinate system, different edges connecting from the same node v can share information through $\boldsymbol{X}_l(v)$. For example, the connections from a central node v, which has stronger global connectivities across the network, share the same $\boldsymbol{X}_l(v)$ with relatively higher scale in equation (3.2) and hence can have larger connectivities.

In summary, the objective function of LOCUS for connectivity matrix decomposition problem without any penalty turns into:

$$\min \sum_{i-1}^{N} \|\boldsymbol{Y}_i - \sum_{l=1}^{q} a_{il}\mathcal{L}(\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_l')\|_2^2. \tag{3.3}$$

3.2.2.2 Uniform Sparseness

Following our previous work, we do not force any constraint on loading matrix (Guo, 2011; Guo and Tang, 2013). For each latent source $\boldsymbol{S}_l$, existing tensor decomposition framework usually has two types of penalties: low-rank constraint and element-wise sparsity constraint (Raskutti and Yuan, 2015). Low-rank constraints such as nuclear norm are penalizing on the rank, i.e. $\|\boldsymbol{D}_l\|_1$ which cannot guarantee the sparsity on latent sources and hence is not suitable for selecting the connections in each latent source for interpretation (Fan et al., 2016, 2017). The element-wise sparsity constraints are based on adding additional penalty terms on $\boldsymbol{X}_l$, which we refer as coordinate-based penalty (Zhou et al., 2013; Li et al., 2018; Fan and Li, 2001; Zhang et al., 2010). This approach only focuses on adding constraints on $\boldsymbol{X}_l$ but in fact the component we need to be sparse is the constructed $\boldsymbol{S}_l$ from the dot product between $\boldsymbol{X}_l$ and $\boldsymbol{D}_l$ through (3.2). Therefore, the class of element-wise penalization methods might not provide desired sparsity level for our model. Moreover, as we discussed above, each latent coordinate, i.e. $X_{l,r}(v)$, could represent the pathway-specific activity level for brain node v, which can have substantially different expression level across the brain and latent sources. In this case, the element-wise approach need to specify a large number of coordinate-specific or node-specific penalization terms to adjust to the heterogeneity among $X_{l,r}(v)$'s, which is not realistic.

In fact, the goal is to ensure sparsity on $\boldsymbol{S}_l$ for selecting the important brain connections in each latent source, which is the key component of primary interest. Therefore, instead of coordinate-based penalty, we propose a more intuitive approach to directly penalize on $\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_1'$ for the desired property on constructed latent sources,

$$\min \sum_{i=1}^{N} \|\boldsymbol{Y}_i - \sum_{l=1}^{q} a_{il}\mathcal{L}(\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_l')\|_2^2 + \phi \sum_{l=1}^{q}\sum_{u<v} |\boldsymbol{X}_l(u)'\boldsymbol{D}_l\boldsymbol{X}_l(v)|, \tag{3.4}$$

where ϕ is the weight for penalty term. The penalty term in (3.4) is to penalize the

off-diagonal part of constructed latent source, i.e. $\|\mathcal{L}(\boldsymbol{X}_1\boldsymbol{D}_l\boldsymbol{X}_l')\|_1$, which is the same as L1 penalty on $\boldsymbol{S}_l$. We refer this as L1-based uniform sparsity, and other extensions with L2, SCAD or MCP can also be easily applied.

We note that for each latent source, penalization term in (3.4) is actually an angle-based penalization. For example, in lth latent source, if we focus on node v and condition on others nodes, minimizing the penalty term with respect to $\boldsymbol{X}_l(v)$ is equivalent as finding the direction to be as orthogonal as other nodes with weights $\boldsymbol{D}_l$. From a geometric perspective, this approach is trying to penalize the angle of node vectors given the weights instead of the magnitude which is fundamentally different from element-wise penalty. Moreover, by directly penalizing on $\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_1'$, we can precisely control desired sparsity level on the constructed latent source and this approach is named as uniform sparsity.

In the next section, we will further illustrate the property of uniform sparseness from a statistical perspective.

3.2.3 The Underlying Bayesian Model for Locus

Section 3.2.2 introduced the matrix decomposition procedure of Locus. In this section, we will present the underlying Bayesian model for Locus and show that the uniform sparsity penalty actually corresponds to a joint prior specification on $\boldsymbol{X}_l$ and $\boldsymbol{D}_l$. It will further demonstrate the advantage of the uniform sparsity penalty over the existing penalty techniques. Moreover, we will show that solving the objective functions of (3.4) is same as using the maximum a posteriori, a.k.a MAP, estimators for the underlying Bayesian model.

For each connectivity measure $Y_i(u, v)$ of subject i, the first level model decomposes it onto the space expanded by q components with component-specific loading:

$$Y_i(u, v) = a_{i1}S_1(u, v) + ... + a_{iq}S_q(u, v) + e_i(u, v), \tag{3.5}$$

where e_i represents the first-level residual of decomposition and is assumed to be i.i.d $N(0, \sigma^2)$. Clearly, this is the same as (3.1). Based on equation (3.2), $S_l(u, v)$ is fully determined by the joint distribution of $\boldsymbol{X}_l(u)$, $\boldsymbol{X}_l(v)$ and $\boldsymbol{D}_l$, and we can see that the dependence between different edges connected to the same node is captured by this structure. For example, in lth latent source, two edges connected to node v are $S_l(u_1, v) = \boldsymbol{X}_l(u_1)'\boldsymbol{D}_l\boldsymbol{X}_l(v)$ and $S_l(u_2, v) = \boldsymbol{X}_l(u_2)'\boldsymbol{D}_l\boldsymbol{X}_l(v)$ which share the same component $\boldsymbol{X}_l(v)$ to borrow information from each other. Therefore, such low-rank representation not only save the number of parameters in the model but also allow information to be shared across 'near-by' edges (edges from the same node). Next, following the our previous work (Guo, 2011; Wang and Guo, 2019), we assume each latent sources are independent, $f(\boldsymbol{X}_1, .., \boldsymbol{X}_q, \boldsymbol{D}_1, .., \boldsymbol{D}_q) = \prod_{l=1}^{q} f(\boldsymbol{X}_l, \boldsymbol{D}_l)$. Finally, different from existing works with separate priors on $\boldsymbol{D}_l$ and $\boldsymbol{X}_l$ (Park and Casella, 2008; Goh et al., 2017), the L1-based uniform sparsity is equivalent as forcing a joint prior on $\boldsymbol{D}_l$ and $\boldsymbol{X}_l$:

$$f(\boldsymbol{X}_l, \boldsymbol{D}_l) \propto \prod_{u<v} exp(-\tau|\sum_{r} \lambda_{l,r} X_{l,r}(u) X_{l,r}(v)|). \tag{3.6}$$

To show the equivalence between this Bayesian model with Locus in (3.4), we see that the posterior distribution of $[\boldsymbol{A}, \boldsymbol{X}_1, \boldsymbol{D}_1, ..., \boldsymbol{X}_q, \boldsymbol{D}_q|\boldsymbol{Y}]$ is proportional to $\prod_i g(\boldsymbol{Y}_i; \boldsymbol{a}_i\boldsymbol{S}, \sigma^2\boldsymbol{I}_p)$ $\prod_l f(\boldsymbol{X}_l, \boldsymbol{D}_l)$, where $g(\cdot)$ denotes the pdf of multivariate Gaussian distribution, $\boldsymbol{S} \equiv [\mathcal{L}(\boldsymbol{X}_1\boldsymbol{D}_1\boldsymbol{X}_1')$,..., $\mathcal{L}(\boldsymbol{X}_q\boldsymbol{D}_q\boldsymbol{X}_q')]$ and $\boldsymbol{a}_i$ has non-informative prior.

Therefore, the objective function for MAP of this Bayesian model is

$$\sum_i -\frac{1}{\sigma^2}\|\boldsymbol{Y}_i - \boldsymbol{a}_i\boldsymbol{S}\|_2^2 - \tau \sum_{l=1}^{q} \sum_{v<u} |\boldsymbol{X}_l(v)'\boldsymbol{D}_l\boldsymbol{X}_l(u)| + C, \tag{3.7}$$

where C is the constant only involving σ and τ. Maximizing (3.7) is equivalent as minimizing (3.4) with properly selected tuning parameter. Therefore, we can see that one unique point of uniform sparsity penalization method is that it jointly models

all elements in $\boldsymbol{X}_l$ and $\boldsymbol{D}_l$ to quantify the interactions among them, rather than modeling each term independently like Bayesian lasso (Park and Casella, 2008) or Bayesian reduced-rank model (Goh et al., 2017).

3.2.4 Estimation

In this section, we will introduce the algorithm for solving the optimization problem (3.4) of Locus. First we want to point out that (3.4) is a non-convex optimization (as shown in appendix B). The key to solve this optimization problem is to explore the block multi-convex structure in the problem (Sun and Li, 2017).

First, we will introduce the data preprocessing step prior to LOCUS which is commonly used in ICA literature and can significantly simplify the complexity. After that, we will introduce the algorithm based on the block multi-convex property, and at last we will discuss about how to select the tuning parameter for Locus.

3.2.4.1 Preprocessing Step

Preprocessing steps such as centering, dimension reduction and whitening are commonly used prior to decomposition algorithm for facilitating the subsequent problem (Hyvärinen et al., 2001). Prior to Locus, we follow our previous preprocessing procedure to facilitate the decomposition algorithm (Guo and Tang, 2013; Wang and Guo, 2019; Shi and Guo, 2016; Guo, 2011; Guo and Pagnoni, 2008). First, for connectivity data decomposition, the mean connectivity pattern across subjects is not of interest. Therefore, we first center all columns of $\boldsymbol{Y}$. Next, we reduce the dimension and whiten the data with $\widetilde{\boldsymbol{Y}} = \boldsymbol{H}\boldsymbol{Y}$, where $\boldsymbol{H} = (\boldsymbol{\Lambda}_q - \widetilde{\sigma}_q^2\boldsymbol{I})^{-1/2}\boldsymbol{V}_q'$. $\boldsymbol{V}_q$ and $\boldsymbol{\Lambda}_q$ contains the first q eigenvectors and eigenvalues based on singular value decomposition of $\boldsymbol{Y}$. The residual variance, $\widetilde{\sigma}_q^2$, is the average of the smallest $N - q$ eigenvalues that are not included in $\boldsymbol{\Lambda}_q$, representing the variability in $\boldsymbol{Y}$ that is not accounted by the first q components. The parameter q, which is the number of

sources, can be determined using the Laplace approximation method (Minka, 2000). Here $\widetilde{\boldsymbol{Y}} = [\widetilde{\boldsymbol{y}}(1,2), \widetilde{\boldsymbol{y}}(1,3), ..., \widetilde{\boldsymbol{y}}(V-1,V)]$ is of dimension $q \times V(V-1)/2$ where each column represents one connection. The problem (3.4) for the preprocessed data turns into

$$\min \sum_{i=1}^{q} \|\widetilde{\boldsymbol{Y}}_i - \sum_{l=1}^{q} \widetilde{a}_{il} \mathcal{L}(\boldsymbol{X}_l \boldsymbol{D}_l \boldsymbol{X}_l')\|_2^2 + \phi \sum_{l=1}^{q} \sum_{u<v} |\boldsymbol{X}_l(u)' \boldsymbol{D}_l \boldsymbol{X}_l(v)|. \tag{3.8}$$

Here we denote $\widetilde{\boldsymbol{A}} = \{\widetilde{a}_{il}\} \in \mathcal{R}^{q \times q}$ which is $\boldsymbol{HA}$, also known as mixing matrix in the literature, and is commonly assumed as orthogonal to facilitate the algorithm (Hyvärinen and Oja, 2000; Beckmann and Smith, 2004; Wang and Guo, 2019).

3.2.4.2 Algorithm for Solving Locus

In this section, we will present the algorithm to solve (3.8), which is to estimate the unknown parameters $\Theta = \{\widetilde{\boldsymbol{A}}, \boldsymbol{X}_1, .., \boldsymbol{X}_q, \boldsymbol{D}_1, .., \boldsymbol{D}_1\}$. We note that the problem has a block multi-convex structure in it where the problem is convex in $\boldsymbol{X}_l(v)$ when we condition on all other parameters. Based on this property we propose the following updating algorithm.

Overall, at tth iteration, the algorithm contains two major steps on latent sources and loading matrix:

- Update latent coordinate and weights $[\boldsymbol{X}_l, \boldsymbol{D}_l]$ for each basis $l = 1, ..., q$ given $\widetilde{\boldsymbol{A}}$:

$$[\hat{\boldsymbol{X}}_l^{(t)}, \hat{\boldsymbol{D}}_l^{(t)}] = \text{argmin}_{\boldsymbol{X}_l, \boldsymbol{D}_l} \left\| \widetilde{\boldsymbol{A}}_l^{(t-1)\prime} \widetilde{\boldsymbol{Y}} - \mathcal{L}(\boldsymbol{X}_l \boldsymbol{D}_l \boldsymbol{X}_l')' \right\|_2^2 + \phi \sum_{u<v} |\boldsymbol{X}_l(u)' \boldsymbol{D}_l \boldsymbol{X}_l(v)|, \tag{3.9}$$

  where $\widetilde{\boldsymbol{A}}_l^{(t-1)}$ is the estimated vector at $(t-1)$th iteration for lth column of $\widetilde{\boldsymbol{A}}$.

- Update $\widetilde{\boldsymbol{A}}$ with $\widetilde{\boldsymbol{A}}^{(t)} = \mathcal{H}\left(\widetilde{\boldsymbol{Y}} \hat{\boldsymbol{S}}^{(t)\prime} (\hat{\boldsymbol{S}}^{(t)} \hat{\boldsymbol{S}}^{(t)\prime})^{-1}\right)$ where $\mathcal{H}$ is the orthogonal transformation and $\hat{\boldsymbol{S}}^{(t)} = [\mathcal{L}(\hat{\boldsymbol{X}}_1^{(t)} \hat{\boldsymbol{D}}_1^{(t)} \hat{\boldsymbol{X}}_1^{(t)\prime}), ..., \mathcal{L}(\hat{\boldsymbol{X}}_q^{(t)} \hat{\boldsymbol{D}}_q^{(t)} \hat{\boldsymbol{X}}_q^{(t)\prime})]'$.

We note that (3.9) holds because of the orthogonality on $\widetilde{\boldsymbol{A}}$, where $\|\widetilde{\boldsymbol{Y}}(u,v) - \widetilde{\boldsymbol{A}}\boldsymbol{S}(u,v)\|_2^2 = \|\widetilde{\boldsymbol{A}}'\widetilde{\boldsymbol{Y}}(u,v) - \boldsymbol{S}(u,v)\|_2^2$ when $\widetilde{\boldsymbol{A}}$ is orthogonal ($\boldsymbol{S}(u,v) = [\boldsymbol{X}_1(u)'\boldsymbol{D}_l\boldsymbol{X}_1(v), ..., \boldsymbol{X}_q(u)'\boldsymbol{D}_q\boldsymbol{X}_q(v)]'$). Therefore, (3.9) separates the latent sources out and updates each latent source one by one. However, (3.9) is still non-convex and solving it is not trivial. In the following subsection, we will propose a node-moving algorithm to solve (3.9) which takes advantage of the block multi-convex property in the problem by breaking down the original problem into multiple convex sub-problems.

Node-Moving Algorithm:

Although directly solving $[\boldsymbol{X}_l, \boldsymbol{D}_l]$ in (3.9) is challenging, when we narrow down to a single node coordinate across all pathway $\boldsymbol{X}_l(v)$ and condition on the weights $\boldsymbol{D}_l$ and other node coordinates $\boldsymbol{X}_l(-v)$, the node-v-specific sub-problem turns to be convex:

$$\hat{\boldsymbol{X}}_l^{(t)}(v) = \text{argmin}_{\boldsymbol{x}\in\mathcal{R}^{R_l}} \left\|\widetilde{\boldsymbol{Y}}(-v,v)'\widetilde{\boldsymbol{A}}_l^{(t-1)} - \hat{\boldsymbol{X}}_l^{(t-1)}(-v)\hat{\boldsymbol{D}}_l^{(t-1)}\boldsymbol{x}\right\|_2^2 + \phi \sum_{u=1,u\neq v}^{V} |\hat{\boldsymbol{X}}_l{}^{(t-1)}(u)'\hat{\boldsymbol{D}}_l{}^{(t-1)}\boldsymbol{x}|, \tag{3.10}$$

where $\widetilde{\boldsymbol{Y}}(-v,v) = [\widetilde{\boldsymbol{y}}(1,v), ..., \widetilde{\boldsymbol{y}}(v-1,v), \widetilde{\boldsymbol{y}}(v,v+1), ..., \widetilde{\boldsymbol{y}}(v,V)]$ is of dimension $q \times (V-1)$ which is the sub-matrix of $\widetilde{\boldsymbol{Y}}$ with only the columns having edges connected to node v, and $\boldsymbol{X}_l(-v)$ is $\boldsymbol{X}_l$ with vth row removed. One way to understand this is that to update the coordinates for node v, we need all connections to node v which is $\widetilde{\boldsymbol{Y}}(-v,v)$. A detailed derivation between (3.9) and (3.10) is shown in Appendix B. Based on (3.10), we see that after narrowing into only one node and conditioning on others, the loss function is free of tensor product operator and $\mathcal{L}$ function. The first part of (3.10) is same as a standard regression-based least square problem and is easy to solve. The penalty part in (3.10) turns to be a group-wise penalization with $V-1$ terms of weights $\boldsymbol{X}_l(u)\boldsymbol{D}_l$, which is same as the adaptive-weighting for different pathway. Geometrically, after we fixed the latent node coordinates for other

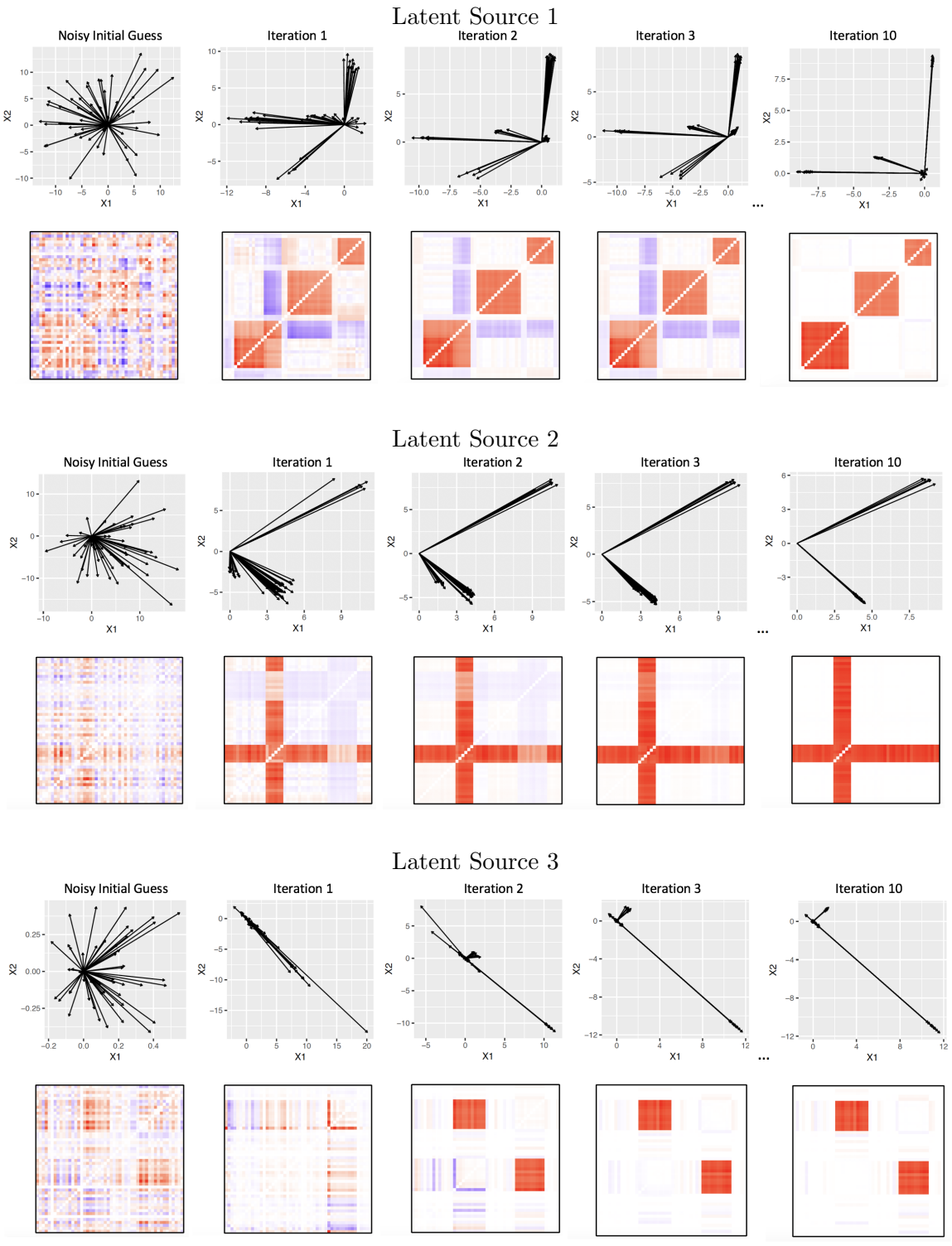

Figure 3.3: Illustration of the Node Moving algorithm until the 10th iteration for Locus method based on a simulated dataset from the setting 1 with middle level variance and 100 samples. The algorithm starts with a noisy estimate which can hardly show the pattern, and after 10 iterations $\boldsymbol{X}_l(v)$'s are grouped into several clusters and those clusters are becoming orthogonal with each other, resulting in a sparse and low-rank latent sources.

nodes, we are actually moving node coordinate $\boldsymbol{X}_l(v)$ to be as orthogonal as others adjusted by the weights $\boldsymbol{D}_l$. We do this movement for all node coordinates one by one. This is the reason why this is named Node-Moving algorithm. Figure 3.3 shows a simulated example about how $\boldsymbol{X}_l(v)$'s are being grouped into several clusters and those clusters are becoming more and more orthogonal with each other reflecting a sparse and low-rank latent source. We note that (3.10) can be easily solved based on existing numerical optimization tools. But, in the case that the rank of $\boldsymbol{X}_l(-v)$ is R_l, (3.10) can be further decomposed into

$$\widetilde{\boldsymbol{S}}_l^{(t)}(-v,v) = \text{argmin}_{\boldsymbol{S}\in\mathcal{R}^{V-1}} \left\| \widetilde{\boldsymbol{Y}}(-v,v)'\hat{\boldsymbol{A}}_l^{(t-1)} - \boldsymbol{S} \right\|_2^2 + \phi \sum_{i=1}^{V-1} |S_i|, \tag{3.11}$$

$$\hat{\boldsymbol{X}}_l^{(t)}(v) = \hat{\boldsymbol{D}_l}^{-1}(\hat{\boldsymbol{X}}_l(-v)'\hat{\boldsymbol{X}}_l(-v))^{-1}\hat{\boldsymbol{X}}_l(-v)'\widetilde{\boldsymbol{S}}_l^{(t)}(-v,v). \tag{3.12}$$

Here, (3.11) estimates the $V-1$ edges connected to node v with desired sparseness property, which is equivalent to model each edges separately. (3.12) maps $\widetilde{\boldsymbol{S}}_l^{(t)}(-v,v)$ back to low-rank space expanded by the row space of $\boldsymbol{X}_l(-v)$ and further reshaped by directional matrix $\boldsymbol{D}_l^{-1}$. (3.11) has an explicit solution as shown in Fan and Li (2001). After looping across all nodes, we need to scale each column $\boldsymbol{X}_l^{(t)}$ to have L2 norm as 1.

The last step is to update $\boldsymbol{D}_l$ for $l = 1, ..., q$. Conditioned on $\boldsymbol{X}_l^{(t)}$ and $\hat{\boldsymbol{A}}_l^{(t-1)}$, the optimization problem for $\boldsymbol{D}_l$ turns into:

$$\text{diag}(\hat{\boldsymbol{D}}_l^{(t)}) = \text{argmin}_{\boldsymbol{x}\in\mathcal{R}^{R_l}} \left\| \widetilde{\boldsymbol{Y}}'\hat{\boldsymbol{A}}_l^{(t-1)} - \boldsymbol{Z}_l^{(t)}\boldsymbol{x} \right\|_2^2 + \phi \sum_{u<v} |\boldsymbol{Z}_l^{(t)}(u,v)\boldsymbol{x}|, \tag{3.13}$$

where $\boldsymbol{Z}_l^{(t)}$ is of dimension $p \times R_l$, the rth column of $\boldsymbol{Z}_l^{(t)}$ is $\mathcal{L}(\boldsymbol{X}_{l,r}^{(t)}\boldsymbol{X}_{l,r}^{(t)\prime})$ with $\boldsymbol{X}_{l,r}^{(t)}$ to be the rth column of $\boldsymbol{X}_l^{(t)}$, and $\boldsymbol{Z}_l^{(t)}(u,v) = [X_{l,1}^{(t)}(v)X_{l,1}^{(t)}(u), ..., X_{l,R_l}^{(t)}(v)X_{l,R_l}^{(t)}(u)]$. In fact, $\sum_{u<v} |\boldsymbol{Z}_l^{(t)}(u,v)\boldsymbol{x}| = \|\boldsymbol{Z}_l^{(t)}\boldsymbol{x}\|_1$. It is clear that, this problem has the same formation as (3.10) which can be solved similarly. Now, we have finished the Node-

Moving algorithm.

The detailed algorithm for learning Locus is summarized in Algorithm 2. The stopping criteria is based on $\widetilde{\boldsymbol{A}}$ and $\boldsymbol{S}$ instead of $\boldsymbol{X}_l$ and $\boldsymbol{D}_1$ in our model. After the algorithm converges, we can estimate the loading matrix based on $\hat{\boldsymbol{A}} = \boldsymbol{Y}\hat{\boldsymbol{S}}'(\hat{\boldsymbol{S}}'\hat{\boldsymbol{S}})^{-1}$, which finished estimating all components in Locus. We also note that selection of low-rank dimensions for each basis requires many hyper-parameters to tune, $R_1, ..., R_q$. In the next section, we will discuss an approach to automatically select R_l for each basis.

Algorithm 2 Node-Moving Algorithm for Learning LOCUS

Tuning Parameter: Select hyper-parameters $q, \phi, R_1, ..., R_q$.
Initial: Preprocess the data $\widetilde{\boldsymbol{Y}} = \boldsymbol{H}\boldsymbol{Y}$, and initialize $\widetilde{\boldsymbol{A}}^{(0)}, \boldsymbol{X}_l^{(0)}, \boldsymbol{D}_l^{(0)}$, for $l = 1, ..., q$.
repeat
  Update $[\boldsymbol{X}_l, \boldsymbol{D}_l]$:
  For all $l = 1...q, v = 1...V$,
  1. Estimate $V-1$ edges in source l connected to node v, $\widetilde{\boldsymbol{S}}_l^{(t)}(-v, v)$, via (3.11);
  2. Calculate $\hat{\boldsymbol{X}}_l^{(t)}(v)$ by projecting $\hat{\boldsymbol{S}}_l^{(t)}(-v, v)$ onto low-rank space via (3.12).
  For all $l = 1...q$,
  3. Update $\boldsymbol{D}_l^{(t)}$ based on (3.13);
  4. Calculate $\hat{\boldsymbol{S}}^{(t)} = [\mathcal{L}(\hat{\boldsymbol{X}}_1\hat{\boldsymbol{D}}_1^{(t)}\hat{\boldsymbol{X}}_1'), ..., \mathcal{L}(\hat{\boldsymbol{X}}_q\hat{\boldsymbol{D}}_q^{(t)}\hat{\boldsymbol{X}}_q')]$.
  Update $\widetilde{\boldsymbol{A}}$:
  1. $\widetilde{\boldsymbol{A}}^{(t)} = \widetilde{\boldsymbol{Y}}\hat{\boldsymbol{S}}^{(t)\prime}(\hat{\boldsymbol{S}}^{(t)\prime}\hat{\boldsymbol{S}}^{(t)})^{-1}$
  2. Conduct orthogonal transformation on $\widetilde{\boldsymbol{A}}^{(t)}$.
until $\frac{\|\widetilde{\boldsymbol{W}}^{(t)} - \widetilde{\boldsymbol{W}}^{(t-1)}\|_F}{\|\widetilde{\boldsymbol{W}}^{(t-1)}\|_F} < \epsilon_1$ and $\frac{\|\hat{\boldsymbol{S}}^{(t)} - \hat{\boldsymbol{S}}^{(t-1)}\|_F}{\|\hat{\boldsymbol{S}}^{(t-1)}\|_F} < \epsilon_2$

Finally, We note that the Node-Moving algorithm is based on the orthogonality assumption on $\widetilde{\boldsymbol{A}}$, which can significantly improve the computational complexity. But, in fact, the original optimization problem in 3.4 still has block multi-convex property and Node-Moving algorithm can be generalized for the original problem. Therefore, we summarize and generalize the block multi-convex property of Locus optimization problem into Theorem 3.2.1 and provide the detailed proof in Appendix B. This further confirms the property in Locus and justifies the advantage of Node-Moving

algorithm in efficiency.

Theorem 3.2.1 (Block Multi-Convex)**.** *If function f has the form of*

$$f(\boldsymbol{X}_1, ..., \boldsymbol{X}_q, \boldsymbol{D}_1, ..., \boldsymbol{D}_q, \boldsymbol{A}) = \left\| \boldsymbol{Y} - \boldsymbol{A} \begin{bmatrix} \mathcal{L}(\boldsymbol{X}_1 \boldsymbol{D}_1 \boldsymbol{X}_1')' \\ \vdots \\ \mathcal{L}(\boldsymbol{X}_q \boldsymbol{D}_q \boldsymbol{X}_q')' \end{bmatrix} \right\|_F^2 + \phi \sum_{l=1}^{q} |\mathcal{L}(\boldsymbol{X}_l \boldsymbol{D}_l \boldsymbol{X}_l')|_1,$$

where $\boldsymbol{Y}$ and ϕ are constants, then f is a non-convex but multi-block-convex function.

3.2.4.3 Tuning Parameter Selection

R_l is the number of pathways which controls the maximum information one latent source can store. Larger R_l always yields better fit to the data but can lead to over-fitting. With the help of sparse penalty directly on $\boldsymbol{S}$, one naive way is to specify a common large rank for all basis and penalize it via uniform sparseness. However, this can be computationally expensive and the numerical performance can be very unstable. For example, with a large R_1, $\boldsymbol{X}_l(-v)$ could be less than full rank, and the numerical benefits in (3.11-3.12) will no longer hold. On the other hand, a common rank for all basis is not realistic based on the real data where different basis has exhibited quite different properties. On the other hand, smaller R_l limits the information the basis can store and hence tends to have lower accuracy. To overcome these issues and simplify the human effort for picking R_l one by one, we proposed the following approach to automatically select R_l with a similar strategy as in PCA.

Instead of selecting all R_1 to R_q separately, we use a single parameter to automatically decide what R_l should be for each latent source. Define $\rho \in (0, 1)$ to be proportion of signal we aim to capture in each latent source for the low-rank decomposition, which is similar as the way for selecting the number of components in PCA. Specifically, when updating $\boldsymbol{S}$, we estimated $\widetilde{\boldsymbol{S}}$ which is prior to low-rank mapping. The difference between $\boldsymbol{S}_l$ and $\widetilde{\boldsymbol{S}}_l$ represents the limits for current R_l. The larger

R_l, smaller the difference should be. Therefore, we adaptively change R_l to be the smallest integer such that $\|\boldsymbol{S}_l - \widetilde{\boldsymbol{S}}_l\|/\|\widetilde{\boldsymbol{S}}_l\| \leq 1 - \rho$. Finally we can select ρ and ϕ based on commonly used BIC approach (Zhou et al., 2013; Sun and Li, 2017; Wang et al., 2016).

3.3 Simulation Study

In this section, we investigate the performance of our model based on simulation study, and demonstrate its superior performance compared with alternative approaches.

3.3.1 Synthetic Data

Motivated by the findings from real data application as shown in Figure (3.2), we first simulate the latent source signal with diagonal block shape, crossing shape and off-diagonal block shape as shown in Figure 3.4 (A). We further point out that these patterns are commonly existed in the real data (Amico et al., 2017; Amico and Goñi, 2018a) and hence serve as a more realistic setting to brain imaging applications. Moreover, the patterns in Figure 3.4 (A) are well supported with the low-rank structure. To fully evaluate the performance of our model, we consider a more challenging setting for the latent source signal with diagonal triangle shape, off-diagonal circle shape and long-range hollow square shape shown in in Figure 3.4 (B). In general, this is a more difficult pattern for the model with low-rank structure as shown in Zhou et al. (2013) which usually needs a larger number of pathway to approximate the shape. In both settings, we have $V = 50$ and $q = 3$. We consider two sample sizes, $N = 50, 100$. To generate the loading matrix, we sampled without replacement from the estimated loading matrix based on InfoMax ICA and PNC study with 515 subjects. At last, we add Gaussian noise to the data with 3 levels: low variance = 1, mid variance = 3^2 and high variance = 6^2. We note that based on the real data, after controlling on the loading and source matrices to have the same signal levels,

the variance for the noise is close to 3.24 which is covered within our settings. In total, we have $2 \times 2 \times 3$ settings being considered and for each setting we generate 100 simulation runs to quantify the uncertainty level.

3.3.2 Simulation Specification

We compared the performance of Locus with the commonly used connICA approach based on FastICA and InfoMax. To understand the performance of symmetric low-rank decomposition, we further compare it with the dictionary learning (DL) method with L1 penalty. Moreover, to better understand the effectiveness of uniform sparsity, we further compare it with the two commonly used penalization approach: element-wise penalty and low-rankness:

$$\min \sum_{i=1}^{N} \|\boldsymbol{Y}_i - \sum_{l=1}^{q} a_{il}\mathcal{L}(\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_l')\|_2^2 + \phi \sum_{l=1}^{q} \|\boldsymbol{X}_l\|_1, \tag{3.14}$$

$$\min_{\boldsymbol{S}_l \in \mathcal{R}^{V\times V}, \boldsymbol{S}_l = \boldsymbol{S}_l'} \sum_{i=1}^{N} \|\boldsymbol{Y}_i - \sum_{l=1}^{q} a_{il}\mathcal{L}(\boldsymbol{S}_l)\|_2^2 + \phi \sum_{l=1}^{q} \|\boldsymbol{S}_l\|_*, \tag{3.15}$$

where $\|\cdot\|_*$ is the nuclear norm, and in the following sections we denote (3.14) as Loc-L1 and (3.15) as Loc-Nuclear. In simulation, BIC is used to select the tuning parameter for the methods with hyper-parameter.

3.3.2.1 Evaluating Metrics

Following our previous work, we used Pearson correlation as the metric to evaluate methods' signal recovery ability. Specifically, we first conduct a source-matching procedure and then calculate the Pearson correlation between the truth and the model-based estimates on latent sources and loading matrices (Beckmann and Smith, 2004; Wang and Guo, 2019). We further calculate the standard deviation of these correlations across 100 simulation runs to evaluate the robustness. Results are summarized in Tables 3.1 and 3.2. Figures 3.6, 3.7 visualize the averaged correlation and its

standard deviation across 100 runs on latent sources for setting 1 and 2.

More importantly, we evaluated methods' reproducibility on latent sources based on 2 adjusted summary statistics with Pearson correlation and Jaccard Index. Specifically, the reproducibility on lth latent source is defined as:

$$r_l = \frac{\frac{1}{B}\sum_{b=1}^{B} \max_{i\in\{1,..,q\}}\{h(\boldsymbol{S}_l, \boldsymbol{S}_i^{(b)})\} - \frac{1}{Bq}\sum_{b=1}^{B}\sum_{i=1}^{q} h(\boldsymbol{S}_l, \boldsymbol{S}_i^{(b)})}{1 - \frac{1}{Bq}\sum_{b=1}^{B}\sum_{i=1}^{q} h(\boldsymbol{S}_l, \boldsymbol{S}_i^{(b)})}, \tag{3.16}$$

where $B = 100$ is the number of simulation runs, $\boldsymbol{S}_l$ is the true lth latent source, $\boldsymbol{S}_i^{(b)}$ is the ith latent source for bth simulation run, $h(\cdot)$ is the measure of similarity which is set to be Pearson correlation or Jaccard Index. We note (3.16) is related to Cohen's kappa coefficient where we adjust the measure based on baseline effect. Figures 3.8, 3.9 visualize the averaged reproducibility across all latent sources for setting 1 and 2.

Finally, to better understand the methods' performance on latent sources, we randomly selected 4 simulated datasets with mid level variance for visualization. Specifically, for these selected runs, the estimated latent sources are visualized in Figures 3.5,3.10.

3.3.3 Simulation Results

Locus: First, based on the results in Tables 3.1 and 3.2, our method provides more accurate estimates for latent sources and subject-specific loadings compared with other methods in both settings. Moreover, based on Figures 3.8 and 3.9, Locus provides more reproducible results on latent sources compared with other methods in both settings. Figures 3.5 and 3.10 visualized the estimated latent sources from 4 randomly selected simulation runs. These figures showed that Locus can more accurately identified the connections that are activated in latent sources. Moreover, another impressive results of Locus is that it can accurately penalize the connectivity out of activated region to zero as shown in Figures 3.5 and 3.10 which is a very helpful

property for selecting the connectivities of interest. On the other hand, Locus has much smaller variability across simulation runs which indicates that Locus is more stable and can potential have better reproducibility for application.

ConnICA: Based on our simulation study, both FastICA and InfoMax have lower performance compared with Locus in terms of correlation on latent sources and loading matrix. Specifically, FastICA performs poorly when latent sources follow the low-rank block-shape structure motivated by the real brain data, and InfoMax has a lower performance than FastICA when latent sources have the non-square shape as in setting 2. We also found that InfoMax has a smaller SD than FastICA, indicating that InfoMax could have a more robust estimation. As shown in Figures 3.5, 3.10 (B, C), the results from connICA methods are all very noisy, and their estimated latent sources are mixed up with each other. Moreover, as clearly shown by the intensity level plot in Figures 3.5, 3.10 (B, C), it is hard to select appropriate threshold for the results of connICA in most cases when signals are mixed together.

Effects of Penalty: First, as we expected and discussed above, Loc-Nuclear cannot ensure sparsity where the results are noisy compared with Locus. Moreover, for the second latent source setting, Loc-Nuclear failed to recover the third latent source as shown in Figure 3.10 (F), which indicate that low-rankness might not be sufficient enough for recovering more complicated latent sources. On the other hand, the performance of Loc-L1 is even worse. As shown in Figure 3.5 and 3.10 (E), although Loc-L1 can provide sparse estimation, the estimated latent sources contain structured noise. For the first block-shape setting, the estimated latent sources from Loc-L1 contain block-shape noises. As shown in Figure 3.5 (E), such structured noise can be hardly distinguished from the real underlying pattern. In the second latent source setting, the problem is even more serious where the estimated latent sources contains weird low-rank structure which are more clear in the intensity plots Figure 3.10 (E). These results reveals the weakness for element-wise penalty - the penalization on each

Table 3.1: Simulation results for comparing Locus with other methods with 100 simulation runs for the first setting. Values presented are mean and standard deviation of correlations between true and estimated: latent sources and loading matrices for the first setting.

Term	N	Var.	Locus	ConnICA FastICA	ConnICA InfoMax	DL	Loc L1	Loc Nuclear
Latent Source Corr. (SD)	50	Low	0.998 (0.001)	0.818 (0.043)	0.953 (0.001)	0.860 (0.004)	0.873 (0.086)	0.970 (0.001)
		Mid	0.986 (0.004)	0.777 (0.029)	0.871 (0.005)	0.876 (0.026)	0.851 (0.048)	0.888 (0.004)
		High	0.887 (0.032)	0.599 (0.050)	0.699 (0.021)	0.686 (0.026)	0.799 (0.050)	0.716 (0.020)
	100	Low	0.999 (0.001)	0.823 (0.046)	0.959 (0.001)	0.973 (0.010)	0.928 (0.097)	0.977 (0.001)
		Mid	0.995 (0.001)	0.805 (0.044)	0.918 (0.002)	0.941 (0.012)	0.843 (0.130)	0.936 (0.002)
		High	0.964 (0.011)	0.722 (0.037)	0.811 (0.008)	0.821 (0.012)	0.818 (0.055)	0.830 (0.007)
Loading Matrix Corr. (SD)	50	Low	0.995 (0.001)	0.860 (0.046)	0.932 (0.002)	0.912 (0.007)	0.930 (0.055)	0.966 (0.002)
		Mid	0.997 (0.001)	0.855 (0.030)	0.928 (0.009)	0.929 (0.037)	0.968 (0.015)	0.962 (0.007)
		High	0.958 (0.013)	0.770 (0.060)	0.880 (0.036)	0.845 (0.036)	0.928 (0.048)	0.899 (0.032)
	100	Low	0.999 (0.001)	0.833 (0.049)	0.925 (0.001)	0.960 (0.013)	0.910 (0.067)	0.959 (0.001)
		Mid	0.997 (0.001)	0.837 (0.042)	0.927 (0.005)	0.967 (0.017)	0.939 (0.056)	0.960 (0.004)
		High	0.984 (0.002)	0.830 (0.038)	0.913 (0.014)	0.920 (0.014)	0.930 (0.052)	0.941 (0.011)

coordinate cannot guarantee the desired property on constructed latent sources. This further confirms the superiority of our proposed penalization method in decomposing the connectivity matrices. Finally, as for DL, although the estimated latent sources for setting 1 have good alignment with the truth, in the setting 2 the estimated latent sources are still mixed with each other. Moreover, as shown in Figures 3.5, 3.10 (D), the estimated latent sources of DL are noisy even with L1 penalty. This is partially caused by the selection of tuning parameter which favors the BIC criteria, and another reason might be that only the L1 penalty without low-rank structure tends to over-parameterize the model and hence produce noisy results.

Table 3.2: Simulation results for comparing Locus with other methods with 100 simulation runs for the second setting. Values presented are mean and standard deviation of correlations between true and estimated: latent sources and loading matrices for the second setting.

Term	N	Var.	Locus	ConnICA FastICA	ConnICA InfoMax	DL	Loc L1	Loc Nuclear
Latent Source Corr. (SD)	50	Low	0.961 (0.001)	0.943 (0.046)	0.918 (0.001)	0.931 (0.001)	0.831 (0.012)	0.944 (0.001)
		Mid	0.954 (0.014)	0.870 (0.035)	0.842 (0.005)	0.832 (0.026)	0.741 (0.052)	0.869 (0.005)
		High	0.811 (0.036)	0.699 (0.030)	0.659 (0.016)	0.620 (0.023)	0.688 (0.039)	0.683 (0.019)
	100	Low	0.962 (0.001)	0.949 (0.048)	0.923 (0.001)	0.836 (0.003)	0.835 (0.008)	0.950 (0.001)
		Mid	0.960 (0.002)	0.905 (0.044)	0.883 (0.003)	0.819 (0.011)	0.798 (0.035)	0.911 (0.003)
		High	0.939 (0.015)	0.801 (0.037)	0.768 (0.008)	0.685 (0.020)	0.694 (0.043)	0.800 (0.008)
Loading Matrix Corr. (SD)	50	Low	0.996 (0.001)	0.964 (0.041)	0.906 (0.001)	0.899 (0.001)	0.911 (0.007)	0.942 (0.001)
		Mid	0.992 (0.018)	0.943 (0.037)	0.911 (0.004)	0.880 (0.028)	0.887 (0.059)	0.947 (0.004)
		High	0.899 (0.041)	0.903 (0.045)	0.862 (0.025)	0.773 (0.033)	0.865 (0.058)	0.875 (0.027)
	100	Low	0.998 (0.001)	0.958 (0.048)	0.881 (0.001)	0.801 (0.004)	0.901 (0.005)	0.919 (0.001)
		Mid	0.996 (0.001)	0.949 (0.049)	0.885 (0.003)	0.820 (0.014)	0.870 (0.073)	0.924 (0.003)
		High	0.967 (0.030)	0.934 (0.050)	0.876 (0.009)	0.761 (0.026)	0.818 (0.050)	0.914 (0.008)

3.4 Application to rs-fMRI data from the Philadelphia Neurodevelopmental Cohort (PNC)

3.4.1 PNC Study and Data Description

The PNC is a collaborative project from the Brain Behavior Laboratory at the University of Pennsylvania and the Children's Hospital of Philadelphia (CHOP), funded by NIMH through the American Recovery and Reinvestment Act of 2009 (Satterthwaite, Elliott, Ruparel, Loughead, Prabhakaran, Calkins, Hopson, Jackson, Keefe, Riley et al., 2014; Satterthwaite, Wolf, Roalf, Ruparel, Erus, Vandekar, Gennatas, Elliott, Smith, Hakonarson et al., 2014). In this paper, we conducted the same rs-fMRI preprocessing procedure as in Wang et al. (2016), including skull stripping, signal stabilization, standard image registration, time series band filtering, spatial smoothing, motion correction and a validated confound regression procedure, where the preprocessing script is released from the 1000 Functional Connectomes Project. We further conducted a quality control based on displacement analysis in the same manner as Satterthwaite, Wolf, Roalf, Ruparel, Erus, Vandekar, Gennatas, Elliott, Smith, Hakonarson et al. (2014); Wang et al. (2016). The preprocessed rs-fMRI data were further mapped to the Power's node system (Power et al., 2011) including 264 nodes. To facilitate the understanding of the functional roles of the nodes, we assigned them to 10 functional networks corresponding to the major resting state networks (RSNs) described by Smith et al. (2009). 32 of the 264 nodes were not strongly associated with any RSN maps, and were therefore not included. At last, we constructed the brain connectivity matrix by calculating the Pearson's correlation of each pair of time series among the remaining 232 nodes. We further conducted a fisher z transformation to have the final functional connectivity matrices for 515 subjects of 26796 edges.

3.4.2 Connectivity Analysis for PNC Study

We applied our proposed model Locus and InfoMax-based connICA to the preprocessed functional connectivity data from PNC study. In the model, we set the number of latent sources to be 30 for both Locus and connICA. To compare the results from these 2 methods, we matched the estimated latent sources from Locus and connICA based on correlation. With a matching criteria at 0.6, we have 18 matched latent sources sources, parts of which are visualized in Figure 3.11. As expected, the estimated latent sources from Locus are much sparser and cleaner than those from connICA.

First, we studied the reproducibility of the results from Locus and compare with connICA. To evaluate whether the finding is reproducible, we randomly bootstrapped the subjects for 200 times and ran Locus and connICA for these bootstrapped samples. We used the reproducibility measures in (3.16) and also consider the same two adjusted summary statistics - Pearson correlation and Jaccard Index - to measure the reproducibility on latent sources. Specifically, $B = 200$ is the number of bootstrap, $\boldsymbol{S}_i^{(b)}$ is the ith latent source for bth bootstrap sample. Figure 3.12 shows the reproducibility from the 18 matched latent sources from connICA and our proposed method. Our method has significantly higher reproducibility compared with connICA in both Pearson correlation and Jaccard Index ($p = 0.005$ for Pearson, $p = 0.010$ for jaccard).

Moreover, among the 18 matched latent sources with connICA, 6 of them have a correlation-based reproducibility higher than 0.7 for Locus which are visualized in Figure 3.11. Figure 3.13 further shows the intensity level for these 6 high-reproducible latent sources across all edges. As shown by this Figure, these matched latent sources share similar patterns but the Locus provides much stronger signal to noise ratio where the intensity profile from connICA are much more noisy. Finally, we visualized

the top 1% edges for these 6 latent sources from Locus based on BrainNetViewer in Figure 3.14.

To further examine the difference between connICA and Locus, Figure 3.15 compared the difference between connICA and Locus for the 3 most correlated latent sources: Latent Source (LS) 1, 3 and 4. We also note that although LS3 from connICA and Locus are highly correlated ($r = 0.703$), the corresponding loading vectors are quite different in terms of its association with age. We found that the loading vector for LS3 of Locus is significantly associated with age ($p = 0.027$) while the loading vector for LS3 of connICA is not ($p = 0.952$). As shown in Figure 3.11, the activated regions for LS3 are within-cerebellum connections and the connections from cerebellum to rest of the brain. Given the limited number of nodes in cerebellum, the signals are relatively weaker where Locus showed its ability to remove the global noise while connICA cannot. As further shown in Figure 3.15, we threshold the latent source from Locus and connICA at 0.08, and examine the connectivities which are preserved by connICA but penalized by Locus (blue dots in the first row). We found that those connectivities for LS3 do not contain a clear pattern and also are not associated with cerebellum. Similar for LS1 and LS4. This further justifies that it is easier for Locus to remove the unrelated connectivity pattern. For connICA, to achieve the same sparsity, it will remove more informative connections.

On the other side, Figure 3.16 shows two latent sources estimated from Locus which has relatively high reproducibility based on Pearson correlation (LS7: 0.64; LS8: 0.55) but cannot be matched from connICA. Figure 3.17 visualized the top 1% edges of these 2 latent sources. These two latent sources contains some global connectivity patterns: LS7 contains the with-Lat Vis connections and the connections between Lat Vis and EC, FPL and DMN, which is significantly associated with gender ($p = 0.005$); LS8 contains the within-EC connections and the connections between EC and SM, Op Vis, Aud, which is significantly associated with age ($p < 0.001$).

3.5 Discussion

In this paper, we proposed a novel signal separation framework - Locus - specifically designed for decomposing the brain connectivity matrices. The proposed method simultaneously specified the low-rank structure for all underlying brain subnetworks, which saves a large number of parameters as well as allows the information to be shared for edges connected to the same node. Moreover, we proposed a novel sparsity penalization method for the proposed decomposition framework which not only ensures the desired sparsity on constructed latent sources and also gives analytic solution in the estimation algorithm. We point out that the proposed penalization approach is different from existing methods and is much more intuitive and achieved better numerical performance than low-rank penalty or element-wise penalty. Furthermore, we showed that the optimization problem of Locus is non-convex but exists the block multi-convex property via conditioning. By taking advantage of these nice properties, we proposed an efficient Node-Moving algorithm to estimate the unknown parameters in the model which provides analytic solutions at each step. Finally, we proposed a highly practical procedure to select the number of pathways in low-rank decomposition across all sub-networks, which significantly simplifies the workload for hyper-parameter tuning and provides a nice and intuitive interpretation of the selection procedure.

We evaluated the performance of the proposed method based on extensive simulation studies and compared it with the commonly used connICA framework based on InfoMax and FastICA in neuroscientific field. We also examined the performance of the proposed penalization approach with low-rank penalty and element-wise penalty based on simulation. Results show that our approach can recover the true underlying signal by penalizing the noise to nearly zero which cannot be achieved by existing methods. Based on our study, we found that the low-rank penalty cannot ensure spar-

sity while element-wise penalty gives biased estimation on the signal with structured noise.

The next step is to further evaluate method's reproducibility in different settings, such as the reproducibility based on different imaging acquisition approaches. We also point out that even though the proposed penalty utilized the L1 norm, other types of norms like L2, MCP, SCAD can also be directly applied. Moreover, the proposed Node-Moving algorithm can also provide analytic solution for those specifications. Furthermore, the proposed framework can be applied to any types of connectivity data with upper-triangular structure like structural connecitivty data from DTI or other functional connectivity measure such as mutual information or casuality measures. As for the proposed penaltization approach, it can also be extended to a more general tensor-decomposition task to ensure the sparisty on constructed tensor or signal from low-rank structure. For example, a regression setting with connectivity data can be directly explored. We will release the R package - Locus - to CRAN soon to include the current setting as well as other type of norms like SCAD and L2.

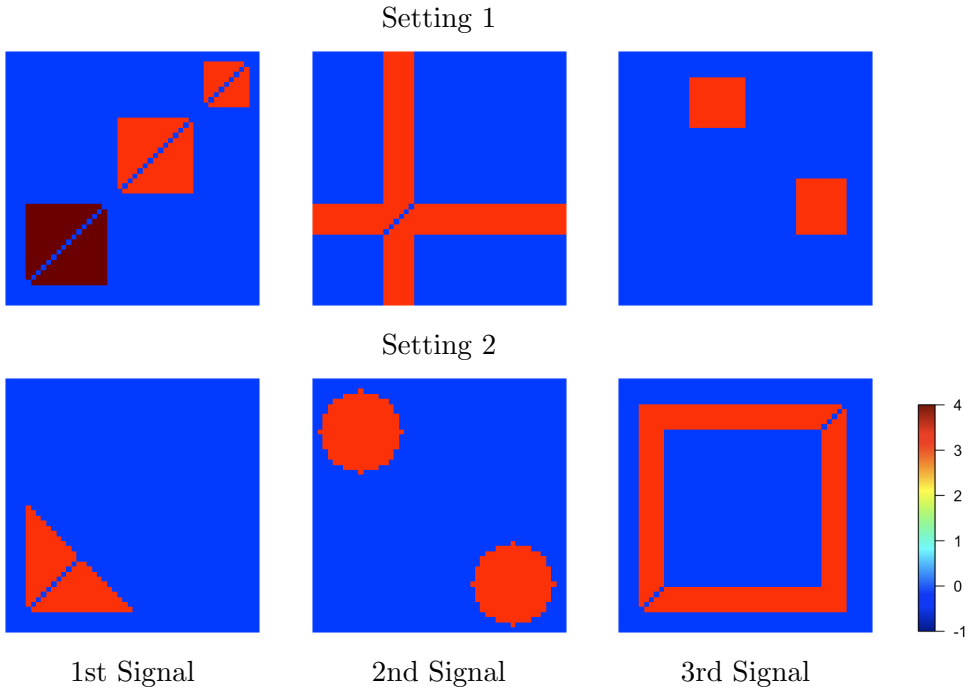

Figure 3.4: Generated underlying true source signals of 2 settings in the simulation study

Figure 3.5: Estimated latent signals of 4 randomly selected simulation runs in setting 1 across all methods. The first row of each panel is a direct visualization of estimated latent signal, and the second row of each panel is the trace plot of the estimated latent signal.

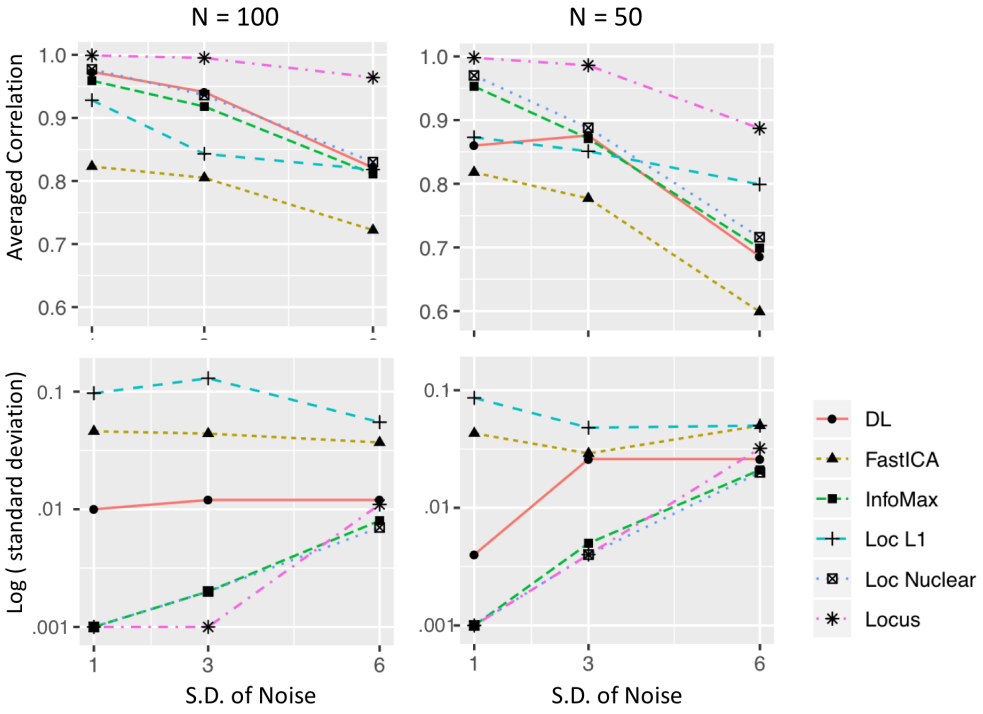

Figure 3.6: Simulation results of latent sources for comparing Locus with other methods across 100 simulation runs based on the first setting. The first row represents the averaged Pearson correlation between true and estimated latent sources. The second row represents the standard deviation of Pearson correlation between true and estimated latent sources in log scale.

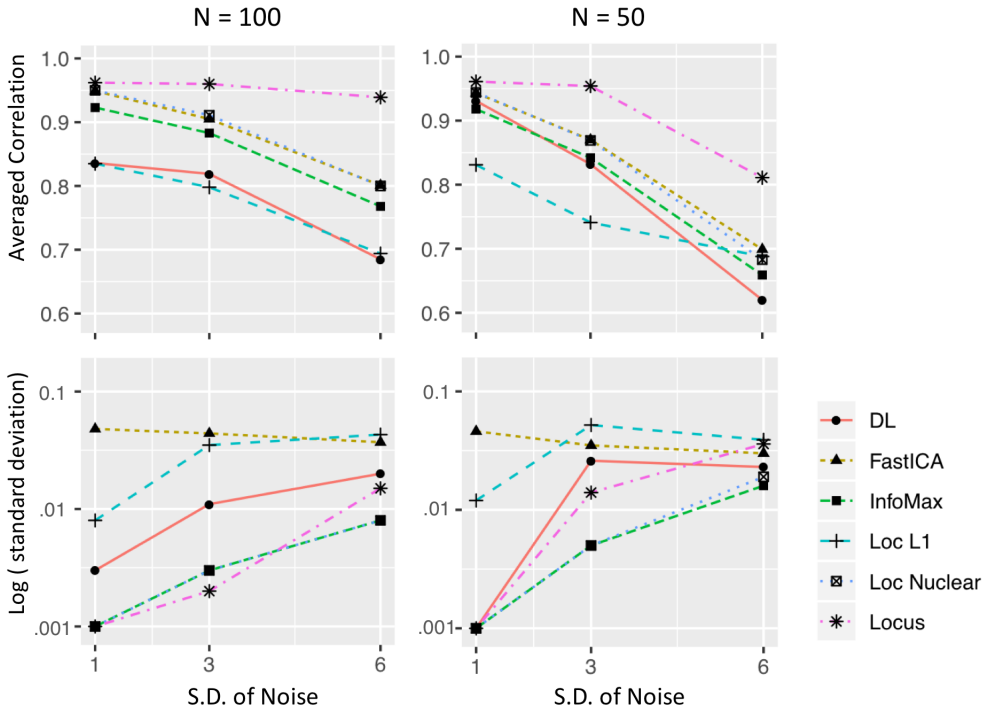

Figure 3.7: Simulation results of latent sources for comparing Locus with other methods across 100 simulation runs based on the second setting. The first row represents the averaged Pearson correlation between true and estimated latent sources. The second row represents the standard deviation of Pearson correlation between true and estimated latent sources in log scale.

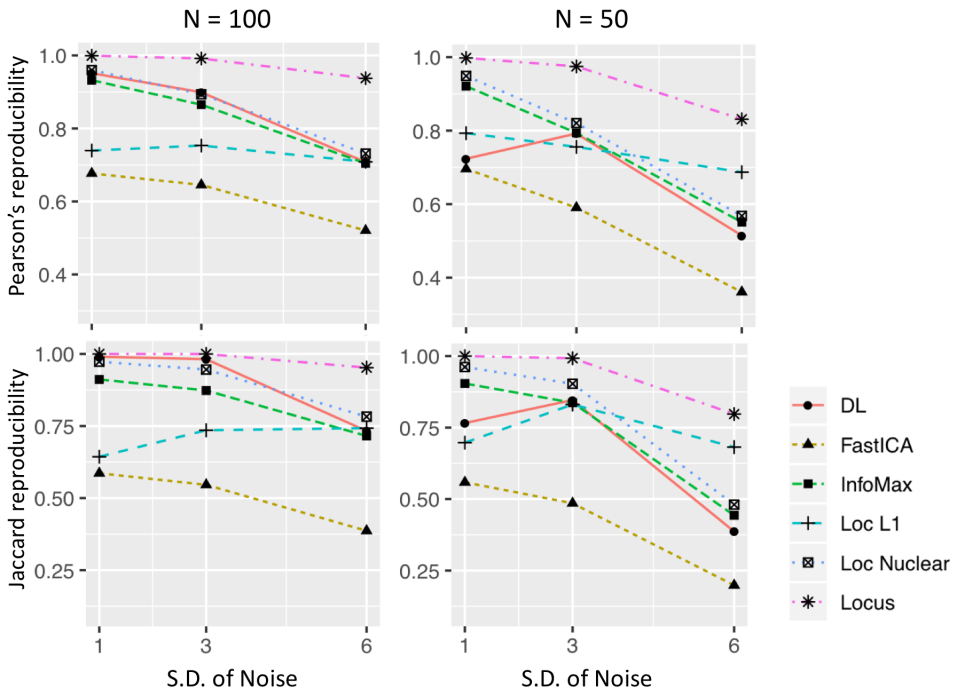

Figure 3.8: Simulation results of methods' reproducibility on latent sources for comparing Locus with other methods across 100 simulation runs based on the first setting. The first row represents the averaged adjusted Pearson correlation between true and estimated latent sources. The second row represents the averaged adjusted jaccard index between true and estimated latent sources.

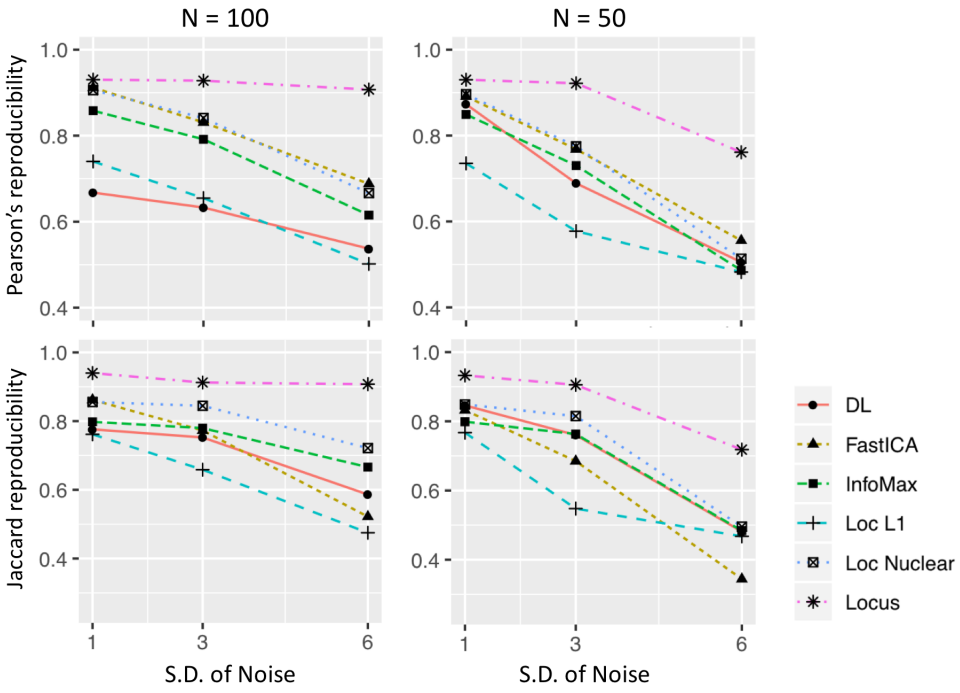

Figure 3.9: Simulation results of methods' reproducibility on latent sources for comparing Locus with other methods across 100 simulation runs based on the second setting. The first row represents the averaged adjusted Pearson correlation between true and estimated latent sources. The second row represents the averaged adjusted jaccard index between true and estimated latent sources.

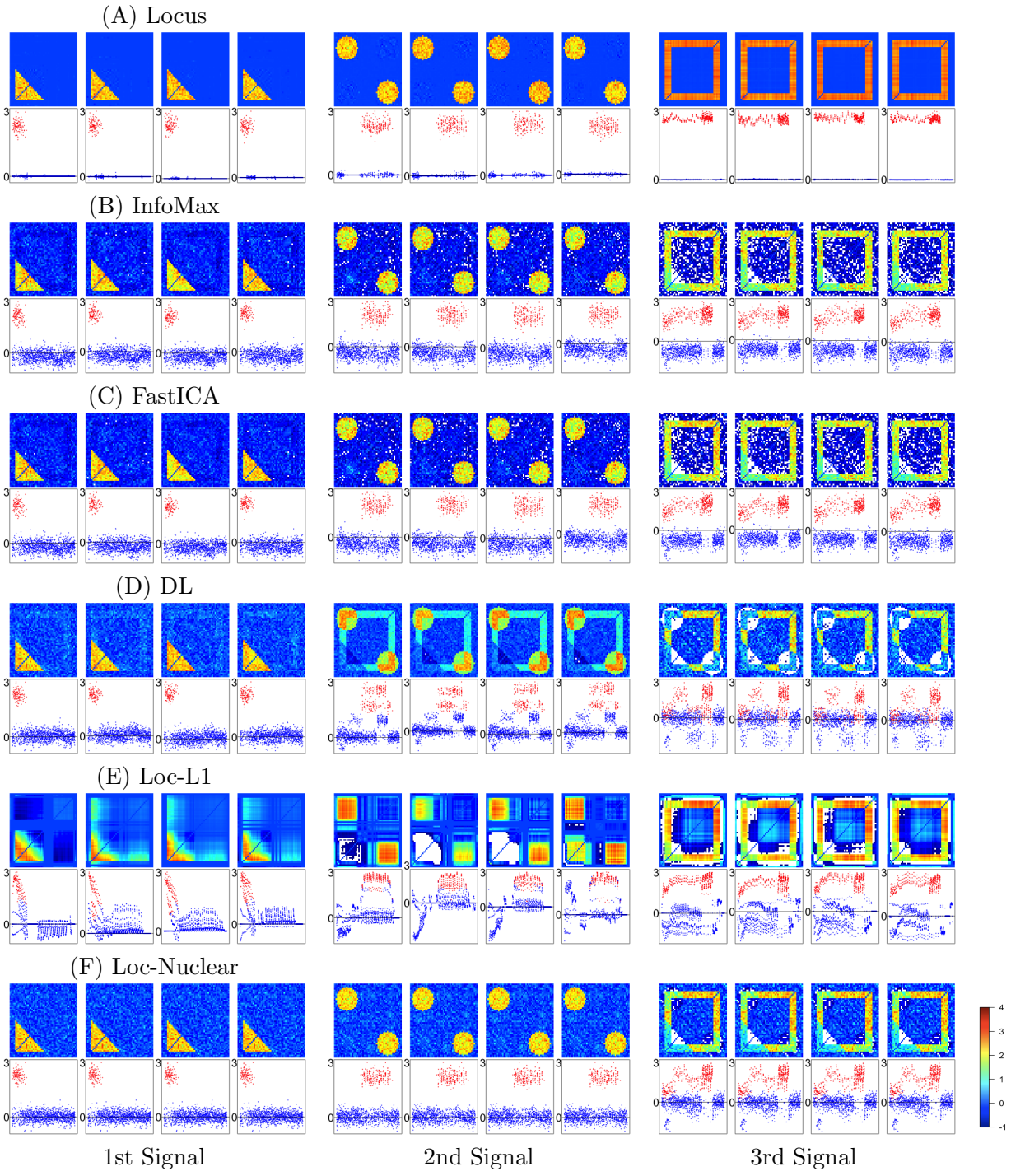

Figure 3.10: Estimated latent signals of 4 randomly selected simulation runs in setting 2 across all methods. The first row of each panel is a direct visualization of estimated latent signal, and the second row of each panel is the trace plot of the estimated latent signal.

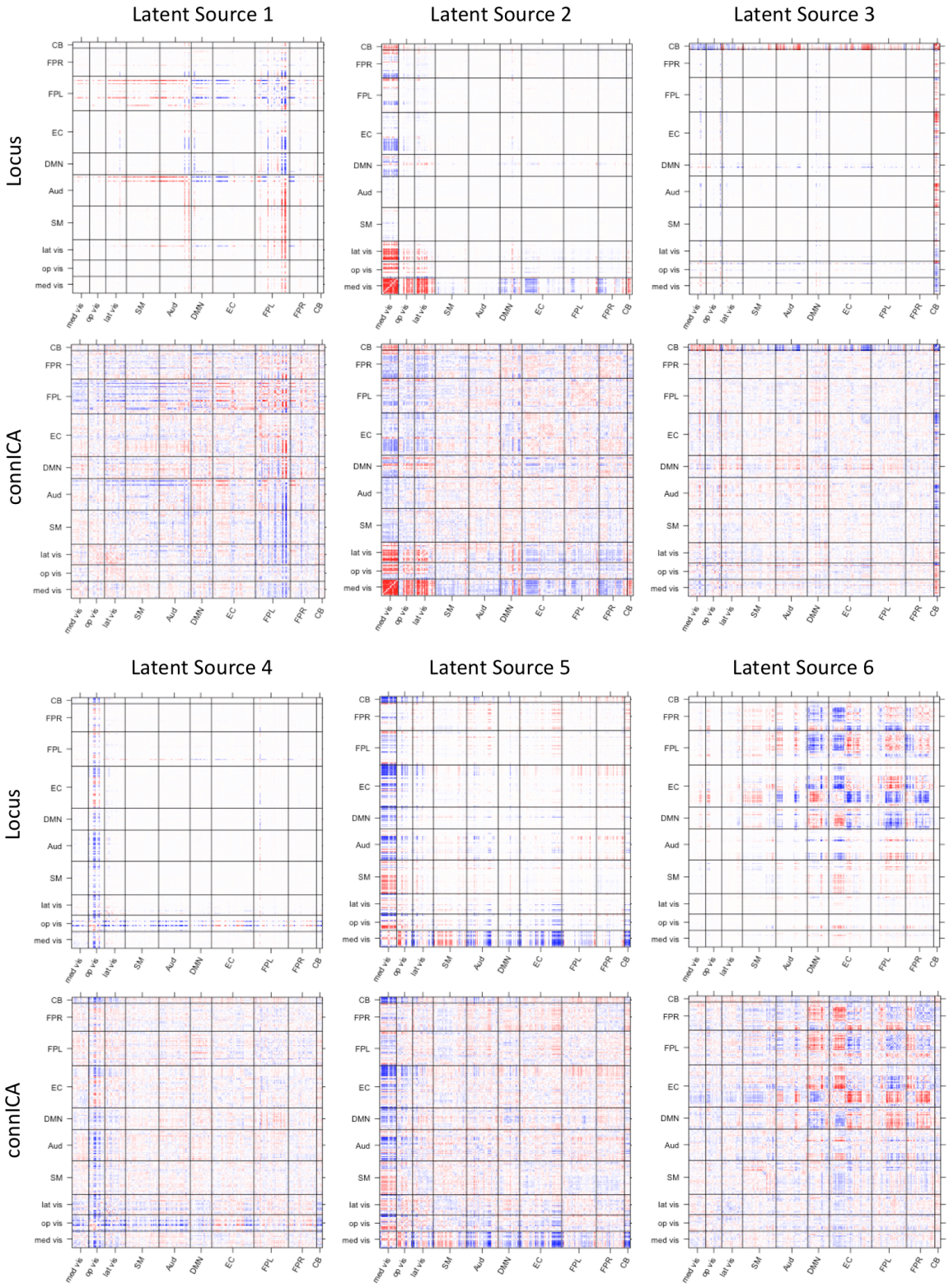

Figure 3.11: Heatmap of six matched latent sources between Locus and connICA with high reproducibility, where these six latent sources estimated from Locus have a Pearson-based reproducibility higher than 0.7.

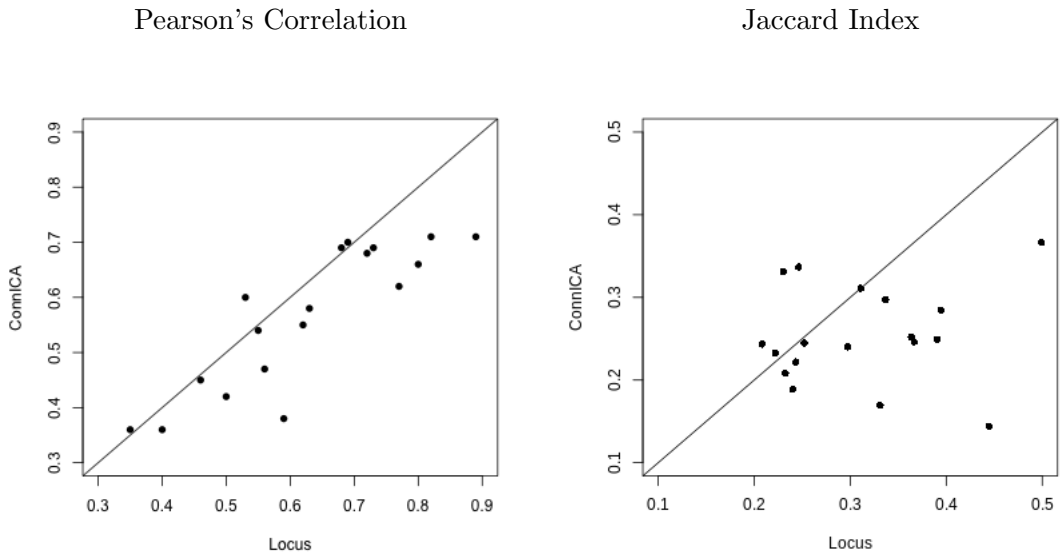

Figure 3.12: Reproducibility analysis for 18 matched latent sources from Locus and connICA. Left is based on Pearson's correlation and right is for Jaccard Index. It is shown that for the matched latent sources Locus tends to have higher reproducibility compareed to connICA approach.

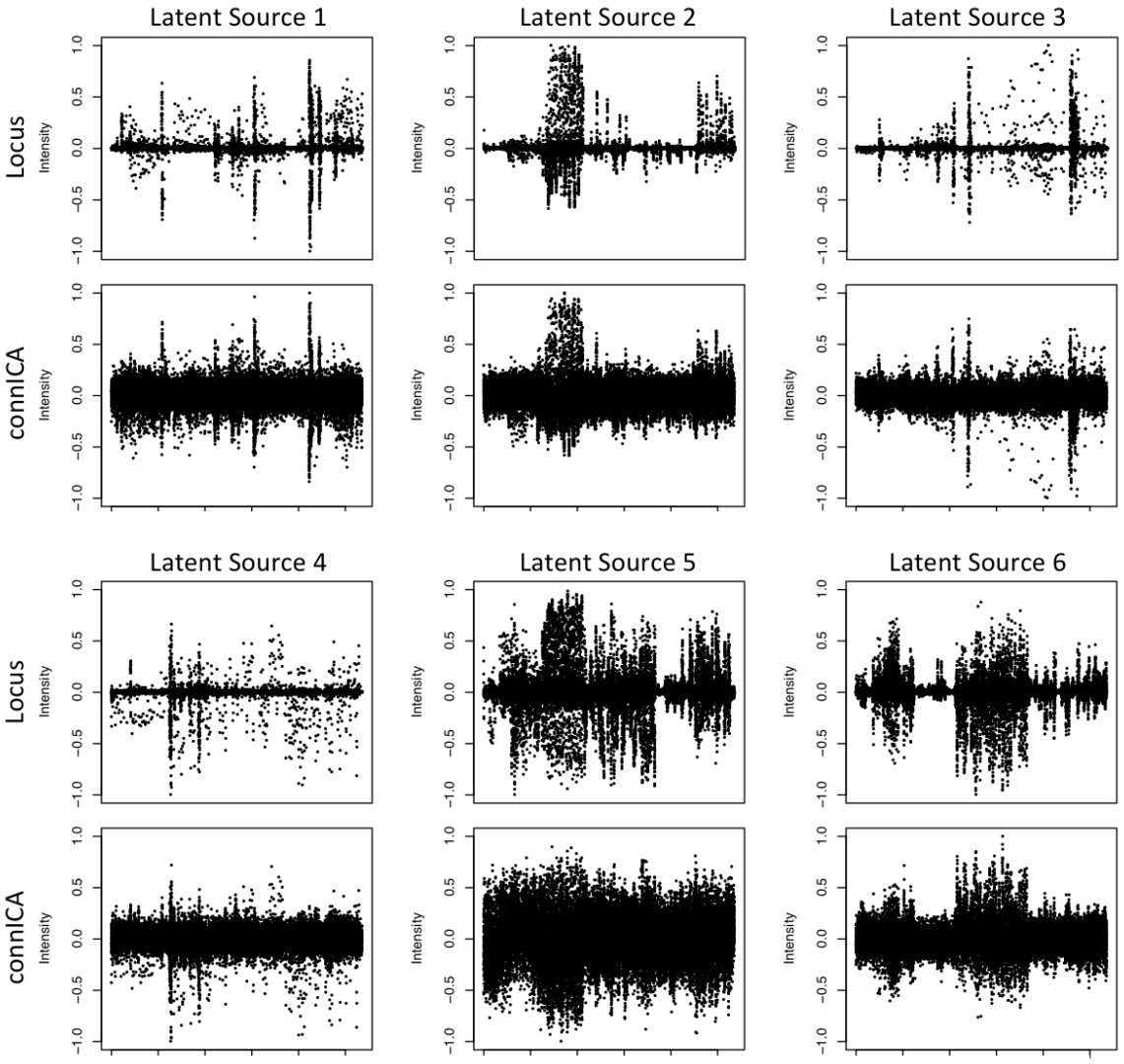

Figure 3.13: Intensity plot of six matched latent sources between Locus and connICA with high reproducibility, where these six latent sources estimated from Locus have a Pearson-based reproducibility higher than 0.7.

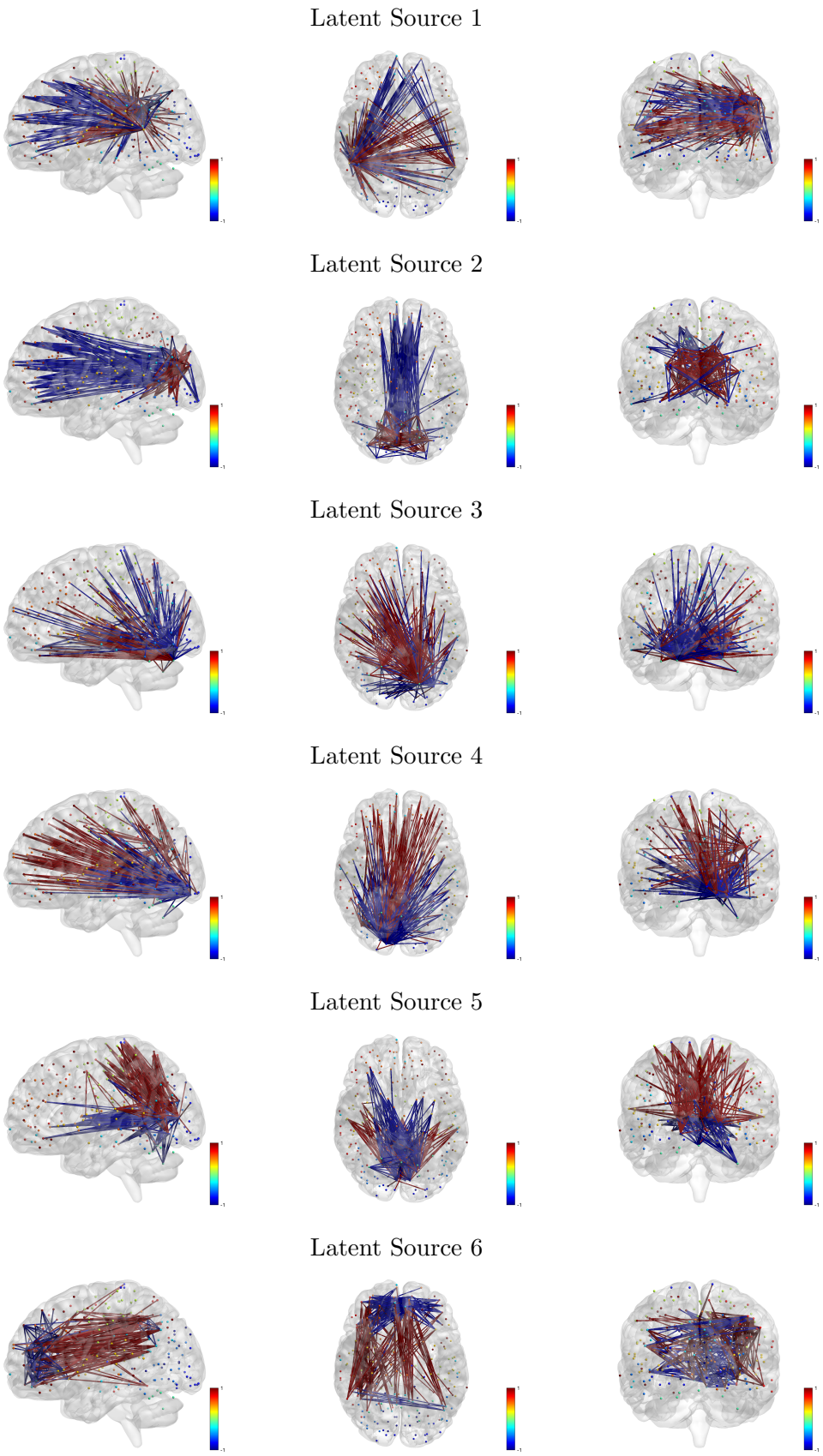

Figure 3.14: Visualizing the top 1% brain connectivities of the 6 matched latent signals based on Locus using BrainNetViewer.

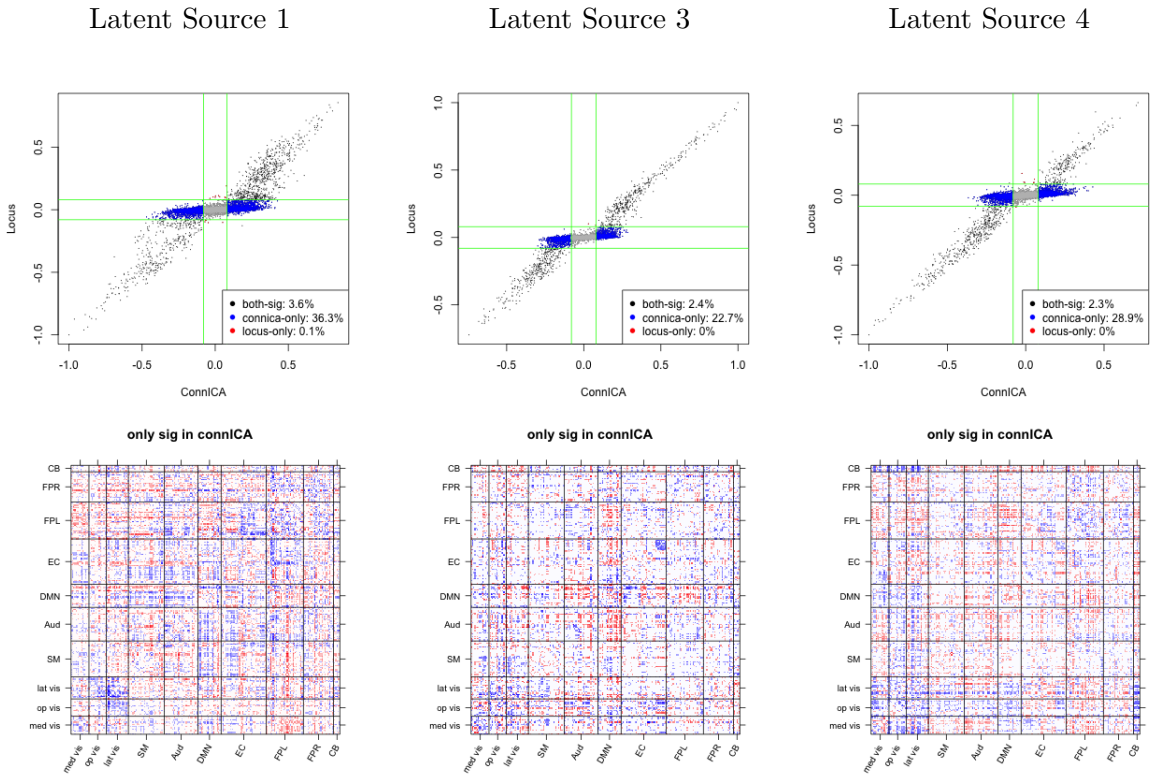

Figure 3.15: Comparison between Locus and connICA. We selected the three most correlated latent sources from the 2 methods, and show the difference between them. First row shows the scatter plot of the intensities from Locus and connICA with a threshold at 0.08, where blue dots represent the edges only significant in connICA but not in Locus. In the second row, those blue dots are visualized in the heatmap which are the edges only significant in connICA but not for Locus.

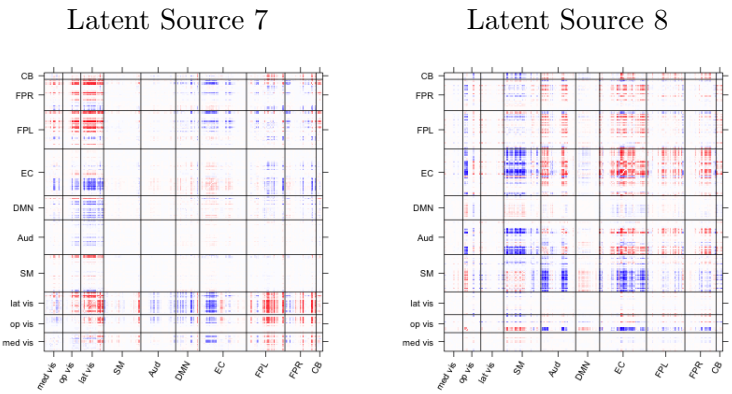

Figure 3.16: Two estimated latent sources based on Locus which are not identified by connICA. These 2 latent sources have relatively high reproducibility and are significantly associated with subjects' clinical outcomes, i.e. gender and age,

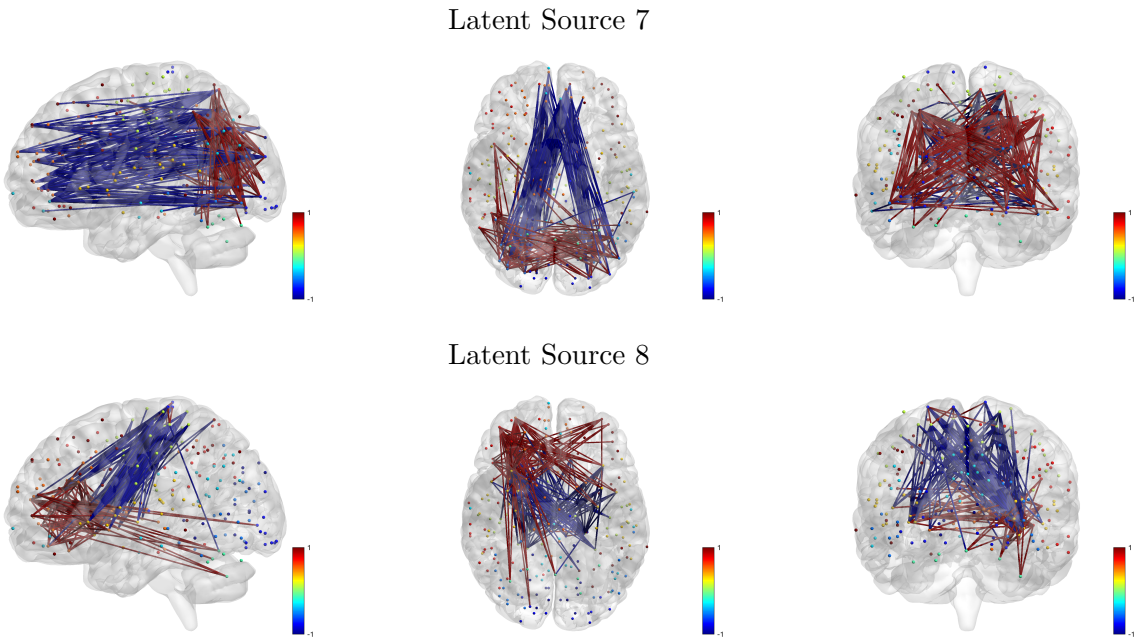

Figure 3.17: Visualizing the top 1% brain connectivities of the 2 estimated latent signals from Locus which are not identified by connICA

Chapter 4

A deep learning framework for brain network analysis with brain subnetwork structure

4.1 Introduction

In recent years, brain network oriented analysis has caused great attention in neuroscientific field. Serving as human's neuro-fingerprint, brain network based studies provide great insights into many fundamental questions in neuroscience research, such as neurodevelopment, neuroanatomy, neural basis of cognition, functional brain imaging, mental disorders and electrophysiology (Bullmore and Sporns, 2009; Deco et al., 2011; Satterthwaite, Wolf, Roalf, Ruparel, Erus, Vandekar, Gennatas, Elliott, Smith, Hakonarson et al., 2014; Wang and Guo, 2019; Finn et al., 2015). In neuroimaging studies, brain network measures are derived from either functional or structural brain imaging to reflect the functional or structural connections among spatially distinct locations across the brain (Bullmore and Sporns, 2009). For example, functional connectivity, usually derived from functional magnetic resonance imaging (fMRI), measures the temporal dynamics in neural processing of spatially disjoint brain areas (Wang et al., 2016), and structural (or anatomical) connectivity measures the existence and structural integrity of tracts connecting different brain areas (i.e. white matter tracts connecting cortical areas/nuclei) (Qiu et al., 2015). They both reflects brain network structure from a different perspectives. Recently, researchers found that brain network provides a new perspective to explore the underlying mechanism of many important clinical outcomes, such as mental disorder, human's intelligence, depression, dementia and other cognitive issues (Finn et al., 2015; Amico and Goñi, 2018b; Satterthwaite, Wolf, Roalf, Ruparel, Erus, Vandekar, Gennatas, Elliott, Smith, Hakonarson et al., 2014; Liu et al., 2017; Wang and Guo, 2019; Hu et al., 2016; Wada et al., 2019). Many existing evidence supports to use brain network as a modern biomarker to drive clinical decisions (Kiefer et al., 2015; Wu et al., 2015; Liu et al., 2017). For example, Liu et al. (2017) summarized and demonstrated the usage of brain network as a new guidance for early diagnosis and treatment of brain disorders.

Table 4.1: A summary of discussed deep learning works in brain network study.

Methods	Less Parameters	Interpreta-bility	Flexibility	Reference
BrainNetCNN	✔	✘	✘	*Hamarneha et al (2017)*
CNN	✔	✘	✔	*Wada et al (2019)*
GCN	✘	✘	✔	*Sarah et al (2018)*
Graph Kernel	✔	✘	✘	*Zhang et al (2015); Jie et al (2014); Dodero et al (2015)*
word2vec + CNN	✔	✘	✔	*Lu and Xiang (2018)*
Autoencoder	✘	✘	✔	*Munsell et al. (2015); Hu et al (2016)*
Graph measures	✔	✔	✘	*Rubinov and Sporns (2009)*

Kawahara et al. (2017) used structural connectivity to predict neurodevelopmental outcomes in preterm infants. Craddock et al. (2009) developed a machine learning model based on functional connectivity to predict depression state.

Although researchers have demonstrated the potential of using brain network as predictor in many applications, brain network based prediction is still facing several major challenges. First of all, brain network data is a complex system which is usually high dimensional (Chung, 2018) and encompasses a range of underlying subnetwork structures (i.e. default mode network) (Smith et al., 2009; Amico and Goñi, 2018b) shown in Figure 4.1. This makes it hard to extract or analyze the information within brain network data, leading to a high possibility of spurious findings in neuroscientific research (Button et al., 2013). As a result, traditional multivariate statistical methods or machine leaning models cannot fully explore the information in brain network data.

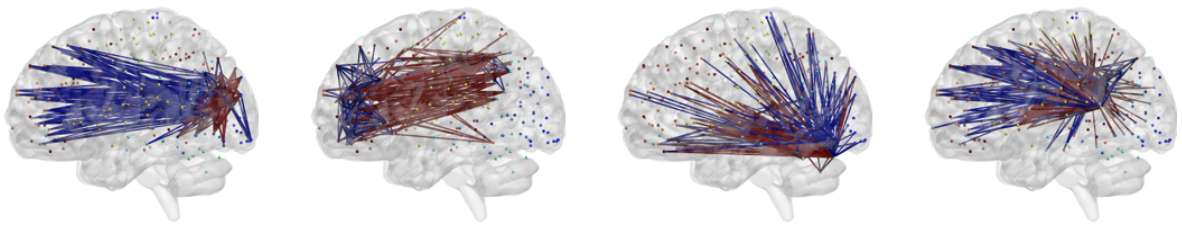

Figure 4.1: Visualizing some highly reproducible brain functional subnetworks derived from PNC study based on BrainNetViewer from Chapter 3.

Nowadays, deep learning (DL) methods are getting really popular in the field,

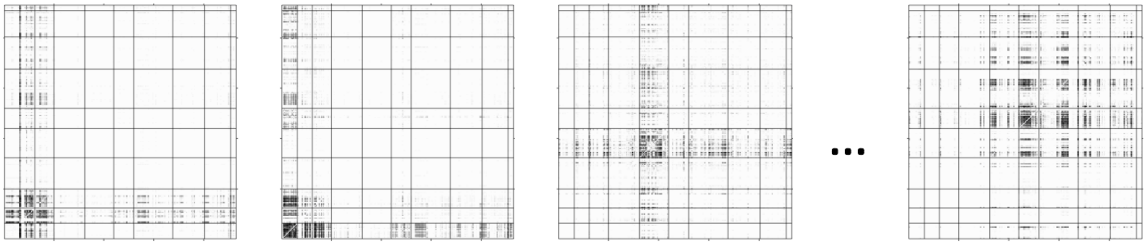

Figure 4.2: Heatmap of some highly reproducible binary brain functional subnetwork masks derived from PNC study from Chapter 3.

and DL based brain network analysis has also caused great attention in Neuroimaging society (Marblestone et al., 2016). Deep learning model denotes a class of machine learning models that uses multiple layers to progressively extract higher level features from the raw input, which is a generalized concept of neural network (Goodfellow et al., 2016). DL models are well-known for its high predictive power in various applications, which can handle complex data structures and learn non-linear patterns from the data by stacking various neural network layers appropriately. However, neuroimaging data are usually expensive and neuroimaging studies usually have limited sample size, which ranges from dozens to hundreds (Button et al., 2013). In contrast, deep learning methods usually have thousands to millions of parameters to train, which is a typical $n << p$ problem. Furthermore, DL based models are usually criticized for the low interpretability and often referred as black box methods (Heaven, 2019). This becomes a very serious limitation in neuroscientific research, where understanding the underlying brain mechanism is preferred over classification performance. Finally, DL models are usually complex, difficult to train and does not have appropriate model diagnostics, which makes it hard to be trusted in driving clinical decisions.

In practice, one commonly used strategy in DL-based brain network studies is to first extract low-dimensional features from brain network data and then feed these extracted features to a DL model. For example, Rubinov and Sporns (2010) summarized a list of metrics which can be derived from brain network and is widely used

in existing brain network oriented studies. Hu et al. (2016); Munsell et al. (2015) used autoencoder to extract features from brain network data and build subsequent classifier on it. Another example is from Meng and Xiang (2018) which use word2vec algorithm to encode the network information into node level embeddings and then utilized a convolutional neural network (CNN) on it for prediction. Although this strategy can somehow reduce the burden in terms of less parameters for subsequent neural network models, the framework is generally less flexible, and it also failed to take advantage of deep learning's most powerful ability in automatic feature processing & selection.

The other strategy is to develop non-standard DL models for network-valued data. One class of models is based on matrix-valued kernel functions (Jie et al., 2014; Zhang et al., 2015; Dodero et al., 2015). Specifically, brain network data are usually stored as symmetric connectivity matrix and some well-defined kernels can be used to quantify the similarity among different brain networks so that the existing classification methods such as kernel regression, support vector machine can be directly applied here. Although theoretically solid, this class of methods suffers from low predictive ability and the overall performance is highly sensitive to the selection of the kernel function. Furthermore, kernel based methods do not have a direct interpretation and therefore is less preferred. In recent years, graph convolutional neural network (GCN) is another class of DL models getting really popular in the neural network literature (Wu et al., 2019) where graph convolution stands for conducting a convolutional operation on a network. Although relevant, most existing GCN works are from a different perspective than brain network analysis. For example, existing GCN models are mainly focusing on node level tasks within a network, such as node level clustering or classification. But in brain network analysis, we are interested at using a whole brain network for prediction/classifcation across all population (Parisot et al., 2018).

Convolutional neural network (CNN) is a very powerful class of DL models specifically designed for processing imaging data (Goodfellow et al., 2016). By convoluting a rectangle-shaped filter across the image data, CNN can deal with high dimensional image data efficiently and save a large number of parameters. Although brain connectivity matrices are usually visualized as an image, i.e. heatmap, CNN is not directly applicable because the standard rectangle filter is not well-defined for network data - the element in a brain connectivity matrix stands for an edge between 2 nodes in the network while the elements covered by the standard CNN filter does not make any sense (Wada et al., 2019). To resolve such issue, Kawahara et al. (2017) proposed a novel CNN model, BrainNetCNN, for analyzing brain network data by defining a cross-shaped filter in convolutional layer. BrainNetCNN assumes that the neighborhood of an edge (u, v) are all edges either connected to node u or node v, which is covered by the cross-shaped filter. Although, such specification is appropriate defined and saves a large number of parameters , the BrainNetCNN model is still facing several challenges in terms of model interpretation and being less flexible for the complex brain subnetwork structures. Furthermore, from the existing neuroimaging literatures, some robust brain subnetwork structure are available as described in Chapter 3. Moreover, Incorporating such information as a prior into DL model could significantly improve model's reliability. As a summary, we listed all discussed works from 3 perspectives in Table 4.1: model's flexibility, interpretability and whether model has less parameters than a fully connected neural network model. From this table, none of these methods achieve these 3 properties at same time. Therefore, in this topic, we are planning to propose a novel deep learning framework to address these issues by incorporting existing brain subnetwork structure.

In this paper, we propose a novel deep learning framework for brain network analysis by incorporating existing brain subnetwork information. First of all, we extend the standard convolutional filter by defining an adaptively shaped graph convolu-

tional layer (Mconv) by customizing the filter's shape based on existing subnetwork's masks. Compared to traditional CNN model using rectangle filter or BrainNetCNN model using cross-shaped filter, our filter is highly flexible based on existing subnetwork information where the nearby neighborhood of an edge is defined based on the brain subnetwrok structure, which reflects the underlying brain connectivty patterns. Compared with a fully connected layer with same number of outputs, our model saves the number of parameters by q times where q denotes the number of subnetworks being considered. Moreover, the parameters in such subnetwork driven layer can be mapping back to each brain subnetwork, which makes it easier for visualization and interpretation. Following Mconv, we further define a Mask2score layer by constraining the information only propagating within each subnetwork and only allow interaction across subnetwork at the final layer. This specification is to further ensure the interpretability of the model parameters, and it naturally provides an efficient and robust model training strategy by viewing the model as an ensemble learning across q subnetworks. Based on the experiment study, we demonstrate the advantage of our model specification as well as the training strategy. More details will be presented in following chapters.

This paper is organized as follows: in chapter 4.2 we present the methodology of the proposed methods and discusses model train strategies; chapter 4.3 contains the real data application of the proposed method and chapter 4.4 is for discussion and conclusion.

4.2 Methodology

In this chapter, we will first present the proposed deep learning framework for brain network data analysis. After that, we will show a novel algorithm to estimate the unknown parameters in the model.

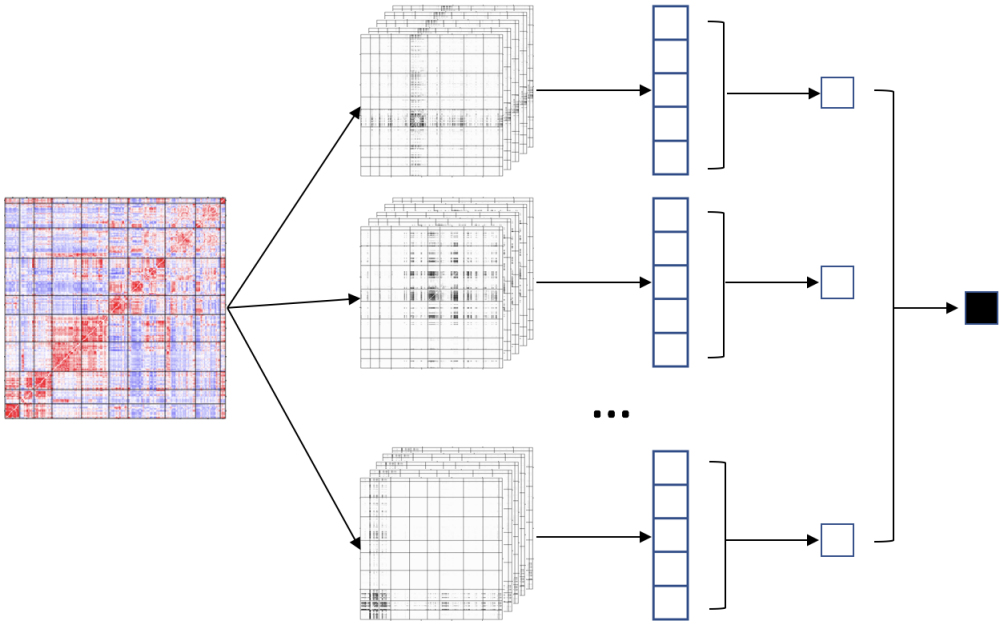

Figure 4.3: A visualization of DLconv modeling framework for brain network data analysis. This DLconv model contains a Mconv layer with 5 filters for each sub-network, a Mask2Score framework with 1 layer combining the output from Mconv into subnetwork-specific output, and a final layer combining the information from all subnetworks into the final output.

4.2.1 Notation

Assume $\boldsymbol{Y} \in \mathcal{R}_{V\times V}$ to be one observed brain connectivity matrix and $o \in \mathcal{R}$ to be the outcome. As shown in Figure 4.2, assume we have q binary masks representing the sub-network structure within the brain, $\boldsymbol{M} = \{\boldsymbol{M}_1, ..., \boldsymbol{M}_q\}$ where $\boldsymbol{M}_l$ is of shape $V \times V$ and $\boldsymbol{M}_l(u, v) \in \{0, 1\}$ representing the lth latent subnetwork mask of the brain network. We note that we do not explicitly specify the data structure on o which can either be categorical or continuous, which only affects the selection of loss function and activation function in the final layer of a neural network. In this paper, we aim at proposing a general DL framework for brain network analysis. Different types of o can be handled within this framework.

4.2.2 DLconv - Model Specification

Generally, a DLconv model consists of three components: the input component controlling with way to read in the data, information transformation component to process the information, and the final output component to link the information with the output. We will introduce these 3 components in the order here.

Mconv - Input Layer:

We first define a graph convolutional layer based on existing masks (Mconv) to take brain network data as input. In convolution operator for a network, the fundamental question is how to define the neighborhood for an edge. Based on the subnetwork structure stored in mask $\boldsymbol{M}$, we can flexibly specify the shape of filter in this graph convolution layer. Specifically, based on $\boldsymbol{M}$, the Mconv layer with H latent outputs contains h $V \times V$ convolutional filters for each mask, $\boldsymbol{W}_l = \{\boldsymbol{W}_{l1}, ..., \boldsymbol{W}_{lH}\}$, where $\boldsymbol{W}_{lj}(u, v)$ is constrained to be zero if $\boldsymbol{M}_l(u, v) = 0$. The operator of the Mconv layer is defined as

$$x_{lh} = \sigma\Big(\sum_{u,v}(\boldsymbol{Y} \circ \boldsymbol{W}_{lh})_{u,v}\Big) \tag{4.1}$$

for $l = 1, .., q,\ h = 1, ..., H$, where $\circ$ represents for Hadamard product. Given the property of $\boldsymbol{W}_{lh}$, the operator within σ is same as conducting a global convolution across the lth mask, i.e. $\sum_{(u,v) \text{ s.t. } M_l(u,v)=1} \boldsymbol{Y}(u, v)\boldsymbol{W}_{lh}(u, v)$.

Mconv layer has 3 major advantages. First, compared with a fully connected layer (FullNN) with same number of outputs, Mconv layer aves parameters from qHV^2 to HV^2. Second, the parameters $\boldsymbol{W}_l$'s can be directly mapping back to the brain network system. Compared with the cross-shaped filter in BrainNetCNN, this unique property makes the estimated filters in Mconv easier for visualization and interpretation. Thrid, Mconv layer incorporates the existing knowledge about the brain subnetwork structure which flexibly defines the neighborhood of an edge connection. Different

levels of brain subnetwork structure can be used simultaneously.

Mask2Score - Transformation Component:

In this section, we define how the information extracted from Mconv to be propagated in the model before reaching to the final layer. We already know each mask $\boldsymbol{M}_l$ corresponds to a subnetwork and the output of lth mask as $\boldsymbol{x}_l = [x_{l1}, ..., x_{lH}]^T$ summarizes all the information within such subnetwork. To further ensure the interpretability of our DL model, we specify an one-way information flow for each subnetwork as in our information transformation component.

Specifically, we propose a **Mask2Score** framework by forbidding interaction across subnetworks but only allowing information propagating within each subnetwork:

$$z_l = f_k \circ ... \circ f_1(\boldsymbol{x}_l; \boldsymbol{\beta}_{1l}, .., \boldsymbol{\beta}_{kl}), \tag{4.2}$$

where f_1,..,f_k denotes k layers in Mask2Score and $\boldsymbol{\beta}_l, .., \beta_k$ are the parameters belonging to these layers, and z_l is a single output summarizing all the information of lth subnetwork.

Last Layer:

In the last layer, we simply do a linear combination of the information across all subnetworks as our final output, i.e. $g(\sum_{l=1}^{q} \alpha_l z_l + \alpha_0)$. The reason of this over-simplified setting is we want to transform all the subnetwork level information into a scale based on Mconv and Mask2Score, so that in this last layer the parameters can be understood easily. For example, by comparing the weight α_c and α_b, we will have a direct view of the importance of these 2 subnetworks for the outcome. As a result, the layer layer is doing a simple generalized linear model on q subnetwork-specific features without allowing higher order interactions among them. If there is a strong evidence that higher order interaction between subnetworks, Last Layer can simply add these second or thrid order interaction terms like a traditional statistical model,

i.e. $z_c * z_b$.

As a summary, the overall framework for DLconv is visualized in Figure 4.3. Based on this Figure, we can clearly see that our DLconv is essentially an ensemble learning approach across q separate models for each subnetworks, and parameters at each layer can be directly interpretable for neuroscientific research.

4.2.3 Estimation

In this section, we will discuss the estimation method for the proposed DLconv model, and show it can be solved based on standard optimization toolbox. We will also introduce a updating procedure specifically designed for estimating the parameters in DLconv in a robust manner.

4.2.3.1 Objective Function for DLconv

To deal with the parameters in DLconv layer, we need to first reformat the network valued data into mask-specific vectors. First, we define a mask2vec operator as $\mathcal{M}(\boldsymbol{Y}, \boldsymbol{M}) = \text{vec}(\{\boldsymbol{Y}(u,v) \text{ if } \boldsymbol{M}(u,v) = 1\})$, which is to extract all elements in data within the mask. Next, apply $\mathcal{M}$ to data and filters, and we have the mask-specific inputs $\boldsymbol{y}_l = \mathcal{M}(\boldsymbol{Y}, \boldsymbol{M}_l)$and $\boldsymbol{w}_l = [\mathcal{M}(\boldsymbol{Y}, \boldsymbol{W}_{l1}), ..., \mathcal{M}(\boldsymbol{Y}, \boldsymbol{W}_{lH})]$, of dimension $p_l \times H$, where p_l is the number of edges in $\boldsymbol{M}_l$. As a result, we can simply show that x_{lh} from equation (4.1) is same as $\sigma(\boldsymbol{y}_l'\boldsymbol{w}_l)_h$. As a result, the final output from DLconv is

$$\hat{o} = g\Big(\alpha_0 + \sum_{l=1}^{q} \alpha_l f_k \circ .. \circ f_1\big(\sigma(\boldsymbol{y}_l'\boldsymbol{w}_l); \boldsymbol{\beta}_{1l}, .., \boldsymbol{\beta}_{kl}\big)\Big), \tag{4.3}$$

where α_l is the weight for output from lth sub-network representing its importance, each column of $\boldsymbol{w}_l$ can be mapping back to whole-brain network system. From equation (4.3), we can easily see that DLconv essentially trains q sub-models in an ensemble manner, which naturally provides an efficient way to do subnetwork-specific training. Given outcome o, selected loss function L and penalty function p from a

specific network analysis task, the optimization problem for DLconv model can be summarized as

$$\text{argmin}_{\Theta} L(o, \hat{o}; \Theta) + p(\Theta), \tag{4.4}$$

where $\Theta = \{\boldsymbol{w}_l, \boldsymbol{\beta}_{il}, \alpha_0, \alpha_l; l = 1, .., q; i = 1, .., k\}$. (4.4) can be optimized using standard optimization toolbox, such as Tensorflow, via back propagation (BP) algorithm. In next section, we will discuss the learning algorithm for (4.4) in details.

4.2.3.2 Learning Algorithm

Here we consider 2 strategies to learn the parameters in DLconv.

Strategy 1 - Joint: standard BP algorithm jointly applied to the whole model using existing DL optimization toolbox.

Although strategy 1 is straightforward, it still faces some challenges. First, although DLconv model has saved a large number of parameters than fully connected neural network, there is still more parameters than the sample size which could lead to unstable results. Furthermore, BP algorithm can easily over-emphasize into the high predictive subnetworks and the gradient for other subnetworks in Mconv layer can vanish quickly, pausing those parameters' update. As a result, the estimated parameters in Mconv would not be accuracy. Therefore, to overcome these limitations, we propose another strategy for DLconv model estimation.

Strategy 2 - SepIC: making use of the inherent structure of DLconv, we can train the parameters in each subnetwork's pathway in parallel, and combine them as our final classifier:

1. Estimate parameters $[\boldsymbol{\beta}_l, \boldsymbol{w}_l]$ in l th pathway by fixing other $\alpha_i = 0$, $i \neq l$:

$$\hat{\boldsymbol{\beta}}_l, \hat{\boldsymbol{w}}_l, \sim= \text{argmin } L(o, g\Big(\alpha_0 + \alpha_l f_k \circ ... \circ f_1(\sigma(\boldsymbol{y}_l' \boldsymbol{w}_l); \boldsymbol{\beta}_{1l}, .., \boldsymbol{\beta}_{kl})\Big)) + p(\boldsymbol{\beta}_{1l}, .., \boldsymbol{\beta}_{kl}, \boldsymbol{w}_l);$$

2. Conditioned on $\hat{\boldsymbol{\beta}}_l, \hat{\boldsymbol{w}}_l, l = 1, .., q$, update parameters α's in the Final Layer.

SepIC approach further breaks the original DLconv problem into smaller pieces which are easier to solve. This naturally relieves the burden to train DLconv with limited sample size. Furthermore, by training each subnetwork in parallel, DLconv will not be over-emphasized into a smaller proportion of subnewtorks where each pathway can be treated equally. Furthermore, along with the training process, SepIC approach can provide a sub-network specific training/testing performance, which can be further used to understand the univariate performance on each subnetwork without being affected by others.

4.3 Experiments

In this chapter, we will evaluate the performance of DLconv based on real brain network data from 3 perspectives. First, we will compare its predictive ability with some commonly used classification methods based on functional and structural brain connectivity data. Second, we will examine the stability of the proposed methods based on different initialization. Finally, we will conduct a DLconv-based brain network analysis to interpret the findings from DLconv and illustrate its power in real data analysis.

4.3.1 Study Design and Implementation

We used the brain structural (SC) and functional (FC) connectivity data from PNC study to evaluate DLconv's performance. Specifically, we have 235 SC matrices and 515 FC matrices, and we will use subject's gender as outcome and treat FC or SC as predictors for gender prediction. The SC and FC matrix are both symmetric and of dimension 232×232.

As for DLconv, we consider a setting with 5 filters for each mask in Mconv layer,

1 layer in mask2score and 1 final layer, which is a DL model with 3 hidden layers in total. We estimated the parameters based on both strategies - Joint and SepIC. We compare the predictive performance of DLconv with 1: a fully connected neural network model (Full NN) with same number of hidden layers and nodes, 2: a deep learning model with the same Mconv layer plus a 2-layer Full NN with same size as DLconv (Mconv + Full NN), 3: a BrainNetCNN model with the same size as our DLconv model (same number of parameters), denoted as BrainNetCNN$_q$, 4: the best BrainNetCNN setting in Kawahara et al. (2017), denoted as BrainNetCNN$_{sml}$, 5: support vector machine (SVM), 6: logistic regression with L1 penalty and 7: a random forest (RF) model. We also examined other settings for DLconv which does not check the predictive performance significantly.

As for implementation, we used sigmoid as the activation function, used ADAM with learning rate at 0.0001 as optimizer, dropout rate to be 0.2, plus a L2 penalty with $\lambda = 0.5$. The number of epochs is set to be 1500. The key evaluating metric is set to be area under the curve (AUC) for the receiver operating characteristic (ROC) curve. We also consider other 4 metrics including classification accuracy, F1 score, precision and recall. All models are implemented via Tensorflow 1.12.0 in Python 3.6.3.

4.3.2 Results

4.3.2.1 Predictive Performance

We first examine the predictive performance of our proposed DLconv model with other methods. We used both structural and functional connectivity data to predict subject's gender outcome. We used 5 fold cross validation framework and for each data split we run the same model 3 times with different different model initialization. The averaged and SD evaluating metrics across all runs are summarized in Table 4.2 for FC and in Table 4.3 for SC. The methods in Table 4.2 and Table 4.3 are ordered

Table 4.2: Functional connectivity based gender predictive performance for comparing DLconv with other methods with 5-fold cross validation. Values presented are mean and standard deviation of the evaluating metrics for testing dataset.

Methods	AUC (SD)	Accuracy (SD)	Fl score (SD)	Precision (SD)	Recall (SD)
DLconv (SepIC)	0.811 (0.083)	0.796 (0.057)	0.830 (0.036)	0.814 (0.086)	0.859 (0.067)
FullNN	0.790 (0.077)	0.752 (0.061)	0.803 (0.032)	0.752 (0.068)	0.870 (0.050)
DLconv (Joint)	0.787 (0.070)	0.751 (0.049)	0.791 (0.030)	0.780 (0.078)	0.817 (0.085)
BrainNetCNN_sml	0.773 (0.059)	0.741 (0.045)	0.782 (0.030)	0.770 (0.029)	0.815 (0.030)
SVM	0.754 (0.054)	0.770 (0.044)	0.809 (0.028)	0.777 (0.033)	0.845 (0.032)
BrainNetCNN_5q	0.746 (0.111)	0.714 (0.070)	0.770 (0.041)	0.761 (0.095)	0.806 (0.087)
Mconv+FullNN	0.741 (0.071)	0.710 (0.040)	0.765 (0.027)	0.720 (0.044)	0.824 (0.078)
Logistic	0.708 (0.053)	0.724 (0.048)	0.771 (0.037)	0.735 (0.028)	0.811 (0.053)
RF	0.605 (0.058)	0.645 (0.046)	0.739 (0.033)	0.639 (0.028)	0.878 (0.063)

Table 4.3: Structural connectivity based gender predictive performance for comparing DLconv with other methods with 5-fold cross validation. Values presented are mean and standard deviation of the evaluating metrics for testing dataset.

Methods	AUC (SD)	Accuracy (SD)	Fl score (SD)	Precision (SD)	Recall (SD)
DLconv (SepIC)	0.843 (0.063)	0.809 (0.045)	0.825 (0.042)	0.854 (0.050)	0.802 (0.065)
FullNN	0.840 (0.038)	0.806 (0.039)	0.819 (0.039)	0.863 (0.063)	0.801 (0.097)
DLconv (Joint)	0.832 (0.060)	0.800 (0.064)	0.823 (0.053)	0.841 (0.080)	0.813 (0.074)
Mconv+FullNN	0.828 (0.063)	0.800 (0.058)	0.827 (0.045)	0.839 (0.107)	0.838 (0.108)
BrainNetCNN_sml	0.813 (0.072)	0.790 (0.061)	0.801 (0.055)	0.832 (0.082)	0.800 (0.077)
BrainNetCNN_5q	0.699 (0.141)	0.751 (0.122)	0.776 (0.087)	0.802 (0.131)	0.767 (0.059)
Logistic	0.691 (0.043)	0.689 (0.039)	0.725 (0.025)	0.736 (0.071)	0.725 (0.060)
SVM	0.658 (0.020)	0.647 (0.022)	0.618 (0.058)	0.786 (0.061)	0.514 (0.069)
RF	0.626 (0.067)	0.647 (0.067)	0.735 (0.056)	0.650 (0.117)	0.870 (0.044)

based on the average AUC metric. Based on this study, we can see our method is always among the top 3 across all methods being considered. Specifically, SepIC training procedure does improve the predictive performance than the Joint training for our DLconv model in both SC and FC settings. Moreover, from this study, we do notice that deep learning based methods tend to have a higher predictive performance than traditional machine learning approaches. Although SVM has a relatively high accuracy and F1 score in FC based gender classification, as for SC its predictive performance is quite limited. Furthermore, the FullNN method with same number of hidden nodes has a pretty good predictive performance in both cases, which is very close to DLconv (SepIC). This finding is as expected because the number of parameters in FullNN is far more than DLconv approach. It is good to see that DLconv has a comparable or even higher performance than FullNN approach. As for Mconv + FullNN, its predictive performance for FC is pretty limited compared with DLconv or FullNN approaches. BrainNetCNN models are around the middle of all methods in both settings.

4.3.2.2 Stability Analysis on Model Initialization

It is a well-known problem that deep learning models are highly sensitive to the different initialization. In this session, we investigate the effect of initialization on DLconv based on FC data. Specifically, we use one data split where 80% data is used as training and other 20% is for testing. We re-train our model 50 times based on different weight initialization. We used FullNN and Mconv+FullNN as a comparison here. The average AUC and loss across epoch is visualized as a solid line in Figure 4.4. The shadow area is for 95% quantile of AUC or loss across 50 runs. It is clear to see that DLconv (Joint) provides a much more stable training than other 2 methods. The testing AUC metric gets stable after 200 epoches while FullNN and Mconv+Full have very unstable testing AUC. This finding demonstrates the robustness in our model

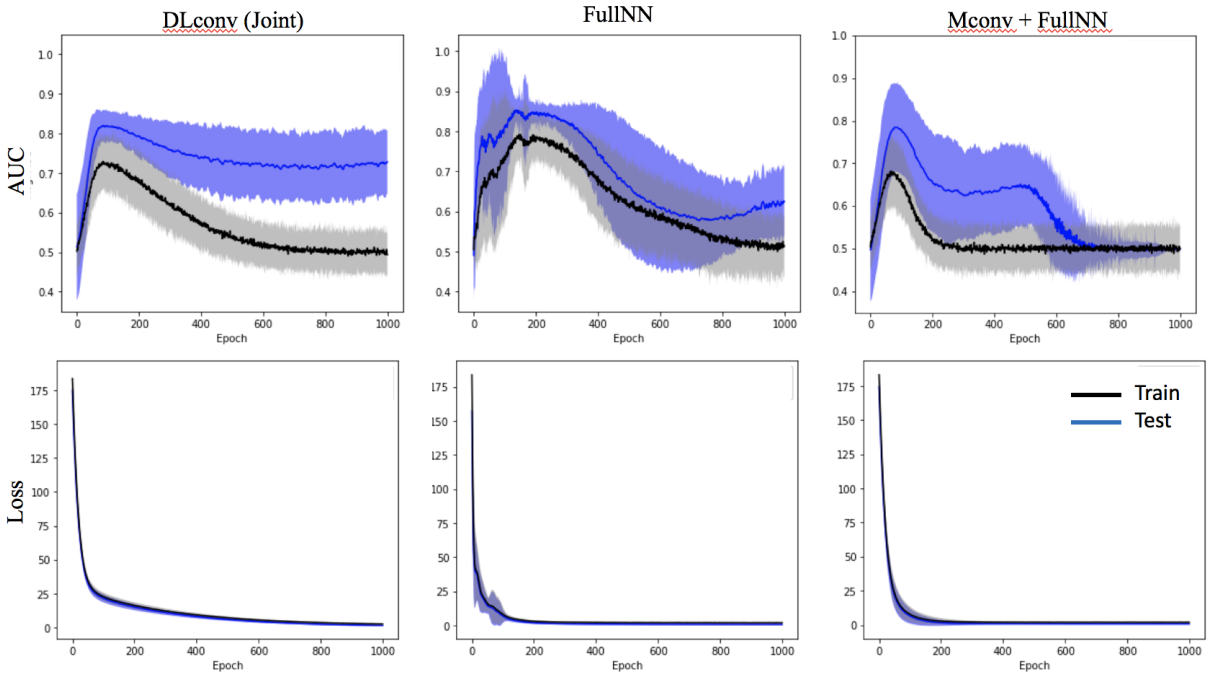

Figure 4.4: Model performance stability analysis across 50 initialization from DLconv, FullNN, Mconv + FullNN. Solid line represents the average and shadow area represents the 95% quantile.

and further indicates that the finding from our DLconv could be more reproducible than the FullNN approach.

4.3.2.3 Predictive Functional Brain Subnetworks for Gender Effect

In this section, we dig deep into the finding from DLconv model for functional brain network analysis on gender difference. To validate our finding and avoid uncertainty in DL trainning, we bootstrap the FC dataset for 50 times and re-run our DLconv models based on Joint and SepIC methods. First, we visualize the weight for the final layer, $\boldsymbol{w}_l$, across 50 runs in Figure 4.5 for both methods. We observe a clear difference on the weights between SepIC and Joint, where the weights from DLconv (Joint) tend to be more random than DLconv (SepIC). There is a clear order of subnetworks based on DLconv (SepIC), while for DLconv (Joint) there is no clear pattern. We also visualize the testing AUC for each subnetwork across 50 runs in Figure 4.6 which shares a similar pattern with the weight of DLconv (SepIC) in Figure 4.5. In Figure

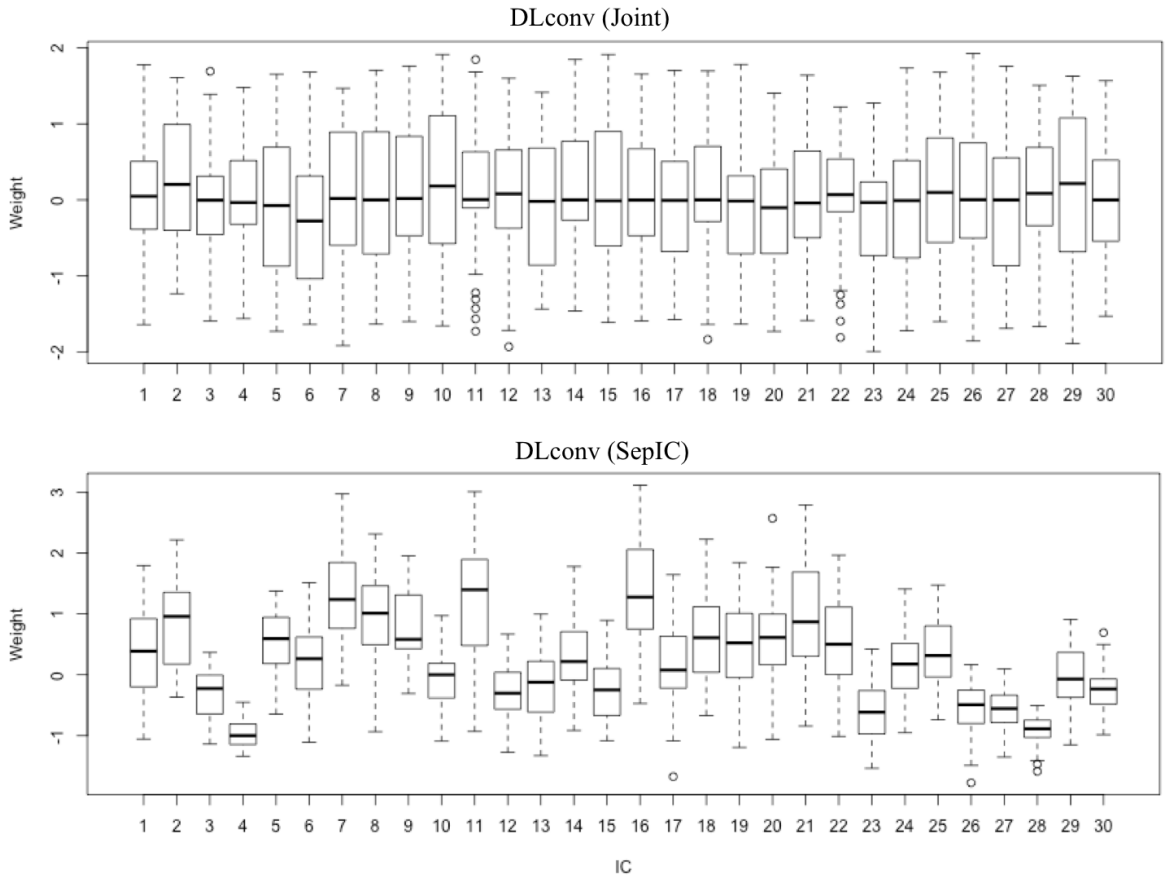

Figure 4.5: Boxplot of subnetwork-specific weights on the last Layer of DLconv across 50 bootstrap runs from two training strategies.

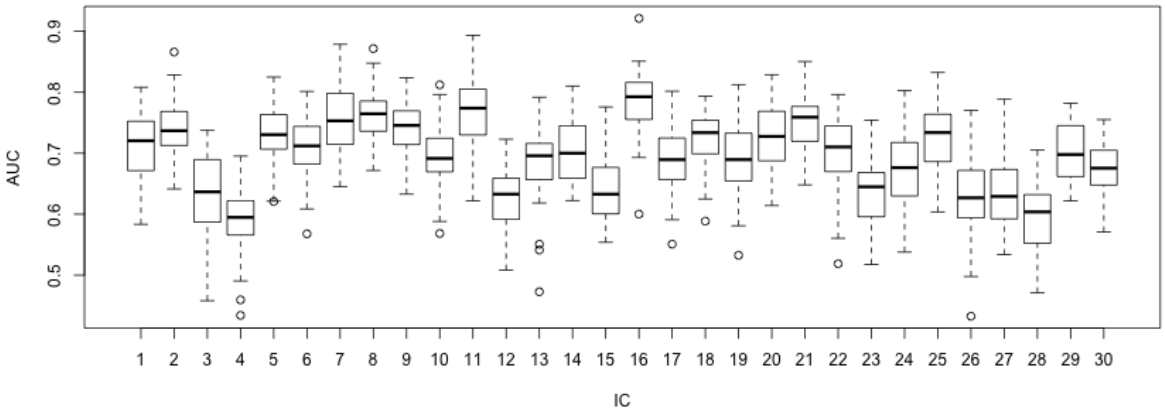

Figure 4.6: Boxplot of subnetwork-specific AUC for testing dataset across 50 bootstrap runs of DLconv model trained by SepIC strategy.

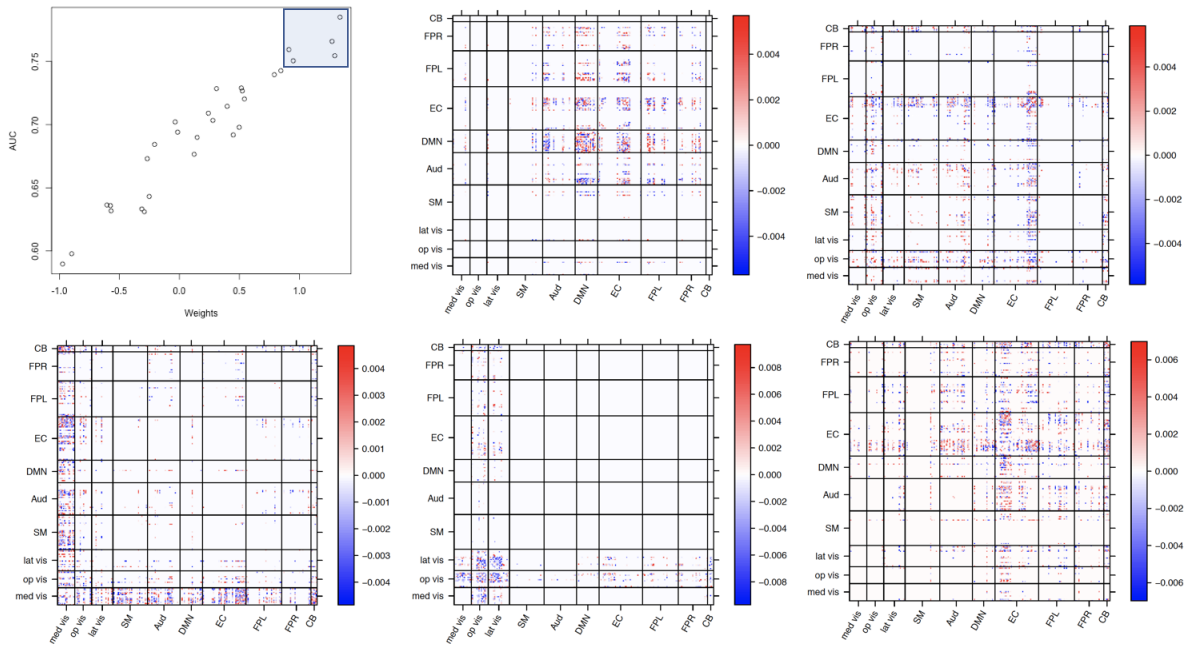

Figure 4.7: The 5 most predictive functional brain subnetworks for gender difference based on DLconv trained via SepIC algorithm. Subnetworks are selected based on average weight or AUC across 50 bootstrap runs and the visualized subnetwork-specific filters are the ones with largest weight in mask2score layer in the best performed model across 50 runs.

4.7, we show a scatter plot between weight and test AUC for each subnetwork based on DLconv (SepIC). The top 5 subnetworks are consistent in both the weight and test AUC (the five subnetwork having a univariate test AUC ¿ 0.75). We selected the best performed DLconv model across 50 runs and further selected the filter with the largest absolute weight in mask2score layer, i.e. $\boldsymbol{W}_{l,\text{argmax}\{\beta_l^{(1)},..,\beta_l^{(5)}\}}$. The selected filters are visualized as heatmap in Figure 4.7. These represents the 5 most differential patterns between male and female in the brain network.

4.4 Discussion and Conclusion

In this paper, we propose a novel deep learning framework - DLconv - for brain network analysis based on existing brain subnetwork structure. Specifically, we propose a subnetwork mask driven graph convolutional layer - Mconv - to extract the informa-

tion within the brain network data, where the neighborhood of any brain connections is specified by the subnetwork structure to appropriately deal with the unique structure in brain network. Furthermore, to ensure the interpretability within DLconv, we propose a mask2score framework to ensure the information only propagating within the same subnetwork and only combine the whole brain information at the final layer. Our DLconv not only saves a large number of parameters but also provide highly interpretable structure for neuroscientific research. Moreover, the subnetwork-separate structure naturally trains an ensemble model within the brain network. This finding motivates us to propose a more robust estimation procedure - SepIC - for DLconv model training by using an emsemble learning procedure.

We evaluate the performance of the proposed model based on extensive real data studies from 3 perspectives. First, based on the gender classification problem with structural or functional brain network data, DLconv achieves a better predictive performance than existing deep learning works for brain network. Second, we evaluate model's reliability across different model initialization and show that DLconv structure provides a more robust and stable estimation than full connected neural network. Finally, we illustrate the advantage of using DLconv plus SepIC in real network data analysis in terms of interpretability and flexiblity. Based on 50 bootstrap runs, we identified several highly predictive brain functional subnetwork patterns for gender difference. This finding is of great interest to understand the brain subnetwork difference between male and female, which provides great insights.

As for the future work, we plan to extend the experiments to other datasets to further confirm the performance of DLconv in different applications, such as multiclass prediction or continuous outcome regression. Currently, we only foucs on PNC study for gender prediction, which relatively limited. As for methodology development, there are two potential future directions along with this work. First direction is motivated by our real data study. We noticed that for some subnetworks filters share

a strong similarity with each other, while others are not. This finding might indicate that the same number of filters is not very efficient. We can potentially change the number of filter for each subnetwork to be it more flexible. One way to do this is to start with a large number of filters and conduct a PCA on each subnetwork to select the appropriate number of filters. The second direction is to extend the DLconv framework for time-varying brain network data. From a statistical perspective, DLconv treats each subnetwork data as independent samples but this is not correct in time-varying brain network analysis. We can build in some sequantial deep learning layers after Mconv layer to link the information across temporal domain. This extension is of great interest because time-varying brain network proposes much richer insights into the problem.

Appendix A

Appendix for Chapter 2

1. Q-function in E step:

The detailed expression for the complete data log-likelihood function at each voxel v is:

$$l_v(\Theta) = \sum_{i=1}^{N}\sum_{j=1}^{K}\left[\log g\left(\boldsymbol{y}_{ij}(v); \boldsymbol{A}_{ij}\boldsymbol{s}_{ij}(v), \boldsymbol{E}\right) + \log g\left(\boldsymbol{s}_{ij}(v); \boldsymbol{s}_0(v) + \boldsymbol{b}_i(v) + \boldsymbol{C}_j(v)\boldsymbol{x}_i^*, \tau^2\boldsymbol{I}\right)\right]$$
$$+ \sum_{i=1}^{N}\log g\left(\boldsymbol{b}_i(v); \boldsymbol{0}, \boldsymbol{D}\right) + \log g\left(\boldsymbol{s}_0(v); \boldsymbol{\mu}_{\boldsymbol{z}(v)}, \boldsymbol{\Sigma}_{\boldsymbol{z}(v)}\right) + \sum_{\ell=1}^{q}\log \pi_{l,\boldsymbol{z}_l(v)}$$

where $\boldsymbol{C}_j(v) = [\boldsymbol{\alpha}_j(v), \boldsymbol{\beta}_j(v)']$ of dimension $q \times (p+1)$, $\boldsymbol{x}_i^* = [1, \boldsymbol{x}_i']'$ and $g(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the pdf of multivariate normal distribution for random vector x with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.

We derive the Q function in E step as follows,

$$Q(\Theta|\hat{\Theta}^{(k)}) = E[l(\Theta; \mathcal{Y}, \mathcal{X}, \mathcal{S}, \mathcal{B}, \mathcal{Z})|\mathcal{Y}]$$
$$= Q_1(\Theta|\hat{\Theta}^{(k)}) + Q_2(\Theta|\hat{\Theta}^{(k)}) + Q_3(\Theta|\hat{\Theta}^{(k)}) + Q_4(\Theta|\hat{\Theta}^{(k)}) + Q_5(\Theta|\hat{\Theta}^{(k)}),$$

where

$$Q_1(\Theta|\hat{\Theta}^{(k)}) = -\frac{NKV}{2}\log|\boldsymbol{E}| - \frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{N}\sum_{j=1}^{K}\mathrm{tr}\Big\{\Big[\boldsymbol{y}_{ij}(v)\boldsymbol{y}_{ij}(v)' - 2\boldsymbol{A}_{ij}E[\boldsymbol{s}_{ij}(v)|\mathbf{y}(v);\hat{\Theta}^{(k)}]\boldsymbol{y}_{ij}(v)' + \boldsymbol{A}_{ij}E[\boldsymbol{s}_{ij}(v)\boldsymbol{s}_{ij}(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}]\boldsymbol{A}_{ij}'\Big]\boldsymbol{E}^{-1}\Big\},$$

$$Q_2(\Theta|\hat{\Theta}^{(k)}) = -\frac{NKVq}{2}\log|\tau^2| - \frac{1}{2\tau^2}\sum_{v=1}^{V}\sum_{i=1}^{N}\sum_{j=1}^{K}\mathrm{tr}\Big\{\Big[E[\boldsymbol{s}_{ij}(v)\boldsymbol{s}_{ij}(v)' + \boldsymbol{s}_0(v)\boldsymbol{s}_0(v)' + \boldsymbol{b}_i(v)\boldsymbol{b}_i(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}] + 2E[\boldsymbol{b}_i(v)\boldsymbol{s}_0(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}] + 2\boldsymbol{x}_i^{*,T}\boldsymbol{C}_j(v)'E[\boldsymbol{s}_0(v) + \boldsymbol{b}_i(v) - \boldsymbol{s}_{ij}(v)|\mathbf{y}(v);\hat{\Theta}^{(k)}] + \boldsymbol{C}_j(v)\boldsymbol{x}_i^*\boldsymbol{x}_i^{*,T}\boldsymbol{C}_j(v)' - 2E[\boldsymbol{s}_0(v)\boldsymbol{s}_{ij}(v)' + \boldsymbol{b}_i(v)\boldsymbol{s}_{ij}(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}]\Big]\Big\},$$

$$Q_3(\Theta|\hat{\Theta}^{(k)}) = -\frac{NV}{2}\log|\boldsymbol{D}| - \frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{N}\mathrm{tr}\Big\{\boldsymbol{D}^{-1}E[\boldsymbol{b}_i(v)\boldsymbol{b}_i(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}]\Big\},$$

$$Q_4(\Theta|\hat{\Theta}^{(k)}) = -\frac{1}{2}\sum_{v=1}^{V}\sum_{\ell=1}^{q}\sum_{j=1}^{m}p[z_\ell(v)=j|\mathbf{y}(v);\hat{\Theta}^{(k)}]\Big\{\log\sigma_{\ell,j}^2 + \frac{1}{\sigma_{\ell,j}^2}\Big[\mu_{\ell,j}^2 + E[s_0^{(\ell)}(v)^2|z_\ell(v)=j;\mathbf{y}(v),\hat{\Theta}^{(k)}] - 2\mu_{\ell,j}E[s_0^{(\ell)}(v)|z_\ell(v)=j,\mathbf{y}(v);\hat{\Theta}^{(k)}]\Big]\Big\},$$

$$Q_5(\Theta|\hat{\Theta}^{(k)}) = \sum_{v=1}^{V}\sum_{\ell=1}^{q}\sum_{j=1}^{m}p[z_\ell(v)=j|\mathbf{y}(v);\hat{\Theta}^{(k)}]\log\pi_{\ell,j},$$

2. Details about the E step of the exact EM algorithm.

In this section, we provide the details about the derivation in the exact E step. By collapsing our model across N subjects and K visits, for $v = 1, .., V$,

$$\begin{aligned}\boldsymbol{y}(v) &= \boldsymbol{A}\Big(\boldsymbol{U}^{(c)}\boldsymbol{\mu}_{\boldsymbol{z}(v)} + \boldsymbol{U}^{(c)}\boldsymbol{\psi}(v) + \boldsymbol{H}\boldsymbol{b}(v) + \boldsymbol{C}^*(v)\boldsymbol{X}^* + \boldsymbol{\gamma}(v)\Big) + \boldsymbol{e}(v), \qquad \text{(A.1)}\\ &= \boldsymbol{A}\boldsymbol{U}^{(c)}\boldsymbol{\mu}_{\boldsymbol{z}(v)} + \boldsymbol{A}\boldsymbol{C}^*(v)\boldsymbol{X}^* + \boldsymbol{A}\boldsymbol{R}\boldsymbol{r}_{\boldsymbol{z}(v)} + \boldsymbol{e}(v),\end{aligned}$$

where $\boldsymbol{A} = blockdiag(\boldsymbol{A}_{11}, .., \boldsymbol{A}_{NK})$, $\boldsymbol{b}(v) = [\boldsymbol{b}_1(v)', .., b_N(v)']'$, $\boldsymbol{\gamma}(v) = [\boldsymbol{\gamma}_{11}(v)', .., \boldsymbol{\gamma}_{NK}(v)']'$, $\boldsymbol{e}(v) = [\boldsymbol{e}_{11}(v)', .., \boldsymbol{e}_{NK}(v)']'$, $\boldsymbol{U}^{(c)} = \boldsymbol{1}_{NK} \otimes \boldsymbol{I}_q$, $\boldsymbol{H} = (\boldsymbol{I}_N \otimes \boldsymbol{1}_K) \otimes \boldsymbol{I}_q$, $\boldsymbol{C}^*(v) = \boldsymbol{I}_N \otimes [\boldsymbol{C}_1(v)', .., \boldsymbol{C}_K(v)']'$, $\boldsymbol{X}^* = [\boldsymbol{x}_1^{*,T}, ..., \boldsymbol{x}_N^{*,T}]'$ $\boldsymbol{R} = [\boldsymbol{H}, \boldsymbol{U}^{(c)}, \boldsymbol{I}_{qNK}]$, $\boldsymbol{r}_{\boldsymbol{z}(v)} = [\boldsymbol{b}(v)', \boldsymbol{\psi}'_{\boldsymbol{z}(v)}, \boldsymbol{\gamma}(v)']'$. Conditioned on latent variable $\boldsymbol{z}(v)$, A.1 can be represented as:

$$\begin{aligned}\boldsymbol{y}(v) - \boldsymbol{A}\boldsymbol{U}^{(c)}\boldsymbol{\mu}_{\boldsymbol{z}(v)} - \boldsymbol{A}\boldsymbol{C}^*(v)\boldsymbol{X}^*\Big|\boldsymbol{r}_{\boldsymbol{z}(v)}, \boldsymbol{z}(v) &\sim N(\boldsymbol{A}\boldsymbol{R}\boldsymbol{r}_{\boldsymbol{z}(v)}, \boldsymbol{\Upsilon}_v), \qquad \text{(A.2)}\\ \boldsymbol{r}_{\boldsymbol{z}(v)}|\boldsymbol{z}(v) &\sim N(\boldsymbol{0}, \boldsymbol{\Gamma}_{\boldsymbol{z}(v)})\end{aligned}$$

where $\boldsymbol{\Upsilon}_v = \boldsymbol{I}_{NK} \otimes \boldsymbol{E}_v$, $\boldsymbol{\Gamma}_{\boldsymbol{z}(v)} = blockdiag(\boldsymbol{I}_N \otimes \boldsymbol{D}, \boldsymbol{\Sigma}_{\boldsymbol{z}(v)}, \tau^2\boldsymbol{I}_{qNK})$. From (A.2), we can derive the conditional distribution of $[\boldsymbol{r}_{\boldsymbol{z}(v)}|\boldsymbol{y}(v), \boldsymbol{z}(v)]$ through Bayes' Theorem,

$$\begin{aligned}\boldsymbol{r}_{\boldsymbol{z}(v)}\Big|\boldsymbol{y}(v), \boldsymbol{z}(v) &\sim N(\boldsymbol{\mu}_{\boldsymbol{r}(v)|\boldsymbol{y}(v)}, \boldsymbol{\Sigma}_{\boldsymbol{r}(v)|\boldsymbol{y}(v)}),\\ \boldsymbol{\mu}_{\boldsymbol{r}(v)|\boldsymbol{y}(v)} &= \boldsymbol{\Sigma}_{\boldsymbol{r}(v)|\boldsymbol{y}(v)}\boldsymbol{R}'\boldsymbol{A}'\boldsymbol{\Upsilon}^{-1}\Big[\boldsymbol{y}(v) - \boldsymbol{A}\boldsymbol{U}^{(c)}\boldsymbol{\mu}_{\boldsymbol{z}(v)} - \boldsymbol{A}\boldsymbol{C}^*(v)\boldsymbol{X}^*\Big],\\ \boldsymbol{\Sigma}_{\boldsymbol{r}(v)|\boldsymbol{y}(v)} &= \left(\boldsymbol{\Gamma}^{-1}_{\boldsymbol{z}(v)} + \boldsymbol{R}'\boldsymbol{A}'\boldsymbol{\Upsilon}^{-1}\boldsymbol{A}\boldsymbol{R}\right)^{-1}.\end{aligned}$$

Next, we evaluate the conditional distribution of $\boldsymbol{L}(v)$. Given that $\boldsymbol{L}(v) = \boldsymbol{P}\boldsymbol{r}_{\boldsymbol{z}(v)} + \boldsymbol{Q}_{\boldsymbol{z}(v)}$, we have $\boldsymbol{L}(v)\Big|\boldsymbol{y}(v), \boldsymbol{z}(v) \sim N(\boldsymbol{P}\boldsymbol{\mu}_{\boldsymbol{r}(v)|\boldsymbol{y}(v)} + \boldsymbol{Q}_{\boldsymbol{z}(v)}, \boldsymbol{P}\boldsymbol{\Sigma}_{\boldsymbol{r}(v)|\boldsymbol{y}(v)}\boldsymbol{P}')$, where

$$\boldsymbol{P} = \begin{pmatrix}\boldsymbol{I}_{qN} & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{I}_q & \boldsymbol{0}\\ \boldsymbol{H} & \boldsymbol{U}^{(c)} & \boldsymbol{I}_{qNK}\end{pmatrix}, \boldsymbol{Q}_{\boldsymbol{z}(v)} = \begin{pmatrix}\boldsymbol{0}\\ \mu_{\boldsymbol{z}(v)}\\ \boldsymbol{U}^{(c)}\mu_{\boldsymbol{z}(v)} + \boldsymbol{C}^*(v)\boldsymbol{X}^*\end{pmatrix}.$$

Based on Bayes' Theorem, we have

$$p[\boldsymbol{z}(v) \mid \boldsymbol{y}(v)] \propto \Big(\prod_{l=1}^{q} \pi_{l,z_l(v)}\Big) g(\boldsymbol{A}\boldsymbol{U}^{(c)}\boldsymbol{\mu}_{\boldsymbol{z}(v)} + \boldsymbol{A}\boldsymbol{C}^*(v)\boldsymbol{X}^*, \boldsymbol{A}\boldsymbol{R}\boldsymbol{\Gamma}_{\boldsymbol{z}(v)}\boldsymbol{R}'\boldsymbol{A}' + \boldsymbol{\Upsilon}_v).$$

By integrating out $p[\boldsymbol{z}(v) \mid \boldsymbol{y}(v)]$, we obtain the conditional distribution of $\boldsymbol{L}(v)$.

3. Details about the M step of the exact EM

In this section, we provide the M step of the exact EM.

- Update the time-specific covariate effects $\boldsymbol{C}_j(v)$: for $j = 1, .., K$, $v = 1, .., V$,

$$\hat{\boldsymbol{C}}_j(v)^{(k+1)} = \left(\sum_{i=1}^{N} \boldsymbol{x}_i^* \boldsymbol{x}_i^{*,T}\right)^{-1} \sum_{i=1}^{N} \left\{ \boldsymbol{x}_i^* \left(E[\boldsymbol{s}_{ij}(v)' - \boldsymbol{s}_0(v)' - \boldsymbol{b}_0(v)' | \mathbf{y}(v); \hat{\Theta}^{(k)}] \right) \right\}.$$

- Update the mixing matrices $\boldsymbol{A}_{ij}$: for $i = 1, .., N$, $j = 1, .., K$,

$$\hat{\boldsymbol{A}}_{ij}^{(k+1)} = \left\{ \sum_{v=1}^{V} \boldsymbol{y}_{ij}(v) E[\boldsymbol{s}_{ij}(v)' | \mathbf{y}(v); \hat{\Theta}^{(k)}] \right\} \left\{ \sum_{v=1}^{V} E[\boldsymbol{s}_{ij}(v)\boldsymbol{s}_{ij}(v)' | \mathbf{y}(v); \hat{\Theta}^{(k)}] \right\}^{-1},$$

  and then update $\widehat{\boldsymbol{A}}_{ij}^{(k+1)} = \mathcal{H}(\breve{\boldsymbol{A}}_{ij}^{(k+1)})$ where $\mathcal{H}(\cdot)$ is the orthogonalization transformation.

- Update the first level variance term $\boldsymbol{E}_v = \sigma_0^2 \boldsymbol{I}_q$ with:

$$\begin{aligned}\hat{\sigma}_0^{2(k+1)} =& \frac{1}{NKVq} \sum_{v=1}^{V} \sum_{i=1}^{N} \sum_{j=1}^{K} \Big\{ \boldsymbol{y}_{ij}(v)'\boldsymbol{y}_{ij}(v) - 2\boldsymbol{y}_{ij}(v)' \widehat{\boldsymbol{A}}_{ij}^{(k+1)} E[\boldsymbol{s}_{ij}(v) | \mathbf{y}(v); \hat{\Theta}^{(k)}] \\ &+ \operatorname{tr}\left[\widehat{\boldsymbol{A}}_{ij}^{(k+1)} E[\boldsymbol{s}_{ij}(v)\boldsymbol{s}_{ij}(v)' | \mathbf{y}(v); \hat{\Theta}^{(k)}] \widehat{\boldsymbol{A}}_{ij}^{(k+1)\prime} \right] \Big\}.\end{aligned}$$

- Update subject-specific variance term $\boldsymbol{D}$:

$$\hat{\boldsymbol{D}}^{(k+1)} = \frac{1}{NV} \sum_{v=1}^{V} \sum_{i=1}^{N} E[\boldsymbol{b}_i(v)\boldsymbol{b}_i(v)' | \mathbf{y}(v); \hat{\Theta}^{(k)}],$$

- Update second level variance term $\tau^2 \boldsymbol{I}_q$:

$$\begin{aligned}\hat{\tau}^{2(k+1)} =& \frac{1}{NKVq}\sum_{v=1}^{V}\sum_{i=1}^{N}\sum_{j=1}^{K}\mathrm{tr}\Big\{E[\boldsymbol{s}_{ij}(v)\boldsymbol{s}_{ij}(v)' + \boldsymbol{s}_0(v)\boldsymbol{s}_0(v)' + \boldsymbol{b}_i(v)\boldsymbol{b}_i(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}] \\ &+ 2E[\boldsymbol{b}_i(v)\boldsymbol{s}_0(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}] + 2\boldsymbol{x}_i^{*,T}\boldsymbol{C}_j(v)'E[\boldsymbol{s}_0(v) + \boldsymbol{b}_i(v) - \boldsymbol{s}_{ij}(v)|\mathbf{y}(v);\hat{\Theta}^{(k)}] \\ &+ \boldsymbol{C}_j(v)\boldsymbol{x}_i^*\boldsymbol{x}_i^{*,T}\boldsymbol{C}_j(v)' - 2E[\boldsymbol{s}_0(v)\boldsymbol{s}_{ij}(v)' + \boldsymbol{b}_i(v)\boldsymbol{s}_{ij}(v)'|\mathbf{y}(v);\hat{\Theta}^{(k)}]\Big\},\end{aligned}$$

- Update $\pi_{\ell,j}$:

$$\hat{\pi}_{\ell,j}^{(k+1)} = \frac{1}{V}\sum_{v=1}^{V} p[\boldsymbol{z}_\ell(v) = j|\mathbf{y}(v);\hat{\Theta}^{(k)}].$$

- Update $\mu_{\ell,j}$:

$$\hat{\mu}_{\ell,j}^{(k+1)} = \frac{\sum_{v=1}^{V} p[\boldsymbol{z}_\ell(v) = j|\mathbf{y}(v);\hat{\Theta}^{(k)}]E[s_{0\ell}(v)|\boldsymbol{z}_\ell(v) = j,\mathbf{y}(v);\hat{\Theta}^{(k)}]}{V\hat{\pi}_{\ell,j}^{(k+1)}}.$$

- Update $\sigma_{\ell,j}^2$:

$$\hat{\sigma}_{\ell,j}^{2(k+1)} = \frac{\sum_{v=1}^{V} p[\boldsymbol{z}_\ell(v) = j|\mathbf{y}(v);\hat{\Theta}^{(k)}]E[s_{0\ell}(v)^2|\boldsymbol{z}_\ell(v) = j,\mathbf{y}(v);\hat{\Theta}^{(k)}]}{V\hat{\pi}_{\ell,j}^{(k+1)}} - [\hat{\mu}_{\ell,j}^{(k+1)}]^2.$$

Here, $E[s_{0\ell}(v) \mid z_\ell(v) = j, \mathbf{y}(v);\Theta]$, $E[s_{0\ell}(v)^2 \mid z_\ell(v) = j, \mathbf{y}(v);\Theta]$ and $p[z_\ell(v) = j \mid \mathbf{y}(v);\Theta]$ are the marginal conditional moments and probability related to the ℓth IC. They are derived by summing across all the possible states of the other $q-1$ ICs as follows,

$$E[s_{0\ell}(v) \mid z_\ell(v) = j, \mathbf{y}(v);\Theta] = \frac{\sum_{\boldsymbol{z}(v)\in\mathcal{R}^{(\ell,j)}} p[\boldsymbol{z}(v) \mid \mathbf{y}(v);\Theta]E[s_{0\ell}(v) \mid \mathbf{y}(v), \boldsymbol{z}(v);\Theta]}{p[z_\ell(v) = j \mid \mathbf{y}(v);\Theta]}, \tag{A.3}$$

$$p[z_\ell(v) = j \mid \mathbf{y}(v);\Theta] = \sum_{\boldsymbol{z}(v)\in\mathcal{R}^{(\ell,j)}} p[\boldsymbol{z}(v) \mid \mathbf{y}(v);\Theta].$$

where $\mathcal{R}^{(\ell,j)}$ is defined as $\{\boldsymbol{z}^r \in \mathcal{R} : z_\ell^r = j\}$ for all $\ell = 1,..,q, j = 1,...,m$.

4. Statistical inference for testing covariate effects in L-ICA:

In this section, we present the statistical inference procedure for testing covariate effects in L-ICA. We first stack the fMRI data from all visits of a subject to have the subject-specific fMRI data $\boldsymbol{y}_i(v)$ of dimension $qK \times 1$ which is $[\boldsymbol{y}_{i1}(v)', ..., \boldsymbol{y}_{iK}(v)']'$, and a non-hierarchical form of L-ICA is derived by combining equations (2.1),(2.2) and (2.3),

$$\boldsymbol{A}_i'\boldsymbol{y}_i(v) = \boldsymbol{U}\boldsymbol{\mu}_{\boldsymbol{z}(v)} + \boldsymbol{\alpha}(v) + \boldsymbol{X}_i\boldsymbol{\beta}(v) + \boldsymbol{U}\boldsymbol{\psi}_{\boldsymbol{z}(v)} + \boldsymbol{U}\boldsymbol{b}_i(v) + \boldsymbol{\gamma}_i(v) + \boldsymbol{A}_i'\boldsymbol{e}_i(v), \quad \text{(A.4)}$$

where $\boldsymbol{A}_i = blkdiag(\boldsymbol{A}_{i1}, ..., \boldsymbol{A}_{iK})$, $\boldsymbol{\gamma}_i(v) = [\boldsymbol{\gamma}_{i1}(v)', ..., \boldsymbol{\gamma}_{iK}(v)']'$, $\boldsymbol{e}_i(v) = [\boldsymbol{e}_{i1}(v)', ..., \boldsymbol{e}_{iK}(v)']'$, $\boldsymbol{\alpha}(v) = [\boldsymbol{\alpha}_1(v)', \boldsymbol{\alpha}_2(v)', .., \boldsymbol{\alpha}_K(v)']'$, $\boldsymbol{\beta}(v) = [\text{vec}\,[\boldsymbol{\beta}_1(v)']', ..., \text{vec}\,[\boldsymbol{\beta}_K(v)']']'$, $\boldsymbol{U} = \boldsymbol{1}_K \otimes \boldsymbol{I}_q$ and $\boldsymbol{X}_i = \boldsymbol{I}_K \otimes (\boldsymbol{x}_i' \otimes \boldsymbol{I}_q)$. The model in (A.4) is further re-written as

$$\begin{aligned} \boldsymbol{y}_i^*(v) &= \boldsymbol{X}_0\boldsymbol{\alpha}^*(v) + \boldsymbol{X}_i\boldsymbol{\beta}(v) + \boldsymbol{\zeta}_i(v), \quad \text{(A.5)} \\ &= \boldsymbol{X}_i^*\boldsymbol{C}^*(v) + \boldsymbol{\zeta}_i(v), \end{aligned}$$

where $\boldsymbol{y}_i^*(v) = \boldsymbol{A}_i'\boldsymbol{y}_i(v)$, $\boldsymbol{X}_i^* = [\boldsymbol{X}_0, \boldsymbol{X}_i]$, $\boldsymbol{X}_0 = \begin{pmatrix} 1 & \boldsymbol{0}_{K-1}' \\ \boldsymbol{1}_{K-1} & \boldsymbol{I}_{K-1} \end{pmatrix} \otimes \boldsymbol{I}_q$, $\boldsymbol{\alpha}^*(v) = [\boldsymbol{\mu}_{\boldsymbol{z}(v)}', \boldsymbol{\alpha}_2(v)', .., \boldsymbol{\alpha}_K(v)']'$, $\boldsymbol{C}^*(v) = [\boldsymbol{\alpha}^*(v)', \boldsymbol{\beta}(v)']'$ and $\boldsymbol{\zeta}_i(v) = \boldsymbol{U}\boldsymbol{\psi}_{\boldsymbol{z}(v)} + \boldsymbol{U}\boldsymbol{b}_i(v) + \boldsymbol{\gamma}_i(v) + \boldsymbol{A}_i'\boldsymbol{e}_i(v) \sim N(\boldsymbol{0}, \boldsymbol{W}_i(v))$ is the multivariate zero-mean Gaussian noise term where $\boldsymbol{W}_i(v) = \boldsymbol{U}(\boldsymbol{\Sigma}_{\boldsymbol{z}(v)} + \boldsymbol{D})\boldsymbol{U}' + \boldsymbol{A}_i\boldsymbol{E}_v\boldsymbol{A}_i' + \tau^2\boldsymbol{I}_{qK}$, which can be shown as $\boldsymbol{W}_i(v) = \boldsymbol{W}(v) = \boldsymbol{U}(\boldsymbol{\Sigma}_{\boldsymbol{z}(v)} + \boldsymbol{D})\boldsymbol{U}' + (\sigma_0^2 + \tau^2)\boldsymbol{I}_{qNK}$.

5. Details about the pre-whitening prior to ICA:

Following previous work (Beckmann and Smith, 2004), we perform preliminary analysis to prewhiten the data so that the noise covariance can be assumed to be isotropic

across voxels in the probabilistic ICA model. Specifically, if the original covariance of the noise $\boldsymbol{e}_{ij}(v)$ is known as $\sigma_0^2\boldsymbol{E}_v$, we can use the Cholesky decomposition $\boldsymbol{E}_v = \boldsymbol{K}_v\boldsymbol{K}_v'$ to rewrite model (2.1) as

$$\boldsymbol{K}_v^{-1}\boldsymbol{y}_{ij}(v) = \boldsymbol{K}_v^{-1}\boldsymbol{A}_{ij}\boldsymbol{s}_{ij}(v) + \boldsymbol{K}_v^{-1}\boldsymbol{e}_{ij}(v), \tag{A.6}$$

and obtain a new representation,

$$\bar{\boldsymbol{y}}_{ij}(v) = \bar{\boldsymbol{A}}_{ij}\boldsymbol{s}_{ij}(v) + \bar{\boldsymbol{e}}_{ij}(v), \tag{A.7}$$

where $\bar{\boldsymbol{e}}_{ij}(v) \sim N(\boldsymbol{0}, \sigma_0^2\boldsymbol{I})$. Therefore, the noise covariance becomes isotropic and standardized across voxels.

When $\boldsymbol{E}_v$ is unknown, the prewhitening can be achieved by the following iterative procedure: (1) start with an initial noise covariance $\boldsymbol{E}_v^{(0)}$, prewhiten the data as in (A.6), (2) with voxel-wise prewhitened data, we can readily derive the ML estimates of $\hat{\bar{\boldsymbol{A}}}_{ij,ML}$, $\hat{\boldsymbol{s}}_{ij,ML}(v)$ and $\hat{\sigma}^2_{0,ML}$ (Beckmann and Smith, 2004), (3) re-estimate the noise covariance $\boldsymbol{E}_v$ based on the residuals $\hat{\bar{\boldsymbol{e}}}_{ij}(v)$ from model (A.7), and then repeat steps (1)-(3). By performing the iterative procedure, we obtain the preprocessed data for the subsequent ICA modeling.

6. Robustness of between-group test results based on L-ICA

We conduct additional analyses to evaluate the robustness of the between-group comparison results in the DMN for the ADNI2 study based on the proposed L-ICA. We obtain 51 data sets by applying the leave-one-out procedure on the ADNI2 data, where each data set contains 50 subjects by removing one subject from the original data. We then run L-ICA and conduct between-group comparisons for each of the data sets. We evaluate the consistency of the comparison results for each voxels in the DMN by examining whether the significance of the test result is consistent or

not with the original data. Specifically, for voxel v, the consistency rate is defined as $\frac{1}{51}\sum_{k=1}^{51}\mathbf{1}(sig_k(v) = sig_{org}(v))$, where $sig_{org}(v)$ is a binary indicator that equals 1 if the voxel v showed significant between-group test result in the original data and equals 0 if otherwise, and $sig_k(v)$ is the corresponding binary test significance indicator under the kth leave-one-out dataset. Table A.1 presents the results on the consistency rate across voxels for each of the group comparisons, including AD vs. CN at every visit, EMCI vs. LMCI at every visit. We also examine the consistency for longitudinal changes from baseline to year 2 for the AD group. Specifically, we first present the average consistent rate across all voxels in the network (1st row in Table A.1) Then, we present the average consistency rates separately for voxels that are significant in the original tests (2nd row in A.1) and voxels that are non-significant in the original tests (3rd row in Table A.1). Results show that the group test results based on the L-ICA on average have a consistent rate of over 90% across the DMN and also within both significant voxels and non-significant voxels, indicating the between-group comparison results based on L-ICA are fairly robust for the ADNI2 study.

Table A.1: Consistency of the group comparisons results based on L-ICA for the ADNI2 study.

Averaged Consistency	AD vs CN			LMCI vs EMCI			Year2 vs
Rate	Baseline	Year 1	Year 2	Baseline	Year 1	Year 2	Baseline
All voxels in DMN	0.948	0.949	0.945	0.936	0.943	0.926	0.966
Voxels w. diff.	0.960	0.966	0.968	0.959	0.961	0.955	0.922
Voxels w.o. diff.	0.945	0.946	0.937	0.927	0.938	0.912	0.973

Appendix B

Appendix for Chapter 3

1. Detailed derivation in Node-Moving algorithm:

To see the linkage between (9) and (10), we first transfer (9) into edge-wise form:

$$\begin{aligned}&\left\|\widetilde{\boldsymbol{A}}_l'\widetilde{\boldsymbol{Y}}-\mathcal{L}(\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_l')'\right\|_2^2+\phi\sum_{u<v}|\boldsymbol{X}_l(u)'\boldsymbol{D}_l\boldsymbol{X}_l(v)|\\=&\sum_{u<v}\left(\widetilde{\boldsymbol{A}}_l'\widetilde{\boldsymbol{Y}}(u,v)-\boldsymbol{X}_l(u)'\boldsymbol{D}_l\boldsymbol{X}_l(v)\right)^2+\phi\sum_{u<v}|\boldsymbol{X}_l(u)'\boldsymbol{D}_l\boldsymbol{X}_l(v)|,\end{aligned}$$

which contains $V(V-1)/2$ terms. To update $\boldsymbol{X}_l(v)$ while conditioning on others, we only take the terms involved with $\boldsymbol{X}_l(v)$ which contains $V-1$ terms:

$$\begin{aligned}f(\boldsymbol{X}_l(v))=&\sum_{u=1,u\neq v}^{V}\left(\widetilde{\boldsymbol{A}}_l'\widetilde{\boldsymbol{Y}}(u,v)-\boldsymbol{X}_l(u)'\boldsymbol{D}_l\boldsymbol{X}_l(v)\right)^2+\phi\sum_{u=1,u\neq v}^{V}|\boldsymbol{X}_l(u)'\boldsymbol{D}_l\boldsymbol{X}_l(v)|\\=&\left\|\widetilde{\boldsymbol{A}}_l'\widetilde{\boldsymbol{Y}}(-v,v)-\boldsymbol{X}_l(v)'\boldsymbol{D}_l\boldsymbol{X}_l(-v)'\right\|_2^2+\phi\sum_{u=1,u\neq v}^{V}|\boldsymbol{X}_l(u)'\hat{\boldsymbol{D}}_l\boldsymbol{X}_l(v)|,\end{aligned}$$

which is same as (10). It is clear that $f(\boldsymbol{X}_l(v))$ is convex since the hessian matrix has the form

$$H(\boldsymbol{X}_l(v))=\boldsymbol{D}_l\boldsymbol{X}_l(-v)'\boldsymbol{X}_l(-v)\boldsymbol{D}_l$$

which is not involved with $\boldsymbol{X}_l(v)$ and is positive semi-definite. Then, it is clear to see the block multi-convex property in simplified problem.

2. Proof of Theorem 2.1 Block Multi-Convex:

In 2.1, we will first show the non-convexity by proposing a counter example. Then, we show the block multi-convexity in 2.2.

2.1. Non-convexity in Locus:

We use a toy example to show that problem is not convex. The unknown parameters here are $\Theta = \{\boldsymbol{A}, \boldsymbol{X}_1, .., \boldsymbol{X}_q, \boldsymbol{D}_1, ..., \boldsymbol{D}_q\}$. Denote $f(\Theta) = \sum_{i-1}^{N} \|\boldsymbol{Y}_i - \sum_{l=1}^{q} a_{il}\mathcal{L}(\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_l')\|_2^2 + \phi\sum_{l=1}^{q}\|\mathcal{L}(\boldsymbol{X}_l\boldsymbol{D}_l\boldsymbol{X}_l')\|_1$. When $N = 2$, $\boldsymbol{Y} = [0.33, -0.03, -0.02; 0.46, 0.15, 0.06]$, $\phi = 0$, $q = 2, R_1 = R_2 = 1$, we can find Θ_1 and Θ_2 satisfying that $f(0.5\Theta_1 + 0.5\Theta_2) > 0.5f(\Theta_1) + 0.5f(\Theta_2)$. For example, if we set $\Theta_1 = \{\boldsymbol{A} = \boldsymbol{I}_2, \boldsymbol{D}_1 = 1, \boldsymbol{D}_2 = 1, \boldsymbol{X}_1 = [-0.35, -0.80, -0.50]', \boldsymbol{X}_2 = [-0.86, -0.20, -0.47]'\}$ and $\Theta_2 = \{\boldsymbol{A} = \boldsymbol{I}_2, \boldsymbol{D}_1 = 1, \boldsymbol{D}_2 = 1, \boldsymbol{X}_1 = [0.21, 0.27, 0.94]', \boldsymbol{X}_2 = [-0.06, 0.99, -0.13]'\}$, we have $f(0.5\Theta_1 + 0.5\Theta_2) = 0.542$ and $0.5f(\Theta_1) + 0.5f(\Theta_2) = 0.448$. This shows that f is not convex.

2.2. Block multi-convexity in Locus:

In the paper and Appendix 1, we show that the problem is block multi-convex when $\boldsymbol{A}$ is orthogonal. In this section, we will release this assumption and show a general case. Specifically, we will only show the problem is convex on $\boldsymbol{X}_l(v)$ given other terms.

First, we reformat the function to element version:

$$f = \sum_{i=1}^{N}\sum_{u<v}\left(Y_i(u,v) - \sum_{l=1}^{q} a_{il}\boldsymbol{X}_l(u)\boldsymbol{D}_l\boldsymbol{X}_l(v)'\right)^2 + \phi\sum_{l=1}^{q}\sum_{u<v}|\boldsymbol{X}_l(u)\boldsymbol{D}_l\boldsymbol{X}_l(v)'|$$

Here we focus on term $\boldsymbol{X}_l(v)$ by treating it as unknown $\boldsymbol{x}$, and assume other terms

are given. We have

$$f = \sum_{i=1}^{N} \sum_{u=1,u\neq v}^{V} \Big(Y_i(u,v) - a_{il}\boldsymbol{X}_l(u)\boldsymbol{D}_l\boldsymbol{x} - \sum_{h\neq 1} a_{ih}\boldsymbol{X}_h(u)\boldsymbol{D}_h\boldsymbol{X}_h(v)' \Big)^2 + \phi \sum_{u=1,u\neq v}^{V} |\boldsymbol{X}_l(u)\boldsymbol{D}_l\boldsymbol{x}| + c$$
$$= \sum_{i=1}^{N} \Big(\boldsymbol{Y}_i^{(0)}(-v,v) - a_{il}\boldsymbol{X}_l(-v)\boldsymbol{D}\boldsymbol{x} \Big)^2 + \phi \sum_{u=1,u\neq v}^{V} \|\boldsymbol{X}_l(-v)\boldsymbol{D}_l\boldsymbol{x}\|_1 + c$$

, where c is a constant not involved with $\boldsymbol{X}_l(v)$. Therefore, we have the hessian matrix of $\boldsymbol{X}_l(v)$ to be $\sum_{i=1}^{N} a_{il}^2 \boldsymbol{D}_l\boldsymbol{X}_l(-v)'\boldsymbol{X}_l(-v)\boldsymbol{D}_l$, which is a positive semi-definite matrix. This finishes the proof.

3. Analytic Solution for $\boldsymbol{X}_l(v)$

In this section, we show that when $\boldsymbol{X}_l(-v)$ is full rank, we can have the analytic solution for $\boldsymbol{X}_l(v)$ for (10) based on (11-12). First, we set $\boldsymbol{S}_l(-v,v) = \boldsymbol{X}_l(-v)\boldsymbol{D}_l\boldsymbol{X}_l(v)$ and treat $\boldsymbol{S}_l(-v,v)$ as unknown parameter in (10) which will be same as (11). As shown in Fan and Li (2001), (11) has analytic solution as

$$\hat{\boldsymbol{S}}_l(-v,v) = \mathrm{diag}\Big(\mathrm{sgn}\big(\widetilde{\boldsymbol{Y}}(-v,v)'\widetilde{\boldsymbol{A}}_l\big)\Big)\big(|\widetilde{\boldsymbol{Y}}(-v,v)'\widetilde{\boldsymbol{A}}_l| - \frac{\phi}{2}\mathbf{1}_{V-1}\big)_+,$$

where sgn represents sign function for each element. Next, we map $\hat{\boldsymbol{S}}_l(-v,v)$ to the column space of $\boldsymbol{X}_l(-v)\boldsymbol{D}_l$ via regression framework and when $\boldsymbol{X}_l(-v)$ and $\boldsymbol{D}_l$ are full rank we have

$$\hat{\boldsymbol{X}}_l(v) = \boldsymbol{D}_l^{-1}\big(\boldsymbol{X}_l(-v)'\boldsymbol{X}_l(-v)\big)^{-1}\boldsymbol{X}_l(-v)'\hat{\boldsymbol{S}}_l(-v,v)$$
$$= \boldsymbol{D}_l^{-1}\big(\boldsymbol{X}_l(-v)'\boldsymbol{X}_l(-v)\big)^{-1}\boldsymbol{X}_l(-v)'\mathrm{diag}\Big(\mathrm{sgn}\big(\widetilde{\boldsymbol{Y}}(-v,v)'\widetilde{\boldsymbol{A}}_l\big)\Big)\big(|\widetilde{\boldsymbol{Y}}(-v,v)'\widetilde{\boldsymbol{A}}_l| - \frac{\phi}{2}\mathbf{1}_{V-1}\big)_+.$$

Therefore, the Node-Moving algorithm has analytic solution at each step without the need of gradient-based numerical approximation and hence is highly efficient and reliable.

Bibliography

Airoldi, E. M., Blei, D. M., Fienberg, S. E. and Xing, E. P. (2008), 'Mixed membership stochastic blockmodels', Journal of Machine Learning Research **9**(Sep), 1981–2014.

Amico, E. and Goñi, J. (2018a), 'Mapping hybrid functional-structural connectivity traits in the human connectome', Network Neuroscience **2**(3), 306–322.

Amico, E. and Goñi, J. (2018b), 'The quest for identifiability in human functional connectomes', Scientific reports **8**(1), 8254.

Amico, E., Marinazzo, D., Di Perri, C., Heine, L., Annen, J., Martial, C., Dzemidzic, M., Kirsch, M., Bonhomme, V., Laureys, S. et al. (2017), 'Mapping the functional connectome traits of levels of consciousness', NeuroImage **148**, 201–211.

Attias, H. (2000), 'A variational bayesian framework for graphical models', Advances in neural information processing systems **12**(1-2), 209–215.

Beckmann, C. F., DeLuca, M., Devlin, J. T. and Smith, S. M. (2005), 'Investigations into resting-state connectivity using independent component analysis', Philosophical Transactions of the Royal Society of London B: Biological Sciences **360**(1457), 1001–1013.

Beckmann, C. F. and Smith, S. M. (2004), 'Probabilistic independent component analysis for functional magnetic resonance imaging', Medical Imaging, IEEE Transactions on **23**(2), 137–152.

Beckmann, C. F. and Smith, S. M. (2005), 'Tensorial extensions of independent component analysis for multisubject fmri analysis', Neuroimage **25**(1), 294–311.

Biswal, B. B. and Ulmer, J. L. (1999), 'Blind source separation of multiple signal sources of fmri data sets using independent component analysis', Journal of computer assisted tomography **23**(2), 265–271.

Biswal, B., Zerrin Yetkin, F., Haughton, V. M. and Hyde, J. S. (1995), 'Functional connectivity in the motor cortex of resting human brain using echo-planar mri', Magnetic resonance in medicine **34**(4), 537–541.

Bowman, F. D. (2014), 'Brain imaging analysis', Annual review of statistics and its application **1**, 61–85.

Bullmore, E. and Sporns, O. (2009), 'Complex brain networks: graph theoretical analysis of structural and functional systems', Nature Reviews Neuroscience **10**(3), 186–198.

Button, K. S., Ioannidis, J. P., Mokrysz, C., Nosek, B. A., Flint, J., Robinson, E. S. and Munafò, M. R. (2013), 'Power failure: why small sample size undermines the reliability of neuroscience', Nature Reviews Neuroscience **14**(5), 365–376.

Calhoun, V., Adali, T., Pearlson, G. and Pekar, J. (2001), 'A method for making group inferences from functional mri data using independent component analysis', Human brain mapping **14**(3), 140–151.

Calhoun, V. D., Adali, T., Hansen, L. K., Larsen, J. and Pekar, J. J. (2003), Ica of functional mri data: an overview, in 'in Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation', Citeseer.

Calhoun, V. D., Adali, T., McGinty, V., Pekar, J. J., Watson, T. and Pearlson, G. (2001), 'fmri activation in a visual-perception task: network of areas detected

using the general linear model and independent components analysis', NeuroImage **14**(5), 1080–1088.

Calhoun, V. D., Liu, J. and Adalı, T. (2009), 'A review of group ica for fmri data and ica for joint inference of imaging, genetic, and erp data', Neuroimage **45**(1), S163–S172.

Chen, K., Dong, H. and Chan, K.-S. (2013), 'Reduced rank regression via adaptive nuclear norm penalization', Biometrika **100**(4), 901–920.

Cheng, Q., Gao, X., Martin, R. et al. (2014), 'Exact prior-free probabilistic inference on the heritability coefficient in a linear mixed model', Electronic Journal of Statistics **8**(2), 3062–3076.

Chumbley, J. R. and Friston, K. J. (2009), 'False discovery rate revisited: Fdr and topological inference using gaussian random fields', Neuroimage **44**(1), 62–70.

Chung, M. K. (2018), 'Statistical challenges of big brain network data', Statistics & probability letters **136**, 78–82.

Church, J. A., Fair, D. A., Dosenbach, N. U., Cohen, A. L., Miezin, F. M., Petersen, S. E. and Schlaggar, B. L. (2008), 'Control networks in paediatric tourette syndrome show immature and anomalous patterns of functional connectivity', Brain **132**(1), 225–238.

Contreras, J. A., Goñi, J., Risacher, S. L., Amico, E., Yoder, K., Dzemidzic, M., West, J. D., McDonald, B. C., Farlow, M. R., Sporns, O. et al. (2017), 'Cognitive complaints in older adults at risk for alzheimer's disease are associated with altered resting-state networks', Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring **6**, 40–49.

Cook, R. D. and Forzani, L. (2008), 'Covariance reducing models: An alternative to spectral modelling of covariance matrices', Biometrika **95**(4), 799–812.

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

Dai, T., Guo, Y., Initiative, A. D. N. et al. (2017), 'Predicting individual brain functional connectivity using a bayesian hierarchical model', NeuroImage **147**, 772–787.

Daubechies, I., Roussos, E., Takerkart, S., Benharrosh, M., Golden, C., D'ardenne, K., Richter, W., Cohen, J. and Haxby, J. (2009), 'Independent component analysis for brain fmri does not select for independence', Proceedings of the National Academy of Sciences **106**(26), 10415–10422.

Deco, G., Jirsa, V. K. and McIntosh, A. R. (2011), 'Emerging concepts for the dynamical organization of resting-state activity in the brain', Nature Reviews Neuroscience **12**(1), 43.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977), 'Maximum likelihood from incomplete data via the em algorithm', Journal of the royal statistical society. Series B (methodological) pp. 1–38.

Dettwiler, A., Murugavel, M., Putukian, M., Cubon, V., Furtado, J. and Osherson, D. (2014), 'Persistent differences in patterns of brain activation after sports-related concussion: a longitudinal functional magnetic resonance imaging study', Journal of neurotrauma **31**(2), 180–188.

Dodero, L., Minh, H. Q., San Biagio, M., Murino, V. and Sona, D. (2015), Kernel-based classification for brain connectivity graphs on the riemannian manifold of pos-

itive definite matrices, in '2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)', IEEE, pp. 42–45.

Durante, D., Dunson, D. B. and Vogelstein, J. T. (2017), 'Nonparametric bayes modeling of populations of networks', Journal of the American Statistical Association **112**(520), 1516–1530.

Fan, J., Gong, W. and Zhu, Z. (2017), 'Generalized high-dimensional trace regression via nuclear norm regularization', arXiv preprint arXiv:1710.08083 .

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

Fan, J., Wang, W. and Zhu, Z. (2016), 'A shrinkage principle for heavy-tailed data: High-dimensional robust low-rank matrix recovery', arXiv preprint arXiv:1603.08315 .

Finn, E. S., Shen, X., Scheinost, D., Rosenberg, M. D., Huang, J., Chun, M. M., Papademetris, X. and Constable, R. T. (2015), 'Functional connectome fingerprinting: identifying individuals using patterns of brain connectivity', Nature neuroscience **18**(11), 1664.

Franks, A. and Hoff, P. (2016), 'Shared subspace models for multi-group covariance estimation', arXiv preprint arXiv:1607.03045 .

Gao, X., Ombao, H. and Gillen, D. (2017), 'Fisher information matrix of binary time series', arXiv preprint arXiv:1711.05483 .

Gao, X., Shahbaba, B. and Ombao, H. (2017), 'Modeling binary time series using gaussian processes with application to predicting sleep states', arXiv preprint arXiv:1711.05466 .

Gao, X., Shen, W. and Ombao, H. (2018), 'Regularized matrix data clustering and its application to image analysis', arXiv preprint arXiv:1808.01749 .

Genovese, C. R., Lazar, N. A. and Nichols, T. (2002), 'Thresholding of statistical maps in functional neuroimaging using the false discovery rate', Neuroimage **15**(4), 870–878.

Glasser, M. F., Sotiropoulos, S. N., Wilson, J. A., Coalson, T. S., Fischl, B., Andersson, J. L., Xu, J., Jbabdi, S., Webster, M., Polimeni, J. R. et al. (2013), 'The minimal preprocessing pipelines for the human connectome project', Neuroimage **80**, 105–124.

Goh, G., Dey, D. K. and Chen, K. (2017), 'Bayesian sparse reduced rank multivariate regression', Journal of multivariate analysis **157**, 14–28.

Gong, G., He, Y., Concha, L., Lebel, C., Gross, D. W., Evans, A. C. and Beaulieu, C. (2008), 'Mapping anatomical connectivity patterns of human cerebral cortex using in vivo diffusion tensor imaging tractography', Cerebral cortex **19**(3), 524–536.

Goodfellow, I., Bengio, Y. and Courville, A. (2016), Deep learning, MIT press.

Greicius, M. D., Krasnow, B., Reiss, A. L. and Menon, V. (2003), 'Functional connectivity in the resting brain: a network analysis of the default mode hypothesis', Proceedings of the National Academy of Sciences **100**(1), 253–258.

Greicius, M. D., Srivastava, G., Reiss, A. L. and Menon, V. (2004), 'Default-mode network activity distinguishes alzheimer's disease from healthy aging: evidence from functional mri', Proceedings of the National Academy of Sciences of the United States of America **101**(13), 4637–4642.

Guo, Y. (2011), 'A general probabilistic model for group independent component analysis and its estimation methods', Biometrics **67**(4), 1532–1542.

Guo, Y. and Pagnoni, G. (2008), 'A unified framework for group independent component analysis for multi-subject fmri data', NeuroImage **42**(3), 1078–1093.

Guo, Y. and Tang, L. (2013), 'A hierarchical model for probabilistic independent component analysis of multi-subject fmri studies', Biometrics **69**(4), 970–981.

Hagmann, P., Thiran, J.-P., Jonasson, L., Vandergheynst, P., Clarke, S., Maeder, P. and Meuli, R. (2003), 'Dti mapping of human brain connectivity: statistical fibre tracking and virtual dissection', Neuroimage **19**(3), 545–554.

Heaven, D. (2019), 'Why deep-learning ais are so easy to fool', Nature **574**(7777), 163.

Himberg, J., Hyvärinen, A. and Esposito, F. (2004), 'Validating the independent components of neuroimaging time series via clustering and visualization', Neuroimage **22**(3), 1214–1222.

Hoff, P. (2008), Modeling homophily and stochastic equivalence in symmetric relational data, in 'Advances in neural information processing systems', pp. 657–664.

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

Hu, C., Ju, R., Shen, Y., Zhou, P. and Li, Q. (2016), Clinical decision support for alzheimer's disease based on deep learning and brain network, in '2016 IEEE International Conference on Communications (ICC)', IEEE, pp. 1–6.

Hyvärinen, A., Karhunen, J. and Oja, E. (2001), Independent component analysis, Vol. 46, John Wiley & Sons.

Hyvärinen, A. and Oja, E. (2000), 'Independent component analysis: algorithms and applications', Neural networks **13**(4), 411–430.

Jie, B., Zhang, D., Wee, C.-Y. and Shen, D. (2014), 'Topological graph kernel on multiple thresholded functional connectivity networks for mild cognitive impairment classification', Human brain mapping **35**(7), 2876–2897.

Johnson, K. A., Minoshima, S., Bohnen, N. I., Donohoe, K. J., Foster, N. L., Herscovitch, P., Karlawish, J. H., Rowe, C. C., Carrillo, M. C., Hartley, D. M. et al. (2013), 'Appropriate use criteria for amyloid pet: a report of the amyloid imaging task force, the society of nuclear medicine and molecular imaging, and the alzheimer's association', Journal of Nuclear Medicine **54**(3), 476–490.

Kawahara, J., Brown, C. J., Miller, S. P., Booth, B. G., Chau, V., Grunau, R. E., Zwicker, J. G. and Hamarneh, G. (2017), 'Brainnetcnn: Convolutional neural networks for brain networks; towards predicting neurodevelopment', NeuroImage **146**, 1038–1049.

Kemmer, P. B., Guo, Y., Wang, Y. and Pagnoni, G. (2015), 'Network-based characterization of brain functional connectivity in zen practitioners', Frontiers in psychology **6**.

Kemmer, P. B., Wang, Y., Bowman, F. D., Mayberg, H. and Guo, Y. (2018), 'Evaluating the strength of structural connectivity underlying brain functional networks', Brain Connectivity **8**(10), 579–594.

Kiefer, A. W., Barber Foss, K., Reches, A., Gadd, B., Gordon, M., Rushford, K., Laufer, I., Weiss, M. and Myer, G. D. (2015), 'Brain network activation as a novel biomarker for the return-to-play pathway following sport-related brain injury', Frontiers in neurology **6**, 243.

Kostantinos, N. (2000), 'Gaussian mixtures and their applications to signal processing', Advanced Signal Processing Handbook: Theory and Implementation for Radar, Sonar, and Medical Imaging Real Time Systems .

Lang, E. W., Tomé, A. M., Keck, I. R., Górriz-Sáez, J. and Puntonet, C. G. (2012), 'Brain connectivity analysis: a short survey', Computational intelligence and neuroscience **2012**, 8.

Lee, M. H., Smyser, C. D. and Shimony, J. S. (2013), 'Resting-state fmri: a review of methods and clinical applications', American Journal of Neuroradiology **34**(10), 1866–1872.

Lee, S., Zipunnikov, V., Reich, D. S. and Pham, D. L. (2015), 'Statistical image analysis of longitudinal ravens images', Frontiers in neuroscience **9**, 368.

Li, X., Xu, D., Zhou, H. and Li, L. (2018), 'Tucker tensor regression and neuroimaging analysis', Statistics in Biosciences **10**(3), 520–545.

Li, Y., Zhu, H., Chen, Y., An, H., Gilmore, J., Lin, W. and Shen, D. (2009), Lst-gee: Longitudinal analysis of neuroimaging data, in 'Medical Imaging 2009: Image Processing', Vol. 7259, International Society for Optics and Photonics, p. 72590F.

Liu, J., Li, M., Pan, Y., Lan, W., Zheng, R., Wu, F.-X. and Wang, J. (2017), 'Complex brain network analysis and its applications to brain disorders: a survey', Complexity **2017**.

Lukemire, J., Wang, Y., Verma, A. and Guo, Y. (2018), 'Hint: A toolbox for hierarchical modeling of neuroimaging data', arXiv preprint arXiv:1803.07587 .

Marblestone, A. H., Wayne, G. and Kording, K. P. (2016), 'Toward an integration of deep learning and neuroscience', Frontiers in computational neuroscience **10**, 94.

Mckeown, M. J., Makeig, S., Brown, G. G., Jung, T.-P., Kindermann, S. S., Kindermann, R. S., Bell, A. J. and Sejnowski, T. J. (1998), 'Analysis of fmri data by blind separation into independent spatial components', Human Brain Mapping **6**, 160–188.

McLachlan, G. and Peel, D. (2004), Finite mixture models, John Wiley & Sons.

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

Minka, T. P. (2000), Automatic choice of dimensionality for pca, in 'NIPS', Vol. 13, pp. 598–604.

Munsell, B. C., Wee, C.-Y., Keller, S. S., Weber, B., Elger, C., da Silva, L. A. T., Nesland, T., Styner, M., Shen, D. and Bonilha, L. (2015), 'Evaluation of machine learning algorithms for treatment outcome prediction in patients with epilepsy based on structural connectome data', Neuroimage **118**, 219–230.

Nowicki, K. and Snijders, T. A. B. (2001), 'Estimation and prediction for stochastic blockstructures', Journal of the American statistical association **96**(455), 1077–1087.

Parisot, S., Ktena, S. I., Ferrante, E., Lee, M., Guerrero, R., Glocker, B. and Rueckert, D. (2018), 'Disease prediction using graph convolutional networks: application to autism spectrum disorder and alzheimer's disease', Medical image analysis **48**, 117–130.

Park, T. and Casella, G. (2008), 'The bayesian lasso', Journal of the American Statistical Association **103**(482), 681–686.

Poil, S.-S., De Haan, W., van der Flier, W. M., Mansvelder, H. D., Scheltens, P. and Linkenkaer-Hansen, K. (2013), 'Integrative eeg biomarkers predict progression to alzheimer's disease at the mci stage', Frontiers in aging neuroscience **5**, 58.

Power, J. D., Cohen, A. L., Nelson, S. M., Wig, G. S., Barnes, K. A., Church, J. A., Vogel, A. C., Laumann, T. O., Miezin, F. M., Schlaggar, B. L. et al. (2011), 'Functional network organization of the human brain', Neuron **72**(4), 665–678.

Qiu, A., Mori, S. and Miller, M. I. (2015), 'Diffusion tensor imaging for understanding brain development in early life', Annual review of psychology **66**, 853–876.

Rabusseau, G. and Kadri, H. (2016), Low-rank regression with tensor responses, in 'Advances in Neural Information Processing Systems', pp. 1867–1875.

Raskutti, G. and Yuan, M. (2015), 'Convex regularization for high-dimensional tensor regression', arXiv preprint arXiv:1512.01215 **639**.

Rubinov, M. and Sporns, O. (2010), 'Complex network measures of brain connectivity: uses and interpretations', Neuroimage **52**(3), 1059–1069.

Satterthwaite, T. D., Elliott, M. A., Ruparel, K., Loughead, J., Prabhakaran, K., Calkins, M. E., Hopson, R., Jackson, C., Keefe, J., Riley, M. et al. (2014), 'Neuroimaging of the philadelphia neurodevelopmental cohort', Neuroimage **86**, 544–553.

Satterthwaite, T. D., Wolf, D. H., Roalf, D. R., Ruparel, K., Erus, G., Vandekar, S., Gennatas, E. D., Elliott, M. A., Smith, A., Hakonarson, H. et al. (2014), 'Linked sex differences in cognition and functional connectivity in youth', Cerebral cortex **25**(9), 2383–2394.

Seeley, W. W., Crawford, R. K., Zhou, J., Miller, B. L. and Greicius, M. D. (2009), 'Neurodegenerative diseases target large-scale human brain networks', Neuron **62**(1), 42–52.

Shi, R. (2016), Some Novel Statistical Methods for Neuroimaging Data Analysis, PhD thesis, Emory University.

Shi, R. and Guo, Y. (2016), 'Investigating differences in brain functional networks using hierarchical covariate-adjusted independent component analysis', The annals of applied statistics **10**(4), 1930.

Smith, S. M., Fox, P. T., Miller, K. L., Glahn, D. C., Fox, P. M., Mackay, C. E., Filippini, N., Watkins, K. E., Toro, R., Laird, A. R. et al. (2009), 'Correspondence of the brain's functional architecture during activation and rest', Proceedings of the National Academy of Sciences **106**(31), 13040–13045.

Smith, S. M., Miller, K. L., Salimi-Khorshidi, G., Webster, M., Beckmann, C. F., Nichols, T. E., Ramsey, J. D. and Woolrich, M. W. (2011), 'Network modelling methods for fmri', Neuroimage **54**(2), 875–891.

Solo, V., Poline, J.-B., Lindquist, M. A., Simpson, S. L., Bowman, F. D., Chung, M. K. and Cassidy, B. (2018), 'Connectivity in fmri: blind spots and breakthroughs', IEEE transactions on medical imaging **37**(7), 1537–1550.

Storey, J. D. (2011), False discovery rate, in 'International encyclopedia of statistical science', Springer, pp. 504–508.

Sun, W. W. and Li, L. (2017), 'Store: sparse tensor response regression and neuroimaging analysis', The Journal of Machine Learning Research **18**(1), 4908–4944.

Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B. and Joliot, M. (2002), 'Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain', Neuroimage **15**(1), 273–289.

Verbeke, G. (1997), Linear mixed models for longitudinal data, in 'Linear mixed models in practice', Springer, pp. 63–153.

Virta, J., Li, B., Nordhausen, K. and Oja, H. (2017), 'Independent component analysis for tensor-valued data', Journal of Multivariate Analysis **162**, 172–192.

Wada, A., Tsuruta, K., Irie, R., Kamagata, K., Maekawa, T., Fujita, S., Koshino, S., Kumamaru, K., Suzuki, M., Nakanishi, A. et al. (2019), 'Differentiating alzheimer's

disease from dementia with lewy bodies using a deep learning technique based on structural brain connectivity', Magnetic Resonance in Medical Sciences **18**(3), 219.

Wang, L., Durante, D., Jung, R. E. and Dunson, D. B. (2017), 'Bayesian network–response regression', Bioinformatics **33**(12), 1859–1866.

Wang, Y. and Guo, Y. (2019), 'A hierarchical independent component analysis model for longitudinal neuroimaging studies', NeuroImage **189**, 380–400.

Wang, Y., Kang, J., Kemmer, P. B. and Guo, Y. (2016), 'An efficient and reliable statistical method for estimating functional connectivity in large scale brain networks using partial correlation', Frontiers in neuroscience **10**.

Wang, Y., Wu, H. and Yu, T. (2017), 'Differential gene network analysis from single cell rna-seq', Journal of Genetics and Genomics **44**(6), 331–334.

Wang, Y., Zhao, Y., Zhang, L., Liang, J., Zeng, M. and Liu, X. (2013), 'Graph construction based on re-weighted sparse representation for semi-supervised learning', JOURNAL OF INFORMATION &COMPUTATIONAL SCIENCE **10**(2), 375–383.

Wu, G.-R., Stramaglia, S., Chen, H., Liao, W. and Marinazzo, D. (2013), 'Mapping the voxel-wise effective connectome in resting state fmri', PloS one **8**(9), e73670.

Wu, J., Quinlan, E. B., Dodakian, L., McKenzie, A., Kathuria, N., Zhou, R. J., Augsburger, R., See, J., Le, V. H., Srinivasan, R. et al. (2015), 'Connectivity measures are robust biomarkers of cortical function and plasticity after stroke', Brain **138**(8), 2359–2369.

Wu, K., Taki, Y., Sato, K., Qi, H., Kawashima, R. and Fukuda, H. (2013), 'A longitudinal study of structural brain network changes with normal aging', Frontiers in human neuroscience **7**, 113.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C. and Yu, P. S. (2019), 'A comprehensive survey on graph neural networks', arXiv preprint arXiv:1901.00596 .

Xia, Y. and Li, L. (2017), 'Hypothesis testing of matrix graph model with application to brain connectivity analysis', Biometrics **73**(3), 780–791.

Xu, L., Cheung, C., Yang, H. and Amari, S. (1997), Maximum equalization by entropy maximization and mixture of cumulative distribution functions, in 'Proc. of ICNN97', pp. 1821–1826.

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

Zhang, J., Zhou, L., Wang, L. and Li, W. (2015), 'Functional brain network classification with compact representation of sice matrices', IEEE Transactions on Biomedical Engineering **62**(6), 1623–1634.

Zhao, X.-H., Wang, P.-J., Li, C.-B., Hu, Z.-H., Xi, Q., Wu, W.-Y. and Tang, X.-W. (2007), 'Altered default mode network activity in patient with anxiety disorders: an fmri study', European Journal of Radiology **63**(3), 373–378.

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

Zhou, H., Li, L. and Zhu, H. (2013), 'Tensor regression with applications in neuroimaging data analysis', Journal of the American Statistical Association **108**(502), 540–552.