Statistical Methods for Imaging Data, Imaging Genetics and Sparse Estimation in Linear Mixed Models

by

Eugene A. Opoku
B.Sc., University of Ghana, 2012
M.Sc., Brock University, 2015

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Mathematics and Statistics

Statistical Methods for Imaging Data, Imaging Genetics and Sparse Estimation in Linear Mixed Models

by

Eugene A. Opoku
B.Sc., University of Ghana, 2012
M.Sc., Brock University, 2015

Supervisory Committee

Dr. Farouk Nathoo., Co-Supervisor
(Department of Mathematics and Statistics)

Dr. Ejaz S. Ahmed., Co-Supervisor
(Department of Mathematics and Statistics)

Dr. Mary Lesperance, Departmental Member
(Department of Mathematics and Statistics)

Dr. Zhang Xuekui, Departmental Member
(Department of Mathematics and Statistics)

Supervisory Committee

Dr. Farouk Nathoo., Co-Supervisor
(Department of Mathematics and Statistics)

Dr. Ejaz S. Ahmed., Co-Supervisor
(Department of Mathematics and Statistics)

Dr. Mary Lesperance, Departmental Member
(Department of Mathematics and Statistics)

Dr. Zhang Xuekui, Departmental Member
(Department of Mathematics and Statistics)

ABSTRACT

This thesis presents research focused on developing statistical methods with emphasis on techniques that can be used for the analysis of data in imaging studies and sparse estimations for applications in high-dimensional data. The first contribution addresses the pixel/voxel-labeling problem for spatial hidden Markov models in image analysis. We formulate a Gaussian spatial mixture model with Potts model used as a prior for mixture allocations for the latent states in the model. Jointly estimating the model parameters, the discrete state variables and the number of states (number of mixture components) is recognized as a difficult combinatorial optimization. To overcome drawbacks associated with local algorithms, we implement and make comparisons between iterated conditional modes (ICM), simulated annealing (SA) and hybrid ICM with ant colony system (ACS-ICM) optimization for pixel labelling, parameter estimation and mixture component estimation.

In the second contribution, we develop ACS-ICM algorithm for spatiotemporal modeling of combined MEG/EEG data for computing estimates of the neural source

activity. We consider a Bayesian finite spatial mixture model with a Potts model as a spatial prior and implement the ACS-ICM for simultaneous point estimation and model selection for the number of mixture components. Our approach is evaluated using simulation studies and an application examining the visual response to scrambled faces. In addition, we develop a nonparametric bootstrap for interval estimation to account for uncertainty in the point estimates. In the third contribution, we present sparse estimation strategies in linear mixed model (LMM) for longitudinal data. We address the problem of estimating the fixed effects parameters of the LMM when the model is sparse and predictors are correlated. We propose and derive the asymptotic properties of the pretest and shrinkage estimation strategies. Simulation studies is performed to compare the numerical performance of the Lasso and adaptive Lasso estimators with the pretest and shrinkage ridge estimators. The methodology is evaluated through an application of a high-dimensional data examining effective brain connectivity and genetics.

In the fourth and final contribution, we conduct an imaging genetics study to explore how effective brain connectivity in the default mode network (DMN) may be related to genetics within the context of Alzheimer's disease. We develop an analysis of longitudinal resting-state functional magnetic resonance imaging (rs-fMRI) and genetic data obtained from a sample of 111 subjects with a total of 319 rs-fMRI scans from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database. A Dynamic Causal Model (DCM) is fit to the rs-fMRI scans to estimate effective brain connectivity within the DMN and related to a set of single nucleotide polymorphisms (SNPs) contained in an empirical disease-constrained set. We relate longitudinal effective brain connectivity estimated using spectral DCM to SNPs using both linear mixed effect (LME) models as well as function-on-scalar regression (FSR).

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

I would like to thank:

my Supervisors, Dr. Farouk Nathoo and Prof. Ejaz Ahmed, for research guidance, mentoring, support, encouragement, and patience.

Committee members, for for helpful comments and insights.

DEDICATION

I dedicate this work to my wife Abigial Andoh and my wonderful son Jayden for their support and love throughout the entire doctorate program.
A special feeling of gratitude to my loving parents, Juliana and Dominic Opoku, whose words of encouragement and push for tenacity ring in my ears. My siblings Millicent, Jephter and Obed have never left my side and are very special.

Chapter 1

Introduction

1.1 Background

This thesis mainly focuses on development of statistical methods for applications in imaging data, imaging genetics and sparse estimations in the linear mixed model for longitudinal data. This wide range of problems motivated by both theory and applications provides significant challenges for the development and evaluation of statistical methodology. In part we address the use of statistical techniques to obtain better solutions of difficult combinatorial optimization problems that frequently occur in numerous statistical applications. This thesis develops the Ant colony system (ACS) algorithm in Chapter 2 and 3 as a tool to help investigate problems in imaging analysis including source reconstruction and pixel labeling problems to improve the overall performance over local algorithms. Improvements include higher posterior density values, pixel labelling accuracy and efficient mixture component estimation.

In addition, in Chapter 4 we consider a field pertaining to the development of statistical methods of inference for longitudinal data in the context of sparse estimation strategies. These strategies are motivated by shrinkage and pretest estimation procedures by Ahmed and Nicol (2012), Ahmed and Raheem (2012), and Lisawadi et al. (2016). This area of research is discussed further in Chapter 4 which involve estimation problems for the sparse linear mixed model. In Chapter 5 we develop analytical methods for the joint analysis of imaging phenotypes and high-dimensional genetic data which presents computational and theoretical challenges for existing analytical methods. We apply statistical methods in Chapter 5 to analyze brain imaging and genetic data related to Alzheimer disease to explore the genetic architecture of brain

connectivity. The methodological approaches applied in this thesis are designed to increase the power to detect small genetic effects in imaging genetic studies, first by reducing the complexity of the data using an out of sample genome-wide scan of disease to select priority subset of single nucleotide polymorphism (SNPs); and second by relating imaging phenotypes to SNPs using linear mixed model and functional mixed model with varying assumptions about data generating mechanism.

In Chapter 2 we consider the investigation of optimization algorithms for spatial hidden Markov models (HMMs) for the pixel labeling problem. The pixel labeling problem involves assigning a label from some set labels to each pixel in an image. This problem is phrased in terms of combinatorial optimization and based on Markov Random Fields (MRF). The MRF based pixel labelling allows the label selection of a pixel to be conditioned on the local interaction between pixel and its neighbors within a neighborhood system. Combinatorial optimization for pixel labeling has been extensively used in image segmentation and computer vision (see, e.g., Destrempes et al., 2005; Ouadfel et al., 2003; Szeliski et al., 2008).

We consider the framework where the data is assumed to constitute a 2-dimensional image with each pixel represented by a continuous measurement and where the goal is to classify each pixel into one of a finite number of states. We develop a spatial mixture model with Gaussian components and labeling based on the Potts model. The Gaussian mixture model (GMM) incorporates a labelling process allocating each pixel to one of the latent states. The labelling process is assumed to follow a Potts model which allows for spatial dependence among neighboring pixels. The optimal labeling of the image pixels is estimated by maximizing a posterior probability (MAP) of the labelling space given the image pixels. The GMM framework involves jointly estimating the model parameters, the discrete state variables and the number of states (number of mixture components). This problem is known to be a difficult combinatorial optimization problem and as a result there has been considerable discussion of algorithms and applications for dealing with such problems where solutions can be efficiently computed (see, e.g., Kim et al., 2000; Green et al., 2002; Kato et al., 2006; Zhang et al., 2007; Destrempes et al., 2005).

To address this problem, we consider an application of iterated conditional modes (ICM) and simulated annealing (SA), a popular and known technique for pixel labelling and parameter estimation. We discuss hybrid ICM with ant colony system (ACS-ICM) optimization as a better alternative. A comparative analysis is performed between the three algorithms through simulation studies under different levels of spa-

tial dependence in the true image. An important contribution of our study is that we consider the case where the number of latent states is unknown and we develop an estimator based on automatic pruning of redundant mixture components for mixture components estimation.

In Chapter 3 we extend our investigation to the development of an ACS- ICM algorithm for computing solutions to inverse problems. The hybrid ACS-ICM algorithm is applied for computing estimates of neural source activity for a combined EEG and MEG data. Magnetoencephalography (MEG) and electroencephalography (EEG) are neuroimaging modalities used to study the function of the brain non-invasively using an array of sensors placed on (EEG) or above the scalp (MEG). These sensor arrays can be used to capture the time-varying electromagnetic field that exists around the head as a result of electrical neural activity within the brain. This work addresses the problem of reconstruction of the electrical activity of the brain from the observed EEG/MEG data which is recognized as an ill-posed inverse problem (von Helmholtz, 1853). The neuro-electromagnetic inverse problem is challenging and ill-posed due to its non-uniqueness because an infinite number of different source distributions can produce the same electromagnetic field on the scalp.

The methods that have been proposed in statistical literature for solving the neuroelectromagnetic inverse problem fall into several categories. The approach can be probabilistic/Bayesian including maximum a posterior estimate or maximum likelihood estimates. Wipf and Nagarajan (2009) provide an extensive review of this statistical problem and discuss recent developments on spatiotemporal analysis of MEG/EEG data including the multiple sparse priors (MSP) approach. Friston et al. (2008) propose an approach based on Gaussian scale mixtures incorporating a potentially large number of covariance components representing spatial patterns of neural activity. This has been extended recently by Henson et al. (2010) who develop methodology for incorporating the spatial patterns found in fMRI data into the covariance components of the MEG/EEG model. Nathoo et al. (2014) propose a spatial mixture formulation where the profile of electrical activity within the brain is represented through location-specific spike-and-slab priors based on a spatial logistic specification. Daunizeau and Friston (2007) proposed meso-state space model (MSM) that models the cortical activity in terms of a set of locally distributed and temporally coherent meso-sources for either MEG or EEG data. Song et al. (2019) builds on the MSM and develop a joint model that combines MEG/EEG data, whereby brain neural activity is modeled from the Gaussian spatial mixture model. The neural

source activity is described in terms of a few hidden states, with each state having its own dynamics and a Potts model used in representing the spatial dependence in the mixture model.

Iterated Conditional Modes (ICM) algorithm is implemented by Song et al. (2019) for simultaneous point estimation and model selection for the latent Gaussian mixture model. While useful, ICM present significant challenges such as sensitivity to initial values and convergence of the algorithm heavy dependence on the initial values chosen. To address this issue and provide better solution, we design and implement Ant Colony System (ACS) optimization coupled with Iterated Conditional Modes (ICM) for computing estimates of the neural source activity and incorporate a non-parametric bootstrap for interval estimation. The proposed methodology is compared with ICM through simulation studies, and is applied to the analysis of a multimodal neuroimaging study examining the neural response to face perception.

In Chapter 4, we present efficient sparse estimation strategies in linear mixed model for applications in high-dimensional setting. The linear mixed model (LMM) is used to model longitudinal and repeated measurements data in the fields of medical and biological science applications. Statistical inference on parameters of the LMM has been extensively studied (see Searle et al., (2006) for comprehensive review). A considerable amount of research has been produced on estimation of fixed and random effects using either maximum likelihood or restricted maximum likelihood (see, e.g., Gumedze and Dunne, 2011; Lindstrom and Bates, 1988; Verbeke and Molenberghs, 2009).

In statistical inference, determining the appropriate statistical model estimator for use in representing the sample data is an interesting and challenging problem. We investigate the fixed effect estimation problem for sparse model with correlated predictors. A number of approaches including pretest and shrinkage estimation have been proposed for regression analysis (see, e.g., Ahmed and Nicol, 2012; Lisawadi et al., 2016; Ahmed and Raheem, 2012; Raheem et al., 2012; Ahmed and Opoku, 2017). Ahmed (2012) provide a comprehensive review of Stein-type shrinkage and pretest estimation for large sample estimation in classical multiple regression model for estimation of regression coefficients. More recently, Ahmed and Opoku (2017) propose an improved estimation strategy using sub-model selection and post-estimation for the LMM. Within this framework, linear shrinkage and shrinkage pretest estimation strategies are developed which combine full model and sub-model estimators in an effective way as a trade-off between bias and variance. Raheem, Ahmed, and Dok-

sum (2012) extend this study by using a likelihood ratio test to develop James-Stein shrinkage and pretest estimation methods based on LMM for longitudinal data.

The problem of multicollinearity among predictor variables is common for high-dimensional data. In this case, some biased estimations, such as shrinkage estimation, partial least squares estimation (Geladi and Kowalski, 1986) and Liu estimators (Liu, 2003) are proposed to deal with the multicollinearity. We focus on the estimation of fixed effects parameters in linear mixed models when there are many potential predictors that have a weak or no influence on the response of interest. We introduce three estimation strategies using ridge estimation technique as the baseline estimator. These strategies are motivated by pretest estimation and shrinkage estimation procedures. In the context of two competing LMM (the full model and the candidate sub-model), we propose the pretest and shrinkage estimation techniques that shrinks the full model estimate in the direction of the sub-model estimate. The asymptotic properties of the pretest and shrinkage estimators including the derivation of bias and risk are established. Simulation studies are used to investigate the Lasso and adaptive Lasso estimation methods and compare their relative performance with the ridge-type shrinkage estimators. Our proposed methodology is evaluated through application to a real genetic and brain connectivity edge weight data.

In Chapter 5, we apply statistical methods to analyze brain imaging and genetics data within the context of Alzheimer's disease for an imaging genetics study. This development is to understand the effects of genetic variations on brain function or structure and their impact on behavior and disease phenotypes. Our study is motivated by an imaging genetics study of the Alzheimer's Disease Neuroimaging Initiative (ADNI), where the objective is to examine the association between effective brain connectivity in the default mode network (DMN) and a set of single nucleotide polymorphism (SNPs) candidate genes. Alzheimer's Disease (AD) is a disease-related loss of memory and other cognitive abilities of sufficient severity to interfere with activities of daily living (Reitz et al., 2011).

There have been a number of applications of imaging genetics in understanding Alzheimer's disease (see, e.g., O'Donovan et al., 2008; O'Donovan et al., 2009; Stein et al., 2010; Hibar et al., 2011; Ge et al., 2012; Zhu et al., 2014; Greenlaw et al., 2017; Szefer et al., 2017; Lu et al., 2017; Song et al., 2019). Bookheimer et al. (2000) conducted an MRI-based neuroimaging genetics study that involves a test of the effect of variation within the apolipoprotein E (APOE) gene, associated with risk for Alzheimer's disease, on brain activity during memory tasks in healthy older adults

with intact cognition. They showed that carriers of the risk-associated $\epsilon 4$ allele had greater activity in regions affected by Alzheimer's disease, including hippocampal, parietal, and prefrontal regions during a memory task. Meda et al. (2012) used quantitative intermediate phenotypes derived from magnetic resonance imaging data from the ADNI database to test for association with gene-gene interactions within 212 known biological pathways. They tested approximately 151 million SNP-SNP interactions for association with 12-month regional atrophy rates using linear regression, with sex, APOE $\epsilon 4$ carrier status, age, education, and clinical status as covariates. Thompson et al. (2013) provide a comprehensive review of recent statistical approaches for the analysis of genetic data and brain connectivity focusing on both diffusion tensor imaging (DTI) and fMRI data. More recently, Zhang et al. (2018, 2019) discuss methods for the analysis of brain connectivity with an emphasis on structural connectomes.

Our analyses involve examining effective connectivity networks from rs-fMRI data within the default mode network (DMN). We consider networks comprised of four (DMN4 - 16 connections) and subsequently six (DMN6 - 36 connections) core regions of the DMN. The specific choice of regions which represent the nodes of the network is motivated by existing literature examining connectivity in the DMN (see, e.g., Wu et al., 2011; Sharaev et al., 2016; Xu et al., 2017; Glahn et al., 2010; Xu et al., 2017). Glahn et al. (2010b) estimated the importance of genetic effects on the default-mode network by examining covariation patterns in functional connectivity. The heritability for the default-mode functional connectivity was 42%. Although, neuroanatomical variation in this network was also heritable, the genetic factors that influence default-mode functional connectivity and grey-matter density seem to be distinct, suggesting that unique genes influence the structure and function of the network. Establishing the heritability of default-mode functional connectivity would authorize the use of resting-state networks as intermediate phenotype and establishing the heritability of effective connectivity for networks based on these regions.

The methodological approach for the analysis is designed in two stages where we view both disease and rs-fMRI as measures of the brain and our goal is to relate these measures of the brain to genetics. The first stage involves using an out-of-sample genome-wide scan of disease to select a subset of SNPs and this serves as a constraint on the SNPs that we relate to effective brain connectivity in subsequent analysis. The second stage is by relating effective brain connectivity as characterized through spectral DCM to the empirical disease-constrained subset of genetic variables using

longitudinal analyses based on both linear mixed effect (LME) models and function-on-scalar (FSR) regression. We apply the analysis to both DMN6 and DMN4. In addition, we implement a parametric bootstrap for testing SNP coefficients and make comparisons with p-values obtained from asymptotic null distributions.

1.2 Contributions

This thesis makes four contributions to the development of statistical methodology for the analysis of imaging data, imaging genetics and sparse estimation in linear mixed model. Chapters 2, 3, 4 and 5 of this thesis each correspond to one paper. The final chapter concludes with a discussion of future work. As of this writing, Chapters 2, 3 and 5 has been published and Chapter 4 paper has been accepted for publication. This is summarized as follows:

1. Opoku, E. A., Ahmed, S. E., Nelson, T., & Nathoo, F. S. (2020, July). Parameter and Mixture Component Estimation in Spatial Hidden Markov Models: A Comparative Analysis of Computational Methods. In International Conference on Management Science and Engineering Management (pp. 340-355).

2. Opoku, E. A., Ahmed, S. E., Song, Y., & Nathoo, F. S. (2021). Ant Colony System Optimization for Spatiotemporal Modelling of Combined EEG and MEG Data. Entropy, 23(3), 329.

3. Opoku, E. A., Ahmed, S. E., & Nathoo, F. S. (2021). Sparse Estimation Strategies in Linear Mixed Effect Models for High-Dimensional Data Application. Accepted for publication in Entropy Journal.

4. Nie, Y., Opoku, E., Yasmin, L., Song, Y., Wang, J., Wu, S., ... & Nathoo, F. S. (2020). Spectral dynamic causal modelling of resting-state fMRI: an exploratory study relating effective brain connectivity in the default mode network to genetics. Statistical Applications in Genetics and Molecular Biology.

Chapter 2

Parameter and Mixture Component Estimation in Spatial Hidden Markov Models

2.1 Introduction

The investigation of optimization algorithms for spatial hidden Markov models (HMMs) for the pixel/voxel-labeling problem is an important area of research and has many applications including those in image analysis where a Potts model (Wu, 1982) is often used as a prior for the latent states of an image (see, e.g., Ayasso and Ali, 2010; Johnson et al., 2013; Song et al., 2019). We consider here the setting where the data constitute a 2-dimensional image with each pixel represented by a continuous measurement and where the goal is to classify each pixel into one of a finite number of states. A discussion of various algorithms and applications for dealing with such problems can be found in Destrempes et al. (2005), Szeliski et al. (2008) and Ouadfel and Batouche (2003). A novel aspect of our study is that we consider the case where the number of latent states is unknown and we develop an estimator based on automatic pruning of redundant mixture components.

The iterated conditional modes algorithm (ICM, Besag, 1986) and the expectation maximization (EM) algorithm (Aleksandar and Zhang, 2006) will converge to a local optimum and they are the most widely used techniques for parameter estimation within this setting. ICM is a deterministic algorithm that iteratively maximizes the full conditional densities of model parameters and latent variables. While com-

putationally efficient and widely used, ICM and EM can be very sensitive to initial values. The EM algorithm is used for simultaneous Markov random field (MRF) image segmentation and parameter estimation in Destrempes et al. (2005).

A number of global optimization methods have been investigated including the simulated annealing algorithm (SA, Ingber, 1993) and the genetic algorithm (GA, Kim et al., 2000). Salzenstein and Pieczynski (1997) considered a Bayesian framework where they propose a general method of estimation applicable to the case of hidden data called the Iterative Conditional Estimation (ICE) algorithm. Their approach combines ICE with unsupervised fuzzy Bayesian image segmentation using hidden fuzzy Markov fields. Maniezzo and Carbonaro (2002) propose a clonal selection algorithm (CSA) and the use of Markov Chain Monte Carlo (MCMC) for HMM estimation for applications to brain magnetic resonance (MR) image segmentation. Their proposed approach employs a three-step iterative process that consists of MCMC-based class label estimation, bias field correction and CSA-based model parameter estimation.

Moving beyond point estimation, Green and Richardson (2002) develop a reversible jump MCMC algorithm for fully Bayesian inference for a mixture of Poisson distributions with the Potts model used as a prior for mixture allocations and with an unknown number of latent states in the model handled using reversible jump MCMC. Kato and Pong (2006) develop a Bayesian framework that employs a novel stochastic search algorithm for computing estimates of the hidden MRF model parameters that also incorporates the EM algorithm for maximizing the posterior density. This procedure is applied to the estimation of HMMs for images where each pixel has an associated multivariate observation with the likelihood based on multivariate distributions. Zhang and Seitz (2007) propose MRF-based stereo algorithms that apply an iterative algorithm for MAP estimation.

To overcome drawbacks associated with local algorithms such as ICM and EM, one alternative is the simulated annealing algorithm (Ingber, 1993). SA is a stochastic algorithm for combinatorial optimization that applies Monte Carlo sampling at each iteration to a modified objective function that corresponds to the original objective raised to the power $1/T_j$ at successively decreasing values of the temperature T_j. The algorithm is based on initially setting the temperature to relatively high values leading to easier movement across the parameter space and then decreasing according to a specified cooling schedule until the temperature is close to zero.

The SA algorithm can use either the Metropolis algorithm or Gibbs sampling and

the two sampling procedures when used within SA have been shown to be asymptotically equivalent in some settings (see, Li, 2009, Chapter 10) but in general Gibbs-SA is different from the standard SA algorithm that uses the Metropolis algorithm. For the Potts-Gaussian mixture model Gibbs-SA is extremely convenient computationally when combined with a chequerboard updating scheme and we will thus focus on the Gibbs sampling variant of SA in the rest of the paper.

While SA can theoretically reach a global optimum for some theoretically chosen cooling schedules (Shen et al., 2007), this algorithm is not a panacea as the choice of an optimal cooling schedule can be difficult in practice with parameter spaces of high-dimension. This is the case for the spatial mixture model considered here. Theoretically optimal cooling schedules are too slow to be of practical use and can be difficult to approximate so more practical faster empirically driven schedules must be used instead (Geman and Geman, 1993). Thus even when well-tuned empirically the algorithm may fail to find a global optimum with very complex objective functions such as the log-posterior density of a spatial hidden Markov model incorporating the Potts model.

The primary focus of our work lies with the Ant Colony System (ACS) optimization algorithm, a search algorithm based on the behaviour of real ants searching for food (see, e.g., Dorigo and Gambardella, 1997; Ouadfel and Batouche, 2003) and its application to spatial hidden Markov models when the number of mixture components is either known or unknown. ACS is a population-based approach based on a group of ants each constructing solutions (pixel labellings and parameter estimates) using pheromone information accumulated by the entire group of ants. Each ant is guided by a common function representing the distribution of pheromone which serves as a mechanism for the ants to communicate with each other regarding the quality of their estimates. In the variant of ACS considered here, we also incorporate the ICM algorithm within each iteration of ACS to conduct a local search that serves to improve the quality of the solutions found as well as estimate the mixture model parameters.

We study the performance of ACS within the context of a Gaussian spatial mixture model (GMM) for continuous data over pixels in a 2-dimensional grid. The GMM incorporates a labelling process allocating each pixel to one of K latent states. The labelling process is assumed to follow a Potts model which allows for spatial dependence among neighbouring pixels with a hyperparameter, known as the inverse temperature, that controls the degree of spatial dependence. In addition to the ACS

algorithm, we also consider the estimates obtained from Gibbs-SA and ICM and make comparisons.

Yang et al. (2012) propose a spatiotemporal model of land use change based on ant colony optimization using Markov chains and cellular automata. An important contribution related to our work is Emdadi et al. (2019) who develop an ant colony optimization algorithm 'AntMarkov' for temporal hidden Markov models and compare their proposed algorithm with the Baum-Welch algorithm, Viterbi-Training, the structured nonnegative matrix factorization algorithm and the Tabu-Search method. An important distinction between their work and our work is our consideration of a spatial rather than temporal hidden Markov model. Wang and Ju (2011) develop a hybrid ant colony optimization and Baum-Welch algorithm for estimation in temporal hidden Markov models. Sagayam and Hemanth (2018) develop an artificial bee colony (ABC) algorithm for estimation of a 1-dimensional hidden Markov model developed for hand gesture recognition applications. Bouzbita et al. (2019) consider the ACS algorithm for a temporal hidden Markov model classifier and focus on dynamic adaptation of the algorithm's tuning parameters.

Closely related to our work is the work of Ouadfel and Batouche (2003) who consider ant colony optimization for image segmentation based on a Markov random field. There are several key differences between their work and ours. First, we consider the case where the number of mixture components is unknown and one has the goal of estimating the number of latent states. Second, we consider estimation of pixel labels and mixture model parameters which leads to an ACS-ICM algorithm; whereas, Ouadfel and Batouche (2003) only describe algorithms for the pixel labelling (image segmentation in their case) problem via ACS. Third, we carefully consider the tuning of both the ACS-ICM and Gibbs-SA algorithms carefully and adopt formal procedures for optimizing over tuning parameters. Fourth, we examine differences in the relative performance of ACS-ICM and competing algorithms under different levels of spatial dependence in the true scene.

2.2 Statistical Model

Let $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ be a vector of continuous values representing a 2-dimensional image with n pixels and with y_i being the value of the image at the i^{th} pixel. We assume that y_i follows a Gaussian-Potts mixture model so that at the first level the

data are distributed as

$$y_i|\mathbf{Z},\boldsymbol{\mu},\boldsymbol{\sigma}^2 \overset{ind}{\sim} \prod_{\ell=1}^{K} \mathcal{N}(\mu_\ell,\sigma_\ell^2)^{I(Z_{i\ell}=1)}. \tag{2.1}$$

Pixels are assigned to mixture components through a labelling process $\mathbf{Z} = (\mathbf{Z}_1', \mathbf{Z}_2', \ldots, \mathbf{Z}_n')'$ where $\mathbf{Z_i}$ is a vector of K binary variables indicating the mixture component to which observation y_i has been assigned with $\sum_{l=1}^{K} Z_{i\ell} = 1$. We assume that the allocation process follows a Potts model having the following probability mass function:

$$P(\boldsymbol{Z}|\beta) = \frac{\exp\{\beta \sum_{h\sim j} \delta(\boldsymbol{Z_j}, \boldsymbol{Z_h})\}}{G(\beta)}, \quad \delta(\boldsymbol{Z_j}, \boldsymbol{Z_h}) = 2\boldsymbol{Z_j'Z_h} - 1,$$

where $G(\beta)$ is the normalizing constant for this probability mass function, $\beta \geq 0$ is a hyper-parameter, known as the inverse temperature parameter, which governs the degree of spatial dependence, and $i \sim j$ indicates that pixel i and j are neighbors. In what follows we assume a first-order neighborhood structure over a 2-dimensional regular grid of pixels.

The model parameters are the component specific means and variances, $\boldsymbol{\mu} = (\mu_1, \mu_2, ..., \mu_K)'$ and $\boldsymbol{\sigma^2} = (\sigma_1^2, \sigma_2^2, ..., \sigma_K^2)'$. Priors completing the model specification are assigned as follows:

$$\sigma_\ell^2 \overset{iid}{\sim} \text{Inverse-Gamma}(a_{\sigma^2}, b_{\sigma^2}),\ \ell = 1, 2, ...K,$$

$$\mu_\ell \overset{iid}{\sim} \mathcal{N}(v_\mu, w_\mu), \ell = 1, ..., K.$$

For the current study we assume β is known. In practice its value can be varied as part of a sensitivity analysis. The model unknowns are thus:

$$\boldsymbol{\Theta} = \{\boldsymbol{Z}, \{\mu_1, \mu_2, ..., \mu_K\}, \{\sigma_1^2, \sigma_2^2, ..., \sigma_K^2\}\}.$$

The number of mixture components is estimated using automatic pruning of mixture components in the optimization algorithm. Assuming the value of K used in the algorithms is larger than the true number of mixture components we obtain the estimated pixel labelings $\hat{\boldsymbol{Z}}$ and then count the number of non-empty mixture components to obtain an estimate of the number of latent states $\hat{K} = \sum_{j=1}^{K} I\{\sum_{i=1}^{n} \hat{Z}_{ij} > 0\}$. After estimation the $\hat{K}$ non-empty mixture components are re-ordered according to the estimated component means in decreasing order.

2.3 Estimation

The MAP estimate is obtained as

$$\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmax}} \log P\big(\boldsymbol{\Theta}|\mathbf{y}\big) = \underset{\mathbf{Z},\boldsymbol{\mu},\boldsymbol{\sigma^2}}{\operatorname{argmax}} \log \mathrm{P}(\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \mathbf{y})$$

where $\mathrm{P}(\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \mathbf{y})$ is the joint probability of $\boldsymbol{\Theta}$ and the data $\mathbf{y}$ and takes the form

$$\begin{aligned} P(\boldsymbol{\Theta}, \mathbf{y}) &= P(\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \mathbf{y}) \\ &= P(\mathbf{y}|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) \times P(\mathbf{Z}|\beta) \times P(\boldsymbol{\mu}) \times P(\boldsymbol{\sigma}^2) \\ &= [\prod_{i=1}^{n} P(y_i|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2)] \times \mathrm{Potts}(\mathbf{Z}|\beta) \times P(\boldsymbol{\mu}) \times P(\boldsymbol{\sigma}^2) \\ &\approx [\prod_{i=1}^{n} P(y_i|\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2)] \times \mathrm{PL}(\mathbf{Z}|\beta) \times \prod_{\ell=1}^{K} P(\mu_\ell) P(\sigma_\ell^2) \end{aligned}$$

$$= \prod_{i=1}^{n} \left[\prod_{\ell=1}^{K} N(y_i; \mu_\ell, \sigma_\ell^2)^{I(Z_{i\ell}=1)} \right] \times \left[P(\mathbf{Z}_i|\mathbf{Z}_{(-i)}, \beta) \right] \tag{2.2}$$

$$\times \prod_{\ell=1}^{K} IG(\sigma_\ell^2; a_{\sigma^2}, b_{\sigma^2}) \times \prod_{\ell=1}^{K} N(\mu_\ell; w_\mu, v_\mu) \tag{2.3}$$

Where $\mathrm{Potts}(\mathbf{Z}; \beta)$ denotes the joint probability mass function of the Potts model with inverse temperature parameter β evaluated at $\mathbf{Z}$ and $\mathrm{PL}(\mathbf{Z}|\beta)$ is the corresponding pseudolikelihood approximation (Besag, 1977) that we use moving forward; $N(\mathbf{y}; \mu, \sigma^2)$ denotes the density of the normal distribution with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\sigma}^2$ evaluated at $\mathbf{y}$; $IG(\sigma_\ell^2; a_{\sigma^2}, b_{\sigma^2})$ denotes the density of the inverse-gamma distribution with parameters a_{σ^2} and b_{σ^2} evaluated at σ_ℓ^2. The pseudolikelihood approximation is used to avoid the difficult computation of the normalizing constant $G(\beta)$ associated with the Potts model. Equations (2.2) and (2.3) together comprise the objective function that we wish to maximize over $\boldsymbol{\Theta} = \{\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2\}$.

2.3.1 Iterated Conditional Modes

The iterated conditional modes (ICM) algorithm (Besag,1986) is well known and well studied. It proceeds by iteratively maximizing full conditional distributions of the

model parameters and latent variables. For the spatial model under consideration, the labelling process variables $\mathbf{Z}$ are updated using an efficient chequerboard updating scheme (see Besag,1986 and Song et al., 2019). Within this scheme $\mathbf{Z}$ is partitioned into two blocks $\mathbf{Z} = \{\mathbf{Z}_W, \mathbf{Z}_B\}$ according to a 2-dimensional chequerboard arrangement, where $\mathbf{Z}_W$ corresponds to the 'white' pixels and $\mathbf{Z}_B$ corresponds to the 'black' pixels. Under a Markov random field with first-order neighbourhood structure the elements of $\mathbf{Z}_W$ are conditionally independent given $\mathbf{Z}_B$, and vice versa. This allows for simultaneous updating of all elements of $\mathbf{Z}_W$ followed by simultaneous updating of all elements of $\mathbf{Z}_B$ and this can be made relatively fast using multiple cores. Convergence of the algorithm is based on monitoring the relative change in the objective function. The ICM algorithm for the spatial hidden Markov model is presented in Algorithm 1.

2.3.2 Annealed Gibbs Sampling

Simulated annealing (SA), introduced by Metropolis et al. (1953) and by Kirkpatrick et al. (1983), is a stochastic algorithm for combinatorial optimization (see also, Robert and Casella, 2013). The algorithm is inspired by the annealing process in which materials are raised to higher energy levels and then cooled in search of an optimal low energy configuration. This notion is implemented through the introduction of a temperature parameter that is used to provide a rescaling of the objective function that may allow the algorithm to avoid local optima.

At higher values of the temperature parameter the rescaled objective function has a flatter more uniform shape which makes it easy for the algorithm to move around the parameter space and explore the values of the function. As the temperature parameter moves towards zero the rescaled function becomes increasingly concentrated around small neighbourhoods of its local optima. We work with a variant of SA that applies the Gibbs sampler at each iteration, which we call Gibbs-SA, successively at decreasing values of the temperature parameter.

The algorithm is run in stages and each stage consists of a sequence of Gibbs sampling iterations with the target distribution being proportional to $P(\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \mathbf{y})^{\frac{1}{T_j}}$, where T_j is the temperature for stage j. Initially, this temperature is set to a relatively high value which encourages the Gibbs draws to move around the parameter space. The value of T_j is decreased according to a carefully chosen schedule and as the temperature approaches zero the system freezes near an optimum of the objective

function. The sequence of T_j values is known as the cooling schedule and in what follows we create this sequence based on the rule $T_{\text{new}} = T_{\text{old}} * k$ for $k \in (0, 1)$ and where k is a tuning parameter. The tuning parameters for the algorithm are thus k and the initial temperature T_0 and these are chosen using an outer level optimization carried out over the tuning parameters. We optimize over the tuning parameters (T_o, k) using the Nelder-Mead optimization algorithm (Singer and Nelder, 2009) which is implemented by using the 'optim' function in the R programming language (R Development Core Team, 2017).

2.3.3 Ant Colony System - Iterated Conditional Modes

Ant Colony System (ACS) optimization is a population based metaheuristic method that can be used to find approximate solutions to combinatorial optimization problems. This algorithm was introduced by Dorigo and Gambardella (1997) as an approach to the travelling salesmen problem and is inspired by the behaviour of an ant colony based on how ants attempt to find an optimal path to a food source.

The ACS algorithm is based on a set of agents each representing an ant searching for solutions to the problem of maximizing the posterior density. A key idea is the leaving of a marker representing a pheromone trail quantifying the quality of a path found by a given agent. The process of constructing a solution is stochastic and is biased by this pheromone trail which is represented by a function in the algorithm. Ants modify the pheromone trail (and thus the function that represents it) when a particular path or estimate is chosen and this represents information available to the other ants that are also searching for solutions. It is through the pheromone trail that the ants communicate information about the solutions chosen and the objective function. Ants follow certain paths probabilistically and the probability of a given path depends on the current value of the pheromone trail. As more ants find the same path it becomes reinforced through corresponding changes to the pheromone trail. The ants incrementally find an optimal solution through the evolving pheromone function.

Our proposed algorithm for the spatial hidden Markov model combines ACS with ICM where the model parameters are assumed fixed and the mixture allocations variables are updated using ACS with a set number of ants and then the ICM algorithm is used for both updating the model parameters and also for a local search over the mixture allocation variables. Thus the update steps corresponding to ACS are com-

bined with running ICM to convergence at each iteration. The combination of ACS for updating the pixel labels with ICM for updating the mixture model parameters is a key difference between our algorithm and that described in Ouadfel and Batouche (2003).

The algorithm has four tuning parameters. The first denoted $q_o \in (0, 1)$ controls the degree of stochasticity with larger values corresponding to less stochasticity and thus less random exploration of the parameter space. When a solution is chosen another tuning parameter τ_0 controls the amount of pheromone reinforcing this solution in the information available to the other ants. A third tuning parameter ρ controls the evaporation of pheromone and finally a fourth tuning parameter N_{ants} controls the number of ants. The number of ants used for this study is 20 ($N_{ants} = 20$). As with Gibbs-SA the remaining tuning parameters (q_o, τ_0, ρ) for ACS-ICM are chosen using an outer level optimization using the Nelder-Mead algorithm.

2.4 Simulation Studies

The algorithms I-III described in the previous section are compared using two simulation studies. In the first study we fix the number of mixture components at the true value (chosen as $K_{true} = 3$) and focus on parameter estimation and pixel labelling while in the second study the algorithms are run with $K = 10$ and the data are generated based on $K_{true} = 3$ and we compare the sampling distributions of $\hat{K}_{ICM}$, $\hat{K}_{SA}$ and $\hat{K}_{ACS}$. The simulated data are based on a 100 (10×10) pixel image. In each study we run three sets of simulations each based on 100 simulation replicates with the data simulated from the Gaussian Potts mixture model with three mixture components. The level of spatial dependence in the underlying image varies across the three sets with the true pixel labels $\mathbf{Z}$ drawn from the Potts model (a single draw is used for each of the three sets) by varying the inverse-temperature parameter as $\beta = 0$, $\beta = 0.2$ and $\beta = 1.1$. The same datasets are used for each of the two studies with the difference being that K is fixed at the true value ($K_{true} = 3$)) in study I and $K = 10$ in study II where we focus on the sampling distribution of $\hat{K}$. Given the true labels, each data replicate is simulated from the Gaussian mixture components with $\mu_1 = 5$, $\mu_2 = 7$ and $\mu_3 = 9$ and $\sigma_1^2 = 1$, $\sigma_2^2 = 2$ and $\sigma_3^2 = 3$.

For each dataset we fit the model using ICM, Gibbs-SA and ACS-ICM. To ensure a fair comparison, the same starting values are used to initialize all of the algorithms and all algorithms are run to convergence of the objective function. The pixel labelings

Algorithm 1 - ICM algorithm

1. $\Theta \leftarrow$ Set Initial Value
2. Converged $\leftarrow 0$
3. While Converged $= 0$ do
4. for l $= 1, \ldots .$ K do
   $\sigma_l^2 \leftarrow \left[\frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1)(y_i-\mu_l)^2}{2} + b_{\sigma^2}\right] / \left[\frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1)}{2} + a_{\sigma^2} + 1\right]$
   end for
5. for l $= 1, \ldots .$ K do
   $\mu_l \leftarrow \left[\frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1) y_i}{\sigma_l^2} + \frac{w_\mu}{v_\mu}\right] D^2$, where $D^2 = \left[\frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1)}{\sigma_l^2} + \frac{1}{v_\mu}\right]^{-1}$

   end for
6. Let $\mathbb{B}$ denote the indices for 'black' voxels and $\mathbb{W}$ denote the indices for 'white' voxels.
7. For $\kappa \in \mathbb{B}$ *simultaneously*

   $Z_{\kappa q} \leftarrow 1$ and $Z_{\kappa l} \leftarrow 0, \ \forall l \neq q$
   where $q = \text{argmax}_{h \in \{1,\ldots,K\}} P(h)$, and

   $$P(h) = \frac{\sigma_h^{-1} \times \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(y_i-\mu_h)^2 + 4\beta \sum_{v \in \delta_\kappa} Z_{vh}\right)}{\sum_{l=1}^{K} \sigma_l^{-1} \times \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(y_i-\mu_l)^2 + 4\beta \sum_{v \in \delta_\kappa} Z_{vl}\right)}$$

   where δ_κ contains the indices for the neighbours of pixel κ.

   end for
8. For $\kappa \in \mathbb{W}$ *simultaneously*

   $Z_{\kappa q} \leftarrow 1$ and $Z_{\kappa l} \leftarrow 0, \ \forall l \neq q$
   where $q = \text{argmax}_{h \in \{1,\ldots,K\}} P(h)$, and

   $$P(h) = \frac{\sigma_h^{-1} \times \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(y_i-\mu_h)^2 + 4\beta \sum_{v \in \delta_\kappa} Z_{vh}\right)}{\sum_{l=1}^{K} \sigma_l^{-1} \times \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(y_i-\mu_l)^2 + 4\beta \sum_{v \in \delta_\kappa} Z_{vl}\right)}$$

   end for
9. check for convergence; set Converged $= 1$ if converged.

   end while

Algorithm 2 Gibbs Simulated Annealing

1. Set tuning parameters: Initial temperature T_0 and scaling constant k

2. $\Theta \leftarrow$ Initial Value

3. At iteration j, run N iterations of the Gibbs sampler with target $\propto P(\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \mathbf{y})^{\frac{1}{T_j}}$

   Gibbs sampling:
   Begin for l=1, ... K do

   $\sigma_l^2 \sim$ Inverse-Gamma$(a_l^\star, b_l^\star)$
   where $a_l^\star = \frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1)}{2T_j} + \frac{a_{\sigma 2}+1}{T_j} - 1$ and $b_l^\star = \frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1)(y_i-\mu_l)^2}{2} + \frac{b_{\sigma 2}}{T_j}$

   End for

   Begin for l=1, ... K do

   $\mu_l \sim N(m_l, D_l^2)$
   $m_l = \left[\frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1) y_i}{T_j \sigma_l^2} + \frac{w_\mu}{T_j \times v_\mu}\right] D_l^2$, and $D_l^2 = \left[\frac{\sum_{i=1}^{n} \boldsymbol{I}(Z_{il}=1)}{T_j \times \sigma_l^2} + \frac{1}{T_j \times v_\mu}\right]^{-1}$
   End for

   Use the chequerboard scheme to sample $\mathbf{Z}$ based on the pixel-specific full conditional distributions:

   $\mathbf{Z_i} \sim$ multinomial$(1, prob = (P[Z_{i1}], \ldots, P[Z_{iK}])'), i = 1, \ldots, n$

   $$P[Z_{ih}] = P(Z_{ih} = 1) = \frac{\sigma_h^{-1/T_j} \times \exp\left(-\frac{1}{2T_j\sigma_h^2}\sum_{i=1}^{n}(y_i-\mu_h)^2 + \frac{4\beta}{T_j}\sum_{v\in\delta_\kappa} Z_{vh}\right)}{\sum_{l=1}^{K} \sigma_l^{-1} \times \exp\left(-\frac{1}{2T_j\sigma_h^2}\sum_{i=1}^{n}(y_i-\mu_l)^2 + \frac{4\beta}{T_j}\sum_{v\in\delta_\kappa} Z_{vl}\right)}$$

   where δ_κ contains the indices for the neighbours of pixel κ. Set Θ based on the values of $\mathbf{Z}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2$ obtained from the final iteration of Gibbs sampling.

4. Until $T_j \approx 0$. Update $T_j = kT_{j-1}$ and go back to step 3.
   Return final values of σ_l^2, μ_l and Z_v

Algorithm 3 - ACS-ICM Algorithm

1: $\Theta \leftarrow$ Initial Value; set tuning parameters τ_o, q_o, ρ and N_{ants}.

2: Initialize pheromone information $\tau(i,\ell) = \tau_o$, for each $(i,\ell) \in \{1,\dots n\} \times \{1,\dots K\}$ representing information gathered by ants about the pixel labelling.

3: Construct candidate solutions for each of N_{ants} ants. For ant j, we find a candidate pixel labelling $\mathbf{Z}^{(j)} = (\mathbf{Z}_1'^{(j)}, \mathbf{Z}_2'^{(j)}, \dots, \mathbf{Z}_n'^{(j)})'$.

- Construct candidate by assigning label l to pixel s using the rule:

$$\ell = \begin{cases} \arg\max_u \tau(s,u) & \text{if } q \le q_o \\ \text{RANDOM} & \text{if } q > q_o \end{cases}$$

  where if $q > q_o$ the label for pixel s is drawn randomly from $\{1,\dots,K\}$ with probability

$$p(s,\ell) = \frac{\tau(s,\ell)}{\sum_{u\in\Lambda}\tau(s,u)},$$

  and where $q \sim$ uniform$[0,1]$.

- Assuming pixel s is assigned label l set:

$$\tau(s,\ell) = (1-\rho)\tau(s,\ell) + \rho\tau_o$$

  and for all $k \neq l$ set:

$$\tau(s,k) = (1-\rho)\tau(s,k)$$

  where ρ is a tuning parameter in (0,1), which represents evaporation of the pheromone trail and $\tau_o > 0$.

4: Run ICM to convergence on the solutions obtained from all ants while also updating mixture component parameters $\boldsymbol{\mu}, \boldsymbol{\sigma}^2$.

5: For all N_{ants} solutions, evaluate the quality of each ant's solution using objective function: OBJ$(\mathbf{Z}^{(j)}, \boldsymbol{\mu}^{(j)}, \boldsymbol{\sigma}^{2(j)})$. Keep track of the best value. The current solution for each ant serves as the starting value for the next iteration.

6: Apply a global updating of the pheromone function. For the *best ever* solution (s,ℓ) update the function as:

$$\tau(s,\ell) = (1-\rho)\tau(s,\ell) + \rho\tau_o$$

and for all $k \neq l$ set:

$$\tau(s,k) = (1-\rho)\tau(s,k).$$

Check for convergence. Go back to step 3.

7: Return pixel labelling $\mathbf{Z}$ and model parameters $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ from the best ever solution.

are randomly initialized independently with equal probability for each class. Given the random labelings, the mixture component means are taken as the average of the data values for those pixels assigned to a given component and the mixture component variances are taken as the corresponding sample variances. For pixels in which the random labelings result in a mixture component having no pixels assigned to it, the prior mode is used as the initial value for the component model parameters.

To compare the performance of the algorithms, we first store the optimized objective function value obtained from each of the three algorithms for each of the 300 datasets (100 for each level of spatial dependence). In addition to the objective function values we compute for each pixel, the proportion of simulation replicates in which the estimated label is equal to the true label. We then examine the distribution of these proportions across the image and this is displayed in Figure 2.1 A clear pattern emerges showing that ACS-ICM gives the most accurate pixel labelings, followed by Gibbs-SA and then ICM and this ordering is constant across levels of spatial dependence.

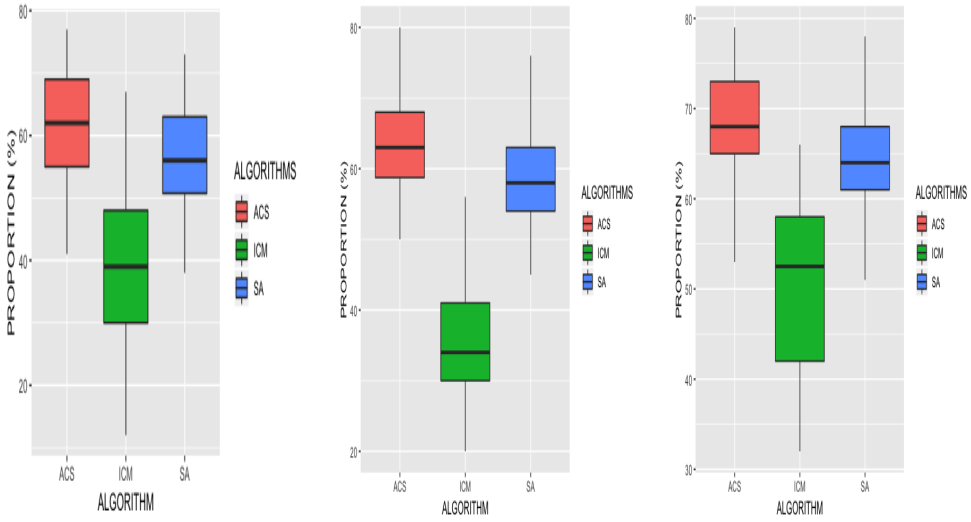


Figure 2.1: Distribution (across 100 simulation replicates) of the proportion of correctly labelled pixels with $\beta = 0$ (left), $\beta = 0.2$ (center) and $\beta = 1.1$ (right).

Figure 2.2 shows three pairs plots comparing the final objective function values obtained from each of the algorithms for the three different levels of spatial dependence. Again, a clear pattern emerges showing that ACS-ICM yields the highest objective function values in almost all cases, followed by Gibbs-SA and then followed by ICM. It is clear that ICM underperforms substantially compared to ACS-ICM

and Gibbs-SA as expected and it is also clear that ACS-ICM has the best overall performance.

For mixture component estimation, Table 2.1 compares the bias and mean-squared-error of $\hat{K}_{ICM}$, $\hat{K}_{SA}$ and $\hat{K}_{ACS}$ for the three different levels of spatial dependence. We observe that the estimator of the number of mixture components obtained from ACS-ICM exhibits the best performance in terms of both bias and MSE with $\text{Bias}(\hat{K}_{ACS}) < \text{Bias}(\hat{K}_{SA}) < \text{Bias}(\hat{K}_{ICM})$ and $\text{MSE}(\hat{K}_{ACS}) < \text{MSE}(\hat{K}_{SA}) < \text{MSE}(\hat{K}_{ICM})$ and that this ordering is seen at all three levels of spatial dependence. We also note that the bias is positive in all cases so that the number of mixture components is over-estimated.

With respect to computation time, with all algorithms programmed in the R software and run on on an Intel E5-2683 v4 Broadwell 2.1Ghz processsor with 16GB RAM the average (over simulation replicates) computation time to convergence for a single image is 15 minutes for the ICM algorithm, 50 minutes for the Gibbs-SA algorithm and 180 minutes for ACS-ICM.

Table 2.1: Bias and Mean Square Error (MSE) of estimated number of mixture components ($\hat{K}$) from the 100 simulation replicates with different spatial correlations when the algorithms are run with $K = 10$.

$\beta = 0$			
	ICM	SA	ACS-ICM
$\text{Bias}(\hat{K})$	3.13	2.03	1.64
$\text{MSE}(\hat{K})$	10.47	4.61	3.10
$\beta = 0.2$			
	ICM	SA	ACS-ICM
$\text{Bias}(\hat{K})$	2.86	1.89	1.42
$\text{MSE}(\hat{K})$	8.70	3.93	2.30
$\beta = 1.1$			
	ICM	SA	ACS-ICM
$\text{Bias}(\hat{K})$	2.69	1.76	1.32
$\text{MSE}(\hat{K})$	7.71	3.46	1.96

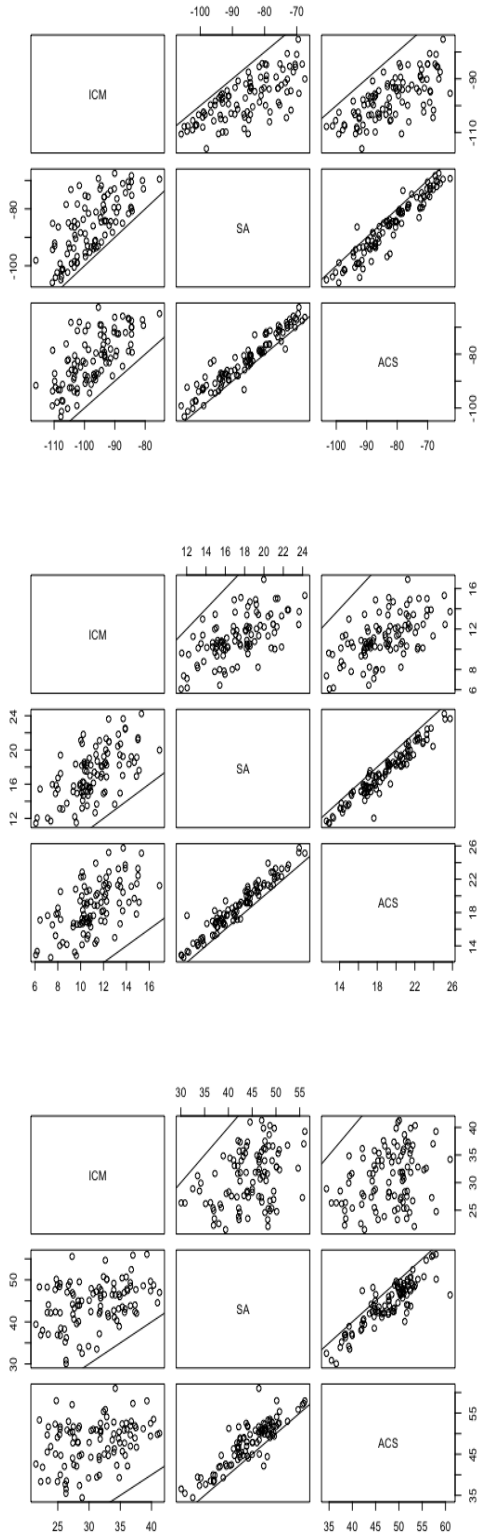


Figure 2.2: Comparison of objective function values obtained for each dataset and algorithm with $\beta = 0$ (top), $\beta = 0.2$ (middle) and $\beta = 1.1$ (bottom).

2.5 Discussion

We have studied the problem of computation for MAP estimation for the spatial hidden Markov model with both known and unknown number of mixture components. Our studies demonstrate a superior performance of the ACS-ICM algorithm when compared to Gibbs-SA and ICM algorithms in terms of objective function values, pixel labelling accuracy and mixture component estimation. This relative performance in the three algorithms appears constant across different levels of spatial dependence. Our most important result is the demonstration that the ACS-ICM outperforms Gibbs-SA when both algorithms are carefully tuned with an outer-level optimization that uses the Nelder-Mead algorithm to select tuning parameters. While theoretical results associated with simulated annealing allow for convergence to a global optimum for some tuning schedule, in practice this can be difficult to achieve even after careful tuning as demonstrated in our studies. ACS-ICM exhibits superior performance with an equal effort made in tuning both algorithms. Our results suggest that ACS-ICM is a useful and potentially powerful approach for MAP estimation with spatial mixture models and may be a preferred approach when the increased computation time is feasible.

We are currently extending our investigation to the development of an ACS-ICM algorithm for computing solutions to inverse problems where existing approaches have used the ICM algorithm in combination with spatial mixture models (Song et al., 2019). The algorithm may have considerable potential for application to other statistical problems involving combinatorial optimization such as MAP estimation with spike-and-slab variable selection (see, e.g., Rockova and George, 2014) and optimization for deep learning (LeCun et al., 2015). Along these lines Desell et al. (2015) develop an ant colony optimization algorithm for training recurrent neural networks. These and other statistical and experimental design problems are potentially fruitful areas for further study of ACS-ICM.

Chapter 3

Ant Colony System Optimization for Spatiotemporal Modelling of Combined EEG and MEG Data

3.1 Introduction

Electroencephalography (EEG) and Magnetoencephalography (MEG) are two non-invasive approaches for measuring electrical activity of the brain with high temporal resolution. These neuroimaging techniques allow us to study brain dynamics and the complex informational exchange processes in the human brain. They are widely used in many clinical and research applications (see, e.g., Chowdhury et al, 2015; Zhang et al., 2014), though estimating the evoked-response activity within the brain from electromagnetic fields measured outside of the skull remains a challenging inverse problem with infinitely many different sources within the brain that can produce the same observed data (Baillet et al., 2001).

Proposed solutions to the MEG/EEG inverse problem have been based on distributed source and dipolar methods (Sorrentino et al., 2017). In the case of distributed source methods, every location on a fine grid within the brain has associated neural activation source parameters. In this case, the number of unknown current sources exceeds the number of MEG or EEG sensors and estimation thus requires constraints through regularization or priors to obtain a solution. For such methods, various steps have been taken to regularize the solution by choosing minimum-norm solutions or by limiting the spatiotemporal variation of the solution. These approaches

impose L_2 or L_1 (Giraldo et al., 2017) norm regularization constraints that serve to stabilize and condition the source parameter estimates. However, these methods do not consider the temporal nature of the problem. Long et al. (2011) propose a dynamic state-space model that accounts for both spatial and temporal correlations within and across candidate intra-cortical sources using Bayesian estimation and Kalman filtering. Dipolar methods, on the other hand, assume that the actual current distribution can be explained by a small set of current dipoles with unknown locations, amplitudes and orientations (see Sorrentino et al., 2017 for review). Hence, the resulting inverse problem becomes non-linear and a number of dipoles is to be estimated. Proposed solutions to this problem include algorithms such as simulated annealing (Rytsar et al., 2010) to address nonlinear optimization in the localization of neuromagnetic sources.

From the perspective of Bayesian approaches, the ill-posed nature of the inverse problem requires incorporation of prior assumptions when choosing an appropriate solution out of an infinite set of candidates. For instance, Wipf and Nagarajan (2009), who consider Gaussian scale mixture models, with flexible, large covariance components representing spatial patterns of neural activity. Henson et al. (2009) propose a hierarchical linear model with Gaussian errors in a Parametric Empirical Bayes (PEB) framework whose random terms are drawn from multivariate Gaussian distributions and covariances factor into temporal and spatial components at the sensor and source levels. Friston et al. (2008) propose an application of empirical Bayes to the source reconstruction problem with automatic selection of multiple cortical sources. Daunizeau and Friston (2007) develop the Mesostate-Space Model (MSM) based on the assumption that the unknown neural brain activity can be specified in terms of a set of locally distributed and temporally coherent meso-sources for either MEG or EEG data, while Olier et al. (2013) extend this approach to propose a Switching Mesostate-Space Model (SMSM) to allow flexibility by accounting for complex brain processes that cannot be characterized by linear and stationary Gaussian dynamics.

By extending and building on the MSM, Song et al. (2019) develop a Bayesian spatial finite mixture model incorporating the following two conditions, taken directly from Song et al. (2019):

1. *relaxing the assumption of independent mixture allocation variables and modeling mixture allocations using the Potts model, which allows for spatial dependence in allocations.*

2. *formulate the model for combined MEG and EEG data for joint source localization.*

This spatiotemporal model describes a joint model that combines MEG and EEG data, in which brain neural activity is modeled from the Gaussian spatial mixture model. The neural source activity is described in terms of a few hidden states, with each state having its own dynamics and a Potts model used in representing the spatial dependence in the mixture model.

For the Bayesian mixture model formulated, an Iterated Conditional Modes (ICM) algorithm was developed by Song et al. (2019) for simultaneous point estimation and model selection for the number of mixture components in the latent process. Whilst ICM is a very simple and computationally efficient algorithm, convergence of this algorithm is sensitive to starting values and local optima. This issue was left unresolved in Song et al. (2019). Here we investigate the potential for finding better solutions, and focus on implementing a population-based optimization algorithm-based Ant Colony System (ACS) (Dorigo and Gambardella, 1997).

ACS is a metaheuristic optimization algorithm inspired by the biological behavior of ants constructing solutions based on their collective foraging behavior (Dorigo and Gambardella, 1997). ACS has been successfully applied in several areas such as clustering, data mining and image segmentation problems (see, e.g., Inkaya et al., 2015; Parpinelli et al., 2015; Sharma and Buddhiraju, 2018). ACS is a constructive algorithm that uses an analogue of ant trail pheromones to learn about good features of solutions in combinatorial optimization problems. New solutions are generated using a parameterized probabilistic model, the parameters of which are updated using previously generated solutions so as to direct the search towards promising areas of the solution space. The model used in ACS is known as pheromone, an artificial analogue of the chemical substance used by real ants to mark trails from the nest to food sources. Based on this representation, each artificial ant constructs a part of the solution based on concentration of pheromone information released by other ants. The amount of pheromone deposited by an ant reflects the quality of the good solutions built and the traversed path. The pheromone deposited and volatilized adds solution components to partial solutions. After some time and based on more ants' communications through pheromone information, they tend to follow the same optimal paths yielding the optimal solution, in our context maximization of the posterior density.

As an alternative to the ICM algorithm, we thus implement the ACS algorithm

coupled with a local search ICM algorithm to provide a new approach to model estimation and potentially better estimates of the model parameters. This approach is evaluated and found to provide significant improvements. Within the context of a simpler spatial mixture, ACS has been implemented for a Gaussian Potts mixture model in Opoku et al. (2020) and has been shown to outperform both the Simulated Annealing and ICM algorithms for parameter and mixture component estimation. The theoretical guarantees associated with simulated annealing to reach a global optimum is dependent on the choice of a cooling schedule. The choice of an optimal cooling schedule can be difficult in practice for large spatiotemporal models. ACS has also proved to be competitive with genetic and other optimization algorithms in several tasks, mainly in image classification and the traveling salesman problem (see, e.g., Ouadfel and Batouche, 2003; Shweta and Singh, 2013).

Ant Colony Optimization (ACO) algorithms are implemented to solve Constraint Satisfaction Problems (CSP) where ACO solutions to CSP face the challenge of high cost and low solution quality. Motivated by this challenge, Guan et al. (2019) propose Ant Colony Optimization based on information Entropy (ACOE). The idea is based on incorporating a local search that uses a crossover operation to optimize the best solution according to the feedback of information entropy. This is performed by comparing the difference of the information entropy between the current global best solution and the best solution in the current iteration. Datasets from four classes of binary CSP test cases were generated and then ACOE was implemented for comparison. Results showed that ACOE outperformed Particle Swarm Optimization (PSO), a Differential Evolution (DE) algorithm and Artificial Bee Colony (ABC) in terms of the solution quality, data distribution and convergence performance.

To our knowledge, this is the first attempt at solving the neuroelectromagnetic inverse problem for combined EEG/MEG data using a population-based optimization approach combined with a spatial mixture model. The primary contribution of this paper is the design and implementation of the ACS algorithm to the dynamic spatial model and its evaluation. Importantly, we demonstrate improved results both in the estimation of neural activity and model selection uniformly across all conditions considered.

3.2 Methods

This section describes the Bayesian spatial mixture model developed in Song et al. (2019), the design and implementation of both the ICM and ACS-ICM algorithm.

3.2.1 Model

We provide details and mathematical description of the joint model below. Let $\boldsymbol{M}(t) = (M_1(t), M_2(t), ..., M_{n_M}(t))'$ and $\boldsymbol{E}(t) = (E_1(t), E_2(t), ..., E_{n_E}(t))'$ denote the MEG and EEG, respectively, at time t, $t = 1, \ldots, T$; where n_M and n_E denote the number of MEG and EEG sensors, the model assumes:

$$\begin{aligned}
\boldsymbol{M}(t) &= \boldsymbol{X}_M \boldsymbol{S}(t) + \boldsymbol{\epsilon}_M(t), \quad \boldsymbol{\epsilon}_M(t)|\sigma_M^2 \overset{iid}{\sim} MVN(\boldsymbol{0}, \sigma_M^2 \boldsymbol{H}_M), \;\; t = 1, \ldots, T, \\
\boldsymbol{E}(t) &= \boldsymbol{X}_E \boldsymbol{S}(t) + \boldsymbol{\epsilon}_E(t), \quad \boldsymbol{\epsilon}_E(t)|\sigma_E^2 \overset{iid}{\sim} MVN(\boldsymbol{0}, \sigma_E^2 \boldsymbol{H}_E), \;\; t = 1, \ldots, T,
\end{aligned}$$

where $\boldsymbol{X}_M$ and $\boldsymbol{X}_E$ denote $n_M \times P$ and $n_E \times P$ forward operators, respectively computed based on Maxwell's equations under the quasi-static assumption (Sarvas, 1987) for EEG and MEG; $\boldsymbol{H}_M$ and $\boldsymbol{H}_E$ are known $n_M \times n_M$ and $n_E \times n_E$ matrices, respectively, which can be obtained from baseline data providing information on the covariance structure of EEG and MEG sensor noise; and $\boldsymbol{S}(t) = (S_1(t), \ldots S_P(t))'$ represents the magnitude and polarity of neural currents sources over a fine grid covering the cortical surface. In this case, P represents a large number of point sources of potential neural activity within the brain covering the cortical surface. It is assumed that the P cortical locations are embedded in a 3D regular grid composed of N_v voxels to allow efficient computational implementation. Given this grid of voxels, a mapping $v : \{1, ..., P\} \to \{1, ..., N_v\}$ is defined such that $v(j)$ is the index of the voxel containing the jth cortical location. We assume a latent Gaussian mixture with allocations at the level of voxels:

$$S_j(t)|\boldsymbol{\mu}(t), \boldsymbol{\alpha}, \boldsymbol{Z} \overset{ind}{\sim} \prod_{l=1}^{K} \mathcal{N}(\mu_l(t), \alpha_l)^{Z_{v(j)l}}, \tag{3.1}$$

$j = 1, \ldots, P, \;\; t = 1, \ldots, T$; where $\boldsymbol{Z} = (\boldsymbol{Z}_1', \boldsymbol{Z}_2', ..., \boldsymbol{Z}_{N_v}')'$ is a labeling process defined over the grid of voxels such that for each $v \in \{1, ..., N_v\}$, $\boldsymbol{Z}_v' = (Z_{v1}, Z_{v2}, ..., Z_{vK})$ with $Z_{vl} \in \{0, 1\}$ and $\sum_{l=1}^{K} Z_{vl} = 1$; $\boldsymbol{\mu}(t) = (\mu_1(t), \mu_2(t), ..., \mu_K(t))' = (\mu_1(t), \boldsymbol{\mu}^A(t)')'$, where $\boldsymbol{\mu}^A(t) = (\mu_2(t), ..., \mu_K(t))'$ denotes the mean of the "active" states over different

components of activity and $\mu_1(t) = 0$ for all t, so that the first component corresponds to an "inactive" state. The variability of the l^{th} mixture component about its mean $\mu_l(t)$ is represented by $\alpha_l, l = 1, \ldots, K$.

The labeling process assigns each voxel to a latent state and is assumed to follow a Potts model:

$$P(\boldsymbol{Z}|\beta) = \frac{\exp\{\beta \sum_{h \sim j} \delta(\boldsymbol{Z_j}, \boldsymbol{Z_h})\}}{G(\beta)}, \quad \delta(\boldsymbol{Z_j}, \boldsymbol{Z_h}) = 2\boldsymbol{Z_j'Z_h} - 1,$$

where $G(\beta)$ is the normalizing constant for this probability mass function, $\beta \geq 0$ is a hyper-parameter that governs the strength of spatial cohesion, and $i \sim j$ indicates that voxels i and j are neighbors, with a first-order neighborhood structure over the 3D regular grid. The mean temporal dynamics for active components is assumed to follow a first-order vector autoregressive process:

$$\boldsymbol{\mu}^A(t) = \boldsymbol{A\mu}^A(t-1) + \mathbf{a}(t), \;\; \mathbf{a}(t)|\sigma_a^2 \overset{i.i.d}{\sim} MVN(\mathbf{0}, \sigma_a^2\mathbf{I})$$

$t = 2, \ldots, T$, $\boldsymbol{\mu}^A(1) \sim MVN(\mathbf{0}, \sigma_{\mu_1}^2\boldsymbol{I})$, with $\sigma_{\mu_1}^2$ fixed and known, but σ_a^2 unknown and assigned an inverse-Gamma (a_a, b_a) hyper-prior. Although Song et al. (2019) adopted a pseudo-likelihood approximation to the normalizing constant of the Potts model and then assigned a uniform prior to the spatial parameter to control the degree of spatial correlation, we fixed the inverse temperature parameter and vary it as part of a sensitivity analysis.

For model selection, the number of mixture components, the value of K, in Equation (3.1) will not be known prior and so it is estimated simultaneously with model parameters. Thus this approach achieves simultaneous point estimation and model selection. We can obtain a simple estimate for the number of mixture components based on the estimated allocation variables $\hat{\boldsymbol{Z}}$ when the algorithm is run with a sufficiently large value of K. This is achieved by running the algorithm with a value of K that is larger than the expected number of mixture components. For example, the value of K can be set as $K = 15$ when running the algorithm. The j^{th} location on the cortex is allocated to one of the mixture components based on the estimated value of $\hat{\boldsymbol{Z}}_{v(j)}$, where $\hat{\boldsymbol{Z}}_{v(j)} = (\hat{Z}_{v(j)_1}, \hat{Z}_{v(j)_2}, \ldots, \hat{Z}_{v(j)_K})'$ and $\hat{Z}_{v(j)_l} = 1$ if location j is allocated to component $l \in \{1, \ldots, K\}$. When the algorithm is run with a value of K that is large, there will result empty mixture components that have not been assigned any voxel locations under $\hat{\boldsymbol{Z}}$. In a sense these empty components have been

automatically pruned out as redundant. The estimated number of mixture components can be obtained by counting the number of non-empty mixture components as follows:

$$\hat{K} = \sum_{l=1}^{K} I\{\sum_{v=1}^{n_v} \hat{Z}_{v_l} \neq 0\}.$$

This estimator requires us to run our algorithm only once for a single value of K and then the resulting number of mixture components assigned a location in $\hat{\boldsymbol{Z}}$ is determined and $\hat{K} \leq K$.

In order to reduce the dimension of parameter space and computing time, we apply clustering to the estimated neural sources. This is achieved by implementing a K-means algorithm to cluster the P locations on the cortex into a smaller number of $J \leq P$ clusters, assuming that $S_j(t) = S_l(t)$ for cortical locations l, j belonging to the same cluster. We investigated different values of $J = 250, 500, 1000$ in our simulation studies. Within the ICM algorithm, the labeling process $\boldsymbol{Z}$ is updated using an efficient chequerboard updating scheme (Song et al., 2019). The update scheme starts with partitioning $\boldsymbol{Z}$ into two blocks $\boldsymbol{Z} = \{\boldsymbol{Z}_W, \boldsymbol{Z}_B\}$ based on a three-dimensional chequerboard arrangement, where $\boldsymbol{Z}_W$ corresponds to "white" voxels and $\boldsymbol{Z}_B$ corresponds to "black" voxels. Under the Markov random field prior with a first-order neighborhood structure, the elements of $\boldsymbol{Z}_W$ are conditionally independent given $\boldsymbol{Z}_B$, the remaining parameters, and the data $\boldsymbol{E}$, $\boldsymbol{M}$. This allows us to update $\boldsymbol{Z}_W$ in a single step, which involves simultaneously updating its elements from their full conditional distributions. The variables $\boldsymbol{Z}_B$ are updated in the same way.

3.2.2 Ant Colony System

Ant Colony System (ACS) is a population-based optimization algorithm introduced by Dorigo and Gambardella (1997). The basic structure of this algorithm is designed to solve the traveling salesman problem in which the aim is to find the shortest path to cover a given set of cities without revisiting any one of them. The inspiring source and development of this algorithm is the observation of the foraging behavior of real ants in their colony. This behavior is exploited in artificial ant colonies for the search of approximate solutions to discrete optimization problems, for continuous optimization problems, and for important problems in telecommunications, such as routing and load balancing, telecommunication network design, or problems in bioinformatics (see, e.g., Reimann et al., 2004; Shmygelska and Hoos, 2005). At the core of this

algorithm is the communication between the ants by means of chemical pheromone trails, which enables them to collectively find short paths between their nest and food source. The framework of this algorithm can be categorized into four main parts: (1) construction of an agent ant solution, (2) local pheromone update of the solution, (3) improving solution by local search, and (4) global pheromone update of the best solution.

At each step of this constructive algorithm a decision is made concerning which solution component to add to the sequence of solution components already built. These decisions are dependent on the pheromone information, which represents the learned experience of adding a particular solution component given the current state of the solution under construction. The accumulated amount of pheromone mirrors the quality of the solution constructed based on the value of the objective function. The pheromone update aims to concentrate the search in regions of the search space containing high quality solutions while there is a stochastic component facilitating random exploration of the search space. In particular, the reinforcement of solution components depending on the solution quality is an important ingredient of ACS algorithms. To learn which components contribute to good solutions can help assembling them into better solutions. In general, the ACS approach attempts to solve an optimization problem by iterating the following two steps: (1) candidate solutions are constructed using a pheromone model, that is, a parameterized probability distribution over the solution space; (2) the candidate solutions are used to modify the pheromone values in a way that is deemed to bias future sampling toward high quality solutions.

The posterior distribution of the dynamic model takes the form $P(\boldsymbol{\Theta}|\boldsymbol{E}, \boldsymbol{M}) = P(\boldsymbol{\Theta}, \boldsymbol{E}, \boldsymbol{M})/P(\boldsymbol{E}, \boldsymbol{M})$, where:

$$
\begin{aligned}
P(\boldsymbol{\Theta}, \boldsymbol{E}, \boldsymbol{M}) &= P(\boldsymbol{E}, \boldsymbol{M}|\boldsymbol{\Theta})P(\boldsymbol{\Theta}) = P(\boldsymbol{E}|\boldsymbol{\Theta})P(\boldsymbol{M}|\boldsymbol{\Theta})P(\boldsymbol{\Theta}) \\
&= \prod_{t=1}^{T} MVN(\boldsymbol{E}(t); \boldsymbol{X}_E \boldsymbol{S}(t), \sigma_E^2 \boldsymbol{H}_E) \times MVN(\boldsymbol{M}(t); \boldsymbol{X}_M \boldsymbol{S}(t), \sigma_M^2 \boldsymbol{H}_M) \\
&\times IG(\sigma_E^2; a_E, b_E) \times IG(\sigma_M^2; a_M, b_M) \times [\prod_{j=1}^{p} \prod_{t=1}^{T} \prod_{l=1}^{K} \mathcal{N}(S_j(t); \mu_l(t), \alpha_l)^{Z_{v(j)l}}] \\
&\times [\prod_{t=2}^{T} MVN(\boldsymbol{\mu}^A(t); \boldsymbol{A}\mu^A(t-1), \sigma_a^2 \boldsymbol{I})] \times MVN(\boldsymbol{\mu}^A(1); \boldsymbol{0}, \sigma_{\mu_1}^2 \boldsymbol{I}) \times \text{Potts}(\boldsymbol{Z}; \beta) \\
&\times \prod_{l=1}^{K} IG(\alpha_l; a_\alpha, b_\alpha) \times \times [\prod_{i=1}^{K-1} \prod_{j=1}^{K-1} \mathcal{N}(A_{ij}; 0, \sigma_A^2)] \times IG(\sigma_a^2; a_a, b_a)
\end{aligned}
\tag{3.2}
$$

where $MVN(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the density of the $\dim(\mathbf{x})$-dimensional multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ evaluated at $\mathbf{x}$; $IG(x; a, b)$ denotes the density of the inverse gamma distribution with parameters a and b evaluated at x; $\mathcal{N}(x; \mu, \sigma^2)$ denotes the density of the normal distribution with mean μ and variance σ^2 evaluated at x; $\text{Potts}(\boldsymbol{Z}; \beta)$ is the joint probability mass function of the Potts model with parameter β evaluated at $\boldsymbol{Z}$. Equation (3.2) represents the objective function to be maximized over $\boldsymbol{\Theta}$. The goal is to optimize over $\boldsymbol{\Theta} = \{\boldsymbol{S}(t), \boldsymbol{Z}, \boldsymbol{\mu}(t), \boldsymbol{\alpha}, \sigma_E^2, \sigma_M^2, \boldsymbol{A}, \sigma_a^2\}$ maximizing the posterior (3.2).

ACS is based on set of agents, each representing artificial ants that construct solutions as sequences of solution components. Agent ant k builds a solution by allocating label ℓ from a set of voxel labels $\Lambda = \{1, \dots K\}$ to the voxel $s \in \{1, ..., N_v\}$ based on a probabilistic transition rule $p^k(s, \ell)$. The transition rule quantifies the probability of ant k, assigning voxel s to label ℓ. This transition rule depends on the pheromone information $\tau(s, \ell)$ of the coupling (s, ℓ) representing the quality of assigning voxel s to label ℓ based on experience gathered by ants in the previous iteration. We let:

$$
\ell = \begin{cases} \arg\max_u \tau(s, u) & \text{if } q \leq q_o \\ p^k(s, \ell) & \text{if } q > q_o \end{cases}
$$

$$
p^k(s, \ell) = \frac{\tau(s, \ell)}{\sum_{u \in \Lambda} \tau(s, u)} \tag{3.3}
$$

where ℓ is a label for voxel s selected according to the transition rule above; $q \sim Uniform(0,1)$; $q_o \in (0,1)$ is a tuning parameter. An artificial ant chooses, with probability q_0, the solution component that maximizes the pheromone function $\tau(s,\ell)$ or it performs, with probability $1-q_0$, a probabilistic construction step according to (3.3). The ACS pheromone system consists of two update rules; one rule is applied whilst constructing solutions (local pheromone update rule) and the other rule is applied after all ants have finished constructing a solution (global pheromone update rule). After assigning a label to a voxel, an ant modifies the amount of pheromone of the chosen couples (s,ℓ) by applying a local pheromone update (3.4):

$$\tau(s,\ell) \leftarrow (1-\rho)\tau(s,\ell) + \rho\tau_o \tag{3.4}$$

where $\rho \in (0,1)$ is a tuning parameter that controls evaporation of the pheromone and τ_o is the initial pheromone value. This operation simulates the natural process of pheromone evaporation preventing the algorithm from converging too quickly (all ants constructing the same solution) and getting trapped into a poor solution. In practice, the effect of this local pheromone update is to decrease the pheromone values via evaporation $(1-\rho)\tau(s,\ell)$ on the visited solution components, making these components less desirable for the subsequents ants. The value of the evaporation rate indicates the relative importance of the pheromone values from one iteration to the following one. If ρ takes a value near 1, then the pheromone trail will not have a lasting effect, and this mechanism increases the random exploration of the search space within each iteration and helps avoid a too rapid convergence of the algorithm toward a sub-optimal region of the parameter space, whereas a small value will increase the importance of the pheromone, favoring the exploitation of the search space near the current solution.

To improve all solutions constructed and also update the other model parameters, we considered incorporating ICM as a local search method. Here, the ICM algorithm is used for both updating the model parameters and also for a local search over the mixture allocation variables. Thus, the update steps corresponding to ACS are combined with running ICM to convergence at each iteration. Finally, after all solutions have been constructed by combined ACS and ICM steps, the quality of all solutions is evaluated using the objective function where the corresponding best solution is selected. We use a global update rule, where pheromone evaporation is again applied on the best solution chosen. Assuming voxel j is assigned to label v for the best

solution, the global update is given as:

$$\tau(j,v) \leftarrow \begin{cases} (1-\rho)\tau(j,v) + \rho\tau_o, \\ (1-\rho)\tau(j,k), & \text{and for all } k \neq v \end{cases}$$

The steps described are performed repeatedly until a change in the objective function becomes negligible and the model parameters from the best solution are returned as the final parameter estimates. The optimal values for the tuning parameters (q_o, τ_o, ρ) used in our ACS algorithm depend on the data. The strategy we adopt for choosing the tuning parameters is by using an outer level optimization on top of the ACS algorithm to optimize over tuning parameters (q_o, τ_o, ρ) within updates at the outer level based on the Nelder–Mead algorithm (Singer and Nelder, 2009) applied to optimize over tuning parameters.

It is well-known that the ICM algorithm is sensitive to initial values and the authors of Song et al. (2019) found this to be the case with the ICM algorithm developed for the spatiotemporal mixture model. The solution obtained, and even the convergence of the algorithm depend rather heavily on the starting values chosen. In the case of ACS, regardless of the initial values, the algorithm finds a better solution with the optimal tuning parameters and this solution tends to be quite stable. This is because ACS is a stochastic search procedure in which the pheromone update concentrates the search in regions of the search space containing high quality solutions to reach an optimum. When considering a stochastic optimization algorithm, there are at least two possible types of convergence that can be considered: convergence in value and convergence in solution. With convergence in value, we are interested in evaluating the probability that the algorithm will generate an optimal solution at least once. On the contrary, with convergence in solution we are interested in evaluating the probability that the algorithm reaches a state that keeps generating the same optimal solution. The convergence proofs are presented in Gutjahr (2000) and Gutjahr (2002). Gutjahr (2000) proved convergence with a probability of $1-\epsilon$ for the optimal solution and more in general for any optimal solution in Gutjahr (2002) of the ACS algorithm. This supports the argument that theoretically the application of ACS to source reconstruction should improve ICM.

The local search ICM algorithm procedure is presented in Algorithm 4 and the ACS-ICM algorithm is presented in Algorithm 5. Convergence of the ICM algorithms is monitored by examining the relative change of the Frobenius norm of the estimated

neural sources on consecutive iterations.

Algorithm 4 presents a detailed description of the ICM algorithm. The ICM algorithm requires full conditional distributions of each model parameter where the mode of the distribution is taken as the update step for the parameter. The full conditional distribution are described and presented in Song et al. (2019). This ICM algorithm is embedded in our ACS-ICM algorithm.

Algorithm 4 Iterated Conditional Modes (ICM) Algorithm.

1: $\Theta = \{\boldsymbol{S}(t), \boldsymbol{Z}, \boldsymbol{\mu}(t), \boldsymbol{\alpha}, \sigma_E^2, \sigma_M^2, \boldsymbol{A}, \sigma_a^2\} \leftarrow$ Initial Value

2: Converged $\leftarrow 0$

3: **while** Converged $= 0$ **do**

4: $\sigma_M^2 \leftarrow \Big[\sum_{t=1}^{T} \frac{1}{2}\big(\boldsymbol{M}(t) - \boldsymbol{X}_M\boldsymbol{S}(t)\big)'\boldsymbol{H}_M^{-1}\big(\boldsymbol{M}(t) - \boldsymbol{X}_M\boldsymbol{S}(t)\big) + b_M\Big] / \Big[a_M + \frac{TN_M}{2} + 1\Big]$

5: $\sigma_E^2 \leftarrow \Big[\sum_{t=1}^{T} \frac{1}{2}\big(\boldsymbol{E}(t) - \boldsymbol{X}_E\boldsymbol{S}(t)\big)'\boldsymbol{H}_E^{-1}\big(\boldsymbol{E}(t) - \boldsymbol{X}_E\boldsymbol{S}(t)\big) + b_E\Big] / \Big[a_E + \frac{TN_E}{2} + 1\Big]$

6: $\sigma_a^2 \leftarrow \Big[\sum_{t=2}^{T} \frac{1}{2}(\boldsymbol{\mu}^A(t) - \boldsymbol{A}\boldsymbol{\mu}^A(t-1))'(\boldsymbol{\mu}^A(t) - \boldsymbol{A}\boldsymbol{\mu}^A(t-1)) + b_a\Big] / \Big[a_a + \frac{(T-1)(K-1)}{2} + 1\Big]$

7: $vec(\boldsymbol{A}) \leftarrow \Big(\frac{1}{\sigma_a^2}\Big(\sum_{t=2}^{T}\boldsymbol{\mu}^A(t)'\boldsymbol{Kr_t}\Big) \times \boldsymbol{C}_1^{-1}\Big)'$, where $\boldsymbol{C}_1 = \frac{1}{\sigma_A^2}\boldsymbol{I}_{(K-1)^2} + \frac{1}{\sigma_a^2}\Big(\sum_{t=2}^{T}\boldsymbol{Kr_t}'\boldsymbol{Kr_t}\Big)$,
and $\boldsymbol{Kr_t} = \Big(\boldsymbol{\mu}^A(t-1)' \otimes \boldsymbol{I}_{K-1}\Big)$

8: **for** $l = 1, ..., K$ **do**

9: $\alpha_l \leftarrow \Big[\frac{\sum_{j=1}^{P}\sum_{t=1}^{T} Z_{v(j)l}(S_j(t) - \mu_l(t))^2}{2} + b_\alpha\Big] / \Big[\frac{T\sum_{j=1}^{P} Z_{v(j)l}}{2} + a_\alpha + 1\Big]$

10: **end for**

11: $\boldsymbol{\mu}(1) \leftarrow \Big(\Big(\sum_{j=1}^{P}(S_j(1)\vec{I}_{K-1})'\boldsymbol{D}_j + \frac{1}{\sigma_a^2}\boldsymbol{\mu}^A(2)'\boldsymbol{A}\Big) \times \boldsymbol{B}_1^{-1}\Big)'$, where $\boldsymbol{B}_1 = \sum_{j=1}^{P}\boldsymbol{D}_j + \frac{1}{\sigma_a^2}\boldsymbol{A'A} + \frac{1}{\sigma_{\mu_1}^2}\boldsymbol{I}_{K-1}$, $\boldsymbol{D}_j = \text{Diag}(\frac{Z_{v(j)l}}{\alpha_l}, l = 2, ..., K)$, $\vec{I}_{K-1} = (1, 1, \ldots, 1)'$ with dim $(\vec{I}_{K-1}) = K - 1$

12: **for** $t = 2, ..., T-1$ **do**

13: $\boldsymbol{\mu}(t) \leftarrow \Big(\Big(\sum_{j=1}^{P}(S_j(t)\vec{I}_{K-1})'\boldsymbol{D}_j + \frac{1}{\sigma_a^2}(\boldsymbol{\mu}^A(t+1))'\boldsymbol{A} + \frac{1}{\sigma_a^2}(\boldsymbol{\mu}^A(t-1)'\boldsymbol{A}')\Big) \times \boldsymbol{B}_2^{-1}\Big)'$
where $\boldsymbol{B}_2 = \sum_{j=1}^{P}\boldsymbol{D}_j + \frac{1}{\sigma_a^2}(\boldsymbol{A'A} + \boldsymbol{I}_{K-1})$

14: **end for**

15: $\boldsymbol{\mu}(T) \leftarrow \Big(\Big(\sum_{j=1}^{P}(S_j(T)\vec{I}_{K-1})'\boldsymbol{D}_j + \frac{1}{\sigma_a^2}(\boldsymbol{\mu}^A(T-1)'\boldsymbol{A}')\Big) \times \boldsymbol{B}_3^{-1}\Big)'$
where $\boldsymbol{B}_3 = \sum_{j=1}^{P}\boldsymbol{D}_j + \frac{1}{\sigma_a^2}\boldsymbol{I}_{k-1}$

16: **for** $j = 1, ..., P$ **do**

Algorithm 4 *Cont.*

17: $\boldsymbol{S}_j \leftarrow -\frac{1}{2}\boldsymbol{\Sigma}_{S_j}\boldsymbol{W}_{2j}$ $\triangleright$ $\boldsymbol{S}_j = (S_j(1), S_j(2), ..., S_j(T))'$
$\boldsymbol{\Sigma}_{S_j}^{-1} = W_{1j}\boldsymbol{I}_T$, $\boldsymbol{W}'_{2j} = (W_{2j}(1), W_{2j}(2), ..., W_{2j}(T))$

where $W_{1j} = \frac{1}{\sigma_M^2}\Big(\boldsymbol{X}_M[,j]'\boldsymbol{H}_M^{-1}\boldsymbol{X}_M[,j]\Big) + \frac{1}{\sigma_E^2}\Big(\boldsymbol{X}_E[,j]'\boldsymbol{H}_E^{-1}\boldsymbol{X}_E[,j]\Big) + \sum_{l=1}^{K}\frac{Z_{v(j)l}}{\alpha_l}$

$W_{2j}(t) = \frac{1}{\sigma_M^2}\Big(-2\boldsymbol{M}(t)'\boldsymbol{H}_M^{-1}\boldsymbol{X}_M[,j] + 2(\sum_{v\neq j}\boldsymbol{X}_M[,v]S_v(t))'\boldsymbol{H}_M^{-1}\boldsymbol{X}_M[,j]\Big)$
$+ \frac{1}{\sigma_E^2}\Big(-2\boldsymbol{E}(t)'\boldsymbol{H}_E^{-1}\boldsymbol{X}_E[,j] + 2(\sum_{v\neq j}\boldsymbol{X}_E[,v]S_v(t))'\boldsymbol{H}_E^{-1}\boldsymbol{X}_E[,j]\Big) - 2\sum_{l=1}^{K}\frac{\mu_l(t)}{\alpha_l}$

$\boldsymbol{X}_M[,j]$, $\boldsymbol{X}_E[,j]$ denote the jth column of $\boldsymbol{X}_E$ and $\boldsymbol{X}_M$

18: **end for**

19: Let $\mathbb{B}$ denote the indices for "black" voxels and $\mathbb{W}$ denote the indices for "white" voxels.

20: **for** $\kappa \in \mathbb{B}$ *simultaneously* **do**

21: $Z_{\kappa q} \leftarrow 1$ and $Z_{\kappa l} \leftarrow 0$, $\forall l \neq q$
where $q = \text{argmax}_{h\in\{1,...,K\}} P(h)$, and

22: $$P(h) = \frac{\alpha_h^{-TN_{j\kappa}/2}\times\exp\Big(-\frac{1}{2}\sum_{j|v(j)=\kappa}\alpha_h^{-1}\sum_{t=1}^{T}(S_j(t)-\mu_h(t))^2+2\beta\sum_{v\in\delta_\kappa}Z_{vh}\Big)}{\sum_{l=1}^{K}\alpha_l^{-TN_{j\kappa}/2}\times\exp\Big(-\frac{1}{2}\sum_{j|v(j)=\kappa}\alpha_l^{-1}\sum_{t=1}^{T}(S_j(t)-\mu_l(t))^2+2\beta\sum_{v\in\delta_\kappa}Z_{vl}\Big)}$$

where $N_{j\kappa}$ is the number of cortical locations contained in voxel κ.

23: **end for**

24: **for** $\kappa \in \mathbb{W}$ *simultaneously* **do**

25: $Z_{\kappa q} \leftarrow 1$ and $Z_{\kappa l} \leftarrow 0$, $\forall l \neq q$
where $q = \text{argmax}_{h\in\{1,...,K\}} P(h)$, and

26: $$P(h) = \frac{\alpha_h^{-TN_{j\kappa}/2}\times\exp\Big(-\frac{1}{2}\sum_{j|v(j)=\kappa}\alpha_h^{-1}\sum_{t=1}^{T}(S_j(t)-\mu_h(t))^2+2\beta\sum_{v\in\delta_\kappa}Z_{vh}\Big)}{\sum_{l=1}^{K}\alpha_l^{-TN_{j\kappa}/2}\times\exp\Big(-\frac{1}{2}\sum_{j|v(j)=\kappa}\alpha_l^{-1}\sum_{t=1}^{T}(S_j(t)-\mu_l(t))^2+2\beta\sum_{v\in\delta_\kappa}Z_{vl}\Big)}$$

where $N_{j\kappa}$ is the number of cortical locations contained in voxel κ.

27: **end for**

28: Check for convergence. Set Converged = 1 if so.

29: **end while**

Algorithm 5 Ant Colony System (ACS)-ICM Algorithm.

1: $\Theta \leftarrow$ Initial Value; set tuning parameters τ_o, q_o, ρ and N_{ants}.

2: Initialize pheromone information $\tau(i,\ell) = \tau_o$, for each $(i,\ell) \in \{1,\dots N_v\} \times \{1,\dots K\}$ representing information gathered by ants.

3: Construct candidate solutions for each of N_{ants} ants. For ant j, we find a candidate voxel labeling $\mathbf{Z}^{(j)} = (\mathbf{Z}_1^{\prime(j)}, \mathbf{Z}_2^{\prime(j)}, \dots, \mathbf{Z}_{N_v}^{\prime(j)})'$. This is done sequentially for each ant j.

- Construct candidate by assigning label l to voxel s using the transition probability rule:

$$\ell = \begin{cases} \arg\max_u \tau(s,u) & \text{if } q \le q_o \\ p(s,\ell) & \text{if } q > q_o \end{cases}$$

  where if $q > q_o$ the label for voxel s is drawn randomly from $\{1,\dots,K\}$ with probability

$$p(s,\ell) = \frac{\tau(s,\ell)}{\sum_{u\in\Lambda}\tau(s,u)},$$

  and where $q \sim$ uniform$[0,1]$.

- Assuming voxel s is assigned label ℓ set:

$$\tau(s,\ell) \leftarrow (1-\rho)\tau(s,\ell) + \rho\tau_o$$

  and for all $k \neq l$

$$\tau(s,\ell) \leftarrow (1-\rho)\tau(s,k)$$

  where ρ is a tuning parameter in (0,1), which represents evaporation of the pheromone trails and $\tau_o > 0$.

4: For all ants, improve candidate solutions by running ICM to convergence (this also allows an update to the other model parameters) $\boldsymbol{\Theta} = \{\{\boldsymbol{\mu}^A(1), \boldsymbol{\mu}^A(2), ..., \boldsymbol{\mu}^A(T)\}, \{\alpha_1, \alpha_2, ..., \alpha_k\}, \sigma_E^2, \sigma_M^2, \{S_j(t), t = 1, 2\dots, T, j = 1, 2, ..., P\}, \boldsymbol{A}, \sigma_a^2\}$.

5: For all N_{ants} solutions, evaluate the quality of each ant's solution using objective function: $P(\boldsymbol{\Theta}, \boldsymbol{E}, \boldsymbol{M})$. Keep track of the best value. The current solution for each ant serves as the starting value for the next iteration.

Algorithm 5 *Cont.*

6: Apply global updating of the pheromone function. For the best solution, (s, ℓ) update the pheromone as follows:

Assuming voxel s is assigned label ℓ set:

$$\tau(s, \ell) \leftarrow (1 - \rho)\tau(s, \ell) + \rho\tau_o$$

and for all $k \neq \ell$:

$$\tau(s, \ell) \leftarrow (1 - \rho)\tau(s, k)$$

Check for convergence via increase in $logP(\boldsymbol{\Theta}, \boldsymbol{E}, \boldsymbol{M})$. Go back to step 3

7: Return all voxel labeling $\mathbf{Z}$ and model parameters $\boldsymbol{\Theta}$ from the best solution.

3.3 Simulation Studies

In this section, we use a simulation study to evaluate the performance of our algorithm. The simulation study assesses the quality of the source estimates and the optimized objective function values obtained when using our proposed algorithm in comparison to the existing ICM algorithm developed in Song et al. (2019). We then make comparisons between ACS-ICM and the ICM algorithm applied to combined simulated EEG and MEG data.

3.3.1 Simulation Approach

The MEG and EEG data were both generated from four scenarios with two, three, four and nine latent states corresponding to regions of neural activity. In each of the four cases, one of the states is inactive, while the remaining states represent different regions of brain activity generated by Gaussian signals. The temporal profile of brain activity at each of the brain locations in the activated regions is depicted in Appendix A, Figures A.1 and A.2. We projected the source activity at 8196 brain locations from the cortex onto the MEG and EEG sensor arrays using the forward operators $\boldsymbol{X}_M$ and $\boldsymbol{X}_E$. The simulated data were then obtained by adding Gaussian noise at each sensor, where the variance of the noise at each sensor was set to be 5% of the temporal variance of the signal at that sensor. The number of mixture components K was set to be the true number of latent states (either two, three, four, or nine) in the model. We simulated 500 replicate datasets and both ACS-ICM and ICM were

applied to each dataset. For each simulated dataset we applied our algorithm with J = 250, 500, 1000 clusters so as to evaluate how the performance varies as this tuning parameter changes. We initialized both algorithms using the same starting values. For each replicate we computed the correlation between the estimated sources and the true sources $\text{Corr}[(\boldsymbol{S}(1)', \boldsymbol{S}(2)', \ldots, \boldsymbol{S}(T)'), (\hat{\boldsymbol{S}}(1)', \hat{\boldsymbol{S}}(2)', \ldots, \hat{\boldsymbol{S}}(T)')]$ as a measure of agreement. This measure was also averaged over the 500 replicate datasets to compute average correlation. In addition, we estimated the Mean-Squared Error (MSE) of the estimator $\hat{S}_j(t)$ based on the R = 500 simulation replicates for each brain location j and time point t. The Total MSE (TMSE) was computed by adding all the MSE's over brain locations and time points. This was done separately for locations in active and inactive regions.

In our simulation studies, the ACS-ICM algorithm had four tuning parameters. The first denoted as $q_o \in (0, 1)$ controlled the degree of stochasticity, with larger values corresponding to less stochasticity and thus less random exploration of the parameter space. When a solution is chosen, another tuning parameter τ_0 controlled the amount of pheromone reinforcing this solution in the information available to the other ants. A third tuning parameter ρ controlled the evaporation of pheromone, and finally a fourth tuning parameter N_{ants} controlled the number of ants. The number of ants (N_{ants}) was fixed at 10, a value for which we have seen generally good performance. This was chosen based on computing efficiency and similar results (objective function values) from using $N_{ants} \geq 10$. The remaining optimal tuning parameters (q_o, τ_0, ρ) for all simulations cases were chosen using an outer level optimization using the Nelder–Mead algorithm.

3.3.2 Simulation Results

Evaluation of Neural Source Estimation

We present the average correlation between the estimated values and the truth for the algorithms considered in our study in Table 3.1. Inspecting Table 3.1, we observe that for all cases considered for the true number of latent states (either two, three, four, or nine), the estimates obtained from the ACS-ICM algorithm yielded a higher average correlation than those obtained from ICM. In addition, with respect to the number of clusters, ACS-ICM resulted in a higher average correlation than ICM uniformly for all cluster sizes (250, 500, 1000). In summary, the average correlation was significantly improved when estimates were computed using the ACS-ICM algorithm for both large

and small numbers of latent states as well as cluster sizes. In addition, we present in Appendix A, Figure A.4, violin plots comparing the correlation values obtained from each of the algorithms for different simulation cases across all replicates. These plots show the entire distribution and provide a better assessment of each algorithm for simulation replicates. Observing Figure A.4, we can see that ACS-ICM provides the highest correlation values uniformly in all simulation scenarios.

The TMSE for all simulation scenarios is presented in Appendix A, Table A.1. To improve the readability of the results from TMSE values, we computed the relative percentage improvement in TMSE of the neural source estimators from ICM to ACS-ICM. Here, using ICM as the reference algorithm, the relative percentage improvement is defined as the ratio of the difference in TMSE between ICM and ACS-ICM to its ICM TMSE value multiplied by 100. The results of this computation are presented in Table 3.2. In all simulation scenarios for Table 3.2, ACS-ICM performed better and showed a significant improvement as compared to ICM.

Specifically with respect to the number of clusters, ACS-ICM was roughly 10% better than ICM with respect to TMSE when the cluster size was 250. For both small and large numbers of latent states, ACS-ICM was better than ICM in the active region with significant improvements. This shows that ACS-ICM outperforms ICM in active regions using both small and large numbers of latent states. The total MSEs were decomposed into total variance and total squared bias for the same distinct cases of the simulation depicted in Table 3.2. From the results, when we consider active regions with different numbers of clusters, we observe that ACS-ICM was better than ICM based on the total squared bias due to the percentage of relative change. Based on the total variance we also noticed a similar positive change from ICM to ACS-ICM uniformly for all values of K. It is also clear that for inactive regions, ACS-ICM was better than ICM for both total variance and squared bias for all simulation cases considered. Overall, these results from the TMSE demonstrate a significant improvement obtained from our algorithm when considering total squared bias and variance for our simulation studies. This improvement was observed uniformly across all conditions.

We present in Figure 3.1 boxplots comparing the final objective function values obtained from each of the algorithms for the different simulation scenarios across all replicates. Again, a clear pattern emerged showing that ACS-ICM yielded the highest objective function values uniformly in all cases. Overall, ACS-ICM outperformed the ICM algorithm uniformly with respect to both neural source estimates and the values

of the objective function. This indicates the superiority of the ACS-ICM algorithm over ICM for computing neural source estimates for the spatiotemporal model.

Table 3.1: Simulation study I—Average (Ave.) correlation between the neural source estimates and the true values for the ICM and ACS-ICM algorithms.

		$K = 2$	$K = 3$	$K = 4$	$K = 9$
Algorithm	**Clusters**	**Ave. Corr.**	**Ave. Corr.**	**Ave. Corr.**	**Ave. Corr.**
ICM	250	0.60	0.63	0.62	0.54
ACS-ICM	250	0.64	0.67	0.63	0.59
ICM	500	0.53	0.55	0.49	0.44
ACS-ICM	500	0.56	0.61	0.53	0.46
ICM	1000	0.41	0.43	0.40	0.37
ACS-ICM	1000	0.46	0.47	0.45	0.43

Evaluation of Mixture Component Estimation

In addition to evaluating point estimation and objective function maximization, we also evaluated model selection, comparing $\hat{K}_{ACS}$ and $\hat{K}_{ICM}$, that is, the estimators obtained from ACS-ICM and ICM, respectively. We focused on estimating the number of mixture components and evaluating the sampling distribution of $\hat{K}_{ACS}$ and $\hat{K}_{ICM}$. The following five scenarios were considered in our experiments:

1. Two latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A.1, panel (a).

2. Three latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A.1, panel (c).

3. Four latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A.1, panel (e).

4. Four latent states with Gaussian and sinusoidal source activity in the active regions depicted in Appendix A, Figure A.1, panel (g).

5. Nine latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A.2, panel (a).

We simulated the data for each of the five scenarios considered, and added 5% Gaussian noise at the sensors with 1000 replicate datasets used in each case. The

Table 3.2: Simulation study I-Percentage of relative improvement in Total Mean-Squared Error (TMSE) of the neural source estimators decomposed into variance and squared bias from ICM to ACS-ICM. This total was obtained separately for locations in active regions and then for the inactive region.

		Active Region			**Inactive Region**		
Algorithm	**Clusters**	**TMSE (%)** %	**(Bias)^2** %	**Variance** %	**TMSE (%)** %	**(Bias)^2** %	**Variance** %
				$K = 2$			
ICM→ACS-ICM	250	9.78	11.11	8.93	9.93	6.15	13.16
ICM→ACS-ICM	500	6.63	4.39	8.57	9.48	9.71	9.26
ICM→ACS-ICM	1000	2.95	2.04	4.03	3.86	1.57	5.70
				$K = 3$			
ICM→ACS-ICM	250	5.10	7.80	2.76	4.97	4.31	5.60
ICM→ACS-ICM	500	24.85	25.42	24.31	15.89	18.26	13.61
ICM→ACS-ICM	1000	36.57	53.61	20	10.83	10	11.61
				$K = 4$			
ICM→ACS-ICM	250	12.24	11.11	12.88	8.90	14.19	4.94
ICM→ACS-ICM	500	17.94	14.75	20.71	2.86	3.10	2.64
ICM→ACS-ICM	1000	29.28	30.30	30.65	2.76	3.62	2.06
				$K = 9$			
ICM→ACS-ICM	250	31.14	22.77	27.65	15.44	13.07	17.58
ICM→ACS-ICM	500	14.83	11.40	18.20	17.0	20.39	13.71
ICM→ACS-ICM	1000	23.79	25.52	22.08	8.65	7.03	10.14

algorithms were run with an upper bound of $K = 10$ for each of the 5000 simulated datasets. For each dataset, we computed the value of the estimator, and histograms representing the sampling distributions are presented in Figure 3.2, for each of the five cases above illustrating the sampling distribution of $\hat{K}_{ICM}$ (panels (a)–(e)) and $\hat{K}_{ACS}$ (panels (f)–(j)) corresponding to the first and second row, respectively. Observing Figure 3.2, where the true signals are well separated in the simulation experiments, in all cases except for the case with a larger number of latent states ($K = 9$), the mode of the sampling distributions corresponds to the true number of latent states for both the ACS-ICM and ICM algorithms. In the case of nine neural sources, ACS-ICM gave better and improved results than ICM. Additionally, Table 3.3 reports both the bias and mean-squared error of the estimators from ACS-ICM ($\hat{K}_{ACS}$) and ICM ($\hat{K}_{ICM}$).

From Table 3.3, both ACS-ICM and ICM are biased and over-estimated for the small number of latent states but underestimated for the large number of latent states. More importantly, the estimate for the number of mixture components obtained from ACS-ICM exhibited the best performance in terms of both bias and MSE uniformly for all cases considered. This is based on $|\text{Bias}(\hat{K}_{ACS})| < |\text{Bias}(\hat{K}_{ICM})|$ and $\text{MSE}(\hat{K}_{ACS}) < \text{MSE}(\hat{K}_{ICM})$.

We repeated the simulation studies for all five cases but where true signals are less well separated by altering the true signals depicted in Appendix A, Figure A.3. We present histograms depicted in Figure 3.3, for each of the five cases above, illustrating the sampling distribution of $\hat{K}_{ICM}$ (panels (a)–(e)) and $\hat{K}_{ACS}$ (panels (f)–(j)). In this case, the mode of the sampling distribution corresponds to the true number of latent states when $K = 2$ and $K = 3$ but not for the case with four and nine latent states with both algorithms. In Table 3.3 we compare the bias and mean square error of $\hat{K}_{ICM}$ and $\hat{K}_{ACS}$ under this simulation settings. Similarly, under these settings, ACS-ICM outperformed ICM in terms of the bias and mean square error; thus, $|\text{Bias}(\hat{K}_{ACS})| < |\text{Bias}(\hat{K}_{ICM})|$ and $\text{MSE}(\hat{K}_{ACS}) < \text{MSE}(\hat{K}_{ICM})$. In summary, for model selection, based on the results presented in Table 3.3, ACS-ICM showed an overall better performance over ICM uniformly for all eight conditions considered.

Table 3.3: Simulation study II—bias and Mean Square Error (MSE) of estimated number of mixture components ($\hat{K}$) from the 1000 simulation replicates when the algorithms were run with $K = 10$.

	$K = 2$		$K = 3$		$K = 4$		$K = 9$	
Algorithm	**Bias($\hat{K}$)**	**MSE($\hat{K}$)**	**Bias($\hat{K}$)**	**MSE($\hat{K}$)**	**Bias($\hat{K}$)**	**MSE($\hat{K}$)**	**Bias($\hat{K}$)**	**MSE($\hat{K}$)**
The case where the true signals were well-separated								
ICM	0.11	0.13	0.06	0.42	0.20	0.44	−2.54	6.19
ACS-ICM	0.04	0.06	0.02	0.38	0.10	0.31	−2.01	4.46
The case where the true signals were less well-separated								
ICM	0.11	0.13	0.525	0.58	−1.02	1.63	−4.83	16.12
ACS-ICM	0.05	0.07	0.35	0.41	−1.00	1.31	−3.68	10.47

Whereas the ACS-ICM algorithm showed superiority in terms of quality of source estimates, a drawback is that it is computationally expensive relative to ICM due to its population-based and iterative procedure. Notwithstanding, this might not be a serious challenge for source localization problems, which do not require real-time solutions in most situations. With regards to computation time, on the Niagara cluster running R software on a single core (Intel Skylake 2.4 GHz, AVX512), ICM computed source estimates in approximately 2 min whereas ACS-ICM computed estimates in

roughly 6 h and 30 min.

3.4 Application to Scrambled Face Perception MEG/ EEG Data

In this section, we present the application of our methodology for comparison with EEG and MEG data measuring an event-related response to the visual presentation of scrambled faces in a face perception study. In addition, we demonstrate how a nonparametric bootstrap can be used to obtain standard errors, confidence intervals and T-maps. The data from both MEG and EEG were obtained from a single subject in an experimental paradigm that involved repeated random presentation of a picture showing either a face or a scrambled face while the subject was required to make a symmetry judgement. The scrambled faces were created through 2D Fourier transformation, random phase permutation, inverse transformation and outline-masking of each face. The experiment involved a sequence of trials, each lasting 1800 ms, where in each trial the subject was presented with one of the pictures for a period of 600 ms while being required to make a four-way, left–right symmetry judgment while brain activity was recorded over the array. Both scrambled faces and unscrambled faces were presented to the subject; however, our analysis will focus only on trials involving scrambled faces. This produced a multivariate time series for each trial, and the trial-specific time series were then averaged across trials to create a single multivariate time series; the average evoked response is depicted in Figure 3.4, panel (a), for MEG data, and panel (c), for EEG data. Looking from a spatial perspective, at a given time point, each array recorded a spatial field such as that depicted in Figure 3.4, panel (b), which shows the MEG spatial field at a particular time point, and Figure 3.4, panel (d), which shows the EEG spatial field at the same time point. This experiment was conducted while EEG data were recorded, and then again on the same subject while MEG data were recorded.

The EEG data were acquired on a 128-sensors ActiveTwo system with a high sampling rate of 2048 Hz and down-sampled to 200 Hz. The EEG data were re-referenced to the average over good channels. The resulting EEG data were a trial-specific multivariate time series and contained 128 sensors, 161 time points and 344 trials. For real data analysis, the trial-specific time series were averaged across trials to produce a single average evoked response. The MEG data were acquired on 274

sensors with a CTF/VSM system, with a high sampling rate of 480 Hz and down-sampled to 200 Hz. The MEG data obtained were a trial -specific multivariate time series and contained 274 sensors, 161 time points and 336 trials. We obtained a temporal segment of the data from time point t = 50 to t = 100, resulting in 51 time points for both the EEG and MEG data. The trial-specific time series were averaged across trials to produce a single average evoked response. Detailed description of the data and related analysis can be found in Henson et al. (2003, 2007, 2009a, 2009b, 2010). In addition, a link to the open access data repository used for analysis can be found here: `https://www.fil.ion.ucl.ac.uk/spm/data/mmfaces` (accessed on 14 November 2020).

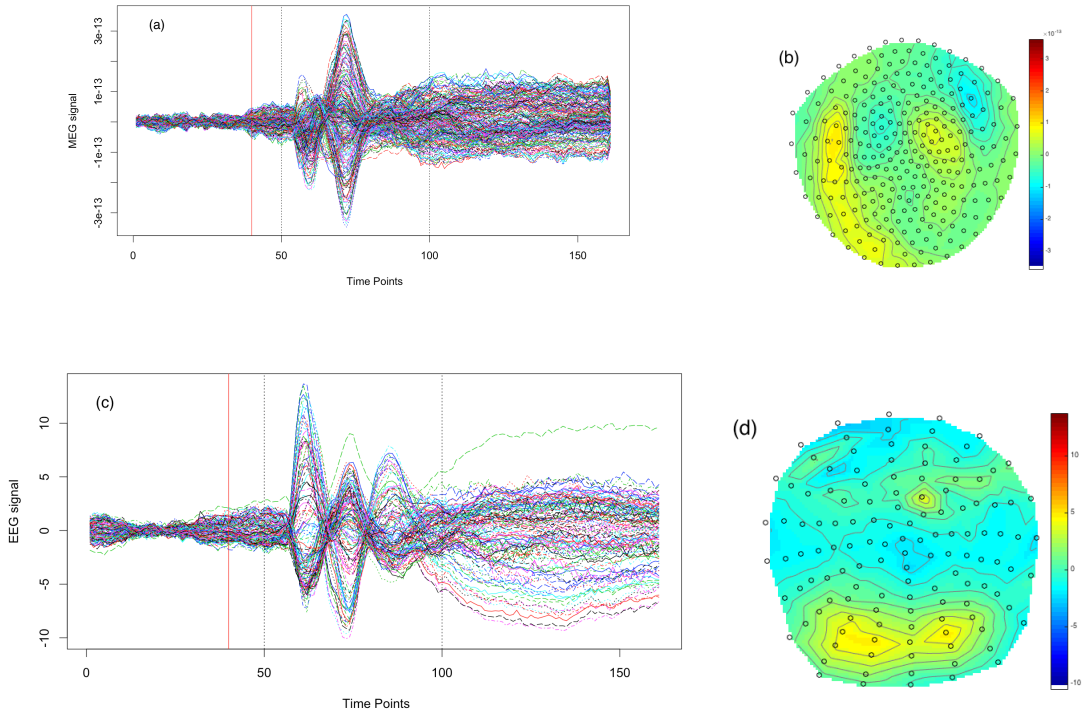


Figure 3.4: The Magnetoencephalography (MEG) and Electroencephalography (EEG) data considered in the face perception study: panels (**a**,**c**) show the time series observed at each MEG sensor and EEG sensor, respectively; panels (**b**,**d**) depict the spatially interpolated values of the MEG data and the EEG data, respectively, each observed at $t = 80$, roughly 200 ms after presentation of the stimulus. In panels (**b**,**d**) the black circles correspond to the sensor locations after projecting these locations onto a 2-dimensional grid (for presentation).

We set the upper bound at $K = 10$ mixture components, voxels as $n_v = 560$,

$\beta = 0.3$ (hyperparameter of Potts model) and a cluster size of J = 250. For our real data application, the optimal tuning parameters $(q_o, \tau_0, \rho, N_{ants}) = (0.43, 0.05, 0.64, 10)$ were selected similarly using the Nelder–Mead algorithm. First, the ICM algorithm was run to convergence and the estimates obtained were used as the initial values for the ACS-ICM algorithm. Our primary interest lies in the estimated neural sources $\hat{\boldsymbol{S}}(t)$ and we computed the total power of these estimated sources obtained from both algorithms at each brain location, which was then mapped onto the cortex. The cortical maps showing the spatial patterns from the estimated power of the reconstructed sources are displayed in Figure 3.5. The first and second row depict the corresponding results obtained from the ICM and ACS-ICM algorithms, respectively. As shown in Figure 3.5, the greatest power occurred on the bilateral ventral occipital cortex for both estimated sources from the ACS-ICM and ICM algorithms. Interestingly, the results from ACS-ICM estimates also differed when compared with the results from ICM in the left ventral frontal and right ventral temporal regions. In particular, the ACS-ICM estimate detected higher power, whereas ICM showed low activation in these regions. The estimated source locations of these region is responsible for high-level visual processing. Therefore, the cortical power map seems to represent regions that would be expected to show scrambled face-related activity. To compare the general quality of the estimates from ACS-ICM versus ICM, we show the plot of the final objective function values obtained from the algorithms in Figure 3.6. We see clearly that the application of ACS-ICM has led to higher quality estimates with much larger posterior density values.

The ACS-ICM algorithm used to maximize the posterior distribution produces only point estimates of the neural source activity. In addition to the point estimates, we applied a nonparametric bootstrap on the trial-specific multivariate time series to obtain confidence interval estimates and characterize the variability in our source estimates, which is another extension to Song et al. (2019). The interval estimates were constructed by resampling the trial-specific MEG/EEG time series data with replacement. From each resampled dataset, we obtained the average evoked response and then run the ACS-ICM algorithm for a total of 400 nonparametric bootstrap replicates. This procedure was made feasible using parallel computation on a large number of computing cores. We constructed a cortical map of the bootstrap standard deviations of the total power of the estimated source. To account for uncertainty in our point estimates, we constructed a T-map and this is depicted in Figure 3.7. A T-map is the ratio of the ACS-ICM point estimate of the source activity to its bootstrap

standard deviations. The T-map represents the best depiction of reconstructed power since it accounts for standard errors that a simple map of the point estimates does not. Broadly, the T-map seems to indicate similar results to those obtained from point estimates, in particular with respect to high power activation on the bilateral ventral occipital cortex and right ventral temporal region. An interesting observation from the T-map is the detection of a high signal in the left ventral temporal region but a low activation from the point estimate.

In addition, we present the temporal summary from our bootstrap replicates representing the interval estimation for the estimated temporal profile of brain activity at the peak location of the T-map. The interval estimate represents a 95% confidence interval depicted in Figure 3.8. One of the key components of our work is varying the inverse temperature parameter for sensitivity analysis. We fixed the inverse temperature at $\beta = (0.1, 0.44)$ and run the ACS-ICM algorithm to convergence. We run our algorithm together with $K = 10$, $n_v = 560$ and a cluster size of J = 250. For $\beta = 0.1$, the corresponding results obtained are depicted in the first row of Figure 9. The results indicate activation on the bilateral ventral occipital cortex. Additionally, at $\beta = 0.44$, the power map results from ACS-ICM, depicted in the second row of Figure 3.9, differ when compared with results from ACS-ICM at $\beta = 0.1$ In particular, the highest power signals occured in the right ventral temporal region where there was low activation for using $\beta = 0.1$.

For our real data application we applied both algorithms with J = 500 clusters so as to evaluate how the performance varies as this tuning parameter changes. The results are displayed in Appendix B. The corresponding results obtained from ACS-ICM are displayed in the second row of Figure A.5. Examining Figure A.5, ACS-ICM seems to indicate similar results to those obtained from using a tuning parameter of J = 250, in particular with respect to activation on the bilateral ventral occipital cortex. For our sensitivity analysis, we present results obtained from using inverse temperature ($\beta = 0.1$ and $\beta = 0.44$) displayed in Figure **??**. We observe that from ACS-ICM, the spatial spread of the high power occurs on the bilateral ventral occipital cortex. In addition, source estimates obtained from ACS-ICM indicate bilateral activation in the occipital cortex, and activation in the right temporal and right frontal regions of the brain. These estimated source locations reveal activation in areas known to be involved in the processing of visual stimuli. More interestingly, ACS-ICM also detected high power in a region on the corpus callosum; given that the inverse problem is ill-posed with an infinite number of possible configurations this may be the reason.

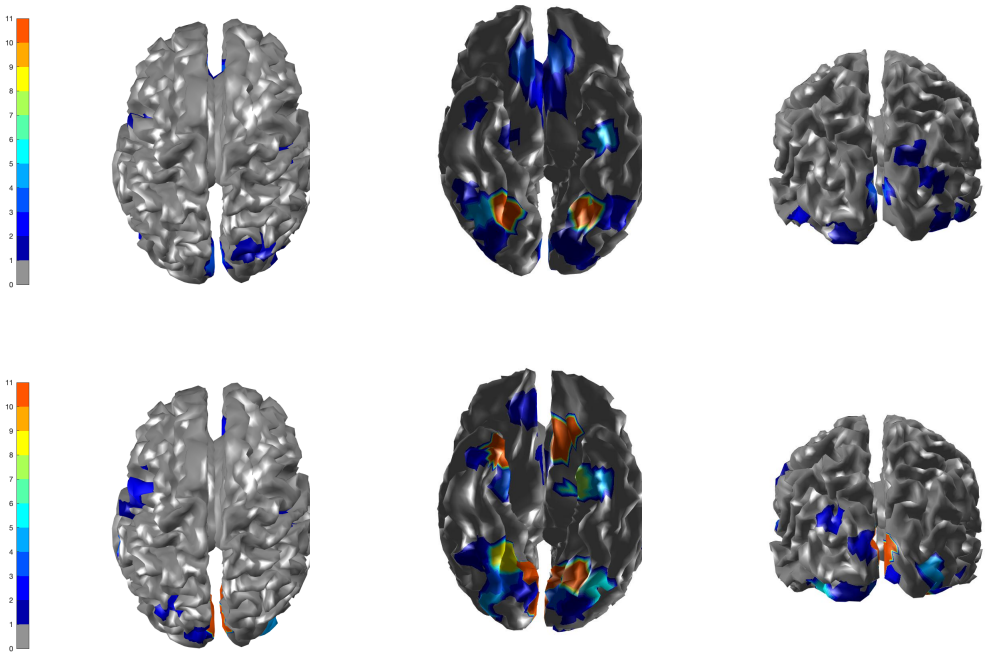


Figure 3.5: Brain activation for scrambled faces—the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** displays results from our ICM algorithm applied to the combined MEG and EEG data; **Row 2** displays results from ACS-ICM applied to the combined MEG and EEG data.

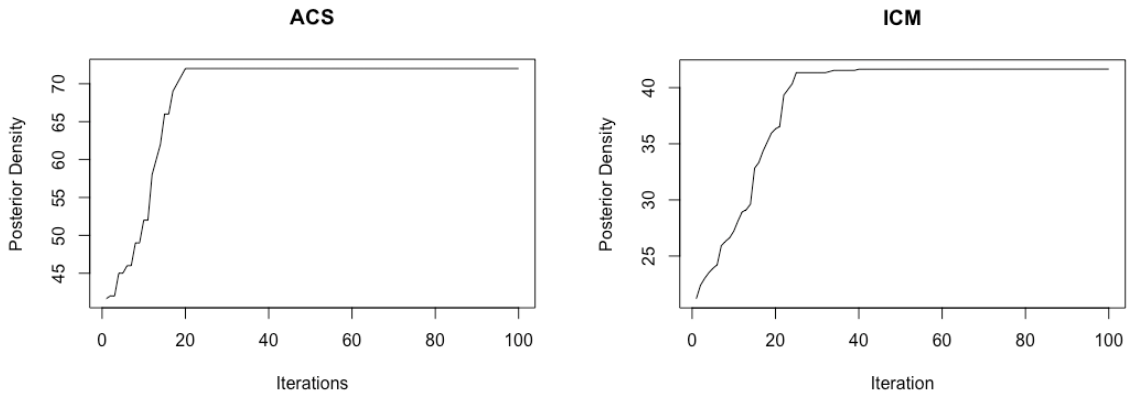


Figure 3.6: Objective function values obtained from the data with the ACS-ICM (**left**) and ICM (**right**) algorithms.

In our real data analysis, the required computation time for ICM was 3 min on a single core (Intel Skylake 2.4 GHz, AVX512) with R software, whereas the computation time for the ACS-ICM was roughly 7 h. The choice of cluster size will

have an impact on the computational time required by the algorithm. With regards to ACS-ICM, the required computing time for a cluster size of 250 was approximately 7 h, whereas for a cluster size of 500, ACS-ICM required 12 h of computing time. While there is a substantially increase in computation, the paper has demonstrated uniform improvements in the quality of the solutions, in terms of both source estimation and model selection. Furthermore, the bootstrap can be implemented in parallel on a computing cluster to obtain standard errors with no increase to the required computation time.

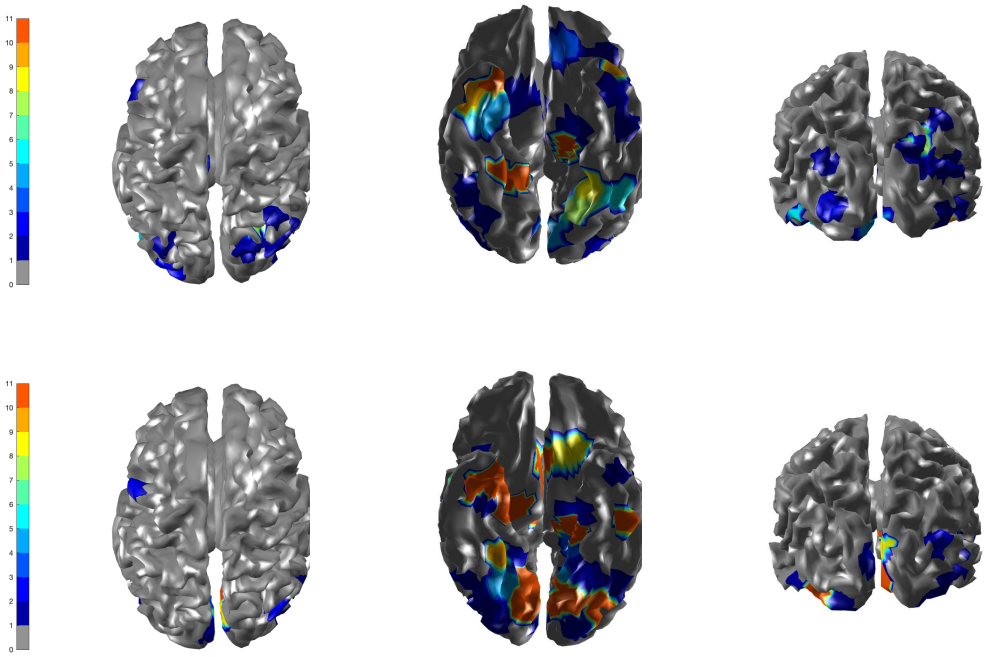


Figure 3.7: The spatial profile of brain activity from ACS-ICM based on our bootstrap replicates. **Row 1** displays standard deviations of the total power of the estimated source activity; **Row 2** displays the T-map.

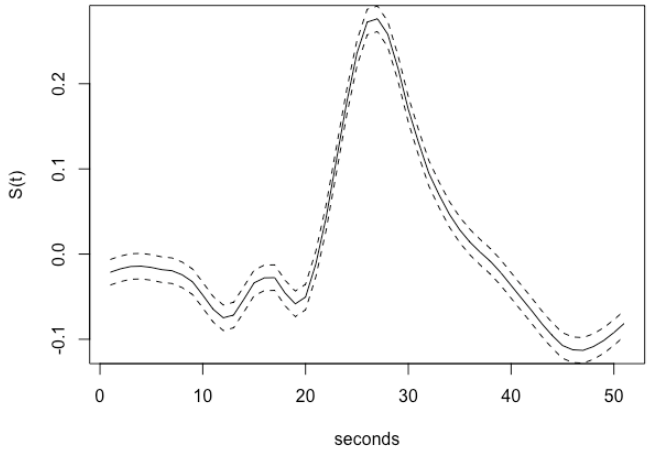


Figure 3.8: The 95% confidence interval for the estimated temporal profile of brain activity at the peak location of the T-map from the bootstrap replicates.

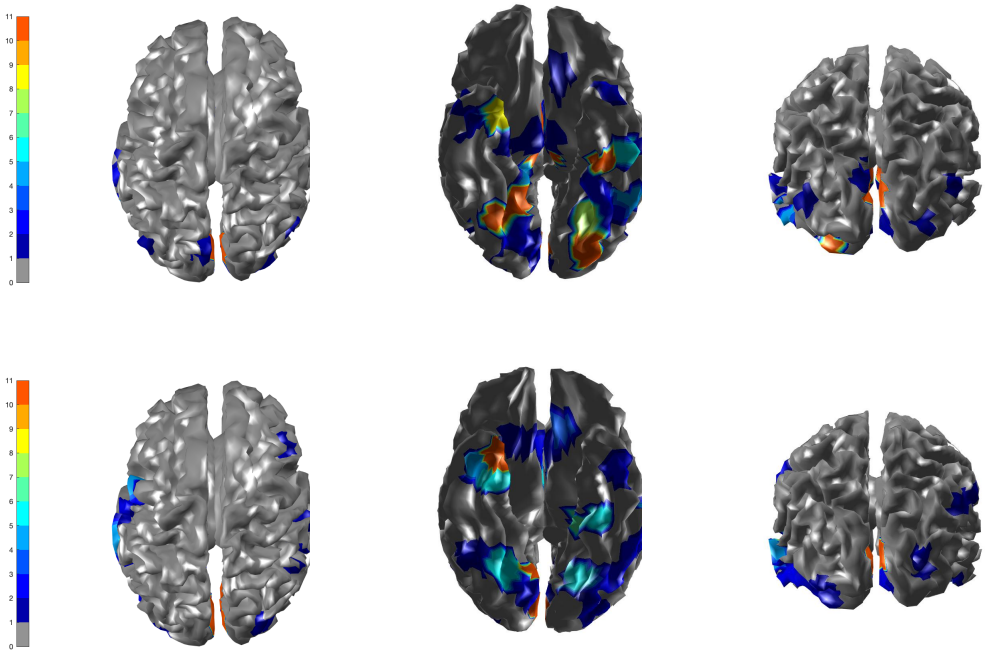


Figure 3.9: Brain activation for scrambled faces—the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** displays results from our ACS-ICM algorithm applied to the combined MEG and EEG data with $\beta = 0.1$; **Row 2** displays results from ACS-ICM applied to the combined MEG and EEG data with $\beta = 0.44$.

3.4.1 Residual Diagnostics for the Scrambled Faces MEG and EEG Data

We assessed the goodness of fit of the model by checking the residual time series plot, normal quantile–quantile plot and residuals versus fitted values after running the ACS-ICM and ICM algorithms. This was done by computing the residuals for both EEG and MEG after applying both algorithms. The residuals were computed as $\hat{\boldsymbol{\epsilon}}_M(t) = \boldsymbol{M}(t) - \boldsymbol{X}_M\hat{\boldsymbol{S}}(t)$ and $\hat{\boldsymbol{\epsilon}}_E(t) = \boldsymbol{E}(t) - \boldsymbol{X}_E\hat{\boldsymbol{S}}(t)$ at each time point $t = 1, \ldots, T$. The assumption made for the residuals was that they should be draws from a mean-zero Gaussian distribution if the assumed model generated the observed data. The residual time series plot for EEG and MEG from the ACS-ICM algorithm is displayed in Figure 3.9, panels (a) and (b). The plots from Figure 3.9, panels (a) and (b), also depicts residuals time series plots obtained from ICM for EEG and MEG, respectively. Examining the plots, the residual time series plots obtained from both algorithms exhibit similar patterns for MEG and EEG. However, there are significant improvements seen in estimates from ACS-ICM. Specifically for the EEG data, there are sensors with relatively large peaks remaining from the ICM but significant improvements from ACS-ICM as we observe no fewer residuals patterns relative to ICM. In the case of MEG data, we observe that the residuals obtained from ACS-ICM reveal few sensors with peaks remaining as compared to ICM, where there are more sensors with large peaks and residuals.

In Figures 3.9 and 3.9, panels (c) and (d), we show plots of the residuals versus fitted values from ACS-ICM and ICM. For the EEG data, the ACS-ICM residuals reveal fewer extreme values with smaller residual patterns but more outliers are seen in the residuals obtained from ICM comparably. The residuals obtained from ICM are characterized by higher values to the left of zero and lower values to the right of zero. In the case of MEG data, the residuals obtained from ACS-ICM also show fewer extreme values with a smaller residual pattern but a similar resemblance for residuals obtained from the ICM algorithm with few extreme values outside the zero band. We observe more extreme values in the residual plot obtained from ICM than that obtained from ACS-ICM. This signifies improvements of the ACS-ICM algorithm over ICM. Inspecting Figure 3.9, panels (e) and (f), reveals normal quantile–quantile plots for the EEG and MEG residuals obtained from the ACS-ICM algorithm. There is no deviation from normality observed from the EEG and MEG data. Hence, the Gaussian assumption holds from using the ACS-ICM algorithm. In the case of the

ICM, in Figure 3.9, panels (e) and (f) depict the normal quantile–quantile plots for the EEG and MEG data. In this case we observe a clear divergence from the normal distribution for the EEG and MEG residuals. In particular, we see a strong deviation from normality in the left and right tail of the distribution for the EEG data. There is also a deviation from normality in the right tail of the distribution for the MEG data.

In summary, the residual analysis revealed the use of the ACS-ICM algorithm resulted in estimates with a better fit of the spatial mixture model for the EEG and MEG data relative to ICM. Thus our proposed approach leads to improvements in point estimation and model selection uniformly in all settings in simulation studies and in our application with larger objective function values and improved model fit based on residual analysis.

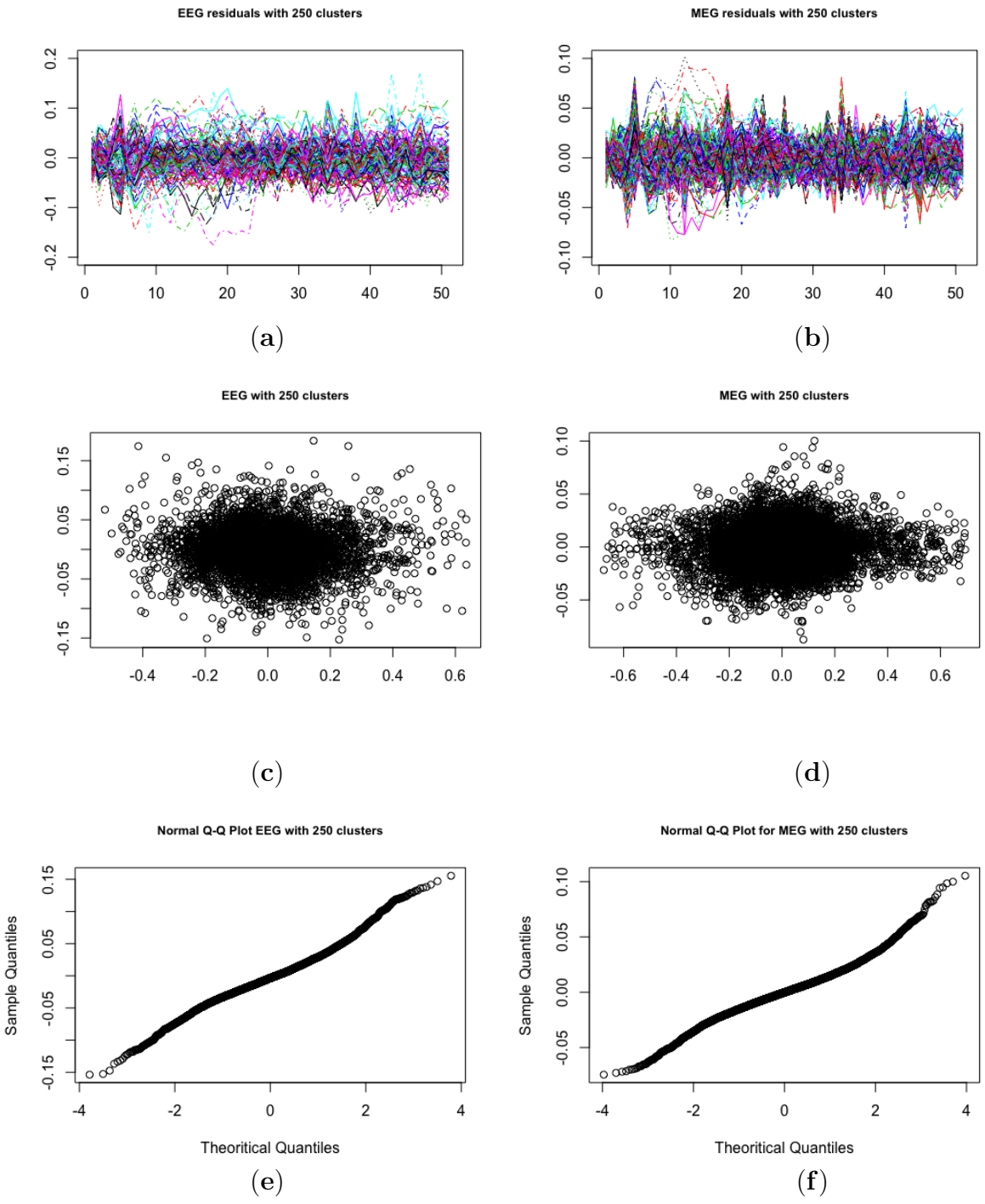


Figure 3.9: Brain activation for scrambled faces using the ACS-ICM algorithm—residual diagnostics: time series of residuals, (**a**) EEG, (**b**) MEG; residuals versus fitted values, (**c**) EEG, (**d**) MEG; residual normal quantile–quantile plots, (**e**) EEG, (**f**) MEG.

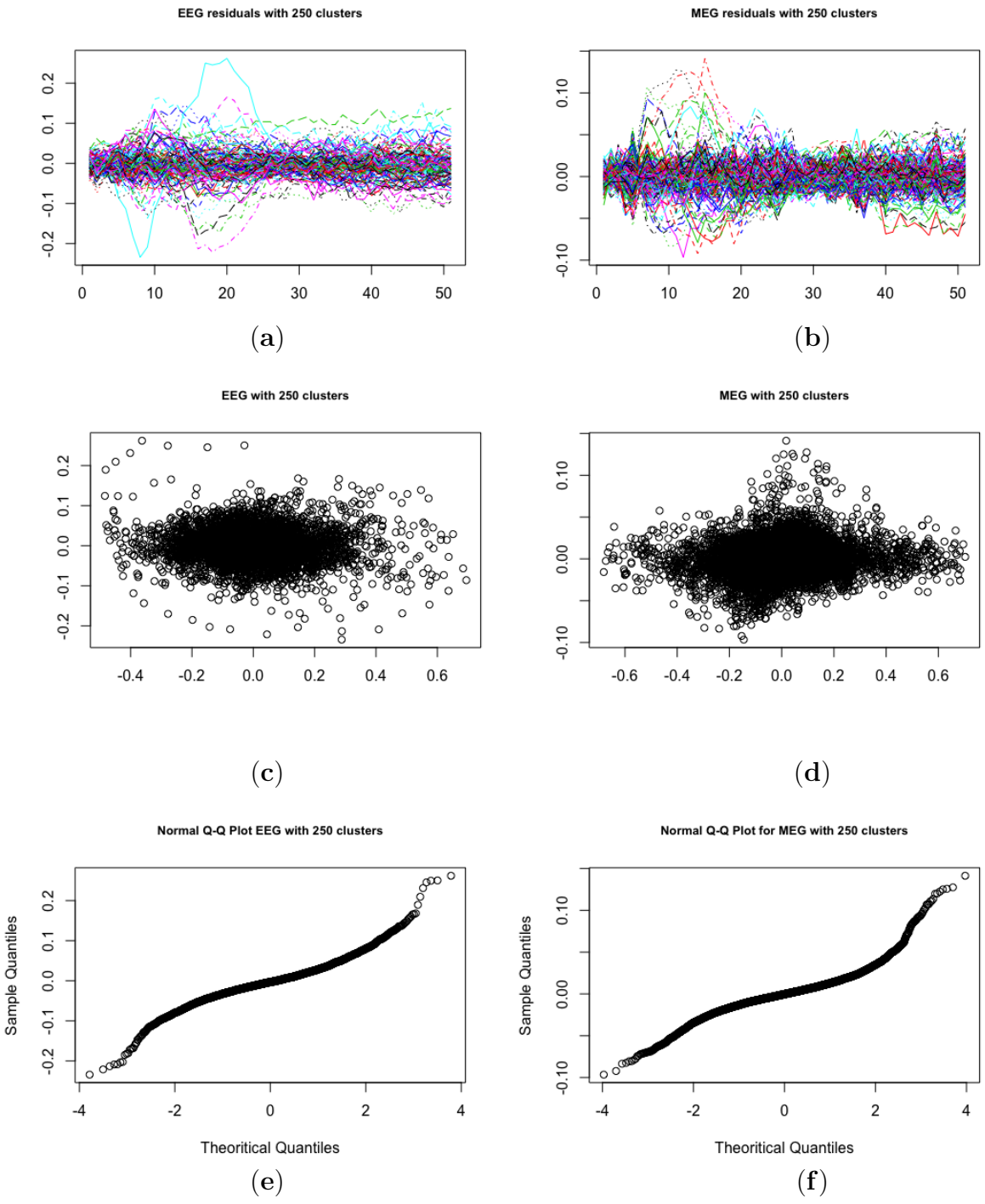


Figure 3.9: Brain activation for scrambled faces using icm algorithm—residual diagnostics: time series of residuals, (**a**) EEG, (**b**) MEG; residuals versus fitted values, (**c**) EEG, (**d**) MEG; residual normal quantile–quantile plots, (**e**) EEG, (**f**) MEG.

3.5 Discussion and Conclusions

In this section, we provide numerical results obtained in the data analysis, limitations of the proposed approach and the prospects for future research. We have developed an ACS-ICM algorithm for spatiotemporal modeling of combined MEG/EEG data for solving the neuroelectromagnetic inverse problem. Adopting a Bayesian finite mixture model with a Potts model as a spatial prior, the focus of our work has been to improve source localization estimates, model selection and model fit. The primary contribution is the design and implementation of the ACS-ICM algorithm as an approach for source localization that result in better performance over ICM, which is very positive uniformly in every setting on simulation studies and real data application. Another key development is the technique implemented in choosing the tuning parameters for the ACS-ICM by using an outer level optimization that numerically optimizes the choice of the tuning parameters for this algorithm. This strategy ensures that the optimal tuning parameters based on the data and problem complexity are selected.

3.5.1 Numerical Results

In our simulation studies, we observed four significant improvements associated with ACS-ICM over ICM: (1) ACS-ICM neural source estimates provided improved correlation between estimated and truth sources uniformly across all settings considered; (2) the objective function values obtained from the posterior density values for ACS-ICM were larger than those obtained from ICM uniformly across all settings considered; (3) ACS-ICM showed significant improvement with respect to the total mean square error for all cluster sizes considered compared to ICM; (4) ACS-ICM exhibited improved performance in terms of both bias and mean square error for the non-regular problem of estimating number of mixture components.

Moreover, the application of ACS-ICM to real data led to higher quality estimates with larger maximized posterior density values. These improvements have demonstrated the advantage of the ACS-ICM algorithm when compared with ICM in both the face perception analysis as well as the simulation studies. In addition to implementing the ACS-ICM algorithm for point estimation, we demonstrated how a nonparametric bootstrap can be used to obtain standard errors, confidence intervals and T-maps for the proposed methodology. This was done to account for uncertainty in our point estimates of the neural source activity.

3.5.2 Limitations of the Proposed Approach

An important limitation of the simulation studies is the use of white noise added to the signals. This is because MEG/EEG data would have structured noise that arise from, e.g., motion, and such noise would be spatially correlated. The spatially correlated noise will make the simulation scenarios more challenging, which we expect to result in a decline in performance. We did not pursue this scenario in our simulation and we will consider it in our future studies.

3.5.3 Prospects for Future Research

In our current work, we are implementing ACS-ICM for the spatial mixture model developed in Song et al. (2019). We hope in the future to extend the model by considering a robust error structure in the MEG/EEG model. The model currently assumes that the errors are independent in time. This will be extended by allowing for an autoregressive structure. A second extension would be to relax the assumption that the errors have a Gaussian distribution by incorporating a multivariate t distribution for the error terms. Integrating these extensions, we will develop a new joint model for the MEG/EEG data and implement the ACS-ICM and ICM algorithms for the neuroelectromagnetic inverse problem.

Furthermore, when we obtained the source estimates from the ACS-ICM algorithm, we mapped a function of them (the total power) on the cortex and in that map we used no thresholding. That is to say, the locations were not thresholded so we can see all the locations with estimated power. For our future studies we hope to map the total power on the cortex with a threshold so that we can see the locations with highest power. In a better way to choose the threshold, our next objective is to extend this work by implementing thresholding of cortical maps using random field theory (Brett et al., 2003). Random field theory is mainly applied in dealing with thresholding problems encountered in functional imaging. This is used to solve the problem of finding the height threshold for a smooth statistical map, which gives the required family-wise error rate. In going forward with our current work, the idea is to take the point estimate obtained from ACS-ICM and standard errors (obtained from bootstrap) to provide estimates of p-values for t-statistics pertaining to the number of activated voxels comprising a particular region.

It should be noted that the ACS-ICM algorithm and spatial model developed can also be applied to studies involving multiple subjects. Expanding from a single sub-

ject model to a model developed for multiple subjects would be of great interest for the MEG/EEG inverse problem. This will be based on developing a fully Bayesian analysis based on a divide and conquer Markov Chain Monte Carlo (MCMC) method (Vehtari et al., 2020). This approach for Bayesian computation with multiple subjects is to partition the data into partitions, perform local inference for each piece separately, and combine the results to obtain a global posterior approximation.

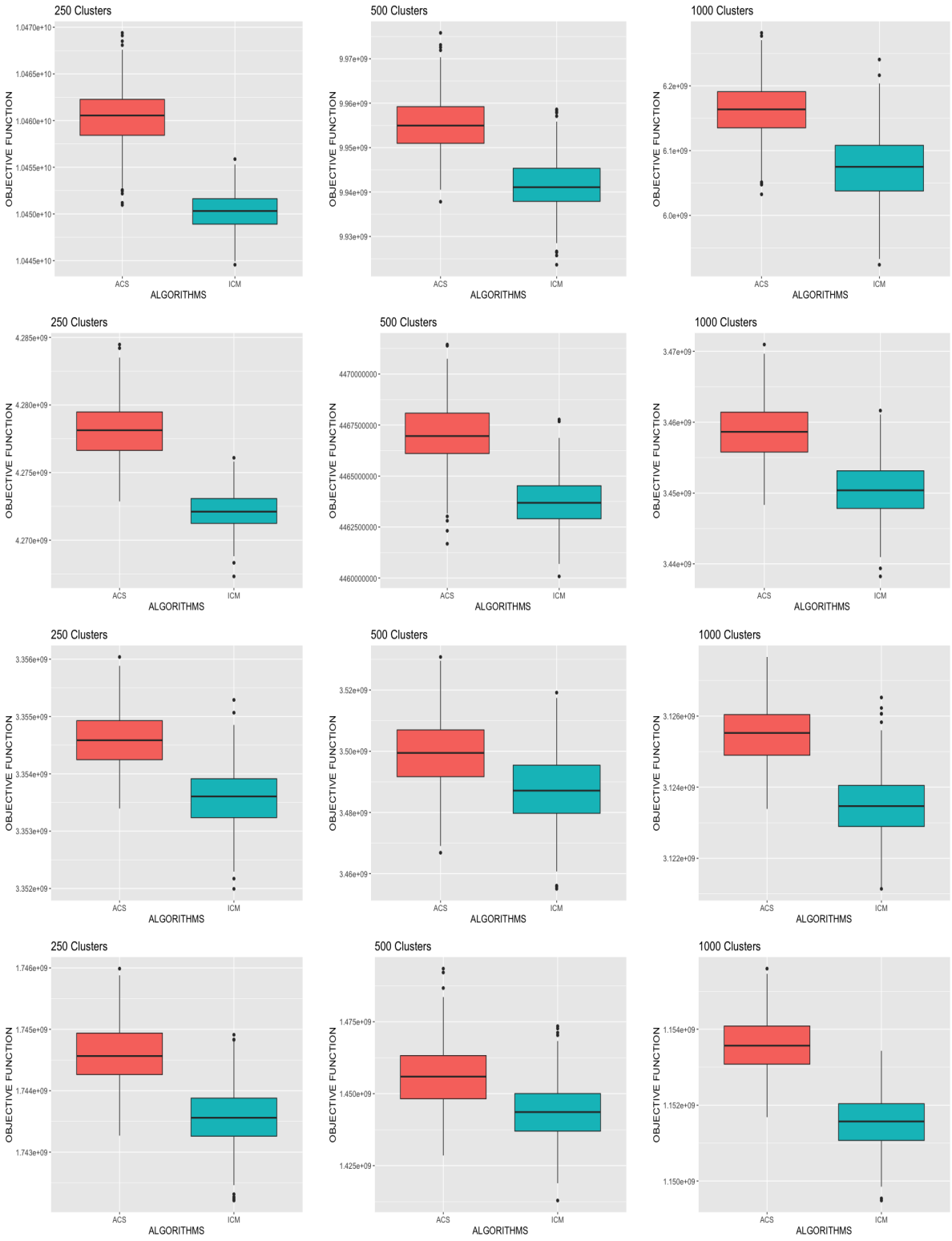


Figure 3.1: Box-plots comparing the objective function values obtained in the simulation studies for the ICM and ACS-ICM algorithms. The first row corresponds to the case when $K = 2$, second row corresponds to when $K = 3$, third row is when $K = 4$ and the last row is when $K = 9$.

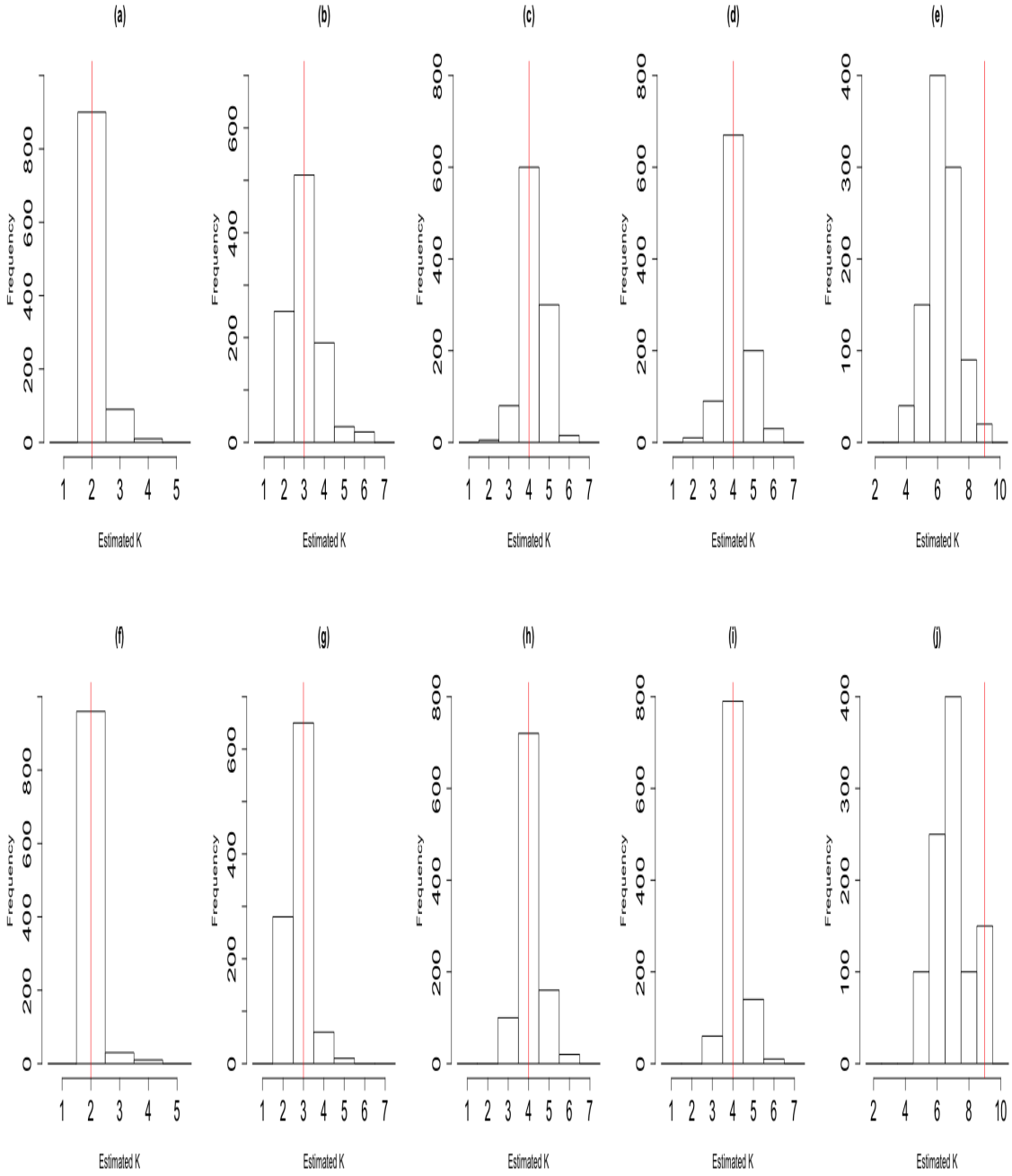


Figure 3.2: Histograms illustrating the sampling distribution of $\hat{K}$ in the case where the true signals were well separated in the simulation studies. The first row corresponds to the sampling distribution of $\hat{K}_{ICM}$; panel (**a**), $K = 2$; panel (**b**), $K = 3$; panel (**c**), $K = 4$ with three Gaussian signals; panel (**d**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**e**), $K = 9$ with eight Gaussian signals. The second row corresponds to the sampling distribution of $\hat{K}_{ACS}$; panel (**f**), $K = 2$; panel (**g**), $K = 3$; panel (**h**), $K = 4$ with three Gaussian signals; panel (**i**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**j**), $K = 9$ with eight Gaussian signals. In each case the vertical red line indicates the true number of latent states underlying the simulated data.

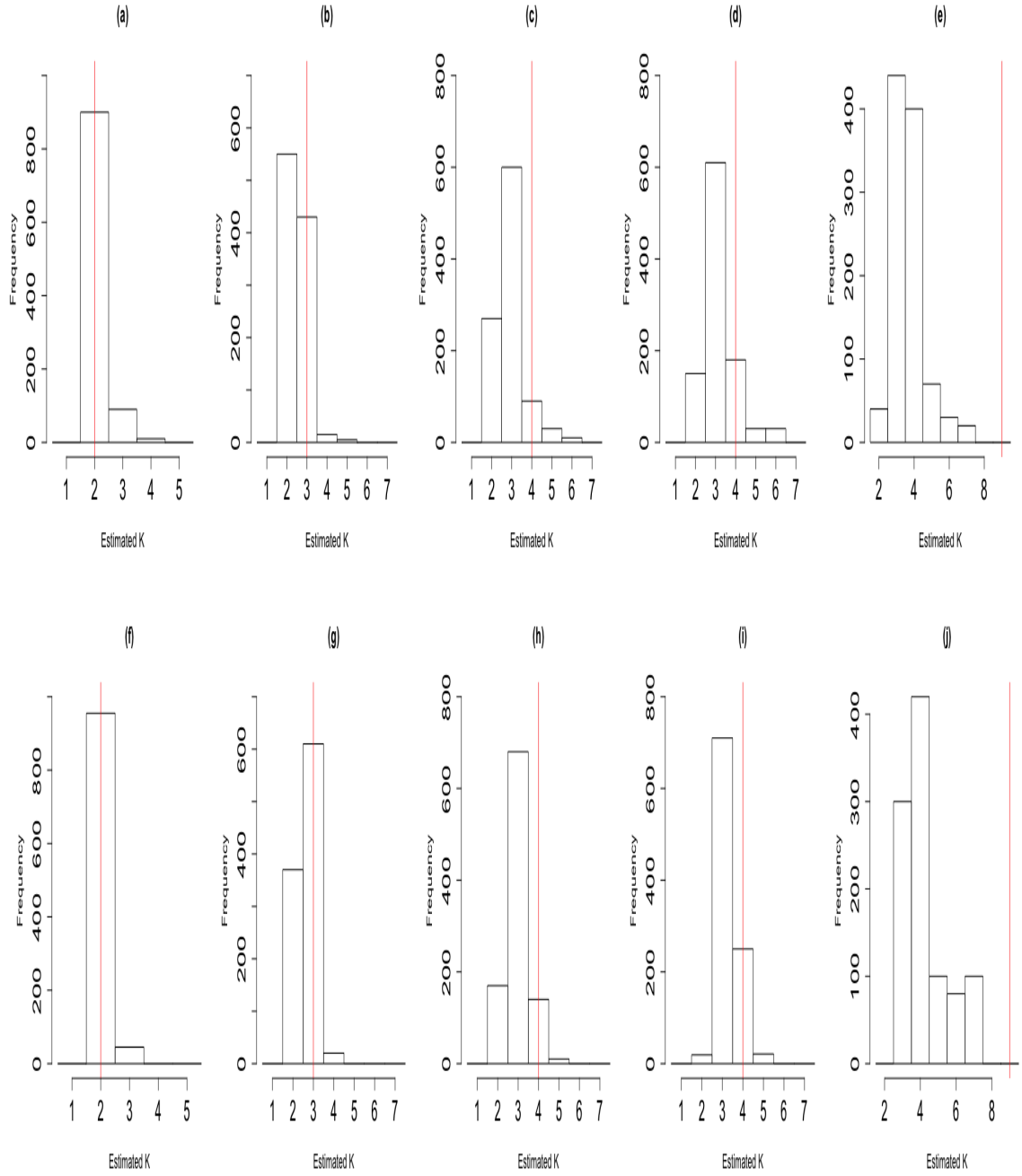


Figure 3.3: Histograms illustrating the sampling distribution of $\hat{K}$ in the case where the true signals were less well-separated in the simulation studies. The first row corresponds to the sampling distribution of $\hat{K}_{ICM}$; panel (**a**), $K = 2$; panel (**b**), $K = 3$; panel (**c**), $K = 4$ with three Gaussian signals; panel (**d**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**e**), $K = 9$ with eight Gaussian signals. The second row corresponds to the sampling distribution of $\hat{K}_{ACS}$; panel (**f**), $K = 2$; panel (**g**), $K = 3$; panel (**h**), $K = 4$ with three Gaussian signals; panel (**i**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**j**), $K = 9$ with eight Gaussian signals. In each case the vertical red line indicates the true number of latent states underlying the simulated data.

Chapter 4

Sparse Estimation Strategies in Linear Mixed Effect Models for High-dimensional Data Application

4.1 Introduction

In many fields such as bio-informatics, physical biology, and epidemiology, the response of interest is represented by repeated measures of some variables of interest that are collected over a specified time period for different independent subjects or individuals. These types of data are commonly encountered in medical research where the responses are subject to various time-dependent and time-constant effects such as pre- and post-treatment types, gender effect, baseline measures among others. A widely-used statistical tool in the analysis and modeling of longitudinal and repeated measures data is the linear mixed effects models (LMM) (Laird and Ware, 1982; Longford, 1993). This model provides an effective and flexible way to describe the means and the covariance structures of a response variable, after accounting for within subject correlation.

The rapid growth in the size and scope of longitudinal data has created a need for innovative statistical strategies in high-dimensional data analysis. Classical methods are based on the assumption that the number of predictors is less than the number of observations. However, there is an increasing demand for efficient prediction strategies for analysis of high-dimensional data, where the number of observed data elements (sample size) are smaller than the number of predictors in a linear model

context. Existing techniques that deal with high-dimensional data mostly rely on various penalized estimators. Due to the trade-off between model complexity and model prediction, the statistical inference of model selection becomes an extremely important and challenging problem in high-dimensional data analysis.

Over the years, many penalized regularization approaches have been developed to do variable selection and estimation simultaneously. Among them, the least absolute shrinkage and selection operator (Lasso) is commonly used (Tibshirani, 1996). It is a useful estimation technique in part due to its convexity and computational efficiency. The Lasso approach is based on an ℓ_1 penalty for regularization of regression parameters. Schelldorfer, Buhlmann, and Van de Geer (2011) have provided a comprehensive summary of the consistency properties of the Lasso approach. Related penalized likelihood methods have been extensively studied in the literature, see for example Tran (2011), Huang and Zhang (2008), Kim, Choi, and Oh (2008), Wang and Leng (2007), Yuan and Lin (2006), Leng, Lin, and Wahba (2006). The penalized likelihood methods have a close connection to Bayesian procedures. Thus the Lasso estimate corresponds to a Bayes method that puts a Laplacian (double-exponential) prior on the regression coefficients (Park and Casella, 2008).

In this paper, our interest lies in estimating the fixed effect parameters of the LMM using ridge estimation technique when it is assumed that some prior information is available in the form of potential linear restrictions on the parameters. Such restrictions can arise in the form of prior information about the fixed effects predictors. One possible source of prior information is using a Bayesian approach. An alternative source of prior information may be obtained from previous studies or expert knowledge that search for sparsity patterns.

We consider the problem of fixed effect parameter estimation for LMMs when there exist many predictors relative to the sample size that may have weak or no effect on the response variable. These predictors may be classified into two groups: sparse and non-sparse. Thus, there are two choices to be considered: a full model with all predictors, and a sub-model that contains only non-sparse predictors. When the sub-model based on available subspace information is true (i.e., the assumed restriction holds), it then provides more efficient statistical inferences than those based on a full model. In contrast, if the sub-model is not true, the estimates could become biased and inefficient. The consequences of incorporating subspace information therefore depend on the quality or reliability of the information being incorporated into the estimation procedure. One way to deal with uncertain subspace information is to

use a pretest estimation strategy. The validity of the information is tested before incorporation into a final estimator. Another approach is shrinkage estimation, which shrinks the full model estimator to a plausible alternative (sub-model estimator) by utilizing subspace information. Besides these estimation strategies, there is a growing literature on simultaneous model selection and estimation. These approaches are known as penalty strategies. By shrinking some regression coefficients toward zero, the penalty methods simultaneously select a sub-model and estimate its regression parameters. Several authors have investigated the pretest, shrinkage, and penalty estimation strategies in partial linear model, Poisson regression model and Weibull censored regression model (see Ahmed and Nicol (2012), Ahmed and Raheem (2012), and Lisawadi, Shah, and Ahmed (2016)).

To formulate the problem, we suppose that the vector of the fixed effects parameter $\boldsymbol{\beta}$ in the LMM can be partitioned into two sub-vectors $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2')'$, where $\boldsymbol{\beta}_1$ is the coefficient vector of non-sparse predictors and $\boldsymbol{\beta}_2$ is the coefficient vector of sparse predictors. Our interest lies in the estimation of $\boldsymbol{\beta}_1$ when $\boldsymbol{\beta}_2$ is close to zero. To deal with this problem in the context of a low dimensional data, Ahmed and Opoku (2017) propose an improved estimation strategy using submodel selection and post-estimation for the LMM. Within this framework, linear shrinkage and shrinkage pretest estimation strategies are develop which combine full model and sub-model estimators in an effective way as a trade-off between bias and variance. Raheem, Ahmed, and Doksum (2012) extend this study by using a likelihood ratio test to develop James-Stein shrinkage and pretest estimation methods based on LMM for longitudinal data. In addition, the non-penalty estimators are compared with several penalty estimators (Lasso, adaptive Lasso and Elastic Net) for best performance.

In most real data situations, there is also the problem of multicollinearity among predictor variables for high-dimensional data. Various biased estimation techniques such as shrinkage estimation, partial Least squares estimation (Geladi and Kowalski, 1986) and Liu estimators (Liu, 2003) have been implemented to deal with this problem but the widely used technique is the ridge estimation (Hoerl and Kennard, 1970). The ridge estimator overcome the weakness of least squares estimator with a smaller mean squared error. To overcome and combat multicollinearity, Yüzbaşı and Ahmed (2016) propose pretest and Stein-type ridge regression estimators for linear and partially linear models. Furthermore, Yüzbaşı and Ahmed (2017) also develop shrinkage estimation based on Liu regression to overcome multicollinearity in linear models.

Our primary focus is on the estimation and prediction problem for linear mixed effect models when there are many potential predictors that have a weak or no influence on the response of interest. This method simultaneously controls overfitting using general least square estimation with a roughness penalty. We propose pretest and shrinkage estimation strategies using the ridge estimation technique as a base estimator and numerically compare their performance with those of two popular Lasso-type estimators: Lasso and adaptive Lasso. Our proposed estimation strategy is applied through application on both low and high-dimensional real data.

4.2 Model and Estimation Strategies

In this section, we present the linear mixed effect model and the proposed estimation strategies.

4.2.1 Linear mixed model

Suppose that we have a sample of N subjects. For the i^{th} subject, we collect the response variable y_{ij} for the jth time, where $i = 1 \ldots, n; j = 1 \ldots, n_i$ and $N = \sum_{i=1}^{n} n_i$. Let $\boldsymbol{Y_i} = (y_{i1}, \ldots y_{in_i})'$ denotes the $n_i \times 1$ vector of responses from the ith subject. Let $\mathbf{X}_i = (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i})'$ and $\mathbf{Z}_i = (\mathbf{z}_{i1}, \ldots, \mathbf{z}_{in_i})'$ be $n_i\times$ p and $n_i\times$q known fixed-effects and random-effect design matrix for the ith subject of full rank p and q respectively. The linear mixed effect model (Laire and Ware, 1982) for a vector of repeated responses $\boldsymbol{Y_i}$ on the ith subject is assumed to have the form

$$\mathbf{Y}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{a}_i + \boldsymbol{\epsilon}_i, \tag{4.1}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)'$ is the p$\times$1 vector of unknown, fixed-effect parameters or regression coefficients, $\boldsymbol{a_i}$ is the q$\times$1 vector of unobservable random effects for the ith subject, assumed to come from a multivariate normal distribution with zero mean and a covariance matrix $\mathbf{G}$, where $\mathbf{G}$ is an unknown $q \times q$ covariance matrix and $\boldsymbol{\epsilon_i}$ denotes $n_i\times$1 vector of error terms assumed to be normally distributed with zero mean, covariance matrix $\sigma^2\mathrm{I}_{n_i}$. Further, $\boldsymbol{\epsilon_i}$ are assumed to be independent of the random effects $\boldsymbol{a_i}$.

The marginal distribution for the response y_i is normal with mean $\mathbf{X}_i\boldsymbol{\beta}$ and covariance matrix $Cov(\boldsymbol{Y}_i) = \boldsymbol{Z_i}\sigma_i^2\boldsymbol{Z_i^T} + \sigma^2\boldsymbol{I_n}$. By stacking the vectors, the mixed

model can be can be expressed as $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{a} + \boldsymbol{\epsilon}$. From the Eqn 4.1, the distribution of the model follows $\boldsymbol{Y} \sim \mathcal{N}_n(\boldsymbol{X}\boldsymbol{\beta}, \hat{\boldsymbol{V}})$, where $E(\boldsymbol{Y}) = \boldsymbol{X}\boldsymbol{\beta}$ with covariance, $\hat{\boldsymbol{V}} = \sum_{i=1}^{m} \boldsymbol{Z_i}\sigma_i^2\boldsymbol{Z_i^T} + \sigma^2 \mathrm{I}_n$.

4.2.2 Ridge full model and sub-model estimator

The generalized least square estimator (GLS) is defined as $\hat{\boldsymbol{\beta}}^{\mathrm{GLS}} = (\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{V}^{-1}\mathbf{Y}$ and the ridge full model estimator can be obtained by introducing a penalized regression so that $\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T\mathbf{V}^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + k\boldsymbol{\beta}^{\boldsymbol{T}}\boldsymbol{\beta} \right\}$ and

$$\hat{\boldsymbol{\beta}}^{\mathrm{Ridge}} = (\mathbf{X}^T\boldsymbol{V}^{-1}\mathbf{X} + k\mathbf{I})^{-1}\mathbf{X}^T\boldsymbol{V}^{-1}\mathbf{Y},$$

where $\hat{\boldsymbol{\beta}}^{\mathrm{Ridge}}$ is the ridge full model estimator and $k \in [0, \infty)$ is the tuning parameter. If k= 0, $\hat{\boldsymbol{\beta}}^{\mathrm{Ridge}}$ is the GLS estimator and $\hat{\boldsymbol{\beta}}^{\mathrm{Ridge}} = 0$ for k sufficiently large. We select the value of k using cross validation.

We let $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)$, where $\mathbf{X}_1$ is an $n \times p_1$ sub-matrix containing the non-sparse predictors and $\mathbf{X}_2$ is an $n \times p_2$ sub-matrix that contain the sparse predictors. Accordingly, $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2)$ where $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ have dimensions p_1 and p_2, respectively, with $p_1 + p_2 = p$, $p_i \geq 0$ for i = 1, 2. A sub-model is defined as
$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{a} + \boldsymbol{\epsilon}$ subject to $\boldsymbol{\beta}^T\boldsymbol{\beta} \leq \phi$ and $\boldsymbol{\beta}_2 = \mathbf{0}$ which corresponds to
$\boldsymbol{Y} = \boldsymbol{X_1}\boldsymbol{\beta_1} + \boldsymbol{Z}\boldsymbol{a} + \boldsymbol{\epsilon}$ subject to ${\boldsymbol{\beta_1}}^T\boldsymbol{\beta_1} \leq \phi$.

The sub-model estimator $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}$ of $\boldsymbol{\beta}_1$ has the form $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} = (\mathbf{X}_1^T\mathbf{V}^{-1}\mathbf{X}_1 + k\mathbf{I})^{-1}\mathbf{X}_1^T\mathbf{V}^{-1}\mathbf{Y}$. We denote $\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}$ as the full model ridge estimator of $\boldsymbol{\beta}_1$ and given as

$$\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} = (\mathbf{X}_1^T\mathbf{V}^{-1/2}\mathrm{M}_{X_2}\mathbf{V}^{-1/2}\mathbf{X}_1 + k\mathrm{I})^{-1}\mathbf{X}_1^T\mathbf{V}^{-1/2}\mathrm{M}_{X_2}\mathbf{V}^{-1/2}\mathbf{Y},$$

where $\mathbf{M}_{X_2} = \mathrm{I} - \mathbf{P} = \mathbf{I} - \mathbf{V}^{-1/2}\mathbf{X}_2(\mathbf{X}_2\mathbf{V}^{-1}\mathbf{X}_2)^{-1}\mathbf{X}_2^T\mathbf{V}^{-1/2}$.

4.2.3 Pretest ridge estimation strategy

Generally, the sub-model estimator will be more efficient than the full-model estimator if the information embodied in the imposed linear restrictions is valid, thus $\boldsymbol{\beta}_2$ is close to zero. However if the information is not valid the sub-model estimator is likely to be more biased and may have a higher risk than the full-model estimator. There is, therefore, some doubt as to whether or not to impose the restrictions on the model's parameter. It is in response to this uncertainty that a statistical test

may be used to determine the validity of the proposed restrictions. Accordingly, the procedure to follow in practice is pre-test the validity of the restrictions and if the outcome of the pretest suggests that they are correct then the model parameters are estimated incorporating the restrictions. If the pre-test rejects the restrictions then the parameters are estimated from the sample information alone. This motivates the consideration of the pretest estimation strategy for the LMM.

The pretest estimator is a combination of the full model estimator $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$ and sub-model estimator $\hat{\boldsymbol{\beta}}_1^{\text{RSM}}$, through an indicator function $\text{I}(\text{L}_n \leq d_{n,\alpha})$, where L_n is an appropriate test statistic to test $H_0 : \boldsymbol{\beta}_2 = \mathbf{0}$ versus $H_A : \boldsymbol{\beta}_2 \neq \mathbf{0}$. Moreover, $d_{n,\alpha}$ is an α level critical value based on distribution of L_n under H_o. We define test statistics based on the log-likelihood ratio test as $\text{L}_n = 2\big\{\ell^*(\hat{\boldsymbol{\beta}}^{\text{RFM}} \mid \mathbf{Y}) - \ell^*(\hat{\boldsymbol{\beta}}^{\text{RSM}} \mid \mathbf{Y})\big\}$.

Under H_0, the the test statistic L_n follows a chi-square distribution with p_2 degrees of freedom. The pretest test ridge estimator $\hat{\boldsymbol{\beta}}_1^{\text{RPT}}$ of $\boldsymbol{\beta}_1$ is then defined by

$$\hat{\boldsymbol{\beta}}_1^{\text{RPT}} = \hat{\boldsymbol{\beta}}_1^{\text{RFM}} - (\hat{\boldsymbol{\beta}}_1^{\text{RFM}} - \hat{\boldsymbol{\beta}}_1^{\text{RSM}})\text{I}(\text{L}_n \leq d_{n,\alpha}), \quad p_2 \geq 1.$$

4.2.4 Shrinkage ridge estimation strategy

The pre-test estimator is a discontinuous function of $\hat{\boldsymbol{\beta}}_1^{\text{RSM}}$ and $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$ and depends on the hard threshold ($d_{n,\alpha} = \chi^2_{p_2,\alpha}$). We address this limitation by defining the shrinkage ridge estimator based on soft thresholding. The shrinkage ridge estimator of $\boldsymbol{\beta}_1$, denoted as $\hat{\boldsymbol{\beta}}_1^{\text{RSE}}$, is defined as

$$\hat{\boldsymbol{\beta}}_1^{\text{RSE}} = \hat{\boldsymbol{\beta}}_1^{\text{RSM}} + (\hat{\boldsymbol{\beta}}_1^{\text{RFM}} - \hat{\boldsymbol{\beta}}_1^{\text{RSM}})(1 - (p_2 - 2)\text{L}_n^{-1}), \quad p_2 \geq 3.$$

Here, $\hat{\boldsymbol{\beta}}_1^{\text{RSE}}$ is the linear combination of the full model $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$ and sub-model $\hat{\boldsymbol{\beta}}_1^{\text{RSM}}$. If $\text{L}_n \leq (p_2 - 2)$, then a relatively large weight is placed on $\hat{\boldsymbol{\beta}}_1^{\text{RSM}}$ otherwise, more weight is on $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$. A setback with $\hat{\boldsymbol{\beta}}_1^{\text{RSE}}$ is that it is not a convex combination of $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$ and $\hat{\boldsymbol{\beta}}_1^{\text{RSM}}$. This can cause over-shrinkage which gives the estimator opposite sign of $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$. This could happen if $(p_2 - 2)\text{L}_n^{-1}$ is larger than one. To counter this, we use the positive-part shrinkage ridge estimator (RPS) defined as

$$\hat{\boldsymbol{\beta}}_1^{\text{RPS}} = \hat{\boldsymbol{\beta}}_1^{\text{RSM}} + (\hat{\boldsymbol{\beta}}_1^{\text{RFM}} - \hat{\boldsymbol{\beta}}_1^{\text{RSM}})(1 - (p_2 - 2)L_n^{-1})^+, p_2 \geq 3$$

where $(1 - (p_2 - 2)\text{L}_n^{-1})^+ = \max(0, 1 - (p_2 - 2)\text{L}_n^{-1})$. The RPS estimator will control possible over-shrinking in the RSE estimator.

4.3 Asymptotic results

In this section, we derive the asymptotic distributional bias and risk of the estimators considered in Section 2. We examine the properties of the estimators for increasing n and as $\boldsymbol{\beta}_2$ approaches the null vector under the sequence of local alternatives defined as

$$K_n : \boldsymbol{\beta}_2 = \boldsymbol{\beta}_{2(n)} = \frac{\boldsymbol{\kappa}}{\sqrt{n}}, \tag{4.2}$$

where $\boldsymbol{\kappa} = (\kappa_1, \kappa_2 \ldots, \kappa_{p_2})' \in \mathbb{R}^{p_2}$ is a fixed vector. The vector $\frac{\boldsymbol{\kappa}}{\sqrt{n}}$ is a measure of how far local alternatives K_n differ from the subspace information $\boldsymbol{\beta}_2 = \mathbf{0}$. In order to evaluate the performance of the estimators, we define the asymptotic distributional bias of an estimator $\hat{\boldsymbol{\beta}}_1^*$

$$\mathrm{ADB}(\hat{\boldsymbol{\beta}}_1^*) = \lim_{n\to\infty} E\big\{\sqrt{n}(\hat{\boldsymbol{\beta}}_1^* - \boldsymbol{\beta}_1)\big\},$$

where $\hat{\boldsymbol{\beta}}_1^*$ is any one of the estimators $\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}$, $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}$, $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}$, $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}$ and $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}$.

Assumption 1

We make the following two regularity conditions to establish the asymptotic properties of the estimators.

1. $\frac{1}{n}\max_{1\le i\le n} \mathbf{x}_i^{\mathrm{T}}\big[\mathbf{X}^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{X}\big]^{-1}\mathbf{x}_i \to \mathbf{0}$ as n$\to\infty$, where $\mathbf{x}_i^{\mathrm{T}}$ is the ith row of $\mathbf{X}$.
2. $\mathbf{B}_n = n^{-1}\big[\mathbf{X}^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{X}\big]^{-1} \to \mathbf{B}$, for some finite $\mathbf{B} = \begin{pmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{pmatrix}$.

Theorem 1.

If $k_n/\sqrt{n} \to \lambda_o$ and $\mathbf{B}$ is non-singular, the distribution of the full model ridge estimator, $\hat{\boldsymbol{\beta}}_n^{\mathrm{RFM}}$ is

$$\sqrt{n}(\hat{\boldsymbol{\beta}}_n^{\mathrm{RFM}} - \boldsymbol{\beta}) \xrightarrow{D} \mathcal{N}(-\lambda_o\mathbf{B}^{-1}\boldsymbol{\beta}, \mathbf{B}^{-1}),$$

where $\xrightarrow{D}$ denotes convergence in distribution.

For proof, see Theorem 2 of Knight and Fu (2000).

Proposition 1.

Under assumption 1 together with Theorem 1, we have

$$\begin{pmatrix} \varphi_1 \\ \varphi_3 \end{pmatrix} \xrightarrow{D} \mathcal{N}\left[\begin{pmatrix} -\boldsymbol{\mu}_{11.2} \\ \boldsymbol{\delta} \end{pmatrix}, \begin{pmatrix} \boldsymbol{B}_{11.2}^{-1} & \boldsymbol{\Phi} \\ \boldsymbol{\Phi} & \boldsymbol{\Phi} \end{pmatrix}\right],$$
$$\begin{pmatrix} \varphi_3 \\ \varphi_2 \end{pmatrix} \xrightarrow{D} \mathcal{N}\left[\begin{pmatrix} \boldsymbol{\delta} \\ -\boldsymbol{\gamma} \end{pmatrix}, \begin{pmatrix} \boldsymbol{\Phi} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B}_{11}^{-1} \end{pmatrix}\right],$$

where $\varphi_1 = \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1)$, $\varphi_2 = \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1)$, $\varphi_3 = \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}})$, $\boldsymbol{\gamma} = \boldsymbol{\mu}_{11.2} + \boldsymbol{\delta}$, $\boldsymbol{\delta} = \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\boldsymbol{\kappa}$, $\Phi = \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\mathbf{B}_{22.1}^{-1}\mathrm{B}_{21}\mathbf{B}_{11}^{-1}$, $\mathbf{B}_{22.1} = \mathbf{B}_{22} - \mathbf{B}_{21}\mathbf{B}_{11}^{-1}\mathbf{B}_{12}$, $\boldsymbol{\mu} = -\lambda_o\mathbf{B}^{-1}\boldsymbol{\beta} = \begin{pmatrix}\boldsymbol{\mu}_1\\ \boldsymbol{\mu}_2\end{pmatrix}$ and $\boldsymbol{\mu}_{11.2} = \boldsymbol{\mu}_1 - \mathbf{B}_{12}\mathbf{B}_{22}^{-1}((\boldsymbol{\beta}_2 - \boldsymbol{\kappa}) - \boldsymbol{\mu}_2)$.

Proof: The relationship between the sub-model and full model estimators of $\boldsymbol{\beta_1}$, we use the argument and equation: $\hat{\mathbf{Y}} = \mathbf{Y} - \mathbf{X}_2\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}$;

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} &= \arg\min_{\boldsymbol{\beta}_1}\left\{(\hat{\mathbf{Y}} - \mathbf{X}_1\boldsymbol{\beta}_1)^{\mathrm{T}}\mathbf{V}^{-1}(\hat{\mathbf{Y}} - \mathbf{X}_1\boldsymbol{\beta}_1) + \lambda||\boldsymbol{\beta}_1||^2\right\}\\
&= \left[\mathbf{X}_1^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{X}_1 + \lambda\mathrm{I}_{p_1}\right]^{-1}\mathbf{X}_1^{\mathrm{T}}\mathbf{V}^{-1}\hat{\mathbf{Y}}\\
&= \left[\mathbf{X}_1^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{X}_1 + \lambda\mathrm{I}_{p_1}\right]^{-1}\mathbf{X}_1^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{Y} - \left[\mathbf{X}_1^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{X}_1 + \lambda\mathrm{I}_{p_1}\right]^{-1}\mathbf{X}_1^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{X}_2\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\\
&= \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \left[\mathbf{X}_1\mathbf{V}^{-1}\mathbf{X}_1 + \lambda\mathrm{I}_{p_1}\right]^{-1}\mathbf{X}_1^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{X}_2\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\\
&= \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}
\end{aligned}$$

From Theorem 1, we partition $\sqrt{n}(\hat{\boldsymbol{\beta}}^{\mathrm{RFM}} - \boldsymbol{\beta})$ as $\sqrt{n}(\hat{\boldsymbol{\beta}}^{\mathrm{RFM}} - \boldsymbol{\beta}) = \big(\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1), \sqrt{n}(\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}} - \boldsymbol{\beta}_2)\big)$. We obtain $\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1) \xrightarrow{D} \mathcal{N}_{p_1}(-\boldsymbol{\mu}_{11.2}, \mathbf{B}_{11.2}^{-1})$, where $\mathbf{B}_{11.2}^{-1} = \mathbf{B}_{11} - \mathbf{B}_{12}\mathbf{B}_{22}^{-1}\mathbf{B}_{21}$. We have shown that $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} = \hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}$. Using this expression and under the local alternative $\{K_n\}$, we get

$$\begin{aligned}
\varphi_2 &= \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1\big)\\
&= \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big)\\
&= \varphi_1 + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\sqrt{n}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\\
\varphi_3 &= \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}})\\
&= \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big) - \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1\big)\\
&= \varphi_1 - \varphi_2
\end{aligned}$$

Since φ_2 and φ_3 are linear functions of φ_1, as $n \to \infty$, they are also asymptotically normally distributed. Their mean vectors and covariance matrices are derived as follows:

$$
\begin{aligned}
E(\varphi_1) &= E\Big(\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big)\Big) = -\boldsymbol{\mu}_{11.2} \\
E(\varphi_2) &= E\Big(\varphi_1 + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\sqrt{n}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\Big) \\
&= E(\varphi_1) + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\sqrt{n}E(\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}) \\
&= -\boldsymbol{\mu}_{11.2} + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\boldsymbol{\kappa} = -(\boldsymbol{\mu}_{11.2} - \boldsymbol{\delta}) = -\boldsymbol{\gamma} \\
E(\varphi_3) &= E(\varphi_1 - \varphi_2) = -\boldsymbol{\mu}_{11.2} - (-(\boldsymbol{\mu}_{11.2} - \boldsymbol{\delta})) = \boldsymbol{\delta} \\
Var(\varphi_1) &= \mathbf{B}_{22.1}^{-1}
\end{aligned}
$$

$$
\begin{aligned}
Var(\varphi_2) &= Var\Big(\varphi_1 + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\sqrt{n}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\Big) \\
&= Var(\varphi_1) + \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\mathbf{B}_{22.1}^{-1}\mathrm{B}_{21}\mathbf{B}_{11}^{-1} \\
&+ 2Cov\Big[\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1), \sqrt{n}(\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}} - \boldsymbol{\beta}_2)\Big](\mathbf{B}_{11}^{-1}\mathbf{B}_{12})^{\mathrm{T}} \\
&= \mathbf{B}_{22.1}^{-1} - \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\mathbf{B}_{22.1}^{-1}\mathrm{B}_{21}\mathbf{B}_{11}^{-1} = \mathbf{B}_{11}^{-1} \\
Var(\varphi_3) &= Var\Big(\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}\big)\Big) \\
&= Var\Big(\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\big)\Big) \\
&= \mathbf{B}_{11}^{-1}\mathbf{B}_{12}Var\Big[\sqrt{n}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\Big](\mathbf{B}_{11}^{-1}\mathbf{B}_{12})^{\mathrm{T}} \\
&= \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\mathbf{B}_{22.1}^{-1}\mathrm{B}_{21}\mathbf{B}_{11}^{-1} = \boldsymbol{\Phi} \\
Cov(\varphi_1, \varphi_3) &= Cov\Big[\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big), \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}\big)\Big] \\
&= Var\Big(\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big)\Big) - Cov\Big[\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big), \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1\big)\Big] \\
&= Var(\varphi_1) - Cov\Big[\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big), \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big) + \sqrt{n}\mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}}\Big] \\
&= \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\mathbf{B}_{22.1}^{-1}\mathrm{B}_{21}\mathbf{B}_{11}^{-1} = \boldsymbol{\Phi}
\end{aligned}
$$

$$
\begin{aligned}
Cov(\varphi_2, \varphi_3) &= Cov\Big[\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1\big), \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}\big)\Big] \\
&= Cov\Big[\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1\big), \sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1\big)\Big] - Var\Big(\sqrt{n}\big(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1\big)\Big) \\
&= \mathbf{B}_{11.2}^{-1} - \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\mathbf{B}_{22.1}^{-1}\mathrm{B}_{21}\mathbf{B}_{11}^{-1} - \mathbf{B}_{11}^{-1} \\
&= \mathbf{B}_{11.2}^{-1} - \big(\mathbf{B}_{11.2}^{-1} - \mathbf{B}_{11}^{-1}\big) - \mathbf{B}_{11}^{-1} = \mathbf{0}
\end{aligned}
$$

Therefore, the asymptotic distributions of the vectors φ_2 and φ_3 are obtained as follows:

$$
\begin{aligned}
\varphi_2 = \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1) &\xrightarrow{D} \mathcal{N}_{p_1}(-\boldsymbol{\gamma}, \mathbf{B}_{11}^{-1}) \\
\varphi_2 = \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}) &\xrightarrow{D} \mathcal{N}_{p_1}(\boldsymbol{\delta}, \boldsymbol{\Phi})
\end{aligned}
$$

We introduce the lemmas given in Judge and Brock (1978) to aid us with the proof of the bias and covariance of the estimators.

Lemma 1.

Let $\mathbf{V} = (\mathrm{V}_1, \mathrm{V}_2, \ldots \mathrm{V}_p)^{\mathrm{T}}$ be a p-dimensional normal vector distributed as $\mathcal{N}_p(\boldsymbol{\mu_v}, \boldsymbol{\Sigma_p})$, then for a measurable function Ψ, we have

$$
\begin{aligned}
E\big[\mathbf{V}\Psi(\mathbf{V}^{\mathrm{T}}\mathbf{V})\big] &= \boldsymbol{\mu}_v E\big[\Psi\chi_{p+2}^2(\Delta)\big] \\
E\big[\mathbf{V}\mathbf{V}^{\mathrm{T}}\Psi(\mathbf{V}^T\mathbf{V})\big] &= \boldsymbol{\Sigma}_p E\big[\Psi\chi_{p+2}^2(\Delta)\big] + \boldsymbol{\mu}_v\boldsymbol{\mu}_v^{\mathrm{T}} E\big[\Psi\chi_{p+4}^2(\Delta)\big]
\end{aligned}
$$

where $\chi_k^2(\Delta)$ is a noncentral chi-square distribution with k degrees of freedom and non-centrality parameter Δ.

Theorem 2

Under the condition of Theorem 1 and the local alternatives K_n, the ADBs of the

proposed estimators are

$$
\begin{aligned}
ADB(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}) &= -\boldsymbol{\mu}_{11.2},\\
ADB(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}) &= -\boldsymbol{\mu}_{11.2} - \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\boldsymbol{\delta} = -\boldsymbol{\gamma},\\
ADB(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}) &= -\boldsymbol{\mu}_{11.2} - \boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2,\alpha};\Delta),\\
ADB(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}) &= -\boldsymbol{\mu}_{11.2} - (p_2-2)\boldsymbol{\delta}E(\chi^{-2}_{p_2+2}(\Delta)),\\
ADB(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}) &= -\boldsymbol{\mu}_{11.2} - \boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2-2};\Delta)\big\} - (p_2-2)\boldsymbol{\delta}E\big\{\chi^{-2}_{p_2+2}(\Delta)\mathrm{I}(\chi^{-2}_{p_2+2} > p_2-2)\big\},
\end{aligned}
$$

where $\Delta = \boldsymbol{\kappa}^{\mathrm{T}}\mathbf{B}^{-1}_{22.1}\boldsymbol{\kappa}$, $\mathbf{B}_{22.1} = \mathbf{B}_{22} - \mathbf{B}_{21}\mathbf{B}_{11}^{-1}\mathbf{B}_{12}$, and $\mathrm{H}_v(x;\Delta)$ is the cumulative distribution function of the non-central chi-squared distribution with non-centrality parameter Δ and v degrees of freedom, and $E(\chi_v^{-2j}(\Delta))$ is the expected value of the inverse of a noncentral χ^2 distribution with v degrees of freedom and noncentrality parameter Δ,

$$E(\chi_v^{-2j}(\Delta)) = \int_0^\infty x^{-2j}d\mathrm{H}_v(x,\Delta).$$

Proof

$$
\begin{aligned}
\mathrm{ADB}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}) &= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1)\big\}\\
&= -\boldsymbol{\mu}_{11.2}.
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{ADB}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}) &= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}} - \boldsymbol{\beta}_1)\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_2^{\mathrm{RFM}} - \boldsymbol{\beta}_1)\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}} - \boldsymbol{\beta}_{\mathbf{1}})\big\} - E\big\{\lim_{n\to\infty}\sqrt{n}(\mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_{\mathbf{2}}^{\mathrm{RFM}})\big\}\\
&= -\boldsymbol{\mu}_{11.2} - E\big\{\lim_{n\to\infty}\sqrt{n}(\mathbf{B}_{11}^{-1}\mathbf{B}_{12}\hat{\boldsymbol{\beta}}_{\mathbf{2}}^{\mathrm{RFM}})\big\}\\
&= -\boldsymbol{\mu}_{11.2} - \mathbf{B}_{11}^{-1}\mathbf{B}_{12}\boldsymbol{\kappa} = -(\boldsymbol{\mu}_{11.2} + \boldsymbol{\delta}) = -\boldsymbol{\gamma}.
\end{aligned}
$$

Using Lemma 1,

$$\begin{aligned}
\mathrm{ADB}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}) &= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}-\boldsymbol{\beta}_{\mathbf{1}})\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})\mathrm{I}(\mathrm{L}_n\le d_{n,\alpha})-\boldsymbol{\beta}_{\mathbf{1}})\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\boldsymbol{\beta}_{\mathbf{1}})\big\}-E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})\mathrm{I}(\mathrm{L}_n\le d_{n,\alpha})\big\}\\
&= -\boldsymbol{\mu}_{11.2}-E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})\mathrm{I}(\mathrm{L}_n\le d_{n,\alpha}))\big\}\\
&= -\boldsymbol{\mu}_{11.2}-\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta).
\end{aligned}$$

$$\begin{aligned}
\mathrm{ADB}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}}) &= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}}-\boldsymbol{\beta}_{\mathbf{1}})\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})(p_2-2)\mathrm{L}_n^{-1}-\boldsymbol{\beta}_{\mathbf{1}})\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\boldsymbol{\beta}_{\mathbf{1}})\big\}-E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})(p_2-2)\mathrm{L}_n^{-1}\big\}\\
&= -\boldsymbol{\mu}_{11.2}-E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})(p_2-2)\mathrm{L}_n^{-1}\big\}\\
&= -\boldsymbol{\mu}_{11.2}-(p_2-2)\boldsymbol{\delta}E(\chi^{-2}_{p_2+2}(\Delta)).
\end{aligned}$$

$$\begin{aligned}
\mathrm{ADB}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPS}}) &= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPS}}-\boldsymbol{\beta}_{\mathbf{1}})\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}}+(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})(1-(p_2-2)\mathrm{L}_n^{-1})\mathrm{I}(\mathrm{L}_n>p_2-2)-\boldsymbol{\beta}_{\mathbf{1}})\big\}\\
&= E\big\{\sqrt{n}\big[\hat{\boldsymbol{\beta}}_{1}^{\mathrm{RSM}}+(\hat{\beta}_{1}^{\mathrm{RFM}}-\hat{\beta}_{1}^{\mathrm{RSM}})(1-\mathrm{I}(\mathrm{L}_n\le p_2-2))\\
&\quad-(\hat{\beta}_{1}^{\mathrm{RFM}}-\hat{\beta}_{1}^{\mathrm{RSM}})(p_2-2)\mathrm{L}_n^{-1}\mathrm{I}(\mathrm{L}_n>p_2-2)-\boldsymbol{\beta}_{1}\big]\big\}\\
&= E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\boldsymbol{\beta}_{\mathbf{1}})\big\}-E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})(p_2-2)\mathrm{I}(\mathrm{L}_n\le p_2-2)\big\}\\
&\quad-E\big\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})(p_2-2)\mathrm{L}_n^{-1}\mathrm{I}(\mathrm{L}_n>p_2-2)\\
&= -\boldsymbol{\mu}_{11.2}-\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2-2};\Delta)\big\}-(p_2-2)\boldsymbol{\delta}E\big\{\chi^{-2}_{p_2+2}(\Delta)\mathrm{I}(\chi^{-2}_{p_2+2}>p_2-2)\big\}.
\end{aligned}$$

Since the ADBs of the estimators are in non-scalar form, we define the following asymptotic quadratic bias (AQDB) of $\hat{\boldsymbol{\beta}}_1^*$ by

$$\mathrm{AQDB}(\hat{\boldsymbol{\beta}}_1^*)=\Big(\mathrm{ADB}(\hat{\boldsymbol{\beta}}_1^*)\Big)'\mathbf{B}_{11.2}\Big(\mathrm{ADB}(\hat{\boldsymbol{\beta}}_1^*)\Big),$$

where $\mathbf{B}_{11.2}=\mathbf{B}_{11}-\mathbf{B}_{12}\mathbf{B}_{22}^{-1}\mathbf{B}_{21}$.

Corollary 1

Suppose Theorem 2 hold. Then, under $\{K_n\}$, the AQDBs of the estimators are

$$\begin{aligned}
\mathrm{AQDB}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}) &= \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2},\\
\mathrm{AQDB}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}) &= \boldsymbol{\gamma}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\gamma},\\
\mathrm{AQDB}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}) &= \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2} + \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta)\\
&\quad + \boldsymbol{\delta}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) + \boldsymbol{\delta}^{T}\mathbf{B}_{11.2}\boldsymbol{\delta}\mathbf{H}^2_{p_2+2}(\chi^2_{p_2};\Delta),\\
\mathrm{AQDB}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}) &= \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2} + (p_2-2)\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\delta}E\big(\chi^{-2}_{p_2+2}(\Delta)\big)\\
&\quad + (p_2-2)\boldsymbol{\delta}^{T}\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2}E\big(\chi^{-2}_{p_2+2}(\Delta)\big) + (p_2-2)^2\boldsymbol{\delta}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\delta}\bigg(E\big(\chi^{-2}_{p_2+2}(\Delta)\big)\bigg)^2,\\
\mathrm{AQDB}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}) &= \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2} + \big(\boldsymbol{\delta}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2} + \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\delta}\big)\big[\mathbf{H}_{p_2+2}(p_2-2;\Delta)\\
&\quad + (p_2-2)E\big\{\chi^{-2}_{p_2+2}(\Delta)\mathrm{I}(\chi^{-2}_{p_2+2}(\Delta) > p_2-2)\big\}\big] + \boldsymbol{\delta}^{\mathrm{T}}\mathbf{B}_{11.2}\boldsymbol{\delta}\bigg[\mathbf{H}_{p_2+2}(p_2-2;\Delta)\\
&\quad + (p_2-2)E\big\{\chi^{-2}_{p_2+2}(\Delta)\mathrm{I}(\chi^{-2}_{p_2+2}(\Delta) > p_2-2)\big\}\bigg]^2.
\end{aligned}$$

When $\mathbf{B}_{11.2} = \mathbf{0}$, the AQDB of all estimators are equivalent, and the estimators are therefore asymptotically unbiased. If we assume that $\mathbf{B}_{11.2} \neq 0$, the results for the bias of the estimators can be summarized as follows

1. The AQDB of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}$ is an unbounded function of $\boldsymbol{\gamma}^T\mathbf{B}_{11.2}\boldsymbol{\gamma}$.
2. The AQDB of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}$ starts from $\boldsymbol{\mu}_{11.2}^T\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2}$ at $\Delta = 0$, and when Δ increases, it increases to the maximum point and then decreases to zero.
3. The characteristics of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}$ and $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}$ are similar to $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}$. The AQDB of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}$ and $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}$ similarly start from $\boldsymbol{\mu}_{11.2}^T\mathbf{B}_{11.2}\boldsymbol{\mu}_{11.2}$ at $\Delta = 0$, and increases to a point, and then decreases towards zero, since $E\big\{\chi^{-2}_{p_2+2}(\Delta)\big\}$ is a non-increasing on of Δ.

In order to compute the risk functions, we first compute the asymptotic covariance of the estimators. The asymptotic covariance of an estimator $\hat{\boldsymbol{\beta}}_1^*$ is expressed as

$$\mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^*) = \lim_{n\to\infty} E\big\{n(\hat{\boldsymbol{\beta}}_1^* - \boldsymbol{\beta}_1)(\hat{\boldsymbol{\beta}}_1^* - \boldsymbol{\beta}_1)^{\mathrm{T}}\big\}.$$

Lemma 2

Suppose Theorem 1 hold, the covariance matrices of the estimators are

$$
\begin{aligned}
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}) &= \mathbf{B}_{11.2}^{-1} + \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{\mathrm{T}},\\
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}}) &= \mathbf{B}_{11}^{-1} + \boldsymbol{\gamma}\boldsymbol{\gamma}^{\mathrm{T}},\\
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}) &= \mathbf{B}_{11.2}^{-1} + \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{\mathrm{T}} + 2\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) - \boldsymbol{\Phi}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta)\\
&\quad + \boldsymbol{\delta}\boldsymbol{\delta}^{\boldsymbol{T}}\big[2\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) - \mathbf{H}_{p_2+4}(\chi^2_{p_2};\Delta)\big],\\
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}}) &= \mathbf{B}_{11.2}^{-1} + \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{T} + 2(p_2-2)\boldsymbol{\mu}_{11.2}^{T}\boldsymbol{\delta}E\Big(\chi^{-2}_{p_2+2}(\Delta)\Big)\\
&\quad - (p_2-2)\boldsymbol{\Phi}\Big\{2E\Big(\chi^{-2}_{p_2+2}(\Delta)\Big) - (p_2-2)E\Big(\chi^{-4}_{p_2+2}(\Delta)\Big)\Big\}\\
&\quad + (p_2-2)\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\Big\{-2E\big(\chi^{-2}_{p_2+4}(\Delta)\big) + 2E(\chi^{-2}_{p_2+2}(\Delta)) + (p_2-2)E\Big(\chi^{-4}_{p_2+4}(\Delta)\Big)\Big\},\\
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{1}^{\mathrm{RPS}}) &= \mathrm{Cov}(\hat{\boldsymbol{\beta}}_{1}^{\mathrm{RSE}}) + 2\boldsymbol{\delta}\boldsymbol{\mu}_{11.2}^{\mathrm{T}}E\bigg(\Big\{1-(p_2-2)\chi^{-2}_{p_2+2}(\Delta)\Big\}\mathrm{I}\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - 2\boldsymbol{\Phi}E\bigg(\Big\{1-(p_2-2)\chi^{-2}_{p_2+2}(\Delta)\Big\}I\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - 2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\big(\{1-(p_2-2)\chi^{-2}_{p_2+4}(\Delta)\}\mathrm{I}(\chi^2_{p_2+4}(\Delta)\le p_2-2)\big)\\
&\quad + 2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\bigg(\Big\{1-(p_2-2)\chi^{-2}_{p_2+2}(\Delta)\Big\}\mathrm{I}\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - (p_2-2)^2\boldsymbol{\Phi}E\bigg(\chi^{-4}_{p_2+2}(\Delta)\mathrm{I}\Big(\chi^2_{p_2+2,\alpha}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - (p_2-2)^2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\bigg(\chi^{-4}_{p_2+2,\alpha}(\Delta)\mathrm{I}\Big(\chi^2_{p_2+2,\alpha}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad + \boldsymbol{\Phi}\mathbf{H}_{p_2+2}\big(p_2-2;\Delta\big) + \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+4}\big(p_2-2;\Delta\big).
\end{aligned}
$$

Proof

We first start by computing the asymptotic covariance of the estimator $\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}$ as:

$$
\begin{aligned}
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}) &= E\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}} - \boldsymbol{\beta}_{\mathbf{1}})\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}} - \boldsymbol{\beta}_{\mathbf{1}})^{\mathrm{T}}\}\\
&= E(\boldsymbol{\varphi}_1\boldsymbol{\varphi}_1^{\mathrm{T}}) = \mathrm{Cov}(\boldsymbol{\varphi}_1\boldsymbol{\varphi}_1^{\mathrm{T}}) + E(\boldsymbol{\varphi}_1)E(\boldsymbol{\varphi}_1^{\mathrm{T}})\\
&= \mathbf{B}_{11.2}^{-1} + \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{\mathrm{T}}.
\end{aligned}
$$

Also, similarly the asymptotic covariance of the estimator $\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}}$ is obtained as:

$$\begin{aligned}\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}}) &= E\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}}-\boldsymbol{\beta}_{\mathbf{1}})\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}}-\boldsymbol{\beta}_{\mathbf{1}})^{\mathrm{T}}\}\\ &= E(\varphi_2\varphi_2^{\mathrm{T}}) = Cov(\varphi_2\varphi_2^{\mathrm{T}}) + E(\varphi_2)E(\varphi_2^{\mathrm{T}})\\ &= \mathbf{B}_{11}^{-1} + \boldsymbol{\gamma}\boldsymbol{\gamma}^{\mathrm{T}}.\end{aligned}$$

The asymptotic covariance of the estimator $\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}$ is obtained as:

$$\begin{aligned}\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}) &= E\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}-\boldsymbol{\beta}_{\mathbf{1}})\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}-\boldsymbol{\beta}_{\mathbf{1}})^{\mathrm{T}}\}\\ &= E\big\{\lim_{n\to\infty} n\big[(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\boldsymbol{\beta}_{\mathbf{1}}) - (\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})I(L_n\le d_{n,\alpha})\big]\\ &\quad\big[(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\boldsymbol{\beta}_{\mathbf{1}}) - (\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}}-\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSM}})I(L_n\le d_{n,\alpha})\big]^{\mathrm{T}}\big\}\\ &= E\Big\{[\varphi_1-\varphi_3 I(L_n\le d_{n,\alpha})][\varphi_1-\varphi_3 I(L_n\le d_{n,\alpha})]^{\mathrm{T}}\Big\}\\ &= E\Big\{\varphi_1\varphi_1^{\mathrm{T}} - 2\varphi_3\varphi_1^{\mathrm{T}}I(L_n\le d_{n,\alpha}) + \varphi_3\varphi_3^{\mathrm{T}}I(L_n\le d_{n,\alpha})\Big\}\end{aligned}$$

Thus, we need to find $E\{\varphi_1\varphi_1^{\mathrm{T}}\}$, $E\{\varphi_3\varphi_1^{\mathrm{T}}I(L_n\le d_{n,\alpha})\}$ and $E\{\varphi_3\varphi_3^{\mathrm{T}}I(L_n\le d_{n,\alpha})\}$. The first term is $E\{\varphi_1\varphi_1^{\mathrm{T}}\} = \mathbf{B}_{11.2}^{-1} + \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{\mathrm{T}}$. Using Lemma 1, the third term is computed as:

$$E\{\varphi_3\varphi_3^{\mathrm{T}}I(L_n\le d_{n,\alpha})\} = \boldsymbol{\Phi}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) + \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+4}(\chi^2_{p_2};\Delta).$$

The second term $E\{\varphi_3\varphi_1^{\mathrm{T}} I(L_n \le d_{n,\alpha})\}$ can be computed from normal theory as

$$\begin{aligned}
E\Big\{\varphi_3\varphi_1^{\mathrm{T}} I(L_n \le d_{n,\alpha})\Big\} &= E\Big\{E\big(\varphi_3\varphi_1^{\mathrm{T}} I(L_n \le d_{n,\alpha})|\varphi_3\big)\Big\} = E\Big\{\varphi_3 E\big(\varphi_1^{\mathrm{T}} I(L_n \le d_{n,\alpha})|\varphi_3\big)\Big\} \\
&= E\{\varphi_3[-\mu_{11.2} + (\varphi_3 - \delta)]^{\mathrm{T}} I(L_n \le d_{n,\alpha})\} \\
&= -E\{\varphi_3\mu_{11.2} I(L_n \le d_{n,\alpha})\} + E\{\varphi_3(\varphi_3 - \boldsymbol{\delta})^{\mathrm{T}} I(L_n \le d_{n,\alpha})\} \\
&= -\boldsymbol{\mu}_{11.2}^{\mathrm{T}} E\{\varphi_3 I(L_n \le d_{n,\alpha})\} + E\{\varphi_3\varphi_3^{\mathrm{T}} I(L_n \le d_{n,\alpha})\} \\
&\quad - E\{\varphi_3\delta^{\mathrm{T}} I(L_n \le d_{n,\alpha})\} \\
&= -\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) + \Big\{Cov(\varphi_3\varphi_3^{\mathrm{T}})\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) \\
&\quad + E(\varphi_3)E(\varphi_3^{\mathrm{T}})\mathbf{H}_{p_2+4}(\chi^2_{p_2};\Delta) - \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta)\Big\} \\
&= -\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) + \boldsymbol{\Phi}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) + \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+4}(\chi^2_{p_2};\Delta) \\
&\quad - \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta)
\end{aligned}$$

Putting all the terms together and simplifying, we obtain

$$\begin{aligned}
&\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RPT}}) \\
&\quad = \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{\mathrm{T}} + 2\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) + \mathbf{B}_{11.2}^{-1} - \boldsymbol{\Phi}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) - \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+4}(\chi^2_{p_2};\Delta) \\
&\quad + 2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) \\
&\quad = \mathbf{B}_{11.2}^{-1} + \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{\mathrm{T}} + 2\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\boldsymbol{\delta}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) - \boldsymbol{\Phi}\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) \\
&\qquad + \boldsymbol{\delta}\boldsymbol{\delta}^{\boldsymbol{T}}\big[2\mathbf{H}_{p_2+2}(\chi^2_{p_2};\Delta) - \mathbf{H}_{p_2+4}(\chi^2_{p_2};\Delta)\big].
\end{aligned}$$

The asymptotic covariance of the estimator $\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}}$ can be obtained by

$$\begin{aligned}
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}}) &= E\{\lim_{n\to\infty}\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}} - \boldsymbol{\beta}_{\mathbf{1}})\sqrt{n}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}} - \boldsymbol{\beta}_{1})^{\mathrm{T}}\} \\
&= E\{\lim_{n\to\infty} n\big[(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \boldsymbol{\beta}_1) - (\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}})(p_2-2)L_n^{-1}\big] \\
&\quad \big[(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RFM}} - \boldsymbol{\beta}_1) - (\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}})(p_2-2)L_n^{-1}\big]^{\mathrm{T}}\} \\
&= E\Big\{[\boldsymbol{\varphi}_1 - \boldsymbol{\varphi}_3(p_2-2)L_n^{-1}][\boldsymbol{\varphi}_1 - \boldsymbol{\varphi}_3(p_2-2)L_n^{-1}]^{\mathrm{T}}\Big\} \\
&= E\Big\{\varphi_1\varphi_1^{\mathrm{T}} - 2(p_2-2)\varphi_3\varphi_1^{\mathrm{T}}L_n^{-1} + (p_2-2)^2\varphi_3\varphi_3^{\mathrm{T}}L_n^{-2}\Big\}
\end{aligned}$$

We need to compute $E\{\varphi_3\varphi_3^{\mathrm{T}}L_n^{-2}\}$ and $E\{\varphi_3\varphi_1^{\mathrm{T}}L_n^{-1}\}$. By using Lemma 1, the first

term is obtained as follows:

$$E\big\{\varphi_3\varphi_3^{\mathrm{T}}L_n^{-2}\big\} = \boldsymbol{\Phi}E\big(\chi^{-4}_{p_2+2}(\Delta)\big) + \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\big(\chi^{-4}_{p_2+4}(\Delta)\big).$$

The second term is computed from normal theory

$$\begin{aligned}
E\bigg\{\varphi_3\varphi_1^{\mathrm{T}}L_n^{-1}\bigg\} &= E\bigg\{E\big(\varphi_3\varphi_1^{\mathrm{T}}L_n^{-1}|\varphi_3\big)\bigg\} = E\bigg\{\varphi_3 E\big(\varphi_1^{\mathrm{T}}L_n^{-1}|\varphi_3\big)\bigg\} \\
&= E\big\{\varphi_3[-\boldsymbol{\mu}_{11.2} + (\varphi_3 - \boldsymbol{\delta})]^{\mathrm{T}}L_n^{-1}\big\} \\
&= -E\big\{\varphi_3\boldsymbol{\mu}_{11.2}L_n^{-1}\big\} + E\big\{\varphi_3(\varphi_3 - \boldsymbol{\delta})^{\mathrm{T}}L_n^{-1}\big\} \\
&= -\boldsymbol{\mu}_{11.2}^{\mathrm{T}}E\{\varphi_3 L_n^{-1}\} + E\{\varphi_3\varphi_3^{\mathrm{T}}L_n^{-1}\} - E\big\{\varphi_3\boldsymbol{\delta}^{\mathrm{T}}L_n^{-1}\big\}
\end{aligned}$$

From above, we can find $E\big\{\varphi_3\boldsymbol{\delta}^{\mathrm{T}}L_n^{-1}\big\} = \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\big(\chi^{-2}_{p_2+2}(\Delta)\big)$ and $E\big\{\varphi_3 L_n^{-1}\big\} = \boldsymbol{\delta}E\big(\chi^{-2}_{p_2+2}(\Delta)\big)$. Putting all together and simplifying we obtain

$$\begin{aligned}
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{\mathrm{RSE}}) &= \mathbf{B}_{11.2}^{-1} + \boldsymbol{\mu}_{11.2}\boldsymbol{\mu}_{11.2}^{T} + 2(p_2-2)\boldsymbol{\mu}_{11.2}^{T}\boldsymbol{\delta}E\Big(\chi^{-2}_{p_2+2}(\Delta)\Big) \\
&\quad - (p_2-2)\boldsymbol{\Phi}\Big\{2E\Big(\chi^{-2}_{p_2+2}(\Delta)\Big) - (p_2-2)E\Big(\chi^{-4}_{p_2+2}(\Delta)\Big)\Big\} \\
&\quad + (p_2-2)\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\Big\{-2E\big(\chi^{-2}_{p_2+4}(\Delta)\big) + 2E(\chi^{-2}_{p_2+2}(\Delta)) + (p_2-2)E\Big(\chi^{-4}_{p_2+4}(\Delta)\Big)\Big\}.
\end{aligned}$$

We derive the covariance of the estimator $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}$. Since
$\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}} = \hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}} - (\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}})\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)$

$$
\begin{aligned}
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}) &= E\Big\{ \lim_{n\to\infty} \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}} - \boldsymbol{\beta}_1)\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}} - \boldsymbol{\beta}_1)^{\mathrm{T}} \Big\} \\
&= E\Bigg\{ \lim_{n\to\infty} \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}} - \boldsymbol{\beta}_1) - \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}})\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}(\mathrm{L}_n \le p_2 - 2) \\
&\times \Big[\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}} - \boldsymbol{\beta}_1) - \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}} - \hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}})\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big]^T \Bigg\} \\
&= E\Big\{ \lim_{n\to\infty} \sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}} - \boldsymbol{\beta}_1)\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}} - \boldsymbol{\beta}_1)^{\mathrm{T}} - 2\boldsymbol{\varphi}_3\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}} - \boldsymbol{\beta}_1)^{\mathrm{T}}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}(\mathrm{I} \\
&+ \boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}^2\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} \\
&= \mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}) - 2E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\sqrt{n}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}} - \boldsymbol{\beta}_1)^{\mathrm{T}}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}^2\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} \\
&+ E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}^2\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} \\
&= \mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}) - 2E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\boldsymbol{\varphi}_1^{\mathrm{T}}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} \\
&+ 2E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}(p_2-2)\mathrm{L}_n^{-1}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} \\
&+ E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}^2\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} \\
&= \mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}) - 2E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\boldsymbol{\varphi}_1^{\mathrm{T}}\big\{1 - (p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} \\
&- E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}(p_2-2)^2\mathrm{L}_n^{-2}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} + E\Big\{ \lim_{n\to\infty} \boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\}
\end{aligned}
$$

We first compute the last term in the equation above $E\Big\{\boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\}$ as $E\Big\{\boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}\mathrm{I}(\mathrm{L}_n \le p_2 - 2)\Big\} = \boldsymbol{\Phi}\mathbf{H}_{p_2+2}(p_2-2;\Delta) + \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+4}(p_2-2;\Delta)$.

Using Lemma 1 and from the normal theory, we find,

$$
\begin{aligned}
&E\Big\{\boldsymbol{\varphi}_3\boldsymbol{\varphi}_1^{\mathrm{T}}\{1-(p_2-2)\mathrm{L}_n^{-1}\}\mathrm{I}(\mathrm{L}_n\le p_2-2)\Big\}\\
&=E\Big\{E\big(\boldsymbol{\varphi}_3\boldsymbol{\varphi}_1^{\mathrm{T}}\{1-(p_2-2)\mathrm{L}_n^{-1}\}\mathrm{I}(\mathrm{L}_n\le p_2-2)|\boldsymbol{\varphi}_3\big)\Big\}\\
&=E\Big\{\boldsymbol{\varphi}_3E\big(\boldsymbol{\varphi}_1^{\mathrm{T}}\{1-(p_2-2)\mathrm{L}_n^{-1}\}\mathrm{I}(\mathrm{L}_n\le p_2-2)|\boldsymbol{\varphi}_3\big)\Big\}\\
&=E\Big\{\boldsymbol{\varphi}_3[\boldsymbol{\mu}_{11.2}+(\boldsymbol{\varphi}_3-\boldsymbol{\delta})]^{\mathrm{T}}\{1-(p_2-2)\mathrm{L}_n^{-1}\}\mathrm{I}(L_n\le p_2-2)\Big\}\\
&=-\boldsymbol{\mu}_{11.2}E\Big(\boldsymbol{\varphi}_3\big\{1-(p_2-2)L_n^{-1}\big\}\mathrm{I}\big(\mathrm{L}_n\le p_2-2\big)\Big)\\
&+E\Big(\boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}\big\{1-(p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}\big(\mathrm{L}_n\le p_2-2\big)\Big)\\
&-E\Big(\boldsymbol{\varphi}_3\boldsymbol{\delta}^{\mathrm{T}}\big\{1-(p_2-2)\mathrm{L}_n^{-1}\big\}\mathrm{I}\big(\mathrm{L}_n\le p_2-2\big)\Big)\\
&=-\boldsymbol{\delta}\boldsymbol{\mu}_{11.2}^{\mathrm{T}}E\bigg(\Big\{1-(p_2-2)\chi_{p_2+2}^{-2}(\Delta)\Big\}\mathrm{I}\Big(\chi_{p_2+2}^{-2}(\Delta)\le p_2-2\Big)\bigg)\\
&+\boldsymbol{\Phi}E\bigg(\Big\{1-(p_2-2)\chi_{p_2+2}^{-2}(\Delta)\Big\}\mathrm{I}\Big(\chi_{p_2+2}^{-2}(\Delta)\le p_2-2\Big)\bigg)\\
&+\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\bigg(\Big\{1-(p_2-2)\chi_{p_2+4}^{-2}(\Delta)\Big\}\mathrm{I}\Big(\chi_{p_2+4}^{-2}(\Delta)\le p_2-2\Big)\bigg)\\
&-\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\bigg(\Big\{1-(p_2-2)\chi_{p_2+4}^{-2}(\Delta)\Big\}\mathrm{I}\Big(\chi_{p_2+4}^{-2}(\Delta)\le p_2-2\Big)\bigg).
\end{aligned}
$$

$$
\begin{aligned}
E\Big\{\boldsymbol{\varphi}_3\boldsymbol{\varphi}_3^{\mathrm{T}}(p_2-2)^2\mathrm{L}_n^{-2}\mathrm{I}(\mathrm{L}_n\le p_2-2)\Big\}&=(p_2-2)^2\boldsymbol{\Phi}E\bigg(\chi_{p_2+2}^{-4}(\Delta)\mathrm{I}\Big(\chi_{p_2+2}^{2}(\Delta)\le p_2-2\Big)\bigg)\\
&+(p_2-2)^2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}E\bigg(\chi_{p_2+2}^{-4}(\Delta)\mathrm{I}\Big(\chi_{p_2+2}^{2}(\Delta)\le p_2-2\Big)\bigg)
\end{aligned}
$$

Putting all the terms together, we obtain

$$
\begin{aligned}
\mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}) = \mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}) &+ 2\boldsymbol{\delta}\boldsymbol{\mu}_{11.2}^{\mathrm{T}} E\bigg(\Big\{1-(p_2-2)\chi_{p_2+2}^{-2}(\Delta)\Big\}\mathrm{I}\Big(\chi_{p_2+2}^2(\Delta)\le p_2-2\Big)\bigg) \\
&- 2\boldsymbol{\Phi} E\bigg(\Big\{1-(p_2-2)\chi_{p_2+2}^{-2}(\Delta)\Big\}\mathrm{I}\Big(\chi_{p_2+2}^2(\Delta)\le p_2-2\Big)\bigg) \\
&- 2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}} E\big(\{1-(p_2-2)\chi_{p_2+4}^{-2}(\Delta)\}\mathrm{I}(\chi_{p_2+4}^2(\Delta)\le p_2-2)\big) \\
&+ 2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}} E\bigg(\Big\{1-(p_2-2)\chi_{p_2+2}^{-2}(\Delta)\Big\}\mathrm{I}\Big(\chi_{p_2+2}^2(\Delta)\le p_2-2\Big)\bigg) \\
&- (p_2-2)^2\boldsymbol{\Phi} E\bigg(\chi_{p_2+2}^{-4}(\Delta)\mathrm{I}\Big(\chi_{p_2+2,\alpha}^2(\Delta)\le p_2-2\Big)\bigg) \\
&- (p_2-2)^2\boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}} E\bigg(\chi_{p_2+2}^{-4}(\Delta)\mathrm{I}\Big(\chi_{p_2+2}^2(\Delta)\le p_2-2\Big)\bigg) \\
&+ \boldsymbol{\Phi}\mathbf{H}_{p_2+2}\big(p_2-2;\Delta\big) + \boldsymbol{\delta}\boldsymbol{\delta}^{\mathrm{T}}\mathbf{H}_{p_2+4}\big(p_2-2;\Delta\big).
\end{aligned}
$$

Following the asymptotic covariance matrix, we define the asymptotic risk of an estimator $\hat{\boldsymbol{\beta}}_1^*$ as $\mathrm{R}(\hat{\boldsymbol{\beta}}_{\mathbf{1}}^{*}) = \mathrm{tr}\Big(\mathbf{Q}\mathrm{Cov}(\hat{\boldsymbol{\beta}}_1^*)\Big)$. $\mathbf{Q}$ is a positive definite matrix of weights with dimensions of $p \times p$. A choice of $\mathbf{Q}$ is the identity matrix $\mathbf{I}$, which is used in this study.

Theorem 2. Under the local alternatives (K_n) and from Lemma 2, the risk of the

estimators are

$$
\begin{aligned}
\mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}\big] &= \mathrm{tr}\Big(\mathbf{Q}\mathbf{B}_{11.2}^{-1}\Big) + \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\mu}_{11.2},\\
\mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}\big] &= \mathrm{tr}\Big(\mathbf{Q}\mathbf{B}_{11}^{-1}\Big) + \boldsymbol{\gamma}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\gamma},\\
\mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}\big] &= \mathrm{tr}\Big(\mathbf{Q}\mathbf{B}_{11.2}^{-1}\Big) + \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\mu}_{11.2} + 2\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\delta}\mathbf{H}_{p_2+2}\Big(\chi^2_{p_2};\Delta\Big)\\
&\quad - \mathrm{tr}\big(\mathbf{Q}\boldsymbol{\Phi}\big)\mathbf{H}_{p_2+2}\Big(\chi^2_{p_2};\Delta\Big) + \boldsymbol{\delta}\mathrm{Q}\boldsymbol{\delta}^{\mathrm{T}}\Big[2\mathbf{H}_{p_2+2}\big(\chi^2_{p_2};\Delta\big) - \mathbf{H}_{p_2+4}\big(\chi^2_{p_2};\Delta\big)\Big],\\
\mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}\big] &= \mathrm{tr}\Big(\mathbf{Q}\mathbf{B}_{11.2}^{-1}\Big) + \boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\mu}_{11.2} + 2(p_2-2)\boldsymbol{\mu}_{11.2}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\delta}E\Big(\chi^{-2}_{p_2+2}(\Delta)\Big)\\
&\quad - (p_2-2)\mathrm{tr}(\mathbf{Q}\boldsymbol{\Phi})\Big[E\Big(\chi^{-2}_{p_2+2}(\Delta)\Big) - (p_2-2)E\Big(\chi^{-4}_{p_2+2}(\Delta)\Big)\Big]\\
&\quad + (p_2-2)\boldsymbol{\delta}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\delta}\Big[2E\Big(\chi^{-2}_{p_2+2}(\Delta)\Big) - 2E\big(\chi^{-2}_{p_2+4}(\Delta)\big) - (p_2-2)E\big(\chi^{-4}_{p_2+4}(\Delta)\big)\Big],\\
\mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}\big] &= \mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}\big] + 2\boldsymbol{\delta}\mathbf{Q}\boldsymbol{\mu}_{11.2}^{\mathrm{T}}E\bigg(\Big\{1-(p_2-2)\chi^{-2}_{p_2+2}(\Delta)\Big\}\mathrm{I}\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - 2\mathrm{tr}(\mathbf{Q}\boldsymbol{\Phi})E\bigg(\Big\{1-(p_2-2)\chi^{-2}_{p_2+2}(\Delta)\Big\}\mathrm{I}\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - 2\boldsymbol{\delta}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\delta}E\big(\{1-(p_2-2)\chi^{-2}_{p_2+4}(\Delta)\}\mathrm{I}(\chi^2_{p_2+4}(\Delta)\le p_2-2)\big)\\
&\quad + 2\boldsymbol{\delta}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\delta}E\bigg(\Big\{1-(p_2-2)\chi^{-2}_{p_2+2}(\Delta)\Big\}\mathrm{I}\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - (p_2-2)^2\mathrm{tr}(\mathbf{Q}\boldsymbol{\Phi})E\bigg(\chi^{-4}_{p_2+2}(\Delta)\mathrm{I}\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad - (p_2-2)^2\boldsymbol{\delta}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\delta}E\bigg(\chi^{-4}_{p_2+2}(\Delta)\mathrm{I}\Big(\chi^2_{p_2+2}(\Delta)\le p_2-2\Big)\bigg)\\
&\quad + \mathrm{tr}(\mathbf{Q}\boldsymbol{\Phi})\mathbf{H}_{p_2+2}\big(p_2-2;\Delta\big) + \boldsymbol{\delta}^{\mathrm{T}}\mathbf{Q}\boldsymbol{\delta}\mathbf{H}_{p_2+4}\big(p_2-2;\Delta\big).
\end{aligned}
$$

From Theorem 2, when $\mathbf{B}_{12}=\mathbf{0}$, the risks of estimators $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}$, $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}$, $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSE}}$, and $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPS}}$ are reduced to common value $\mathrm{tr}(\mathbf{Q}\mathbf{B}_{11.2}^{-1})+\boldsymbol{\mu}_{11.2}^{T}\mathbf{Q}\boldsymbol{\mu}_{11.2}$, the risk of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}$. If $\mathbf{B}_{12}\neq\mathbf{0}$, the results can be summarized as follows:

1. The risk of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}$ remains constant while the risk of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RSM}}$ is an unbounded function of Δ since $\Delta\in[0,\infty)$.

2. The risk of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}$ increases as Δ moves away from zero, achieves it maximum and then decreases towards the risk of the full model estimator.

3. The risk of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}$ is smaller than the risk of $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}$ for small values in the neighborhood of Δ and for the rest of the parameter space, $\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}$ outperforms $\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}$, thus, $\mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RFM}}\big] > \mathrm{R}\big[\hat{\boldsymbol{\beta}}_1^{\mathrm{RPT}}\big]$.

4. Comparing the risks of $\hat{\boldsymbol{\beta}}_1^{\text{RSE}}$ and $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$, it can be seen that the estimator $\hat{\boldsymbol{\beta}}_1^{\text{RSE}}$ outperforms $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$ that is, $\text{R}\big[\hat{\boldsymbol{\beta}}_1^{\text{RSE}}\big] \leq \text{R}\big[\hat{\boldsymbol{\beta}}_1^{\text{RFM}}\big]$ for all $\Delta \geq 0$.

4.4 Simulation Studies

In this section, we conduct a simulation study to assess the performance of the suggested estimators under consideration. The criterion for comparing the performance of any estimator in our study is the mean square error. We simulate the response from the following LMM model

$$\boldsymbol{Y}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{a}_i + \boldsymbol{\epsilon}_i,$$

where $\boldsymbol{\epsilon_i} \sim \mathcal{N}(0, \sigma^2\text{I}_{n_i})$ with $\sigma^2 = 1$. We generate random effect covariate $\mathbf{a}_i$ from a multivariate normal distribution with zero mean and covariance matrix $\mathbf{G} = 0.5\text{I}_{2\times 2}$, where $\text{I}_{2\times 2}$ is 2×2 identity matrix. The design matrix $\mathbf{X}_i = (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i})'$ is generated from a n_i-multivariate normal distribution with mean vector and covariance matrix Σ_x. Furthermore, we consider the off-diagonal elements of the covariance matrix Σ_x are equal to ρ, which is the coefficient of correlation between any two predictors, with $\rho = 0.3, 0.7, 0.9$. The ratio of the largest eigenvalue to the smallest eigenvalue of matrix $\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X}$ is calculated as a condition number index (CNI; Besley, 1991) which assess the existence of multicollinearity in the design matrix. If the condition number is larger than 30, then the model have significant multicollinearity. Our simulations are based on the linear mixed effects model in (1) with $n = 60$ and 100 subjects.

We consider a situation when the model is assumed to be sparse. In this study our interest lies in testing the hypothesis $H_o : \boldsymbol{\beta}_2 = \mathbf{0}$, and our goal is to estimate the fixed effect coefficient $\boldsymbol{\beta}_1$. We partition the fixed effects coefficients as $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2')' = (\boldsymbol{\beta}_1', \mathbf{0}_{p_2})'$. The coefficients $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are p_1 and p_2 dimensional vectors respectively, with $p = p_1 + p_2$.

In order to investigate the behavior of the estimators, we define $\Delta^* = ||\boldsymbol{\beta} - \boldsymbol{\beta}_o||$, where $\boldsymbol{\beta}_o = (\beta_1^T, \mathbf{0}_{p_2})^{\text{T}}$ and $||.||$ is the euclidean norm. We considered Δ values between 0 and 4. If $\Delta^* = 0$, then we will have $\boldsymbol{\beta} = (1, 1, 1, 1, \underbrace{0, 0, \ldots, 0}_{p_2})^{\text{T}}$ to generate the response under null hypothesis. On the other hand, when $\Delta^* \geq 0$, say $\Delta = 4$, we will have $\boldsymbol{\beta} = (1, 1, 1, 1, 4, \underbrace{0, 0, \ldots, 0}_{p_2-1})^{\text{T}}$ to generate the response under the local alternative hypothesis. In our simulation study, we consider the number of fixed

effect or predictor variables: $(p_1, p_2) \in \{(5, 40), (5, 500), (5, 1000)\}$. Each realization is repeated 5000 times to obtain consistent results and compute the MSE of suggested estimators with $\alpha = 0.05$.

Based on the simulated data, we calculate the mean square error (MSE) of all the estimators as $\text{MSE}(\hat{\boldsymbol{\beta}}) = \frac{1}{5000}\sum_{j=1}^{5000}(\hat{\beta} - \beta)(\hat{\beta} - \beta)$, where $\hat{\boldsymbol{\beta}}$ denotes any one of $\hat{\beta}^{\text{RSM}}, \hat{\beta}^{\text{RPT}}, \hat{\beta}^{\text{RSE}}$ and $\hat{\beta}^{\text{RPS}}$, in the jth repetition. We use the relative mean squared efficiency (RMSE), or the ratio of MSE for risk performance comparison. The RMSE of an estimator $\hat{\boldsymbol{\beta}}^*$ with respect to the baseline estimator $\hat{\boldsymbol{\beta}}_1^{\text{RFM}}$ is defined as $\text{RMSE}(\hat{\boldsymbol{\beta}}_1^{\text{RFM}} : \hat{\boldsymbol{\beta}}_1^*) = \dfrac{\text{MSE}(\hat{\boldsymbol{\beta}}_1^{\text{RFM}})}{\text{MSE}(\hat{\boldsymbol{\beta}}_1^*)}$, where $\boldsymbol{\beta}_1^*$ is one of the suggested estimators under consideration.

4.4.1 Simulation results

In this subsection, we present the results from our simulation study. We report the results for $n = 60, 100$ and $p_1 = 5$ with different values of ρ are shown in Table 4.1. Furthermore, we plot the RMSEs against Δ^* for easier comparison in Figures 4.1 and 4.2. The findings can be summarized as follows:

1. When $\Delta^* = 0$, the sub-model RSM outperforms all other estimators. As $\Delta^* = 0$ moves from zero, the RMSE of the sub-model decreases and goes to zero.

2. The pretest ridge estimator RPT outperforms shrinkage ridge and positive Stein ridge estimators in the case of $\Delta^* = 0$. However, for large p_2 values while keeping p_1 and n fixed, RPT is less efficient than RPS and RSE. In the case of Δ^* being larger than zero, the RMSE of RPT decreases, and it remains below 1 for immediate values of Δ^*, after that the RMSE of RPT increases and approaches one for larger values of Δ^*.

3. RPS performs better than RSE in the entire parameter space induced by Δ^* as presented in Table 4.1 and 4.2. Similarly, both shrinkage estimators RPS and RSE outperforms the full ridge model estimator irrespective of the corrected sub-model selected. This is consistent with the asymptotic theory presented in Section 3.

4. Δ^* which measures the degree of deviation from the assumption 1 on the parameter space, it is clear that one cannot go wrong with the use of shrinkage

estimators even if the selected sub-model is wrongly specified. As evident from Tables 4.1 & 4.2, Figures 4.1 & 4.2, if the selected sub-model is correct, that is $\Delta^* = 0$ then the shrinkage estimators are relatively efficient compared with the ridge full model estimator. On the other hand, if the sub-model is misspecified, the gain slowly diminishes. However, in terms of risk, the shrinkage estimators are at least good as the full ridge model estimator. Therefore the use of shrinkage estimators make sense in real-life application when a sub-model cannot be correctly specified.

5. The RMSE of the ridge-type estimators are an increasing function of the amount of multicollinearity. This indicates the ridge-type estimators perform better than classical estimator in the presence of multicollinearity among predictor variables.

4.4.2 Comparison with Lasso-type estimators

We compare our listed estimators with penalty estimators namely Lasso and adaptive Lasso. A 10-fold cross-validation is used for selecting the optimal value of the penalty parameters that minimizes the mean square errors for the Lasso and adaptive Lasso estimators. The results for $\rho = 0.3, 0.7, 0.9$, $n = 60, 100$, $p_1 = 10$ and $p_2 = 50, 500, 5000, 1000$ are presented in Table 4.3. We observe the following from Table 4.3.

1. The performance of the sub-model estimator is the best among all estimators.

2. The pretest ridge estimator performs better than the other estimators. However, for larger values of p_2 the shrinkage ridge estimators outperforms the pretest estimator.

3. The performance of the Lasso and aLasso estimators are comparable when ρ is small. The pretest and shrinkage estimators remain stable for a given value of ρ.

4. For large values of p_2, the shrinkage and pretest estimators outperforms the lasso-type estimators. This indicates the superiority of the shrinkage estimators over ℓ_1 penalty estimators. Therefore shrinkage estimators are preferable when there is multicollinearity in our predictor variables.

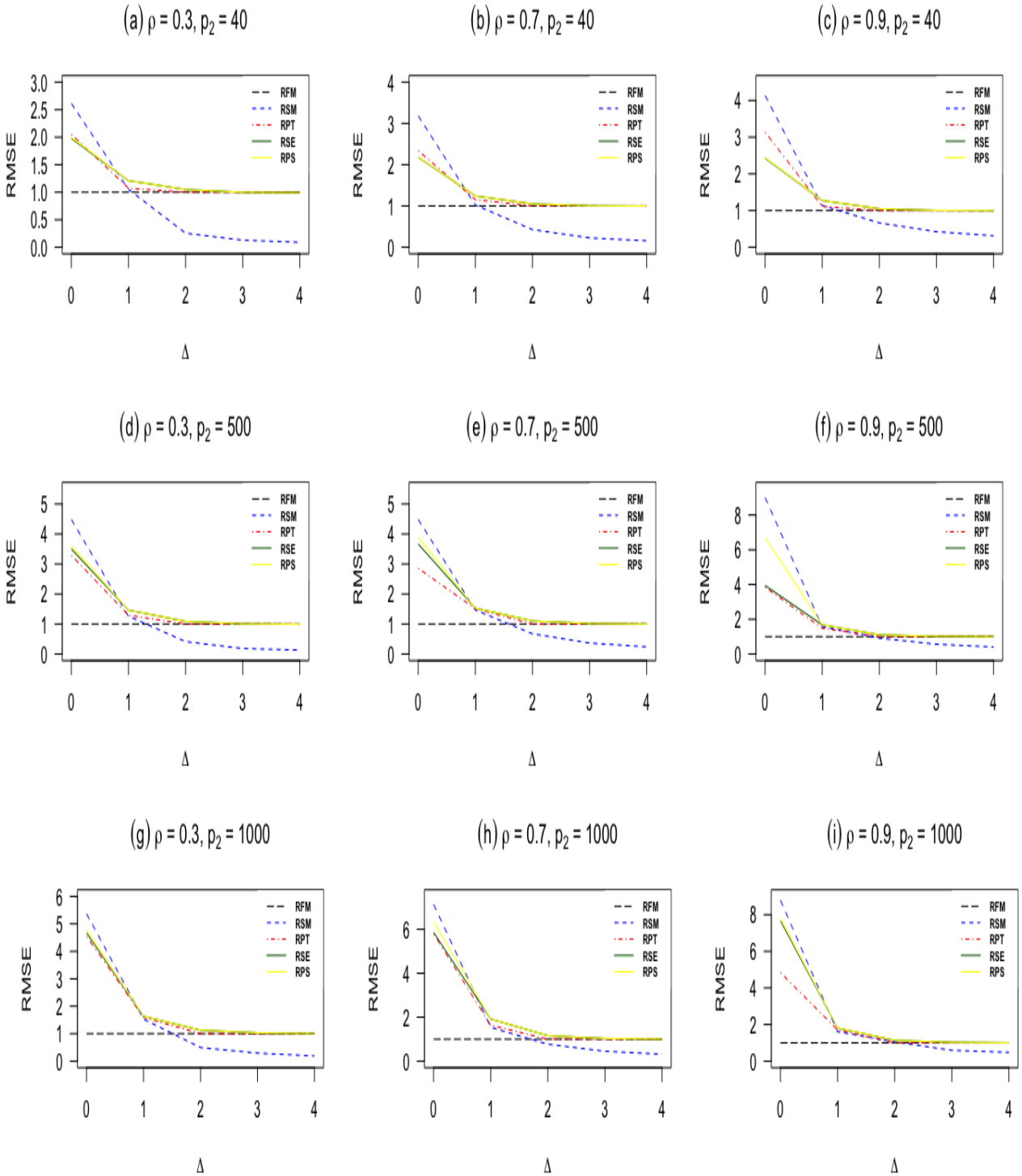


Figure 4.1: RMSE of estimators as a function of the non-centrality parameter Δ when n=60, and $p_1 = 5$.

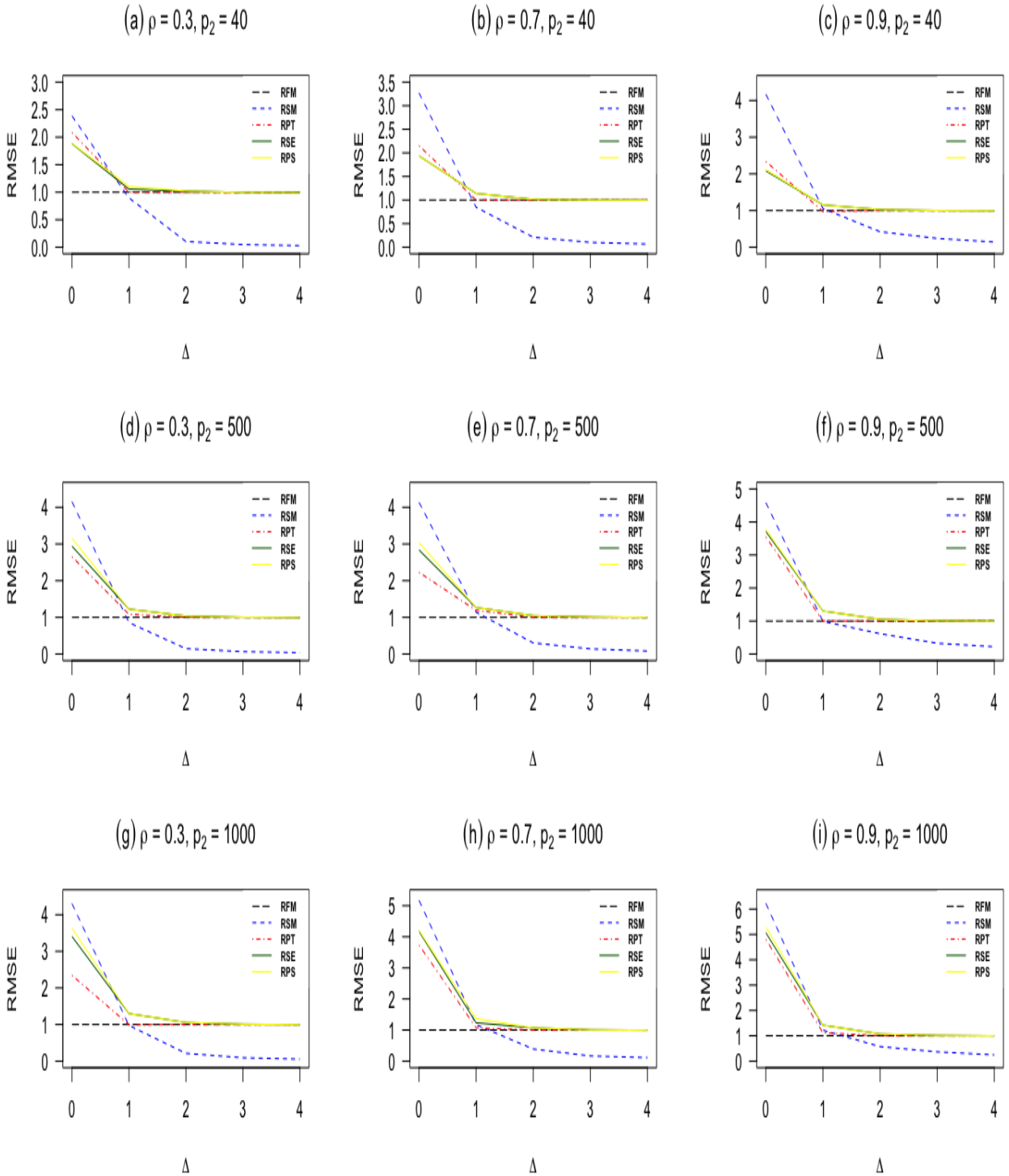


Figure 4.2: RMSE of estimators as a function of the non-centrality parameter Δ when n=100, and $p_1 = 5$.

Table 4.1: RMSEs of RSM, RPT, RSE, and RPS estimators with respect to $\hat{\beta}^{\mathrm{RFM}}$ when $\Delta \geq 0$ for $p_1 = 5$ and $n = 60$.

ρ	p_2	Δ	CNT	RSM	RPT	RSE	RPS
0.3	50	0	361	2.61	2.07	1.94	1.96
		1		1.05	1.07	1.20	1.25
		2		0.25	0.95	1.04	1.05
		3		0.12	0.98	0.99	1.00
		4		0.08	1.00	1.00	1.00
	500	0	613	4.48	3.29	3.48	1.96
		1		1.26	1.12	1.26	1.29
		2		0.41	0.97	1.08	1.09
		3		0.18	0.99	1.00	1.00
		4		0.13	1.00	1.00	1.00
	1000	0	693	5.36	4.53	4.67	4.71
		1		1.53	1.21	1.35	1.39
		2		0.49	1.01	1.13	1.14
		3		0.28	0.99	0.99	0.99
		4		0.10	1.00	1.00	1.00
0.7	50	0	1352	3.18	2.33	2.17	2.18
		1		1.04	1.11	1.20	1.23
		2		0.42	1.03	1.04	1.04
		3		0.23	0.98	0.99	1.00
		4		0.14	1.00	1.00	1.00
	500	0	1789	4.48	2.76	2.94	3.02
		1		1.08	1.43	1.52	1.53
		2		0.67	1.03	1.07	1.06
		3		0.35	0.98	1.00	1.00
		4		0.19	1.00	1.00	1.00
	1000	0	2134	6.82	5.24	5.30	3.02
		1		1.16	1.32	1.42	1.53
		2		0.75	1.10	1.15	1.16
		3		0.39	0.99	1.00	1.00
		4		0.11	1.00	1.00	1.00

Table 4.2: RMSEs of RSM, RPT, RSE, and RPS estimators with respect to $\hat{\beta}^{\mathrm{RFM}}$ when $\Delta \geq 0$ for $p_1 = 5$, and $n = 100$.

ρ	p_2	Δ	CNT	RSM	RPT	RSE	RPS
0.3	50	0	150	2.38	2.09	1.88	1.90
		1		0.89	1.01	1.05	1.08
		2		0.21	0.94	1.01	1.02
		3		0.06	0.94	0.99	1.00
		4		0.02	1.00	1.00	1.00
	500	0	340	4.15	2.65	2.99	3.17
		1		0.87	1.08	1.18	1.21
		2		0.14	0.96	1.03	1.05
		3		0.06	0.99	0.99	1.00
		4		0.03	1.00	1.00	1.00
	1000	0	536	4.30	2.75	3.02	3.08
		1		0.96	1.09	1.13	1.15
		2		0.21	0.8	1.03	1.03
		3		0.09	1.00	1.00	1.00
		4		0.04	1.00	1.00	1.00
0.7	50	0	997	3.27	2.15	2.09	2.11
		1		0.85	1.02	1.09	1.10
		2		0.21	0.98	1.02	1.02
		3		0.06	0.99	0.99	0.99
		4		0.01	1.00	1.00	1.00
	500	0	1589	4.13	2.22	2.35	2.39
		1		1.04	1.19	1.21	1.20
		2		0.30	0.97	1.05	1.05
		3		0.14	1.00	1.00	1.00
		4		0.08	1.00	1.00	1.00
	1000	0	1751	5.17	3.71	4.03	4.09
		1		1.01	1.15	1.24	1.25
		2		0.39	1.04	1.07	1.06
		3		0.16	0.99	1.00	1.00
		4		0.11	1.00	1.00	1.00

Table 4.3: RMSEs of estimators with respect to $\hat{\beta}^{\mathrm{RFM}}$ when $\Delta = 0$ for $p_1 = 10$.

n	ρ	p_2	CNT	RSM	RPT	RSE	RPS	LASSO	aLASSO
60	0.3	50	35.64	3.31	2.25	1.82	1.95	1.23	1.28
		500	452.76	4.13	3.71	2.61	3.01	1.47	1.52
		1000	1265.34	5.02	4.28	4.61	4.78	1.96	2.15
		2000	4567.56	7.13	5.10	6.18	6.39	2.70	3.06
	0.7	50	61.34	3.52	3.05	2.51	2.55	1.14	1.21
		500	743.17	4.49	3.65	3.41	3.50	1.36	1.58
		1000	2350.89	5.84	4.11	4.32	4.61	1.68	1.95
		2000	6908.39	8.10	5.31	6.24	6.29	1.84	2.02
	0.9	50	120.21	4.21	3.61	3.34	3.35	1.10	1.05
		500	950.98	4.82	3.3.8	3.72	3.73	1.21	1.16
		1000	5892.51	6.35	4.10	5.01	5.13	1.42	1.31
		2000	8352.73	8.51	4.63	5.24	5.38	1.61	1.35
100	0.3	50	31.21	2.91	2.54	2.12	2.23	1.32	1.36
		500	356.64	3.75	3.31	2.84	2.92	1.54	1.61
		1000	975.32	4.25	2.53	3.42	3.61	1.92	2.06
		2000	2764.84	5.61	4.25	4.91	5.08	2.31	2.46
	0.7	50	52.79	3.18	2.61	2.30	2.37	1.28	1.53
		500	578.43	4.28	3.05	3.52	3.59	1.46	2.07
		1000	1281.66	5.10	3.26	3.78	3.82	1.84	2.52
		2000	3498.30	6.12	3.01	4.26	4.33	2.27	2.41
	0.9	50	79.41	4.11	3.41	3.21	3.28	1.28	1.21
		500	681.43	4.35	3.55	3.41	3.50	1.43	1.51
		1000	1470.32	5.82	3.18	4.01	4.14	1.72	1.79
		2000	4105.90	7.04	4.57	5.22	5.32	1.87	1.96

4.5 Real data application

We consider two real data analyses using Amsterdam Growth and Health Data and a genetic and brain network connectivity edge weight data to illustrate the performance of the proposed estimators.

4.5.1 Amsterdam Growth and Health Data (AGHD)

The AGHD data is obtained from the Amsterdam Growth and Health Study (Kemper, 1995). The goal of this study was to investigate the relationship between lifestyle and health in adolescence into young adulthood. The response variable Y is the total serum cholesterol measured over six time points. There are five covariates: X_1 is the baseline fitness level measured as the maximum oxygen uptake on a treadmill, X_2 is the amount of body fat estimated by the sum of the thickness of four skinfolds, X_3 is a smoking indicator (0=no, 1=yes), X_4 is the gender (1=female, 2=male), and time measurement as X_5.

A total of 147 subjects participated in the study where all variables were measured at $n_i = 6$ time occasions. In order to apply the proposed methods, we use a two step approach since we have no uncertain prior information (UPI) owing to acquaintance with the data.

Firstly, we apply a variable selection based on AIC procedure to select the sub-model. For the AGHD data, we fit a linear mixed model with all the five covariates for both fixed and random effects by two stage selection procedure for the purpose of choosing both the random and fixed effects. The analysis found X_2 and X_5 to be significant covariates for prediction of the response variable serum cholestrol and the other variables is ignored since they are not significantly important. Based on this information, a submodel is chosen to be X_2 and X_5 and the full model includes all the covariates. We construct the shrinkage estimators from the full-model and sub-model. In terms of null hypothesis, the restriction can be written as $\boldsymbol{\beta}_2 = (\beta_1, \beta_3, \beta_4) = (0, 0, 0)$ with $p = 5$, $p_1 = 2$ and $p_2 = 3$.

To evaluate the performance of the estimators, we obtain the mean square prediction error (MSPE) using bootstrap samples. We draw 1000 bootstrap samples of the 147 subjects from the data matrix $\{(Y_{ij}, \mathbf{X}_{ij}), i = 1, 2, \ldots, 147; j = 1, 2, \ldots, 6\}$. We then calculate the relative prediction error (RPE) of $\boldsymbol{\beta}_1^*$ with respect to $\boldsymbol{\beta}_1^{\text{RFM}}$, the

full model estimator. The RPE is defined as

$$\text{RPE}(\boldsymbol{\beta}_1^{\text{RFM}} : \boldsymbol{\beta}_1^*) = \frac{\text{MSPE}(\boldsymbol{\beta}_1^{\text{RFM}})}{\text{MSPE}(\boldsymbol{\beta}_1^*)} = \frac{(\mathbf{Y} - \mathbf{X}_1\boldsymbol{\beta}_1^{\text{RFM}})'(\mathbf{Y} - \mathbf{X}_1\boldsymbol{\beta}_1^{\text{RFM}})}{(\mathbf{Y} - \mathbf{X}_1\boldsymbol{\beta}_1^*)'(\mathbf{Y} - \mathbf{X}_1\boldsymbol{\beta}_1^*)},$$

where $\boldsymbol{\beta}_1^*$ is one of the listed estimators. If RPE > 1, then $\boldsymbol{\beta}_1^*$ outperforms $\boldsymbol{\beta}_1^{\text{RFM}}$.

Table 4.4 reports the estimates, standard error of the active predictors and RPEs of the estimators with respect to the full model. As expected, the sub-model estimator $\boldsymbol{\beta}_1^{\text{RSM}}$ has the maximum RPE because it's computed when the sub-model is correct that is, $\Delta^* = 0$. It is evident by the RPE values in Table 4.4 that the shrinkage estimators are superior to the penalty estimators (Lasso and aLasso). Furthermore, the positive shrinkage is efficient than the shrinkage estimator.

Table 4.4: Estimate, standard error for the active predictors and RPEs of estimators with respect to full-model estimator for the Amsterdam Growth and Health Study data.

	RFM	RSM	RPT	RSE	RPS	LASSO	aLASSO
Estimate(β_2)	0.381	0.395	0.392	0.389	0.390	0.624	0.611
Standard error	0.104	0.102	0.100	0.009	0.008	0.081	0.079
Estimate (β_5)	0.137	0.125	0.131	0.130	0.133	0.101	0.105
Standard error	0.012	0.010	0.009	0.011	0.010	0.013	0.012
RPE	1.000	1.683	1.355	1.337	1.349	1.109	1.126

4.5.2 Resting-State Effective Brain Connectivity and Genetic data

This data comprises longitudinal resting-state functional magnetic resonance imaging (rs-fMRI) effective brain connectivity network and genetic data obtained from a sample of 111 subjects with a total of 319 rs-fMRI scans from the Alzheimers Disease Neuroimaging Initiative (ADNI) database. The 111 subjects comprise 36 cognitively normal (CN), 63 mild cognitive impairment (MCI) and 12 Alzheimer's Disease (AD) subjects. The response is a network connection between regions of interest estimated from an rs-fMRI scan within the default mode network, and we observe a longitudinal sequence of such connections for each subject with the number of repeated measurements. For this data analysis, we consider the network edge weight from the left intraparietal cortex to posterior cingulate cortex (LIPC $\rightarrow$ PCC) as our response.

The genetic data are single nucleotide polymorphism (SNPs) from non-sex chromosomes, i.e., chromosome 1 to chromosome 22. SNPs with minor allele frequency less than 5% are removed as are SNPs with a Hardy-Weinberg equilibrium p-value lower than 10^{-6} or a missing rate greater than 5%. After preprocessing we are left with 1,220,955 SNPs and the longitudinal rs-fMRI effective connectivity network using the 111 subjects with rs-fMRI data.

In order to apply the proposed methods, we use genome- wide association study (GWAS) for screening the genetic data to reduce the number of SNPs to 100. We implement a second screening through multinomial regression to reduce the dimension to obtain a sub-model. This is done by applying a multinomial logistic regression to identify a much smaller subset of SNPs and this yields a subset of the top 10 SNPs. This showed the top 10 SNPs are the most important predictors and the other 90 SNPs are ignored as not significant. We now have two model which are the full model with all 100 SNPs and submodel with 10 SNPs selected. Finally, we construct the pretest and shrinkage techniques from the full-model and submodel.

We draw 1000 bootstrap samples with replacement from the corresponding data matrix $\{(Y_{ij}, \mathbf{X}_{ij}), i = 1, \ldots, 111; j = 1, \ldots, n_i\}$. We report the RPE based on (5) of the estimators based on the bootstrap simulation with respect to the full model ridge estimator in Table 4.5 . We observe that the RPE of the sub-model, pretest, shrinkage and positive shrinkage ridge estimators outperforms the full model estimator. Clearly, the sub-model estimator has the maximum RPE since it's computed when the candidate sub-model is correct, i.e., $\Delta = 0$. Both shrinkage estimators outperform the pretest estimator. Particularly, the postive shrinkage performed better than the shrinkage estimator. The performance of both ridge shrinkage and pretest estimators is better than the ℓ_1 (Lasso and adaptive Lasso) estimators. Thus, the data analysis is in line with our simulation and theoretical findings.

Table 4.5: RPEs of estimators for submodel.

	RFM	RSM	RPT	RSE	RPS	LASSO	aLASSO
RPE	1.000	1.521	1.386	1.426	1.473	0.983	0.990

4.6 Discussion

In this paper, we present efficient estimation strategies for the LMM for a longitudinal data when the model is sparse. We considered the sparse estimation of fixed effects parameters in linear mixed model when some of the predictors may have a very weak influence on the response of interest. We introduced pretest and shrinkage estimation in our model using the ridge estimation as the reference estimator when predictors are correlated. In addition, we established the asymptotic properties of the pretest and shrinkage ridge estimators. Our theoretical findings demonstrate that shrinkage ridge estimators outperform the ridge full model estimator and perform relatively better than the sub-model estimator in a wide range of the parameter space.

Additionally, a Monte Carlo simulation was conducted to investigate and assess the behavior of proposed estimators when a selected sub-model is the true model or not. As expected, the sub-model ridge estimator outshines shrinkage and penalty when the selected sub-model is true. However, when this assumption is violated, the shrinkage and pretest ridge estimators outperform the sub-model. Furthermore, when the number of sparse predictors are extremely large relative to the sample size , the shrinkage estimators outperform the pretest ridge estimator. These numerical results are consistent with our asymptotic results.

We also assess the relative performance of Lasso-type estimators with our ridge-type estimators. We observe that the performance of pretest and shrinkage ridge estimators are superior to the Lasso-type estimators when predictors are highly correlated. Application of the estimation techniques is applied to the analysis of real datasets where the shrinkage ridge estimators are superior.

The real data and simulation study confirms the dominance of shrinkage estimators over the Lasso-type estimators for moderate to large numbers of sparse predictors. The numerical studies indicate that our pretest and shrinkage estimation strategy improved the estimation performance of selected sub-models. In summary, the results of these data analyses strongly corroborate the findings of the simulation study and suggest the use of the shrinkage estimation strategy when no uncertain prior information (UPI) is available about the parameter subspace.

In our future work, we will focus on other penalty estimators like the Elastic-Net, the minimax concave penalty (MCP), and the smoothly clipped absolute deviation method (SCAD) as estimation strategy in LMM for the longitudinal data. These estimators will be assessed and compared with the ridge estimators. Another interesting

extension will be integrating two sub-models by incorporating pretest and shrinkage estimation strategies in high dimensional sparse mixed effect models. The goal is to improve the estimation accuracy of the active set of the fixed effects parameters by combining an over-fitted model estimator with an under-fitted one (Ahmed et al., 2016; Ahmed and Yüzbaşı, 2017).

Chapter 5

Spectral Dynamic Causal Modelling of Resting-State fMRI: An Exploratory Study Relating Effective Brain Connectivity in the Default Mode Network to Genetics

5.1 Introduction

Alzheimer's Disease (AD) is a neurodegenerative disorder characterized by cognitive decline and progressive dementia and is thought to be caused by aberrant connections between cerebral regions involved in cognitive functioning (Li et al., 2013). Imaging genetics is an important area of scientific investigation in the search for genetic biomarkers of neurodegenerative disease, and in increasing our understanding of the genetic basis of brain structure and function in health and disease.

The development of analytical methods for the joint analysis of neuroimaging phenotypes and genetic data is an important area of statistical research with many challenges. Recent reviews are provided in Liu and Calhoun (2014) and Nathoo et al. (2019). A great deal of work in imaging genetics has focussed on methods and analysis for examining the relationship between brain structure and genetics (see e.g., Stein et al., 2010; Hibar et al., 2011; Ge et al., 2012; Zhu et al., 2014; Greenlaw et al., 2017; Szefer et al., 2017; Lu et al., 2017; Song et al., 2019). Thompson et al.

(2013) give an extensive overview of methods for the analysis of genetic data and brain connectivity with a broad focus on both diffusion tensor imaging (DTI) and fMRI data. These authors discuss the heritability of both structural and functional brain connectivity. Methods for the analysis of brain connectivity with an emphasis on structural connectomes is discussed in Zhang et al. (2018, 2019).

Our focus in this study is in exploring potential associations between brain connectivity and genetics within the context of Alzheimer's disease and mild cognitive impairment. Effective brain connectivity and causal inference is discussed in Lindquist and Sobel (2016), and functional connectivity analysis for fMRI data is reviewed in Cribben and Fiecas (2016). Patel et al. (2006) and Chen et al. (2016) develop Bayesian approaches for modelling brain connectivity and Bowman et al. (2012) consider the analysis of fMRI functional connectivity using a multimodal approach. Here, effective connectivity refers to a directed measure of dependence from one brain region to another (see, e.g., Friston, 1994), while functional connectivity refers to the correlation between measured time series over different locations.

We conduct an analysis examining the relationship between genetics and effective brain connectivity as measured by rs-fMRI within the default mode network (DMN). The DMN consists of a set of brain regions that tend to be active in resting-state, when a subject is mind wandering with no intended task. In this state DMN regions will exhibit low frequency signals that tend to couple together. We consider networks comprised of four (DMN4 - 16 connections) and subsequently six (DMN6 - 36 connections) core regions of the DMN. The DMN4 analyses are based on network nodes located at the medial prefrontal cortex (MPFC), the posterior cingulate cortex (PCC), the left and right intraparietal cortex (LIPC and RIPC) with the Montreal Neurological Institute (MNI) locations for these regions depicted in Figure 5.1. The DMN6 analyses are based on six regions whose MNI locations are specified in Table 5.1. The regions we consider in DMN4 are a subset of the regions considered in DMN6 which, in addition to the original four regions, have added the left inferior temporal region (LIT) and right inferior temporal region (RIT).

Our analyses involve examining effective connectivity networks from rs-fMRI data using Dynamic Causal Modeling (DCM; Li et al., 2011; Friston et al., 2003; Friston et al., 2014; Razi et al., 2015; Friston et al., 2017), a nonlinear state-space framework for inferring interaction between latent neuronal states. We apply DCM to rs-fMRI using the SPM12 (v7219) software (Penny et al., 2011). Resting-state fMRI data are examined with the goal of investigating the potential interaction between different

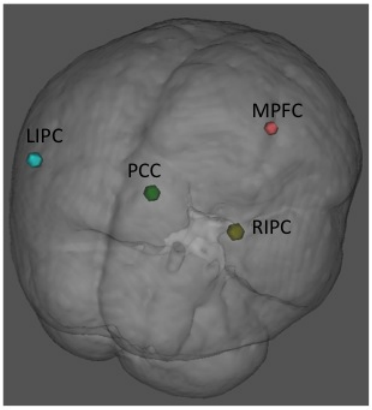


Figure 5.1: The locations of the four regions within the default mode network (DMN) examined in our DMN4 study: the medial prefrontal cortex (MPFC), the posterior cingulate cortex (PCC), the left and right intraparietal cortex (LIPC and RIPC) with MNI coordinates MPFC (3, 54, -2), the PCC (0, -52, 26), LIPC (-50, -63, 32) and RIPC (48, -69, 35).

areas of the brain and to explore the potential association between this neuronal interaction in four (DMN4) and then eight (DMN6) core regions of the DMN and SNPs that are contained in a disease-constrained set. The DCM framework leads to directed networks and these networks are related to genetic data using longitudinal rs-fMRI analysis. Previous literature focussing on AD has found alterations to both effective and functional resting-state connectivity in the DMN (see e.g., Wu et al., 2011; Yan et al., 2013; Luo et al., 2018). Zhong et al. (2014) conduct an rs-fMRI study and demonstrate changes in directed functional connectivity in the DMN for subjects with AD, while Dipasquale et al. (2015) apply high-dimensional independent component analysis (ICA) to an rs-fMRI study and demonstrate changes in functional connectivity in subjects with AD.

Our specific choice of regions which represent the nodes of the network is motivated by existing literature examining connectivity in the DMN and establishing the heritability of effective connectivity for networks based on these regions. These regions are a subset of the DMN regions considered in the rs-fMRI study of Wu et al., (2011), which demonstrated altered DMN functional and effective connectivity in AD, and for DMN4 they are the same regions considered in the rs-fMRI study

	DMN Region	MNI Coordinate
1	Posterior cingulate (PCC)	0, -52, 26
2	Medial Prefrontal (MPFC)	3, 54, -2
3	Left intraparietal cortex (LIPC)	-50, -63, 32
4	Right intraparietal cortex (RIPC)	48, -69, 35
5	Left inferior temporal (LIT)	-61, -24, -9
6	Right inferior temporal (RIT)	58, -24, -9

Table 5.1: The MNI coordinates associated with the regions of interest in our study. The first four rows correspond to the network DMN4 while DMN6 corresponds to all eight rows.

of Sharaev et al. (2016), which investigated internal DMN relationships. Xu et al. (2017) examine effective connectivity in these four regions of the DMN using DCM and structural equation modelling in a twin study based on a sample of $n = 46$ pairs of twins with rs-fMRI. These authors find evidence for the heritability of effective connectivity in this network. We note that this study uses stochastic DCM rather than spectral DCM and the two approaches are not equivalent. They estimate the heritability of DMN effective connectivity in these regions to be 0.54 (that is, the proportion of variability in DMN effective connectivity that can be attributed to genetics is 0.54). Their study provides evidence that there are genes involved in DMN effective connectivity for the network nodes depicted in Figure 1. This work paves the way for our study of relationships between effective connectivity and genetic markers in the same network.

Glahn et al. (2010) use an extended pedigree design and rs-fMRI to examine genetic influence on functional connectivity within the DMN. Their study estimates the heritability of DMN functional connectivity to be 0.424 ± 0.17. That is to say, these authors estimate the proportion of variability of DMN functional connectivity that can be attributed to genetics to be 0.42 ± 0.17. This estimate is within error bounds of the heritability estimate of 0.54 obtained by Xu et al. (2017) for DMN effective connectivity. Importantly, Glahn et al. (2010) also suggest that the genetic factors that influence DMN functional connectivity and the genetic factors that influence gray matter density in these regions seem to be distinct. This result then motivates the search for genetic markers associated with DMN connectivity. Stingo et al. (2013) focus on relating brain connectivity to genetics and develop a Bayesian hierarchical mixture model for studies involving fMRI data. The mixture components of the proposed model correspond to the classification of the study subjects

into subgroups, and the allocation of subjects to these mixture components is linked to genetic covariates with regression parameters assigned spike-and-slab priors.

Our strategy for data analysis proceeds as follows. We view both disease and rs-fMRI as measures of the brain and our fundamental interest is to relate these measures of the brain to genetics. The former is a coarse measure with three categories while the latter is a far a more detailed measure allowing for the investigation of statistical dependencies in the temporal rs-fMRI signal at different regions.

We use an out-of-sample genome-wide scan of disease to select a priority subset of SNPs and this serves as a constraint on the SNPs that we relate to effective brain connectivity in subsequent analysis. We relate effective brain connectivity as characterized through spectral DCM to the empirical disease-constrained subset of genetic variables using longitudinal analyses based on both linear mixed effect (LME) models and function-on-scalar (FSR) regression (Morris, 2015). When testing for SNP effects in both models we compare p-values obtained from standard asymptotic distributions to those obtained from the parametric bootstrap.

5.2 Data and Preprocessing

Data used in the preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). The ADNI was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of ADNI has been to test whether serial magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and early AD. For up-to-date information, see `www.adni-info.org`. ADNI is an ongoing, longitudinal, multicenter study designed to develop clinical, imaging, genetic, and biochemical biomarkers for the early detection and tracking of AD.

The selection criteria for our sample is as follows. We first begin with ADNI2 subjects (1437 at this stage) and consider those subjects with genome-wide data (774 left at this stage) and also with at least one resting-state fMRI scan at 3 Tesla (111). This leads to 111 subjects comprising 36 cognitively normal (CN), 63 MCI and 12 AD subjects, with these subjects having a mean age of 73.8 years with the range being 56.3-95.6 years, 46 of these subjects being male, 5 being left-handed, and with education measured in years ranging from 11 to 20.

Table 5.2 presents several summary statistics associated with our sample, including a summary on the Apolipoprotein E (APOE) gene. The APOE gene is a known genetic determinant of AD risk and individuals carrying the $\epsilon 4$ allele are at an increased risk of AD (see, e.g., Liu et al. 2013). Table 5.2 summarizes the number of APOE $\epsilon 4$ alleles for the subjects in each disease category. In line with expectations from the literature (Genin et al., 2011), a signal from the APOE gene is present in the data (p-value = 0.0045; Fisher's Exact Test) with the AD group having a higher percentage of subjects with at least one $\epsilon 4$ allele. In fact, the data summaries in Table 5.2 indicate that all but one of the AD subjects have at least one $\epsilon 4$ allele of the APOE gene.

Table 5.2: Distribution of demographic variables (obtained at the baseline visit) across disease groups within our sample of 111 subjects. The p-values in the final column are based on a one-way ANOVA for continuous variables and a Fisher's exact test for categorical variables.

		AD	MCI	CN	p-value
n		12	63	36	
GENDER (% of group)	Female	8 (66.7)	34 (54.0)	23 (63.9)	0.6402
	Male	4 (33.3)	29 (46.0)	13 (36.1)	
HAND (% of group)	Left	1 (8.3)	2 (3.2)	2 (5.6)	0.5036
	Right	11 (91.7)	61 (96.8)	34 (94.4)	
Age - mean (sd)		75.82 (7.91)	72.70 (7.66)	75.35 (6.73)	0.15
EDUCATION - mean (sd)		16.33 (2.53)	16.06 (2.66)	16.19 (2.14)	0.929
APOE $\epsilon 4$ Alleles (% of group)	Zero	1 (8.3)	35 (55.6)	24 (66.6)	0.0045
	One	9 (75.0)	22 (34.9)	11 (30.6)	
	Two	2 (16.7)	6 (9.5)	1 (2.8)	

Diagnostic classification of AD participants was made by ADNI investigators according to diagnostic criteria for probable AD established by the National Institute of Neurological and Communicative Disorders and Stroke and the Alzheimer's Disease Related Disorders Association (NINCDS-ADRA; McKhann et al., 1984). Participants in the AD cohort also exhibited abnormal memory function on the Logical Memory II subscale of the revised Wechsler Memory Scale (WMS II, ≤ 8 for 16 years education and above), a Mini Mental State Exam (MMSE) between 20 and 26 (inclusive), and a Clinical Dementia Rating of 0.5 (very mild) or 1 (mild). All control participants were free of memory complaints and deemed cognitively normal based on clinical assessments by the site physician showing an absence of significant impairment in

cognitive functioning and performance of daily activities. Participants in the control cohort also exhibited normal memory function on the Logical Memory II subscale of the revised WMS (WMS II, ≤ 9 for 16 years of education and above), a MMSE score between 24 and 30 (inclusive), and a Clinical Dementia Rating of 0.

As described in Bondi et al. (2014), the ADNI criteria for MCI are: 1) subjective memory complaints reported by themselves, study partner, or clinician; 2) objective memory loss defined as scoring below an education-adjusted cut-off score on delayed recall of Story A of the WMS-R Logical Memory Test (score =8 for those with =16 years of education; score =4 for those with 8-15 years of education; score =2 for those with 0-7 years of education); 3) global CDR score of 0.5; and 4) general cognitive and functional performance sufficiently preserved such that a diagnosis of dementia could not be made by the site physician at the time of screening.

Our analysis examines longitudinal rs-fMRI data from 111 subjects with repeated measurements resulting in a total of 424 rs-fMRI scans to begin with. Each scan leads to a network which has an associated follow-up time. As the longitudinal data in this study are quite sparse, we apply the Principal Analysis by Conditional Expectation (PACE; Muller, 2008) method which focuses on recovering the entire continuous-time temporal trajectory of the network edge parameters. As a result of the sparsity of the longitudinal data, it is necessary to borrow information across subjects to recover the temporal trend of the network edge parameters for individual subjects. Doing so requires us to restrict the temporal trajectory within a time window (of length 500 days) covering all subjects. The proportion of subjects with scans outside of the 500 day window is not sufficient to recover the temporal trajectory outside of the window, and thus we restrict our analysis to the 319 rs-fMRI scans falling within it. The advantage of doing this is that we are able to use the function-on-scalar regression model where the response is the temporal profile of the network edge parameters.

The function-on-scalar regression model considers the association of the temporal trend of the network edge parameters with genetic variables, while the linear mixed effects model only focuses on the association of the network edge parameters with the genetic variables. The mixed-effects model simply treats the network edge parameters at multiple time points as repeated measurements and accounts for clustering, which ignores the time order and trend of the network edge parameters. We view this as a key aspect of the longitudinal data. Therefore, the function-on-scalar model may be able to find a stronger association between the network edge parameters with genetics as it accounts explicitly for time ordering (and not just clustering) in the

network response. Thus after preprocessing the data in this way we are left with 319 scans obtained from 111 subjects for the longitudinal analysis.

MRI for these subjects are collected at 3 Tesla (3T). MRI data are downloaded with permission from the ADNI. All images were acquired on 3T Philips MRI scanners across 10 North American acquisitions sites according to the standardized ADNI protocol. Whole-brain anatomical MRI scans were acquired sagittally, with a T1-weighted magnetization-prepared rapid acquisition with gradient echo (MPRAGE) sequence, with the following parameters: 1.2 mm slice thickness, 256 by 256 by 170 acquisition matrix, echo time (TE) of 3 ms, in-plane voxel dimension of 1 mm^2, repetition time (TR) of 7 ms, and flip angle of 9 degrees. Functional MRI scans were obtained during resting-state; participants were instructed to lay quietly in the scanner with their eyes open. Resting state fMRI scans were obtained with a T2*-weighted echo-planar imaging sequence with the following parameters: 140 volumes, 64 by 64 by 48 acquisition matrix (voxel size = 3.3 mm^3), TE of 30 ms, TR of 3000 ms, and flip angle of 80 degrees.

The freely available software package PLINK (Purcell et.al., 2007) is used for genomic quality control and preprocessing. The genetic data are SNPs from non-sex chromosomes, i.e., chromosome 1 to chromosome 22. SNPs with minor allele frequency less than 5% are removed as are SNPs with a Hardy-Weinberg equilibrium p-value lower than 10^{-6} or a missing rate greater than 5%. After preprocessing we are left with 1,220,955 SNPs for each of 111 subjects for relating genetic data to effective connectivity and 1,220,955 SNPs for each of an additional 663 subjects for selecting a disease-constrained subset of SNPs.

5.2.1 rs-fMRI Data Preprocessing and Network Estimation

The fMRI and anatomical data are pre-processed using a combination of the FSL software (available at `http://fsl.fmrib.ox.ac.uk/fsl/fslwiki/`) and the SPM12 software (available at `http://www.l.ion.ucl.ac.uk/spm/software/spm12/`). Non-brain tissue in the raw T1 images is removed using the automated Brain Extraction Tool (Smith, 2002), followed by manual verification and optimization for each subject. Blood-Oxygen-Level Dependent (BOLD) image data preprocessing is performed in FSL's FEAT as follows: each functional image is motion corrected and registered to their high-resolution T1 structural image that is linearly registered to standard stereotaxic space using a 12 degree-of-freedom transformation. A non-linear registra-

tion of the structural image to standard stereotactic space is also applied to account for potential local deformations in brains of the patient group. Each subject's imaging data are normalized to a standardized space defined by an MNI template brain. We conducted manual checking for correct normalization of every image.

Given a set of R brain regions of interest, DCM in the case of fMRI models the haemodynamic response over these regions through a nonlinear state-space formulation with a model allowing for interaction between regions and with model parameters that characterize effective connectivity and, when relevant, how this connectivity is modulated by experimental inputs. In the case of resting-state fMRI with no experimental inputs, the model can be expressed as (see, Razi et al., 2017)

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{v}(t) \\ \boldsymbol{y}(t) &= h(\boldsymbol{x}(t), \boldsymbol{\theta}) + \boldsymbol{e}(t), \end{aligned} \tag{5.1}$$

where $\boldsymbol{x}(t) = (x_1(t), \ldots, x_R(t))'$ are latent variables used to represent the states of neuronal populations at some time t and $\dot{\boldsymbol{x}}(t)$ is a time-derivative defining a differential equation approximating neuronal dynamics, with the $R \times R$ matrix $\boldsymbol{A}$ approximating effective connectivity to first-order; $h(\boldsymbol{x}(t), \boldsymbol{\theta})$ is a nonlinear mapping from hidden neuronal states to the predicted haemodynamic response also depending on parameters $\boldsymbol{\theta}$ (see, e.g., Friston, 2007, for details on the form of this nonlinear mapping); $\boldsymbol{y}(t) = (y_1(t), \ldots, y_R(t))'$ with $y_j(t)$ being a summary of the response obtained from all voxels within region j; $\boldsymbol{v}(t)$ and $\boldsymbol{e}(t)$ represent neuronal state noise and measurement noise respectively.

For resting-state fMRI, the DCM can be fit in the time-domain using Bayesian filtering based on a mean-field variational Bayes approximation (see, e.g., Li et al., 2011) which involves inference on both model parameters and latent states. Alternatively, the model can be fit in the spectral domain using an approach known as spectral DCM (Friston et al., 2014). The latter approach involves relating the theoretical cross spectra associated with the dynamic model to the sample cross spectra in order to estimate parameters. Thus, it is somewhat akin to a method of moments approach. More specifically, Friston et al. (2014) assume a parameterized power law form for the spectral densities of the noise terms in the state-space model and then express the empirical cross spectra as the sum of the theoretical cross spectra and measurement error. This formulation then yields a likelihood for the observed cross spectra statistic depending on the time-invariant parameters but not depending on

the latent variables $\boldsymbol{x}(t)$. This likelihood for the summary statistic is then combined with a prior distribution for the model parameters and an approximation to the associated posterior distribution for these parameters is obtained using variational Bayes. Razi et al. (2015) report simulation results that demonstrate estimators obtained from spectral DCM having higher accuracy (in the sense of mean-squared error) than those obtained from stochastic DCM. In addition, the former has a higher computational efficiency since estimation of the latent states is not required. We use this approach to estimate effective connectivity networks within the DMN.

The DMN includes the posterior cingulate cortex/precuneus (PCC), medial prefrontal cortex (MPFC), bilateral inferior parietal lobule (IPL), and other regions including the inferior temporal gyrus. To estimate effective connectivity within the regions of the DMN depicted in Figure 1 and Table 1, we use spectral DCM as implemented in SPM12. Blood-oxygen-level-dependent (BOLD) time series from the DMN regions of interest are obtained by extracting time series from all voxels within an 8mm radius of the associated MNI coordinate, and then applying a principle component analysis and extracting the first eigenvariate. For simplicity, the information contained in the other eigenvariates is not considered in our analysis so that the result is a single representative time series for each region of interest. This procedure is repeated to obtain a collection of four (DMN4) or six (DMN6) time series for each subject. An example of the resulting time-series data for a single subject over DMN6 is depicted in Figure 5.2.

A 16-parameter (DMN4) or 36-parameter (DMN6) graph with weights representing effective connectivity between and within regions is then estimated for each subject. This graph is based on an estimate of the parameter $\boldsymbol{A}$, an $R \times R$ non-symmetric matrix, in equation (5.1) obtained from spectral DCM. We fit the DCM in SPM12 with the option of one state per region and with the model fit to the cross spectral density (which corresponds to spectral DCM). The imaging preprocessing pipeline is summarized in Figure 5.3. We note that slice-timing correction as implemented in SPM12 is applied to all of the functional images. All slices of one volume are interpolated in time to the reference slice (reference slice = 24 out of 48). Spatial smoothing is performed in SPM12 where we convolve image volumes with a spatially stationary Gaussian filter (i.e. a Gaussian kernel) of $8 \times 8 \times 8$ mm^3 full width half max (FWHM).

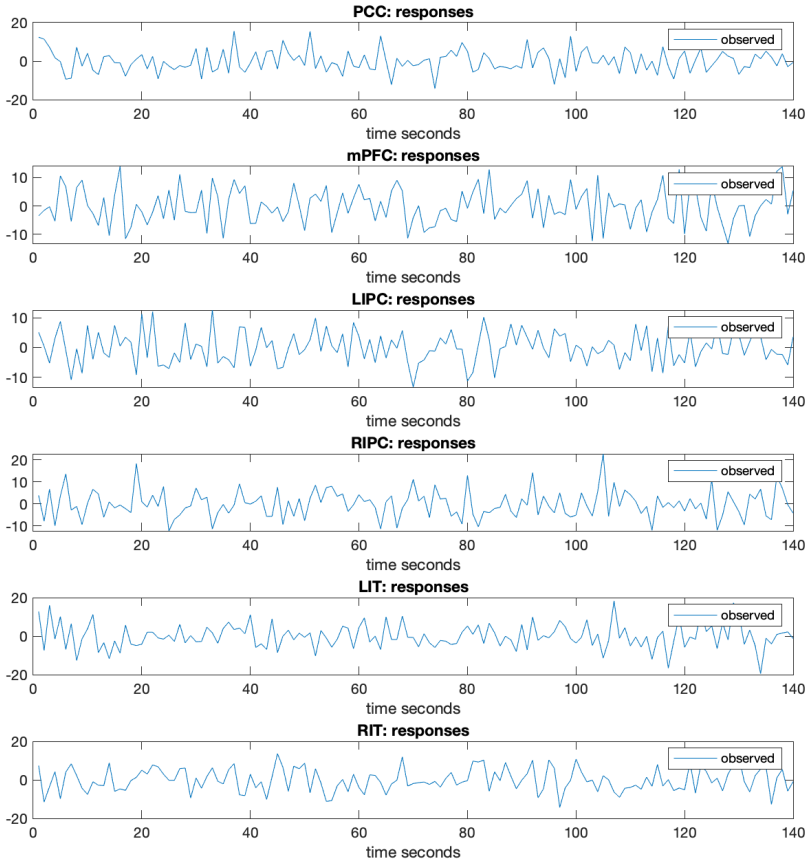


Figure 5.2: An example of the rs-fMRI data used to estimate the effective connectivity network for a single subject from the six regions of interest (PCC, MPFC, LIPC, RIPC, LIT, RIT).

5.3 Selection of the Disease-Constrained Set of SNPs

Beginning with the 1,220,955 SNPs discussed in Section 2, we conduct a genome-wide association study (GWAS) with the goal of identifying a smaller subset of SNPs that are potentially associated with disease (CN/MCI/AD). This subset serves as a constraint in subsequent analysis examining effective brain connectivity and it is selected out-of-sample. That is to say, the selection of the constrained SNP subset does not involve the 111 subjects with rs-fMRI data but rather is obtained from taking the original sample of 774 subjects having genome-wide data and removing those 111 subjects with rs-fMRI. The remaining 663 ADNI2 subjects have genome-wide data

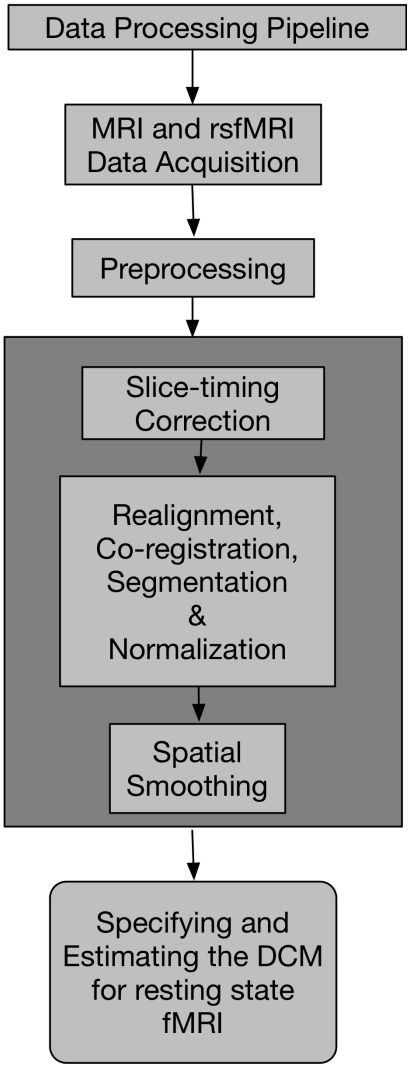


Figure 5.3: The neuroimaging data preprocessing pipeline used in our study. Slice-timing correction as implemented in SPM12 is applied to all of the functional images. All slices of one volume are interpolated in time to the reference slice (reference slice = 24 out of 48). Spatial smoothing is performed in SPM12 where we convolve image volumes with a spatially stationary Gaussian filter (i.e. a Gaussian kernel) of $8 \times 8 \times 8$ mm^3 full width half max (FWHM).

only and we conduct the GWAS on the data from these subjects in order to identify a smaller subset of SNPs that are potentially associated with disease (CN/MCI/AD). This yields a subset of top 100 SNPs which we then relate to the longitudinal rs-fMRI effective connectivity networks using the 111 subjects with rs-fMRI data. This approach has two primary advantages in producing a more reliable subset of SNPs than within sample selection:

1. The disease-constrained set is based on a larger sample size of 663 subjects compared with 111 subjects.

2. Selecting the subset out-of-sample avoids double usage of the data in selecting the priority subset of SNPs and in relating that subset to longitudinal rs-fMRI.

A multinomial logistic regression with disease category as the response is fit for each SNP to assess that SNP's marginal association with disease after adjusting for covariates representing age, sex, handedness, and education. SNPs are included in the model as the number of a particular allele so that a SNP's effect on the log-odds ratio is additive. We sort the SNPs by the resulting p-values from a likelihood ratio test, where the null hypothesis corresponds to the case where the probability distribution of disease does not depend on the given SNP. A subset of the top 100 SNPs is selected based on this ranking. The distribution of p-values by chromosome and the cut-off for selecting the best subset of 100 SNPs is depicted in Figure 5.4. Each of the 100 SNPs in the selected constrained subset has a p-value below 7.5×10^{-5}, which represents the cut-off indicated in Figure 5.4.

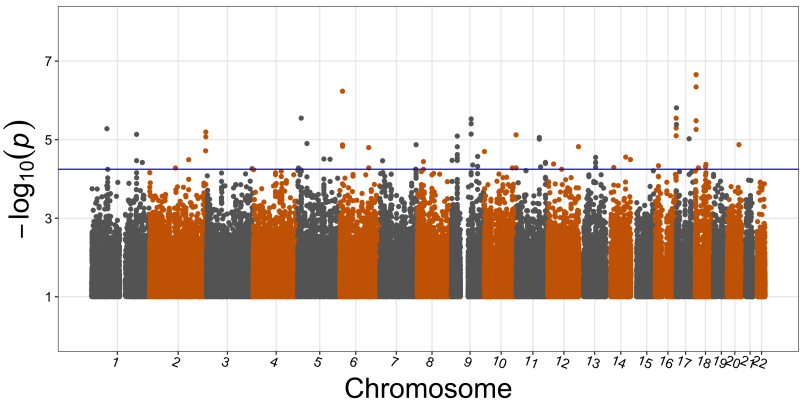


Figure 5.4: The p-values associating disease status with SNPs adjusting for age, sex, handedness, and education. The blue line represents the cutoff used to obtain the top 100 SNPs which corresponds to a p-value threshold of 7.5×10^{-5}.

While Table 5.2 indicates an APOE signal in our data from the $\epsilon 4$ allele of the gene, we note that there are no APOE related SNPs in our constrained subset of the top 100. The highest ranking APOE related SNP is rs4802234 which has a p-value of 6.9×10^{-4}. Thus our choice to use a p-value threshold of 7.5×10^{-5} in subsequent analyses eliminates the highest ranking APOE related SNP, even though there appears to be some evidence of an APOE signal in the data. We note that we

have defined APOE related SNPs as those within a 1 million base pair range of APOE and their p-values ranging between 6.9×10^{-4} to 0.999. Despite this, all subsequent regression models that we use will include a variable representing APOE genotype (coded as number of $\epsilon 4$ alleles) in order to account for its known importance and to adjust for its potential association with effectivity brain connectivity.

5.4 Resting-State Effective Brain Connectivity by Genetics

5.4.1 Longitudinal Analysis with Four DMN Regions

We conduct the longitudinal analysis using both a linear mixed effects model incorporating subject-specific random effects as well as a function-on-scalar regression. In each case the response is a network connection between two regions of interest estimated from an rs-fMRI scan using DCM, and we observe a longitudinal sequence of such connections for each subject with the number of repeated measurements ranging from 1 to 4 with a median of 3 and a third-quartile of 4. In both networks at an initial q-value threshold of 0.1 no effects are found. We report on exploratory patterns of associations with relatively high ranks that exhibit stability to the differing assumptions made by the statistical procedures described below.

For a given connection, we will fit a regression model relating that connection to a single SNP, for each of the 100 SNPs in the disease-constrained set. These regression models for the time-sequence of connection values also include covariates representing age, sex, right/left hand, education and APOE$\epsilon 4$. With 16 possible connections a total of $16 \times 100 = 1600$ models are fit and in each case a test comparing the model with and without the SNP from the constrained-set is conducted. The resulting set of 1600 p-values arising from this mass-univariate longitudinal approach are adjusted for multiplicity using an FDR adjustment (Benjamini and Hochberg, 1995) in order to produce for each edge-SNP pair a q-value. If a q-value = α cut-off is selected then among all rejections based on this cut-off, there are about $100 \times \alpha$ % that are falsely rejected. In this exploratory analysis we focus primarily on the rank of each edge-SNP pair by the p-value but we present the q-value as well.

The linear mixed effect (LME; Kuznetsova et al., 2017) model has the form

$$Y_{ij} = \mu + X_i\beta + \boldsymbol{Z}_i^T\boldsymbol{\alpha} + b_i + \epsilon_{ij}$$

where Y_{ij} is the network edge weight estimated from the j^{th} rs-fMRI scan of the i^{th} subject, $i = 1, 2, ..., 111$, $j = 1, 2, ..., m_i$; X_i is the SNP coded additively in the model; $\boldsymbol{Z}_i$ contains the remaining covariates with coefficient vector $\boldsymbol{\alpha}$, b_i is a subject-specific random effect and ϵ_{ij} is an error term with the random effect and errors assumed Gaussian and independent. The likelihood ratio test is conducted for the hypothesis test $H_0 : \beta = 0$ against $H_1 : \beta \neq 0$ and the corresponding p-value obtained.

In addition to obtaining a p-value from the standard χ_1^2 asymptotic null distribution for the likelihood ratio statistic we also apply a parametric bootstrap. For the linear mixed effects model the parametric bootstrap procedure proceeds by first fitting both the null (excluding the SNP) and alternative (including the SNP) model using maximum likelihood and the observed value of the likelihood ratio statistic is obtained. Data are then generated under the null model and both the null and alternative model are fit to each simulation replicate and a realization of the test statistic under the null hypothesis is obtained. This is repeated for $n_{sim} = 300,000$ replicates and the bootstrap p-value is obtained as the proportion of null test statistic realizations as large or larger than the observed test statistic.

We apply the PACE approach for sparse functional data to obtain the estimated trajectories of network edge weights over time t, $\widehat{Y}_i(t)$. By applying PACE, the estimated trajectory of each network edge weight can be obtained and expressed as

$$\widehat{Y}_i(t) = \mu(t) + \sum_{j=1}^{J} \xi_{ij}\phi_j(t),$$
$$\widehat{\boldsymbol{Y}}(t) = \mu(t) + \boldsymbol{\xi}_{n\times J}\boldsymbol{\Phi}(t)_{J\times 1},$$

where $\mu(t)$ represents the mean function common to all n subjects,

$$\xi_{ij} = \int \{Y_i(t) - \mu(t)\}\phi_j(t)dt$$

is the j-th functional principal component (FPC) score for $Y_i(t)$, with $\phi_j(t)$ being the corresponding eigenfunction, and $J = 3$ since three functional principal components explain over 95% of the total of variation.

The function-on-scalar regression model has the form

$$Y_i(t) = X_i\alpha(t) + \boldsymbol{Z}_i^T\boldsymbol{\beta}(t) + \epsilon_i(t),$$

where X_i is the SNP with corresponding regression function $\alpha(t)$, $\boldsymbol{Z}_i$ is a vector holding covariates representing age, sex, right/left hand, education and APOEϵ4 with corresponding regression function $\boldsymbol{\beta}(t)$ and the error term $\epsilon_i(t)$ is a random error processes. As the raw data $Y_i(t)$ is sparse we replace it in the regression model with the PACE-estimated trajectory of each network edge weight. We test $H_0 : \alpha(t) = 0$ corresponding to the model that excludes the genetic marker using an F-test. As with our use of linear mixed models we apply a mass univariate approach and apply the model to all $16 \times 100 = 1600$ possible combinations of network edge and SNP, and apply an FDR correction for multiplicity. We note that the degrees of freedom for the F-distribution needs to be carefully computed. In the F-test for FSR the degrees of freedom are setup by applying the idea of Satterthwaite's approximation described in Shen and Faraway (2004).

In addition to obtaining p-values from the F-distribution we also apply a parametric bootstrap. To the best of our knowledge, there is no existing package or function in R that can be used for simulation with the function-on-scalar regression (FSR) model. We therefore propose the following procedure to generate functional data for bootstrapping the test statistics under the null and its corresponding p-value for hypothesis testing the SNP regression function in FSR models.

The two FSR models considered in the hypothesis test for the SNP effect are:

1. Null model: $Y_i(t) = \sum_{j=1}^{P} Z_{ij}\beta_j(t) + \epsilon_i(t)$, where the SNP regression function $\alpha(t) = 0$.

2. Alternative hypothesis: $Y_i(t) = X_i\alpha(t) + \sum_{j=1}^{P} Z_{ij}\beta_j(t) + \epsilon_i(t)$, where $\alpha(t) \neq 0$.

For each bootstrap sample we simulate the residual function $\epsilon_i(t)$ with the empirical mean and covariance functions of the estimated residual function $\hat{\epsilon}_i(t)$ of the FSR model under the null. We note that in the FSR model, each residual term $\epsilon_i(t)$ is assumed to arise from a Gaussian process with mean zero and covariance function $r(s,t) = COV[\epsilon_i(t), \epsilon_i(s)]$. We then generate the functional response $Y_i(t)$ using the null FSR model with the original scalar covariates Z_{ij}, their corresponding estimated functional coefficients $\hat{\beta}_j(t)$, and the simulated residual functions from the previous step. We then re-fit the two FSR models from the null and alternative hypotheses

to the functional responses simulated under the null and compute the F-statistic as a realization under the null. This procedure is repeated to obtain $n_{sim} = 10,000$ replicates and the bootstrap p-value is obtained as the proportion of null test statistic realizations as large or larger than the observed test statistic.

The LME assumes that the effective-connectivity is time-invariant and all of the session-specific fluctuations are captured by the noise term in the LME. The FSR model on the other hand allows for session-specific fluctuations through the time-varying regression functions. This allows the FSR analysis to be potentially more powerful since all of the fluctuations are not allocated to the noise term. An important point is that LME and FSR are approaches that make fundamentally different assumptions about the data generating mechanism.

In Tables 5.3 and 5.4 we display the top ten ranked edge-SNP associations with the smallest p-values for DMN4 as obtained from both LME and FSR. The bootstrap p-values (B-p-values) as well as the FDR adjusted q-values are also indicated in the tables. As an exploratory analysis of these results we intersect the top 20 edge-SNP associations out of 1600 (based on the p-value) from LME with the top 20 edge-SNP associations out of 1600 (based on the p-value) from FSR. Our rationale for doing this is that both LME and FSR are based on different modelling assumptions with different model fitting procedures used in each. Therefore, signals that appear as highly ranked from both approaches simultaneously may represent genuine signal in the data since it is unlikely that noise will be stable to the different assumptions and procedures applied to the data. Intersecting the top 20 associations from both methods produces a very clear pattern of 8 associations that are presented in Table 5.5. The pattern reveals a potential signal involving connections LIPC $\rightarrow$ RIPC, LIPC $\rightarrow$ PCC and PCC $\rightarrow$ PCC and SNPs from chromosome 9 and 13.

5.4.2 Longitudinal Analysis with Six DMN Regions

We apply the same approaches to the networks estimated using spectral DCM for the network of 6 regions listed in Table 5.1. In this case the models are applied to $36 \times 100 = 3600$ possible combinations of network edge and SNP, and a p-value, B-p-value and FDR corrected q-value are obtained to adjust the p-value for multiplicity.

In Tables 5.6 and 5.7 we rank the ten associations with the smallest p-values for DMN6 as obtained from LME and FSR. We again conduct an exploratory analysis of these results by intersecting the top 20 edge-SNP associations out of 3600 (based

Rank	Network Edge	SNP	p-value	B-p-value	q-value	Chromosome
1	LIPC → PCC	kgp9433690_G	0.00025	0.00040	0.28	9
2	LIPC → PCC	rs13287994_A	0.00035	0.00065	0.28	9
3	PCC → PCC	rs9317920_G	0.0013	0.0022	0.45	13
4	LIPC → RIPC	rs13287994_A	0.0020	0.0026	0.45	9
5	LIPC → PCC	kgp4931190_C	0.0021	0.0030	0.45	9
6	PCC → PCC	kgp12216228_G	0.0025	0.0036	0.45	13
7	PCC → PCC	rs1935110_T	0.0025	0.0033	0.45	13
8	PCC → MPFC	kgp1147116_A	0.0027	0.0039	0.45	9
9	LIPC → RIPC	kgp4931190_C	0.0028	0.0042	0.45	9
10	LIPC → LIPC	rs2646852_G	0.0030	0.0033	0.45	1

Table 5.3: The results of the linear mixed effects model longitudinal analysis of pairs of network edges with SNPs that are relatively highly ranked based on the p-values, parametric-bootstrap p-values (B-p-value) and q-values (after adjusting for 1600 tests) performed using the DMN4 PACE data.

Rank	Network Edge	SNP	p-value	B-p-value	q-value	Chromosome
1	LIPC→ RIPC	rs13287994_A	0.00074	0.0010	0.24	9
2	RIPC→ PCC	rs7935380_T	0.00093	0.0010	0.24	11
3	RIPC→ PCC	rs2646852_G	0.0011	0.0014	0.24	1
4	MPFC→ MPFC	kgp2936399_A	0.0012	0.0013	0.24	5
5	LIPC→ RIPC	kgp9433690_G	0.0013	0.0022	0.24	9
6	LIPC → RIPC	kgp4931190_C	0.0014	0.0014	0.24	9
7	RIPC → RIPC	kgp12216228_G	0.0014	0.0020	0.24	13
8	RIPC → RIPC	rs1935110_T	0.0014	0.0020	0.24	13
9	MPFC → MPFC	rs2646852_G	0.0014	0.0023	0.24	1
10	RIPC → RIPC	rs9317920_G	0.0017	0.0029	0.24	13

Table 5.4: The results of the function-on-scalar regression longitudinal analysis of pairs of network edges with SNPs that are relatively highly ranked based on the p-values and q-values after adjusting for 1600 tests performed using the DMN4 data.

Rank LME	Rank FSR	Network Edge	SNP	Chromosome
4	1	LIPC → RIPC	rs13287994_A	9
9	6	LIPC → RIPC	kgp4931190_C	9
11	5	LIPC → RIPC	kgp9433690_G	9
1	14	LIPC → PCC	kgp9433690_G	9
2	18	LIPC → PCC	rs13287994_A	9
3	16	PCC → PCC	rs9317920_G	13
6	19	PCC → PCC	kgp12216228_G	13
7	20	PCC → PCC	rs1935110_T	13

Table 5.5: The rank (out of 1600) of the intersection of the top 20 DMN4 connection-SNP pairs for FSR and LME combined. These associations are highlighted as potential signals as a result of their stability.

Rank	Network Edge	SNP	p-value	B-p-value	q-value	Chromosome
1	LIPC → PCC	rs13287994_A	7.33e-05	2.00e-04	0.18	9
2	LIPC → PCC	kgp9433690_G	1.21e-04	3.00e-05	0.18	9
3	LIPC → LIT	kgp9433690_G	1.59e-04	9.99e-05	0.18	9
4	LIPC → LIT	rs13287994_A	2.041e-04	1.50e-04	0.18	9
5	MPFC → MPFC	rs17102906_C	3.61e-04	1.50e-04	0.26	11
6	RIPC → RIT	kgp10801842_G	4.52e-04	3.50e-04	0.27	5
7	LIPC → LIT	kgp4931190_C	6.25e-04	4.50e-04	0.29	9
8	LIT → LIT	kgp3071374_C	7.53e-04	4.70e-04	0.29	6
9	LIT → LIT	rs9388153_T	8.63e-04	5.20e-04	0.29	6
10	LIPC → PCC	kgp4931190_C	8.69e-04	0.0012	0.29	9

Table 5.6: The results of the linear mixed effects model longitudinal analysis of pairs of network edges with SNPs that are relatively highly ranked based on the p-values, parametric-bootstrap p-values (B-p-value) and q-values (after adjusting for 3600 tests) performed using the DMN6 PACE data.

on the p-value) from LME with the top 20 edge-SNP associations out of 3600 (based on the p-value) from FSR. For DMN6 we see seven intersecting top 20 associations involving self connections (regions modulating their own activity) for MPFC, LIT and RIT involving SNPs on chromosomes 11 and 6. Regions in chromosome 6 have been previously implicated for Late-Onset Alzheimer Disease (Naj et al., 2010). There are also two intersecting top 20 associations between PCC → RIPC and SNPs on chromosome 18, and an intersecting association between LIT → LIPC and SNP rs6021246 on chromosome 20. The stability of these associations makes them of potential interest as signal between the larger network and disease-constrained SNPs though we emphasize that these results are exploratory. While we see a non-trivial change in the results when moving from DMN4 to DMN6 this can be explained simply by noting that many of the higher ranked SNP-edge pairs in DMN6 involve the two added nodes corresponding to ROIs RIT and LIT.

Tables 5.3, 5.4, 5.6 and 5.7 make comparisons between the p-values obtained from the χ^2_1 and F-distribution and the B-p-values obtained from the parametric bootstrap procedures developed for both LME and FSR. Generally, the p-values are inline with the B-p-values and the comparison from the four tables suggest that the asymptotic distributions provide an adequate approximation for the case of these data. To investigate this further we plot in Figures 5.5 and 5.6 the histograms of the 3600 (1600 in the case of DMN4) p-values obtained from all of the tests relating SNPs to effective connectivity as obtained from the χ^2_1 and F-distribution. As most

Rank	Network Edge	SNP	p-value	B-p-value	q-value	Chromosome
1	RIPC → LIT	rs11601321_G	2.00e-04	1.00e-04	0.30	11
2	LIT → LIT	rs17102906_C	2.70e-04	2.00e-4	0.30	11
3	RIT → RIT	rs17102906_C	3.50e-04	7.00e-4	0.30	11
4	LIT → LIT	kgp3071374_C	3.60e-4	8.00e-4	0.30	6
5	LIT → LIT	rs9388153_T	4.90e-04	5.00e-4	0.30	6
6	MPFC → MPFC	rs17102906_C	5.10e-4	7.00e-4	0.30	11
7	RIPC → RIPC	kgp3071374_C	9.40e-4	1.50e-3	0.42	6
8	MPFC → MPFC	kgp3071374_C	1.07e-3	1.40e-3	0.42	6
9	RIT → RIPC	rs3862175_C	1.15e-3	1.10e-3	0.42	18
10	PCC → RIPC	rs949200_T	1.28e-3	1.40e-3	0.42	18
10	PCC → RIPC	kgp8588069_A	1.28e-3	1.70e-3	0.42	18

Table 5.7: The results of the function-on-scalar regression longitudinal analysis of pairs of network edges with SNPs that are relatively highly ranked based on the p-values, parametric-bootstrap p-values (B-p-value) and q-values after adjusting for 3600 tests performed using the DMN6 PACE data.

Rank LME	Rank FSR	Network Edge	SNP	Chromosome
5	6	MPFC → MPFC	rs17102906_C	11
14	14	MPFC → MPFC	rs9388153_T	6
11	8	MPFC → MPFC	kgp3071374_C	6
8	4	LIT → LIT	kgp3071374_C	6
9	5	LIT → LIT	rs9388153_T	6
17	2	LIT → LIT	rs17102906_C	11
18	3	RIT → RIT	rs17102906_C	11
12	19	LIT → LIPC	rs6021246_C	20
19	10	PCC → RIPC	rs949200_T	18
20	10	PCC → RIPC	kgp8588069_A	18

Table 5.8: The rank (out of 3600) of the intersection of the top 20 DMN6 connection-SNP pairs for FSR and LME combined. These associations are highlighted as potential signals as a result of their stability.

SNPs should be under the null, adequate performance of the asymptotic distributions should be indicated by uniform distributions. In all four cases (DMN4-LME, DMN6-LME, DMN4-FSR, DMN6-FSR) the distributions of the p-values look reasonably uniform which, in addition to the comparisons between p-values and B-p-values made in Tables 5.3, 5.4, 5.6 and 5.7, further suggests that the asymptotic null distributions are preforming adequately in the cases considered here.

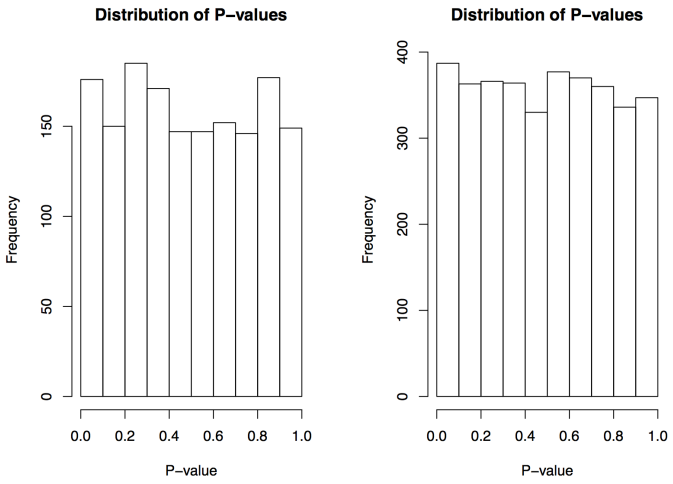


Figure 5.5: Distribution of p-values obtained from the χ^2_1 null distribution for LME for DMN4 (left) and DMN6 (right).

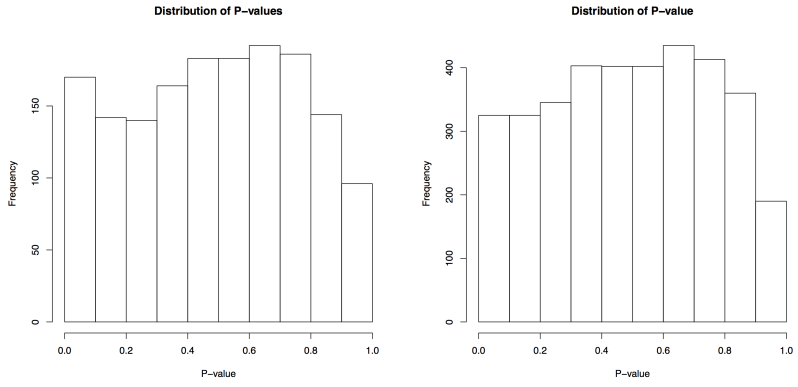


Figure 5.6: Distribution of p-values obtained from the null F-distribution for FSR for DMN4 (left) and DMN6 (right).

5.5 Discussion

We have examined the association between longitudinal effective brain connectivity and genetics based on an empirically derived disease-constrained set of SNPs using LME and FSR, with networks estimated from rs-fMRI time series using spectral DCM. Our analysis is based on obtaining a disease-constrained subset of SNPs from the outcome of the out-of-sample GWAS in Section 5.3, the purpose being to focus attention on potential associations that are empirically most relevant for disease. We have examined both a 4-region and a 6-region network depicted in Figure 5.1 and Table 5.1.

We have examined both a 4-region and a 6-region network so as to obtain a more complete picture of the data. Indeed, our results from DMN4 obtained after intersecting the top associations reveal an interesting pattern that would not have been noticed if only the larger DMN6 network was analyzed where many of the top-ranked edge-SNP pairs involve LIT and RIT. As part of future work we intend to increase the number of regions to a much larger size (e.g. networks where the number of nodes is an order of magnitude greater and spanning multiple known resting-state networks) and examine measures of network topology as opposed to network edge values as a potential useful low-dimensional phenotype.

After adjusting for multiple comparisons none of the associations examined have a q-value below 0.1. By intersecting the top 20 associations and focussing on stability to the assumptions made by LME and FSR we do observe some patterns that may be of interest from an exploratory point of view. The most interesting patterns can be summarized as follows:

1. DMN4: LIPC $\rightarrow$ RIPC and three SNPs on chromosome 9 (rs13287994, kgp4931190, kgp9433690) exhibit top 20 associations that are stable under LME and FSR.

2. DMN4: LIPC $\rightarrow$ PCC and two SNPs on chromosome 9 (kgp9433690, rs13287994) exhibit top 20 associations that are stable under LME and FSR. The large genomic screen conducted by Pericak-Vance et al. (2000) identified potential AD-related regions on chromosome 9.

3. DMN4: PCC $\rightarrow$ PCC and three SNPs on chromosome 13 (rs9317920, kgp12216228, rs1935110) exhibit top 20 associations that are stable under LME and FSR.

4. DMN6: PCC $\rightarrow$ RIPC and two SNPs on chromosome 18 (rs949200, kgp8588069) exhibit top 20 associations that are stable under LME and FSR.

5. DMN6: Several self-connections involving the MPFC, LIT, RIT on chromosomes 6 and 11. Regions in chromosome 6 have been previously implicated for Late-Onset Alzheimer Disease (Naj et al., 2010).

Our comparisons of the parametric bootstrap procedures for LME and FSR to the p-values obtained from the χ^2_1 and F-distribution suggest that the asymptotic distributions are performing adequately for the cases considered here. This is evident both in the comparisons made in Tables 5.3, 5.4, 5.6 and 5.7 as well as in Figures 5.5 and 5.6. Further investigation of this issue, in particular for testing regression functions in FSR, is an avenue for future work.

Our current analysis suggests follow-up analyses looking at high resolution anatomical and diffusion tensor imaging and structural connectivity in a separate sample of subjects with such an analysis focussed on a small number of targeted SNPs suggested by the current analysis. From a methodological perspective, our approach of determining the disease-constrained set first and subsequently examining relationships with effective brain connectivity motivate the development of a joint model examining both disease and longitudinal effective connectivity simultaneously with a model structure that allows for joint variable (SNP) selection.

While connectivity may be related to the degeneration of gray matter our regression analyses have not incorporated gray matter density as a covariate in either the LME or FSR models. In future work we will look to reporting on an analysis that includes gray matter density in the ROIs for each subject as well as examine a much larger set of ROIs including bilateral hippocampi and the combination of structural and effective connectivity.

We note again that our analysis summarizes ROI specific activity by extracting the time series at all voxels within a sphere having radius 8mm around an associated MNI coordinate for the ROI. A principal component analysis is applied to these time series and the first eignenvariate is used as a region-specific summary. Subsequent analysis may consider additional eigenvariates which may hold useful information representing effective connectivity.

While our analysis has focussed on hypothesis tests examining individual SNPs, an alternative is to cluster the SNPs using hierarchical clustering and fit the LME and FSR models to the first genetic principal component scores of each cluster. In supplementary analyses not reported in Section 4 we have clustered the top 100 SNPs into 22 clusters based on hierarchical clustering to facilitate dimension reduction prior to relating genetics to effective brain connectivity. After applying SNP clustering and

relating the first principal component score of the SNP clusters to effective connectivity network edges both LME and FSR find simultaneously a rank 1 association between a specific cluster of 12 SNPs (rs2465362, kgp9051645, kgp5238984, kgp9055011, kgp3071374, rs9388153, rs2860410, kgp22784175, rs7335304, rs9317920, kgp12216228, rs1935110) and the self connection PCC $\rightarrow$ PCC (p-value $= 3.52 \times 10^{-4}$, q-value $=$ 0.0502 for LME; p-value $= 3.05 \times 10^{-4}$, q-value $= 0.0585$ FSR). This potential association between the resting-state activity of the posterior cingulate cortex and genetics is further supported by our results reported in Table 5.5 where individual associations between three SNPs in this cluster and PCC $\rightarrow$ PCC are also reported in the high ranking intersection set obtained from combining LME and FSR.

The neuropathological mechanisms that underly the effective connectivity observed in this study could be related to the usual hallmarks of AD and MCI. For example, amyloid plaques, neurofibrillary tangles and/or structural neurodegeneration may underly effective connectivity as well. Therefore, our current results suggest follow-up studies incorporating measures of amyloid beta and tau (as measured in CSF or by PET imaging) and measures of brain structure (as measured by high resolution anatomical and diffusion tensor imaging) and examination of the relationship between these measures and the genetic variables suggested by this analysis.

Chapter 6

Conclusions and Future Directions

In this thesis we have developed statistical methods for the analysis of imaging data and sparse estimations in linear mixed models.

In Chapter 2, we discuss the problem of computation for MAP estimation for the spatial hidden Markov model with both known and unknown number of mixture components. A spatial mixture models with Gaussian components and labelings based on the Potts model is developed. Our primary focus lies with the design and implementation of hybrid ACS-ICM optimization algorithm and its application to spatial hidden Markov models when the number of mixture components is either known or unknown. In addition to the ACS algorithm, we implement Gibbs-SA and ICM and make comparisons. Our studies demonstrate a superior performance of the ACS-ICM algorithm when compared to Gibbs-SA and ICM algorithms in terms of objective function values, pixel labelling accuracy and mixture component estimation. This relative performance in the three algorithms appears constant across different levels of spatial dependence. Our results suggest that ACS-ICM is a useful and potentially powerful approach for MAP estimation with spatial mixture models and may be a preferred approach when the increased computation time is feasible.

The algorithm may have considerable potential for application to other statistical problems involving combinatorial optimization such as MAP estimation with spike-and-slab variable selection (Rockova et al., 2014) and optimization for deep learning (Lecun et al., 2015). A potentially important extension of this work would be to develop an ant colony optimization algorithm for training recurrent neural networks. These and other statistical and experimental design problems are potentially fruitful areas for further study of ACS-ICM.

In Chapter 3, we implement ACS-ICM algorithm for solving the inverse problem

associated with combined EEG and MEG data that appears to result in some improvements over the ICM algorithm developed in Song et al. (2019). We adopt a Bayesian finite mixture model with a Potts model as a spatial prior, the focus of our work has been to improve source localization estimates, model selection and model fit. Our first contribution is the design and implementation of the ACS-ICM algorithm as an approach for source localization that result in better performance over ICM, which is very positive uniformly in every setting on simulation studies and real data application.

Comparing the results in several simulation scenarios, with those obtained with ACS-ICM and the ICM shows revealing performance, at the price of a higher computational complexity. ACS-ICM neural source estimates provided improved correlation, larger objective function values, significant improvement with respect to the total mean square error, and improved performance in terms of both bias and mean square error for the non-regular problem of estimating number of mixture components uniformly across all settings considered. Furthermore, the application of ACS-ICM to real data led to higher quality estimates with larger maximized posterior density values. These improvements have demonstrated the practical advantage of the of the ACS-ICM algorithm when compared with ICM in both the face perception analysis as well as in our simulation studies.

Our second contribution is the technique implemented in choosing the tuning parameters for the ACS-ICM by using an outer level optimization that numerically optimizes the choice of the tuning parameters for this algorithm. This strategy ensures that the optimal tuning parameters based on the data and problem complexity are selected. For our third contribution, to account for uncertainty in our point estimates we compute the interval estimation for the estimated temporal profile of brain activity at the cortical location with greatest power in the estimated signal of the T-map. While our algorithm requires a larger amount of computing time than the alternative approach considered here, we emphasize that an MCMC implementation of our model would require a prohibitively large amount of computing time. We note that our approach, as with most in the literature for spatiotempoal model-based solutions to the electromagnetic inverse problem assume Gaussian errors. We hope in the future to extend the model by considering a robust error structure by incorporating a multivariate t distribution and allowing for an autoregressive structure in the MEG/EEG model. In addition, the ACS-ICM algorithm and spatial model developed is only applicable to data from a single subject. A great interest and development is

to expand from a single subject model to a model developed for multiple subjects for the MEG/EEG inverse problem. This will be based on developing a fully Bayesian analysis based on a divide and conquer Markov Chain Monte Carlo (MCMC) method.

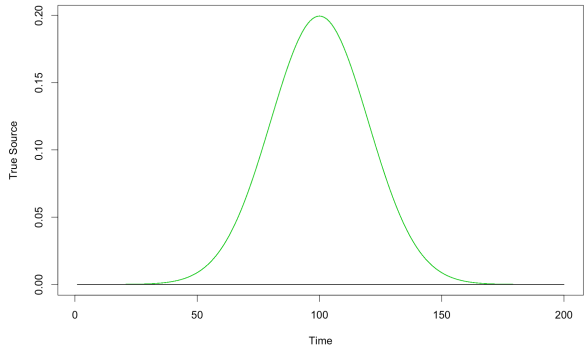
In Chapter 4, we consider the fixed effects estimation problem in a linear mixed model for a longitudinal data when some of the predictors may have a very weak influence on the response of interest. We introduced pretest and shrinkage estimation in our model using the ridge estimation as the reference estimator when predictors are correlated. We compare the performance of the full model, sub model, pretest and shrinkage estimators when the full model may be sparse. The asymptotic properties including risk and bias of the pretest and shrinkage ridge estimators are derived and established. We evaluate the risk properties of the proposed estimators numerically through simulation studies. Our simulation studies demonstrate that the sub-model ridge estimator outshines shrinkage and penalty when the selected sub-model is true. However, when this assumption is violated, the shrinkage and pretest ridge estimators outperform the sub-model. In addition, the shrinkage estimators perform better than the pretest estimators for large or moderate number of sparse predictors. We further investigate the Lasso-type penalty estimators and compare their relative performance with the ridge estimators. We observe that the performance of pretest and shrinkage ridge estimators are superior to the Lasso-type estimators when predictors are highly correlated. An interesting and useful development is an extension of the estimation strategies to other penalty estimators like the Elastic-Net, the minimax concave penalty (MCP), and the smoothly clipped absolute deviation method (SCAD) in LMM for the high-dimensional data. These estimators will be evaluated and compared with the proposed ridge estimators.

In Chapter 5, we examined the association between longitudinal effective brain connectivity and genetics based on an empirically derived disease-constrained set of SNPs using LME and FSR, with networks estimated from rs-fMRI time series using spectral DCM. The analysis is based on obtaining a disease-constrained subset of SNPs from the outcome of the out-of-sample GWAS. The LME assumes that the effective-connectivity is time-invariant and all of the session-specific fluctuations are captured by the noise term in the LME whilst the FSR model allows for session-specific fluctuations through the time-varying regression functions. Due to different modelling assumptions with different model fitting procedures used in each, we consider signals that appear as highly ranked from both approaches simultaneously may represent genuine signal in the data. From an exploratory point of view, after inter-

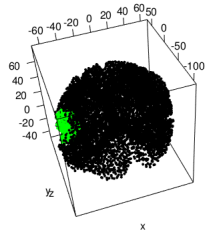
secting the top 20 associations from LME and FSR, we observe a pattern that reveals a potential signal involving connections LIPC $\rightarrow$ RIPC, LIPC $\rightarrow$ PCC and PCC $\rightarrow$ PCC and SNPs from chromosome 9 and 13 in the DMN4. For the six regions of the DMN (DMN6), a pattern for MPFC, LIT and RIT involving SNPs on chromosomes 11 and 6 is seen. This result corroborates a study by Naj et al., 2010 where regions in chromosome 6 have been previously implicated for Late-Onset Alzheimer Disease. Moving beyond hypothesis tests examining individual SNPs, we cluster the SNPs using hierarchical clustering and fit the LME and FSR models to the first genetic principal component scores of each cluster. We find a potential association between the resting-state activity of the posterior cingulate cortex and genetics (12 SNPs). A limitation of the proposed methodology is that our analysis summarizes ROI specific activity by extracting the time series at all voxels within a sphere having radius 8mm around an associated MNI coordinate for the ROI and principal component analysis is applied to these time series and the first eigenvariate is used as a region-specific summary. Extending the approach to hold useful information representing effective connectivity, subsequent analysis may consider additional eigenvariates. There are some neuropathological mechanisms that underlay the effective connectivity observed in this analysis which could be related to AD and MCI. A potentially important extension of the proposed work would be incorporating measures of amyloid beta and tau (as measured in CSF or by PET imaging) and measures of brain structure (as measured by high resolution anatomical and diffusion tensor imaging) and examination of the relationship between these measures and the genetic variables suggested by this analysis.

A

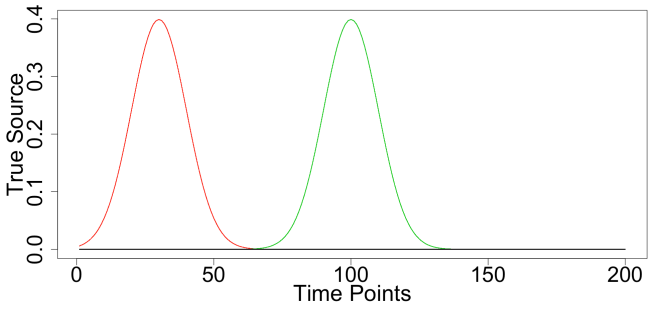
Appendix A



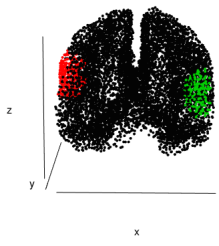
(a)



(b)



(c)



(d)

Figure A.1: *Cont.*

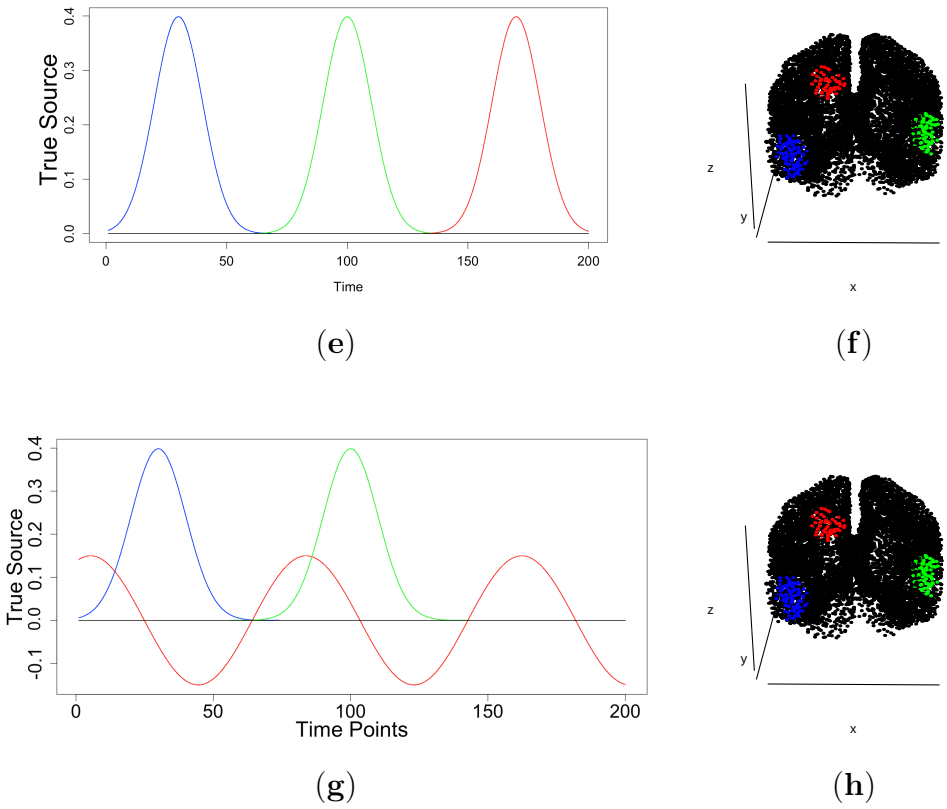
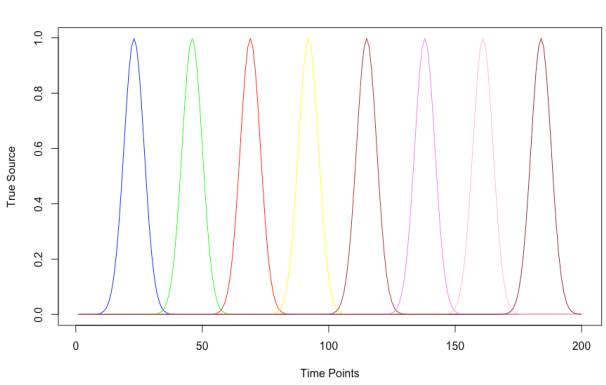
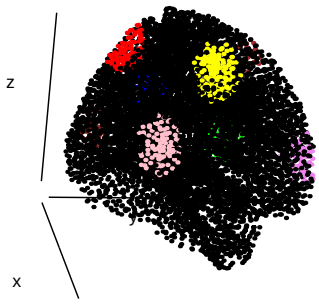


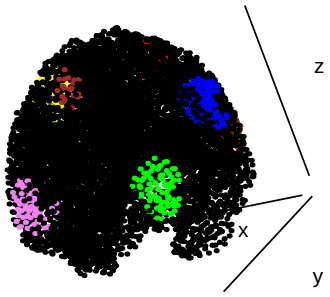
Figure A.1: The true signal $S_j(t)$ used in each of the distinct active and inactive regions in the simulation studies of Section 3.3 for $K = 2$; panel (**a**), $K = 3$; panel (**c**) and $K = 4$; panel (**e**) & (**g**) are depicted in the left column. The right column presents the true partition of the cortex into active and inactive states for the corresponding states for $K = 2$; panel (**b**), $K = 3$; panel (**d**) and $K = 4$; panel (**f**) & (**h**).



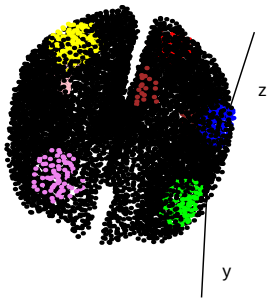
(**a**)



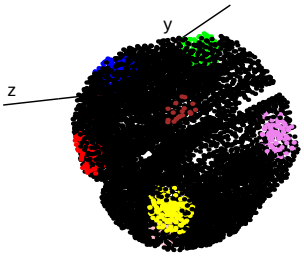
(**b**)



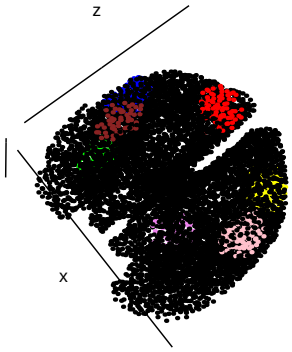
(**c**)



(**d**)

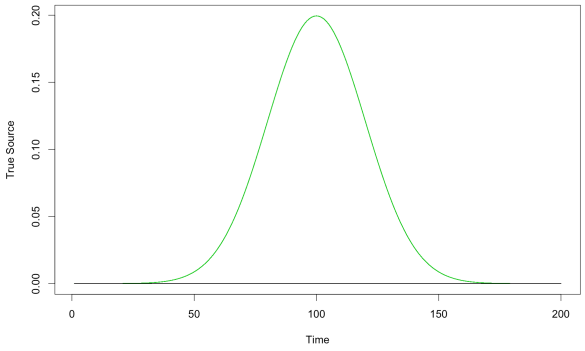


(**e**)

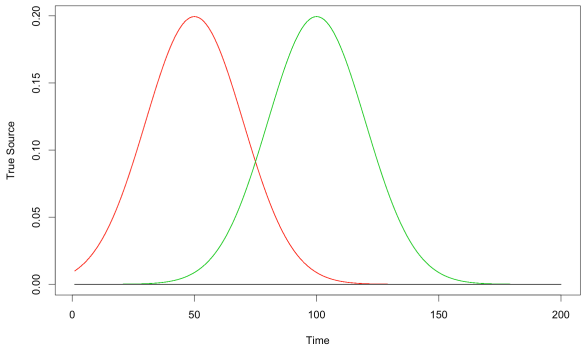


(**f**)

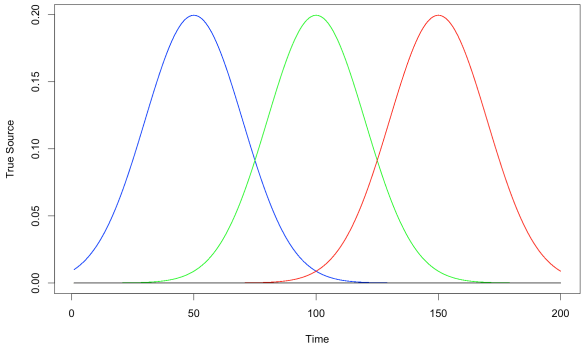
Figure A.2: The true signal $S_j(t)$ in panel (**a**) and true partition of the cortex into active and inactive states for the case of $K = 9$ states (panel **b**–**f**) used in simulation studies of Section 3.3.



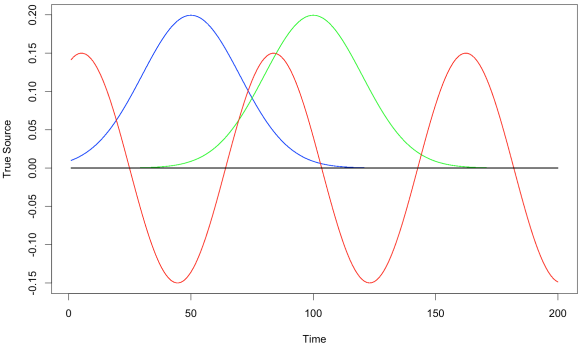
(**a**)



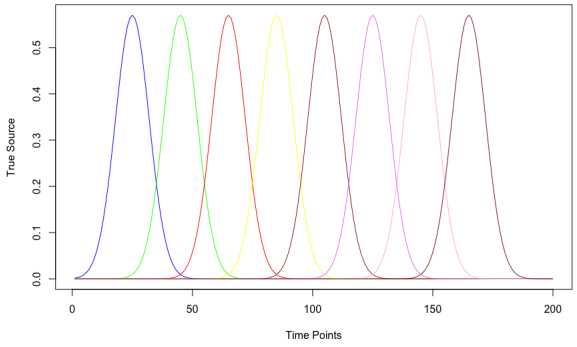
(**b**)



(**c**)



(**d**)



(**e**)

Figure A.3: The true signal $S_j(t)$ used in in each of the distinct active and inactive regions for $K = 2$; panel (**a**), $K = 3$; panel (**b**), $K = 4$; panel (**c**) & (**d**), $K = 9$; panel (**e**) in the simulation study of Section 3.3.2, in the second part of the study where the mixture components were less well separated.

Table A.1: Simulation study I—Total Mean-Squared Error (TMSE) of the neural source estimators decomposed into variance and squared bias for the ICM and ACS-ICM algorithms. This total was obtained separately for locations in active regions and then for the inactive region.

		Active Region			**Inactive Region**		
Algorithm	**Clusters**	**TMSE**	**(Bias)²**	**Variance**	**TMSE**	**(Bias)²**	**Variance**
				$K = 2$			
ICM	250	92	36	56	141	65	76
ACS-ICM	250	83	32	51	127	61	66
ICM	500	196	91	105	211	103	108
ACS-ICM	500	183	87	96	191	93	98
ICM	1000	271	147	124	285	127	158
ACS-ICM	1000	263	144	119	274	125	149
				$K = 3$			
ICM	250	490	237	253	523	255	268
ACS-ICM	250	465	219	246	497	244	253
ICM	500	1203	582	621	705	345	360
ACS-ICM	500	904	434	470	593	282	311
ICM	1000	1657	817	840	674	321	353
ACS-ICM	1000	1051	379	672	601	289	312
				$K = 4$			
ICM	250	776	378	396	674	289	385
ACS-ICM	250	681	336	345	614	248	366
ICM	500	1404	651	753	804	387	417
ACS-ICM	500	1152	555	597	781	375	406
ICM	1000	2493	1100	1393	796	359	437
ACS-ICM	1000	1763	797	966	774	346	428
				$K = 9$			
ICM	250	2100	918	1182	1541	727	814
ACS-ICM	250	1446	709	737	1303	632	671
ICM	500	2515	1246	1269	1260	618	642
ACS-ICM	500	2142	1104	1038	1046	492	554
ICM	1000	3561	1720	1839	1549	740	809
ACS-ICM	1000	2714	1281	1433	1415	688	727

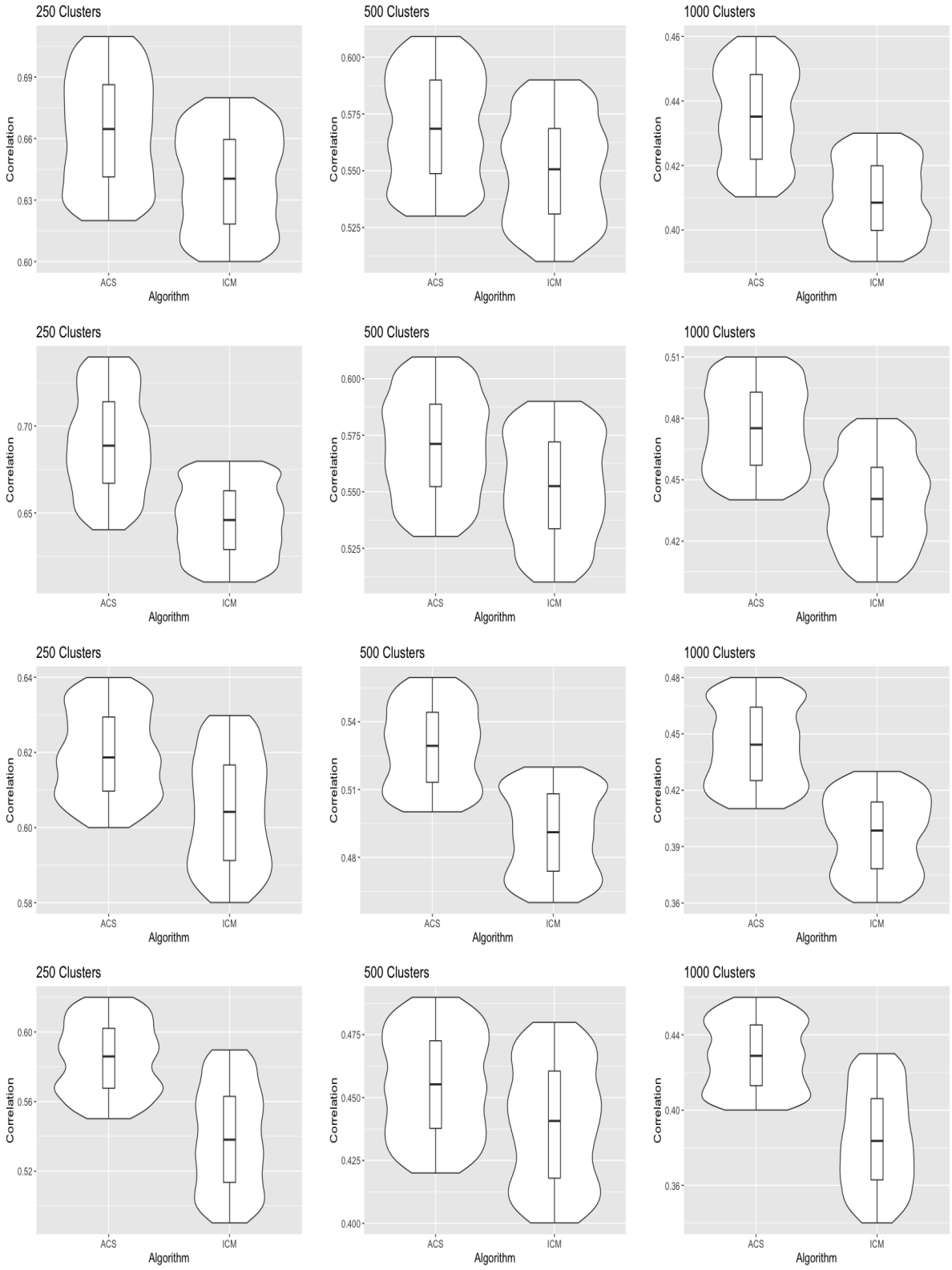


Figure A.4: Violin plots comparing the correlation values obtained in the simulation studies for the ICM and ACS-ICM algorithms. The first row corresponds to the case when $K = 2$, the second row corresponds to when $K = 3$, the third row is when $K = 4$ and the last row is when $K = 9$.

A.1

We present results from the ACS-ICM and ICM algorithms with tuning parameters for the cluster size of 500. The cortical maps displaying the spatial patterns of the total power for estimated sources are represented below.

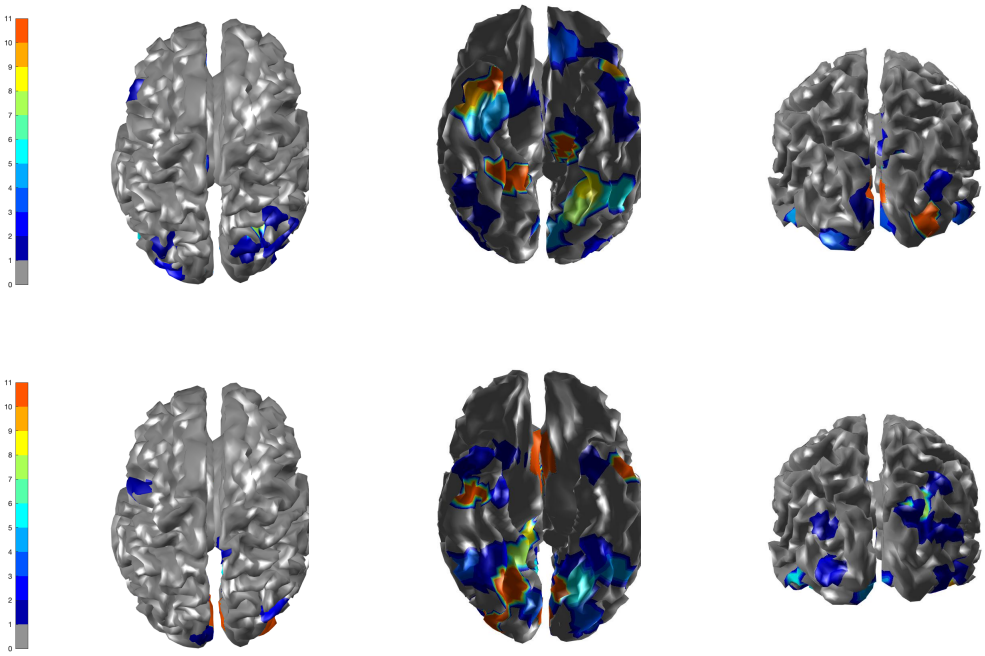


Figure A.5: Brain activation for scrambled faces—the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** displays results from our ICM algorithm applied to the combined MEG and EEG data; **Row 2** displays results from ACS applied to the combined MEG and EEG data.

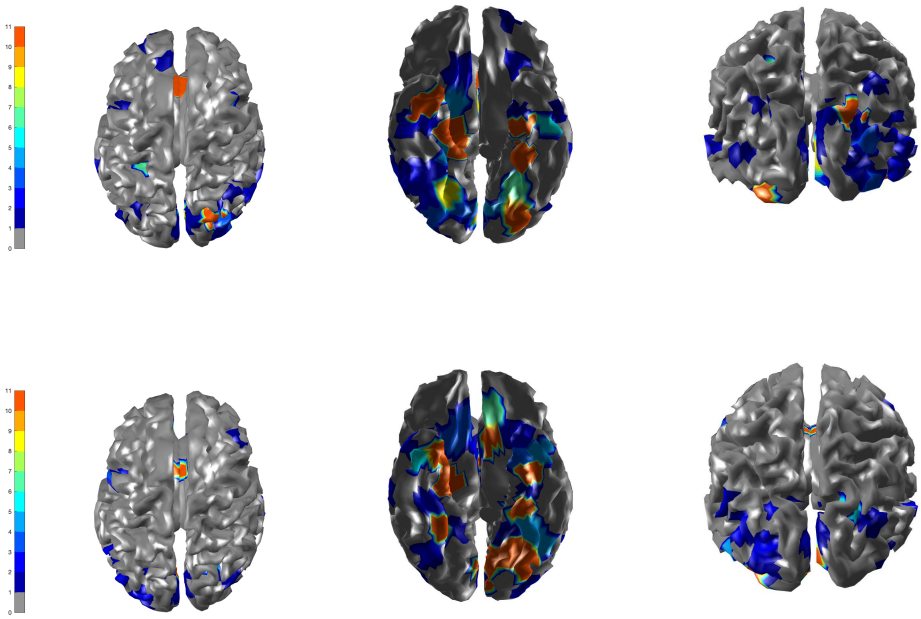


Figure A.6: Brain activation for scrambled faces-the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** and **Row 2** displays results from our ACS-ICM applied to the combined MEG and EEG data with $\beta = 0.1$ and $\beta = 0.44$ respectively.

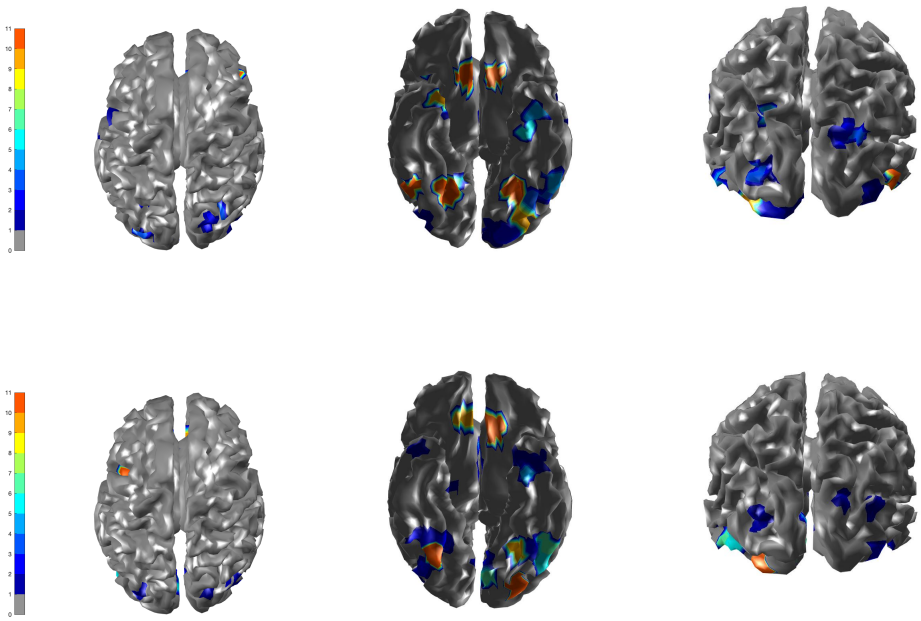


Figure A.7: The spatial profile of brain activity from ACS-ICM based on our bootstrap replicates. **Row 1** displays standard deviations of the total power of the estimated source activity; **Row 2** displays the T-map.

Bibliography

[1] Ahmed, S. E. (2014). Penalty, shrinkage and pretest strategies: variable selection and estimation (p. 2014). New York: Springer.

[2] Ahmed, S. E., & Nicol, C. J. (2012). An application of shrinkage estimation to the nonlinear regression model. Computational Statistics & Data Analysis, 56(11), 3309–3321.

[3] Ahmed, S. E., & Opoku, E. A. (2017). Submodel selection and post-estimation of the linear mixed models. In Proceedings of the Tenth International Conference on Management Science and Engineering Management (pp. 633-646). Springer, Singapore.

[4] Ahmed, S. E., & Raheem, S. E. (2012). Shrinkage and absolute penalty estimation in linear regression models. Wiley Interdisciplinary Reviews: Computational Statistics, 4(6), 541–553.

[5] Ahmed, S. E., Kim, H., Yıldırım, G., & Yüzbaşı, B. (2016, June). High-Dimensional Regression Under Correlated Design: An Extensive Simulation Study. In International Workshop on Matrices and Statistics (pp. 145-175). Springer, Cham.

[6] Ahmed, S. E., & Yüzbaşı, B. (2017). High dimensional data analysis: Integrating submodels. In Big and Complex Data Analysis (pp. 285-304). Springer, Cham.

[7] Ayasso H, Mohammad-Djafari A. (2010). Joint ndt image restoration and segmentation using gauss–markov–potts prior models and variational bayesian computation. IEEE Transactions on Image Processing 19(9):2265–2277.

[8] Belsley, D. A. (2014). Conditioning diagnostics. Wiley StatsRef: Statistics Reference Online.

[9] Benjamini, Y. & Hochberg, Y., (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal statistical society: series B (Methodological), 57(1), pp.289-300.

[10] Besag J. (1977). Efficiency of pseudolikelihood estimation for simple gaussian

fields. Biometrika pp 616–618.
[11] Besag J. (1986). On the statistical analysis of dirty pictures. Journal of the Royal Statistical Society: Series B (Methodological) 48(3):259–279.
[12] Bondi MW, Edmonds EC, Jak AJ, Clark LR, Delano-Wood L, McDonald CR, Nation DA, Libon DJ, Au R, Galasko D, Salmon DP, (2014). Neuropsychological criteria for mild cognitive impairment improves diagnostic precision, biomarker associations, and progression rates. Journal of Alzheimer's Disease; 42(1):275-89.
[13] Bouzbita S., El Afia A., Faizi R. (2019). Hidden markov model classifier for the adaptive acs-tsp pheromone parameters. In: bioinspired Heuristics for Optimization, Springer, pp 153–169.
[14] Cai, S., Peng, Y., Chong, T., Zhang, Y., M von Deneen, K., Huang, L., & Alzheimer's Disease Neuroimaging Initiative. (2017). Differentiated effective connectivity patterns of the executive control network in progressive MCI: a potential biomarker for predicting AD. Current Alzheimer Research, 14(9), 937-950.
[15] Cribben, I. and Fiecas, M., (2016). Functional connectivity analyses for fMRI data. Handbook of Neuroimaging Data Analysis, 369.
[16] Desell T., Clachar S., Higgins J.,Wild B. (2015). Evolving deep recurrent neural networks using ant colony optimization. In: European Conference on Evolutionary Computation in Combinatorial Optimization, Springer, pp 86–98.
[17] Destrempes F., Mignotte M., Angers JF (2005). A stochastic method for bayesian esti- mation of hidden markov random field models with application to a color model. IEEE Transactions on Image Processing 14(8):1096–1108.
[18] Dipasquale, O., Griffanti, L., Clerici, M., Nemni, R., Baselli, G., & Baglio, F. (2015). High-dimensional ICA analysis detects within-network functional connectivity damage of default-mode and sensory-motor networks in Alzheimer?s disease. Frontiers in human neuroscience, 9, 43.
[19] Dogandzic A.,Zhang B. (2006). Distributed estimation and detection for sensor networks using hidden markov random field models. IEEE Transactions on Signal Processing 54(8):3200–3215.
[20] Dorigo M,Gambardella L.M. (1997). Ant colony system:a cooperative learning approach to the traveling salesman problem. IEEE Transactions on evolutionary computation 1(1):53–66.
[21] Emdadi A.,Moughari FA., Meybodi FY., Eslahchi C. (2019). A novel algorithm for parameter estimation of hidden markov model inspired by ant colony optimization. Heliyon 5(3):e01,299.

[22] Ejaz Ahmed, S., & Yuzba, B. (2016). Big data analytics: integrating penalty strategies. International Journal of Management Science and Engineering Management, 11(2), 105–115.
[23] Friston, K.J., (1994). Functional and effective connectivity in neuroimaging: a synthesis. Human brain mapping, 2(1-2), pp.56-78.
[24] Friston, K., Zeidman, P., & Litvak, V. (2015). Empirical Bayes for DCM: a group inversion scheme. Frontiers in systems neuroscience, 9, 164.
[25] Friston, K.J., Harrison, L. & Penny, W. (2003). Dynamic causal modelling. Neuroimage, 19(4), pp.1273-1302.
[26] Friston, K. J., Preller, K. H., Mathys, C., Cagnan, H., Heinzle, J., Razi, A., & Zeidman, P. (2017). Dynamic causal modelling revisited. Neuroimage.
[27] Friston, K.J., Kahan, J., Biswal, B. & Razi, A., (2014). A DCM for resting state fMRI. Neuroimage, 94, pp.396-407.
[28] Ge, T., Feng, J., Hibar, D.P., Thompson, P.M. & Nichols, T.E., (2012). Increasing power for voxel-wise genome-wide association studies: the random field theory, least square kernel machines and fast permutation procedures. Neuroimage, 63(2), pp.858-873.
[29] Geladi, P., & Kowalski, B. R. (1986). Partial least-squares regression: a tutorial. Analytica chimica acta, 185, 1-17.
[30] Geman S, Geman D. (1993). Stochastic relaxation, gibbs distributions and the bayesian restoration of images. Journal of Applied Statistics 20(5-6):25–62.
[31] Genin, E., Hannequin, D., Wallon, D., Sleegers, K., Hiltunen, M., Combarros, O., Bullido, M.J., Engelborghs, S., De Deyn, P., Berr, C. & Pasquier, F., 2011. APOE and Alzheimer disease: a major gene with semi-dominant inheritance. Molecular psychiatry, 16(9), p.903.
[32] Glahn, D.C., Winkler, A.M., Kochunov, P., Almasy, L., Duggirala, R., Carless, M.A., Curran, J.C., Olvera, R.L., Laird, A.R., Smith, S.M. & Beckmann, C.F., (2010). Genetic control over the resting brain. Proceedings of the National Academy of Sciences, 107(3), pp.1223-1228.
[33] Green PJ, Richardson S. (2002). Hidden markov models and disease mapping. Journal of the American statistical association 97(460):1055–1070.
[34] Greenlaw, K., Szefer, E., Graham, J., Lesperance, M., Nathoo, F.S. & Alzheimerś Disease Neuroimaging Initiative, (2017). A Bayesian group sparse multi-task regression model for imaging genetics. Bioinformatics, 33(16), pp.2513-2522.
[35] Hibar, D.P., Stein, J.L., Kohannim, O., Jahanshad, N., Saykin, A.J., Shen, L.,

Kim, S., Pankratz, N., Foroud, T., Huentelman, M.J. & Potkin, S.G., (2011). Voxelwise gene-wide association study (vGeneWAS): multivariate gene-based association testing in 731 elderly subjects. Neuroimage, 56(4), pp.1875-1891.

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

[37] Huang, J. & Ma, S. (2008). Adaptive Lasso for sparse high-dimensional regression models. Statistica Sinica, 1603–1618.

[38] Ingber L. (1993). Simulated annealing: Practice versus theory. Mathematical and com- puter modelling 18(11):29–57.

[39] Judge, G. G., Bock, M. E., & Bock, M. E. (1978). The statistical implications of pre-test and Stein-rule estimators in econometrics (Vol. 25). North-Holland.

[40] Ju QW, et al., (2011). Aco-based bw algorithm for parameter estimation of hidden markov models. International Journal of Computer Applications in Technology 41(3/4):281–286.

[41] Jung, K., Friston, K.J., Pae, C., Choi, H.H., Tak, S., Choi, Y.K., Park, B., Park, C.A., Cheong, C. & Park, H.J., (2018). Effective connectivity during working memory and resting states: a DCM study. NeuroImage, 169, pp.485-495.

[42] Kass, R.E. & Raftery, A.E., (1995). Bayes factors. Journal of the american statistical association, 90(430), pp.773-795.

[43] Kato Z, Pong TC (2006). A markov random field image segmentation model for color textured images. Image and Vision Computing 24(10):1103–1114.

[44] Kim E., Park S., Kim H. (2000). A genetic algorithm-based segmentation of markov random field modeled images. IEEE Signal processing letters 7(11):301–303.

[45] Kim, Y., Choi, H., & Oh, H. S. (2008). Smoothly clipped absolute deviation on high dimensions. Journal of the American Statistical Association, 103(484), 1665–1673.

[46] Knight, K., & Fu, W. (2000). Asymptotics for lasso-type estimators. Annals of statistics, 1356–1378.

[47] Kuznetsova, A., Brockhoff, P.B. and Christensen, R.H.B., 2017. lmerTest package: tests in linear mixed effects models. Journal of Statistical Software, 82(13).

[48] Laird, N. M., & Ware, J. H. (1982). Random-effects models for longitudinal data. Biometrics, 963–974.

[49] LeCun Y., Bengio Y., Hinton G. (2015). Deep learning. nature 521(7553):436.

[50] Leng, C., Lin, Y., & Wahba, G. (2006). A note on the Lasso and related procedures in model selection. Statistica Sinica, 16, 1273?- 1284.

[51] Li, B., Daunizeau, J., Stephan, K. E., Penny, W., Hu, D., & Friston, K. (2011). Generalised filtering and stochastic DCM for fMRI. neuroimage, 58(2), 442-457.
[52] Li, R., Yu, J., Zhang, S., Bao, F., Wang, P., Huang, X., & Li, J. (2013). Bayesian network analysis reveals alterations to default mode network connectivity in individuals at risk for Alzheimer's disease. PLoS One, 8(12), e82104.
[53] Lisawadi, S., Kashif Ali Shah, M., & Ejaz Ahmed, S. (2016). Model selection and post estimation based on a pretest for logistic regression models. Journal of Statistical Computation and Simulation, 86(17), 3495–3511.
[54] Li SZ (2009). Markov random field modeling in image analysis. Springer Science & Business Media.
[55] Lindquist, M.A. & Sobel, M.E., (2016). Effective connectivity and causal inference in neuroimaging. Handbook of Neuroimaging Data Analysis, p.419.
[56] Liu, C. C., Kanekiyo, T., Xu, H., & Bu, G. (2013). Apolipoprotein E and Alzheimer disease: risk, mechanisms and therapy. Nature Reviews Neurology, 9(2), 106.
[57] Liu, J., & Calhoun, V. D. (2014). A review of multivariate analyses in imaging genetics. Frontiers in neuroinformatics, 8, 29.
[58] Longford, N. T. (1993). Regression analysis of multilevel data with measurement error. British Journal of Mathematical and Statistical Psychology, 46(2), 301–311.
[59] Liu, K. (2003). Using Liu-type estimator to combat collinearity. Communications in Statistics-Theory and Methods, 32(5), 1009-1020.
[60] Lu, Z.H., Khondker, Z., Ibrahim, J.G., Wang, Y., Zhu, H. & Alzheimer?s Disease Neuroimaging Initiative, (2017). Bayesian longitudinal low-rank regression models for imaging genetic data from longitudinal studies. NeuroImage, 149, pp.305-322.
[61] Luo, C., Li, Q., Lai, Y., Xia, Y., Qin, Y., Liao, W., Li, S., Zhou, D., Yao, D. & Gong, Q., (2011). Altered functional connectivity in default mode network in absence epilepsy: a resting-state fMRI study. Human brain mapping, 32(3), pp.438-449.
[62] Luo, X., Li, K., Jia, Y.L., Zeng, Q., Jiaerken, Y., Qiu, T., Huang, P., Xu, X., Shen, Z., Guan, X. & Zhou, J., (2019). Altered effective connectivity anchored in the posterior cingulate cortex and the medial prefrontal cortex in cognitively intact elderly APOE e4 carriers: a preliminary study. Brain imaging and behavior, 13(1), pp.270-282.
[63] Maniezzo V., Carbonaro A. (2002). Ant colony optimization: an overview. In: Essays and surveys in metaheuristics, Springer, pp 469–492.
[64] Marchini, J., Howie, B., Myers, S., McVean, G. & Donnelly, P., (2007). A new

multipoint method for genome-wide association studies by imputation of genotypes. Nature genetics, 39(7), p.906.

[65] McKhann, G., Drachman, D., Folstein, M., Katzman, R., Price, D., & Stadlan, E. M. (1984). Clinical diagnosis of Alzheimer's disease: Report of the NINCDS-ADRDA Work Group under the auspices of Department of Health & Human Services Task Force on Alzheimer's Disease. Neurology, 34(7), 939-939.

[66] Morris, J.S., 2015. Functional regression. Annual Review of Statistics and Its Application, 2, pp.321-359.

[67] Muller, H.G., 2008. Functional modeling of longitudinal data. Longitudinal data analysis, 1, pp.223-252.

[68] Naj AC, Beecham GW, Martin ER, Gallins PJ, Powell EH, Konidari I, Whitehead PL, Cai G, Haroutunian V, Scott WK, Vance JM, (2010). Dementia revealed: novel chromosome 6 locus for late-onset Alzheimer disease provides genetic evidence for folate-pathway abnormalities. PLoS genetics, 6(9).

[69] Nathoo, F. S., Kong, L., Zhu, H., & Alzheimer's Disease Neuroimaging Initiative. (2019). A review of statistical methods in imaging genetics. Canadian Journal of Statistics, 47(1), 108-131.

[70] Nie, Y., Opoku, E., Yasmin, L., Song, Y., Wang, J., Wu, S., ... & Nathoo, F. S. (2020). Spectral dynamic causal modelling of resting-state fMRI: an exploratory study relating effective brain connectivity in the default mode network to genetics. Statistical Applications in Genetics and Molecular Biology, 1(ahead-of-print).

[71] Ouadfel S.,Batouche M. (2003). Ant colony system with local search for markov random field image segmentation. In: Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429), IEEE, vol 1, pp I–133.

[72] Park, T., & Casella, G. (2008). The bayesian lasso. Journal of the American Statistical Association, 103(482), 681-686.

[73] Pericak-Vance MA, Grubber J, Bailey LR, Hedges D, West S, Santoro L, Kemmerer B, Hall JL, Saunders AM, Roses AD, Small GW. Identification of novel genes in late-onset Alzheimer's disease. Experimental gerontology. 2000 Dec 1;35(9-10):1343-52.

[74] Penny, W. D., Friston, K. J., Ashburner, J. T., Kiebel, S. J., & Nichols, T. E. (Eds.). (2011). Statistical parametric mapping: the analysis of functional brain images. Elsevier.

[75] Pillai, K.C.S., 1955. Some new test criteria in multivariate analysis. The Annals of Mathematical Statistics, 26(1), pp.117-121.

[76] Purcell, S., Neale, B., Todd-Brown, K., Thomas, L., Ferreira, M. A., Bender, D., D., Maller, J., Sklar, P., De Bakker, P.I., Daly, M.J. & Sham, P. C. (2007). PLINK: a tool set for whole-genome association & population-based linkage analyses. The American journal of human genetics, 81(3), 559-575.

[77] Raheem, S. E., Ahmed, S. E., & Doksum, K. A. (2012). Absolute penalty and shrinkage estimation in partially linear models. Computational Statistics & Data Analysis, 56(4), 874–891.

[78] Razi, A., Kahan, J., Rees, G., & Friston, K. J. (2015). Construct validation of a DCM for resting state fMRI. Neuroimage, 106, 1-14.

[79] Robert C., Casella G. (2013). Monte Carlo statistical methods. Springer Science & Business Media.

[80] Rouder, J.N., Morey, R.D., Speckman, P.L. & Province, J.M., 2012. Default Bayes factors for ANOVA designs. Journal of Mathematical Psychology, 56(5), pp.356-374.

[81] Sagayam KM, Hemanth DJ (2018). Abc algorithm based optimization of 1-d hidden markov model for hand gesture recognition applications. Computers in Industry 99:313–323.

[82] Salzenstein F, Pieczynski W. (1997). Parameter estimation in hidden fuzzy markov random fields and image segmentation. Graphical models and image processing 59(4):205–220.

[83] Sharaev, M.G., Zavyalova, V.V., Ushakov, V.L., Kartashov, S.I. & Velichkovsky, B.M., (2016). Effective connectivity within the default mode network: dynamic causal modeling of resting-state fMRI data. Frontiers in human neuroscience, 10, p.14.

[84] Schelldorfer, J., Buhlmann, P., & DE GEER, S. V. (2011). Estimation for high-dimensional linear mixed-effects models using l_1-penalization. Scandinavian Journal of Statistics, 38(2), 197–214.

[85] Shen Q, Faraway J, (2004). An F test for linear models with functional responses. Statistica Sinica, 1:1239-57.

[86] Shen Y, Kiatsupaibul S,Zabinsky ZB, Smith RL(2007). Analytically derived cooling schedule for simulated annealing. Journal of Global Optimization 38(3):333–365.

[87] Singer S., Nelder J. (2009). Nelder-mead algorithm. Scholarpedia 4(7):2928.

[88] Smith, S. M. (2002). Fast robust automated brain extraction. Human brain mapping, 17(3), 143-155.

[89] Stein, J.L., Hua, X., Lee, S., Ho, A.J., Leow, A.D., Toga, A.W., Saykin, A.J., Shen, L., Foroud, T., Pankratz, N. & Huentelman, M.J., (2010). Voxelwise genome-

wide association study (vGWAS). neuroimage, 53(3), pp.1160-1174.
[90] Stingo, F.C., Guindani, M., Vannucci, M. & Calhoun, V.D., (2013). An integrative Bayesian modeling approach to imaging genetics. Journal of the American Statistical Association, 108(503), pp.876-891.
[91] Song, Y., Ge, S., Cao, J., Wang, L., & Nathoo, F. S. (2019). A Bayesian Spatial Model for Imaging Genetics. arXiv preprint arXiv:1901.00068.
[92] Song Y., Nathoo F., Babul A. (2019). A potts-mixture spatiotemporal joint model for combined magnetoencephalography and electroencephalography data. Canadian Journal of Statistics DOI: 10.1002/cjs.11519.
[93] Szefer, E., Lu, D., Nathoo, F., Beg, M.F. & Graham, J., (2017). Multivariate association between single-nucleotide polymorphisms in Alzgene linkage regions and structural changes in the brain: discovery, refinement and validation. Statistical applications in genetics and molecular biology, 16(5-6), pp.367-386.
[94] Szeliski R., Zabih R., Scharstein D., Veksler O., Kolmogorov V., Agarwala A, Tappen M, Rother C. (2008). A comparative study of energy minimization methods for markov random fields with smoothness-based priors. IEEE transactions on pattern analysis and machine intelligence 30(6):1068–1080.
[95] Team, R. C. (2018). R development core team. r: A language and environment for statistical computing. r foundation for statistical computing, vienna, austria; 2014.
[96] Thompson, P.M., Ge, T., Glahn, D.C., Jahanshad, N. & Nichols, T.E., (2013). Genetics of the connectome. Neuroimage, 80, pp.475-488.
[97] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267–288.
[98] Tran, M. N. (2011). The loss rank criterion for variable selection in linear regression analysis. Scandinavian Journal of Statistics, 38(3), 466–479.
[99] Twisk, J. W. R., Kemper, H. C. G., & Mellenbergh, G. J. (1995). Longitudinal development of lipoprotein levels in males and females aged 12?28 years: the Amsterdam Growth and Health Study. International journal of epidemiology, 24(1), 69–77.
[100] Wang, H., & Leng, C. (2007). Unified LASSO estimation by least squares approximation. Journal of the American Statistical Association, 102(479), 1039–1048.
[101] Wu FY (1982). The potts model. Reviews of modern physics 54(1):235.
[102] Wu, X., Li, R., Fleisher, A.S., Reiman, E.M., Guan, X., Zhang, Y., Chen, K. &

Yao, L., (2011). Altered default mode network connectivity in Alzheimer's disease–a resting functional MRI and Bayesian network study. Human brain mapping, 32(11), pp.1868-1881.

[103] Xu, J., Yin, X., Ge, H., Han, Y., Pang, Z., Liu, B., Liu & Friston, K. (2016). Heritability of the effective connectivity in the resting-state default mode network. Cerebral cortex, 27(12), 5626-5634.

[104] Yang X., Zheng XQ, Lv LN (2012). A spatiotemporal model of land use change based on ant colony optimization, markov chain and cellular automata. Ecological Modelling 233:11–19.

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

[106] Yuzbasi, B., Ahmed, S. E., & Gungor, M. (2017). Improved penalty strategies in linear regression models. REVSTAT-Statistical Journal, 15(2), 251–276.

[107] Zhang L., Seitz SM (2007). Estimating optimal parameters for mrf stereo from a single image pair. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(2):331–342.

[108] Yan, H., Zhang, Y., Chen, H., Wang, Y., & Liu, Y. (2013). Altered effective connectivity of the default mode network in resting-state amnestic type mild cognitive impairment. Journal of the International Neuropsychological Society, 19(4), 400-409.

[109] Zeidman, P., Jafarian, A., Corbin, N., Seghier, M.L., Razi, A., Price, C.J. & Friston, K.J., (2019). A guide to group effective connectivity analysis, part 1: First level analysis with DCM for fMRI. NeuroImage.

[110] Zhong, Y., Huang, L., Cai, S., Zhang, Y., von Deneen, K. M., Ren, A., Junchan, R., & Alzheimer's Disease Neuroimaging Initiative. (2014). Altered effective connectivity patterns of the default mode network in Alzheimer's disease: an fMRI study. Neuroscience letters, 578, 171-175.

[111] Zhang, Z., Descoteaux, M., Zhang, J., Girard, G., Chamberland, M., Dunson, D., Srivastava, A. & Zhu, H., (2018). Mapping population-based structural connectomes. NeuroImage, 172, pp.130-145.

[112] Zhang, Z., Allen, G.I., Zhu, H. & Dunson, D., (2019). Tensor network factorizations: Relationships between brain structural connectomes and traits. NeuroImage, 197, pp.330-343.

[113] Zhu, H., Khondker, Z., Lu, Z. & Ibrahim, J.G., (2014). Bayesian generalized low rank regression models for neuroimaging phenotypes and genetic markers. Journal of

the American Statistical Association, 109(507), pp.977-990.