BAYESIAN ANALYSIS OF ULTRA-HIGH DIMENSIONAL NEUROIMAGING DATA

Michelle Ferreira Miranda

A dissertation submitted to the faculty at the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistics and Operations Research.

Chapel Hill
2014

Approved by:
Joseph G. Ibrahim
Hongtu Zhu
Yufeng Liu
Marc Niethammer
Haipeng Shen
ABSTRACT

Michelle Ferreira Miranda: Bayesian analysis of ultra-high dimensional neuroimaging data
(Under the direction of Joseph G. Ibrahim and Hongtu Zhu)

Medical imaging technologies have been generating extremely complex data sets. This dissertation makes further contributions to the development of statistical tools motivated by modern biomedical challenges. Specifically we develop methods to characterize varying associations between ultra-high dimensional imaging data and low-dimensional clinical outcomes.

The first part of this dissertation is motivated by the major limitations faced by traditional voxel-wise models, where voxels are commonly treated as independent units, and the assumption of Gaussian distribution of the neuroimaging measurements is usually flawed. We develop a class of hierarchical spatial transformation models to model the spatially varying associations between imaging measurements in a three-dimensional (3D) volume (or 2D surface) and a set of covariates. The proposed approach include a spatially varying Box-Cox transformation model and a Gaussian Markov random field model.

The second part is motivated by the challenges faced by ultra-high dimensional datasets. In particular, we introduce a method to predict clinical outcomes from ultra-high dimensional covariates. The proposed models reduce dimensionality to a manageable level and further apply dimension reduction techniques, e.g. principal components analysis and tensor decompositions to extract and select low-dimensional important features.
ACKNOWLEDGEMENTS

I would like to first thank my advisors Hongtu Zhu and Joseph Ibrahim. Joseph, thank you for teaching me Bayesian statistics in such a beautiful and inspiring way. Hongtu, thank you for the lessons in imaging analysis and for your determination in making things work. I am always amazed by your ability to manage your students, knowing in details what we are doing, even though we are so many. Both of you contributed tremendously to my path as a researcher and I am forever grateful for that.

Many others contributed to this achievement in a direct or indirect way:

• Susan and Diana, for the many hours spent at the Looking Glass studying for the comps;

• Eric and Dominik, for being my very first friends, for the weekly beer nights, for the dancing nights;

• Sean, for making me laugh with your funny stories;

• The quartet on Main street, Susan, Diana, Sean and Eric for the trips, countless board games and dinner nights. I am grateful for having such amazing friends!!!

• My friend and first roommate Fernanda, who taught me so much about kindness and became part of my family;

• Julien and Marc, for the weekly wine bar nights;

• My salsa friends Derek and Chris, who helped me to be sane during this last couple of years;

• Nilay, Jenny, Lucile and Amit for your friendship;
• My Brazilian friends, who made my first months here so enjoyable;

• My family, who always sent me their love in prayers and positive thinking;

• Bruno, for always being the one I could count for help;

• My sister Izabela, who is always ready to give me the right advice;

• My friends Arianna, Bia, Felipo, Kelly e Taty who have been my rock for decades.

Thank you very much to my best friend Susan Wei, for helping me with my writing, listening to my frustrations and excitement, and for always being there. You always inspired me and pushed me forward!!

Thank you Sepehr, for being my English teacher, my proof reader and for helping me to cross the ending line. Your support and companionship were essential during my final steps.

Finally, thanks to my parents, for their enormously sacrifice to give me a good education. I owe everything to you.
PREFACE

# TABLE OF CONTENTS

LIST OF TABLES ................................................................. x

LIST OF FIGURES ............................................................. xi

CHAPTER 1: INTRODUCTION ......................................................... 1

  1.1 Voxel-wise models ...................................................... 1

  1.2 Low-rank regression models ........................................ 2

  1.3 Outline of Thesis ..................................................... 3

CHAPTER 2: BAYESIAN SPATIAL TRANSFORMATION MODELS .......... 4

  2.1 Introduction .............................................................. 4

  2.2 Model ................................................................. 6

     2.2.1 Model Description ............................................. 6

     2.2.2 Priors ............................................................ 9

     2.2.3 Posterior Computation ...................................... 9

  2.3 Simulation Study ..................................................... 11

  2.4 Application to the ADHD dataset .................................. 15

  2.5 Discussion ............................................................. 21

CHAPTER 3: TENSOR PARTITION REGRESSION MODELS ............... 22

  3.1 Introduction ............................................................. 22

  3.2 Methodology .......................................................... 25

     3.2.1 Preliminaries .................................................. 25

     3.2.2 Tensor Partition Regression Models ...................... 26

     3.2.3 Prior Distributions ........................................... 28

     3.2.4 Posterior Inference .......................................... 29
LIST OF TABLES

3.1 Root mean squared error for 3 different image modalities. The Bayesian decomposition outperforms the alternating least squares in each scenario. There is a smaller error measurement with an increase of the rank $R$. ........ 34

4.1 Deviance information criteria for the partition models $N_s = 8$ and $N_s = 18$ and the non-partition model. Based on the criteria, the models with $N_s = 18$ are preferred. The symbol ** indicates the models that did not converge. ........ 56

4.2 Deviance information criteria for the partition models $N_s = 8$ and $N_s = 24$ and the non-partition model. Based on the criteria, the partition models are preferred. ................................................................. 57

A.1 Sensitivity analysis for $\lambda$ indicating the percentages of voxels, whose Geweke diagnosis statistics are smaller than 1.96, according to the Geweke diagnosis statistics (Geweke, 1992) for each scenario considered. ................. 64
LIST OF FIGURES

2.1 Simulation results: the true $\Lambda = \{\lambda_d, d \in D\}$ pattern in the left panel and the estimated pattern in the right panel. Estimated image is smoother compared with the true image due to the nature of the uniform distribution assumed a priori. This figure appears in color in the electronic version of this article. ........................................... 13

2.2 Simulation results on comparison of STM, GMRF with no transformation, and the voxel-wise linear model. Panels (a)-(d) represent the pattern of $\beta$ used to generate the images; panels (e)-(h) represent the estimated $\beta$ obtained from the least squares estimator in Matlab; panels (i)-(l) represent the posterior mean of $\beta$ obtained by fitting a GMRF model with no transformation; and panels (m)-(p) are the posterior mean of $\beta$ obtained from our STM. The inclusion of the transformation parameter substantially improves the estimation of the true underlying pattern. This figure appears in color in the electronic version of this article. ........................................... 14

2.3 Trace plots for $\beta$, $\tau$ and $\lambda$ for a randomly generated voxel. The results are for a 1000 iterations of the MCMC algorithm and a burn-in sample of 50. The trace plots indicate a fast convergence of the algorithm, confirming its efficiency and good mixing properties. This figure appears in color in the electronic version of this article. ............... 15

2.4 White matter RAVENS map for two randomly selected children from the ADHD study. The image from subject b shows a higher brightness inside the square, reflecting the fact that for the brain of subject b, relatively more white matter was forced to fit the same template (panel (c)) at that particular region. This figure appears in color in the electronic version of this article. ........................................... 17

2.5 ADHD data analysis results: normal probability plots of sixteen random voxels revealing that the imaging measurements extracted from the RAVENS map deviate from the Gaussian distribution. This figure appears in color in the electronic version of this article. .......... 18
2.6 ADHD data analysis results: selected slices showing the estimated \( \Lambda \) for the imaging data obtained from the white matter RAVENS map. Panels (a)-(c) represent respectively, a coronal, sagittal and axial view of selected slices of the brain. The line indicates where the coronal and sagittal slices meet the plane in (c); panel (d) shows the same axial slice as in (c) and represents the location in the brain where \( \Lambda = \{\lambda_d, d \in D\} \) are different from 1, based on a 95% credible interval. This figure appears in color in the electronic version of this article.

2.7 ADHD data analysis results. Top panels: significant regions in the brain where there exists a morphological difference between children with ADHD and children who do not have the disorder, based on a 95% credible interval. Panel (a) is a selected axial slice of the STM estimate overlaid on the Jacob template; (b) is the same selected slice showing the estimates of the spatial model with the transformation parameters \( \Lambda \) fixed and equal to 1 for all voxels also overlaid on the template; (c) and (d) are, respectively, the results of a 3D rendering of the STM and of the no transformation model both overlaid on the Jacob template. Bottom panel: (e) shows selected axial slices of the STM estimates overlaid on the template. Highlighted areas show the significant regions in the brain where there exists a morphological difference between children with ADHD and children who do not have the disorder. This figure appears in color in the electronic version of this article.

3.1 Figure copied from (Kolda and Bader, 2009b). Panel (a) illustrates the CP decomposition of a three way array as a sum of \( R \) components of rank-one tensors, i.e. \( X \approx \sum_{r=1}^{R} a_r \circ b_r \circ c_r \).

3.2 Trace plots in 9 randomly chosen voxels in the white matter RAVENS map by using Bayesian tensor decomposition with \( R = 20 \). The trace plots indicate that the Markov chain converges after around 1000 iterations.

3.3 Simulation 1 for Bayesian tensor decomposition results: panel (a): DTI image; panel (b): white matter RAVENS map; and panel (c): T1-weighted image. In each row, the first image represents an axial slice of the original image and from left to right we have the decomposed images for ranks \( R = 5, 10, 20 \), respectively.
3.4 Results of the 2-D imaging example: (a): $\lambda_0(0)$; (b): $\lambda_0(1) - \lambda_0(0)$; (c): the simulated image from a randomly selected subject from group 1; (d): the estimated projection $P$. Panels (e) and (g) are the posterior mean of the quantity $P = \|\Lambda; A^{(1)}, A^{(2)}, B\|$ for BTRM with $S = 4$ and the no-partition model with $S = 1$, respectively. Panels (f) and (h) are, respectively, the 95% credible interval of $P$ for BTRM($S = 4$) and BTMR($S = 1$) revealing the true underlying location, where differences between both groups exist.

3.5 Results of the 3-D image example. In each panel, we show axial views of the 2 true signal regions and a 3D render of the results overlaid on the template $G_0$ from left to right. Panel (a) show the true effect signal $\lambda_0$. Panels (b) and (c) present the posterior mean of the quantity $P = \|\Lambda; A^{(1)}, A^{(2)}, A^{(3)}, B\|$ for TPRMs with $S = 8$ and with $S = 1$, respectively. Panel (d) shows the results for fPCA. Both TPRMs are able to recover $\lambda_0$, whereas fPCA does not.

3.6 DIC results of 100 simulated examples. Straight line indicates the non-partition model TPRM($S = 1$) and dotted line indicates partition TPRM($S = 8$). The later model is preferred in the majority of generated datasets.

3.7 Axial slices of the posterior mean estimates for the projection $P = \|\lambda; A^{(1)}, A^{(2)}, A^{(3)}, B\|$ for the TPRM (panel (b)) and the estimated projection for fPCA (panel (a)). The colors indicate how strong the differences are between children with ADHD and control. The 3 slices on the right indicate a possible right frontal lobe region, in both models. A strong difference seems to exist in the left frontal lobe and left parietal lobe when observing the TPRM model but not in the fPCA model. The chain was generated with 5000 iterations and the burn-in period is of 3000.

3.8 ADHD data analysis results. Panels (a) and (b) are, respectively, the results of a 3D rendering of the 90% and 95% credible intervals for the projection $P = \|\lambda; A^{(1)}, A^{(2)}, A^{(3)}, B\|$ both overlaid on the Jacob template. Panel (c) and (d) are selected axial slices of the credible intervals (95% and 90%, respectively). We detect two large regions of interest, where morphological differences exist, including the right frontal lobe and left parietal lobe.

4.1 A 3D image $I$ partitioned into subimages $I^{(s)}$.
4.2 Trace plots of the projection $\mathcal{P} = \bigcup s P^{(s)}$, where $P^{(s)} = \left(A^{(s)}\right)^T b^{(s)}$, for 9 random voxels of the partition model with $N_s = 8$, $R = 50$, after a burn-in period of 500. The plots indicate a fast convergence of the chain. ........................................ 54

4.3 Simulation results for the Bayesian sparse partition factor model (BSPFM) with $N_s = 18$, $N_s = 8$ and $N_s = 1$. Panel (a) shows the true underlying signal $X_1$ overlaid on the template $X_0$; panel (b) shows the estimated projection for the PCA model and panels (c)-(e) are posterior mean of the projection for the partition and non-partition model, all results overlaid on the template $X_0$. .................. 55

4.4 ADHD data analysis results for the Bayesian partition partial factor model (BPPFM) with $N_s = 24$ partitions and rank $R = 200$. Panel (b) shows the results of the posterior mean of the projection $\mathcal{P}$ and panel (a) shows the estimated projection for the 2-stage PCA model. The colors indicate the strength of the differences between children with ADHD and control. ........................................ 58

4.5 ADHD data analysis results for the Bayesian partition partial factor model (BPPFM) with $N_s = 24$ partitions and rank $R = 200$. Panel (a) and (b) show the results of a 3D rendering of the 95% and 90% credible intervals for the projection overlaid on the Jacob template, respectively; panels (c) and (d) are selected axial slices of the 90% and 95% credible interval for the projection, respectively. .................. 59

A.1 Sensitivity analysis of $\Lambda$. Panels (a)-(d) represent the true pattern of $\beta$ used to generate the images; (e)-(h): the posterior means of $\beta$ obtained with $(a, b) = (-2.0, 2.5)$; (i)-(l): the posterior means of $\beta$ obtained with $(a, b) = (-3.0, 3.0)$; and (m)-(p): the posterior means of $\beta$ with $(a, b) = (-3.5, 3.5)$. .............................. 64

A.2 Trace plots of $\nu_k$ for $k = 0, \ldots, 3$ after a burn-in of 50 iterations and a total of 1000 MCMC iterations under the three scenarios of $(a, b)$. Rows 1-3 correspond to $(a, b) = (-2.0, 2.5)$, $(a, b) = (-3.0, 3.0)$, and $(a, b) = (-3.5, 3.5)$, respectively. ...................... 65

A.3 Trace plots of $\beta$, $\tau_{\sigma}$ and $\lambda$ for the scenario $(a, b) = (-3.5, 3.5)$ at 4 random selected voxels. The results show fast convergence of the MCMC chain for all parameters. ...................... 66

A.4 Posterior estimates of $\beta$ for different values of $\phi_k$: (a)-(d): $\phi_k = 0.01$; (e)-(h) $\phi_k = 0.1$; (i)-(l) $\phi_k = 1$; and (m)-(p): $\phi_k = 100$. .................. 68

A.5 Posterior estimates of $\beta$ under different specifications of $H_k$. Panels (a)-(d): case (H.1); (e)-(p): case (H.2) for $r_0 = 0, 4, 6$. ................. 69
A.6 The posterior estimates of $\beta$ for the true model with $\lambda_d = 1$ for all $d \in D$. ................................................................. 70

A.7 The posterior estimated image $\hat{\lambda} = \{\hat{\lambda}_d : d \in D\}$ for the true underlying model with $\lambda_d = 1$ for all $d \in D$. ................................. 70

A.8 The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images corresponding to the intercept $\beta_0$ are shown from the left to the right, respectively. ......................... 71

A.9 The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images corresponding to the gender $\beta_1$ are shown from the left to the right, respectively. ......................... 71

A.10 The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images corresponding to the age $\beta_2$ are shown from the left to the right, respectively. ................................. 71

A.11 The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images for the ADHD status $\beta_3$ are shown from the left to the right, respectively. ................................. 72

A.12 Illustration of the one-way dependency structure of the functions. ............... 73

A.13 Simulation results obtained by running MainSTM.m. Panels (a)-(d) represent the pattern of $\beta$ used to generate the images; panels (e)-(h) are estimated $\beta$ obtained from the least squares estimator in Matlab; panels (i)-(l) are the posterior mean of $\beta$ obtained from our STM. ..... 76

A.14 Simulation results: the true $\Lambda = \{\lambda_d, d \in D\}$ pattern in the left panel and the estimated pattern in the right panel. Estimated image is smoother compared with the true image due to the nature of the uniform distribution assumed a priori. ......................................... 76

A.15 Illustration of the one-way dependency structure of the functions. ............... 77
CHAPTER 1: INTRODUCTION

In the past few decades, medical imaging technologies have been improving and a number of expanding modalities are creating impressively accurate and detailed images for less invasive and more precise methods of diagnosis. Although these advances are allowing researchers and clinicians to gain insights of unprecedented quality on the cerebral anatomical structures, connectivity patterns and functional properties, it also brings the challenge of developing automatic methods to categorize and classify brain responses, identify abnormalities in the brain, understand mental thoughts, and reveal important effects of environmental and genetic factors on brain structure and function, among many others.

To face these challenges, classical statistical tools need to be adapted to more complex data structures of typical neuroimaging studies. These data usually take the form of multidimensional arrays with intricate spatial correlation and functional changes that evolve over time. This dissertation makes further contributions to the development of statistical tools motivated by these challenges. More specifically, we develop methods to characterize varying associations between ultra-high dimensional imaging data and low-dimensional response variables.

The tools originated by this research fall into two categories of solutions to establish association between multidimensional images and clinical outcomes and are described as follows.

1.1 Voxel-wise models

Voxel-wise models consist of fitting a linear model at each voxel, modeling the neuroimaging measurement as a function of other clinical or genetic or both variables. These models are available at most software platforms but face some limitations, in particular: (i) the assumption of Gaussian distribution of the neuroimaging measurements is usually
flawed; (ii) the voxels are treated as independent units and the spatial structure of the brain is ignored until the very last stage, when a correction is made on the test statistics.

Chapter 2 simultaneously addresses these issues by developing a class of spatial transformation models (STM) to model the spatially varying associations between imaging measurements in a three-dimensional (3D) volume (or 2D surface) and a set of covariates. The proposed STM includes a spatially varying Box-Cox transformation model for dealing with the issue of non-Gaussian distributed imaging data and a Gaussian Markov random field model to incorporate spatial smoothness of the imaging data.

1.2 Low-rank regression models

A different perspective is to consider imaging data to predict a scalar response. However, a typical neuroimaging study, with magnetic resonance imaging (MRI), produces images of size $256 \times 256 \times 256$, with approximately 16.5 million voxels. Most models are compromised by this ultra-high dimensionality.

A possible solution is to integrate supervised (or unsupervised) dimension reduction techniques with various standard regression models. Given the ultra-high dimension of imaging data, however, it is imperative to use some dimension reduction methods to extract and select “low-dimensional” important features.

The dimension reduction step is often performed by applying principal component analysis (PCA) or high-order tensor decompositions, e.g., CP or Tukey. Further, the top components extracted on the decomposition are used to predict a clinical outcome. A crucial assumption is that the leading components obtained from these decompositions capture the most important features of the multi-dimensional array. However, neuroimaging data are extremely noisy, and regions affecting the outcome are small and often clustered together. As a consequence, it is likely that “effect” regions will not be noticed.

In chapters 3 and 4, we propose models that account for the key features of neuroimaging data: low signal to noise ratio and the spatially clustered effect, while simultaneously reducing data dimensionality. The central idea is as follows. Consider the aforementioned
MRI image of size $256 \times 256 \times 256$, and assume we partition the image into $16^3 = 4,096$ subarrays of size $16 \times 16 \times 16$. If we reduce each $16 \times 16 \times 16$ subarray into a small number of components, not only the total number of reduced features drop to a manageable level, but also we are more likely to capture small clustered effect regions. Both solutions are formulated as supervised hierarchical models, and an efficient Markov chain Monte Carlo algorithm is developed.

1.3 Outline of Thesis

This chapter presents the motivation and context for the statistical methodologies developed henceforth. The remainder of the thesis is divided into two natural parts. The first consists of a voxel-wise model. The second part consists of two stand-alone papers on supervised low-rank modeling. Appendices for the papers appear separately at the very end.
CHAPTER 2: BAYESIAN SPATIAL TRANSFORMATION MODELS

2.1 Introduction

The emergence of various imaging techniques has enabled scientists to acquire high-dimensional imaging data to closely explore the function and structure of the human body in various imaging studies. Several common imaging techniques include magnetic resonance image (MRI), functional MRI, diffusion tensor image (DTI), positron emission tomography (PET), and electroencephalography (EEG), among many others. These imaging studies, such as the Alzheimer’s Disease Neuroimaging Initiative (ADNI), are essential to understanding the neural development of neuropsychiatric and neurodegenerative disorders, the normal brain and the interactive effects of environmental and genetic factors on brain structure and function, among others. A common feature of all these imaging studies is that they have been generating many very high dimensional and complex data sets.

There is a great interest in developing voxel-wise methods to characterize varying associations between high-dimensional imaging data and low-dimensional covariates (Friston, 2007; Lindquist, 2008; Lazar, 2008; Li et al., 2011). These methods usually fit a general linear model to the imaging data from all subjects at each voxel as responses and clinical variables, such as age and gender, as predictors. Subsequently, a statistical parametric map of test statistics or \( p \)-values across all voxels (Lazar, 2008; Worsley et al., 2004) is generated. Several popular neuroimaging software platforms, such as statistical parametric mapping (SPM) (www.fil.ion.ucl.ac.uk/spm/) and FMRIB Software Library (FSL) (www.fmrib.ox.ac.uk/fsl/), include these voxel-wise methods as their key statistical tools.

These voxel-wise methods have several major limitations. First, the general linear model used in the neuroimaging literature usually assumes that the imaging data conform to a Gaussian distribution with homogeneous variance (Ashburner and Friston, 2000; Wager
et al., 2005; Worsley et al., 2004; Zhu et al., 2009). This distributional assumption is important for the valid calculation of \( p \)-values in conventional tests (e.g., F test) that assess the statistical significance of parameter estimates. Moreover, methods of random field theory (RFT) that account for multiple statistical comparisons depend strongly on the parametric assumptions, as well as several additional assumptions (e.g., smoothness of autocorrelation function).

Second, the Gaussian assumption is known to be flawed in many imaging datasets (Ashburner and Friston, 2000; Salmond et al., 2002; Luo and Nichols, 2003; Zhu et al., 2009). It is common to use a Gaussian kernel with the full-width-half-max (FWHM) in the range of 8-16mm to account for registration errors, to make the data normally distributed and to integrate imaging signals from a region, rather than from a single voxel. However, recent research has shown that varying filter sizes in the smoothing methods can result in different statistical conclusions about the activated and deactivated regions, and spatial smoothing biases the localization of brain activity. Thus, it can result in misleading scientific inferences (Jones et al., 2005; Sacchet and Knutson, 2012).

Third, as pointed out in the literature (Li et al., 2011; Yue et al., 2010), the voxel-wise methods treat all voxels as independent units, and thus they ignore important spatial smoothness observed in imaging data. Several promising methods have been proposed to accommodate the varying amount of smoothness across the imaging space by using function-on-scalar regression in the functional data analysis framework (Zhu et al., 2012; Ramsay and Silverman, 2005; Staicu et al., 2010), adaptive smoothing methods within a frequentist framework (Polzehl and Spokoiny, 2006; Li et al., 2011), and spatial priors within the Bayesian framework (Gossel et al., 2001; Penny et al., 2005; Bowman et al., 2008; Smith and Fahrmeir, 2007). However, according to the best of our knowledge, none of them address the two issues including spatial smoothness and the Gaussian assumption simultaneously.
The aim of this paper is to develop a class of spatial transformation models (STMs) to simultaneously address the issues discussed above for the spatial analysis of neuroimaging data given a set of covariates. Our spatial transformation model is a hierarchical Bayesian model. First, we use a Box-Cox transformation model on the response variable assuming an unknown transformation parameter in order to satisfy the normality assumption in the imaging data, and then develop a regression model to characterize the association between the imaging data and the covariates. Second, we use a Gaussian Markov random field (GMRF) prior to capture the spatial correlation and spatial smoothness among the regression coefficients in the neighboring voxels. We develop an efficient Markov chain Monte Carlo (MCMC) algorithm to draw random samples from the desired posterior distribution. Our simulations and real data analysis demonstrate that STM significantly outperforms the standard voxel-wise model in recovering meaningful regions.

The rest of this paper is organized as follows. In Section 2.2, we introduce the STM and its associated prior distributions and Bayesian estimation procedure. In Section 2.3, we compare STM with the standard voxel-wise method using simulated data. In Section 2.4, we apply STM to a real imaging dataset on attention deficit hyperactivity disorder (ADHD). Finally, in Section 5, we present some concluding remarks.

2.2 Model

2.2.1 Model Description

Consider imaging measurements in a common space, which can be either a 3D volume or a 2D surface, and a set of clinical variables (e.g., age, gender, and height) from \( n \) subjects. Let \( \mathcal{D} \) and \( d \), respectively, represent the set of grid points in the common space and the center of a voxel in \( \mathcal{D} \), and \( N_D \) equals the number of voxels in \( \mathcal{D} \). Without loss of generality, \( \mathcal{D} \) is assumed to be a compact set in \( \mathbb{R}^3 \). For the \( i \)-th subject, we observe a univariate imaging measure \( y_i(d) \) at \( d \in \mathcal{D} \) and an \( N_D \times 1 \) vector of imaging measures, denoted by \( Y_{i,\mathcal{D}} = \{y_i(d) : d \in \mathcal{D}\} \). For simplicity, we consider a 3D volume throughout the paper.
We propose a class of spatial transformation models consisting of two major components: a transformation model and a Gaussian Markov random field model. The transformation model is developed to characterize the association between the imaging measures and the covariates at any \( d \in D \) and to achieve normality. Since most imaging measures are positive, we consider the well-known Box-Cox shifted power transformation (Box and Cox, 1964) throughout. Extensions to other parametric transformations are trivial (Sakia, 1992). Let \( y_i(d)^{(\lambda)} \) be the Box-Cox transformation of \( y_i(d) \) given by

\[
y_i(d)^{(\lambda)} = \begin{cases} 
(y_i(d) + c_0)^{\lambda} - 1 \bigg/ \lambda, & \text{if } \lambda \neq 0, \\
\log(y_i(d) + c_0), & \text{if } \lambda = 0,
\end{cases}
\]

where \( c_0 \) is prefixed and chosen such that \( \inf_{i,d}(y_i(d)) > -c_0 \). Our Box-Cox transformation model is given by

\[
y_i^{(\lambda)}(d) = x_i^T \beta(d) + \varepsilon_i(d) \quad \text{for } d \in D,
\]

where \( \beta(d) = (\beta_1(d), \ldots, \beta_p(d))^T \) is a \( p \times 1 \) vector of regression coefficients of interest, \( x_i \) is a \( p \times 1 \) vector of observed covariates for subject \( i \), and \( \varepsilon(d) = (\varepsilon_1(d), \ldots, \varepsilon_n(d))^T \) is an \( n \times 1 \) vector of measurement errors and follows a \( N_n(0, \sigma^2(d)I_n) \) distribution, in which \( I_n \) is an \( n \times n \) identity matrix.

The Gaussian Markov random field (GMRF) model is proposed to capture the spatial smoothness and correlation for each component of \( \{\beta(d) : d \in D\} \) across all voxels. For \( k = 1, \ldots, p \), the vector \( \beta_k = \{\beta_k(d) : d \in D\} \) is defined to be the coefficient set associated with the \( k \)-th covariate across all voxels. By imposing a GMRF for each component \( \beta_k \), we are implicitly modeling the spatial correlations among imaging measurements across voxels. In practice, it is very natural to assume that different \( \beta_k \) images may have different patterns, since different covariates play different roles in characterizing their association.
with the imaging data. Specifically, we assume that

\[ \beta_k \sim N(0, \nu_k^{-1}(I_{ND} + \phi_k H_k)^{-1}), \]

where \( \nu_k > 0 \) and \( \phi_k > 0 \) are, respectively, scale and spatial parameters. When \( \phi_k = 0 \), the elements of \( \beta_k \) are independent, whereas when the value of \( \phi_k \) is large, the model approaches an intrinsic autoregressive model (Ferreira and De Oliveira, 2007; Rue and Held, 2005). The known matrix \( H_k = \{h_k(d, d')\} \) is an \( ND \times ND \) matrix allowing the modeling of different patterns of spatial correlation and smoothness. Let \( N(d) \) be a set of neighboring voxels of voxel \( d \) in a given neighborhood system. Using the properties of GMRF (Rue and Held, 2005), the full conditional distribution of \( \beta_k(d) \) can be written as

\[ \beta_k(d) | \beta_{(k),[d]}, \nu_k, \phi_k \sim N\left( \frac{\phi_k \sum_{d' \in N(d)} h_k(d, d')\beta_k(d')}{1 + \phi_k h_k(d, d)} , \frac{1}{\nu_k[1 + \phi_k h_k(d, d)]} \right), \]

(2.2)

where \( \beta_{k,[d]} \) contains all \( \beta_k(d') \) for all \( d' \in D \) except \( d \). The conditional mean of \( \beta_{(k)}(d) \) is a weighted average of the \( \beta_k(d') \) values in the neighboring voxels of \( d \). As the number of neighboring voxels increases, the conditional variance decreases (Ferreira and De Oliveira, 2007).

A challenging issue is how to specify \( H_k = \{h_k(d, d')\} \) for each \( \beta_k \) in order to explicitly incorporate the spatial correlation and smoothness among neighboring voxels. We set

\[ h_k(d, d') = \begin{cases} \sum_{d' \in N(d)} \omega_k(d, d')^2, & \text{for } d = d', \\ -\omega_k(d, d')^2 1(d' \in N(d)), & \text{for } d \neq d', \end{cases} \]

where \( \omega_k(d, d') \) are some pre-calculated weights and \( 1(A) \) is the indicator function of a set \( A \). For every \( \phi_k \geq 0 \), \( (I_{ND} + \phi_k H_k)^{-1} \) is diagonally dominant and thus positive definite. For computational efficiency, we choose a relatively small neighborhood for each voxel \( d \) by defining \( N(d) = \{d' : ||d - d'||_2 \leq r_0\} \), where \( r_0 \) is a positive scalar and \( || \cdot ||_2 \) denotes the Euclidean distance. There are several ways of choosing the weights \( \omega_k(d, d') \) for any
Ideally, \( \omega(d, d') \) should contain some similarity information, such as spatial distance and imaging similarity, between voxels \( d \) and \( d' \). The simplest example of \( \omega_k(d, d') \) is \( \omega_k(d, d') = K(||d - d'||_2) \), where \( K(u) = \exp (-0.5u^2) \mathbf{1}(u \leq r_0) \). Other choices of \( \omega_k(d, d') \) are definitely possible. For instance, one may borrow information learned from other imaging modalities in order to construct the similarity between \( d \) and \( d' \).

### 2.2.2 Priors

We first consider the priors for the remaining parameters in the first level of model (2.1). Let \( \tau_d = (\sigma^2(d))^{-1} \) and \( U(-a, b) \) denote the uniform distribution on the interval \( (-a, b) \). We specifically assume that for \( d \in \mathcal{D} \),

\[
\tau_d \sim \text{Gamma}(\delta_0/2, \gamma_0/2) \quad \text{and} \quad \lambda_d \sim U(-a, b).
\]

For the second level parameter \( \nu = (\nu_1, \ldots, \nu_p) \), we assume for \( k = 1, \ldots, p \)

\[
\nu_k \sim \text{Gamma}(n_\nu/2, n_\nu s_\nu^2/2),
\]

where \( n_\nu \) and \( s_\nu^2 \) are hyperparameters. The choice of Gamma priors for the precision parameters is common in the literature since it maintains conjugacy (Chen et al., 2000). Other choices are \( \pi(\tau_d) = \tau_d^{-1} \mathbf{1}(\tau_d \geq 0) \) and \( \pi(\nu_k) = \nu_k^{-1} \mathbf{1}(\nu_k \geq 0) \), which are improper but in both cases lead to a proper posterior distribution. The uniform prior for the transformation \( \lambda_d \) was first introduced by Box and Cox (1964) and later adopted by several authors (Sweeting, 1984; Gottardo and Raftery, 2006).

### 2.2.3 Posterior Computation

An efficient Gibbs sampler is proposed to generate a sequence of random observations from the joint posterior distribution \( p(\beta, \lambda, \tau_\sigma, \nu|Y, x) \), where \( Y \) and \( x \), respectively, represent all observed responses and covariates. The Gibbs sampler essentially involves sampling from a series of conditional distributions while each of the modeling components is updated in turn. Although the order of the parameter update does not affect convergence, updating
the higher level parameters first can result in an improvement of the speed of convergence. Details pertaining to each step are presented below.

(i) Update each component of \( \nu = (\nu_1, \ldots, \nu_p) \) from its full conditional distribution,

\[
p(\nu_k|\cdot) \sim \text{Gamma}(0.5n^*_\nu, 0.5n^*_\nu s^2_k),
\]

where \( n^*_\nu = N_D + n_\nu \) and \( n^*_\nu s^2_k = n_\nu s^2_k + \sum_{j=1}^{N_D} \beta_k(j)^2 + \phi_k \beta_k^T H_k \beta_k \).

(ii) Update \( \beta_k(d), k = 1, \ldots, p \), for each voxel \( d \in D \) from its full conditional distribution,

\[
p(\beta_k(d)|\cdot) \sim \text{N} \left( \mu_{\beta_k}(d), \sigma^2_{\beta_k}(d) \right),
\]

where \( \sigma^2_{\beta_k}(d) = (\tau_d \sum_i x^2_{ik} + \theta_k(d))^{-1} \) and

\[
\mu_{\beta_k}(d) = \sigma^2_{\beta_k}(d) \left\{ \tau_d \sum_i [y^{(\lambda_d)}_i(d) - \sum_{l \neq k} x_{il} \beta_l^{(m)}(d)] x_{ik} + \theta_k(d)m_k(d) \right\}.
\]

Moreover, \( \beta^{(m)}(d) = \{ \beta_l^{(m)}(d); l = 1, \ldots, p \} \) is the estimated value of \( \beta(d) \) obtained in the previous iteration of the Gibbs sampler and \( \theta_k(d) \) and \( m_k(d) \) are, respectively, the inverse of the variance and the mean of the Gaussian distribution in (2.2).

(iii) Update \( \tau_\sigma(d) \) for each voxel \( d \in D \) from its full conditional distribution

\[
p(\tau_\sigma(d)|\cdot) \sim \text{Gamma} \left( \frac{1}{2} (n + \delta_0), \frac{1}{2} \sum_i (y^{(\lambda_d)}_i(d) - x_i^T \beta(d))^2 + \gamma_0 \right).
\]

(iv) Update \( \lambda_d \) for each voxel \( d \in D \) from its full conditional distribution

\[
p(\lambda_d|\cdot) \sim \prod_{d \in D} \exp \left\{ \frac{\tau_d}{2} \sum_i (y_i^{(\lambda_d)}(d) - x_i^T \beta(d))^2 \right\} \times \prod_{i=1}^{n} \frac{\beta_i^{\lambda_d-1}(d) \times (b + a)^{-1}}{\beta_i^{\lambda_d-1}(d) \times (b + a)^{-1}}.
\]
The full conditional distribution of $\lambda_d$ does not have a closed form, but sampling methods such as the Slice Sampler (Neal, 2003) or the Adaptive Rejection Metropolis Sampling (ARMS) (Gilks et al., 1995) can be used for such a purpose. The Metropolis-Hastings (MH) algorithm (Hastings, 1970) is also a very useful and easy algorithm for sampling $\lambda_d$. The MH algorithm proceeds as follows:

(a) Generate $\lambda_d^{prop}$ from $\mathcal{N}(\lambda_d^{(t-1)}, \delta_\lambda)$, where $\delta_\lambda > 0$ is a tuning parameter.

(b) Generate $V$ from $U(0, 1)$.

(c) Let $\alpha = \min \left\{ 1, \frac{p(\lambda_d^{prop}|\cdot)}{p(\lambda_d^{(t-1)}|\cdot)} \right\}$. If $V \leq \alpha$, then set $\lambda_d^{(t)} = \lambda_d^{prop}$. Otherwise, set $\lambda_d^{(t)} = \lambda_d^{(t-1)}$.

Full conditional derivations details are presented in A.

2.3 Simulation Study

We carried out a simulation study to examine the finite-sample performance of the STM in establishing an association between the imaging data and a set of covariates. The goals of this simulation study are

(G.1) To examine the ability of STM in capturing different geometric patterns;

(G.2) To examine the posterior estimates of spatially varying transformation parameters under two scenarios, including a no transformation model;

(G.3) To investigate the sensitivity of STM to the specification of $\phi_k$ and $(-a, b)$;

(G.4) To investigate the sensitivity of STM to the matrix $H_k$;

(G.5) To illustrate the fast convergence of the Gibbs sampler algorithm.

We randomly generated $n = 200$ lattices of size $32 \times 32$ according to model (2.1), in which we set $\sigma(d) = 0.3$ for all $d$ and $x_i = (x_{i0}, x_{i1}, x_{i2}, x_{i3})^T$ for $i = 1, \ldots, 200$. The design matrix $\mathbf{x}$ was generated to mimic real data and include an intercept, a continuous
variable, and two columns indicating categories of a discrete variable. They were generated as follows: (i) $x_{i1}$ is generated from $N(5, 1)$; (ii) $x_{i2}$ and $x_{i3}$, respectively, represent the second and third category of a discrete uniform random variable generated from three possible values, each of them representing a category and defined by $x_{iq} = 1(\text{Category } q) - 1(\text{Category 1})$ for $q = 2, 3$ (Pasta, 2005). We generated the values of the transformation parameters $\lambda_d$ from a discrete uniform random variable taking 0.5, 1, or 2. The generated $\Lambda$ structure is presented in the left panel of Figure 2.1. The parameters in $\beta$ are chosen to have a strong spatial correlation and their images are presented in the panels (a)-(d) of Figure 2.2.

For the hyperparameters of $\beta$, we chose a noninformative prior for each $\nu_k$ by setting $n_\nu = 10^{-3}$ and $s^2_\nu = 1$. As for the entries of the matrix $H_k$, we set it as in (2.3) and took the weights as $\omega_k(d, d') = K(||d - d'||_2)$, where $K(u) = \exp\left(-\frac{1}{2}u^2\right)1(u \leq r_0)$ and $r_0 = 2$. For each parameter $\tau_\sigma$, we chose noninformative priors by setting $\delta_0 = 10^{-3}$ and $\gamma_0 = 10^{-3}$. We fixed $\phi_k$ at 10, which indicates a strong spatial dependency among the components of each $\beta_{(k)}$, and then we set $a = b = 3$ for the hyperparameters of $\lambda_d$.

For each simulated dataset, we ran the Gibbs sampler for 1,000 iterations with 50 burn-in iterations. For the simulated examples, each iteration of the Markov chain takes approximately 2.5 seconds when running on a laptop with an i7 processor, 2.67GHz, and 8.0 GB of RAM. We summarize some simulation results based on some selected simulation scenarios below, while some additional results obtained from different simulation scenarios are considered in the Appendix A.

First, Figure 2.1 reveals that the estimated and true structures of $\Lambda = \{\lambda_d, d \in D\}$ show great similarity with each other. As expected, the estimated image $\hat{\Lambda} = \{\hat{\lambda}_d, d \in D\}$ is smoother than the true $\Lambda = \{\lambda_d, d \in D\}$ since a $U(-3, 3)$ prior is assumed for $\lambda_d$, allowing $\lambda_d$ to be sampled within this interval.

Second, we explore whether STM can recover the underlying spatial structure of each coefficient image. See Figure 2.2 for details. We compare the STM with two other models, including a voxel-wise linear model (panels (e)-(h)) and our STM (2.1) with $\lambda_d$ fixed at 1.
Figure 2.1: Simulation results: the true $\Lambda = \{\lambda_d, d \in D\}$ pattern in the left panel and the estimated pattern in the right panel. Estimated image is smoother compared with the true image due to the nature of the uniform distribution assumed \textit{a priori}. This figure appears in color in the electronic version of this article.

across all voxels (panels (i)-(l)). Figure 2.2 reveals that the voxel-wise linear model and STM (2.1) with $\lambda_d$ fixed at 1 cannot capture the pattern of true coefficient images. In contrast, STM (2.1) substantially improves the estimation of the coefficients, recovering their true geometric patterns, as observed in Figure 2.2, panels (m)-(p). Moreover, the STM is robust to the choices of the hyperparameters $\phi_k$ and $(-a, b)$. Furthermore, the correct specification of the matrix $H_k$ can yield good estimates if a reasonable neighborhood system is chosen. Finally, even if the true underlying model does not require spatial transformation parameters, STM can still provide good estimates of $\beta$.

Third, we illustrate the MCMC results for the parameters $\beta$, $\tau_\sigma$ and $\lambda$ at a randomly selected voxel. See Figure 2.3 for details. The trace plots indicate fast convergence of the Gibbs sampler, confirming its efficiency and good mixing properties. In addition, a more detailed diagnostics analysis is presented in the Appendix A. Based on the aforementioned results, we can conclude that the proposed single-site Gibbs sampler algorithm has good mixing properties and reaches convergence rapidly.
Figure 2.2: Simulation results on comparison of STM, GMRF with no transformation, and the voxel-wise linear model. Panels (a)-(d) represent the pattern of $\beta$ used to generate the images; panels (e)-(h) represent the estimated $\beta$ obtained from the least squares estimator in Matlab; panels (i)-(l) represent the posterior mean of $\beta$ obtained by fitting a GMRF model with no transformation; and panels (m)-(p) are the posterior mean of $\beta$ obtained from our STM. The inclusion of the transformation parameter substantially improves the estimation of the true underlying pattern. This figure appears in color in the electronic version of this article.
Figure 2.3: Trace plots for $\beta$, $\tau_0$ and $\lambda$ for a randomly generated voxel. The results are for a 1000 iterations of the MCMC algorithm and a burn-in sample of 50. The trace plots indicate a fast convergence of the algorithm, confirming its efficiency and good mixing properties. This figure appears in color in the electronic version of this article.

2.4 Application to the ADHD dataset

Our model is applied to the Attention Deficit Hyperactivity Disorder data, obtained from the ADHD-200 Consortium, (http://fcon_1000.projects.nitrc.org/indi/adhd200), a self-organized initiative where members from institutions around the world provide de-identified, HIPAA compliant imaging data. The goal of the project is to accelerate the scientific community’s understanding of the neural basis of ADHD, which is one of the most common childhood disorders affecting at least 5-10% of school age children and is associated with substantial lifelong impairment. The symptoms include difficulty staying focused and paying attention, difficulty controlling behavior, and hyperactivity (over-activity).

We analyze the imaging data from the New York University (NYU) Child Study Center. There are 219 subjects, 99 controls and 120 diagnosed with ADHD. Among them, 143 are males and 76 are females with an average age of 11.71 and 11.55 years, respectively. We used the high-resolution T1-weighted MRI images that were acquired using the MPRAGE
(Magnetization-prepared Rapid Acquisition with Gradient Echo) technique. The original T1-weighted images have size $256 \times 256 \times 198 \text{ mm}^3$ and voxel size of $1.0 \times 1.0 \times 1.0 \text{ mm}^3$.

For each subject, the images were first downsampled to the size of $128 \times 128 \times 99 \text{ mm}^3$. This process reduces the number of voxels while maintaining the image features and properties. Next, the images were processed using HAMMER (Hierarchical Attribute Matching Mechanism for Elastic Registration), a free pipeline developed by the Biomedical Research Imaging Center at UNC (available for downloading at http://www.hammersuite.com). The processing steps include skull and cerebellum removal, followed by tissue segmentation to identify the regions of white matter (WM), gray matter (GM) and cerebrospinal fluid (CSF). Then, registration was performed to warp the subject to the space of the Jacob template (Kabani et al., 1998; Davatzikos et al., 2001). Finally, a RAVENS map was calculated for each subject. The RAVENS methodology precisely quantifies the volume of tissue in each region of the brain. The process is based on a volume-preserving spatial transformation that ensures that no volumetric information is lost during the process of spatial normalization.

In Figure 2.4, we illustrate the white matter RAVENS images for two randomly selected subjects (panels (a) and (b)). These images were registered to the space of the template shown in panel (c). When we compare subjects in panels (a) and (b) of Figure 2.4, the image from the subject in panel (b) shows higher brightness inside the green square, reflecting the fact that relatively more white matter is presented in that particular region relative to the template.

We fitted model (2.1) with the white matter RAVENS images as responses and the covariate vector containing intercept, gender, age (previously standardized) and ADHD diagnostic status (1 for ADHD and -1 for control). Our interest is to identify morphological differences in the brain that are associated with the ADHD outcome, while adjusting for age and gender. As in the simulation study, for the hyperparameters of $\beta$, we chose a noninformative prior for each $\nu_k$ by setting $n_\nu = 10^{-3}$ and $s_\nu^2 = 1$. We fixed $\phi_k = 10$ and set $\omega_k(d, d') = K(||d - d'||_2)$, where $K(u) = \exp \left(-\frac{1}{2}u^2\right) 1(u \leq r_0)$ and $r_0 = 2$. For
For each parameter $\tau_d$, we chose a noninformative prior by setting $\delta_0 = 10^{-3}$ and $\gamma_0 = 10^{-3}$. For the transformation parameters $\lambda_d$, we set $a = b = 2$. We ran the Gibbs sampler for 1,000 iterations with 50 burn-in iterations. We calculated the posterior mean and a 95% credible interval for the coefficient associated with ADHD outcome at each voxel. A detailed discussion of credible intervals and its relation to the frequentist confidence interval is provided in Bayarri and Berger (2004). To detect important regions of interest, we created a 5% threshold map by mapping whether the 95% credible interval at each voxel contains 0 or not. Finally, we also fitted a no-transformation model, which is the STM with $\lambda_d$ fixed at 1 for all voxels.

An initial exploratory analysis was performed to examine whether the imaging measurements in the RAVENS map follow the Gaussian distribution. Normal probability plots of the intensities from sixteen random voxels are displayed in Figure 2.5, revealing that for some voxels, the imaging measurements strongly deviate from the Gaussian distribution. Further investigation of the posterior distribution of $\Lambda = \{\lambda_d, d \in \mathcal{D}\}$ reveals that the
transformation parameters are different from 1 for nearly 70% of the voxels, based on a 95% credible interval (Figure 2.6, panel (d)).

Figure 2.5: ADHD data analysis results: normal probability plots of sixteen random voxels revealing that the imaging measurements extracted from the RAVENS map deviate from the Gaussian distribution. This figure appears in color in the electronic version of this article.

We then mapped \( \hat{\Lambda} \) into the template to observe how the transformation parameter varies across the brain. If morphological differences exist in the regions where the transformation parameters are significantly different from 1, then analyzing the imaging data using the standard voxel-wise linear model may lead to spurious conclusions. On the other hand, if the transformation parameters are close to 1 in some regions, the estimates of the STM will be similar to those of the standard voxel-wise linear model in the regions. However, in practice, the location of such regions is unknown.

We compared the results from the STM with those from the no transformation model. Inspecting Figure 2.7, we are able to detect three large regions of interest, where mor-
Figure 2.6: ADHD data analysis results: selected slices showing the estimated \( \hat{\Lambda} \) for the imaging data obtained from the white matter RAVENS map. Panels (a)-(c) represent respectively, a coronal, sagittal and axial view of selected slices of the brain. The line indicates where the coronal and sagittal slices meet the plane in (c); panel (d) shows the same axial slice as in (c) and represents the location in the brain where \( \Lambda = \{ \lambda_d, \ d \in D \} \) are different from 1, based on a 95% credible interval. This figure appears in color in the electronic version of this article.
Phenological differences exist, including the right frontal lobe, the left frontal lobe and left parietal lobe. The frontal lobe has been implicated in planning complex cognitive behavior, personality expression, decision making and moderating social behavior (Yang and Raine, 2009) and morphological differences in this region were previously identified in children with ADHD (Sowell et al., 2003). Although the right frontal lobe is noticeable in all panels of Figure 2.7, the left frontal lobe cannot be seen for the no-transformation model in panel (d) of Figure 2.7. Thus, without the use of data transformations, we may miss some biologically meaningful regions of interest.

Figure 2.7: ADHD data analysis results. Top panels: significant regions in the brain where there exists a morphological difference between children with ADHD and children who do not have the disorder, based on a 95% credible interval. Panel (a) is a selected axial slice of the STM estimate overlaid on the Jacob template; (b) is the same selected slice showing the estimates of the spatial model with the transformation parameters Λ fixed and equal to 1 for all voxels also overlaid on the template; (c) and (d) are, respectively, the results of a 3D rendering of the STM and of the no transformation model both overlaid on the Jacob template. Bottom panel: (e) shows selected axial slices of the STM estimates overlaid on the template. Highlighted areas show the significant regions in the brain where there exists a morphological difference between children with ADHD and children who do not have the disorder. This figure appears in color in the electronic version of this article.
2.5 Discussion

We have proposed a method to model the association between different imaging modalities and clinical outcomes. The proposed model simultaneously overcomes two major limitations of voxel-wise methods that are widely used to model imaging data. First, the lack of normality of imaging measurements is addressed by proposing a spatially varying Box-Cox transformation model. Second, the voxel-wise methods treat all voxels as independent units, and thus they ignore important spatial smoothness observed in imaging data. We address this issue by assuming a Gaussian Markov random field (GMRF) prior to capture the spatial correlation and spatial smoothness among the regression coefficients in neighboring voxels. We developed an efficient Markov chain Monte Carlo (MCMC) algorithm to sample from the joint posterior distribution of the parameters. Our simulations and real data analysis demonstrate that STM significantly outperforms the standard voxel-wise model in recovering meaningful regions of interest.
CHAPTER 3: TENSOR PARTITION REGRESSION MODELS

3.1 Introduction

The aim of this paper is to develop a novel tensor partition regression modeling framework (TPRM) in order to use high-dimension imaging data, denoted by \( x \), to predict a scalar response, denoted by \( y \). The scalar response \( y \) may include cognitive outcome, disease status, and the early onset of disease, among others. In various neuroimaging studies, imaging data are often measured at a large number of grid points in a three (or higher) dimensional space and have a multi-dimensional tensor structure. Without loss of generality, we use \( x = (x_{j_1, \ldots, j_D}) \in \mathbb{R}^{J_1 \times \cdots \times J_D} \) to denote an order \( D \) tensor, where \( D \geq 2 \). Vectorizing \( x \) leads to a \( \prod_{k=1}^{D} J_k \times 1 \) vector. Examples of \( x \) include magnetic resonance imaging (MRI), diffusion tensor imaging (DTI), and positron emission tomography (PET), among many others. These advanced medical imaging technologies are essential to understanding the neural development of neuropsychiatric and neurodegenerative disorders.

Although a large family of regression methods have been developed for supervised learning (Hastie et al., 2009; Breiman et al., 1984; Friedman, 1991; Zhang and Singer, 2010), their computability and theoretical guarantee are compromised by this ultra-high dimensionality of imaging data. The first set of promising solutions is high-dimensional sparse regression (HSR) models, which often take high-dimensional imaging data as unstructured predictors. A key assumption of HSR is its sparse solutions. HSRs not only suffer from diverging spectra and noise accumulation in ultra-high dimensional feature spaces (Fan and Fan, 2008; Bickel and Levina, 2004), but also their sparse solutions are lack of interpretation in neuroimaging studies. Moreover, standard HSRs ignore the inherent spatial structure of the image that possesses wealth of information. To address some limitations of HSRs, a
family of tensor regression models has been developed to preserve the spatial structure of imaging tensor data, while achieving substantial dimensional reduction (Zhou et al., 2013).

The second set of solutions adopts functional linear regression (FLR) approaches, which treat imaging data as functional predictors. However, since most existing FLR models focus on one dimensional curves (Müller and Yao, 2008; Ramsay and Silverman, 2005), generalizations to two and higher dimensional images, however, is far from trivial and requires substantial research (Reiss and Ogden, 2010a). For instance, most estimation methods of FLR based on the fixed basis functions (e.g., tensor product wavelet) are required to solve an ultra-high dimensional optimization problem and suffer the same limitations as those of HSR.

The third set of solutions usually integrates supervised (or unsupervised) dimension reduction techniques with various standard regression models. Given the ultra-high dimension of imaging data, however, it is imperative to use some dimension reduction methods to extract and select 'low-dimensional' important features, while eliminating most redundant features (Johnstone and Lu, 2009; Bair et al., 2006; Fan and Fan, 2008; Tibshirani et al., 2002; Krishnan et al., 2011). Most of these methods first carry out an unsupervised dimension reduction step, often by principal component analysis (PCA), and then fit a regression model based on the top principal components (Caffo et al., 2010). Recently, for ultra-high tensor data, higher order tensors decompositions (e.g. parallel factor analysis and Tucker) have been extensively proposed to extract important information of neuroimaging data (Martinez et al., 2004; Beckmann and Smith, 2005). Although it is intuitive and easy to implement such methods, it is well known that the features extracted from PCA and Tucker can be irrelevant to the response. Similar comments also hold for FLR based on the functional PCA basis functions.

In this paper, we develop a novel TPRM to establish an association between imaging tensor predictors and clinical outcomes. Our TPRM is a hierarchical model with four components: (i) a partition model of dividing high-dimensional tensor covariates into sub-
tensor covariates; (ii) a canonical polyadic decomposition model of reducing sub-tensor covariates to low-dimensional feature vectors; and (iii) a generalized linear model of using the feature vectors to predict clinical outcomes. Moreover, a sparse inducing normal mixture prior is used to select informative feature vectors. Among the four components of TPRM, the key novelty of TPRM lies in the components (i) and (ii).

The first two components (i) and (ii) are designed to specifically address the three key features of neuroimaging data: low signal to noise ratio, the spatially clustered effect, and the tensor structure of imaging data. The neuroimaging data are often very noisy, while the ‘activated’ (or ‘effect’) brain regions associated with the response are usually clustered together and their size can be very small. In contrast, a crucial assumption for the success of most matrix/array decomposition methods (e.g., singular value decomposition) is that the leading components obtained from these decomposition methods capture the most important feature of a multi-dimensional array. Under TPRM, the ultra-high dimensionality of imaging data is dramatically reduced by using the partition model. For instance, let’s consider a standard $256 \times 256 \times 256$ 3D array with 16,777,216 voxels, and its partition model with $32^3 = 32,768$ sub-arrays with size $8 \times 8 \times 8$. If we reduce each $8 \times 8 \times 8$ into a small number of components by using component (ii), then the total number of reduced features is around $O(10^4)$. We can further increase the size of each subarray in order to reduce the size of neuroimaging data to a manageable level, resulting in efficient estimation.

The rest of the article is organized as it follows. In Section 3.2, we introduce TPRM, the priors, and a Bayesian estimation procedure. In Section 3.3, we use simulated data to compare the Bayesian decomposition with several competing methods. In Section 3.4, we apply our model to an attention deficit hyperactivity disorder (ADHD) data set. In Section 3.5, we present some concluding remarks.
3.2 Methodology

3.2.1 Preliminaries

We review several basic facts of tensor (Kolda and Bader, 2009b). A tensor $\mathbf{x} = (x_{j_1...j_D})$ is a multidimensional array, whose order $D$ is determined by its dimension. For instance, a vector is a tensor of order 1 and a matrix is a tensor of order 2. The inner product between two tensors $\mathbf{X} = (x_{j_1...j_D})$ and $\mathbf{Y} = (y_{j_1...j_D})$ in $\mathbb{R}^{J_1 \times \ldots \times J_D}$ is the sum of the product of their entries given by

$$\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{j_1=1}^{J_1} \ldots \sum_{j_D=1}^{J_D} x_{j_1...j_D} y_{j_1...j_D}.$$ 

The outer product between two vectors $\mathbf{a}^{(1)} = (a^{(1)}_{j_1}) \in \mathbb{R}^{J_1}$ and $\mathbf{a}^{(2)} = (a^{(2)}_{j_1}) \in \mathbb{R}^{J_2}$ is a matrix $M = (m_{j_1j_2})$ of size $J_1 \times J_2$ with entries $m_{j_1j_2} = a^{(1)}_{j_1} a^{(2)}_{j_2}$. A tensor $\mathbf{X} \in \mathbb{R}^{J_1 \times \ldots \times J_D}$ is a rank one tensor if it can be written as an outer product of $D$ vectors such that $\mathbf{X} = \mathbf{a}^{(1)} \odot \mathbf{a}^{(2)} \ldots \odot \mathbf{a}^{(D)}$, where $\mathbf{a}^{(k)} \in \mathbb{R}^{J_k}$ for $k = 1, \ldots, D$. Moreover, the parallel factor analysis, also known as PARAFAC or CP decomposition, factorizes a tensor into a sum of rank-one tensors such that

$$\mathbf{X} \approx \sum_{r=1}^{R} \lambda_r \mathbf{a}^{(1)}_r \odot \mathbf{a}^{(2)}_r \ldots \odot \mathbf{a}^{(D)}_r,$$

where $\mathbf{a}^{(k)}_r = (a^{(k)}_{j_1r}) \in \mathbb{R}^{J_k}$ for $k = 1, \ldots, D$ and $r = 1, \ldots, R$. See Figure 3.1 for an illustration of a 3D array.

We need the following notation throughout the paper. Suppose that we observe data $\{(y_i, \mathbf{X}_i, z_i) : i = 1, \ldots, N\}$ from $n$ subjects, where $\mathbf{X}_i$ are tensor imaging data, $z_i$ is a $p_z \times 1$ vector of scalar covariates, and $y_i$ is a scalar response, such as diagnostic status or clinical outcome. If we concatenate all $D$ dimensional tensor $\mathbf{X}_i$s into a $(D + 1)$ dimensional tensor $\tilde{\mathbf{X}} = \{\mathbf{X}_i, i = 1, \ldots, N\} = (x_{j_1,\ldots,j_D,i})$. We consider the CP decomposition of $\tilde{\mathbf{X}}$ as
Figure 3.1: Figure copied from (Kolda and Bader, 2009b). Panel (a) illustrates the CP decomposition of a three way array as a sum of R components of rank-one tensors, i.e. $\mathcal{X} \approx \sum_{r=1}^{R} a_r \circ b_r \circ c_r$.

follows:

$$\tilde{\mathcal{X}} = \|\Lambda; A^{(1)}, \ldots, A^{(D)}, G\|$$ or $$x_{j_1, \ldots, j_D, i} = \sum_{r=1}^{R} \lambda_r a^{(1)}_{j_1r} a^{(2)}_{j_2r} \ldots a^{(D)}_{j_Dr} g_{ir}.$$ (3.2)

where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_R)$, $A^{(d)} = [a^{(d)}_1 \ a^{(d)}_2 \ \ldots \ a^{(d)}_R]$ for $d = 1, \ldots, D$, and $G = (g_{ir})$ is called the factor matrix.

3.2.2 Tensor Partition Regression Models

Our interest is to develop TPRM for establishing the association between responses $y$ and their corresponding imaging covariates $\mathcal{X}$ and clinical covariates $Z$. The first component of TPRM is a partition model of dividing the high-dimensional tensor $\tilde{\mathcal{X}}$ into $S$ disjoint sub-tensor covariates $\tilde{\mathcal{X}}^{(s)}$, that is

$$\tilde{\mathcal{X}} = \bigcup_{s=1}^{S} \tilde{\mathcal{X}}^{(s)} \quad \text{and} \quad \tilde{\mathcal{X}}^{(s)} \cup \tilde{\mathcal{X}}^{(s')} = \emptyset.$$ (3.3)

Although the size of $\tilde{\mathcal{X}}^{(s)}$ can vary across $s$, it is assumed that without loss of generality, $\tilde{\mathcal{X}}^{(s)} \in \mathcal{R}^{p_1 \times \ldots \times p_D}$ and the size of $\tilde{\mathcal{X}}^{(s)}$ is homogeneous such that $S = \prod_{k=1}^{D} (J_k/p_k)$.

The second component of TPRM is a canonical polyadic decomposition model that reduces the sub-tensor covariates $\tilde{\mathcal{X}}^{(s)}$ to low-dimensional feature vectors. Specifically, it is
assumed that for each $s$, we have

$$\bar{X}^{(s)} = \| \Lambda_s; A_s^{(1)}, A_s^{(2)}, \ldots, A_s^{(D)}, G_s \| + \mathcal{E}^{(s)},$$

where $\Lambda_s = \text{diag}(\lambda_1^{(s)}, \ldots, \lambda_R^{(s)})$ consists of the weights for each rank of the decomposition in (3.4), $A_s^{(d)} \in \mathbb{R}^{p_d \times R}$ are the factor matrices along the $d$-th dimension of $X$, and $G_s \in \mathbb{R}^{N \times R}$ is the factor matrix along the subject dimension. It is assumed that the elements of $\mathcal{E}^{(s)} = (e_{j_1 \ldots j_D}^{(s)})$ are measurement errors and $e_{j_1 \ldots j_D}^{(s)} \sim N(0, (\tau^{(s)})^{-1})$. The elements of $G_s$ capture the major variation in $X^{(s)}$ due to subject differences, while the common structure among the subjects is absorbed into the factor matrices $A_s^{(d)}$ for $d = 1, \ldots, D$ (Kolda and Bader, 2009a).

There are two key advantages of using (3.3) and (3.4). First, the use of the partition model (3.3) allows us to concentrate on the most important local features of each sub-tensor, instead of the major variation of the whole image, which may be unassociated with the response of interest. In many applications, although the effect regions associated with responses may be relatively small compared with the whole image, their size can be comparable with that of each sub-tensor. Therefore, one can extract more informative features associated with the response with a high probability. Second, the use of the canonical polyadic decomposition model (3.4) can substantially reduce the dimension of original imaging data. Recall the discussions in Section 1 that the use of $8 \times 8 \times 8$ sub-tensors can substantially reduce imaging size at a scale of $O(10^3)$.

The third component of TPRM is a generalized linear model that links scalar responses $y_i$ and their corresponding reduced imaging features $G_s$ and clinical covariates $z_i$. Specifically, $y_i$ given $g_i$ and $z_i$ follows an exponential family distribution with density given by

$$f(y_i|\theta_i) = h(y_i) \exp\{\eta(\theta_i)T(y_i) - a(\theta_i)\},$$

(3.5)
where \( h(\cdot), \eta(\cdot), T(\cdot), \) and \( a(\cdot) \) are pre-specified functions. Moreover, it is assumed that \( \mu_i = E(y_i|g_i, z_i) \) satisfies

\[
h(\mu_i) = z_i^T \gamma + \sum_{s=1}^{S} g_i^{(s)} T b^{(s)},
\]

\[(3.6)\]

where \( g_i^{(s)} = \text{vec}(G_s) \) is the vectorization of \( G_s \) for all \( s \), and \( \gamma \) and \( b^{(s)} \) are coefficient vectors associated with \( z_i \) and \( g_i^{(s)} \), respectively.

### 3.2.3 Prior Distributions

We consider the priors on the elements of \( b_r^{(s)} \). The magnitude of \( SR \) can be much larger than \( N \) even for small \( R \), and thus model (3.6) is non-identifiable. To deal with this identifiability issue, bimodal sparsity promoting priors are key elements and have been the subject of extensive research (Mayrink and Lucas, 2013; George and McCulloch, 1993, 1997). We assume the following hierarchy:

\[
b_r^{(s)}|\delta_r^{(s)}, \sigma^2 \sim (1 - \delta_r^{(s)}) F(b_r^{(s)}) + \delta_r^{(s)} N(0, \sigma^2),
\]

\[
\delta_r^{(s)}|\pi \sim \text{Bernoulli}(\pi) \quad \text{and} \quad \pi \sim \text{Beta}(\alpha_{0\pi}, \alpha_{1\pi}),
\]

\[(3.7)\]

where \( F(\cdot) \) is a pre-specified probability distribution. A common choice of \( F(\cdot) \) is a degenerate distribution at 0, leading to what is called the ‘spike and slab’ prior (Mitchell and Beauchamp, 1988). A different approach is to consider \( F = N(0, \epsilon) \) with a very small \( \epsilon \) instead of putting a probability mass on \( b_r^{(s)} = 0 \). Thus, \( b_r^{(s)} \)’s are assumed to come from a mixture of two normal distributions. In this case, the hyperparameter \( \sigma^2 \) should be large enough to give support to values of the coefficients that are substantively different from 0, but not so large that unrealistic values of \( b_r^{(s)} \) are supported. In this article, we opt for the latter approach.

The probability \( \pi \) determines whether a particular \( g_i^{(s)} \) is informative for predicting \( y \). A common choice for its prior is a non-informative distribution with \( \alpha_{0\pi} = \alpha_{1\pi} = 1 \). However,
this choice for the hyperparameters implies that its posterior mean is restricted to the interval 
$[1/3, 2/3]$, a undesirable feature in variable selection. To fix this, we choose a “bathtub”
shaped beta distribution, since a prior concentrating most of its mass in the extremes of the
interval $(0, 1)$ is evidently more suitable for variable selection (Gonalves et al., 2013).

We consider the priors on the elements of $A^{(d)}_{(s)r}$, $g^{(s)}_r$, $\tau^{(s)}$, and $\lambda^{(s)}_r$. For $d = 1, \ldots, D$
and $r = 1, \ldots, R$, we assume

$$
A^{(d)}_{(s)r} \sim N(0, p_d^{-1} I_{p_d}), \quad g^{(s)}_r \sim N(0, I_N), \quad \tau^{(s)} \sim \text{Gamma}(\nu_0\tau, \nu_1\tau), \quad \lambda^{(s)}_r \sim N(0, \kappa^{-1}),
$$

where $I_k$ be a $k \times k$ identity matrix. When $p_d$ is large, the columns of the factor matrix
$A^{(d)}_{(s)r}$ are approximately orthogonal, which is consistent with their role in the decomposition
(3.1). However, we only impose that the columns of the factor matrices span the space of
the principal vectors, without explicitly requiring orthonormality (Xinghao Ding and Carin,
2011).

For the remaining elements of TPRM, we assume

$$
\gamma \sim N(0, \nu^{-1} I_q) \quad \text{and} \quad \nu \sim \text{Gamma}(\nu_{0\nu}, \nu_{1\nu}).
$$

### 3.2.4 Posterior Inference

Let $A^{(d)} = [A^{(d)}_1, \ldots, A^{(d)}_S], G = [G_1, \ldots, G_S], B = [b^{(1)}, \ldots, b^{(S)}], \Lambda = [\Lambda_1, \ldots, \Lambda_\nu],$
and $\tau = [\tau^{(1)}, \ldots, \tau^{(S)}].$ Consider $\theta = \{A^{(1)}, \ldots, A^{(D)}, G, \Lambda, \tau, \gamma, \nu, B, \delta, \pi\}$. A Gibbs
sampler algorithm is used to generate a sequence of random observations from the joint
posterior distribution given by

$$
p(\theta, \lambda, \gamma, \nu, \nu_0, \nu_1, \nu_{0\nu}, \nu_{1\nu}) \propto p(y|\theta, \lambda, \nu) p(A^{(1)}_1, \ldots, A^{(D)}_S, G, \Lambda, \tau|\lambda, \gamma, \nu, B, \delta, \pi) p(\delta|\pi) p(\lambda) p(\gamma|\nu) p(\nu).
$$

The Gibbs sampler essentially involves sampling from a series of conditional distributions,
while each of the modeling components is updated in turn.
As an illustration, we divide the whole image into \( S \) equal sized regions and assume 
\[ y_i \sim \text{Bernoulli}(\mu_i) \]
with the link function \( h(\cdot) \) being the probit function. By following Albert and Chib (1993), we define a normally distributed latent variable, \( w_i \), such that
\[ w_i \sim N(\mu_i, 1); \quad y_i = 1(w_i > 0), \]
where \( 1(\cdot) \) is an indicator function of an event.

The complete Gibbs sampler algorithm proceeds as follows.

(a.0) Generate \( w = (w_1, \cdots, w_n)^T \) from
\[
w_i | y_i = 0 \sim 1(w_i \leq 0)N(z_i^T \gamma + \sum_{s=1}^S g_i^{(s)T} b^{(s)}, 1),
\]
\[
w_i | y_i = 1 \sim 1(w_i \geq 0)N(z_i^T \gamma + \sum_{s=1}^S g_i^{(s)T} b^{(s)}, 1).
\]

(a.1) Update \( \tau(s) \) from its full conditional distribution
\[
\tau(s) | \cdots \sim \text{Gamma}(\nu_0 + (N \prod_{d=1}^D p_d)/2, \nu_1 + (1/2) \sum_{i,j_1,\ldots,j_D} (x^s_{ij_1,\ldots,j_D}(s))^2),
\]
where \( x^s_{ij_1,\ldots,j_3}(s) = \{X^{(s)} - \|A^{(s)}; A_s^{(1)}, A_s^{(2)}, \ldots, A_s^{(D)}, L^{(s)}\|\}_{ij_1,\ldots,i_3} \)

(a.2) Update \( \{A_s^{(d)}\}_{jd} \) from its full conditional distribution given by
\[
\{A_s^{(d)}\}_{jd} | \cdots \sim N\left(\frac{\tau(s)\langle \hat{X}_s^{(jd)}; \hat{I}_s^{(-d)} \rangle}{\tau(s)\langle \hat{I}_s^{(-d)}; \hat{I}_s^{(-d)} \rangle + p_d}, \left(\frac{\tau(s)\langle \hat{I}_s^{(-d)}; \hat{I}_s^{(-d)} \rangle + p_d}{\tau(s)\langle \hat{I}_s^{(-d)}; \hat{I}_s^{(-d)} \rangle} \right)^{-1}\right),
\]
where \( \hat{I}_s^{(-d)} = \|A^{(s)}; A_s^{(1)}, \ldots, A_s^{(d-1)}, A_s^{(d+1)}, \ldots, A_s^{(D)}; L^{(s)}\|, \hat{X}_s^{(-r)} \) is given by
\[
X^{(s)} - \|A^{(s)}; A_s^{(1)}, A_s^{(2)}, \ldots, A_s^{(D)}, L^{(s)}\| + \|A^{(s)}; A_s^{(1)}; r, \{A_s^{(2)}; r, \ldots, A_s^{(D)}; r, \{L_s^{(1)}; r\}\},
\]
and \( \hat{X}_s^{(jd)} \) is a subtensor fixed at the entry \( j_d \) along the \( d \)-th dimension of \( \hat{X}_s^{(-r)} \).
(a.3) Update \( \{ L_s \}_{ir} \) from its full conditional distribution given by

\[
\{ L_s \}_{ir} \cdots \sim N \left( \frac{\tau(s) \langle \hat{X}^{s(i)}_{(-r)}, L^s \rangle}{\tau(s) \langle L^s, L^s \rangle + N}, \frac{\tau(s) \langle T^s, L^s \rangle + N}{\tau(s) \langle L^s, L^s \rangle + N} \right),
\]

where \( T^s = \| A_s^{(1)} \equiv \ldots \equiv A_s^{(D)} \| \) and \( \hat{X}^{s(i)}_{(-r)} \) is a subtensor fixed at the \( i \)-th entry along the subject dimension of \( \hat{X}^s_{(-r)} \).

(a.4) Update \( \Lambda^{(s)} \) from its full conditional distribution

\[
\lambda_r^{(s)} \cdots \sim N \left( \frac{\tau(s) \langle \hat{X}^{s}_{(-r)}, L^s \rangle}{\tau(s) \langle L^s, L^s \rangle + \kappa}, \frac{\tau(s) \langle L^s, L^s \rangle + \kappa}{\tau(s) \langle L^s, L^s \rangle + \kappa} \right),
\]

where \( L^s = \| 1_R; A_s^{(1)} \equiv \ldots \equiv A_s^{(D)} \| \) and \( 1_R \) is a vector of ones of size \( R \).

(a.5) Update \( \delta_r^{(s)} \) from its full conditional distribution

\[
\delta_r^{(s)} \sim \text{bernoulli}(\tilde{p}_1/\tilde{p}_1 + \tilde{p}_0),
\]

where \( \tilde{p}_1 = \pi \exp\{-(1/2\sigma^2)(b_r^{(s)})^2\} \) and \( \tilde{p}_0 = \pi \exp\{-(1/2\epsilon)(b_r^{(s)})^2\} \).

(a.6) Update \( b_r^{(s)} \) from its full conditional distribution

\[
b_r^{(s)}|\delta_r^{(s)} = 1 \sim N \left( \sum_i \tilde{w}_i^{(s)} g_{ir}^{(s)} / \sum_i (g_{ir}^{(s)})^2 + 1/\sigma^2, (\sum_i (g_{ir}^{(s)})^2 + 1/\sigma^2)^{-1} \right),
\]

\[
b_r^{(s)}|\delta_r^{(s)} = 0 \sim N \left( \sum_i \tilde{w}_i^{(s)} g_{ir}^{(s)} / \sum_i (g_{ir}^{(s)})^2 + 1/\epsilon, (\sum_i (g_{ir}^{(s)})^2 + 1/\epsilon)^{-1} \right),
\]

where \( \tilde{w}_i^{(s)} = w_i - z^T_i \gamma - \sum_{s'=1}^S g_i^{(s')} T b_r^{(s')} + g_{ir}^{(s)} T b_r^{(s)} \).

(a.7) Update \( \pi \) from its full conditional distribution

\[
\pi | \cdots \sim \text{beta}(\alpha_0 + \sum_{s,r} \delta_r^{(s)}, \alpha_1 + |s|R - \sum_{s,r} \delta_r^{(s)}).
\]
(a.8) Update $\gamma$ from its full conditional distribution

$$
\gamma | \cdots \sim \mathcal{N} \left( \Sigma^*^{-1} Z^T w^*, \Sigma^*^{-1} \right),
$$

where $\Sigma^* = v I_q + Z^T Z$ and $w^* = w - \sum_{s=1}^S g_i^{(s)T} b^{(s)}$.

(a.9) Update $v$ from its full conditional distribution

$$
v | \cdots \sim \text{Gamma} \left( \nu_0 v + q/2, \nu_1 v + (\gamma^T \gamma)/2 \right).
$$

All the tensor operations described in steps (a.1) – (a.4) can be easily computed using Bader et al. (2012), available for download at http://www.sandia.gov/tgkolda/TensorToolbox/index-2.5.html.

### 3.3 Simulation Study

We carried out three sets of simulations to examine the finite-sample performance of TPRM and its associated Gibbs sampler algorithm.

#### 3.3.1 Bayesian tensor decomposition

The goals of the first set of simulations are (i) to compare the proposed Bayesian tensor decomposition method with the alternating least squares method, (ii) to investigate how different choices of the rank $R$ impact the tensor decomposition for distinct image modalities; and (iii) to access the importance of the partition model. We considered 3 different imaging data sets (or tensors) including (I-1) a diffusion tensor image (DTI) of size $90 \times 96 \times 96$, (I-2) a white matter RAVENS map image of size $99 \times 99 \times 70$, and (I-3) a T1-weighted MRI image of size $64 \times 108 \times 99$. We fitted models (3.3) and (3.4) to the three image tensors and decomposed each of them with $R = 5, 10, \text{and } 20$. For the DTI image, we consider 27 partitions of size $30 \times 30 \times 32$, for the RAVENS map we consider 18 partitions of size $33 \times 33 \times 35$, and for the T1, 24 partitions of size $32 \times 27 \times 33$. The
hyperparameters were chosen to reflect non-informative priors, \( \nu_0 = 1, \nu_1 = 10^{-2} \), and \( \kappa = 10^{-6} \).

We run steps \((a.1) - (a.4)\) of the Gibbs sampler algorithm in Section 3.2.4 for 5,000 iterations. The efficiency of the proposed algorithm is observed through trace plots for 9 random voxels. Figure 3.2 shows the trace plots for the reconstructed white matter RAVENS map decomposed with \( R = 20 \). The proposed algorithm is efficient and presents a fast convergence.

At each iteration, we computed the quantity \( I = \sum_{s=1}^{S} \| A_s; A_s^{(1)}, A_s^{(2)}, A_s^{(3)} \| \) for each rank and each partition. Subsequently, we computed the reconstructed image, defined as \( \hat{X} \), and the posterior mean estimate of \( I \) after a burn-in sample of 3,000. For each reconstructed image \( \hat{X} \), we computed its root mean squared error, \( \text{RMSE} = ||\hat{X} - X||_2 / \sqrt{J_1J_2J_3} \). We compare the RMSE for the Bayesian non-partition model, the partition, and the standard alternating least squares method (ALS) (Kolda and Bader, 2009a). Results are shown in Table 3.1. For the images considered in this study, the partition model gives the smallest RMSE, and the Bayesian decomposition gives a smaller RMSE when compared to the standard ALS. As expected, higher is the rank, smaller is the reconstruction error.

We illustrate the importance of the partitions in Figure 3.3. Results are from an axial slice of the original images and the reconstructed images for ranks \( R = 5, 10, \) and 20 for both non-partition (top panels) and partition models (bottom panels) for all three images considered in this section. We clearly see an improvement in reconstruction when the partitions are considered in the model.

### 3.3.2 A 2-dimensional image example

The goals of the second set of simulations are to assess whether TPRM is able to capture regions of interest, that significantly differ between two groups, in a 2-dimensional phantom and to compare TRPM with the functional principal components model (fPCA). We generate
Figure 3.2: Trace plots in 9 randomly chosen voxels in the white matter RAVENS map by using Bayesian tensor decomposition with $R = 20$. The trace plots indicate that the Markov chain converges after around 1000 iterations.

Table 3.1: Root mean squared error for 3 different image modalities. The Bayesian decomposition outperforms the alternating least squares in each scenario. There is a smaller error measurement with an increase of the rank $R$.

<table>
<thead>
<tr>
<th></th>
<th>T1-weighted</th>
<th>WM RAVENS</th>
<th>DTI</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R=5$</td>
<td>BayesianCP</td>
<td>45.3191</td>
<td>1.5853</td>
</tr>
<tr>
<td></td>
<td>ALS</td>
<td>45.3636</td>
<td>1.6013</td>
</tr>
<tr>
<td></td>
<td>Partition</td>
<td>37.3712</td>
<td>1.2178</td>
</tr>
<tr>
<td>$R=10$</td>
<td>BayesianCP</td>
<td>41.7018</td>
<td>1.4382</td>
</tr>
<tr>
<td></td>
<td>ALS</td>
<td>42.4350</td>
<td>1.4533</td>
</tr>
<tr>
<td></td>
<td>Partition</td>
<td>31.3836</td>
<td>1.0186</td>
</tr>
<tr>
<td>$R=20$</td>
<td>BayesianCP</td>
<td>37.1796</td>
<td>1.2885</td>
</tr>
<tr>
<td></td>
<td>ALS</td>
<td>38.3166</td>
<td>1.3166</td>
</tr>
<tr>
<td></td>
<td>Partition</td>
<td>25.1574</td>
<td>0.8085</td>
</tr>
</tbody>
</table>

a data set $\{(y_i, \mathcal{X}_i) : i = 1, \ldots, n\}$ with $n = 200$ according to

$$y_i \sim \text{Bernoulli}(0.5) \quad \text{and} \quad \mathcal{X}_i = X_0(y_i) + \mathcal{E}_i,$$
Figure 3.3: Simulation 1 for Bayesian tensor decomposition results: panel (a): DTI image; panel (b): white matter RAVENS map; and panel (c): T1-weighted image In each row, the first image represents an axial slice of the original image and from left to right we have the decomposed images for ranks $R = 5, 10, 20$, respectively.
where $X_i$, $E_i = (\epsilon_{ij_1j_2})$, and $X_0(y_i)$ are $32 \times 32$ matrices and $X_0(0)$ and $X_0(1) - X_0(0)$ are, respectively, shown in panels (a) and (b) of Figure 3.4. We independently generated $\epsilon_{ij_1j_2}$ from a $N(0, 40^2)$ generator for $j_1, j_2 = 1, \ldots, 32$. Panel (c) in Figure 3.4 shows a generated 2D image from a random subject in group 1, which is almost indistinguishable from random noises. The hyperparameters in Section 3.2.3 are chosen to reflect non-informative priors with $\nu_{0\tau} = 1$, $\nu_{1\tau} = 10^{-4}$, $\sigma^2 = 10^4$, and $\kappa = 10^{-4}$.

We applied two TPRMs with $S = 1$ (no-partition model) and $S = 4$ to the simulated data set. We compared the two TPRMs with a functional principal components model (fPCA), in which we learned the basis functions in the first stage and then included the top $R$ most important principal components as covariates in a logistic regression in the second stage. We set $R = 5$ for all three models, while for TPRMs, we run the Gibbs sampler algorithm for 5000 iterations with a burn-in period of 3000 iterations. We also computed the deviance information criteria (DIC) to compare different models (Spiegelhalter et al., 2002).

We also computed the Bayesian estimate of $P = \|\Lambda; A^{(1)}, A^{(2)}, B\|$ by using MCMC samples. The estimated quantity $P$ represents a projection of the group differences into the image space. Furthermore, we used MCMC samples to construct credible intervals of $P$ in the imaging space. This quantity is extremely important in neuroimaging studies since it allows us to precisely identify significant locations in the brain that are associated with the response variable. Figure 3.4 shows the results. Panels (d), (e), and (g) are the posterior mean estimates of $P$ for the fPCA model, TPRM with $S = 4$, and TPRM with $S = 1$, respectively. Panels (f) and (h) are the $95\%$ credible interval for TPRMs with $S = 4$ and $S = 1$, respectively. The result reveals that the proposed model closely recovers the true underlying location where differences between both groups exist. The DIC for both TPRM models are 276.9539 and 277.1392 for TPRM with $S = 4$ and TPRM with $S = 1$, respectively, indicating a slightly better fit for the partition model.
Figure 3.4: Results of the 2-D imaging example: (a): $X_0(0)$; (b): $X_0(1) - X_0(0)$; (c): the simulated image from a randomly selected subject from group 1; (d): the estimated projection $\mathcal{P}$. Panels (e) and (g) are the posterior mean of the quantity $\mathcal{P} = \Lambda; \mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \mathbf{B}$ for BTRM with $S = 4$ and the no-partition model with $S = 1$, respectively. Panels (f) and (h) are, respectively, the 95% credible interval of $\mathcal{P}$ for BTRM($S = 4$) and BTMR($S = 1$) revealing the true underlying location, where differences between both groups exist.
3.3.3 A 3D image example

The goal of this set of simulations is to examine the finite-sample performance of TPRM in the 3D imaging setting. We used the same simulation setting as the 2D image example except that we simulated the three-dimensional image covariates $X_i(y_i)$ as follows:

$$X_i(y_i) = G_0 + 150y_iX_0 + E_i,$$

where $G_0 \in \mathbb{R}^{64 \times 64 \times 50}$ is a fixed brain template with values ranging from 0 to 250 and the elements of the tensor $E_i \in \mathbb{R}^{64 \times 64 \times 50}$ were independently generated from a $N(0, 50^2)$ generator. Moreover, we set $X_0 = \|1; A^{(1)}, A^{(2)}, A^{(3)}\|$, where $A^{(1)}_0 \in \mathbb{R}^{64 \times 2}$, $A^{(2)}_0 \in \mathbb{R}^{64 \times 2}$, and $A^{(3)}_0 \in \mathbb{R}^{50 \times 2}$ are matrices whose $(23 + j)$-th element of each column is equal to $\sin(j\pi/14)$. Figure 5 (a) presents the exact location of $X_0$ overlaid on $G_0$.

We applied two TPRMs with $S = 8$ and $S = 1$ and fPCA to the simulated data set. We set $R = 30$ for all three models. For TPRMs, we set the same hyperparameters as those in the previous section. We ran the Gibbs sampler algorithm for 5000 iterations with a burn-in period of 3000 iterations. We computed the posterior mean of the projection $P = \|\Lambda; A^{(1)}, A^{(2)}, A^{(3)}, B\|$ and the DIC criteria based on the MCMC samples for all models. Figure 3.5 (b)-(d) present axial slices of the estimated projections obtained from all three models. The figure reveals that both TPRMs are able to recover $X_0$, whereas fPCA does not perform well, presenting extremely noisy results for the estimated projection.

The DIC criteria can be used to help one decide between the partition model and the non-partition model. We repeat the same simulation study 100 times and compute the DIC for both models. Our goal is to observe if the DIC is consistently picking the same model for the proposed scenario. Figure 3.6 shows the result. The DIC for TPRM($S = 8$) ranges from 137.7216 to 138.0222, while DIC for TPRM($S = 1$) ranges from −912.0844 to 3320.10. Negative values indicate substantial conflict between the prior and data, or where the posterior mean is a poor estimator, such as a symmetric bimodal distribution.
Figure 3.5: Results of the 3-D image example. In each panel, we show axial views of the 2 true signal regions and a 3D render of the results overlaid on the template $G_0$ from left to right. Panel (a) show the true effect signal $X_0$. Panels (b) and (c) present the posterior mean of the quantity $P = \| \Lambda; A^{(1)}, A^{(2)}, A^{(3)}, B \|$ for TPRMs with $S = 8$ and with $S = 1$, respectively. Panel (d) shows the results for fPCA. Both TPRMs are able to recover $X_0$, whereas fPCA does not.
(Spiegelhalter et al., 2002) and it happened 5 out of 100 of the generated dataset. Inspecting Figure 3.6 reveals that TPRM($S = 8$) is preferred than TPRM($S = 1$) in the majority of generated datasets.

![Figure 3.6: DIC results of 100 simulated examples. Straight line indicates the non-partition model TPRM($S = 1$) and dotted line indicates partition TPRM($S = 8$). The later model is preferred in the majority of generated datasets.](image)

### 3.4 Real data analysis

We applied the proposed model to the Attention Deficit Hyperactivity Disorder data, from the New York University site, as part of the ADHD-200 Consortium (http://fcon 1000.projects.nitrc.org/indi/adhd200). The Consortium is a self-organized initiative where members from institutions around the world provide de-identified, HIPAA compliant imaging data. The goal of the project is to accelerate the scientific community's understanding of the neural basis of ADHD. ADHD is a common disorder affecting 5-10% of school age children and is associated with substantial lifelong impairment. Symptoms include difficulty staying focused and paying attention, difficulty controlling behavior, and hyperactivity (over-activity).
The data from the New York University (NYU) Child Study Center consists of 219 subjects, 99 controls and 120 diagnosed with ADHD. Among them, 143 are males and 76 are females with an average age of 11.71 and 11.55 years, respectively. We used the high-resolution T1-weighted MRI images that were acquired using the magnetization-prepared rapid acquisition with gradient echo (MPRAGE) technique. The original T1-weighted images have size $256 \times 256 \times 198 \text{mm}^3$ and voxel size of $1.0 \times 1.0 \times 1.0 \text{mm}^3$. For each subject, the images were first downsampled to the size of $64 \times 64 \times 50 \text{mm}^3$. This process reduces the number of voxels while maintaining the image features and properties. Next, the images were processed using HAMMER (Hierarchical Attribute Matching Mechanism for Elastic Registration), a free pipeline developed by the Biomedical Research Imaging Center at UNC (available for downloading at http://www.hammersuite.com). The processing steps include skull and cerebellum removal, followed by tissue segmentation to identify the regions of white matter (WM), gray matter (GM) and cerebrospinal fluid (CSF). Then, registration was performed to warp the subject to the space of the Jacob template (Kabani et al., 1998; Davatzikos et al., 2001). Finally, a RAVENS map was calculated for each subject. The RAVENS methodology precisely quantifies the volume of tissue in each region of the brain. The process is based on a volume-preserving spatial transformation that ensures that no volumetric information is lost during the process of spatial normalization.

We fitted TPRM with ADHD diagnostic status (1 for ADHD and 0 for control) as the response variable, the white matter RAVENS map as the image covariate, and age and gender as clinical covariates. As a comparison, we also consider fPCA. We considered $S = 24$ and $S = 1$ for $R = 30$ and $R = 50$. We also set the hyperparameters $\alpha_{0r} = 1$ and $\alpha_{1r} = 0.001$, to reflect a non-informative prior. For each $s$, the sparse inducing prior for the elements of $B^{(s)}$ follow the mixture in (3.7) with $F = N(0, \epsilon)$ and $\epsilon = 0.0001$. For the probability $\pi$, we opt for the hyperparameters $\alpha_{0\pi} = 0.5$ and $\alpha_{1\pi} = 0.5$ to reflect a bathtub shaped beta prior. We ran the Gibbs sampler for 5,000 iterations with a burn-in period of 3000 iterations.
Figure 3.7 shows posterior mean estimates for the projection \( \hat{P} = \left\| \lambda; A^{(1)}, A^{(2)}, A^{(3)}, B \right\| \) for all models as \( R = 30 \). All models can detect a significant right frontal lobe region. A strong signal in the left frontal lobe and left parietal lobe can be detected by TPRMs, not fPCA. Figure 3.8 shows results for credible intervals for the estimated projection overlaid on the Jacob template. Figure 3.8 reveals two large regions of interest including the right frontal lobe and left parietal lobe, where morphological differences exist. The frontal lobe has been implicated in planning complex cognitive behavior, personality expression, decision making and moderating social behavior (Yang and Raine, 2009). Morphological differences in this region were previously identified in children with ADHD (Sowell et al., 2003) and similar conclusions were previously obtained for this dataset (Miranda et al., 2013).

Figure 3.7: Axial slices of the posterior mean estimates for the projection \( P = \left\| \lambda; A^{(1)}, A^{(2)}, A^{(3)}, B \right\| \) for the TPRM (panel (b)) and the estimated projection for fPCA (panel (a)). The colors indicate how strong the differences are between children with ADHD and control. The 3 slices on the right indicate a possible right frontal lobe region, in both models. A strong difference seems to exist in the left frontal lobe and left parietal lobe when observing the TPRM model but not in the fPCA model. The chain was generated with 5000 iterations and the burn-in period is of 3000.
Figure 3.8: ADHD data analysis results. Panels (a) and (b) are, respectively, the results of a 3D rendering of the 90% and 95% credible intervals for the projection $P = \\lambda; A^{(1)}, A^{(2)}, A^{(3)}, B$ both overlaid on the Jacob template. Panel (c) and (d) are selected axial slices of the credible intervals (95% and 90%, respectively). We detect two large regions of interest, where morphological differences exist, including the right frontal lobe and left parietal lobe.
3.5 Discussion

We have proposed a novel method to establish association between an array and clinical outcomes. Differently from the widely used voxel-wise methods for neuroimaging data, we consider the opposite paradigm and formulate our model in a regression setup by considering a clinical outcome as a response and a matrix/array as a covariate. The proposed method has three key components chosen carefully to overcome the major limitations of using whole neuroimages as covariates. First, the images are partitioned into smaller subsets (patches). We are able then to capture the subject variation by performing a tensor decomposition in each subset. Next, we take an hierarchical approach where information along the subject dimension is entered as a covariate in a generalized linear model approach.

Our simulation studies shows that the Bayesian tensor decomposition successfully recovers true images of different modalities and the reconstruction errors get smaller with an increase of rank numbers in the decomposition. The supervised model is able to efficiently estimate the locations where differences between two groups of subjects occur. In cases with a higher intra-image variability, the partition model is shown to perform better in detecting those differences.

Finally, the proposed model is able to identify meaningful regions in the brain where significant changes occur when comparing subjects with different ADHD outcomes and it provides a great contribution to the field of neuroimaging analysis and a promising direction to deal with array covariates, a important topic in a world of increasing complex datasets.
CHAPTER 4: SPARSE PARTITION FACTOR MODELS

4.1 Introduction

In the past few decades, advanced medical imaging technologies have been generating extremely complex data sets. In neuroscience, imaging-related studies are essential to understanding the neural development of neuropsychiatric and neurodegenerative disorders, the normal brain, and the interactive effects of environmental and genetic factors on brain structure and function. Examples are the ADNI (Alzheimer’s Disease Neuroimaging Initiative) and the ADHD-200 Consortium (Attention of Deficit Hyperactivity Disorder). In these studies, images are usually collected using different techniques, generating distinct image modalities, e.g. magnetic resonance image (MRI), functional MRI, diffusion tensor image (DTI), positron emission tomography (PET), electroencephalography (EEG), among many others.

With these advances, there has been an increasing interest in developing methods to characterize varying associations between high-dimensional imaging data and low-dimensional covariates. The most common solution is to fit a general linear model to the imaging data from all subjects at each voxel as responses and clinical variables as predictors. Such models are known as voxel-wise models and have been the topic of extensive research in the past years (Friston, 2007; Lindquist, 2008; Lazar, 2008; Li et al., 2011; Miranda et al., 2013).

Another set of solutions treat imaging data as predictors, and there are three common strategies in this approach. First, data is represented as a one-dimensional combination of basis function (Ramsay and Silverman, 2005; Reiss and Ogden, 2010b; Müller and Yao, 2008), but extending these methods for three or higher dimensions is far from trivial. Second, high-dimensional sparse regression models are adopted (Fan and Fan, 2008; Bickel and Levina, 2004), but the sparse solution usually lacks interpretation in neuroimaging
studies. Third, supervised or unsupervised dimension reduction techniques are employed (Fan and Fan, 2008; Tibshirani et al., 2002; Krishnan et al., 2011), often extracting the most relevant principal components on a first stage and using them as inputs in a regression model on a second stage (Caffo et al., 2010). A crucial assumption is that the leading components obtained from these decompositions capture the most important features of the multi-dimensional array. However, neuroimaging data are extremely noisy, and regions affecting the outcome are small and often clustered together. As a consequence, it is likely that ’effect’ regions will not be noticed.

We propose a supervised model that makes further contributions to the third category of strategies. More specifically, we propose a partition sparse factor model (PSFM) with four main features: (i) a partition model to divide the multi-dimensional array covariates into subarrays, (ii) a factor model to reduce the subarrays into low-dimensional feature vectors, (iii) a generalized linear model using the feature vectors to predict clinical outcomes, and (iv) a sparse inducing normal mixture prior to select informative feature vectors.

By partitioning each image \( \mathcal{I} \) into equal sized sub-images \( \mathcal{I}^{(s)} \) such that \( \mathcal{I} = \bigcup_s \mathcal{I}^{(s)} \), not only we are more likely to capture small clustered effect regions, but additionally, the proposed model massively reduces the ultra-high dimensionality of the data. Consider a typical image of size \( 128 \times 128 \times 128 \) with 2,097,152 voxels. Suppose we partition the images into 512 sub-images of size \( 16 \times 16 \times 16 \) and apply the low-rank feature extraction in each sub-image. Then, the number of features is reduced dramatically, and it is now viable. Figure 4.1 shows a 3D image partitioned into 24 subsets.

The rest of the article is organized as it follows. In Section 4.2.1, we introduce the model and its associated prior distributions and Bayesian estimation procedure. In Section 4.3, we use synthetic data to compare the proposed partition model with other dimension reduction models. In Section 4.4, we apply our model to a real imaging data set on attention deficit hyperactivity disorder (ADHD). Finally, in Section 4.5, we present some concluding remarks.
4.2 Model

4.2.1 Model Description

Let \( \{I, Z, y\} \) be a set of imaging variables, clinical covariates and an univariate response variable, respectively. For example \( I = \{\text{MRI}\} \), \( Z = \{\text{age, gender, height, etc...}\} \) and \( y = \{\text{disease outcome}\} \). Assume each image \( I \) is partitioned into subsets \( I^{(s)} \), such that \( \bigcup I^{(s)} = I \). Let \( x = \text{vec}(I) \) and \( X \) be a matrix such that each row \( x_i \) corresponds to the imaging information for subject \( i = 1, \ldots, n \). We assume that

\[
X^{(s)} = L^{(s)}(\Lambda^{(s)} C^{(s)}) A^{(s)} + E^{(s)} \tag{4.1}
\]

\[
y_i \sim \text{Exponential family}(\mu_i, \phi),
\]

\[
E(y_i | g_i^{(1)}, \ldots, g_i^{(N_s)}, z_i) = \mu_i,
\]

\[
h(\mu_i) = z_i^T \gamma + \sum_{s=1}^{N_s} g_i^{(s)} T b^{(s)}, \tag{4.2}
\]

where each row of the matrix \( L^{(s)} \) is a \( R \)-vector of common unobserved (latent) factors \( l^{(s)}_i \); the matrices \( \Lambda^{(s)} \) and \( C^{(s)} \) are \( R \times R \) diagonal matrices with \( c_{rr} \in \{0, 1\} \). The product \( \Lambda^{(s)} C^{(s)} \) is still a diagonal matrix and plays a role analogous to the singular values of \( X^{(s)} \);
$A^{(s)} \in \mathbb{R}^{R \times P}$ correspond to the matrix of $R$ latent basis functions used to represent $X^{(s)}$; $E^{(s)}$ is a matrix representing idiosyncratic errors.

The response variable $y$ is associated with the images $I$ only through the $R$ latent factors in each $G^{(s)} = L^{(s)} \Lambda^{(s)} C^{(s)}$ and it follows an exponential family distribution with density $f(y|\theta) = h(y) \exp[\eta(\theta)^T y - a(\theta)]$. Its expected value $\mu_i$ relates to the covariates through the link function $h$ as in 4.2. The vector $b^{(s)}$ consists of coefficients associated with $g_i$’s. Finally, $z_i$ is a $1 \times q$ vector of clinical predictors and $\gamma$ is a $q \times 1$ vector of coefficients associated with $z_i$.

There are two sparse components in the proposed model. First, since $C^{(s)}$ is binary, $\Lambda^{(s)} C^{(s)}$ is expected to be sparse if $R$ is chosen large enough, therefore, only some columns of $L^{(s)}$ are important to represent $X^{(s)}$. Second, we introduce bimodal sparsity promoting priors on the coefficients $b^{(s)}$. This is necessary since it is often the case where $N_s \times R \gg N$ and the model in (4.2) is non-identifiable. These priors are key elements to deal with this identifiability issue (Mayrink and Lucas, 2013; George and McCulloch, 1993, 1997) and they determine whether a particular $g^{(s)}$ is important when modeling the association between the response variable and the corresponding imaging covariate. We assume that each $b_r^{(s)}$ is modeled as having come from a mixture of normal distributions in the following hierarchical approach

$$b_r^{(s)}|\delta_r^{(s)}, \sigma^2 \sim (1 - \delta_r^{(s)})N(0, \epsilon) + \delta_r^{(s)}N(0, \sigma^2) \tag{4.3}$$

$$\delta_r^{(s)}|\pi \sim \text{bernoulli}(\pi)$$

$$\pi \sim \text{beta}(\alpha_0, \alpha_1),$$

The prior probability $\pi$ determines whether a particular $g_r^{(s)}$ should be included in the model. A prior for $\pi$ that concentrates most of its mass in the extremes of the interval $(0, 1)$ is ideal for variable selection and it is achieved by choosing a "bathtub" shaped beta distribution with both hyperparameters $\alpha_{0\pi}$ and $\alpha_{1\pi}$ taken to be less than 1 (Gonalves et al., 2013).
4.2.2 Prior distributions

Let \( L^{(s)} = \{ l_1^{(s)}, \ldots, l_R^{(s)} \} \), \( A^{(s)} = \{ a_1^{(s)}, \ldots, a_p^{(s)} \} \), \( e_{ij}^{(s)} \) be the elements of \( E^{(s)} \) for \( i = \ldots, n \), \( j = 1, \ldots, p \) and \( I_k \) be an identity matrix of size \( k \). For each fixed partition \( s = 1, \ldots, N_s \), and fixed \( r = 1, \ldots, R \) we assume

\[
\begin{align*}
L_r^{(s)} & \sim \mathcal{N}(0, n^{-1}I_n) \\
\lambda_{rr}^{(s)} & \sim \mathcal{N}\left(0, \left(\tau_{\lambda}^{(s)}\right)^{-1}\right) \\
\tau_{\lambda}^{(s)} & \sim \text{Gamma}(\beta_{0\tau_{\lambda}}, \beta_{1\tau_{\lambda}}) \\
\epsilon_{rr}^{(s)} & \sim \text{Bernoulli}(\pi_{e}^{(s)}) \\
\pi_{e}^{(s)} & \sim \text{Beta}(\alpha_{0\pi_{e}}, \alpha_{1\pi_{e}}) \\
A_r^{(s)} & \sim \mathcal{N}(0, R^{-1}I_R) \\
e_{ij}^{(s)} & \sim \mathcal{N}\left(0, \left(\tau_{e}^{(s)}\right)^{-1}\right) \\
\tau_{e}^{(s)} & \sim \text{Gamma}(\beta_{0\tau_{e}}, \beta_{1\tau_{e}}),
\end{align*}
\]

where \( I_K \) represents the identity matrix of size \( K \). When \( n \) and \( R \) are large, the columns of the matrices \( L^{(s)} \) and \( A^{(s)} \) are approximately orthogonal, which is consistent with their role in the decomposition. However, it is only imposed here that the columns of these matrices span the space of the principal vectors, without explicitly requiring orthonormality (Xinghao Ding and Carin, 2011).

For the remaining elements of (4.2) we assume

\[
\gamma \sim \mathcal{N}(0, \nu^{-1}I_q), \quad \nu \sim \text{Gamma}(\nu_{0\nu}, \nu_{1\nu}).
\]

4.2.3 Posterior Inference

Let \( L = \{ L^{(1)}, \ldots, L^{(N_s)} \} \), \( \Lambda = \{ \lambda_r^{(1)}, \ldots, \lambda_r^{(N_s)}; r = 1, \ldots, R \} \), \( C = \{ c_r^{(1)}, \ldots, c_r^{(N_s)}; r = 1, \ldots, R \} \), \( A = \{ A^{(1)}, \ldots, A^{(N_s)} \} \), \( \tau_e = \{ \tau_{e}^{(1)}, \ldots, \tau_{e}^{(N_s)} \} \), \( \tau_{\gamma} = \{ \tau_{\gamma}^{(1)}, \ldots, \tau_{\gamma}^{(N_s)} \} \), \( \pi_e = \{ \pi_{e}^{(1)}, \ldots, \pi_{e}^{(N_s)}; r = 1, \ldots, R \} \), \( b = \{ b^{(1)}, \ldots, b^{(N_s)} \} \), \( \delta = \{ \delta^{(1)}, \ldots, \delta^{(N_s)} \} \) and \( G = \{ G^{(1)}, \ldots, G^{(N_s)} \} \). Consider \( \theta = \{ L, \Lambda, C, A, \tau_e, \tau_{\gamma}, \pi_e, b, \delta, \pi, \gamma, \nu \} \). A Gibbs sampler algorithm is proposed to generate a sequence of random observations from the joint posterior.
The Gibbs sampler essentially involves sampling from a series of conditional distributions while each of the modeling components is updated in turn. Assume we divide the image into $N_s$ equal sized regions and let $Y_i \sim \text{bernoulli}(\mu_i)$ with link function $g$ as the probit function. With these assumptions it is possible to maintain conjugacy when sampling $B$ and $\gamma$ in 4.2, by simpling using the Chib augmentation method (Albert and Chib, 1993). The complete algorithm is as it follows.

(a.0) Generate $w$ from

\[
w_i|y_i = 0 \sim N\left(z_i^T \gamma + \sum_{s=1}^{N_s} g_i^{(s)} b^{(s)}, 1\right)
\]

truncated at the right by 0

\[
w_i|y_i = 1 \sim N\left(z_i^T \gamma + \sum_{s=1}^{N_s} g_i^{(s)} b^{(s)}, 1\right)
\]

truncated at the left by 0

for $s = 1 : N_s$

for $r = 1 : R$

(a.1) Update $l^{(s)}_r$ from its full conditional distribution

\[
l^{(s)}_r| \sim N(\mu^{(s)}_l, \Sigma^{(s)}_l), \quad \Sigma^{(s)}_l = \left(n I_n + \tau^{(s)}_e \sum_{j=1}^{p} (\lambda^{(s)}_{rr})^2 (c^{(s)}_{rr})^2 (a^{(s)}_{rj})^2\right)^{-1}
\]

and $\mu^{(s)}_l = \tau^{(s)}_e \sum_{j=1}^{p} \lambda^{(s)}_{rr} c^{(s)}_{rr} a^{(s)}_{rj} x^{s-r(s)}_j$,

where $x^{s-r(s)}_j = X^{(s)} - L^{(s)}(A^{(s)} C^{(s)}) a^{(s)}_j + c^{(s)}_{rr} \lambda^{(s)}_{rr} a^{(s)}_{rj} l^{(s)}_r$, for $j = 1, \ldots, P$
(a.2) Update $c_{rr}^{(s)}$ from its full conditional distribution

\[
c_{rr}^{(s)} | - \sim \text{Bernoulli}(u_1/(u_0 + u_1)),
\]

\[
u_1 = \pi_c^{(s)} \exp \left( -\frac{1}{2} \sum_{j=1}^{p} (\lambda_{rr}^{(s)})^2 (\delta_{rr}^{(s)})^2 \mathbf{r}_r^{(s)T} \mathbf{r}_r^{(s)} - 2\lambda_{rr}^{(s)} \mathbf{r}_r^{(s)T} \mathbf{x}_j^{*r(s)} \right),
\]

\[
u_0 = 1 - u_1
\]

(a.3) Update $\lambda_{rr}^{(s)}$ from its full conditional distribution

\[
\lambda_{rr}^{(s)} | - \sim \mathcal{N} \left( \frac{\tau_{\lambda}^{(s)} \Sigma_{\lambda}^{(s)}}{\tau_{\lambda}^{(s)} + \Sigma_j^{(s)}}, \frac{\Sigma_j^{(s)}}{\tau_{\lambda}^{(s)} + \Sigma_j^{(s)}} \right)
\]

where $\Sigma_{\lambda}^{(s)} = \left( \tau_{\lambda}^{(s)} + \tau_{\epsilon}^{(s)} \sum_{j=1}^{p} (c_{rr}^{(s)})^2 (\delta_{rr}^{(s)})^2 \mathbf{r}_r^{(s)T} \mathbf{r}_r^{(s)} \right)^{-1}$

(a.4) Update $a_{rj}^{(s)}$ from its full conditional distribution

\[
a_{rj}^{(s)} | - \sim \mathcal{N} \left( \frac{\tau_{\lambda}^{(s)} \Sigma_{\lambda}^{(s)}}{\tau_{\lambda}^{(s)} + \Sigma_j^{(s)}}, \frac{\Sigma_j^{(s)}}{\tau_{\lambda}^{(s)} + \Sigma_j^{(s)}} \right)
\]

where $\Sigma_{a}^{(s)} = \left( 1 + \tau_{\epsilon}^{(s)} \sum_{j=1}^{p} (\lambda_{rr}^{(s)})^2 (\delta_{rr}^{(s)})^2 \mathbf{r}_r^{(s)T} \mathbf{r}_r^{(s)} \right)^{-1}$, for $j = 1, \ldots, P$

(a.5) Update $\delta_{rr}^{(s)}$ from its full conditional distribution

\[
\delta_{rr}^{(s)} \sim \text{Bernoulli}(\bar{p}_1/\bar{p}_1 + \bar{p}_0)
\]

where $\bar{p}_1 = \pi \exp \{-1/(2\sigma^2)(b_r^{(s)})^2\}$ and $\bar{p}_0 = \pi \exp \{-1/(2\epsilon)(b_r^{(s)})^2\}$

(a.6) Update $b_r^{(s)}$ from its full conditional distribution

\[
b_r^{(s)} | \delta_{rr}^{(s)} = 1 \sim \mathcal{N} \left( \sum_i \bar{w}_i^{(s)} g_{ir}^{(s)} / \sum_i (g_{ir}^{(s)})^2 + 1/\sigma^2, (\sum_i (g_{ir}^{(s)})^2 + 1/\sigma^2)^{-1} \right)
\]

\[
b_r^{(s)} | \delta_{rr}^{(s)} = 0 \sim \mathcal{N} \left( \sum_i \bar{w}_i^{(s)} g_{ir}^{(s)} / \sum_i (g_{ir}^{(s)})^2 + 1/\epsilon, (\sum_i (g_{ir}^{(s)})^2 + 1/\epsilon)^{-1} \right)
\]
\[ \tilde{w}_i^{(s)} = w_i - z_i^T \gamma - \sum_{s'=1}^{N_s} g_i^{(s')} b(s') + g_{ir}^{(s)} b_r^{(s)}. \]

\textbf{end} \% \( r = 1 : R \)

(a.7) Update \( \pi_c^{(s)} \) from its full conditional distribution

\[ \pi_c^{(s)}| \sim \text{beta} \left( \alpha_0 \pi_c + c_r^{(s)}, \alpha_1 \pi_c + 1 - c_r^{(s)} \right) \]

(a.8) Update \( \tau_\lambda^{(s)} \) from its full conditional distribution

\[ \tau_\lambda^{(s)}| \sim \text{Gamma} \left( \beta_{0\lambda} + 0.5 R, \beta_{1\lambda} + 0.5 \sum_{r=1}^{R} (\lambda_r^{(s)})^2 \right). \]

(a.9) Update \( \tau_e^{(s)} \) from its full conditional distribution

\[ \tau_e^{(s)}| \sim \text{Gamma} \left( \beta_{0e} + 0.5 np, \beta_{1e} + 0.5 \| X^{(s)} - L^{(s)} (\Lambda^{(s)} C^{(s)}) A^{(s)} \|_F \right). \]

\textbf{end} \% \( s = 1 : N_s \)

(a.10) Update \( \pi \) from its full conditional distribution

\[ \pi| \sim \text{beta} \left( \alpha_0 \pi + \sum_{s,r} \delta_r^{(s)}, \alpha_1 \pi + N_s R - \sum_{s,r} \delta_r^{(s)} \right) \]

(a.11) Update \( \gamma \) from its full conditional distribution

\[ \gamma| \sim \text{N} \left( \Sigma^*^{-1} Z^T w^*, \Sigma^*^{-1} \right), \]

where \( \Sigma^* = v I_q + Z^T Z \) and \( w^* = w - \sum_{s=1}^{N_s} g_i^{(s)} T b_r^{(s)}. \)

(a.12) Update \( v \) from its full conditional distribution

\[ v| \sim \text{Gamma} \left( \nu_0 v + q/2, \nu_1 v + (\gamma^T \gamma)/2 \right). \]
4.3 Simulation Study

We carried out a simulation study to examine the properties of the proposed model. The goals of this simulation study are

(G.1) To compare the partition model with the 2-stage PCA model and with the non-partition model;

(G.2) To illustrate the fast convergence of the Gibbs sampler algorithm;

(G.3) To examine if the proposed model is able to recover location where differences between groups exist;

For each subject \(i = 1, \ldots, 200\) we generate images \(I_i \in \mathbb{R}^{38 \times 48 \times 36}\) assuming the following

\[
y_i \sim \text{bernoulli}(0.5) \quad I_i = X_0 + X_1(y_i) + \epsilon_i,
\]

where \(X_0\) is a brain image template with values ranging from \([0, 250]\) and \(X_1(1)\) is shown in panel (a) of Figure 4.3 and \(X_1(0)\) is zero in every voxel. The elements of \(\epsilon_i\) are assumed independent and \(\epsilon_{ijk} \sim N(0, 50^2)\). The hyperparameters in Section 3.2.3 are chosen to reflect non-informative priors with \(\beta_{0\tau_\alpha} = \beta_{0\tau_e} = \nu_{0\nu} = 1, \beta_{1\tau_\alpha} = \beta_{1\tau_e} = \nu_{1\nu} = 10^{-4}\), \(\alpha_{1\pi_c} = \alpha_{0\pi_e} = 0.5\).

We consider the model described in (4.1) and (4.2) with \(N_s = 18, N_s = 8\) and \(N_s = 1\) (non-partition) model and compare them with the model suggested by Caffo et al. (2010), where we learn the principal components on a first stage and then consider the \(R\) most important principal components as inputs in a generalized linear model framework on a second stage. We also consider cases where \(\Lambda^{(s)} = I_R\) and \(C^{(s)} = I_R\), for \(s = 1, \ldots, N_s\). For both partition models we consider \(R = 20, 50, 100\) and 250 and for the non-partition model \(R = 100, 250\) and 500 and run the MCMC algorithm described in Section 3.2.4 with 1500 iterations and burn-in period of 1000.
Figure 4.2 shows the trace plots of the projection $\mathcal{P} = \bigcup_s P^{(s)}$, where $P^{(s)} = (A^{(s)})^T b^{(s)}$ for 9 random voxels of the partition model with $N_s = 8$, $R = 50$ after burn-in, illustrating the fast convergence of the model. In addition, the deviance information criteria (DIC) is computed (Spiegelhalter et al., 2002) and shown in Table 4.1. From the table, we observe that the models with $N_s = 18$ have the smaller DIC values overall and DIC decreases as the the value of $R$ increases, as expected.

In Figure 4.3 we display the results for the PCA model and BSPFM $N_s = 1$, $R = 100$, $N_s = 8$, $R = 50$ and $N_s = 18$, $R = 20$, with general matrices $Z$ and $C$. Panels (c)-(e) represent the posterior mean of the projection $\mathcal{P}$ for each model overlaid on the template $\mathcal{X}_0$. We observe that the proposed partition models with $N_s = 8$ and $N_s = 18$ are able to precisely recover the true underlying signal $\mathcal{X}_1(1)$ shown in panel (a), while the PCA models (panel (b)) perform poorly in distinguishing the strength of the true underlying signal.
Figure 4.3: Simulation results for the Bayesian sparse partition factor model (BSPFM) with $N_s = 18$, $N_s = 8$ and $N_s = 1$. Panel (a) shows the true underlying signal $X_1$ overlaid on the template $X_0$; panel (b) shows the estimated projection for the PCA model and panels (c)-(e) are posterior mean of the projection for the partition and non-partition model, all results overlaid on the template $X_0$. 
Table 4.1: Deviance information criteria for the partition models $N_s = 8$ and $N_s = 18$ and the non-partition model. Based on the criteria, the models with $N_s = 18$ are preferred. The symbol ** indicates the models that did not converge.

4.4 Application to the ADHD data set

We applied our model to the Attention Deficit Hyperactivity Disorder (ADHD) data from the ADHD-200 Consortium. The consortium emerged when functional neuroimaging investigators working on ADHD came together to establish a large-scale, aggregate resting state fMRI dataset, along with accompanying anatomical and phenotypic data for children and adolescents with the disorder (Milham et al., 2012). The disorder is associated with substantial lifelong impairment and its symptoms include difficulty staying focused and paying attention, difficulty controlling behavior, and hyperactivity (over-activity).

We analyzed the data from New York University (NYU) site, which consists of 219 subjects, 99 controls and 120 diagnosed with ADHD. We considered the high-resolution T1-weighted images acquired using the MPRAGE (Magnetization-prepared Rapid Acquisition with Gradient Echo) technique. The original T1-weighted images have size $256 \times 256 \times 198$ mm$^3$ and voxel size of $1.0 \times 1.0 \times 1.0$ mm$^3$. For each subject, the images were first downsampled to the size of $64 \times 64 \times 50$ mm$^3$. Next, the images were processed using HAMMER (Hierarchical Attribute Matching Mechanism for Elastic Registration), a free pipeline developed by the Biomedical Research Imaging Center at UNC (available for downloading at http://www.hammersuite.com). The processing steps include skull and cerebellum removal, followed by tissue segmentation to identify the regions of white matter.
Table 4.2: Deviance information criteria for the partition models $N_s = 8$ and $N_s = 24$ and the non-partition model. Based on the criteria, the partition models are preferred.

For both partitions, we run the model described in equations (4.1) and (4.2) with $I = \{\text{WM Ravens map}\}$, $Z = \{\text{age, gender}\}$ and $y = \{\text{ADHD outcome}\}$ for different values of the rank $R$. We compared the results from the partition model with those from the no partition model and across different values of $R$. Table 4.2 shows the corresponding deviance information criteria DIC for all models considered. Inspecting the table reveals that the partition models are preferred over the non-partition model, and BPPFM($N_s = 24$) presents a smaller DIC in general. In addition, the estimation of the diagonal matrices $Z$ and $\Lambda$ shows an improvement on the DIC when compared to the traditional factor model where $Z = I$ and $\Lambda = I$.

Figures 4.4 and 3.8 shows the results of BPPFM($N_s = 24$), $R = 200$ for general diagonal matrices $Z$ and $\Lambda$. Panel (b) of Figure 4.4 shows selected axial slices of the
posterior mean estimates for the projection \( \mathcal{P} = \bigcup_s P^{(s)} \), where \( P^{(s)} = (A^{(s)})^T b^{(s)} \).

Panel (a) shows the same axial slices with the estimated projection for the 2-stage PCA model. The colors indicate the strength of the differences between children with ADHD and control.

![Image](image-url)

Figure 4.4: ADHD data analysis results for the Bayesian partition partial factor model (BPPFM) with \( N_s = 24 \) partitions and rank \( R = 200 \). Panel (b) shows the results of the posterior mean of the projection \( \mathcal{P} \) and panel (a) shows the estimated projection for the 2-stage PCA model. The colors indicate the strength of the differences between children with ADHD and control.

Figure 3.8 shows the credible intervals for the projection \( \mathcal{P} \) overlaid on the Jacob template. The red color indicates the locations where the credible interval for the projection does not contain 0. Panel (a) shows a 3D rendering of the 95% credible interval for the projection while panel (b) shows a 90% credible interval. Inspecting Figure 4.5 we are able to detect two large regions of interest where morphological differences exist, including the right frontal lobe and left parietal lobe. The frontal lobe has been implicated in planning complex cognitive behavior, personality expression, decision making and moderating social behavior (Yang and Raine, 2009) and morphological differences in this region were previously identified in children with ADHD (Sowell et al., 2003; Miranda et al., 2013).
Figure 4.5: ADHD data analysis results for the Bayesian partition partial factor model (BPPFM) with $N_s = 24$ partitions and rank $R = 200$. Panel (a) and (b) show the results of a 3D rendering of the 95% and 90% credible intervals for the projection overlaid on the Jacob template, respectively; panels (c) and (d) are selected axial slices of the 90% and 95% credible interval for the projection, respectively.
4.5 Discussion

We have proposed a partition sparse factor regression modeling framework to establish association between clinical outcomes and ultra-high dimensional covariates. PSFM is a supervised hierarchical model with four major components chosen to address the limitations faced by ultra-high dimension neuroimaging data. The main contribution is the partition component, which not only allow us to capture small clustered effect regions, but also massively reduces data dimensionality to manageable levels.

The simulation study demonstrates that PSFM outperforms the most common dimension reduction technique and that models with smaller partitions perform better on recovering the true underlying signal. Ultimately, the proposed model is able to identify meaningful locations in the brain, where differences in white matter are confirmed between typical developing children and children with attention deficit hyperactivity disorder.
CHAPTER 5: FUTURE WORK

In this dissertation, novel methodologies were developed to deal with the challenges brought by complex neuroimaging studies. This chapter presents a few open problems related to methods proposed here.

5.0.1 Spatial transformation models

The spatial transformation parameter $\Lambda$ presents a very clear spatial structure, and a prior that accounts for spatial correlation should be considered. However, it is challenging to sample the components $\lambda_d$’s even when we assume independence \textit{a priori}. The development of a better sampling algorithm is essential to solve this issue.

The strength of correlation of the components of the GMRF, $\phi_k$, should be estimated from the data. Although it is possible to write its full conditional distribution, it is indispensable to compute the eigenvalues of the $N_d \times N_d$ matrix $H_k$. This computation is not viable for neuroimaging datasets. Alternative methods such as empirical Bayes should be taken into account.

5.0.2 Low-rank models

An automated way of establishing the partitions would be a great contribution for both models proposed in chapters 3 and 4. So far, we based our model choice solely on the deviance information criteria. However, a model that optimizes the search for the best image subsets, and further performs partition selection simultaneously with feature selection, would be an excellent addition.

Theoretical properties of the proposed estimators are still not described, and inference on the projection space should be further investigated.
APPENDIX A: CHAPTER 2 SUPPLEMENTARY MATERIAL

A.1 Full conditionals derivations

1. Full conditional for $\tau_d, d = 1, \ldots, N_D$

We have $\pi(\tau_d) \sim \text{Gamma} \left( \frac{\delta_0}{2}, \frac{\tau_0}{2} \right)$, then we compute

$$
\pi(\tau_d | -) \propto \tau_d^{\frac{\delta_0}{2}-1} \tau_d^{\frac{n}{2}} \exp \left\{ -\frac{\tau_d}{2} \sum_{i=1}^{n} (y_i^{(\lambda d)}(d) - x_i^T \beta(d))^2 \right\} \exp \left\{ -\tau_d \gamma_0 \right\}
$$

$$
= \tau_d^{\frac{1}{2}(\delta_0+n)-1} \exp \left\{ -\frac{\tau_d}{2} \sum_{i=1}^{n} (y_i^{(\lambda d)}(d) - x_i^T \beta(d))^2 + \gamma_0 \right\}
$$

Hence the full conditional is $\text{Gamma} \left( \frac{n+\delta_0}{2}, \frac{1}{2} \sum_{i=1}^{n} (y_i^{(\lambda d)}(d) - x_i^T \beta(d))^2 + \gamma_0 \right)$

2. Full conditional for $\beta_k(d)$, for $k = 1, \ldots, p$ and $d = 1, \ldots, N_D$

$$
\pi(\beta_k(d) | -) \propto \exp \left\{ -\frac{\tau_d}{2} \sum_{i=1}^{n} (y_i^{(\lambda d)}(d) - X_{ik} \beta_k(d) - \sum_{l \neq k} X_{il} \beta_l(d))^2 \right\}
$$

$$
\times \exp \left\{ -\frac{\theta_k(d)}{2} (\beta_k(d) - \mu_k(d))^2 \right\}
$$

$$
\propto \exp \left\{ -\frac{\tau_d}{2} \sum_{i} \left[ -2y^*_i x_{ik} \beta_k(d) + x_{ik}^2 \beta_k^2(d) \right] - \frac{\theta_k(d)}{2} \left[ \beta_k^2(d) - 2 \beta_k(d) \mu_k(d) \right] \right\}
$$

$$
= \exp \left\{ -\frac{1}{2} \left( \beta_k^2(d) \tau_d \sum_{i} x_{ik} + \theta_k(d) \right) - \beta_k(d) \left[ \tau_d \sum_{i} y^*_i x_{ik} + \theta_k(d) \mu_k(d) \right] \right\}
$$

$$
= \exp \left\{ -\frac{1}{2} \left( \beta_k^2(d) \tau_d \sum_{i} x_{ik} + \theta_k(d) \right) \left( \beta_k^2(d) - 2 \beta_k(d) \left[ \tau_d \sum_{i} y^*_i x_{ik} + \theta_k(d) \mu_k(d) \right] \right) \right\}
$$

$$
= \exp \left\{ -\frac{1}{2} \left( \beta_k^2(d) \tau_d \sum_{i} x_{ik} + \theta_k(d) \right) \left( \beta_k^2(d) - \frac{\tau_d \left[ \sum_{i} y^*_i x_{ik} + \theta_k(d) \mu_k(d) \right]}{\tau_d \sum_{i} x_{ik} + \theta_k(d)} \right)^2 \right\}
$$

Where

$$
y^*_i = y_i^{(\lambda d)}(d) - \sum_{l \neq k} X_{il} \beta_l(d)
$$

Hence, the full conditional of $\beta_k(d)$ is

$$
N \left( \frac{\tau_d \sum_{i} y^*_i x_{ik} + \theta_k(d) \mu_k(d)}{\tau_d \sum_{i} x_{ik}^2 + \theta_k(d)}, \frac{1}{\tau_d \sum_{i} x_{ik}^2 + \theta_k(d)} \right).
$$
3. Full conditional for $\nu_k$, $k = 1, \ldots, p$

$$
\pi(\nu_k | -) \propto \nu_k^{N_D / 2} \exp \left\{ -\frac{\nu_k}{2} \left[ \frac{N_D}{\sum_{j=1}^{N_D} \beta_{jk}^2 + \phi_k \beta_k^T H \beta_k} \right] \right\} \nu_k^{n_{\nu} / 2 - 1} \exp \left\{ -\frac{n_{\nu} S_{\nu}^2}{2} \right\}
$$

$$
= \nu_k^{\frac{1}{2} (N_D + n_{\nu} - 1)} \exp \left\{ -\nu_k \frac{\sum_{j=1}^{N_D} \beta_{jk}^2 + \phi_k \beta_k^T H \beta_k + n_{\nu} S_{\nu}^2}{2} \right\}
$$

A.2 Sensitivity Analysis

In this appendix, we present a sensitivity analysis of the hyper-parameters $a$ and $b$ of $\lambda = \{\lambda_d; d \in D\}$ and the hyper-parameters $\phi_k$ of $\beta$. There are two goals. One is to examine the finite sample performance of STM and its associated parameter estimates under different scenarios. The other is to evaluate MCMC convergence through a diagnostic analysis.

Sensitivity analysis for $\Lambda$. We consider three different scenarios for $(a, b)$ including $(-2.0, 2.5)$, $(-3.0, 3.0)$ and $(-3.5, 3.5)$. In most applications, the three scenarios of $(a, b)$ represent a reasonable range of $\lambda$. Although it may be desirable to use a wider interval $(a, b)$, very flat priors can lead to slow convergence of the MCMC algorithm. We examine how STM recovers the geometric patterns presented in Section 3. Figure A.1 reveals that regardless of the different choices of $a$ and $b$, the STM is able to capture the true underlying pattern. Thus, STM is robust to the choice of the hyperparameters of $\lambda_d$.

Geweke diagnostic statistics. Under each scenario, we evaluate convergence at each voxel through the Geweke diagnosis statistics (Geweke, 1992). Table A.1 presents the percentages of voxels, whose Geweke diagnosis statistics, computed after 1000 iterations of the Markov chain, are smaller than 1.96 (in absolute value). The numbers are shown to be very similar across the three scenarios for all parameters. Compared with other parameters, the $\beta$’s associated with the indicator variables $\beta_2(d)$ and $\beta_3(d)$ have a smaller proportion of voxels that converge.
Figure A.1: Sensitivity analysis of $\Lambda$. Panels (a)-(d) represent the true pattern of $\beta$ used to generate the images; (e)-(h): the posterior means of $\beta$ obtained with $(a, b) = (-2.0, 2.5)$; (i)-(l): the posterior means of $\beta$ obtained with $(a, b) = (-3.0, 3.0)$; and (m)-(p): the posterior means of $\beta$ with $(a, b) = (-3.5, 3.5)$.

Table A.1: Sensitivity analysis for $\lambda$ indicating the percentages of voxels, whose Geweke diagnosis statistics are smaller than 1.96, according to the Geweke diagnosis statistics (Geweke, 1992) for each scenario considered.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>$(-2.0, 2.5)$</th>
<th>$(-3.0, 3.0)$</th>
<th>$(-3.5, 3.5)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda$</td>
<td>96.09</td>
<td>95.21</td>
<td>96.00</td>
</tr>
<tr>
<td>$\tau$</td>
<td>94.53</td>
<td>94.63</td>
<td>93.65</td>
</tr>
<tr>
<td>$\beta_0$</td>
<td>94.73</td>
<td>94.92</td>
<td>94.34</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>93.55</td>
<td>94.63</td>
<td>94.92</td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>89.06</td>
<td>88.48</td>
<td>89.65</td>
</tr>
<tr>
<td>$\beta_3$</td>
<td>89.06</td>
<td>88.28</td>
<td>89.94</td>
</tr>
</tbody>
</table>
Trace plots for $\nu_k$. We present the trace plots for the parameters $\nu_k$ associated with each $\beta_k$ under the three scenarios in Figure A.2. Figure A.2 reveals that the MCMC chains converge fast and the posterior estimates of $\nu_k$ converge to their true values.

Figure A.2: Trace plots of $\nu_k$ for $k = 0, \ldots, 3$ after a burn-in of 50 iterations and a total of 1000 MCMC iterations under the three scenarios of $(a, b)$. Rows 1-3 correspond to $(a, b) = (-2.0, 2.5), (a, b) = (-3.0, 3.0)$, and $(a, b) = (-3.5, 3.5)$, respectively.

Trace plots of $\beta$, $\tau_\sigma$ and $\lambda$. Figure A.3 presents the trace plots of $\beta$, $\tau_\sigma$ and $\lambda$ for the scenario $(a, b) = (-3.5, 3.5)$ in some selected voxels. For the sake of space, we omit their trace plots for other scenarios and voxels, since they are essentially similar to each other. Figure A.3 reveals that the single-site Gibbs sampler algorithm has good convergence properties.

Sensitivity analysis for $\phi_k$. There are two strategies of determining $\phi_k$. First, for small and moderate $N_D$, it is possible to integrate $\phi_k$ into the Gibbs sampler by sampling from the full conditional distribution of $\phi_k$, which is proportional to $p(\beta_k | \nu_k, \phi_k)p(\phi_k)$, where $p(\phi_k)$ is the prior of $\phi_k$. Different choices of $\phi_k$ have been discussed in Ferreira and De Oliveira (2007). The conditional distribution for $\phi_k$ does not have a simple form, but it can be easily sampled using the slice sampler (Neal, 2003). Sampling $\phi_k$ requires the
Figure A.3: Trace plots of $\beta$, $\tau_\sigma$ and $\lambda$ for the scenario $(a, b) = (-3.5, 3.5)$ at 4 random selected voxels. The results show fast convergence of the MCMC chain for all parameters.
computation of the eigenvalues of a sparse \( N_D \times N_D \) matrix \( H_k \). For an extremely large \( N_D \), calculating the eigenvalues of \( H_k \) can be computationally infeasible. Second, it is common to pre-specify \( \phi_k \) in many applications. Thus, it is important to evaluate the effects of different hyperparameters \( \phi_k \) on parameter estimates.

Inspecting Figure A.4 reveals that as \( \phi_0 \) increases, the posterior estimates of \( \beta_0 \) get worse, whereas there is no visual difference for other parameter estimates under different \( \phi_k \). It is expected that the estimation of the model parameters becomes more and more difficult when \( \phi_k \) is large, since the effective sample size decreases as the correlation among observations increases. It results in a decrease of useful information about the parameters of interest, which is contained in the data (Ferreira and De Oliveira, 2007).

### A.3 Examining the effects of different \( H_k \) on the parameter estimates of \( \beta \)

Recall that \( H_k \) is given by

\[
h_k(d, d') = \begin{cases} 
\sum_{d' \in N(d)} \omega_k(d, d')^2, & \text{for } d = d', \\
-\omega_k(d, d')^2 \mathbf{1}(d' \in N(d)), & \text{for } d \neq d'.
\end{cases}
\]

Throughout the paper we consider \( \omega_k(d, d') = K(||d - d'||_2), \) where \( K(u) = \exp\left(-\frac{1}{2}u^2\right) \mathbf{1}(u \leq 2) \). The following possibilities are considered here:

- **(H.1)** A constant kernel \( K(u) = \mathbf{1}(u \leq 2) \), meaning all neighbors of the voxel \( d \) are given the same weights, \( \omega_k(d, d') = \mathbf{1}(||d - d'||_2 \leq 2) \);

- **(H.2)** The Gaussian kernel \( K(u) = \exp\left(-\frac{1}{2}u^2\right) \mathbf{1}(u \leq r_0), \) for \( r_0 = 0, 4, 6 \).

Other hyperparameters were chosen as described in Section 3 of the paper. As shown in Figure A.5, the corresponding estimates of the intercept in all cases are quite poor, whereas the estimates for the remaining parameters are quite accurate for all cases considered.

**Non-transformation model results**

Our goal is to examine the parameter estimates obtained from STM, when the true model corresponds to \( \lambda_d = 1 \) for all \( d \in \mathcal{D} \). Figure A.6 shows that the STM can reliably
Figure A.4: Posterior estimates of $\beta$ for different values of $\phi_k$: (a)-(d): $\phi_k = 0.01$; (e)-(h) $\phi_k = 0.1$; (i)-(l) $\phi_k = 1$; and (m)-(p): $\phi_k = 100$. 
Figure A.5: Posterior estimates of $\beta$ under different specifications of $H_k$. Panels (a)-(d): case (H.1); (e)-(p): case (H.2) for $r_0 = 0, 4, 6$. 
recover the true pattern in the $\beta$ images. Figure A.7 reveals that the estimated $\lambda_d$’s are close to the true value $1$.

![Figure A.6: The posterior estimates of $\beta$ for the true model with $\lambda_d = 1$ for all $d \in D$.](image)

![Figure A.7: The posterior estimated image $\hat{\Lambda} = \{\hat{\lambda}_d : d \in D\}$ for the true underlying model with $\lambda_d = 1$ for all $d \in D$.](image)

A.4 Additional results

In this subsection we present the estimated $\hat{\beta}_k$ images for the intercept, gender, age, and ADHD status. Figures A.8, A.9, A.10 and A.11, respectively, show the results. The maps include the posterior mean, the standard deviation and the standardized images given by $\hat{\beta}_k/\text{std}(\hat{\beta}_k)$. 
Figure A.8: The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images corresponding to the intercept $\beta_0$ are shown from the left to the right, respectively.

Figure A.9: The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images corresponding to the gender $\beta_1$ are shown from the left to the right, respectively.

Figure A.10: The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images corresponding to the age $\beta_2$ are shown from the left to the right, respectively.
Figure A.11: The posterior mean, the posterior standard deviation (SD), and the standardized value (mean/SD) images for the ADHD status $\beta_3$ are shown from the left to the right, respectively.

A.5 How to run BSTM?

A.5.1 Simulation code

Software Information The code was developed in Matlab version 7.11.0.584 (R2010b) and is available for download at http://onlinelibrary.wiley.com/doi/10.1111/biom.12085/suppinfo

The file BayesianSTM.rar must contain the functions illustrated in Figure A.12. Connected boxes indicate a one-way dependency structure, from left to right, e.g. function MainSTM.m depends on Betad_sampler_sm.m, which depends on GMRFprior_sm.m.

Data setup The response variable should be assembled into a matrix of size $N_D \times (N + 3)$, with each row corresponding to one voxel and the first 3 columns corresponding to voxel ID, x-axis coordinate and y-axis coordinate, respectively. In the main program, the matrix is called Images. An example is given below.

Images(1:5,1:8)
ans =
1.0000 1.0000 1.0000 11.5266 11.2794 11.2476 10.7556 10.8397
2.0000 2.0000 1.0000 4.5181 4.4784 4.5583 4.5681 4.5474
3.0000 3.0000 1.0000 4.5524 4.6535 4.5718 4.5782 4.5165
4.0000 4.0000 1.0000 11.1420 11.4097 10.5087 11.6071 11.2333
5.0000 5.0000 1.0000 36.6839 34.0204 37.2111 37.4458 37.8917

72
Figure A.12: Illustration of the one-way dependency structure of the functions.
The matrix NeiStruc defines the neighborhood structure in the following way:

```matlab
>> [NeiStruc(1:5,1:9) NeiStruc(1:5,LastCol)]
ans =
1 1 1 1 2 33 34 0 0 ... 4
2 2 1 1 2 33 34 35 ... 6
3 3 1 2 3 4 34 35 36 ... 6
4 4 1 3 4 5 35 36 37 ... 6
5 5 1 4 5 6 36 37 38 ... 6
```

It has as many rows as the total number of voxels. The first 3 columns are identical to the first 3 columns of the matrix Images. Columns starting from the 4th indicates which voxels are part of the neighborhood, including the current voxel itself. The last column informs the total number of voxels belonging to the neighborhood, e.g. Voxels 1,2,33 and 34 belong to the neighborhood of voxel 1. Voxel 1 has 4 neighbors.

**Function Arguments**

- **d** = the current voxel
- **Y** = response matrix of size \( N_D \times (N + 3) \), as described previously
- **X** = covariate matrix of size \( N \times p \)
- **Beta_hat** = coefficients matrix sampled in the previous iteration, it’s of size \( N_D \times N \)
- **tau_sigma_hat** = sampled vector of \( \lambda \)’s from the previous iteration, it’s of size \( N_D \times 1 \)
- **nu_hat** = vector of components sampled in the current iteration; size \( p \times 1 \);
- **lambda Ini** = vector of transformation parameters sampled in the previous iteration; size \( N_D \times 1 \)
- **phi Ini** = vector of fixed hyperparameter \( \phi_k \)’s, size \( p \times 1 \)
\( H \) = matrix with neighborhood weights, size \( N_D \times N_D \)

constant = a value added when the simulation data is been generated to make sure one is generating positive values

**Running the code**

Running MainSTM.m will generate Figures A.13 and A.14. Here, the true values of \( \lambda_d \)'s are either 0.8, 1 or 1.2. Figure A.13 shows that a improvement is obtained by considering the proposed STM, when we compare the results with the voxel-wise linear model. One may try different values for \( \lambda \) and for the hyperparameters, as described in the paper and in the Web Supplementary material. The user must allow \( \lambda_d \)'s to be sampled within the correct support by specifying reasonable values its hyperparameters, i.e. the range of the uniform prior distribution. For some examples the slice sampler may stop working if the interval is too wide. For more details on that check the Matlab documentation for the function slicesample. The user is responsible for making sure the MCMC algorithm converges, by analyzing the trace plots of the posterior distributions and/or computing diagnostic statistics.

**A.5.2 Real data code**

The code was developed in Matlab version 7.11.0.584 (R2010b) and is available for download at http://onlinelibrary.wiley.com/doi/10.1111/biom.12085/suppinfo

The file RealDataSTM.rar must contain the functions illustrated in Figure A.15. Connected boxes indicate a one-way dependency structure, from left to right, e.g. function RealDataSTM.m depends on Betad_sampler_rd.m, which depends on GMRFprior_rd.m.

**Data setup**

The data setup should be done exactly how it is described in this current section. We provide the file DataSetup.m which gives an example of how to extract the information from your imaging data. The user is responsible to make the modifications according to one’s dataset. To run RealDataSTM.m one needs to upload 4 files: 1) a covariate matrix \( X \); 2) a response matrix \( Y \), with values extracted from your imaging data; 3) a matrix containing the neighborhood structure of your data, NeiMatr; 4) a matrix containing the weights given
Figure A.13: Simulation results obtained by running MainSTM.m. Panels (a)-(d) represent the pattern of $\beta$ used to generate the images; panels (e)-(h) are estimated $\beta$ obtained from the least squares estimator in Matlab; panels (i)-(l) are the posterior mean of $\beta$ obtained from our STM.

Figure A.14: Simulation results: the true $\Lambda = \{\lambda_d, d \in D\}$ pattern in the left panel and the estimated pattern in the right panel. Estimated image is smoother compared with the true image due to the nature of the uniform distribution assumed a priori.
The covariate matrix $X$

The covariate matrix is a $N \times p$ matrix, where each row corresponds to a subject and the columns correspond to each covariate information. The first column of $X$ corresponds to the intercept term and must be formed by 1’s, unless the intercept term is not to be estimated. The order in which the subjects are listed in the rows of the $X$ matrix must match the order in the $Y$ matrix as described in the next section.

The response matrix $Y$

The response variable consists of imaging data at each voxel and should be assembled into a matrix of size $N_D \times (N + 4)$, with each row corresponding to a voxel and the first 4 columns corresponding to voxel ID, x-axis coordinate, y-axis coordinate, z-axis coordinate, respectively. From columns 5 to the last one, the entries correspond to the neuroimaging measurements, for each subject $i = 1, \ldots, N$. The order in which the subjects are entered should be the same for each voxel, in a way that the whole information for Subject 1, for
example, is contained in Column 5. This order must be the same as in the covariate matrix $X$, e.g. Subject $i$ has its covariate information in the $i$-th row of matrix $X$ and imaging information in the $(i + 4)$-th column of matrix $Y$ as shown below.

\begin{verbatim}
1505 85 55 25 1 2 4 6 6 ...
1506 86 55 25 0 3 4 6 7 ...
1507 87 55 25 0 2 4 7 7 ...
1508 88 55 25 0 2 5 7 7 ...
1509 89 55 25 0 3 4 5 5 ...
1510 90 55 25 0 1 0 3 5 ...
\end{verbatim}

**The neighborhood structure NeiStruc**

The matrix NeiStruc defines the neighborhood structure and it has as many rows as the total number of voxels $N_D$. The first 4 columns are identical to the first 4 columns of the matrix $Y$. Columns starting from the 5-th indicate which voxels are part of the neighborhood, including the current voxel itself. The last column informs the total number of voxels belonging to the neighborhood, e.g. Voxels 1180,1181,1182,1204,... belong to the neighborhood of voxel 1505. Voxel 1505 has 27 neighbors, including itself.

\begin{verbatim}
1505 85 55 25 1180 1181 1182 1204 ... 27
1506 86 55 25 1181 1182 1183 1205 ... 27
1507 87 55 25 1182 1183 1184 1206 ... 24
1508 88 55 25 1183 1184 1185 1207 ... 19
1509 89 55 25 1184 1185 1208 1209 ... 14
1510 90 55 25 1185 1209 1210 1211 ... 13
\end{verbatim}

**The HMatrix**

The entries of the $N_D \times N_D$ HMatrix are defined as in Chapter 2 (2.3). The user can provide his/her own matrix $H$ by specifying different weights. Although this matrix is computed and stored in its full dense form, when running the file RealDataSTM.m, HMatrix
is converted to its sparse form that will be used throughout the MCMC procedure. This process speeds up convergence. The user can optionally upload the sparse form of the matrix, by making a simple change in the file RealDataSTM.m.

**Function Arguments**

The function arguments are exactly the same as described in Section A.5.1. The file DataSetup.m gives an example of how to construct this matrix.

**Comments**

1. The function slice sampler in Matlab may not work if the range of the prior for the elements of $\Lambda$ is too wide. If that occurs, one may try different initial values and/or a narrow prior distribution. For more details about Slice Sampling check the Matlab documentation.

2. It is not possible to know *a priori* how many iterations it will take for the chain to converge. The user is responsible for making sure the MCMC algorithm converges, by analyzing the trace plots of posterior distribution and/or computing diagnostic statistics.
APPENDIX B: CHAPTER 3 DERIVATIONS

B.1 Full conditionals derivations

1. Full conditional for $\tau^{(s)}$

We have $\tau^{(s)} \sim \text{Gamma}(\nu_{0r}, \nu_{1r})$, then we compute

$$
\pi(\tau^{(s)}|-) \propto \tau^{(s)} \frac{\nu_{0r}}{2} - 1 \tau^{(s)} \nu_{0r} \exp \left\{ -\frac{\tau^{(s)}}{2} \sum_{i=1}^{n} (x_{ij_1,...,j_D}^{(s)})^2 \right\} \exp \left\{ -\tau^{(s)} \nu_{1r} \right\}
$$

$$
= \tau^{(s)\nu_{0r} + (N \Pi_{d=1}^{D} p_d)/2} \exp \left\{ -\frac{\tau^{(s)}}{2} \sum_{i=1}^{n} (x_{ij_1,...,j_D}^{(s)})^2 + \nu_{1r} \right\}
$$

where $x_{ij_1,...,j_D}^{(s)} = \{A^{(s)-}\|A^{(1)}, A^{(2)}, \ldots, A^{(D)} \|A^{(s)}, L^{(s)}\}ij_1,...,j_D$.

2. Full conditional for $A^{(d)}_{(s)}$

We have Let $a_{jd^r}$ denote the elements of the matrix $A^{(d)}_{(s)}$. We omit the indexes $d$ and $s$ for a less cluttered notation. We have $a_{jd^r} \sim N(0, p_d^{-1})$, then

$$
\pi(a_{jd^r}|-) \propto \exp \left\{ -\frac{p_d}{2} a_{jd^r}^2 \right\} \exp \left\{ -\frac{\tau^{(s)}}{2} \sum_{j_1,...,j_{d-1}, j_{d+1},...,j_D} (x_{ij_1,...,j_D}^{*} - \lambda_r a_{j_1r} \ldots a_{jd^r} g_{ir})^2 \right\}
$$

$$
\propto \exp \left\{ -\frac{1}{2} \left( \tau^{(s)} \sum_{i} \sum_{j_1,...,j_{d-1}, j_{d+1},...,j_D} (-2x_{ij_1,...,j_D}^{*} \lambda_r a_{j_1r} \ldots a_{jd^r} g_{ir} + \lambda_r^2 a_{j_1r}^2 \ldots a_{jd^r}^2 g_{ir}^2) + p_d a_{jd^r}^2 \right) \right\}
$$

$$
\propto \exp \left\{ -\frac{\varphi_a}{2} a_{jd^r}^2 - 2 \frac{\tau^{(s)} \varphi_a}{\varphi_a - a_{jd^r}} \sum_{i} \sum_{j_1,...,j_{d-1}, j_{d+1},...,j_D} \frac{x_{ij_1,...,j_D}^{*} \lambda_r a_{j_1r} \ldots a_{jd^r} g_{ir}}{a_{jd^r}} \right\}
$$

where $x_{ij_1,...,j_D}^{*} = x_{ij_1,...,j_D} - \sum_{r=1}^{R} \lambda_r a_{j_1r} \ldots a_{jd^r} g_{ir} + \lambda_r a_{j_1r} \ldots a_{jd^r} g_{ir}$ and $\varphi_a = \tau^{(s)} \sum_{i} \sum_{j_{d+1},...,j_D} \left[ \lambda_r^2 a_{j_1r}^2 \ldots a_{jd^r}^2 g_{ir}^2 \right] + p_d$

3. Full conditional for $G^{(s)}$ Let $g_{ir}$ denote the elements of the matrix $G^{(s)}$. We omit the index $s$ for a less cluttered notation. We have $g_{ir} \sim N(0, N^{-1})$, then
where $x_{ij1\ldots j_D}^*$ is the same as above and $\phi_g = \tau(s) \sum_{i,j1\ldots j_D} \lambda_r a_{j_ir}^2 \ldots a_{j_Dr}^2 + N$

4. Full conditional for $\Lambda^{(s)}$

$$
\pi(\lambda_r | -) \propto \exp \left\{ -\frac{1}{2} \left[ \tau(s) \left( \sum_{i,j1\ldots j_D} -2x_{ij1\ldots j_D}^* \lambda_r a_{j_ir} \ldots a_{j_Dr} g_{ir} + \lambda_r^2 a_{j_ir}^2 \ldots a_{j_Dr}^2 g_{ir}^2 \right) + N g_{ir}^2 \right] \right\} \\
\propto \exp \left\{ -\frac{\phi_{\lambda}}{2}\left[ \lambda_r^2 - 2 \frac{\tau(s) \sum_{i,j1\ldots j_D} x_{ij1\ldots j_D}^* \lambda_r a_{j_ir} \ldots a_{j_Dr} g_{ir}}{\varphi_{\lambda}} \right] \right\}
$$

where $x_{ij1\ldots j_D}^*$ is the same as above and $\phi_{\lambda} = \tau(s) \sum_{i,j1\ldots j_D} \lambda_r^2 a_{j_ir}^2 \ldots a_{j_Dr}^2 + \kappa$

5. Full conditional for $\delta_r^{(s)}$

We have that $\delta_r^{(s)} | \sim p(\delta_r^{(s)} | y, b_r^{(s)}, \pi) = p(\delta_r^{(s)} | b_r^{(s)}, \pi)$. Notice that the distribution do not depend on $y$ because of the hierarchical structure, where $\delta$ only affects $y$ through $b$ (George and McCulloch, 1993). With a prior assumption that $\delta_r^{(s)} | \pi \sim \text{bernoulli}(\pi)$, we have

$$
\delta_r^{(s)} | - \sim \text{bernoulli}(\tilde{p}_1 / \tilde{p}_1 + \tilde{p}_0),
$$

where $\tilde{p}_1 = \pi p(b_r^{(s)} | \delta_r^{(s)} = 1) = \pi \exp\{-1/2\sigma^2(b_r^{(s)})^2\}$ and $\tilde{p}_0 = (1-\pi)p(b_r^{(s)} | \delta_r^{(s)} = 0) = \pi \exp\{-1/2\epsilon(b_r^{(s)})^2\}$. 

81
6. Full conditional for $b_r^{(s)}$

\[ b_r^{(s)}|\delta_r^{(s)} = 1, -\propto \exp \left\{ -\frac{1}{2} \sum_i (\tilde{w}_i^{(s)} - b_r^{(s)} g_{ir}^{(s)})^2 \right\} \exp \left\{ -\frac{1}{2} (b_r^{(s)})^2 \right\} \]

\[ \propto \exp \left\{ -\frac{1}{2} \left( \sum_i (g_{ir}^{(s)})^2 + (1/\sigma^2) \right) \left[ (b_r^{(s)})^2 - 2 b_r^{(s)} \frac{\sum_i \tilde{w}_i^{(s)} g_{ir}^{(s)}}{\sum_i (g_{ir}^{(s)})^2 + (1/\sigma^2)} \right] \right\} \]

where $\tilde{w}_i^{(s)} = w_i - Z_i^T \gamma - \sum_{s'=1}^{N_s} g_i^{(s')} T b_r^{(s')} + g_{ir}^{(s)} b_r^{(s)}$.

7. Full conditional for $\pi$ We have that $\pi \sim \text{beta}(\alpha_{0\pi}, \alpha_{1\pi})$, then

\[ \pi|\sim -\propto \pi^{(1-\pi\alpha_{0\pi})} \prod_{r,s} \delta_r^{(s)} (1 - \pi)^{1-\delta_r^{(s)}} \]

8. Full conditional for $\gamma$

\[ \gamma|\sim -\propto \exp \left\{ -\frac{1}{2} (w^* - Z \gamma)^T (w^* - Z \gamma) \right\} \exp \left\{ -\frac{\nu}{2} \gamma^T \gamma \right\} \]

\[ \propto \exp \left\{ -\frac{1}{2} (\gamma - \Sigma^{-1} Z^T w^*)^T \Sigma^{-1} Z^T w^*) \right\} \]

where $\Sigma = \nu I_q + Z^T Z$ and $w^* = w - \sum_{s=1}^{N_s} g_i^{(s)} T b_r^{(s)}$.

9. Full conditional for $\nu$

We have $\nu \sim \text{Gamma}(\nu_{0\nu}, \nu_{1\nu})$, then

\[ \nu|\sim -\propto \nu_{0\nu}^{\nu_{0\nu}-1} \exp \left\{ -\nu_{1\nu} \nu \right\} \nu^{\nu/2} \exp \left\{ -\frac{\nu}{2} \gamma^T \gamma \right\} . \]
REFERENCES


