Title

Statistical Inference for Large and Complex Data

Author

Zhou, Xinkai

Publication Date

2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Los Angeles

Statistical Inference for Large and Complex Data

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Biostatistics

by

Xinkai Zhou

2022

ABSTRACT OF THE DISSERTATION

Statistical Inference for Large and Complex Data

by

Xinkai Zhou
Doctor of Philosophy in Biostatistics
University of California, Los Angeles, 2022
Professor Hua Zhou, Chair

Statistical inference aims to quantify the amount of uncertainty in parameters or functions estimated from a statistical procedure and lies at the heart of modern decision-making. The problem is, however, when data sets become large and high-dimensional, which is the case for many modern health-related applications (electronic health records, multiomics, imaging data, etc.), classical statistical inference tools fail due to computational and methodological issues. The problem is further exacerbated when data sets also exhibit dependency structures or nonignorable missingness due to censoring. This dissertation summarizes our effort in addressing some of these challenges.

Specifically, chapter 1 provides a bag of little bootstrap (BLB) based method for conducting statistical inference of linear mixed models on massive and distributed longitudinal data sets such as electronic health records. For the statistical inference of variance component parameters, our software package `MixedModelsBLB.jl` achieves 200 times speedup on the scale of 1 million subjects (20 million total observations), and is the only currently available tool that can handle more than 10 million subjects (200 million total observations) using a desktop computer.

Chapter 2 provides an extremely flexible and general framework called proximal Markov

Chain Monte Carlo (ProxMCMC) for conducting statistical inference on constrained or regularized estimation procedures, which are indispensable for analyzing high-dimensional data and the inference of which has been considered difficult. Many frequently encountered statistical learning tasks such as constrained lasso, graphical lasso, matrix completion, and sparse low-rank matrix regression fall into this category.

Chapter 3 provides tools for the estimation and inference of heteroscedastic linear models for analyzing censored data using synthetic variables. Our motivating applications are adjusting for treatment effects in studies of quantitative traits and variance quantitative trait loci (vQTL) analysis, which arise frequently in genetic and epidemiological studies, but our method is general and computationally scalable to be applied to other fields of applications where censored data can arise from, for example, measurements that are out of the limit of detection.

The dissertation of Xinkai Zhou is approved.

Sudipto Banerjee

David Elashoff

Kenneth L. Lange

Hua Zhou, Committee Chair

University of California, Los Angeles

2022

To my parents and my wife.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

As the Chinese saying goes, “白驹过隙" (bái jū guò xì), which means “time flies like a galloping pony flashing through a slit", I am quickly approaching the finish line of my PhD journey. Looking back, the past 4 years have been thoroughly fulfilling and enjoyable. This would not have been possible without the people I am going to thank next.

First and foremost, I want to thank my advisor Dr. Hua Zhou, who has been my “captain” throughout this endeavor. He makes sure I am on the right track, offers me time and room to grow at my own pace, and has been extremely understanding, supportive, and encouraging, especially during difficult times like COVID-19. Finally, seeing how humble he is with everyone is a life lesson I will remember forever. He truly leads by example.

Next, I want to thank my collaborators Dr. Eric Chi, Dr. Gang Li, and Dr. Jin Zhou for their tremendous help in my research. Dr. Jin Zhou’s close knowledge of both the statistical and medical literature has provided the motivation for much of my work. Dr. Eric Chi and Dr. Gang Li generously offered their expertise and time to discuss projects with me. I would like to especially thank Dr. Eric Chi for his significant help in my postdoc-searching process.

I am also deeply indebted to my committee members Dr. Sudipto Banerjee, Dr. David Elashoff, and Dr. Kenneth Lange. Dr. David Elashoff brought me to UCLA before I even started the PhD program. He saw the potential in me and continued to support my growth every step of the way. I would not be where I am today without him. Dr. Sudipto Banerjee’s linear model class is arguably one of the best classes I have ever taken. His passion for statistics shines through every class and interaction I have had with him. Over the last 4 years, I had the privilege of observing and learning from Dr. Kenneth Lange through our group meetings. His wit and knowledge of mathematics have been a constant source of inspiration.

During my time at UCLA, I have benefited plenty from talking to Dr. Andrew Holbrook and Dr. Donatello Telesca. They shared with me stories of their academic endeavors, from

which I gained a fresh perspective and was able to make more informed decisions. I also want to thank Dr. Janet Sinsheimer, Dr. Eric Sobel, and Dr. Jeanette C. Papp for supporting me throughout this journey.

Over years I have been very lucky to meet a few professors and mentors who shared their life stories and wisdom with me when I needed them the most. They are Dr. Ian Abramson, Mr. Jianhua Hu, Dr. Tony Torng, Mrs. Wei Wang, and Dr. Lily Xu. Their perspective and candor have helped shape some of my biggest career decisions.

My experience at UCLA would have been monotonous and incomplete if it were not for the friends with whom I shared the journey with. First I want to thank a very special couple: Pengrui Quan and Yao Zhu, who have been my “family in LA". I will always remember what Pengrui said about “Seeing personal qualities as something that make us unique without giving them positive or negative labels." Next, I want to thank Nicholas Marco, with whom I shared an uncountable number of jokes, and I learned more about American culture from him than from anywhere else. I want to also thank Ewing Chen, Lucia Chen, Weixi Feng, Do-Hyun Kim, Emma Landry, Filippo Monti, Xiaolong Li, Yisheng Tay, Qi Wang, Junyue Wu, and Qiuyang Yue for the games, trips, conferences, and everything we have had together. Soccer became a big part of my life recently, but it would have been far less enjoyable without the friends I play with, which include many people from above, and also Parsa Jamshidian, Tomoki Okuno, Soumyakanti Pan, Ami Sheth, Muhan Zhang, Rongtian Zhang, and many others.

Finally, I want to thank my parents, whose love has been a source of infinite strength and optimism. I also want to thank my wife, who came into my life in the latter half of the PhD journey. Her love, understanding, and support accompanied me to the finish line.

The results of chapter 1 are published in Statistical Analysis and Data Mining in 2021. The results of chapter 2 are in preparation for publication.

VITA

Education

2012–2014	M.S. Statistics, University of California, San Diego	La Jolla, USA
2008–2012	B.S. Applied Mathematics, Hohai Univeristy	Nanjing, China

Experience

2018–Present	Graduate Student Researcher	
	Department of Biostatistics and Department of Medicine Statistics Core	
	UCLA	Los Angeles, USA
2015–2018	Senior Statistician	
	Department of Medicine Statistics Core	
	UCLA	Los Angeles, USA
2014–2015	Data Scientist	
	Supplyframe Inc.	Pasadena, USA

PUBLICATIONS

Zhou, X., Zhou, J. J., and Zhou, H. (2022). Bag of little bootstraps for massive and distributed longitudinal data. Statistical Analysis and Data Mining: The ASA Data Science Journal, 15(3), 314-321. *Won the 2022 best student paper award of the ASA statistical computing section.*

Zhou, X., Chi, E. C., and Zhou, H. (2022). Proximal MCMC for Bayesian Inference of Constrained and Regularized Estimation. arXiv preprint arXiv:2205.07378.

INTRODUCTION

Many modern data sets are large and high-dimensional. Three prime examples include the Million Veteran Project, All of US program, and the UK Biobank. They occupy terabytes of storage with sample sizes ranging from 10^5 to 10^6. Moreover, there is a rich set of information for each individual in these programs, making them high-dimensional. In the case of the UK Biobank, participants' genomic, imaging, physical activity, and electronic health records (EHR) data are all available for analysis. These data sets present a great opportunity for scientific discovery, but at the same time, their size and dimensionality impose fresh challenges on existing tools for statistical analysis.

The first chapter focuses on the analysis of longitudinal datasets, where linear mixed models are widely used and the inference for variance component parameters relies on the bootstrap method. However, health systems and technology companies routinely generate massive longitudinal datasets that make the traditional bootstrap method infeasible. To solve this problem, we extend the highly scalable bag of little bootstraps method for independent data to longitudinal data and develop a highly efficient Julia package `MixedModelsBLB.jl`. Simulation experiments and real data analysis demonstrate the favorable statistical performance and computational advantages of our method compared to the traditional bootstrap method.

The second chapter focuses on the statistical inference of constrained and regularized estimation problems, which are commonly encountered when analyzing high-dimensional data. The inference for these problems is traditionally considered difficult from both frequentist and Bayesian perspectives. We propose proximal Markov Chain Monte Carlo (ProxMCMC) as a flexible and general Bayesian inference framework to tackle this problem. Originally introduced in the Bayesian imaging literature, ProxMCMC employs the Moreau-Yosida envelope for a smooth approximation of the total-variation regularization term, fixes nuisance and regularization parameters as constants, and relies on the Langevin algorithm for the

posterior sampling. We extend ProxMCMC to the full Bayesian framework with modeling and data-adaptive estimation of all parameters including the regularization strength parameter. More efficient sampling algorithms such as the Hamiltonian Monte Carlo are employed to scale ProxMCMC to high-dimensional problems. Analogous to the proximal algorithms in optimization, ProxMCMC offers a versatile and modularized procedure for the inference of constrained and non-smooth problems. The power of ProxMCMC is illustrated on various statistical estimation and machine learning tasks.

The third chapter focuses on statistical methods that can handle nonignorable missingness due to informatively censored data. We are motivated by the need for variance quantitative trait loci (vQTL) analysis and adjusting for treatment effect in genetic and epidemiological studies. We introduce weighting to estimation equations to improve the estimation efficiency, extend synthetic variables from positive-valued to real-valued, and derive synthetic variables for higher moments to model heterogeneous variances.

CHAPTER 1

Bag of Little Bootstraps for Massive and Distributed Longitudinal Data

This project has been completed and published in Statistical Analysis and Data Mining at https://doi.org/10.1002/sam.11563.

1.1 Introduction

Linear mixed models (LMMs) are powerful tools for analyzing longitudinal data, which are ubiquitous in medical research and E-commerce applications. For example, Electronic Medical Records (EMR) data contains longitudinal measurements from the same patient over time. However, there are two challenges in applying LMMs to today's problems. The first one is the massive sample size of modern datasets. For instance, the UCLA Health System alone has over 2.5 million *annual* patient visits. Analyzing such datasets with LMMs is challenging, especially if the goal is to make statistical inference on the variance component parameters. For example, to test if subjects have different slopes for a covariate, one needs to test whether the corresponding random effect has zero variance. Statistical tests based on asymptotics are dubious because the limiting distribution of random effect parameters is difficult to derive. Therefore, researchers rely on the bootstrap method [Efr79], which eliminates the need for asymptotics, but is computationally intensive. Specifically, running the traditional bootstrap method on LMMs has a computational cost of $O(BNq^3)$, where B is the number of bootstrap replicates, N is the number of subjects, and q is the number

of random effect parameters. When N is on the scale of millions, the bootstrap method is prohibitively slow.

The second challenge relates to distributed datasets. Modern datasets are often stored at multiple locations: internet companies that harvest large volumes of data store them across data centers worldwide to save data transfer costs; medical centers that collaborate in multi-site studies try to avoid sending data over the internet due to security and privacy concerns. However, to fit LMMs and use the traditional bootstrap method, one has to either move the distributed datasets to one place or communicate model parameters and their derivatives continuously between data centers, which incur high data transfer costs.

To overcome these challenges, we extend the Bag of Little Bootstraps (BLB) method [KTS14] to the longitudinal data setting. It has a computational cost of $O(Bbq^3)$ where $b \ll N$, so it is capable of fitting and making statistical inference of LMMs on massive longitudinal datasets using a fraction of the time compared to the traditional bootstrap method. Moreover, by using the BLB framework, our software, `MixedModelsBLB.jl`, provides a solution to the analysis of distributed longitudinal datasets.

1.2 Method

1.2.1 Model and Notation

Given a longitudinal dataset with N independent clusters (the word "cluster" is used interchangeably with "subjects" in this chapter), let $\boldsymbol{y}_i \in \mathbb{R}^{n_i}$ be the observed response vector of length n_i from subject i, and $\boldsymbol{X}_i \in \mathbb{R}^{n_i \times p}$ and $\boldsymbol{Z}_i \in \mathbb{R}^{n_i \times q}$ be the observed covariates for the fixed and random effect parameters, respectively. Consider an LMM of the form

$$\boldsymbol{y}_i = \boldsymbol{X}_i\boldsymbol{\beta} + \boldsymbol{Z}_i\boldsymbol{b}_i + \boldsymbol{\varepsilon}_i, \tag{1.1}$$

where $\boldsymbol{\beta} \in \mathbb{R}^p$ denotes the fixed effect parameters, $\boldsymbol{b}_i \sim N(\boldsymbol{0}, \boldsymbol{\Sigma})$ denotes the random effect for the i-th subject, $\boldsymbol{\Sigma}$ is a $q \times q$ covariance matrix, and $\boldsymbol{\varepsilon}_i \sim N(\boldsymbol{0}, \sigma_0^2 \boldsymbol{I}_{n_i})$ denotes the random

error. $\boldsymbol{b}_i$ and $\boldsymbol{\varepsilon}_i$ are jointly independent. $\boldsymbol{\Sigma}$ and σ_0^2 are the variance component parameters.

1.2.2 Statistical Inference for LMMs

For fixed effect parameters, statistical inference is usually based on the asymptotic distribution of $\widehat{\boldsymbol{\beta}}$. This approach relies on approximations that may not be accurate when the data is unbalanced or when the residuals have non-constant variance [HH14, BMB15]. For distributed datasets, the asymptotic approach is difficult to implement and is potentially costly because it involves transferring parameters and their derivatives between different data centers.

Statistical inference of variance component parameters is more challenging. For testing if a random effect should be included in the model, one needs to test the hypothesis that the corresponding random effect variance equals zero. Since zero lies on the boundary of the parameter space of variance, the usual regularity condition that the parameter should be an interior point of the parameter space is not met. Testing such hypotheses involve using complex asymptotic or exact null distributions [SL87, CR04, Cra08], which makes it cumbersome to use in practice.

Following the notation in [KTS14, VW13], let $\boldsymbol{w}_i = (\boldsymbol{y}_i, \boldsymbol{X}_i, \boldsymbol{Z}_i) \sim P$ be independent and identically distributed (IID) for $i = 1, ..., N$, and let the corresponding empirical distribution be $\mathbb{P}_N = N^{-1} \sum_{i=1}^{N} \delta_{\boldsymbol{w}_i}$. $\theta(P) = (\boldsymbol{\beta}, \boldsymbol{\Sigma}, \sigma_0^2)$ denotes all model parameters, $\hat{\theta}_N = \hat{\theta}_N(\mathbb{P}_N)$ is an estimate of $\theta(P)$. In its essence, statistical inference of $\hat{\theta}_N = \hat{\theta}_N(\mathbb{P}_N)$ is a summary, denoted by $\xi\{Q_N(P)\}$, of the distribution $Q_N(P)$ of $u(\mathbb{P}_N, P)$, which is a function of $\hat{\theta}_N$ and its form depends on our inferential goal. For example, if we want to quantify the variance of $\hat{\theta}_N$, then $u(\mathbb{P}_N, P) = \hat{\theta}_N$ and ξ is the variance. In practice, since P and $Q_N(P)$ are unknown, we cannot calculate $\xi\{Q_N(P)\}$ directly, but we can estimate it using the observed dataset. The asymptotic approach is one way to perform the estimation where we replace $Q_N(P)$ with the asymptotic distribution of $\theta(P)$. An alternative approach is the bootstrap method [Efr79], which replaces $Q_N(P)$ by its bootstrap approximation.

Given IID data $\boldsymbol{w}_1, ..., \boldsymbol{w}_N$ and its empirical distribution $\mathbb{P}_N$, the bootstrap method first samples N data points with replacement from $\mathbb{P}_N$, which has empirical distribution function $\mathbb{P}_N^*$. From the bootstrap sample, $u(\mathbb{P}_N^*, \mathbb{P}_N)$ can be calculated. This process is repeated many times to obtain $\mathbb{Q}_N^*$, which is the empirical distribution of the u's and serves to approximate $Q_N(P)$. Finally, we use $\xi(\mathbb{Q}_N^*)$ as an estimate of $\xi\{Q_N(P)\}$.

However, the bootstrap method is computationally expensive for large datasets, especially for longitudinal data. In addition, it is awkward to apply the bootstrap method to distributed datasets because re-sampling requires access to the full data. To solve these problems, we extend the Bag of Little Bootstraps (BLB) method [KTS14], which was developed for cross-sectional data, to the longitudinal data setting.

Given a longitudinal dataset with N clusters and a subset size $b < N$, the BLB method first samples s subsets, each consisting of b clusters. The sampling is done without replacement and uniformly at random. Let $I_1, ..., I_s \subset \{1, ..., N\}$ denote the clusters that are in each subset, where $|I_j| = b$ for $1 \leq j \leq s$. Further let $\mathbb{P}_{N,b}^{(j)} = b^{-1} \sum_{i \in I_j} \delta_{\boldsymbol{w}_i}$ denote the empirical distribution for subset j. Then, for each subset, it samples N clusters with replacement to obtain the bootstrap sample and calculates $u(\mathbb{P}_{N,b}^*, \mathbb{P}_{N,b}^{(j)})$, where $\mathbb{P}_{N,b}^*$ denotes the empirical distribution of the bootstrap sample. Re-sampling is repeated B times and the empirical distribution of the u-values on subset j is denoted by $\mathbb{Q}_{N,j}^*$. Finally, BLB estimate of $\xi\{Q_N(P)\}$ is given by

$$s^{-1} \sum_{j=1}^{s} \xi(\mathbb{Q}_{N,j}^*),$$

where $\xi(\mathbb{Q}_{N,j}^*)$ serves as an approximation of $\xi\{Q_N(\mathbb{P}_{N,b}^{(j)})\}$.

The fact that BLB operates on subsets rather than the entire dataset confers two advantages. First, it is more amenable to parallel processing than the bootstrap method. Since each subset is much smaller than the full dataset, we can parallelize at the subset level such that multiple CPU cores can work on multiple subsets at the same time. Secondly, to analyze datasets stored at multiple data centers, BLB can treat each data center as a subset or take

further subsets at each data center, perform analysis on each subset, and obtain the final statistical inference by aggregating parameter estimates from different data centers. Since the final parameter estimates are all we need to transfer between data centers, BLB avoids moving raw data over the internet and incurs minimal communication costs. In contrast, the bootstrap method requires that we either move distributed datasets to one place, which poses security and privacy concerns, or communicate large amounts of intermediate parameter estimates and their derivatives, which incurs high communication costs. We note that in order for BLB to work in distributed data settings, one needs to be comfortable with the assumption that subjects from different data centers are IID samples from the population of interest. When certain variables demonstrate spatial heterogeneity, we expect more variability in the corresponding estimates; see Supplementary Materials S3 for a simulation experiment.

Another feature of BLB is the way it generates bootstrap samples. Given a subset with b clusters, it samples N clusters ($N > b$) with replacement to form a bootstrap sample. Doing so offers three advantages. First, it makes BLB automatic in the sense that re-scaling of the resulting estimates is not needed because the u-values are calculated on datasets that are of the same size as the original data. This contrasts to methods such as subsampling [PRW99] and M out of N bootstrap [BGZ97]. Both methods estimate parameters on datasets that are smaller than the original data, and thus require re-scaling the estimates. The second advantage is that storing BLB re-samples requires $O(b)$ rather than $O(N)$ memory because each re-sample has its support on b distinct clusters. In fact, re-sampling N clusters from b clusters amounts to generating a weight vector from an N-trial uniform multinomial distribution over b objects, so each re-sample can be compactly represented by b clusters and a length-b vector denoting the number of repeats of each cluster. The third advantage is that for estimators that can work with a weighted data representation, the computational time using BLB re-samples scales as $O(b)$ rather than $O(N)$. Many commonly used estimators, including Maximum Likelihood Estimators (MLE) and general M-estimators, fall into this

category. This means that we can use either MLE or Generalized Estimating Equations (GEE) to estimate model parameters.

Finally, BLB for longitudinal data enjoys the same consistency and higher-order correctness guarantee as BLB for IID data. Theoretical analysis of BLB is similar to that of bootstrap and follows from standard empirical process results. Using weak convergence of the bootstrapped empirical process [VW13, Theorem 3.6.3], [KTS14] showed that size n resamples from $\mathbb{P}_{N,b}^{(j)}$ behave asymptotically as if they were drawn directly from P. This together with the delta method for bootstrap [Vaa00, Theorem 23.9] yields the consistency of each individual $\xi\{Q_N(\mathbb{P}_{N,b}^{(j)})\}$ as $b, n \to \infty$. Consistency of BLB is then obtained by using the continuous mapping theorem [Vaa00]. This analysis assumes that the sampling units are IID, which is satisfied in the longitudinal setting because our sampling units are clusters and we assume that clusters are IID. Similar arguments can be made for the proof of higher-order correctness.

Consistency and higher-order correctness of BLB for longitudinal data hold for estimators that are Hadamard differentiable. Since M-estimators are generally Hadamard differentiable [VW13, Vaa00] and both MLE and GEE produce M-estimators, these theoretical properties hold with either MLE or GEE.

In the following sections we present results obtained by MLE. GEE results, which are implemented through an approach called WiSER [GSZ21], are presented in Supplementary Materials S4.

1.3 Computational Strategy

A key component of Algorithm 1 is fitting LMMs, and we do so by maximizing the log-likelihood using the Fisher scoring algorithm. For model (1.1), the log-likelihood for the i-th

Input: Clustered data $\boldsymbol{w}_1, ..., \boldsymbol{w}_N$; b: number of clusters in the subset; s: number of subsets; r: number of bootstrap samples within each subset; u: estimate of LMM parameters; ξ: summary of the distribution of u

Output: An estimate of $\xi\{Q_N(P)\}$

for $j \in 1$ *to* s **do**

  Randomly sample a set $I = \{i_1, ..., i_b\}$ of b indices without replacement from $\{1, ..., N\}$

  // Empirical distribution of the j-th subset

  $\mathbb{P}_{N,b}^{(j)} \leftarrow b^{-1} \sum_{i \in I} \delta_{\boldsymbol{w}_i}$

  // Approximate $\xi\{Q_N(\mathbb{P}_{N,b}^{(j)})\}$ by $\xi(\mathbb{Q}_{N,j}^*)$

  for $k \in 1$ *to* r **do**

    Sample $(n_1, ..., n_b) \sim \text{Mult}(N, \mathbf{1_b}/b)$

    // Empirical distribution of the BLB re-sample

    $\mathbb{P}_{N,k}^* \leftarrow N^{-1} \sum_{l=1}^{b} n_l \delta_{\boldsymbol{w}_{i_l}}$

    Fit model using MLE or GEE on the re-sample to get $u_{N,k}^* \leftarrow u(\mathbb{P}_{N,k}^*, \mathbb{P}_{N,b}^{(j)}) = \hat{\theta}_N(\mathbb{P}_{N,k}^*)$.

  end

  // Empirical distribution of the u-values on subset j

  $\mathbb{Q}_{N,j}^* \leftarrow r^{-1} \sum_{k=1}^{r} \delta_{u_{N,k}^*}$

  $\xi_{N,j}^* \leftarrow \xi(\mathbb{Q}_{N,j}^*)$

end

// The BLB estimate of $\xi\{Q_N(P)\}$ averages $\xi_{N,j}^*$ from subsets

Return $s^{-1} \sum_{j=1}^{s} \xi(\mathbb{Q}_{N,j}^*)$

Algorithm 1: BLB for LMM.

cluster is

$$\ell_i = -\frac{n_i}{2}\log(2\pi) - \frac{1}{2}\text{logdet}(\boldsymbol{Z}_i\boldsymbol{\Sigma}\boldsymbol{Z}_i' + \sigma_0^2\boldsymbol{I}_{n_i}) - \frac{1}{2}(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})'(\boldsymbol{Z}_i\boldsymbol{\Sigma}\boldsymbol{Z}_i' + \sigma_0^2\boldsymbol{I}_{n_i})^{-1}(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta}).$$

Identifying a good starting point is crucial for fast convergence. In practice, we initialize $\boldsymbol{\beta}$ and σ_0^2 with least squares solutions

$$\begin{aligned}\boldsymbol{\beta}^{(0)} &= \left(\sum_i \boldsymbol{X}_i^T\boldsymbol{X}_i\right)^{-1}\left(\sum_i \boldsymbol{X}_i^T\boldsymbol{y}_i\right)\\ \sigma_0^2 &= \left(\sum_i \boldsymbol{r}_i^{(0)T}\boldsymbol{r}_i^{(0)}\right) / \left(\sum_i n_i\right),\end{aligned}$$

where $\boldsymbol{r}_i^{(0)} = \boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta}^{(0)}$. To initialize $\boldsymbol{\Sigma}$, we minimize

$$\sum_i \|\boldsymbol{r}_i^{(0)}\boldsymbol{r}_i^{(0)T} - \boldsymbol{Z}_i\boldsymbol{\Sigma}\boldsymbol{Z}_i^T\|_{\text{F}}^2,$$

which gives

$$\text{vec}\,\boldsymbol{\Sigma}^{(0)} = \left(\sum_i \boldsymbol{Z}_i^T\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i^T\boldsymbol{Z}_i\right)^{-1}\left(\sum_i \boldsymbol{Z}_i^T\boldsymbol{r}_i^{(0)} \otimes \boldsymbol{Z}_i^T\boldsymbol{r}_i^{(0)}\right).$$

Besides a good starting point, we also need to evaluate the gradient and the Fisher information matrix efficiently by exploiting structures in these quantities. For example, by using the Woodbury structure in the marginal covariance $\boldsymbol{Z}_i\boldsymbol{\Sigma}\boldsymbol{Z}_i' + \sigma_0^2\boldsymbol{I}_{n_i}$, we can avoid the storage and decomposition of potentially large $n_i \times n_i$ matrices. See Supplementary Materials S2 for detailed derivation and the implementation strategy.

1.4 Software

Our implementation, `MixedModelsBLB.jl`, is an open-source Julia package available at `https://github.com/xinkai-zhou/MixedModelsBLB.jl`. Users can run the software on Julia v1.5 or later, or use Docker without installing Julia. The package is compatible with a wide range of data inputs, including data frames and datasets that are too large to

fit in memory. Furthermore, it works with a variety of nonlinear programming solvers such as Ipopt [WB06], NLopt [Joh20], and KNITRO [BNW06]. Finally, when the user has access to multiple CPU cores, parallel processing can be turned on to gain further efficiency by processing BLB subsets simultaneously.

We illustrate it on the `sleepstudy` example data [BWT03]. The BLB estimates and the confidence intervals are printed. In addition, parameters estimates from all iterations are returned in an object of type `blbEstimates` for further analyses. See `https://github.com/xinkai-zhou/MixedModelsBLB.jl` for detailed documentation.

1.5 Simulation Study

This section presents two simulation experiments. The first one compares the statistical performance between BLB and bootstrap. The second simulation applies BLB to ultra large data sets to demonstrate its scalability.

In the first simulation, we define the relative error of the confidence intervals as $|c-c_0|/c_0$, where c is the estimated confidence interval width and c_0 is the true confidence interval width. We then compare the relative error of the confidence intervals between BLB and the bootstrap method. To calculate c_0, we generate 1000 datasets of size N from the underlying data generating distribution P, compute $\hat{\theta}_N$ on each of them, and use these estimates to calculate confidence intervals and c_0. To calculate c, we simulate one dataset of size N from P, run BLB and bootstrap, and record the parameter estimates as well as the cumulative processing time (after each bootstrap resample or BLB subset has been processed). To reduce the variation in c induced by a particular dataset, we repeat this process on five simulated datasets and average the resulting relative errors and processing times. We present the trajectory of relative error versus time, where the relative error is averaged over variance components parameters. Note that the time axis provides a single-number summary of parameters b (subset size), s (number of subsets), and r (number of bootstrap iterations

Listing 1.1: Illustrating software usage on the `sleepstudy` data.

```
using MixedModelsBLB, JuliaDB, StatsModels, Random
datatable = JuliaDB.loadtable("test/data/sleepstudy.csv")
blb_ests = blb_full_data(
        MersenneTwister(1),
        datatable;
        feformula   = @formula(Reaction ~ 1 + Days),
        reformula   = @formula(Reaction ~ 1),
        id_name     = "id",
        cat_names   = Array{String,1}(),
        subset_size = 10,
        n_subsets   = 20,
        n_boots     = 500,
        solver      = Ipopt.IpoptSolver(print_level=0),
        verbose     = false,
        nonparametric_boot = true
    )
#Bag of Little Boostrap (BLB) for linear mixed models.
#Number of subsets: 20
#Number of grouping factors per subset: 10
#Number of bootstrap samples per subset: 500
#Confidence interval level: 95%

Variance Components parameters

             Estimate  CI Lower  CI Upper

(Intercept)  1202.18    426.24   2087.30
Residual      826.92    513.64   1180.74


Fixed-effect parameters

             Estimate   CI Lower  CI Upper

(Intercept)  250.88     237.66    263.59
Days          10.79       8.11     13.50
```

on a given subset) for BLB, and of r (number of bootstrap iterations) for bootstrap. We used our package `MixedModelsBLB.jl` for BLB and the `MixedModels.jl` package for bootstrap. Parallel processing was turned off for both methods because the primary focus of this experiment is statistical performance.

We generate data under two settings. In the first one, non-intercept entries of $\boldsymbol{X}_i, \boldsymbol{Z}_i$ and $\boldsymbol{\varepsilon}_i$ are drawn independently from the standard normal distribution. In the second one, $\boldsymbol{X}_{i_k,j} \sim \Gamma(1+5(j-1)/(p-1), 2) - 2\Gamma(1+5(j-1)/(p-1), 2)$, $\boldsymbol{Z}_{i_k,j} \sim \Gamma(1+5(j-1)/(q-1), 2) - 2\Gamma(1+5(j-1)/(q-1), 2)$, and $\boldsymbol{\varepsilon}_{i_k} \sim \Gamma(1,2) - 2$ independently for $k = 1, ..., n_i, j = 1, ..., p$. In both settings, $N = 20,000, n_i = 10$ for all i, $p = 100$, and $q = 2$. For BLB, we set the subset size to be $b = N^\gamma$ where $\gamma \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$, and the number of Monte Carlo iterations to be $r = 200$.

Figure 1.1 shows the results. For all subset sizes, BLB converges to low relative error faster than bootstrap. When the subset size is small ($\gamma = 0.5, 0.6, 0.7$), it takes a very short time for BLB to process each subset, and it takes no more than 10-20 subsets for BLB to reach low relative error (each hinge corresponds to a subset for BLB). When the subset size is larger ($\gamma = 0.8, 0.9$), it takes longer to process each subset, but only a small number of subsets (3-5) is needed to achieve low relative error.

Besides the comparison with bootstrap, we also examined the subsampling method [PRW99] as an alternative. However, we observed similar divergence in relative error for smaller subset sizes as reported by [KTS14]. See Supplementary Materials section S5 for more details.

[Insert Figure 1.1 here.]

The second simulation experiment compares the scalability of BLB and bootstrap. Since data generating distributions do not affect scalability, we only consider the standard normal case. We choose $N = 1$ million, $n_i = 20$, $p = 20$, and $q = 2$. The truth is obtained by simulating 200 instead of 1000 datasets due to the bigger sample size. For bootstrap, we set

the number of Monte Carlo iterations $r = 400$. For BLB, we set $b = N^{0.6} \approx 3981, s = 10$, and $r = 200$. For both procedures, we turn on parallel processing. Specifically, BLB uses ten worker nodes and bootstrap uses two threads. We cannot use ten threads for bootstrap because it makes a copy of the model object and the bootstrap sample on each thread, so it would quickly exhaust the memory on our computer (64GB) if we use more than two threads. Figure 1.2 shows the simulation result. We see that BLB finishes all calculations within 170 seconds, which is more than 200 times faster than bootstrap, and achieves lower relative error (0.0534 versus 0.0603, or an 11% reduction). A rough calculation shows that even if our computer has more memory (> 300GB) so that bootstrap can run with ten threads, it would still take two hours and thus be much slower than BLB.

To see how BLB compares with bootstrap on even larger data sets, we simulated a data set with $N = 10$ million, $n_i = 20$, $p = 20$, and $q = 2$ using the same data generating distribution as above. The entire data set contains 200 million records and the CSV file takes 79 GB disk space. For BLB, we set $b = N^{0.6} \approx 15850, s = 10$, and $r = 200$. BLB finishes all computation within 22 minutes. On the other hand, since the data set exceeds our computer's memory limit, we are unable to run bootstrap.

[Insert Figure 1.2 here.]

Besides these two experiments, we also examined the relationship between the number of bootstrap samples on each subset (r) and relative error; see Supplementary Materials S6 for details.

1.6 Real Data

In this section we apply `MixedModelsBLB.jl` to the Action to Control Cardiovascular Risk in Diabetes trial (ACCORD) dataset [ICB10]. The ACCORD study examined whether the intensive therapy that targets normal glycated hemoglobin (HbA1c) levels ($< 6.0\%$) would reduce cardiovascular events when compared to the standard therapy among patients

with type 2 diabetes who had either established cardiovascular disease (CVD) or additional cardiovascular risk factors. A total of 12,251 patients aged 40–79 years participated; their glucose concentrations were measured every four months in the initial year and then annually up to a maximum of 84 months.

After data cleaning, our analytic dataset consists of 67,063 observations on 10,195 individuals. The outcome of interest is fasting plasma glucose, and the covariates include gender, race, baseline age, BMI, visit number, baseline CVD history, adjusted insulin, and the type of therapy they received. We follow [SRR15] and use insulin units per body weight in kg (adjusted insulin) instead of raw total insulin units. In addition to random intercept, we also included a random slope for the visit number. Since the ground truth is not available for real data, we cannot compare methods using relative error. Instead, we present the 95% confidence intervals given by BLB, bootstrap, and the Wald method. Note that the Wald method can only produce confidence intervals for fixed effect parameters. For this analysis, we used a subset size of 1600 individuals ($\gamma = 0.8$) and ran BLB on 30 subsets, each with 200 bootstrap samples. The subset size was chosen so that we would not get too few observations for certain categories in the unevenly distributed race variable. A sensitivity analysis of other subset sizes is given in Supplementary Materials section S7. For bootstrap, we ran it with 2000 bootstrap samples. Both methods used parallel processing. Table 1.1 shows the results. We find the visit number, BMI, baseline age, race, adjusted insulin, and certain oral medication classes to be significantly associated with fasting plasma glucose. We also find the random slope for visit number to be significant and should be included in the model. Finally, we note that BLB achieves similar inference compared to bootstrap, but uses much less time.

[Insert Table 1.1 here.]

1.7 Conclusion and Future Work

We have developed an algorithm based on the BLB method for the statistical inference of fixed effect and variance component parameters of linear mixed models on large and distributed longitudinal datasets; we also developed a Julia software package `MixedModelsBLB.jl` for this purpose. Unlike the bootstrap method, which typically requires $O(BNq^3)$ computational cost, our method only costs $O(Bbq^3)$, where b is much smaller than N. The simulation and real data results demonstrate the efficiency and statistical performance of our method.

Code availability

The software package is publicly available at `https://github.com/xinkai-zhou/MixedModelsBLB.jl`.

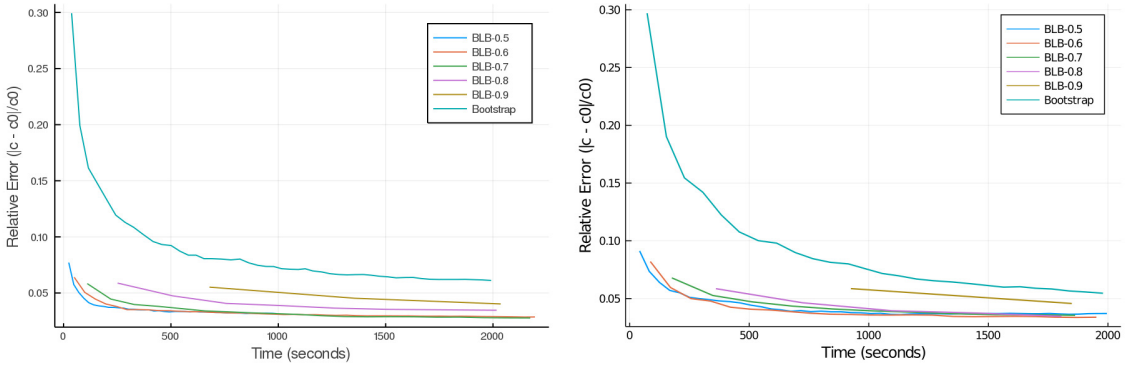


Figure 1.1: Relative error versus processing time for BLB and bootstrap under Normal (left) and Gamma (right) data generating distributions.

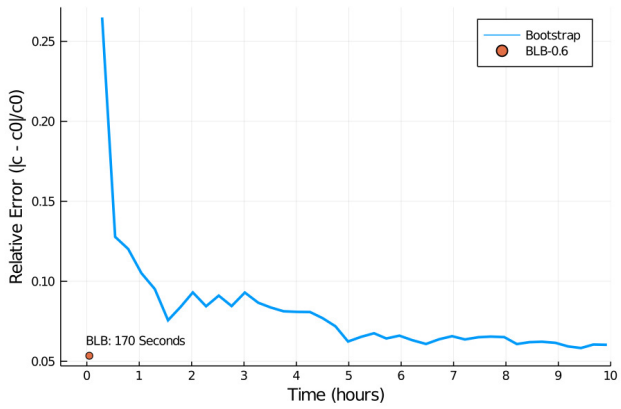


Figure 1.2: Relative error versus processing time on $N = 1$ million subjects and 20 million total observations. BLB subset size was set to $b = N^{0.6} \approx 3981$.

1.8 Supplementary Material

1.8.1 Notation

Our notation for multivariate calculus follows that of a standard text such as [MN99].

We use vec$\boldsymbol{A}$ to denote the vector that stacks the columns of a matrix $\boldsymbol{A}$, and use vech$\boldsymbol{A}$ to denote the vector that only stacks the columns of the lower triangular part of a square matrix $\boldsymbol{A}$. The diag operator has two meanings. For a vector $\boldsymbol{v}$, diag($\boldsymbol{v}$) represents the diagonal matrix with $\boldsymbol{v}$ on the diagonal. For a square matrix $\boldsymbol{M}$, diag($\boldsymbol{M}$) represents the vector of diagonal elements of $\boldsymbol{M}$.

The $mn \times mn$ *commutation matrix* is denoted by $\boldsymbol{K}_{mn}$ and satisfies $\boldsymbol{K}_{mn} \cdot \text{vec}\boldsymbol{A} = \text{vec}\boldsymbol{A}^T$ for an arbitrary $m \times n$ matrix $\boldsymbol{A}$. The $n^2 \times n(n+1)/2$ *duplication matrix* is denoted by $\boldsymbol{D}_n$ and satisfies $\boldsymbol{D}_n \cdot \text{vech}\boldsymbol{A} = \text{vec}\boldsymbol{A}$ for any $n \times n$ symmetric matrix $\boldsymbol{A}$. The $n^2 \times n(n+1)/2$ *copying matrix* is denoted by $\boldsymbol{C}_n$ and satisfies $\boldsymbol{C}_n \cdot \text{vech}\boldsymbol{A} = \text{vec}\boldsymbol{A}$ for any $n \times n$ lower triangular matrix $\boldsymbol{A}$. The Kronecker product of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ of arbitrary shape is denoted by $\boldsymbol{A} \otimes \boldsymbol{B}$.

The Jacobian matrix of a differentiable matrix function $f : \mathbb{R}^{n \times q} \mapsto \mathbb{R}^{m \times p}$ is defined as the $mp \times nq$ matrix

$$\mathrm{D}\, f(\boldsymbol{X}) = \frac{\partial \text{vec} f(\boldsymbol{X})}{\partial (\text{vec}\boldsymbol{X})^T}.$$

This definition includes scalar functions ($m = p = 1$) and vector functions ($p = 1$) as special cases. Given a function composition $h(\boldsymbol{X}) = g(f(\boldsymbol{X}))$, the *chain rule* for the Jacobian matrix

$$\mathrm{D}\, h(\boldsymbol{X}) = \mathrm{D}\, g(\boldsymbol{Y}) \cdot \mathrm{D}\, f(\boldsymbol{X}),$$

where $\boldsymbol{Y} = f(\boldsymbol{X})$.

1.8.2 Gradient, Hessian, expected Hessian, and computational details

We derive the gradient, Hessian, and expected Hessian of the LMM log-likelihood

$$\ell_i = -\frac{n_i}{2}\log(2\pi) - \frac{1}{2}\text{logdet}(\boldsymbol{Z}_i\boldsymbol{\Sigma}\boldsymbol{Z}_i' + \sigma_0^2\boldsymbol{I}_{n_i}) - \frac{1}{2}(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})'(\boldsymbol{Z}_i\boldsymbol{\Sigma}\boldsymbol{Z}_i' + \sigma_0^2\boldsymbol{I}_{n_i})^{-1}(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta}),$$

and describe how to evaluate them efficiently in $O(q^3)$ flops for each i. Due to the positive semidefiniteness constraint on $\boldsymbol{\Sigma}$, we parameterize it in terms of the Cholesky factor $\boldsymbol{\Sigma} = \boldsymbol{L}\boldsymbol{L}^T$. For conciseness, we use the following notation throughout the derivation:

$$\begin{aligned}\boldsymbol{r}_i &= \boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta} \\ \boldsymbol{\Omega}_i &= \boldsymbol{Z}_i\boldsymbol{\Sigma}\boldsymbol{Z}_i' + \sigma_0^2\boldsymbol{I}_{n_i},\end{aligned}$$

1.8.2.1 Gradient

1. Gradient of $\boldsymbol{\beta}$.

$$\begin{aligned}D_{\boldsymbol{\beta}}\ell_i &= -\frac{1}{2}D_{\boldsymbol{\beta}}(\boldsymbol{y}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{y}_i - 2\boldsymbol{y}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{X}_i\boldsymbol{\beta} + \boldsymbol{\beta}'\boldsymbol{X}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{X}_i\boldsymbol{\beta}) \\ &= -\frac{1}{2}(\text{vec}(-2\boldsymbol{X}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{y}_i + 2\boldsymbol{X}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{X}_i\boldsymbol{\beta}))' \\ &= (\text{vec}\boldsymbol{X}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i)'.\end{aligned} \tag{1.2}$$

Thus $\nabla_{\boldsymbol{\beta}}\ell_i = \boldsymbol{X}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i$.

2. Gradient of σ_0^2.

$$\begin{aligned}
D_{\sigma_0^2}\text{logdet}(\boldsymbol{\Omega}_i) &= D_{\boldsymbol{\Omega}_i}\text{logdet}(\boldsymbol{\Omega}_i)D_{\sigma_0^2}\boldsymbol{\Omega}_i \\
&= (\text{vec}\boldsymbol{\Omega}_i^{-1})'\text{vec}(\boldsymbol{I}) \\
&= \text{tr}(\boldsymbol{\Omega}_i^{-1}), \\
D_{\sigma_0^2}\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i &= D_{\sigma_0^2}\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i \\
&= D_{\sigma_0^2}\text{tr}(\boldsymbol{r}_i\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}) \\
&= D_{\boldsymbol{\Omega}_i}\text{tr}(\boldsymbol{r}_i\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1})D_{\sigma_0^2}\boldsymbol{\Omega}_i \\
&= (-\text{vec}(\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}))'\text{vec}(\boldsymbol{I}) \\
&= -\text{tr}(\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}) \\
&= -\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-2}\boldsymbol{r}_i.
\end{aligned} \tag{1.3}$$

Thus

$$\nabla_{\sigma_0^2}\ell_i = -\frac{1}{2}\text{tr}(\boldsymbol{\Omega}_i^{-1}) + \frac{1}{2}\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-2}\boldsymbol{r}_i.$$

3. Gradient of $\boldsymbol{L}$. By the chain rule,

$$\begin{aligned}
D_{\boldsymbol{L}}\text{logdet}(\boldsymbol{\Omega}_i) &= D_{\boldsymbol{\Omega}_i}\text{logdet}(\boldsymbol{\Omega}_i)\cdot D_{\boldsymbol{\Sigma}}\boldsymbol{\Omega}_i\cdot D_{\boldsymbol{L}}\boldsymbol{\Sigma} \\
D_{\boldsymbol{L}}\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i &= D_{\boldsymbol{\Omega}_i}\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i\cdot D_{\boldsymbol{\Sigma}}\boldsymbol{\Omega}_i\cdot D_{\boldsymbol{L}}\boldsymbol{\Sigma},
\end{aligned} \tag{1.4}$$

where

$$\begin{aligned}
D_{\boldsymbol{\Omega}_i}\text{logdet}(\boldsymbol{\Omega}_i) &= (\text{vec}(\boldsymbol{\Omega}_i^{-1}))' \\
D_{\boldsymbol{\Omega}_i}\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i &= D_{\boldsymbol{\Omega}_i}\text{tr}(\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}') \\
&= -(\text{vec}(\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}))' \\
&= -\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\otimes\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1} \\
D_{\boldsymbol{\Sigma}}\boldsymbol{\Omega}_i &= \boldsymbol{Z}_i\otimes\boldsymbol{Z}_i \\
D_{\boldsymbol{L}}\boldsymbol{\Sigma} &= (\boldsymbol{I}_{q^2}+\boldsymbol{K}_{qq})(\boldsymbol{L}\otimes\boldsymbol{I}_q).
\end{aligned}$$

Plugging these into (1.4), we get

$$
\begin{aligned}
& D_{\boldsymbol{L}} \operatorname{logdet}(\boldsymbol{\Omega}_i) \\
= \; & (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)((\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq})(\boldsymbol{L} \otimes \boldsymbol{I}_q)) \\
= \; & (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)(\boldsymbol{L} \otimes \boldsymbol{I}_q) + (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)K_{qq}(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
= \; & (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i) + (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'K_{n_i n_i}(\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i) \\
= \; & (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i) + (K_{n_i n_i}\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i) \\
= \; & (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i) + (\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i) \\
= \; & 2(\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))'(\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i) \\
= \; & 2((\boldsymbol{L}'\boldsymbol{Z}_i' \otimes \boldsymbol{Z}_i')\operatorname{vec}(\boldsymbol{\Omega}_i^{-1}))' \\
= \; & 2(\operatorname{vec}(\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}))' \\
& D_{\boldsymbol{L}}\boldsymbol{r}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}_i \\
= \; & -(\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{r}'\boldsymbol{\Omega}_i^{-1})(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)(\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq})(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
= \; & -(\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{r}'\boldsymbol{\Omega}_i^{-1})(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)(\boldsymbol{L} \otimes \boldsymbol{I}_q) - (\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{r}'\boldsymbol{\Omega}_i^{-1})(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)\boldsymbol{K}_{qq}(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
= \; & -(\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) - \boldsymbol{K}_{11}(\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
= \; & -2(\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i).
\end{aligned}
$$

Thus

$$\nabla_{\operatorname{vech}\boldsymbol{L}}\ell_i = (\operatorname{vech}(-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} + \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}))'.$$

1.8.2.2 Hessian

1. The $(\boldsymbol{\beta}, \boldsymbol{\beta})$ block.

$$
\begin{aligned}
H_{\boldsymbol{\beta},\boldsymbol{\beta}}\ell_i &= -\boldsymbol{X}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{X}_i \\
\mathbb{E}(H_{\boldsymbol{\beta},\boldsymbol{\beta}}\ell_i) &= -\boldsymbol{X}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{X}_i.
\end{aligned}
\tag{1.5}
$$

2. The (σ_0^2, σ_0^2)block.

$$
\begin{aligned}
D_{\sigma_0^2} D_{\sigma_0^2} \operatorname{logdet}(\boldsymbol{\Omega}_i) &= D_{\sigma_0^2} \operatorname{tr}(\boldsymbol{\Omega}_i^{-1}) \\
&= D_{\boldsymbol{\Omega}_i} \operatorname{tr}(\boldsymbol{\Omega}_i^{-1}) D_{\sigma_0^2} \boldsymbol{\Omega}_i \\
&= -(\operatorname{vec}(\boldsymbol{\Omega}_i^{-2}))' \operatorname{vec}(\boldsymbol{I}) \\
&= -\operatorname{tr}(\boldsymbol{\Omega}_i^{-2}) \\
D_{\sigma_0^2} D_{\sigma_0^2} \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i &= D_{\sigma_0^2} (-\boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i) \\
&= -D_{\boldsymbol{\Omega}_i^{-1}} \operatorname{tr}(\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-2}) D_{\boldsymbol{\Omega}_i} \boldsymbol{\Omega}_i^{-1} D_{\sigma_0^2} \boldsymbol{\Omega}_i \\
&= -(\operatorname{vec}(2\boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \boldsymbol{r}_i'))' (-\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1}) \operatorname{vec}(\boldsymbol{I}) \\
&= 2((\boldsymbol{r}_i \boldsymbol{r}_i' \otimes \boldsymbol{\Omega}_i^{-1}) \operatorname{vec}(\boldsymbol{I}))' (\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1}) \operatorname{vec}(\boldsymbol{I}) \\
&= 2(\operatorname{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \otimes \boldsymbol{\Omega}_i^{-1}) (\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1}) \operatorname{vec}(\boldsymbol{I}) \\
&= 2(\operatorname{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-2}) \operatorname{vec}(\boldsymbol{I}) \\
&= 2(\operatorname{vec}(\boldsymbol{I}))' \operatorname{vec}(\boldsymbol{\Omega}_i^{-3} \boldsymbol{r}_i \boldsymbol{r}_i') \\
&= 2\operatorname{tr}(\boldsymbol{\Omega}_i^{-3} \boldsymbol{r}_i \boldsymbol{r}_i') \\
&= 2\boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-3} \boldsymbol{r}_i.
\end{aligned} \tag{1.6}
$$

Thus

$$
\begin{aligned}
H_{\sigma_0^2, \sigma_0^2} \ell_i &= -\frac{1}{2}(-\operatorname{tr}(\boldsymbol{\Omega}_i^{-2})) - \frac{1}{2}(2\boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-3} \boldsymbol{r}_i) \\
&= \frac{1}{2}\operatorname{tr}(\boldsymbol{\Omega}_i^{-2}) - \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-3} \boldsymbol{r}_i \\
\mathbb{E}(H_{\sigma_0^2, \sigma_0^2} \ell_i) &= \frac{1}{2}\operatorname{tr}(\boldsymbol{\Omega}_i^{-2}) - \mathbb{E}\operatorname{tr}(\boldsymbol{\Omega}_i^{-3} \boldsymbol{r}_i \boldsymbol{r}_i') \\
&= \frac{1}{2}\operatorname{tr}(\boldsymbol{\Omega}_i^{-2}) - \operatorname{tr}(\boldsymbol{\Omega}_i^{-2}) \\
&= -\frac{1}{2}\operatorname{tr}(\boldsymbol{\Omega}_i^{-2}).
\end{aligned} \tag{1.7}
$$

3. The $(\operatorname{vech}\boldsymbol{L}, \operatorname{vech}\boldsymbol{L})$ block. Let $\nabla_{\boldsymbol{L}} \ell_i = (\operatorname{vec}(-u + v))'$, where

$$
\begin{aligned}
u &= \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L}, \\
v &= \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{r} \boldsymbol{r}' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L}.
\end{aligned}
$$

Since

$$\begin{aligned}
D_{\boldsymbol{L}}\boldsymbol{\Omega}_i^{-1} &= D_{\boldsymbol{\Omega}_i}\boldsymbol{\Omega}_i^{-1} \cdot D_{\boldsymbol{\Sigma}}\boldsymbol{\Omega}_i \cdot D_{\boldsymbol{L}}\boldsymbol{\Sigma} \\
&= (-\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1})(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)(\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq})(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
&= (-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)(\boldsymbol{L} \otimes \boldsymbol{I}_q) + (-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\boldsymbol{K}_{qq}(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
&= (-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + \boldsymbol{K}_{nn}(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
&= (\boldsymbol{I}_{n^2} + \boldsymbol{K}_{nn})(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i),
\end{aligned}$$

we have

$$\begin{aligned}
& D_{\boldsymbol{L}}u \\
= \;& (\boldsymbol{L}'\boldsymbol{Z}_i' \otimes \boldsymbol{Z}_i')(\boldsymbol{I}_{n^2} + \boldsymbol{K}_{nn})(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + (\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
= \;& (\boldsymbol{L}'\boldsymbol{Z}_i' \otimes \boldsymbol{Z}_i' + (\boldsymbol{L}'\boldsymbol{Z}_i' \otimes \boldsymbol{Z}_i')\boldsymbol{K}_{nn})(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + (\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
= \;& (-\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + (\boldsymbol{L}'\boldsymbol{Z}_i' \otimes \boldsymbol{Z}_i')(\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes -\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L})\boldsymbol{K}_{qq} \\
& +(\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
= \;& (-\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + (\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes (-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}))\boldsymbol{K}_{qq} \\
& +(\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i).
\end{aligned}$$

Furthermore,

$$
\begin{aligned}
&D_{\boldsymbol{L}}(\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\\
&= D_{\boldsymbol{\Omega}_i^{-1}}(\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)D_{\boldsymbol{L}}\boldsymbol{\Omega}_i^{-1}\\
&= ((\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i')D_{\boldsymbol{\Omega}_i^{-1}}\boldsymbol{\Omega}_i^{-1} + (\boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}')D_{\boldsymbol{\Omega}_i^{-1}}\boldsymbol{\Omega}_i^{-1})D_{\boldsymbol{L}}\boldsymbol{\Omega}_i^{-1}\\
&= ((\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i') + (\boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'))D_{\boldsymbol{L}}\boldsymbol{\Omega}_i^{-1}\\
&= ((\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i') + (\boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'))(\boldsymbol{I}_{n^2}+\boldsymbol{K}_{nn})(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\\
&= (\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i' + (\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i')\boldsymbol{K}_{nn}+\\
&\quad \boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}' + (\boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}')\boldsymbol{K}_{nn})(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\\
&= (\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i' + \boldsymbol{K}_{qq}(\boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}')+\\
&\quad \boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}' + \boldsymbol{K}_{qq}(\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i'))(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\\
&= (\boldsymbol{I}_{q^2}+\boldsymbol{K}_{qq})(\boldsymbol{Z}_i'\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}' + \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\otimes\boldsymbol{Z}_i')(-\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\\
&= (\boldsymbol{I}_{q^2}+\boldsymbol{K}_{qq})(-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i - \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i),
\end{aligned}
$$

so

$$
\begin{aligned}
&D_{\boldsymbol{L}}v\\
=\;& (\boldsymbol{L}'\otimes\boldsymbol{I}_q)D_{\boldsymbol{L}}(\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + (\boldsymbol{I}_q\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)D_{\boldsymbol{L}}\boldsymbol{L}\\
=\;& (\boldsymbol{L}'\otimes\boldsymbol{I}_q)(\boldsymbol{I}_{q^2}+\boldsymbol{K}_{qq})(-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i -\\
&\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + (\boldsymbol{I}_q\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\\
=\;& (\boldsymbol{L}'\otimes\boldsymbol{I}_q + \boldsymbol{K}_{qq}(\boldsymbol{I}_q\otimes\boldsymbol{L}'))(-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i -\\
&\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + (\boldsymbol{I}_q\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)\\
=\;& (-\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i - \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) +\\
&\boldsymbol{K}_{qq}(-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i - \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}\otimes\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) +\\
&(\boldsymbol{I}_q\otimes\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i).
\end{aligned}
$$

Thus

$$
\begin{aligned}
& H_{\boldsymbol{L},\boldsymbol{L}}\ell_i \\
= \; & -((-\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + \\
& (\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes (-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}))\boldsymbol{K}_{qq} + (\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)) + \\
& (-\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i - \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + \\
& \boldsymbol{K}_{qq}(-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i - \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + \\
& (\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{r}\boldsymbol{r}'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i). \\
& \mathbb{E}(H_{\boldsymbol{L},\boldsymbol{L}}\ell_i) \\
= \; & -(-\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
& +(\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes (-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L}))\boldsymbol{K}_{qq} + (\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + \\
& (-\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i - \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + \\
& \boldsymbol{K}_{qq}(-\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i - \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) + \\
& (\boldsymbol{I}_q \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
= \; & \boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i + (\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L})\boldsymbol{K}_{qq}.
\end{aligned}
$$

Then

$$\mathbb{E}(H_{\text{vech}\boldsymbol{L},\text{vech}\boldsymbol{L}}\ell_i) = \boldsymbol{C}'\,\mathbb{E}(H_{\boldsymbol{L},\boldsymbol{L}}\ell_i)\boldsymbol{C},$$

where $\boldsymbol{C}$ is the matrix such that $\text{vec}\boldsymbol{L} = \boldsymbol{C} \cdot \text{vech}\boldsymbol{L}$, i.e., it's the duplication matrix where the rows that correspond to the upper triangular elements of $\boldsymbol{L}$ are replaced by $\boldsymbol{0}$.

4. The $(\boldsymbol{\beta}, \sigma_0^2)$ block.

$$\mathbb{E}(H_{\beta,\sigma_0^2}\ell_i) = \mathbb{E}((-\boldsymbol{r}_i'\boldsymbol{\Omega}_i \otimes \boldsymbol{X}_i'\boldsymbol{\Omega}_i)D_{\sigma_0^2}\boldsymbol{\Omega}_i) = \boldsymbol{0}.$$

5. The $(\boldsymbol{\beta}, \text{vech}\boldsymbol{L})$ block. Since $D_{\boldsymbol{L}}\boldsymbol{\Omega}_i^{-1}$ does not involve $\boldsymbol{r}$,

$$\mathbb{E}(H_{\beta,\text{vech}\boldsymbol{L}}\ell_i) = \mathbb{E}(-(\boldsymbol{r}' \otimes \boldsymbol{X}_i')D_{\boldsymbol{L}}\boldsymbol{\Omega}_i^{-1}D_{\text{vech}\boldsymbol{L}}\boldsymbol{L}) = \boldsymbol{0}.$$

6. The $(\sigma_0^2, \text{vech}\boldsymbol{L})$ block.

$$
\begin{aligned}
D_{\boldsymbol{L}} D_{\sigma_0^2} \text{logdet}(\boldsymbol{\Omega}_i) &= D_{\boldsymbol{L}} \text{tr}(\boldsymbol{\Omega}_i^{-1}) \\
&= D_{\boldsymbol{\Omega}_i} \text{tr}(\boldsymbol{\Omega}_i^{-1}) D_{\boldsymbol{L}} \boldsymbol{\Omega}_i \\
&= (-\text{vec}(\boldsymbol{\Omega}_i^{-2}))'(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)(\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq})(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
&= -\text{vec}(\boldsymbol{I})'(\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1})(\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i)(\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq})(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
&= -\text{vec}(\boldsymbol{I})'(\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)(\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq})(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
&= -\text{vec}(\boldsymbol{I})'(\boldsymbol{I}_{n^2} + \boldsymbol{K}_{nn})(\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i)(\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
&= -2\text{vec}(\boldsymbol{I})'(\boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i\boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-1}\boldsymbol{Z}_i) \\
&= -2((\boldsymbol{L}'\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-1})\text{vec}(\boldsymbol{I}))' \\
&= -2(\text{vec}(\boldsymbol{Z}_i'\boldsymbol{\Omega}_i^{-2}\boldsymbol{Z}_i\boldsymbol{L}))'
\end{aligned}
$$

$$
\begin{aligned}
& D_{\boldsymbol{L}} D_{\sigma_0^2} \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \\
= & -D_{\boldsymbol{L}} \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i \\
= & -D_{\boldsymbol{\Omega}_i^{-1}} \mathrm{tr}(\boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1}) D_{\boldsymbol{\Omega}_i} \boldsymbol{\Omega}_i^{-1} D_{\boldsymbol{L}} \boldsymbol{\Omega}_i \\
= & -(\mathrm{vec}(2\boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \boldsymbol{r}_i'))' (-\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1}) D_{\boldsymbol{L}} \boldsymbol{\Omega}_i \\
= & 2((\boldsymbol{r}_i \boldsymbol{r}_i' \otimes \boldsymbol{\Omega}_i^{-1}) \mathrm{vec}(\boldsymbol{I}))' (\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1}) D_{\boldsymbol{L}} \boldsymbol{\Omega}_i \\
= & 2(\mathrm{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \otimes \boldsymbol{\Omega}_i^{-1}) (\boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-1}) D_{\boldsymbol{L}} \boldsymbol{\Omega}_i \\
= & 2(\mathrm{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-2}) D_{\boldsymbol{L}} \boldsymbol{\Omega}_i \\
= & 2(\mathrm{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \otimes \boldsymbol{\Omega}_i^{-2}) (\boldsymbol{Z}_i \otimes \boldsymbol{Z}_i) (\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq}) (\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
= & 2(\mathrm{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \otimes \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i) (\boldsymbol{I}_{q^2} + \boldsymbol{K}_{qq}) (\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
= & 2(\mathrm{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \otimes \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i + \boldsymbol{K}_{nn} (\boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \otimes \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i)) (\boldsymbol{L} \otimes \boldsymbol{I}_q) \\
= & 2(\mathrm{vec}(\boldsymbol{I}))' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i + \boldsymbol{K}_{nn} (\boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L} \otimes \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i)) \\
= & 2((\mathrm{vec}\boldsymbol{I})' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L} \otimes \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i) + (\mathrm{vec}\boldsymbol{I})' (\boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \otimes \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}) \boldsymbol{K}_{qq}) \\
= & 2((\mathrm{vec}(\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L}))' + (\mathrm{vec}(\boldsymbol{L}' \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i))' \boldsymbol{K}_{qq}) \\
= & 2(\mathrm{vec}(\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L} + \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}))'.
\end{aligned}
$$

Together we have

$$
\begin{aligned}
& D_{\boldsymbol{L}} D_{\sigma_0^2} \ell_i \\
= & -\frac{1}{2}(-2(\mathrm{vec}(\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}))' \\
& +2(\mathrm{vec}(\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L} + \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}))') \\
= & (\mathrm{vec}(\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L} - \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L} - \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}))',
\end{aligned}
$$

so

$$
\begin{aligned}
H_{\sigma_0^2, \mathrm{vech}\boldsymbol{L}} \ell_i &= (\mathrm{vec}(\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L} - \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{Z}_i \boldsymbol{L} - \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-1} \boldsymbol{r}_i \boldsymbol{r}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}))' \boldsymbol{C} \\
\mathbb{E}(-H_{\sigma_0^2, \mathrm{vech}\boldsymbol{L}} \ell_i) &= (\mathrm{vec}(-\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L} + \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L} + \boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}))' \boldsymbol{C} \\
&= (\mathrm{vec}(\boldsymbol{Z}_i' \boldsymbol{\Omega}_i^{-2} \boldsymbol{Z}_i \boldsymbol{L}))' \boldsymbol{C}.
\end{aligned}
$$

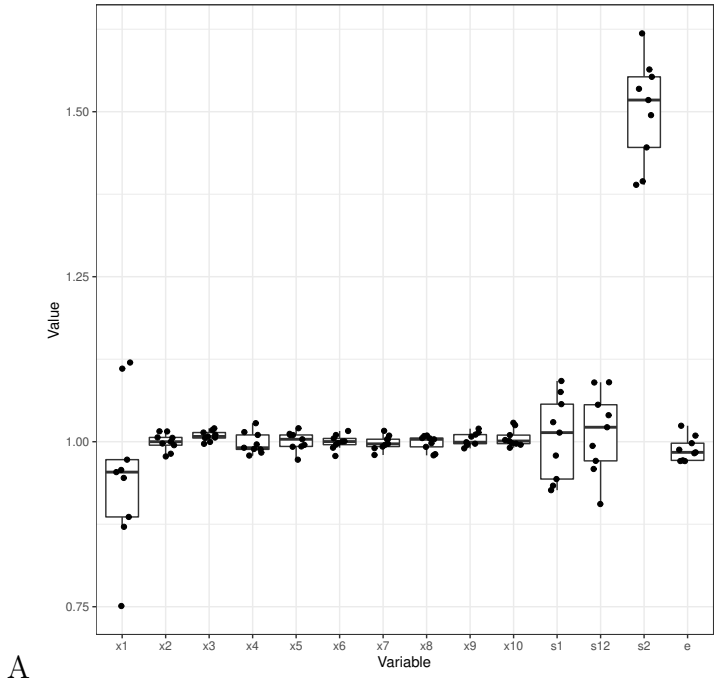
1.8.3 When data centers exhibit spatial heterogeneity

In this section, we examine the effect of spatial correlation on the resulting estimates. Specifically, we generate data sets for 9 data centers that lie on a 3-by-3 grid ($\{(0, 10, 20) \times (0, 10, 20)\}$) and adopt an exponential spatial covariance structure: $\boldsymbol{\Omega}_{kl} = 0.07 \cdot \exp(-0.1 \cdot d_{kl})$, where d_{kl} is the euclidean distance between data centers k and l. The parameters were chosen so that the spatial correlation is roughly 10% of the intra-class correlation. Entries of $\boldsymbol{\Omega}$ ranges between 0.004 to 0.07. Then we generate spatial random effects α_k, $k \in \{1, ..., 9\}$, from the $N(\mathbf{0}, \boldsymbol{\Omega})$ distribution and use them for data generation. The response for subject i from center k thus becomes

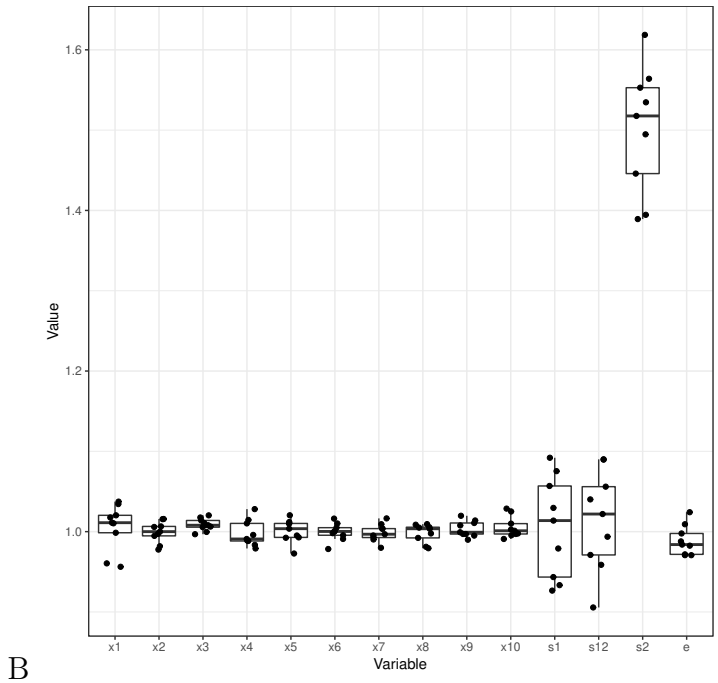
$$\boldsymbol{y}_{ik} = \boldsymbol{X}_{ik}\boldsymbol{\beta} + \boldsymbol{Z}_{ik}\boldsymbol{b}_{ik} + \alpha_k \mathbf{1} + \boldsymbol{\varepsilon}_{ik}. \tag{1.8}$$

The added spatial random effect α_k, which is shared by all subjects from center k, induces correlation among subjects from the same center and also spatial correlation between centers.

We test the BLB procedure, assuming no spatial correlation, on simulated data sets and the result is in Fig. 1.3. There are 10 fixed effects (including the intercept), a random intercept and a random slope in the model. The truth is 1 for all parameters except that the variance of the random slope is 1.5 (for making the random effect covariance matrix positive definite). Each dot in the boxplot represents an estimate from a data center. Compared to the case without spatial random effect, the one with it has more dispersed fixed intercept estimates. This is not surprising because the random spatial effect is absorbed into the fixed intercept and we expect to see wider bars for effects that have spatial heterogeneity. Estimates for all other parameters look quite similar between the two scenarios, suggesting that our method is robust when the data is generated with mild spatial correlation.



A



B

Figure 1.3: Boxplots of estimates for fixed and random effects from nine data centers. A: data was generated with spatial random effect. B: no spatial random effect. $x1, ..., x10$: fixed effects; $s1, s12, s2$: random effect covariance; e: error variance.

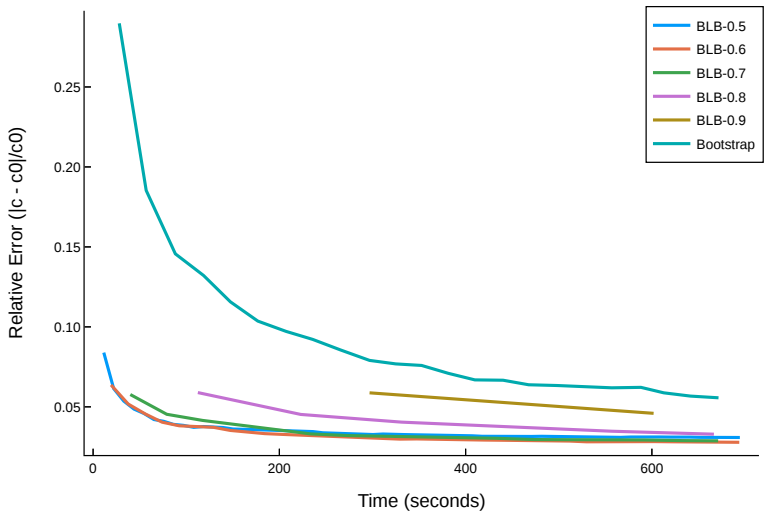


Figure 1.4: Relative error versus processing time for BLB (using GEE rather than maximum likelihood) and bootstrap under a Normal data generating distribution.

1.8.4 Parameter estimation using GEE

As mentioned in Sec 2.2 of the main text, we can use either MLE or GEE to estimate model parameters. For GEE, we incorporate it through an approach called WiSER [GSZ21] that was developed for within-subject variance estimation for linear mixed models and was shown to be equivalent to a specific quadratic GEE with a working covariance structure assuming marginal normality of the response. We repeated our first simulation experiment using this approach and present the results in Fig 1.4.

1.8.5 The performance of subsampling

As mentioned in the main article, we also examined subsampling [PRW99] as an alternative method for performing inference on large longitudinal data sets. The main issue with subsampling is that it may fail to converge for some subset sizes. This phenomenon was observed by [KTS14] on cross sectional datasets, and we observe the same thing for longitudinal data

through our simulation experiments. We run subsampling on the same data sets as we used for the first simulation experiment, and calculate relative error in the same way as before. Figure 1.5 shows the results for the gamma data generating distribution. We can see that subsampling fails to converge when subset size equals $20000^{0.5} \approx 141$, $20000^{0.8} \approx 2759$, or $20000^{0.9} \approx 7429$.

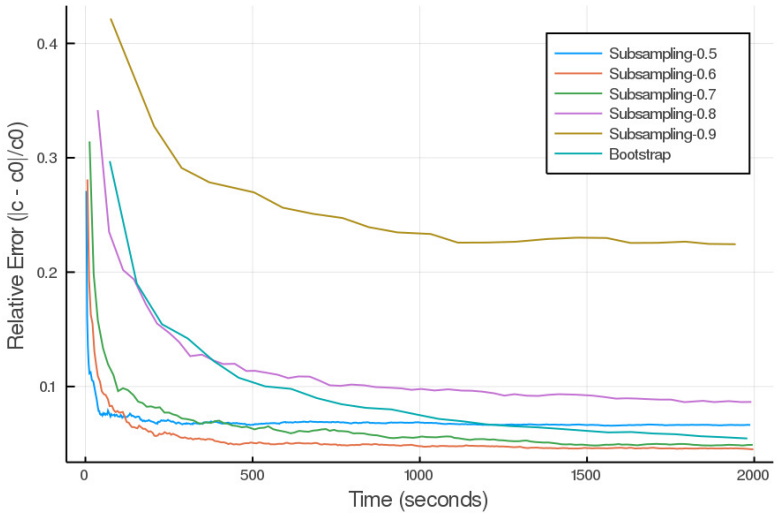


Figure 1.5: Relative error versus processing time for subsampling and bootstrap.

1.8.6 The relationship between r and relative error

We use the simulated data from simulation 1 (the Gaussian case). Instead of changing the subset size, we fix it at $20,000^{0.6} \approx 380$ and vary the number of bootstrap iterations on each subset (r). We tried $r = (100, 200, 300, 400, 500)$ and the relative error is presented in Fig 1.6. We can see that setting r to 100 is too small because the relative error does not go down fast enough compared to other r values. Setting r to 200-500 makes little difference after 500 seconds, but $r = 200$ achieves small relative error in less time and thus we recommend it to users.

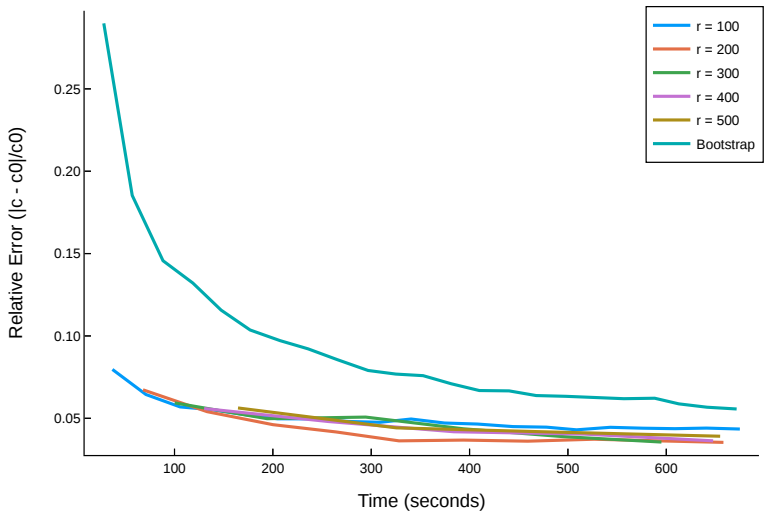


Figure 1.6: Relative error versus processing time for BLB at different r (number of bootstrap iterations on each subset).

1.8.7 Sensitivity analysis of the ACCORD data

As a sensitivity analysis, we compare BLB results at subset sizes 1000, 1600, and 2000 of the ACCORD data. The results are mostly consistent; but we do note the difference in the race variable (Hispanic and Other) at subset size 1000. This is mainly because the race variable is unevenly distributed and has less observations in the Hispanic and Other category so that at smaller subset sizes, some subsets may risk sampling too few observations in those categories to produce stable estimates. See Table 1.2 for a summary of the race variable.

Table 1.1: 95% Confidence Intervals for the ACCORD data using a LMM that includes a random intercept, a random slope, and a covariance term between the random effects. We can see that all three methods give similar results, but BLB is much faster than the bootstrap.

Method	BLB	Bootstrap	Wald
Fixed Effect			
Intercept	(215.45, 232.02)	(216.72, 232.54)	(216.56, 232.56)
Visit Number	(-0.26, -0.22)	(-0.27, -0.22)	(-0.27, -0.22)
BMI	(-0.28, -0.05)	(-0.28, -0.06)	(-0.28, -0.06)
Female	(-2.39, 0.23)	(-2.17, 0.41)	(-2.25, 0.42)
Baseline Age	(-0.86, -0.67)	(-0.87, -0.68)	(-0.87, -0.67)
Race			
Black	(-10.42, -7.08)	(-10.26, -6.98)	(-10.30, -6.95)
Hispanic	(-4.12, 0.97)	(-4.80, 0.18)	(-4.83, 0.20)
Other	(-4.00, 0.32)	(-4.01, 0.13)	(-4.05, 0.10)
CVD History	(-0.87, 1.81)	(-0.32, 2.37)	(-0.34, 2.35)
Adjusted Insulin (units/kg body weight)	(-12.23, -8.64)	(-12.06, -9.36)	(-12.18, -9.35)
Sulphonylureas	(-0.51, 1.72)	(-0.57, 1.48)	(-0.58, 1.47)
Metformin	(-7.35, -4.69)	(-7.27, -4.88)	(-7.23, -4.82)
Meglitinides	(-14.93, -12.58)	(-14.80, -12.41)	(-14.82, -12.37)
Thiazolidinediones	(-21.28, -19.29)	(-21.35, -19.56)	(-21.38, -19.59)
Variance Components			
Intercept	(772.93, 883.30)	(809.81, 890.68)	
Visit Number	(0.20, 0.28)	(0.21, 0.26)	
Intercept : Visit Number	(-6.17, -3.82)	(-6.20, -4.49)	
Residual	(1814.22, 1906.54)	(1837.48, 1884.00)	
Time (second)	230	2650	

Table 1.2: Summary of the race variable for the ACCORD data

Race	White	Black	Hispanic	Other
N (%)	6351 (63%)	1946 (19%)	733 (7%)	1165 (11%)

Table 1.3: BLB 95% Confidence Intervals for the ACCORD data at different subset sizes

Method	Subset Size 1000	Subset Size 1600	Subset Size 2000
Fixed Effect			
Intercept	(219.29, 235.48)	(215.45,232.02)	(213.19, 229.81)
Visit Number	(-0.26, -0.22)	(-0.26, -0.22)	(-0.26, -0.22)
BMI	(-0.30, -0.08)	(-0.28, -0.05)	(-0.28, -0.05)
Female	(-2.19, 0.35)	(-2.39, 0.23)	(-2.40, 0.22)
Baseline Age	(-0.91, -0.72)	(-0.86, -0.67)	(-0.84, -0.64)
Race			
Black	(-9.94, -6.67)	(-10.42, -7.08)	(-10.36, -7.07)
Hispanic	(-5.90, -0.89)	(-4.12, 0.97)	(-4.59, 0.45)
Other	(-4.26, -0.04)	(-4.00, 0.32)	(-3.89, 0.29)
CVD History	(-0.13, 2.47)	(-0.87, 1.81)	(-0.40, 2.30)
Total Injected Insulin	(-12.14, -8.54)	(-12.23, -8.64)	(-12.37, -8.80)
Sulphonylureas	(-0.39, 1.84)	(-0.51, 1.72)	(-0.38, 1.79)
Metformin	(-8.05, -5.38)	(-7.35, -4.69)	(-7.03, -4.44)
Meglitinides	(-14.93, -12.51)	(-14.93,-12.58)	(-14.57, -12.10)
Thiazolidinediones	(-20.77, -18.79)	(-21.28, -19.29)	(-21.25, -19.20)
Variance Components			
Intercept	(781.15, 884.05)	(772.93, 883.30)	(789.33, 901.13)
Visit Number	(0.19, 0.28)	(0.20, 0.28)	(0.20, 0.27)
Intercept : Visit Number	(-6.22, -3.96)	(-6.17, -3.82)	(-6.67, -4.27)
Residual	(1823.22, 1919.71)	(1814.22, 1906.54)	(1807.12, 1899.79)

CHAPTER 2

Proximal MCMC for Bayesian Inference of Constrained and Regularized Estimation

2.1 Introduction

Many statistical learning tasks are posed as penalized maximum likelihood estimation problems, which require solving optimization problems of the form

$$\text{maximize} \quad \ell(\boldsymbol{\theta}) - \rho g(\boldsymbol{\theta}),$$

where $\boldsymbol{\theta}$ is a parameter specifying a model, $\ell(\boldsymbol{\theta})$ is a log-likelihood quantifying lack-of-fit between the model parameterized by $\boldsymbol{\theta}$ and the data, $g(\boldsymbol{\theta})$ is a penalty function that promotes the recovery of parameter estimates that have a desired structure, and ρ is a nonnegative regularization strength parameter that trades off model fit encoded in $\ell(\boldsymbol{\theta})$ with the desired structure encoded in $g(\boldsymbol{\theta})$. Canonical examples of penalty functions for $g(\boldsymbol{\theta})$ include the ℓ_1-norm which incentivizes recovery of sparse models and the nuclear norm which incentivizes recovery of low-rank models. To date, most work has focused exclusively on point estimates without quantifying the uncertainty in the estimates. Lacking tools for assessing the uncertainty in findings from regularized models, practitioners often resort to classical inference tools designed for fixed models. This practice can lead to seriously inflated type I error and is partly to blame for the reproducibility crisis in science [Ioa05].

This challenge has motivated the development of post-selection inference techniques such as simultaneous inference [BBB13, BPS20, KBB20] and selective inference [LSS16, CTT17,

TT18]. A closely related approach calculates confidence intervals for coefficients of high-dimensional linear models through bias-correction [GBR14, ZZ14, JM14]. Most of this literature, however, focuses on variable selection through the ℓ_1-penalty. Extending these strategies to more complicated penalties and constraints is not straightforward. Moreover, caution is warranted when reporting these confidence intervals because their interpretation (e.g., conditional on the selection event) can be quite different from traditional ones.

An alternative approach is to cast the problem in the Bayesian framework. For example, [PC08] introduced the Bayesian lasso. In this work, the ℓ_1-penalty was identified with a Laplace prior and a Gibbs sampler was used to sample from the posterior distribution. This early work helped spark the development of Bayesian variable selection methods, specifically alternative sparsity inducing prior distributions such as the spike-and-slab prior [MB88, GM93], horseshoe prior [CPS10, PS10, PV17], the orthant normal prior [Han11], the correlated Normal-Gamma prior [GB12, GB13], the generalized double Pareto prior [ADL13], and the Dirichlet-Laplace prior [BPP15]. While there have been many innovations in Bayesian techniques for variable selection, more general penalties and constraints beyond sparsity require substantially more problem-specific analysis.

More recently, [Per16] and [DMP18] proposed the proximal Markov Chain Monte Carlo (ProxMCMC) algorithm for quantifying uncertainty in Bayesian imaging applications where the penalties of interest include the total-variation semi-norm [ROF92] and the ℓ_1-norm. To deal with the non-smoothness of these penalties, they employ the Moreau-Yosida envelope to obtain a smooth approximation to the total-variation semi-norm and ℓ_1-norm penalties. Samples from the smooth approximate posterior distribution can be generated via the Langevin algorithm.

Their proposed approach opens the door to a potentially general and flexible framework for performing posterior inference for penalized regression models whenever the penalty term is convex and admits a proximal map which can be computed easily – a situation that holds true for a wide variety of convex penalties. The fly in the ointment, however, is that their

approach requires manually setting the regularization strength parameter ρ. This limitation prevents the previous formulations of ProxMCMC from being a fully Bayesian framework for generating posterior samples for many existing and more importantly potentially yet-to-be-invented non-smooth penalties.

Contributions: In this chapter, we address this limitation and extend ProxMCMC to be fully Bayesian by incorporating penalties and constraints through epigraph priors. Our extended ProxMCMC inference framework is suitable for regularized or constrained statistical learning problems and offers three main advantages.

First, it provides valid and automatic statistical inference even for problems that involve non-smooth and potentially non-convex penalties or constraints. The inference for such problems is traditionally considered difficult.

Second, it is fully Bayesian so that parameter tuning is not required, in contrast to previous ProxMCMC methods [DMP18] where the regularization strength parameters are either manually fixed or needs to be tuned.

Third, the method is highly modular in the sense that its components – model, prior, proximal map, and sampling algorithm – are independent of each other and thus can be modified to accommodate new problems. This feature makes ProxMCMC easily customizable so that practitioners can adapt it to their unique problems. For example, in analyzing compositional data from microbiome studies, one often needs to fit a lasso model where regression coefficients sum to 0. Although such problems are difficult to tackle in a post-selection inference framework, as we will demonstrate later in Section 2.5.1, it is straightforward to solve using our ProxMCMC method.

Finally, we put our ProxMCMC method on firm foundations by providing guarantees on properness of the approximate posterior and show that the approximate posterior can be made arbitrarily close to a target posterior in total-variation for both convex and non-convex penalties and constraints.

The rest of our chapter is organized as follows. Section 2 reviews concepts from convex optimization that underlie the algorithmic building blocks of our ProxMCMC framework. Section 3 illustrates our method using the familiar lasso problem. Section 4 summarizes the key elements from our case study of the lasso to show how our ProxMCMC method can be applied generally. Section 5 presents a variety of illustrative applications. Sections 6 and 7 provide theoretical guarantees and discussions respectively.

2.2 Background

We review concepts from convex analysis essential for ProxMCMC, specifically Moreau-Yosida envelopes and proximal maps. For a more thorough review of proximal maps and their applications in statistics and machine learning, we refer readers to [CW05, CP11, PSW15]. Recall in convex optimization it is often convenient to work with functions that map into the extended reals, $\bar{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$. The indicator function of a set C, denoted $\delta_C(\boldsymbol{x})$, is the function that is zero for all $\boldsymbol{x} \in C$ and is infinity for all $\boldsymbol{x} \notin C$. When the set C is closed and convex, the indicator function $\delta_C(\boldsymbol{x})$ of C is lower-semicontinuous and convex. A function g is proper if it takes on a finite value for some element in its domain. Let $\Gamma(\mathbb{R}^m)$ denote the set of all proper, lower-semicontinuous, convex functions from $\mathbb{R}^m$ into $\bar{\mathbb{R}}$. Let $\|\boldsymbol{x}\|$ denote the Euclidean norm of a point $\boldsymbol{x}$.

2.2.1 Moreau-Yosida Envelopes and Proximal Maps

Definition 1. *Given $g \in \Gamma(\mathbb{R}^m)$ and a positive scaling parameter λ, the Moreau-Yosida envelope of g, denoted by g^λ, is given by*

$$g^\lambda(\boldsymbol{x}) \;=\; \inf_{\boldsymbol{\omega}} \left\{ g(\boldsymbol{\omega}) + \frac{1}{2\lambda}\|\boldsymbol{\omega} - \boldsymbol{x}\|^2 \right\}.$$

The infimum is always attained at a unique point when $g \in \Gamma(\mathbb{R}^m)$, and the minimizer defines the proximal map of g.

Definition 2. *Given $g \in \Gamma(\mathbb{R}^m)$ and a positive scaling parameter λ, the proximal map of g, denoted* prox_g^λ*, is given by*

$$\text{prox}_g^\lambda(\boldsymbol{x}) \;=\; \arg\min_{\boldsymbol{\omega}} \left\{ g(\boldsymbol{\omega}) + \frac{1}{2\lambda}\|\boldsymbol{\omega} - \boldsymbol{x}\|^2 \right\}.$$

The well known Huber function [Bec17, Example 6.54]

$$g^\lambda(x) \;=\; \begin{cases} \frac{1}{2\lambda}x^2 & \text{if } |x| \le \lambda \\ |x| - \frac{\lambda}{2} & \text{otherwise} \end{cases}$$

is the Moreau-Yosida envelope of the absolute value function $g(x) = |x|$. Figure 2.1 shows $g(x)$ and $g^\lambda(x)$ for three different λ values. We can see immediately from this familiar example from robust statistics that the Moreau-Yosida envelope provides a differentiable approximation to a non-smooth function where the approximation improves as λ gets smaller.

In general, the Moreau-Yosida envelope $g^\lambda(\boldsymbol{x})$ has several important properties. First, $g^\lambda(\boldsymbol{x})$ is convex when $g(\boldsymbol{x})$ is convex. Second, if $g(\boldsymbol{x})$ is convex, then $g^\lambda(\boldsymbol{x})$ is always differentiable even if $g(\boldsymbol{x})$ is not, and its gradient can be expressed in terms of $\text{prox}_g^\lambda(\boldsymbol{x})$, namely

$$\nabla g^\lambda(\boldsymbol{x}) \;=\; \frac{1}{\lambda}\left[\boldsymbol{x} - \text{prox}_g^\lambda(\boldsymbol{x})\right]. \tag{2.1}$$

Moreover, $\nabla g^\lambda(\boldsymbol{x})$ is λ^{-1}-Lipschitz since proximal operators are firmly nonexpansive [CP11]. Finally and most importantly, $g^\lambda(\boldsymbol{x})$ converges pointwise to $g(\boldsymbol{x})$ as λ tends to zero [RW09]. In summary, the Moreau-Yosida envelope of a non-smooth function $g(\boldsymbol{x})$ is a Lipschitz-differentiable, arbitrarily close approximation to $g(\boldsymbol{x})$.

The closely related proximal maps play a prominent role in modern statistical learning since many popular non-smooth penalties have unique proximal maps that either have explicit formulas or can be computed efficiently. For example, the proximal map of $g(\boldsymbol{x}) = \|\boldsymbol{x}\|_1$

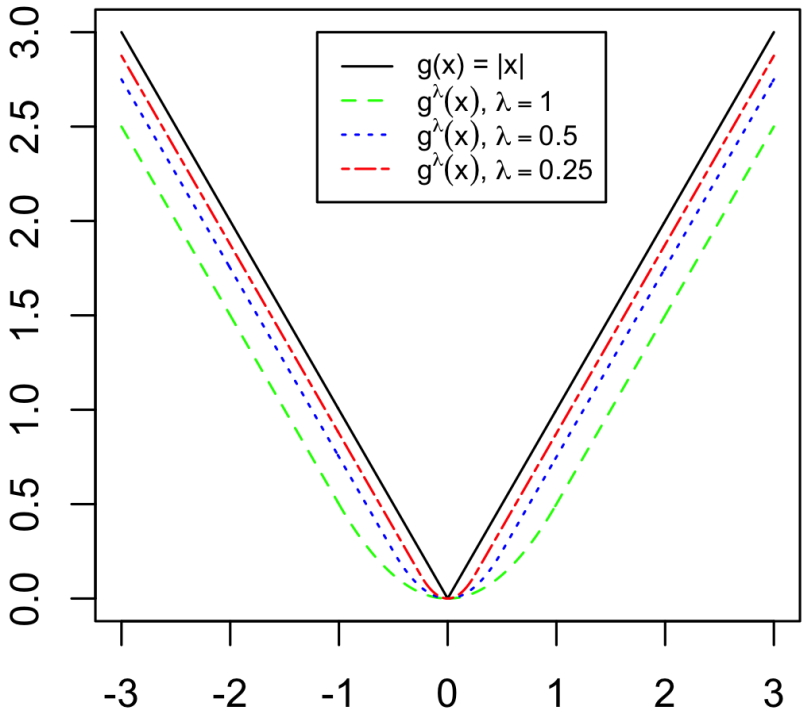


Figure 2.1: The Moreau-Yosida envelope of the absolute value function $g(x) = |x|$.

is the celebrated soft-thresholding operator $S_\lambda(\boldsymbol{x})$ defined by

$$S_\lambda(\boldsymbol{x})_i = \begin{cases} x_i - \lambda & \text{if } x_i > \lambda \\ 0 & \text{if } |x_i| \leq \lambda \\ x_i + \lambda & \text{if } x_i < -\lambda. \end{cases} \tag{2.2}$$

When the function g is an indicator function $\delta_{\mathcal{E}}$ of a set $\mathcal{E}$, the proximal map $\text{prox}_{\delta_{\mathcal{E}}}^{\lambda}(\boldsymbol{x})$ is the Euclidean projection operator onto the set $\mathcal{E}$, namely

$$\mathcal{P}_{\mathcal{E}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{\omega} \in \mathcal{E}} \|\boldsymbol{\omega} - \boldsymbol{x}\|$$

for all $\lambda > 0$. Let $d_{\mathcal{E}}(\boldsymbol{x})$ denote the Euclidean distance of the point $\boldsymbol{x}$ to the set $\mathcal{E}$, namely

$$d_{\mathcal{E}}(\boldsymbol{x}) \;\; = \;\; \inf_{\boldsymbol{y} \in \mathcal{E}} \|\boldsymbol{x} - \boldsymbol{y}\|,$$

then $P_{\mathcal{E}}(\boldsymbol{x})$ is the point in $\mathcal{E}$ that is closest in Euclidean distance to the set $\boldsymbol{x}$, namely

$$d_{\mathcal{E}}(\boldsymbol{x}) \;\; = \;\; \|\boldsymbol{x} - \mathcal{P}_{\mathcal{E}}(\boldsymbol{x})\|,$$

and the Moreau-Yosida envelope $g^\lambda(\boldsymbol{x})$ of $g(\boldsymbol{x}) = \delta_{\mathcal{E}}(\boldsymbol{x})$ is

$$g^\lambda(\boldsymbol{x}) \;\; = \;\; \frac{1}{2\lambda} d_{\mathcal{E}}^2(\boldsymbol{x}) = \frac{1}{2\lambda} \|\boldsymbol{x} - \mathcal{P}_{\mathcal{E}}(\boldsymbol{x})\|^2.$$

2.2.2 Projections onto Epigraphs

The key algorithmic primitive in our ProxMCMC framework is the projection onto the epigraph of a penalty function $g(\boldsymbol{x})$, namely the set

$$\mathcal{E} = \text{epi}(g) = \{(\boldsymbol{x}, \alpha) : g(\boldsymbol{x}) \leq \alpha\}.$$

The Moreau-Yosida envelope of the indicator function $g(\boldsymbol{x}, \alpha) = \delta_{\mathcal{E}}(\boldsymbol{x}, \alpha)$ of $\mathcal{E} = \text{epi}(g)$ plays a central role in defining our Bayesian hierarchical model. The Moreau-Yosida envelope of $\delta_{\mathcal{E}}(\boldsymbol{x}, \alpha)$ is $\frac{1}{2\lambda} d_{\mathcal{E}}^2(\boldsymbol{x}, \alpha)$, which is jointly differentiable in $\boldsymbol{x}$ and α. Subsequently, we can

assign α a prior to incorporate it into posterior inference. Computing with these priors relies on projection onto epigraphs which we describe next.

Projection onto the epigraph of $g(\boldsymbol{x}, \alpha)$ depends on the proximal map of $g(\boldsymbol{x}, \alpha)$ [Bec17, Theorem 6.36], namely

$$\mathcal{P}_{\mathcal{E}}(\boldsymbol{x}, \alpha) \;=\; \begin{cases} (\boldsymbol{x}, \alpha) & g(\boldsymbol{x}) \leq \alpha \\ \left(\text{prox}_g^{\lambda^*}(\boldsymbol{x}), \alpha + \lambda^*\right) & g(\boldsymbol{x}) > \alpha \end{cases}, \tag{2.3}$$

where λ^* is any positive root of the auxiliary function

$$F(\lambda) \;=\; g\left(\text{prox}_g^{\lambda}(\boldsymbol{x})\right) - \lambda - \alpha,$$

and can be found using bisection.

2.3 An illustrative case study

To introduce our framework, we first consider a canonical example: the lasso regression [Tib96]

$$\text{minimize} \quad \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \rho\|\boldsymbol{\beta}\|_1, \tag{2.4}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ is a vector of continuous responses, $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ is a design matrix whose p columns are covariates, $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of regression coefficients that we seek to estimate, and ρ is a nonnegative regularization strength parameter that trades off model fit with sparsity in our estimate of $\boldsymbol{\beta}$. To attack this problem in the ProxMCMC framework, we first write the penalized form (2.4) in an equivalent constrained form

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{\beta}\|_1 \leq \alpha. \end{aligned}$$

There is a one-to-one correspondence between the regularization strength parameter ρ and the constraint parameter α. For this reason we will refer to α as the regularization strength

parameter as well. We specify the following Bayesian hierarchical model for the constrained formulation of the lasso:

- Data likelihood: $\boldsymbol{Y} \mid \boldsymbol{\beta}, \sigma^2 \sim N(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2 \boldsymbol{I})$,
- A prior $\pi(\sigma^2)$ for the variance: $\sigma^2 \sim IG(r_{\sigma^2}, s_{\sigma^2})$, where $IG(r, s)$ denotes the Inverse-Gamma distribution with scale parameter r and shape parameter s (mean $= \frac{r}{s-1}$ for $s > 1$),
- A prior $\pi(\boldsymbol{\beta} \mid \alpha)$ for $\boldsymbol{\beta}$ conditional on α, namely

$$\pi(\boldsymbol{\beta} \mid \alpha) = \frac{p!}{\alpha^p 2^p} \exp\left[-\delta_{\mathcal{E}}(\boldsymbol{\beta}, \alpha)\right],$$

  where $\mathcal{E} = \{(\boldsymbol{\beta}, \alpha) : \|\boldsymbol{\beta}\|_1 \leq \alpha\}$ and $\frac{p!}{\alpha^p 2^p}$ is the reciprocal of the volume of $\mathcal{E}$. Note that $\pi(\boldsymbol{\beta} \mid \alpha)$ is a flat prior over an ℓ_1-ball of radius α.

- A prior $\pi(\alpha)$ on α that controls the ℓ_1-regularization strength: $\alpha \sim IG(r_\alpha, s_\alpha)$.

The distribution $\pi(\boldsymbol{\beta}, \alpha) = \pi(\boldsymbol{\beta} \mid \alpha) \cdot \pi(\alpha)$ specifies a prior on the epigraph $\mathcal{E} = \{(\boldsymbol{\beta}, \alpha) : \|\boldsymbol{\beta}\|_1 \leq \alpha\} \subset \mathbb{R}^{p+1}$. The posterior log-density takes the following form, up to an irrelevant additive constant,

$$\begin{aligned} &\log \pi(\boldsymbol{\beta}, \sigma^2, \alpha) \\ =& -\left(\frac{n}{2} + s_{\sigma^2} + 1\right) \log \sigma^2 - \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + 2r_{\sigma^2}}{2\sigma^2} \\ &- (s_\alpha + 1) \log \alpha - \frac{r_\alpha}{\alpha} - g(\boldsymbol{\beta}, \alpha), \end{aligned}$$

where $g(\boldsymbol{\beta}, \alpha) = \delta_{\mathcal{E}}(\boldsymbol{\beta}, \alpha)$.

The above posterior is not differentiable, but we can approximate it arbitrarily well with a differentiable posterior. The key idea is to approximate the non-smooth function $g(\boldsymbol{\beta}, \alpha)$

by its Moreau-Yosida envelope $g^\lambda(\boldsymbol{\beta}, \alpha)$. The smoothed posterior log-density is

$$\begin{aligned}
&\log \pi^\lambda(\boldsymbol{\beta}, \sigma^2, \alpha) \\
&= -\left(\frac{n}{2} + s_{\sigma^2} + 1\right) \log \sigma^2 - \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + 2r_{\sigma^2}}{2\sigma^2} \\
&\quad - (s_\alpha + 1) \log \alpha - \frac{r_\alpha}{\alpha} - g^\lambda(\boldsymbol{\beta}, \alpha),
\end{aligned}$$

which, due to the smoothness of the Moreau-Yosida envelope $g^\lambda(\boldsymbol{\beta}, \alpha)$ [RW09], can be readily sampled using any of a multitude of sampling algorithms for smooth log-densities. In this work, we use Hamiltonian Monte Carlo (HMC) [Nea11] due to its efficiency and generality. Since HMC works on unconstrained domains, we use the parameterization $(\boldsymbol{\beta}, \log \sigma^2, \log \alpha)$, so

$$\begin{aligned}
&\log \pi^\lambda(\boldsymbol{\beta}, \log \sigma^2, \log \alpha) \\
&= -\left(\frac{n}{2} + s_{\sigma^2}\right) \log \sigma^2 - \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + 2r_{\sigma^2}}{2\sigma^2} \\
&\quad - s_\alpha \log \alpha - \frac{r_\alpha}{\alpha} - g^\lambda(\boldsymbol{\beta}, \alpha).
\end{aligned}$$

We compare ProxMCMC to the Bayesian lasso on the diabetes data set in [EHJ04]. The outcome is a quantitative measure of disease progression over a year, and the covariates are age, sex, body mass index, average blood pressure, and six blood serum measurements. All variables are standardized to have zero mean and unit variance. For the Bayesian lasso, we use the `blasso` function from the `R` package `monomvn` [Gra19] with default parameters. We show the results of the Bayesian lasso with and without using reversible jump MCMC (RJMCMC) to perform model selection. For ProxMCMC, we set $\lambda = 0.001$, $\sigma^2 \sim IG(0.1, 0.1)$, and $\alpha \sim IG(1, 10 + 2)$. We also calculate the 95% selective inference confidence intervals [LSS16] using the `R` package `selectiveInference` [TTT19]. Since this method requires a model to be selected first, we use lasso with 10-fold cross-validation and choose the largest regularization parameter such that the error is within 1 standard error of the minimum (the `lambda.1se` option from the `glmnet` package). Figure 2.2 displays the interval estimates of the regression coefficient computed by each method. We see that for null covariates, the

credible intervals of the Bayesian lasso are narrower when model selection by RJMCMC is used. This is because RJMCMC results in many exact zeros (75% in this example) in the posterior sample, which reduce the variability and thus the width of credible intervals. When RJMCMC is not used, the credible intervals of the null covariates become wider and are similar to those obtained by the ProxMCMC method. The credible intervals for non-null covariates are similar regardless of which method is used. The selective inference confidence intervals have a different interpretation. The coverage guarantee is in the frequentist sense and is conditional on the model being selected, so they are not directly comparable with credible intervals. Nevertheless, it is interesting to note that two of the four intervals for the selected variables are extremely wide.

2.4 Methodology

Having seen how to apply our ProxMCMC method in the special case of the lasso, we next present the framework in greater generality. Our proposed ProxMCMC method consists of three steps.

1. Likelihood and prior. The first step is to specify the likelihood model for the data Y and priors for the model parameters. This is a standard step in Bayesian modeling. For incorporating a penalty function $g(\boldsymbol{x})$ with a regularization strength parameter α, we specify a prior through an indicator function $\delta_{\mathcal{E}}(\boldsymbol{x}, \alpha)$ on the epigraph set

$$\mathcal{E} = \{(\boldsymbol{x}, \alpha) : g(\boldsymbol{x}) \leq \alpha\}.$$

For example, in the lasso example where $\boldsymbol{\beta}$ was a vector of regression coefficients and α was a regularization parameter, $\mathcal{E} = \{(\boldsymbol{\beta}, \alpha) : \|\boldsymbol{\beta}\|_1 \leq \alpha\}$.

Note that the regularization parameter α must be nonnegative and thus requires a prior with nonnegative support. In our experience, placing an inverse Gamma prior on α works well in practice and will be used throughout this chapter.

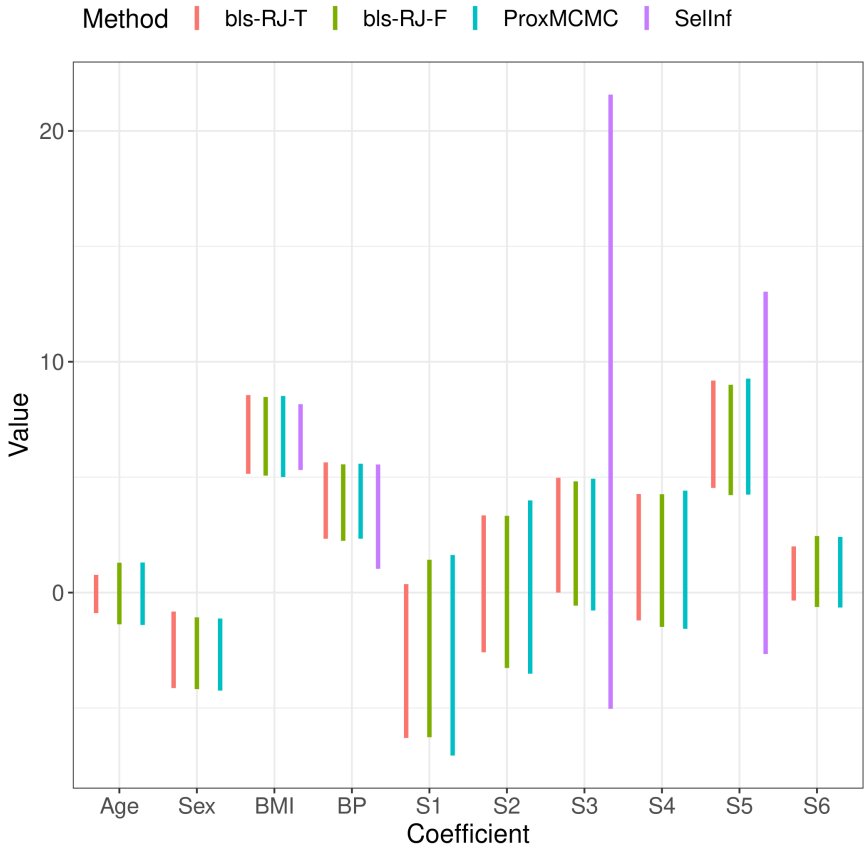


Figure 2.2: The 95% credible intervals calculated by Bayesian lasso (bls) and ProxMCMC. bls-RJ-T denotes that RJMCMC is used in computing the posterior samples of the Bayesian lasso, while bls-RJ-F denotes results when RJMCMC is not used . Also shown are the 95% selective inference confidence intervals (SelInf) for the four variables selected by the lasso using 10-fold cross-validation.

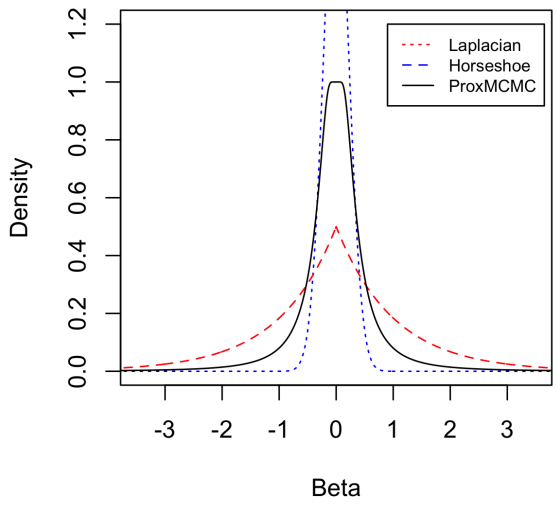


Figure 2.3: The ProxMCMC epigraph prior and two other commonly used shrinkage priors.

To gain a sense of how the ProxMCMC epigraph prior differs from existing alternatives, consider the simple case where we put a ℓ_1-penalty on a single parameter β, namely

$$\mathcal{E} \;=\; \{(\beta, \alpha) : |\beta| \leq \alpha\}.$$

With an $IG(r, s)$ prior on α, the marginal density for β is

$$f_\beta(t) = \int_{|t|}^{\infty} \frac{1}{2\alpha}\pi(\alpha)d\alpha = \frac{s}{2r}\big[1 - F_{IG(r,s+1)}(|t|)\big],$$

where $F_{IG(r,s+1)}(|t|)$ is the cumulative distribution function of $IG(r, s+1)$ evaluated at $|t|$. Figure 2.3 contrasts the prior densities of the ProxMCMC epigraph prior, Laplacian prior, and horseshoe prior. We can see that the ProxMCMC epigraph prior density shrinks small β while allowing strong signals to remain large.

In the multivariate case, when a vector of parameters $\boldsymbol{\beta}$ is penalized, e.g., $\mathcal{E} = \{(\boldsymbol{\beta}, \alpha) : \|\boldsymbol{\beta}\|_1 \leq \alpha\}$, the ProxMCMC epigraph prior enforces negative correlation among the components of $\boldsymbol{\beta}$. This repulsive feature distinguishes it from other Bayesian priors such as the Laplacian or horseshoe prior, where components of $\boldsymbol{\beta}$ are independent of each other conditional on the hyperparameter and marginally positively correlated.

Besides incorporating penalties, our framework can also handle cases where we wish to impose constraints. As before, we will enforce the constraints via an indicator function of the constraint set.

Once the likelihood model and the priors are specified, we can write down the posterior. Let $\boldsymbol{\theta} \in \mathbb{R}^d$ denote all model parameters, which includes the constrained or regularized parameters $\boldsymbol{\tau} \in \mathbb{R}^p$ and all other parameters $\boldsymbol{\eta} \in \mathbb{R}^q$ (so $d = p+q$) including the regularization strength parameter α. Further let $\ell(\boldsymbol{\theta})$ be the log-likelihood, $\pi(\boldsymbol{\eta})$ denote the prior density for $\boldsymbol{\eta}$, and $g(\boldsymbol{\tau}) = \delta_{\mathcal{E}}(\boldsymbol{\tau})$, where $\mathcal{E}$ denotes either the constraint set or the epigraph set depending on the problem. Note that if $\mathcal{E}$ is an epigraph set, then g is a function of both $\boldsymbol{\tau}$ and α, otherwise it is a function of $\boldsymbol{\tau}$ alone. For simplicity, we will write $g(\boldsymbol{\tau})$ unless it is necessary to be more specific. The posterior density is given by

$$\pi(\boldsymbol{\theta} \mid Y) = \frac{e^{-U(\boldsymbol{\theta})}}{\int e^{-U(\boldsymbol{s})}\, d\boldsymbol{s}},$$

where $U(\boldsymbol{\theta}) = f(\boldsymbol{\theta}) + g(\boldsymbol{\tau})$ and $f(\boldsymbol{\theta}) = -\ell(\boldsymbol{\theta}) - \log \pi(\boldsymbol{\eta})$. The posterior $\pi(\boldsymbol{\theta} \mid Y)$ is not differentiable because $g(\boldsymbol{\tau})$ is not differentiable, but $\pi(\boldsymbol{\theta} \mid Y)$ can be smoothed by substituting $g(\boldsymbol{\tau})$ with its Moreau-Yosida envelope $g^\lambda(\boldsymbol{\tau})$ so that both $U^\lambda(\boldsymbol{\theta}) = f(\boldsymbol{\theta}) + g^\lambda(\boldsymbol{\tau})$ and

$$\pi^\lambda(\boldsymbol{\theta} \mid Y) = \frac{e^{-U^\lambda(\boldsymbol{\theta})}}{\int e^{-U^\lambda(\boldsymbol{s})}\, d\boldsymbol{s}}$$

become smooth functions.

2. Gradient. We need to efficiently evaluate the gradient of the smoothed posterior log-density. This is another standard step in Bayesian modeling, and for commonly used likelihood models and priors, the gradient can be computed numerically by auto-differentiation in software packages such as `Stan` [Sta20] and `Turing.jl` [GXG18].

As noted earlier, the existence of the gradient of the Moreau-Yosida envelope $g^\lambda(\boldsymbol{\tau})$ depends on the convexity of $g(\boldsymbol{\tau})$ and thus of the convexity of the regularization term or constraint set $\mathcal{E}$. When $g(\boldsymbol{\tau})$ is convex, which is the case for many commonly used regularization and constraints, proximal mappings have been extensively studied in the optimization literature [Bec17, Chapter 6] and efficient implementations are available from mature libraries

such as FOM Matlab toolbox [BG19], Python package `PyProximal`, and Julia package `ProximalOperators.jl`.

When $g(\boldsymbol{\tau})$ is non-convex, $g^{\lambda}(\boldsymbol{\tau})$ is no longer differentiable. However, as we will show in Section 2.5.4, under certain regularity conditions, $g^{\lambda}(\boldsymbol{\tau})$ will be semidifferentiable and we can calculate a subgradient using the above formula and use it in place of gradient in gradient based samplers.

3. Sampling algorithm. Finally, we invoke a gradient based sampling algorithm such as HMC or the Langevin algorithm to efficiently explore the posterior landscape. Software libraries include `DynamicHMC.jl`, `AdvancedHMC.jl`, and `pyhmc`, to name a few.

Remark: Before we proceed to examples, we pause to highlight ProxMCMC's close connection to distance majorization and proximal distance algorithms [CZL14, XCL17, KZL19, LL21, LWL22, LPZ22]. Proximal distance algorithms are used to solve distance penalty problems.

$$\text{minimize} \quad f(\boldsymbol{\theta}) + \frac{\rho}{2} d_{\mathcal{E}}(\boldsymbol{\theta})^2, \tag{2.5}$$

where $f(\boldsymbol{\theta})$ is typically a negative log-likelihood term quantifying model fit, $\mathcal{E}$ is a target constraint set that we wish our estimate of $\boldsymbol{\theta}$ to be close to, and ρ is a nonnegative tuning parameter that trades off model fit with the amount of constraint violation quantified as the distance to $\mathcal{E}$. A solution to (2.5) is a maximum a posteriori estimate under a distance-to-set prior $\pi(\boldsymbol{\theta}) \propto \exp(-\frac{\rho}{2} d_{\mathcal{E}}(\boldsymbol{\theta})^2)$. Thus, the ProxMCMC method that we introduce in this work provides a fully Bayesian framework for generating posterior samples under a distance-to-epigraph set prior.

2.5 Examples

We present four examples to illustrate the generality of ProxMCMC. Since the potential applications of ProxMCMC are innumerable, our examples are not comprehensive. Nonethe-

less, we chose these four examples because inference in these problems are either unknown or regarded difficult.

2.5.1 Constrained lasso

The constrained lasso problem is formulated as

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \rho\|\boldsymbol{\beta}\|_1 \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{\beta} = \boldsymbol{b}, \end{aligned}$$

where $\boldsymbol{A}$ has full row-rank. The constrained lasso is relevant to the analysis of compositional data, where the rows of $\boldsymbol{A}$ represents proportions of a whole and thus must sum to one. The method has been applied in problems involving consumer spending in economics, topic consumption of documents in machine learning, and the analysis of the human microbiome [GKZ18, JPR20].

The ℓ_1-penalization is reparameterized using an indicator function on the epigraph of the ℓ_1-norm

$$\mathcal{E}_1 = \{(\boldsymbol{\beta}, \alpha) : \|\boldsymbol{\beta}\|_1 \leq \alpha\}.$$

The equality constraint is imposed through an indicator function on the hyperplane

$$\mathcal{E}_2 = \{\boldsymbol{\beta} : \boldsymbol{A}\boldsymbol{\beta} = \boldsymbol{b}\}.$$

We use an $IG(r_{\sigma^2}, s_{\sigma^2})$ prior for σ^2 and an $IG(r_\alpha, s_\alpha)$ prior for α. Using the $(\boldsymbol{\beta}, \log\sigma^2, \log\alpha)$ parameterization, the smoothed posterior log-density up to an irrelevant additive constant is

$$\begin{aligned} & \log\pi^\lambda(\boldsymbol{\beta}, \log\sigma^2, \log\alpha) \\ = & -\left(\frac{n}{2} + s_{\sigma^2}\right)\log\sigma^2 - \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + 2r_{\sigma^2}}{2\sigma^2} \\ & - s_\alpha\log\alpha - \frac{r_\alpha}{\alpha} - g_1^\lambda(\boldsymbol{\beta}, \alpha) - g_2^\lambda(\boldsymbol{\beta}), \end{aligned}$$

where $g_1^\lambda(\boldsymbol{\beta}, \alpha)$ and $g_2^\lambda(\boldsymbol{\beta})$ are the Moreau-Yosida envelopes of the indicator functions $g_1(\boldsymbol{\beta}, \alpha) = \delta_{\mathcal{E}_1}(\boldsymbol{\beta}, \alpha)$ and $g_2(\boldsymbol{\beta}) = \delta_{\mathcal{E}_2}(\boldsymbol{\beta})$, respectively. According to (2.3), the proximal map of $g_1(\boldsymbol{\beta}, \alpha)$ is the projection onto the epigraph $\mathcal{E}_1$

$$\text{prox}_{g_1}^\lambda(\boldsymbol{\beta}, \alpha) = \begin{cases} (\boldsymbol{\beta}, \alpha) & \text{if } \|\boldsymbol{\beta}\|_1 \leq \alpha \\ (S_{\lambda^*}(\boldsymbol{\beta}), \alpha + \lambda^*) & \text{if } \|\boldsymbol{\beta}\|_1 > \alpha \end{cases},$$

where S_λ is the soft-thresholding operator given in (2.2) and λ^* is any positive root of the nonincreasing function $\phi(\lambda) = \|S_\lambda(\boldsymbol{\beta})\|_1 - \lambda - \alpha$ [Bec17]. The proximal map of g_2 is the projection onto the hyperplane given by

$$\text{prox}_{g_2}(\boldsymbol{\beta}) = \boldsymbol{\beta} - \boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{A}^T)^{-1}(\boldsymbol{A}\boldsymbol{\beta} - \boldsymbol{b}),$$

assuming $\boldsymbol{A}$ has full row rank. The gradient of the posterior log-density is given block-wise by

$$\begin{aligned} \frac{\partial \log \pi^\lambda}{\partial \boldsymbol{\beta}} &= \sigma^{-2}\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) - \lambda^{-1}\left[\boldsymbol{\beta} - \text{prox}_{g_1}^\lambda(\boldsymbol{\beta}, \alpha)_{\boldsymbol{\beta}}\right] \\ &\quad -\lambda^{-1}\left[\boldsymbol{\beta} - \text{prox}_{g_2}^\lambda(\boldsymbol{\beta})\right] \\ \frac{\partial \log \pi^\lambda}{\partial \log \sigma^2} &= -\left(\frac{n}{2} + s_{\sigma^2}\right) + \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + 2r_{\sigma^2}}{2\sigma^2} \\ \frac{\partial \log \pi^\lambda}{\partial \log \alpha} &= -s_\alpha + \frac{r_\alpha}{\alpha} - \lambda^{-1}\alpha[\alpha - \text{prox}_g^\lambda(\boldsymbol{\beta}, \alpha)_\alpha]. \end{aligned}$$

2.5.1.1 Simulated microbiome data

We illustrate our proxMCMC method for the constrained lasso using a simulated microbiome data set. The 16S microbiome sequencing technology measures the number of various organisms called operational taxonomic units (OTUs) in samples. For statistical analysis, counts are normalized into proportions for each sample, resulting in a covariate matrix $\boldsymbol{X}$ with each of its rows summing to 1. For identifiability, we need to have a sum-to-zero constraint on the regression coefficients, i.e., $\sum_j \beta_j = 0$. We generate a data set with $n = 300$ samples and $p = 20$ OTUs. We set $\beta_1 = 1$, $\beta_2 = -1$ and the remaining β_j, $j = 3, ..., 20$, to 0 so that 10%

of the entries in $\boldsymbol{\beta}$ are nonzero. The design matrix $\boldsymbol{X}$ is generated as follows. First, for each element in $\boldsymbol{X}$ an i.i.d. sample from a uniform distribution ($U_{[0,1]}$) is drawn. Second, the rows of $\boldsymbol{X}$ are then scaled so that each row of $\boldsymbol{X}$ sums to 1. The noise is generated from a normal distribution with mean 0 and $\sigma = 0.1$ so that the sample signal-to-noise ratio $\text{Var}(\boldsymbol{X}\boldsymbol{\beta})/\sigma^2$ is approximately 0.2. We use $IG(0.1, 0.1)$ as a prior for σ^2 and $IG(1, p+1)$ as a prior for α, set $\lambda = 10^{-6}$, and ran HMC for 10,000 iterations. From Figure 2.6 (a), we can see that the 95% credible intervals provide good coverage for the first 10 coefficients; those for the other coefficients are similar. Figure 2.6 (b) shows the histogram of posterior samples of $\sum_j \beta_j$, which is highly concentrated around 0.

2.5.2 Graphical lasso

Given i.i.d. p-dimensional observations $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$, where $\boldsymbol{x}_i \sim N(\boldsymbol{0}, \boldsymbol{\Sigma})$, the graphical lasso method infers the underlying conditional dependencies among the covariates by estimating the precision matrix $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$ through maximizing the regularized log-likelihood

$$-\frac{n}{2}\text{tr}(\boldsymbol{S}\boldsymbol{\Theta}) + \frac{n}{2}\log\det(\boldsymbol{\Theta}) - \rho\sum_{j\neq k}|\boldsymbol{\Theta}_{jk}|,$$

where $\boldsymbol{S}$ is the sample covariance and ρ is the regularization strength parameter. Equivalently, we can maximize

$$-\frac{n}{2}\text{tr}(\boldsymbol{S}\boldsymbol{\Theta}) + \frac{n}{2}\log\det(\boldsymbol{\Theta}) - g(\boldsymbol{\Theta}, \alpha),$$

where $g(\boldsymbol{\Theta}, \alpha) = \delta_{\mathcal{E}}(\boldsymbol{\Theta}, \alpha)$ and $\mathcal{E} = \{(\boldsymbol{\Theta}, \alpha) : \sum_{j\neq k}|\boldsymbol{\Theta}_{jk}| \leq \alpha\}$. The function $g(\boldsymbol{\Theta}, \alpha)$ can be thought as a uniform prior for $\boldsymbol{\Theta}$ over the ℓ_1-ball $\{\boldsymbol{\Theta} : \sum_{j\neq k}|\boldsymbol{\Theta}_{jk}| \leq \alpha\}$. With an $IG(r_\alpha, s_\alpha)$ prior for α, the smoothed posterior log-density of $(\boldsymbol{\Theta}, \log\alpha)$ is

$$\begin{aligned}\log\pi^\lambda(\boldsymbol{\Theta}, \log\alpha) = &-\frac{n}{2}\text{tr}(\boldsymbol{S}\boldsymbol{\Theta}) + \frac{n}{2}\log\det(\boldsymbol{\Theta}) \\ &- s_\alpha\log\alpha - \frac{r_\alpha}{\alpha} - g^\lambda(\boldsymbol{\Theta}, \alpha).\end{aligned}$$

Since HMC works on unconstrained domains and $\boldsymbol{\Theta}$ needs to be positive definite, we parameterize $\boldsymbol{\Theta}$ in terms of its lower Cholesky factor $\boldsymbol{L}$. Adjusting for the log-Jacobian terms, the

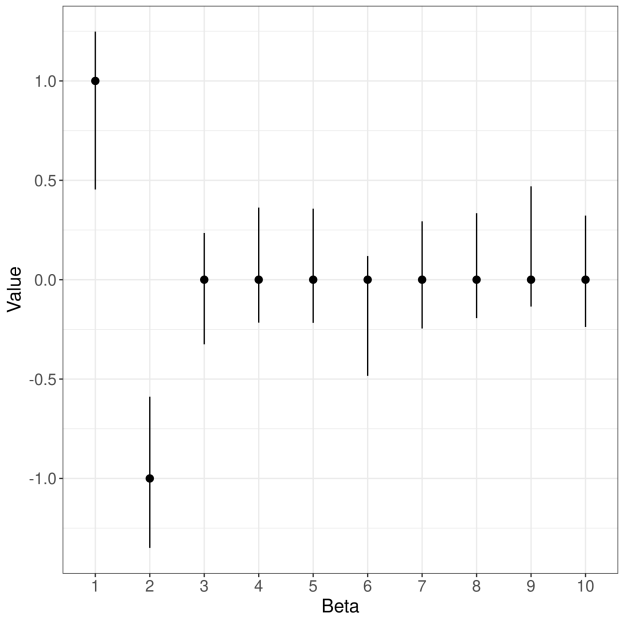


Figure 2.4: 95% credible intervals for the first 10 coefficients. Dots mark the truth.

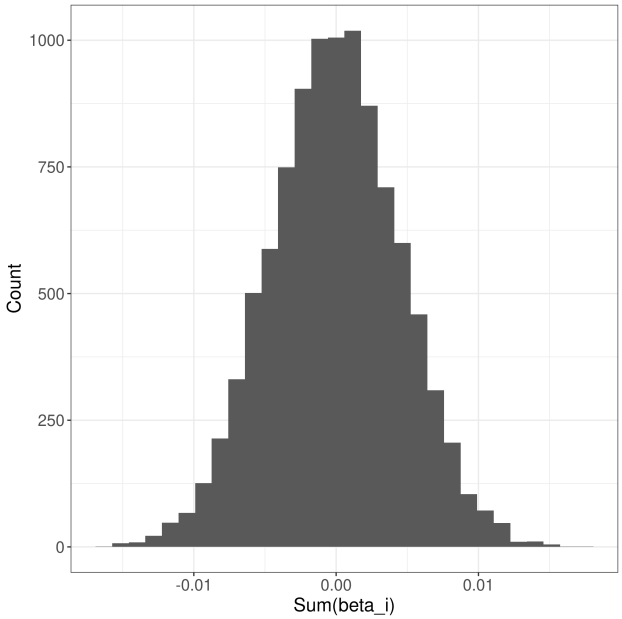


Figure 2.5: Histogram of $\sum_i \beta_i$.

Figure 2.6: Results from the simulated microbiome data

smoothed posterior log-density becomes

$$\begin{aligned}\log \pi^{\lambda}(\boldsymbol{L}, \log \alpha) = &-\frac{n}{2}\mathrm{tr}(\boldsymbol{S}\boldsymbol{L}\boldsymbol{L}^T) + \frac{n}{2}\,\mathrm{logdet}(\boldsymbol{L}\boldsymbol{L}^T) \\ &- s_\alpha \log \alpha - \frac{r_\alpha}{\alpha} - g^{\lambda}(\boldsymbol{L}\boldsymbol{L}^T, \alpha) \\ &+ p\log(2) + \sum_{j=1}^{p}(p-j+2)\boldsymbol{L}_{jj}.\end{aligned}$$

The gradients are given by

$$\begin{aligned}\nabla_{\mathrm{vech}\boldsymbol{L}} \log \pi^{\lambda} &= -\big(n\mathrm{vech}(\boldsymbol{S}\boldsymbol{L})\big)^T + n\big(\mathrm{vech}(\boldsymbol{L}^{-1})^T\big)^T \\ &\quad -\frac{2}{\lambda}\bigg(\mathrm{vech}\Big([\boldsymbol{\Theta} - \mathrm{prox}_g^{\lambda}(\boldsymbol{\Theta}, \alpha)_{\boldsymbol{\Theta}}]\boldsymbol{L}\Big)\bigg)^T \\ &\quad +\Big(\mathrm{vech}\big(\mathrm{diag}(p+1, p, ..., 2)\big)\Big)^T \\ \frac{\partial \log \pi^{\lambda}}{\partial \log \alpha} &= -s_\alpha + \frac{r_\alpha}{\alpha} - \lambda^{-1}\alpha[\alpha - \mathrm{prox}_g^{\lambda}(\boldsymbol{\Theta}, \alpha)_\alpha],\end{aligned}$$

where $\mathrm{vech}(\boldsymbol{A})$ denotes the vector obtained from stacking the columns of the lower triangular part of a square matrix $\boldsymbol{A}$.

2.5.2.1 Cytometry data

We compare ProxMCMC with the Bayesian graphical lasso [Wan12] on the cell-signalling data from [SPP05], which was used in the original graphical lasso paper [FHT08]. The data set contains flow cytometry measurements on $p = 11$ proteins and $n = 7466$ cells. We first use the `R` package `CVglasso` to compute 5-fold cross-validated graphical lasso estimates for $\boldsymbol{\Theta}$ and use them as reference in the results below. For the Bayesian graphical lasso we use the `R` package `BayesianGLasso` [Wan12]. We experimented with both the default prior and a few other prior settings, but found little difference among them. Thus we report the results using the default prior (Gamma distribution with shape parameter 1 and scale parameter 0.1). For ProxMCMC we use an $IG(1, p+1)$ prior for α and set $\lambda = 0.01$. We ran 10,000 iterations for both methods. Figure 2.7 displays the credible intervals. Due to the large number of parameters, we only show the results for the first ten parameters in the plot, but

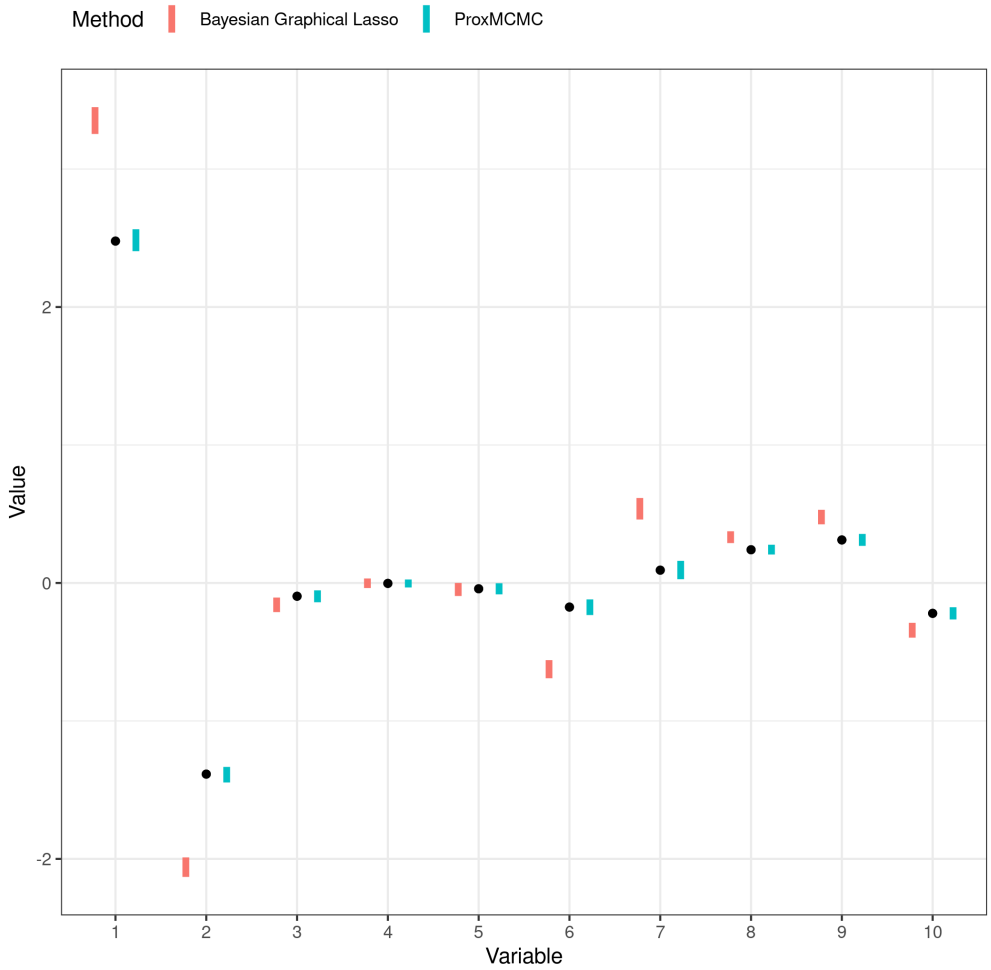


Figure 2.7: Comparing the 95% credible intervals of Bayesian graphical lasso versus ProxMCMC on the cytometry data. Black dots are estimates obtained from 5-fold cross-validated graphical lasso.

the same pattern is observed in other parameters. We can see that the ProxMCMC credible intervals are consistently narrower and provide good coverage of the graphical lasso estimate, whereas those provided by the Bayesian graphical lasso can be wide and fail to cover the cross-validated estimates. Among all 66 parameters, all ProxMCMC credible intervals cover the reference values whereas only 24% of the Bayesian graphical lasso credible intervals do.

2.5.3 Matrix completion

Given a matrix $\boldsymbol{Y}$ with entries only observed on the set $\Omega = \{(i,j) : y_{ij} \text{ is observed}\}$, [MHT10] propose to complete the matrix by minimizing the convex objective function

$$\frac{1}{2}\sum_{(i,j)\in\Omega}(y_{ij}-x_{ij})^2 + \rho\|\boldsymbol{X}\|_*,$$

where ρ is a regularization strength parameter and $\|\boldsymbol{X}\|_*$ is the nuclear norm of the completed matrix $\boldsymbol{X}$. The nuclear norm is defined as $\|\boldsymbol{X}\|_* = \|\boldsymbol{\sigma}(\boldsymbol{X})\|_1 = \sum_i \sigma_i(\boldsymbol{X})$, where $\sigma_1(\boldsymbol{X}) \geq \cdots \geq \sigma_m(\boldsymbol{X}) \geq 0$ are the singular values of $\boldsymbol{X}$. To put the problem into the ProxMCMC framework, we assume $\text{vec}\boldsymbol{Y} \sim N(\text{vec}\boldsymbol{X}, \sigma^2\boldsymbol{I})$. Let $\mathcal{E} = \{(\boldsymbol{X},\alpha) : \|\boldsymbol{X}\|_* \leq \alpha\}$ and $g(\boldsymbol{X},\alpha) = \delta_{\mathcal{E}}(\boldsymbol{X},\alpha)$. With an $IG(r_{\sigma^2}, s_{\sigma^2})$ prior for σ^2 and an $IG(r_\alpha, s_\alpha)$ prior for α, the smoothed posterior log-density using the $\log\sigma^2, \log\alpha$ parameterization is

$$\begin{aligned}
&\log\pi^\lambda(\boldsymbol{X}, \log\sigma^2, \log\alpha) \\
=&-\left(\frac{|\Omega|}{2}+s_{\sigma^2}\right)\log\sigma^2 - \frac{\sum_{(i,j)\in\Omega}(y_{ij}-x_{ij})^2 + 2r_{\sigma^2}}{2\sigma^2} \\
&- s_\alpha\log\alpha - \frac{r_\alpha}{\alpha} - g^\lambda(\boldsymbol{X},\alpha),
\end{aligned}$$

The proximal mapping of $g(\boldsymbol{X},\alpha)$ is the projection given by

$$\begin{aligned}
&\text{prox}_g^\lambda(\boldsymbol{X},\alpha) \\
=&\begin{cases}
(\boldsymbol{X},\alpha) & \text{if } \|\boldsymbol{X}\|_* \leq \alpha \\
(\boldsymbol{U}\text{diag}(S_{\lambda^*}(\sigma(\boldsymbol{X})))\boldsymbol{V}^T, \alpha+\lambda^*) & \text{if } \|\boldsymbol{X}\|_* > \alpha
\end{cases},
\end{aligned}$$

where S_λ is the soft-thresholding operator defined in (2.2) and λ^* is any positive root of the nonincreasing function $\phi(\lambda) = \|S_\lambda(\sigma(\boldsymbol{X}))\|_1 - \lambda - \alpha$. The gradient of the smoothed posterior

log-density is

$$\begin{aligned}
\frac{\partial \log \pi^{\lambda}}{\partial \boldsymbol{X}} &= \sigma^{-2}\left[P_{\Omega}(\boldsymbol{Y}) - P_{\Omega}(\boldsymbol{X})\right] \\
&\quad -\lambda^{-1}[\boldsymbol{X} - \mathrm{prox}_{g}^{\lambda}(\boldsymbol{X}, \alpha)_{\boldsymbol{X}}], \\
\frac{\partial \log \pi^{\lambda}}{\partial \log \sigma^{2}} &= -\left(\frac{|\Omega|}{2} + s_{\sigma^2}\right) + \frac{\sum_{(i,j)\in\Omega}(y_{ij} - x_{ij})^2 + 2r_{\sigma^2}}{2\sigma^2}, \\
\frac{\partial \log \pi^{\lambda}}{\partial \log \alpha} &= -s_{\alpha} + \frac{r_{\alpha}}{\alpha} - \lambda^{-1}\alpha\left[\alpha - \mathrm{prox}_{g}^{\lambda}(\boldsymbol{X}, \alpha)_{\alpha}\right],
\end{aligned}$$

where $P_{\Omega}(\boldsymbol{Y})$ is the projection of $\boldsymbol{Y}$ onto the set of observed entries Ω, namely, the ij-th entry of $P_{\Omega}(\boldsymbol{Y})_{ij}$ is y_{ij} for $(i,j) \in \Omega$ and is zero otherwise. Thus, the difference $P_{\Omega}(\boldsymbol{Y}) - P_{\Omega}(\boldsymbol{X})$ denotes the matrix of residuals of the observed entries.

2.5.3.1 Simulated matrix

We apply our method to a simulated matrix of size 250×200. The truth is generated by $\boldsymbol{Y} = \boldsymbol{Y}_1\boldsymbol{Y}_2 + \sigma\boldsymbol{E}$ where $\boldsymbol{Y}_1$ (250×3), $\boldsymbol{Y}_2$ (3×200), and entries of $\boldsymbol{E}$ are generated from the standard normal distribution and $\sigma = 0.1$. We randomly mask 20% of the entries (9853 missing) and apply ProxMCMC to calculate 95% credible intervals for the missing entries. We use an $IG(0.01, 0.01)$ prior for σ^2 and an $IG(1, 250 \times 200 + 1)$ prior for α, and set $\lambda = 0.001$. Figure 2.8 shows the 95% credible intervals for the first 20 missing entries and we observe that the credible intervals cover the truth well. In fact, all 9853 missing entries are covered by their credible intervals.

2.5.4 Sparse low rank matrix regression

We consider linear regression with matrix covariates, where the rank of coefficient matrix is subject to regularization. One approach is to penalize the nuclear norm of the coefficient matrix [ZL14], the ProxMCMC version of which is very similar to the matrix completion example above because they share the same proximal map. Alternatively, one can constrain the coefficient matrix to have a user-specified rank k [ZLZ13]. Here we explore the second

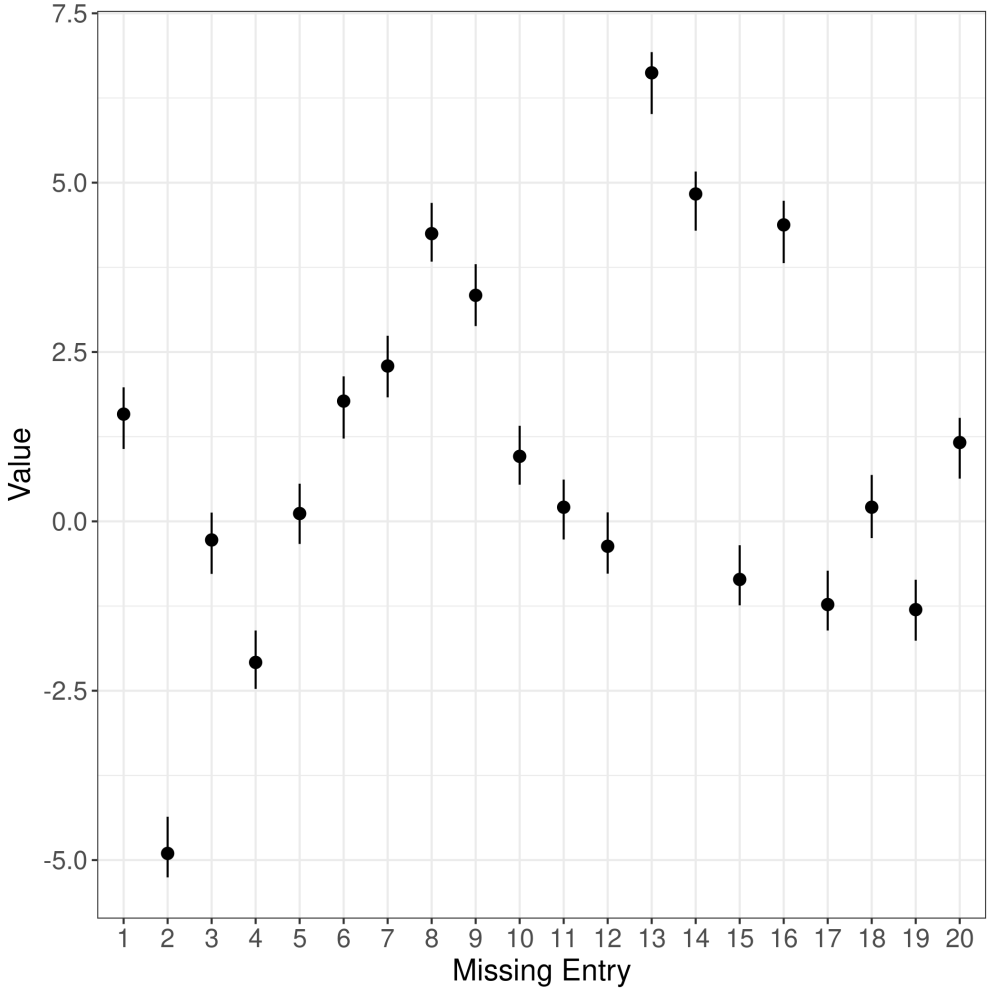


Figure 2.8: 95% credible intervals and truth (dots) for the first 20 missing entries of the simulated matrix.

approach to illustrate the potential of ProxMCMC for problems where the regularization or constraints are not convex.

Let y_i be the response for the i-th sample. Further let $\boldsymbol{Z}_i \in \mathbb{R}^p$ and $\boldsymbol{X}_i \in \mathbb{R}^{q\times r}$ be the vector and matrix covariates, respectively. The model is

$$y_i = \boldsymbol{Z}_i^T\boldsymbol{\gamma} + \langle \boldsymbol{B}, \boldsymbol{X}_i\rangle + \epsilon_i,$$

where $\boldsymbol{\gamma}$ and $\boldsymbol{B}$ are the vector and matrix coefficients, $\langle \boldsymbol{B}, \boldsymbol{X}_i\rangle = \mathrm{tr}(\boldsymbol{B}^T\boldsymbol{X}_i) = \langle \mathrm{vec}\boldsymbol{B}, \mathrm{vec}\boldsymbol{X}_i\rangle$ is the inner product of the two matrices, and $\epsilon_i \sim N(0,\sigma^2)$. We fix $\mathrm{rank}(\boldsymbol{B})$ at a user-specified value k through an explicit constraint $\delta_{\mathcal{E}_1}(\boldsymbol{B})$ where $\mathcal{E}_1 = \{\boldsymbol{B} : \mathrm{rank}(\boldsymbol{B}) = k\}$. To promote sparsity in $\boldsymbol{B}$, we also incorporate $\delta_{\mathcal{E}_2}(\boldsymbol{B},\alpha)$, where $\mathcal{E}_2 = \{(\boldsymbol{B},\alpha) : \|\mathrm{vec}\boldsymbol{B}\|_1 \le \alpha\}$. With a flat prior on $\boldsymbol{\gamma}$ (i.e., $\pi(\boldsymbol{\gamma}) \propto 1$), an $IG(r_{\sigma^2}, s_{\sigma^2})$ prior for σ^2, and an $IG(r_\alpha, s_\alpha)$ prior for α, the smoothed posterior log-density is

$$\begin{aligned}
&\log\pi(\gamma, \boldsymbol{B}, \log\sigma^2, \log\alpha)\\
=&-\frac{\sum_{i=1}^n (y_i - \boldsymbol{Z}_i^T\gamma - \langle \boldsymbol{B}, \boldsymbol{X}_i\rangle)^2 + 2r_{\sigma^2}}{2\sigma^2}\\
&-(\frac{n}{2} + s_{\sigma^2})\log\sigma^2 - s_\alpha\log\alpha - \frac{r_\alpha}{\alpha}\\
&-g_1^\lambda(\boldsymbol{B}) - g_2^\lambda(\boldsymbol{B},\alpha),
\end{aligned}$$

where $g_1^\lambda(\boldsymbol{B})$ and $g_2^\lambda(\boldsymbol{B},\alpha)$ are the Moreau-Yosida envelopes of $g_1(\boldsymbol{B}) = \delta_{\mathcal{E}_1}(\boldsymbol{B})$ and $g_2(\boldsymbol{B},\alpha) = \delta_{\mathcal{E}_2}(\boldsymbol{B},\alpha)$. The proximal map of $g_1(\boldsymbol{B})$ is the projection onto the set $\mathcal{E}_1$ obtained by thresholding the singular values of $\boldsymbol{B}$. However, since $g_1^\lambda(\boldsymbol{B})$ is non-convex, the gradient formula (2.1) for Moreau-Yosida envelope no longer holds. Instead, we resort to the subsmoothness property of Moreau-Yosida envelopes, for which we need the following definitions [RW09].

Definition 3. ***(Prox-boundedness)*** *A function $g : \mathbb{R}^n \to \bar{\mathbb{R}}$ is prox-bounded if there exists $\lambda > 0$ such that its Moreau-Yosida envelope $g^\lambda > -\infty$ for some $\boldsymbol{x} \in \mathbb{R}^n$. The supremum of the set of all such λ is the threshold λ_g of prox-boundedness for g.*

In the ProxMCMC framework, we only need the Moreau-Yosida envelope of indicator

functions, for which we have $g^\lambda(\boldsymbol{x}) > -\infty$ for any $\lambda > 0$, so they are always prox-bounded and the threshold $\lambda_g = \infty$.

Definition 4. ***(Semidifferentiability)*** *Let $g : \mathbb{R}^n \to \bar{\mathbb{R}}$ and $\bar{\boldsymbol{x}}$ be a point such that $g(\bar{\boldsymbol{x}})$ is finite. If the (possibly infinite) limit*

$$\lim_{\tau \downarrow 0, \boldsymbol{w}' \to \boldsymbol{w}} \frac{g(\bar{\boldsymbol{x}} + \tau \boldsymbol{w}') - g(\bar{\boldsymbol{x}})}{\tau}$$

exists, it is the semiderivative of g at $\bar{\boldsymbol{x}}$ for $\boldsymbol{w}$, and g is semidifferentiable at $\bar{\boldsymbol{x}}$ for $\boldsymbol{w}$. If this holds for every $\boldsymbol{w}$, g is semidifferentiable at $\bar{\boldsymbol{x}}$.

By [RW09, Example 10.32], if $g(\boldsymbol{x})$ is lower-semicontinuous, proper, and prox-bounded with threshold λ_g, then for $\lambda \in (0, \lambda_g)$, the Moreau-Yosida envelope $g^\lambda(\boldsymbol{x})$ is semidifferentiable and the subgradient set is

$$\partial g^\lambda(\boldsymbol{x}) \subset \lambda^{-1} \left[\boldsymbol{x} - \operatorname{prox}_g^\lambda(\boldsymbol{x}) \right].$$

The function $g_1(\boldsymbol{B}) = \delta_{\mathcal{E}_1}(\boldsymbol{B})$ satisfies the above conditions, so we can calculate its subgradient using the above formula and use it in place of the gradient in HMC

$$\begin{aligned}
\frac{\partial \log \pi}{\partial \boldsymbol{\gamma}} &= \sigma^{-2} \sum_i (y_i - \boldsymbol{Z}_i^T \boldsymbol{\gamma} - \langle \boldsymbol{B}, \boldsymbol{X}_i \rangle) Z_i, \\
\frac{\partial \log \pi}{\partial \boldsymbol{B}} &= \sigma^{-2} \sum_i (y_i - \boldsymbol{Z}_i^T \boldsymbol{\gamma} - \langle \boldsymbol{B}, \boldsymbol{X}_i \rangle) X_i \\
&\quad -\lambda^{-1} \left[\boldsymbol{B} - \operatorname{prox}_{g_1}^\lambda(\boldsymbol{B}) \right] \\
&\quad -\lambda^{-1} \left[\boldsymbol{B} - \operatorname{prox}_{g_2}^\lambda(\boldsymbol{B}, \alpha)_{\boldsymbol{B}} \right], \\
\frac{\partial \log \pi}{\partial \log \sigma^2} &= -\left(\frac{n}{2} + s_{\sigma^2} \right) \\
&\quad + \frac{\sum_{i=1}^n (y_i - \boldsymbol{Z}_i^T \boldsymbol{\gamma} - \langle \boldsymbol{B}, \boldsymbol{X}_i \rangle)^2 + 2 r_{\sigma^2}}{2\sigma^2}, \\
\frac{\partial \log \pi}{\partial \log \alpha} &= -s_\alpha + \frac{r_\alpha}{\alpha} - \lambda^{-1} \alpha \left[\alpha - \operatorname{prox}_{g_2}^\lambda(\boldsymbol{B}, \alpha)_\alpha \right].
\end{aligned}$$

Since $g_1^\lambda(\boldsymbol{B})$ is non-convex, $\operatorname{prox}_{g_1}^\lambda(\boldsymbol{B})$ is not unique. Our approach is to pick an arbitrary element in the proximal map set, which works well in practice.

2.5.4.1 Detecting the cross-shaped signal

We illustrate the method on a simulated data of cross-shaped signal. The mean responses are $\mu_i = \boldsymbol{Z}_i^T\boldsymbol{\gamma} + \langle \boldsymbol{B}, \boldsymbol{X}_i \rangle$, where $\boldsymbol{Z}_i \in \mathbb{R}^2$ and $\boldsymbol{X}_i \in \mathbb{R}^{16\times 16}$ and their entries are generated from i.i.d. standard normal. We set $\boldsymbol{\gamma} = (1,1)^T$ and $\boldsymbol{B}$ to have a cross shape (Figure 2.12(a)), where the white cross entries equal 1 and the rest 0. The response y_i equals $\mu_i + \epsilon_i$, where ϵ_i ($i = 1, ..., n$ and $n = 100$) are also generated from independent standard normal. We use an $IG(0.01, 0.01)$ prior for σ^2 and an $IG(\sum_i \sigma(\boldsymbol{B}_0)_i, 2)$ prior for α, where $\sigma(\boldsymbol{B}_0)_i$ is the i-th singular value of $\boldsymbol{B}_0$ and $\boldsymbol{B}_0$ is the initial estimate of $\boldsymbol{B}$ obtained by least squares without regularization or constraints. We set the Moreau-Yosida envelope parameter $\lambda = 0.001$. Figure 2.12 shows the true signal $\boldsymbol{B}$ (panel (a)), the posterior mean from 10,000 HMC samples (panel (b)), and the standard error (panel (c)). Due to space limitations, we only show the 95% credible intervals of the 8-th column of $\boldsymbol{B}$ (Figure 2.13), but all entries of $\boldsymbol{B}$ are covered by their 95% credible intervals.

2.6 Theoretical properties

This section presents theoretical results for the ProxMCMC method. The proofs simplify those in [DMP18] because we focus on the Moreau-Yosida envelope of indicator functions. Our proofs, however, extend to non-convex settings while [DMP18] assume convexity. As defined in Section 2.4, $\boldsymbol{\theta} \in \mathbb{R}^d$ represents all model parameters that include both the constrained or regularized parameters $\boldsymbol{\tau} \in \mathbb{R}^p$ and other parameters $\boldsymbol{\eta} \in \mathbb{R}^q$. We also use $\ell(\boldsymbol{\theta})$ for the log-likelihood and $\pi(\boldsymbol{\eta})$ for the prior density of $\boldsymbol{\eta}$. Our main theoretical results are summarized as follows:

Proposition 1. *(1) For any $\lambda > 0$, the smoothed posterior $\pi^\lambda(\boldsymbol{\theta} \mid Y)$ defines a proper density of a probability measure on $\mathbb{R}^d$, so*

$$0 < \int_{\mathbb{R}^d} e^{-U^\lambda(\boldsymbol{\theta})}\, d\boldsymbol{\theta} < \infty.$$

(2) The approximation $\pi^\lambda(\boldsymbol{\theta} \mid Y)$ converges to $\pi(\boldsymbol{\theta} \mid Y)$ in total-variation as $\lambda \downarrow 0$, i.e.,

$$\lim_{\lambda \downarrow 0} \|\pi^\lambda(\boldsymbol{\theta} \mid Y) - \pi(\boldsymbol{\theta} \mid Y)\|_{TV} = 0.$$

Proof. (Posterior properness) The properness of the smoothed posterior $\pi^\lambda(\boldsymbol{\theta} \mid Y)$ follows from the fact that the Moreau-Yosida envelope of an indicator function is always nonnegative. Specifically, when $g = \delta_{\mathcal{E}}(\boldsymbol{\tau})$,

$$g^\lambda(\boldsymbol{\tau}) = \frac{1}{2\lambda} d_{\mathcal{E}}(\boldsymbol{\tau})^2 \geq 0,$$

where $d_{\mathcal{E}}(\boldsymbol{\tau}) = \inf_{\boldsymbol{y} \in \mathcal{E}} d(\boldsymbol{\tau}, \boldsymbol{y})$ is the distance from $\boldsymbol{\tau}$ to $\mathcal{E}$, so $-U^\lambda(\boldsymbol{\theta}) = -f(\boldsymbol{\theta}) - g^\lambda(\boldsymbol{\tau}) \leq -f(\boldsymbol{\theta})$, from which we have

$$e^{-U^\lambda(\boldsymbol{\theta})} \leq e^{-f(\boldsymbol{\theta})}.$$

Since $f(\boldsymbol{\theta}) = -\ell(\boldsymbol{\theta}) - \log \pi(\boldsymbol{\eta})$ and both the likelihood and the priors $\pi(\boldsymbol{\eta})$ are integrable (note that $\boldsymbol{\eta}$ does not include constrained parameters), we have the desired result.

(Convergence in total-variation) Let $c = \int e^{-U(\boldsymbol{s})}\, d\boldsymbol{s}$ and $c_\lambda = \int e^{-U^\lambda(\boldsymbol{s})}\, d\boldsymbol{s}$. Since $g^\lambda(\boldsymbol{x})$ uniformly bounds $g(\boldsymbol{x})$ from below, i.e., $g^\lambda(\boldsymbol{x}) \leq g(\boldsymbol{x})$ for all $\boldsymbol{x}$ [RW09], we have $U^\lambda(\boldsymbol{x}) \leq$

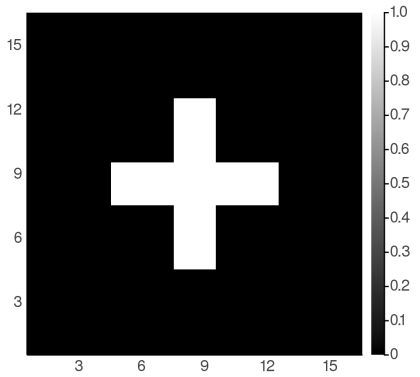


Figure 2.9: True Signal

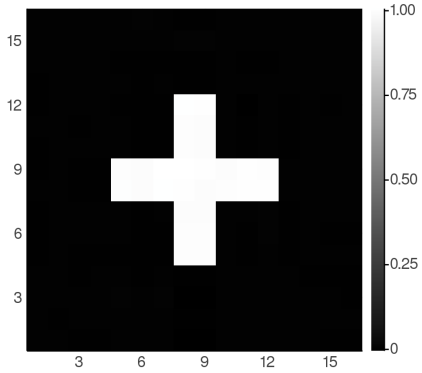


Figure 2.10: Posterior Mean

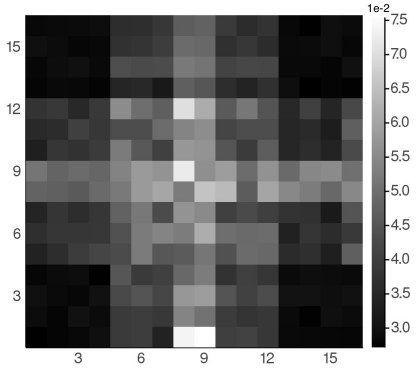


Figure 2.11: Standard Error

Figure 2.12: Proximal MCMC for sparse low rank matrix regression on the cross-shaped data.

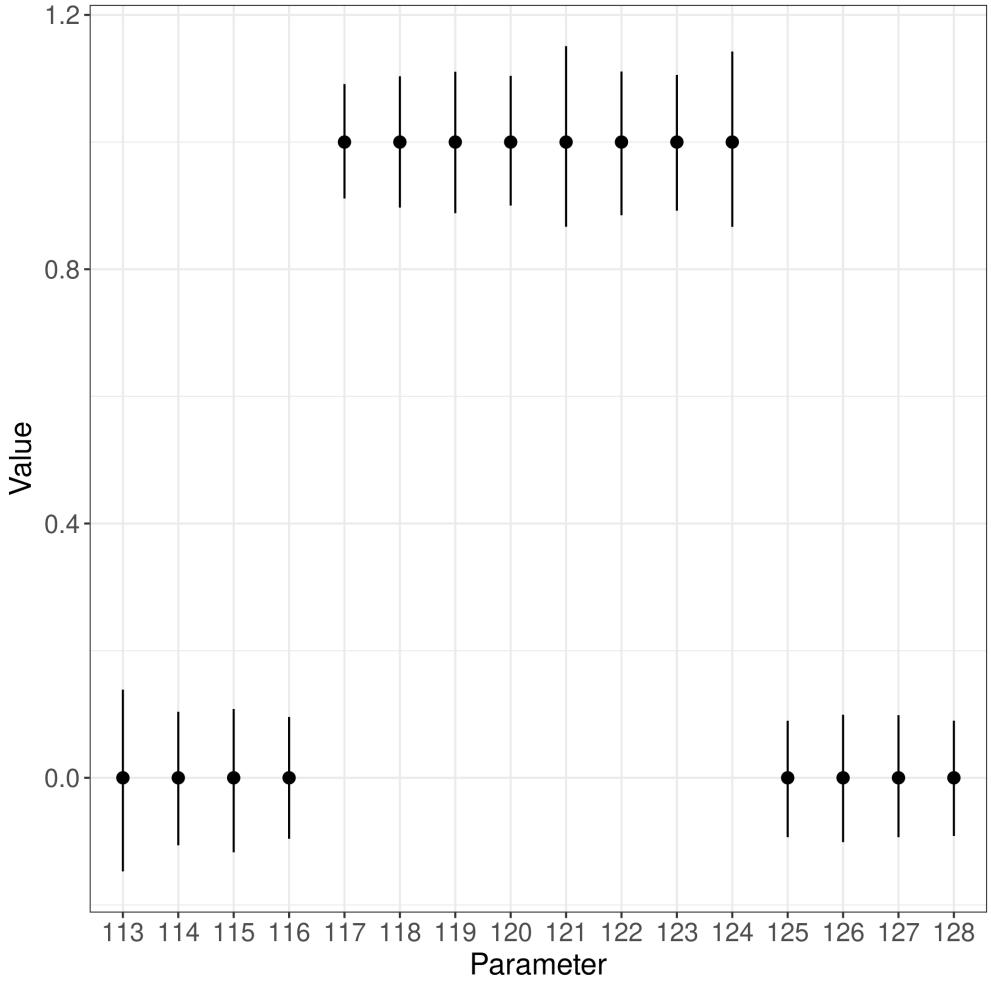


Figure 2.13: 95% credible intervals of the eighth column of the cross-shaped signal. X-axis indicates their position in vec($\boldsymbol{B}$). Dots mark the truth.

$U(\boldsymbol{x})$ and thus $c_\lambda \geq c$. Note that

$$\begin{aligned}
&\|\pi^\lambda - \pi\|_{\mathrm{TV}} \\
&= \int |\pi^\lambda(\boldsymbol{x}) - \pi(\boldsymbol{x})| \, d\boldsymbol{x} \\
&= \int_{\pi^\lambda \geq \pi} \left[\pi^\lambda(\boldsymbol{x}) - \pi(\boldsymbol{x})\right] \, d\boldsymbol{x} \\
&\quad + \int_{\pi^\lambda < \pi} \left[\pi(\boldsymbol{x}) - \pi^\lambda(\boldsymbol{x})\right] \, d\boldsymbol{x}.
\end{aligned}$$

Let $\mathcal{A}_1 = \left\{\boldsymbol{x} : \pi^\lambda \geq \pi\right\}$ and $\mathcal{A}_2 = \left\{\boldsymbol{x} : \pi^\lambda < \pi\right\}$,

$$\begin{aligned}
&\int_{\mathcal{A}_1} \left[\pi^\lambda(\boldsymbol{x}) - \pi(\boldsymbol{x})\right] \, d\boldsymbol{x} \\
&= \int_{\mathcal{A}_1} \pi^\lambda(\boldsymbol{x}) \left[1 - \frac{\pi(\boldsymbol{x})}{\pi^\lambda(\boldsymbol{x})}\right] \, d\boldsymbol{x} \\
&= \int_{\mathcal{A}_1} \pi^\lambda(\boldsymbol{x}) \left[1 - \frac{c_\lambda}{c} e^{g^\lambda(\boldsymbol{x}) - g(\boldsymbol{x})}\right] \, d\boldsymbol{x} \\
&\leq \int_{\mathcal{A}_1} \left[\pi^\lambda(\boldsymbol{x}) - e^{g^\lambda(\boldsymbol{x}) - g(\boldsymbol{x})} \pi^\lambda(\boldsymbol{x})\right] \, d\boldsymbol{x} \\
&\leq 1 - \frac{c}{c_\lambda},
\end{aligned}$$

and

$$\begin{aligned}
&\int_{\mathcal{A}_2} \left[\pi(\boldsymbol{x}) - \pi^\lambda(\boldsymbol{x})\right] \, d\boldsymbol{x} \\
&= \int_{\mathcal{A}_2} \pi(\boldsymbol{x}) \left[1 - \frac{\pi^\lambda(\boldsymbol{x})}{\pi(\boldsymbol{x})}\right] \, d\boldsymbol{x} \\
&= \int_{\mathcal{A}_2} \pi(\boldsymbol{x}) \left[1 - \frac{c}{c_\lambda} e^{g(\boldsymbol{x}) - g^\lambda(\boldsymbol{x})}\right] \, d\boldsymbol{x} \\
&\leq \int_{\mathcal{A}_2} \pi(\boldsymbol{x}) \left[1 - \frac{c}{c_\lambda}\right] \, d\boldsymbol{x} \\
&\leq 1 - \frac{c}{c_\lambda}.
\end{aligned}$$

So $\|\pi^\lambda - \pi\|_{\mathrm{TV}} \leq 2(1 - \frac{c}{c_\lambda})$. By [RW09], when $g(\boldsymbol{x})$ is proper, lower-semicontinuous, and prox-bounded with threshold $\lambda_g > 0$, $g^\lambda(\boldsymbol{x})$ converges pointwise to $g(\boldsymbol{x})$ as $\lambda \downarrow 0$. Moreover, since $g^\lambda(\boldsymbol{x})$ is pointwise non-decreasing as λ decreases, by the monotone convergence theorem,

$\lim_{\lambda \downarrow 0} c_\lambda = c$. Thus

$$\lim_{\lambda \downarrow 0} \|\pi^\lambda - \pi\|_{\mathrm{TV}} \leq \lim_{\lambda \downarrow 0} 2\left(1 - \frac{c}{c_\lambda}\right) = 0.$$

□

We remark that convergence in total-variation only holds when $c \neq 0$, which is violated in the presence of equality constraints. Thus the constrained lasso and sparse low rank matrix regression examples do not enjoy this property.

2.7 Discussion

Our examples demonstrate that the ProxMCMC method is a highly flexible tool for performing statistical inference on regularized or constrained statistical learning problems. We find that it works well when the regularization or constraints are non-smooth and even non-convex. In addition, by adopting epigraph priors, our method is fully Bayesian, eliminating the need for tuning the regularization parameter.

For the Moreau-Yosida envelope parameter λ, a smaller value leads to better satisfaction of the constraints. For example, the histogram of $\sum_j \boldsymbol{\beta}_j$ from the microbiome example is more concentrated around 0 when λ is smaller. Extremely small λ, however, renders slow mixing of the sampling algorithm. We recommend using smaller λ when computational resources allow. Setting $\lambda = 0.001$ works in most applications as the examples show.

Finally, we emphasize that the four examples are meant to whet readers' appetites, not to satiate them. As we mentioned before, the ProxMCMC algorithm is highly modular and can be readily extended to other problems. We hope this work encourages readers to discover new applications of the ProxMCMC algorithm.

CHAPTER 3

Improved Estimation Equations for Semiparametric Censored Linear Regressions

3.1 Introduction

This work aims to address two problems that arise frequently in epidemiological and genetic studies. The first one is how to estimate associations between covariates and an outcome whose values are likely altered through the use of medications. For example, patients on anti-hypertensive medications are likely to have lower blood pressure measurements. Other examples include diabetic patients on glucose-lowering medications and hyperlipidemic patients on lipid-lowering medications. For patients on medications, we only observe the treated outcome. However, in many cases the scientific interests lie in the "untreated" outcome. For example, for studying associations with covariates such as gender and genetics, which are present before medication use, the "untreated" outcome is clearly of interest. Several publications have shown that if we simply ignore medication use, exclude those on medication, or adjust for medication use in regression models, the results would be invalid in most practical situations [WCM94, Coo97, WKC03, TSS05]. [MKH08] proposes an alternative approach of parametrically imputing the "untreated" value as a function of the observed treated value, dose, and type of medication, and then accounting for the variability induced by estimation through multiple imputation. While this method improves upon more naive approaches, it is more difficult to use because it requires two steps and may pose computational difficulties for large data sets.

The second problem is how to perform variance quantitative trait loci (vQTL) analysis on right-censored traits. The medication-use example can be considered right-censored and thus falls into this category. Other examples include time to event outcomes such as time to cardiovascular disease in diabetic patients [BG07]. The importance of vQTL analysis is highlighted by its direct applicability to inferring gene-by-environment interactions (GEI) [YLP12, WZZ19, WMG22], which is a fundamental component in understanding complex trait variation, and yet is challenging to identify due to the difficulty of measuring environmental exposures. The advantage of vQTL is that it works even if we do not have environmental exposure data. In fact, we do not even need a target environmental factor. For quantitative traits that are not subject to censoring, [RV12] provides a good summary of both parametric [Bar37, BF74, FK76] and nonparametric methods [CWB14, RFF10, RV11, Smy89] for detecting vQTL. For quantitative traits that are right-censored, however, we are not aware of any methods or software packages that can perform vQTL analysis.

To address these problems, we adopt the synthetic variable approach proposed by [KSV81], [Leu87], and [Zhe08]. The synthetic variable approach for censored linear regression enjoys simplicity and robustness, but in practice may suffer from low estimation efficiency. To improve efficiency, [LL09] proposes a weighted least squares (WLS) method which estimates the conditional variance of the synthetic data nonparametrically, and then applies the standard WLS principle in the estimation procedure. This work extends the previous one in two directions. First, we derive the second moment synthetic variables and use them to construct a quadratic estimation equation (QEE) for modeling the potentially heterogeneous noise variances among subjects. Second, using the initial estimate of $\boldsymbol{\beta}$ from classical synthetic variable approach and σ^2 from QEE, we compute the *working* variances of synthetic variables, which are then used to construct more efficient estimation equations. This procedure is iterated to obtain more accurate estimates.

The next section gives a detailed account of our methodology, followed by preliminary simulation results.

3.2 Method

Consider the linear model

$$y_i = \boldsymbol{x}_i^T \boldsymbol{\beta} + \epsilon_i, \ \ i = 1, ..., n \tag{3.1}$$

where $\boldsymbol{\beta} \in \mathbb{R}^p$ denotes the regression parameters for the mean, ϵ_i's are independent with mean zero and variance σ_i^2, and the responses y_i are subject to potential censoring. In the right-censoring case, the observed data is $(\tilde{y}_i, \delta_i)$, $i = 1, \ldots, n$, where $\tilde{y}_i = \min\{y_i, c_i\}$ is the observed value and $\delta_i = I(y_i \leq c_i)$ is the censoring indicator. We assume the distribution G of the censoring times c_i is known (will be relaxed later) and independent of y_i and $\boldsymbol{x}_i$. The variance of y_i is further modeled by

$$\text{Var} y_i = \sigma_i^2 = g(\boldsymbol{\tau}, \boldsymbol{w}_i), \tag{3.2}$$

where $\boldsymbol{\tau} \in \mathbb{R}^q$ denotes the regression coefficients for the variance and $\boldsymbol{W} = [\boldsymbol{w}_1, ..., \boldsymbol{w}_n]^T$ is an $n \times q$ design matrix with the first column being a vector of ones. The function g can take on forms such as $g(\boldsymbol{\tau}, \boldsymbol{w}_i) = \exp(\boldsymbol{w}_i^T \boldsymbol{\tau})$.

There is a long history on estimating the unknown regression parameters $\boldsymbol{\beta}$ using synthetic variables. The idea is to construct surrogate responses y_i^* that are unbiased for estimating y_i, i.e., $\mathbb{E}\, y_i^* = \mathbb{E}\, y_i$, and then perform regression analysis using the surrogate responses. [KSV81] proposes the inverse probability-weighted synthetic variables

$$y_i^* = \frac{\delta_i \tilde{y}_i}{1 - G(\tilde{y}_i -)}, \tag{3.3}$$

which we refer to as the KSvR synthetic variable in the rest of the chapter and may be the most widely used in literature. [Leu87] proposes an improved synthetic variable

$$y_i^* = \int_0^{\tilde{y}_i} \frac{1}{1 - G(t-)}\, dt. \tag{3.4}$$

Independent of [Leu87], [Zhe87] proposes a general framework

$$y_i^* = \delta_i \varphi_1(\tilde{y}_i) + (1 - \delta_i)\varphi_2(\tilde{y}_i), \tag{3.5}$$

which includes (3.3) and (3.4) as special cases. The functions φ_i, $i = 1, 2$, are chosen to satisfy $\mathbb{E}(y^* \mid y) = y$ almost surely. [Zhe87] shows that the Leurgans synthetic variables (3.4) have smaller variance than (3.3). Asymptotic normality of coefficient estimate $\widehat{\boldsymbol{\beta}}$ is established in [Zho92, LYZ95].

[BJ79] proposes an iterative procedure based on

$$y_i^* \;=\; \delta_i \tilde{y}_i + (1-\delta_i)\left[\boldsymbol{x}_i^T\boldsymbol{\beta} + \frac{\int_{(\tilde{y}_i - \boldsymbol{x}_i^T\boldsymbol{\beta})+}^{\infty} s\, dF(s)}{1 - F(\tilde{y}_i - \boldsymbol{x}_i^T\boldsymbol{\beta})}\right]$$

where F is the assumed distribution function for ϵ_i. Note that $\mathbb{E}[\mathbb{E}(y_i \mid \delta_i, \tilde{y}_i, \boldsymbol{x}_i) \mid \boldsymbol{x}_i] = \mathbb{E}(y_i \mid \boldsymbol{x}_i) = \boldsymbol{x}_i^T\boldsymbol{\beta}$ since we assume $c_i \perp y_i \mid \boldsymbol{x}_i$ and $\epsilon_i \perp \boldsymbol{x}_i$. When F is unknown, we replace it by the Kaplan-Meier estimate $\widehat{F}$ based on $(\tilde{y}_i - \boldsymbol{x}_i^T\boldsymbol{\beta}, \delta_i)$. Compared to the synthetic variable approach, Buckley-James estimator only requires conditional independence $c_i \perp y_i \mid \boldsymbol{x}_i$ and thus is more reliable in real applications [MH82]. However, the Buckley-James procedure may not converge to a consistent root of the estimation equation and needs to be carefully initialized from a consistent estimate [JLY06]. Furthermore, the Buckley-James procedure cannot handle left- and interval-censored data and generalization to more complex models such as heterogeneous variances is not straightforward.

3.2.1 M-Estimators

To estimate model parameters from (3.1) and (3.2), we propose M-estimators of the form

$$\widehat{\boldsymbol{\beta}} \;=\; \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} w_{i1}(y_{i1}^* - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 \tag{3.6}$$

$$\widehat{\boldsymbol{\tau}} \;=\; \arg\min_{\boldsymbol{\tau}} \sum_{i=1}^{n} w_{i2}[\epsilon_{i2}^* - g(\boldsymbol{\tau}, \boldsymbol{w}_i)]^2, \tag{3.7}$$

where y_{i1}^* and ϵ_{i2}^* are the synthetic variables satisfying $\mathbb{E}\, y_{i1}^* = \mathbb{E}\, y_i$ and $\mathbb{E}\, \epsilon_{i2}^* = \mathbb{E}\, \epsilon_i^2$, respectively. The weights $w_{i1} > 0$ and $w_{i2} > 0$ are inverse of the working variances of y_{i1}^* and ϵ_{i2}^*. To use (3.6) and (3.7), we start with uninformative weights $w_{i1}^2 \equiv w_{i2}^2 \equiv 1$. Equation (3.6) reduces to the classical synthetic variable approach, which produces an initial estimator $\widehat{\boldsymbol{\beta}}^{(1)}$

and fitted values $\widehat{\eta}_{i1}^{(1)} = \boldsymbol{x}_i^T \widehat{\boldsymbol{\beta}}^{(1)}$. By calculating the residuals $\epsilon_i = \tilde{y}_i - \widehat{\eta}_{i1}^{(1)}$ and its second moment synthetic variables ϵ_{i2}^*, we can obtain an initial estimate $\widehat{\boldsymbol{\tau}}^{(1)}$ from (3.7) through nonlinear least squares.

From here, we would like to calculate the *working* variances of synthetic variables y_{i1}^* and ϵ_{i2}^*, and set weights w_{i1} and w_{i2} to be their inverses, which are used in estimation equations (3.6) and (3.7) for improved efficiency. But to calculate the *working* variances, we first need to estimate the distribution of ϵ_i. Since $\epsilon_1, ..., \epsilon_n$ are not identically distributed, we cannot directly use the Kaplan-Meier estimator (KME). Instead, let $\epsilon_{0i} = \epsilon_i / \sqrt{g(\boldsymbol{\tau}, \boldsymbol{w}_i)}$, then $\text{Var}\epsilon_{0i} = 1$ and $\epsilon_{01}, ..., \epsilon_{0n}$ are independent and identically distributed, and have a common distribution function $F_0(t) = P(\epsilon_{0i} \leq t)$ that can be estimated using KME.

Our estimation strategy can be iterated. We will use superscript, $\widehat{\boldsymbol{\beta}}^{(k)}$ and $\widehat{\boldsymbol{\tau}}^{(k)}$, to indicate the estimates after k rounds of estimation. In the next sub-section, we derive synthetic variables y_{i1}^*, ϵ_{i2}^*, and their working variances.

3.2.2 Synthetic variables

To enable the estimation procedure, we construct both first and second moment synthetic variables and derive explicit expressions for their variances. We assume $y_i \in \mathbb{R}$, in contrast to most papers assuming $y_i > 0$. We follow the [Zhe87] framework (3.5) because of its generality. The subscripts $_1$ and $_2$ in y_{i1}^* and ϵ_{i2}^* are used to differentiate between the first and second moment synthetic variables.

For the first moment synthetic variable based on the observed response, let

$$y_{i1}^* = \delta_i \varphi_1(\tilde{y}_i) + (1 - \delta_i)\varphi_2(\tilde{y}_i). \tag{3.8}$$

The condition

$$\begin{aligned} \mathbb{E}(y_{i1}^* \mid y_i) &= \mathbb{E}_G[\delta_i \varphi_1(\tilde{y}_i) + (1 - \delta_i)\varphi_2(\tilde{y}_i) \mid y_i] \\ &= [1 - G(y_i-)]\varphi_1(y_i) + \int_{-\infty}^{y_i-} \varphi_2(t)\, dG(t) = y_i \end{aligned} \tag{3.9}$$

for all y_i ensures the unbiasedness $\mathbb{E}\, y_{i1}^* = \mathbb{E}\,\mathbb{E}(y_{i1}^* \mid y_i) = \mathbb{E}\, y_i$. Assuming G is continuous with density g and differentiability of φ_1, condition (3.9) can be written as a differential equation

$$[1 - G(y_i-)]\varphi_1'(y_i) - g(y_i-)\varphi_1(y_i) + g(y_i-)\varphi_2(y_i-) = 1. \tag{3.10}$$

The variance of synthetic variable (3.8) is

$$\mathrm{Var} y_{i1}^* \;\; = \;\; \mathbb{E}(y_{i1}^* - \mathbb{E}\, y_i)^2 = \mathbb{E}(y_{i1}^* - y_i + y_i - \mathbb{E}\, y_i)^2 = \mathrm{Var} y_i + \mathbb{E}(y_{i1}^* - y_i)^2,$$

where the cross term vanishes because

$$\mathbb{E}(y_{i1}^* - y_i)(y_i - \mathbb{E}\, y_i) = \mathbb{E}\,\mathbb{E}[(y_{i1}^* - y_i)(y_i - \mathbb{E}\, y_i) \mid y_i] = \mathbb{E}\{(y_i - \mathbb{E}\, y_i)\,\mathbb{E}[(y_{i1}^* - y_i) \mid y_i]\} = 0.$$

This expression shows y_{i1}^* always have larger variance than y_i; this is the price we pay without observing all y_i.

For the second moment synthetic variable based on residuals

$$\epsilon_{i2}^* = \delta_i \varphi_1(\tilde{\epsilon}_i) + (1 - \delta_i)\varphi_2(\tilde{\epsilon}_i), \tag{3.11}$$

since

$$\begin{aligned}
\mathbb{E}(\delta_i \phi_1(\tilde{\epsilon}_i) \mid \epsilon_i) &= \mathbb{E}_G[\delta_i \phi_1(\epsilon_i) \mid \epsilon_i] \\
&= \phi_1(\epsilon_i) \int I_{\{y_i - \boldsymbol{x}_i^T\boldsymbol{\beta} \le t - \boldsymbol{x}_i^T\boldsymbol{\beta}\}} dG(t) \\
&= \phi_1(\epsilon_i)[1 - G(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)], \\
\mathbb{E}[(1 - \delta_i)\phi_2(\tilde{\epsilon}_i) \mid \epsilon_i] &= \mathbb{E}_G[(1 - \delta_i)\phi_2(c_i - \boldsymbol{x}_i^T\boldsymbol{\beta}) \mid \epsilon_i] \\
&= \int_{-\infty}^{\infty} I_{\{t - \boldsymbol{x}_i^T\boldsymbol{\beta} < \epsilon_i\}} \phi_2(t - \boldsymbol{x}_i^T\boldsymbol{\beta}) dG(t) \\
&= \int_{-\infty}^{\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-} \phi_2(t - \boldsymbol{x}_i^T\boldsymbol{\beta}) dG(t),
\end{aligned}$$

so

$$\mathbb{E}[\delta_i \phi_1(\tilde{\epsilon}_i) + (1 - \delta_i)\phi_2(\tilde{\epsilon}_i) \mid \epsilon_i] = \phi_1(\epsilon_i)[1 - G(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)] + \int_{-\infty}^{\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-} \phi_2(t - \boldsymbol{x}_i^T\boldsymbol{\beta}) dG(t).$$

The unbiasedness condition $\mathbb{E}(\epsilon_{i2}^* \mid \epsilon_i) = \epsilon_i^2$ dictates that

$$\phi_1(\epsilon_i)[1 - G(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)] + \int_{-\infty}^{\epsilon_i+\boldsymbol{x}_i^T\boldsymbol{\beta}-} \phi_2(t - \boldsymbol{x}_i^T\boldsymbol{\beta})dG(t) = \epsilon_i^2$$

for all ϵ_i, which leads to the differential equation

$$\phi_1'(\epsilon_i)[1 - G(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)] - \phi_1(\epsilon_i)g(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-) + \phi_2(\epsilon_i-)g(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-) = 2\epsilon_i.$$

The variance of the second moment synthetic variable ϵ_{i2}^* is

$$\mathrm{Var}\epsilon_{i2}^* = \mathrm{Var}\epsilon_i^2 + \mathbb{E}(\epsilon_{i2}^* - \epsilon_i^2)^2 = \mathbb{E}(\epsilon_{i2}^*)^2 - (\mathbb{E}\,\epsilon_i^2)^2.$$

Since $(\epsilon_{i2}^*)^2 = \delta_i\phi_1^2(\tilde{\epsilon}_i) + (1-\delta_i)\phi_2^2(\tilde{\epsilon}_i)$,

$$\begin{aligned}
\mathbb{E}\,\delta_i\phi_1^2(\tilde{\epsilon}_i) &= \mathbb{E}\left[\phi_1^2(\epsilon_i)\int I_{\{y_i-\boldsymbol{x}_i^T\boldsymbol{\beta}\le t-\boldsymbol{x}_i^T\boldsymbol{\beta}\}}dG(t)\right] = \mathbb{E}\,\phi_1^2(\epsilon_i)[1 - G(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)] \\
\mathbb{E}(1-\delta_i)\phi_2^2(\tilde{\epsilon}_i) &= \int_{\mathbb{R}} \phi_2^2(t - \boldsymbol{x}_i^T\boldsymbol{\beta}) \int_{t-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{\infty} dF_i(s)dG(t) \\
&= \int_{\mathbb{R}} \phi_2^2(t - \boldsymbol{x}_i^T\boldsymbol{\beta})(1 - F_i(t - \boldsymbol{x}_i^T\boldsymbol{\beta}))dG(t),
\end{aligned}$$

so

$$\mathrm{Var}\epsilon_{i2}^* = \mathbb{E}\,\phi_1^2(\epsilon_i)[1 - G(\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta})] + \int_{\mathbb{R}} \phi_2^2(t - \boldsymbol{x}_i^T\boldsymbol{\beta})(1 - F_i(t - \boldsymbol{x}_i^T\boldsymbol{\beta}))dG(t) - (\mathbb{E}\,\epsilon_i^2)^2.$$

Next we focus on two specific types of synthetic variables that are commonly used in practice.

3.2.2.1 KSvR

If we choose $\varphi_2 \equiv 0$, then the unbiasedness condition (3.9) dictates $\varphi_1(\tilde{y}_i) = \tilde{y}_i/[1-G(\tilde{y}_i-)]$. The variance of the synthetic variable $y_{i1}^* = \delta_i\tilde{y}_i/[1 - G(\tilde{y}_i-)]$ is

$$\begin{aligned}
\mathrm{Var}y_{i1}^* &= \mathbb{E}\,y_i^2[1 - G(y_i-)]^{-1} - (\mathbb{E}\,y_i)^2 \\
&= \mathrm{Var}y_i + \mathbb{E}\,\frac{G(y_i-)}{1 - G(y_i-)}y_i^2 \\
&= \mathrm{Var}y_i + \int \frac{G(\boldsymbol{x}_i^T\boldsymbol{\beta} + s-)}{1 - G(\boldsymbol{x}_i^T\boldsymbol{\beta} + s-)}(\boldsymbol{x}_i^T\boldsymbol{\beta} + s)^2 dF_i(s).
\end{aligned}$$

where $F_i(s) = P(\epsilon_i \le s)$ is the distribution function of ϵ_i. Due to heteroscedasticity, $\epsilon_1, ..., \epsilon_n$ are independent but not identically distributed. To estimate $F_i(s)$, we first calculate $\epsilon_{0i} = \epsilon_i / \sqrt{g(\boldsymbol{\tau}, \boldsymbol{w}_i)}$ so that $\epsilon_{01}, ..., \epsilon_{0n}$ are independent and identically distributed. Then we can estimate the distribution function $F_0(s_0)$ of ϵ_{0i} using the Kaplan-Meier estimator. Since s and s_0 are related through $s = s_0\sqrt{g(\boldsymbol{\tau}, \boldsymbol{w}_i)}$, we have

$$F_i(s) = P(\epsilon_i \le s) = P(\epsilon_{0i}\sqrt{g(\boldsymbol{\tau}, \boldsymbol{w}_i)} \le s_0\sqrt{g(\boldsymbol{\tau}, \boldsymbol{w}_i)}) = P(\epsilon_{0i} \le s_0) = F_0(s_0).$$

The above variance formula generalizes [KSV81] and Example 2 in [Zhe87] to the case $y_i \in \mathbb{R}$.

For the second moment synthetic variable, setting $\phi_2(\tilde{\epsilon}_i) \equiv 0$ gives

$$\phi_1'(\tilde{\epsilon}_i) = \frac{\phi_1(\tilde{\epsilon}_i) g(\tilde{\epsilon}_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)}{1 - G(\tilde{\epsilon}_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)} + \frac{2\tilde{\epsilon}_i}{1 - G(\tilde{\epsilon}_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)},$$

so

$$\phi_1(\tilde{\epsilon}_i) = \frac{\tilde{\epsilon}_i^2}{1 - G(\tilde{\epsilon}_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)}.$$

Substituting $\phi_2(t) = 0$ and $\phi_1(\epsilon_i) = \frac{\tilde{\epsilon}_i^2}{1-G(\tilde{\epsilon}_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)}$ into $\mathrm{Var}\epsilon_{i2}^*$ gives

$$\begin{aligned}
\mathrm{Var}\epsilon_{i2}^* &= \mathbb{E}\frac{\epsilon_i^4}{1 - G(\boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i-)} - \mathbb{E}\,\epsilon_i^4 + \mathbb{E}\,\epsilon_i^4 - (\mathbb{E}\,\epsilon_i^2)^2 \\
&= \mathrm{Var}\epsilon_i^2 + \mathbb{E}\frac{G(\boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i-)}{1 - G(\boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i-)}\epsilon_i^4 \\
&= \mathrm{Var}\epsilon_i^2 + \int \frac{G(\boldsymbol{x}_i^T\boldsymbol{\beta} + s-)}{1 - G(\boldsymbol{x}_i^T\boldsymbol{\beta} + s-)} s^4 dF_i(s).
\end{aligned}$$

3.2.2.2 Leurgans

If we enforce $\varphi_1 \equiv \varphi_2 = \varphi$, then the differential equation (3.10) simplifies to $\varphi_1'(\tilde{y}_i) = \varphi_2'(\tilde{y}_i) = 1/[1 - G(\tilde{y}_i-)]$. Thus $\varphi(\tilde{y}_i) = \tilde{y}_i + \int_{-\infty}^{\tilde{y}-} G(t)/[1 - G(t)]\, dt$, which generalizes [Leu87] and [Zhe87, Example 3] to the case $y_i \in \mathbb{R}$. Intuitively, we compensate a larger observed value by a larger positive quantity because it is more likely to be right-censored. We require $F^{-1}(1) < G^{-1}(1)$, otherwise the integral becomes infinity. The variance of the synthetic

variable $y_{i1}^* = \tilde{y}_i + \int_{-\infty}^{\tilde{y}_i} G(t)/[1-G(t)]\,dt$ is

$$\begin{aligned}\mathrm{Var} y_{i1}^* &= \mathrm{Var} y_i + 2\int_{G^{-1}(0)}^{\infty} [1-F_i(s)][\varphi(s)-s]\,ds \\ &= \mathrm{Var} y_i + 2\int_{G^{-1}(0)}^{\infty} [1-F_i(s)] \int_{-\infty}^{\boldsymbol{x}_i^T\boldsymbol{\beta}+s} \frac{G(t)}{1-G(t)} dt\,ds.\end{aligned}$$

See Supplementary Materials section 3.5.1 for derivation details. When $y_i > 0$, we recover the variance formula given in [Zhe87, Example 3].

For the second moment synthetic variable, setting $\phi_1 = \phi_2 = \phi$ gives

$$\phi'(\tilde{\epsilon}_i) = \frac{2\tilde{\epsilon}_i}{1-G(\tilde{\epsilon}_i + \boldsymbol{x}_i^T\boldsymbol{\beta}-)},$$

so

$$\phi(\tilde{\epsilon}_i) = \tilde{\epsilon}_i^2 + \int_{-\infty}^{\tilde{\epsilon}_i+\boldsymbol{x}_i^T\boldsymbol{\beta}-} \frac{2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})G(t)}{1-G(t)} dt.$$

The working variance is

$$\mathrm{Var}\epsilon_{i2}^* = \mathrm{Var}\epsilon_i^2 + 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{\infty} [1-F(s)]s(\phi(s)-s^2)ds.$$

A detailed derivation is given in Supplementary Materials section 3.5.1. We can evaluate the second term as

$$\begin{aligned}& 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{\infty} [1-F_i(s)]s(\phi(s)-s^2)ds \\ =\ & 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{\infty} [1-F_i(s)]s\Big(\int_{-\infty}^{s+\boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{2tG(t)}{1-G(t)}dt - 2\boldsymbol{x}_i^T\boldsymbol{\beta}\int_{-\infty}^{s+\boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{G(t)}{1-G(t)}dt\Big)ds \\ =\ & 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{\infty} [1-F_i(s)]s\int_{-\infty}^{s+\boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{2tG(t)}{1-G(t)}dtds \\ & -8\boldsymbol{x}_i^T\boldsymbol{\beta}\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{\infty} [1-F_i(s)]s\int_{-\infty}^{s+\boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{G(t)}{1-G(t)}dtds.\end{aligned}$$

3.2.3 Inference

For the mean parameter $\boldsymbol{\beta}$, the asymptotics in [LYZ95] can be directly applied. Asymptotic covariance of $\widehat{\boldsymbol{\beta}}$ is estimated by a sandwich estimator of form $\boldsymbol{A}_n^{-1}\boldsymbol{B}_n\boldsymbol{A}_n^{-1}$. For the KSvR

synthetic variable,

$$\begin{aligned}
\boldsymbol{A}_n &= \frac{1}{n}\sum_{i=1}^{n} w_{i1}\boldsymbol{x}_i\boldsymbol{x}_i^T, \\
\boldsymbol{B}_n &= \frac{1}{n}\sum_{i=1}^{n} w_{i1}(y_{i1}^* - \boldsymbol{x}_i^T\boldsymbol{\beta})^2\boldsymbol{x}_i\boldsymbol{x}_i^T \\
&\quad -\frac{1}{n}\sum_{j=1}^{n}\left\{\frac{-1}{Y_n(\tilde{y}_j)-\Delta N_n(\tilde{y}_j)}\sum_{i=1}^{n} w_{i1}y_{i1}^*\boldsymbol{x}_i I(\tilde{y}_i > \tilde{y}_j)\right\}^{\otimes 2}\left(\sum_{i=1}^{n} I(\tilde{y}_i > \tilde{y}_j)\right)\frac{\Delta N_n(\tilde{y}_j)}{Y_n(\tilde{y}_j)},
\end{aligned}$$

where $Y_n(s) = \sum_{i=1}^{n} I_{\{\tilde{\epsilon}_i \geq s\}}$ is the number of subjects who survive just before time s (natrisk), $N_n(s)$ is the number of failures that occurred by time s, and $\Delta N_n(s) = N_n(s) - N_n(s-) = \sum_{i=1}^{n} I_{\{\tilde{\epsilon}_i \leq s, \delta_i = 0\}}$ is the number of failures that occur at time s (nevents). For the Leurgans synthetic variable, we replace $\boldsymbol{B}_n$ by

$$\begin{aligned}
\boldsymbol{B}_n &= \frac{1}{n}\sum_{i=1}^{n} w_{i1}(y_{i1}^* - \boldsymbol{x}_i^T\boldsymbol{\beta})^2\boldsymbol{x}_i\boldsymbol{x}_i^T \\
&\quad -\frac{1}{n}\sum_{j=1}^{n}\left\{\frac{1}{Y_n(\tilde{y}_j)-\Delta N_n(\tilde{y}_j)}\sum_{i=1}^{n}\frac{w_{i1}\delta_i\boldsymbol{x}_i(\tilde{y}_i - \tilde{y}_j)}{1-\widehat{G}_n(\tilde{y}_i)}\right\}^{\otimes 2}\left(\sum_{i=1}^{n} I(\tilde{y}_i > \tilde{y}_j)\right)\frac{\Delta N_n(\tilde{y}_j)}{Y_n(\tilde{y}_j)}
\end{aligned}$$

3.3 Simulations

We perform a simulation study to evaluate the estimation accuracy of our proposed method. Nonintercept entries of $\boldsymbol{X}$ are generated from independent standard normal. Nonintercept entries of $\boldsymbol{W}$ include both a binary variable ($\sim$ Bernoulli(0.5)) and a standard normal variable. The binary variable is standardized before model-fitting to improve the stability of estimation. The true regression coefficients are $\boldsymbol{\beta}_{\text{true}} = (1.5, 1.0, -0.5, 0.1, 0)$ and $\boldsymbol{\tau}_{\text{true}} = (-0.5, -0.1, 0)$. We vary sample size $N \in \{500, 2000, 5000\}$ and censoring rate $\in \{0.1, 0.25, 0.5, 0.75\}$. Each simulation scenario was run on 200 replicates. The weighted estimates are obtained after five rounds of weighting. Figure 3.1 and 3.2 show the mean squared error (MSE) of parameter estimates under different scenarios. We can see a significant improvement in MSE for the weighted estimates and the difference is especially pronounced for higher censoring rates and larger sample sizes.

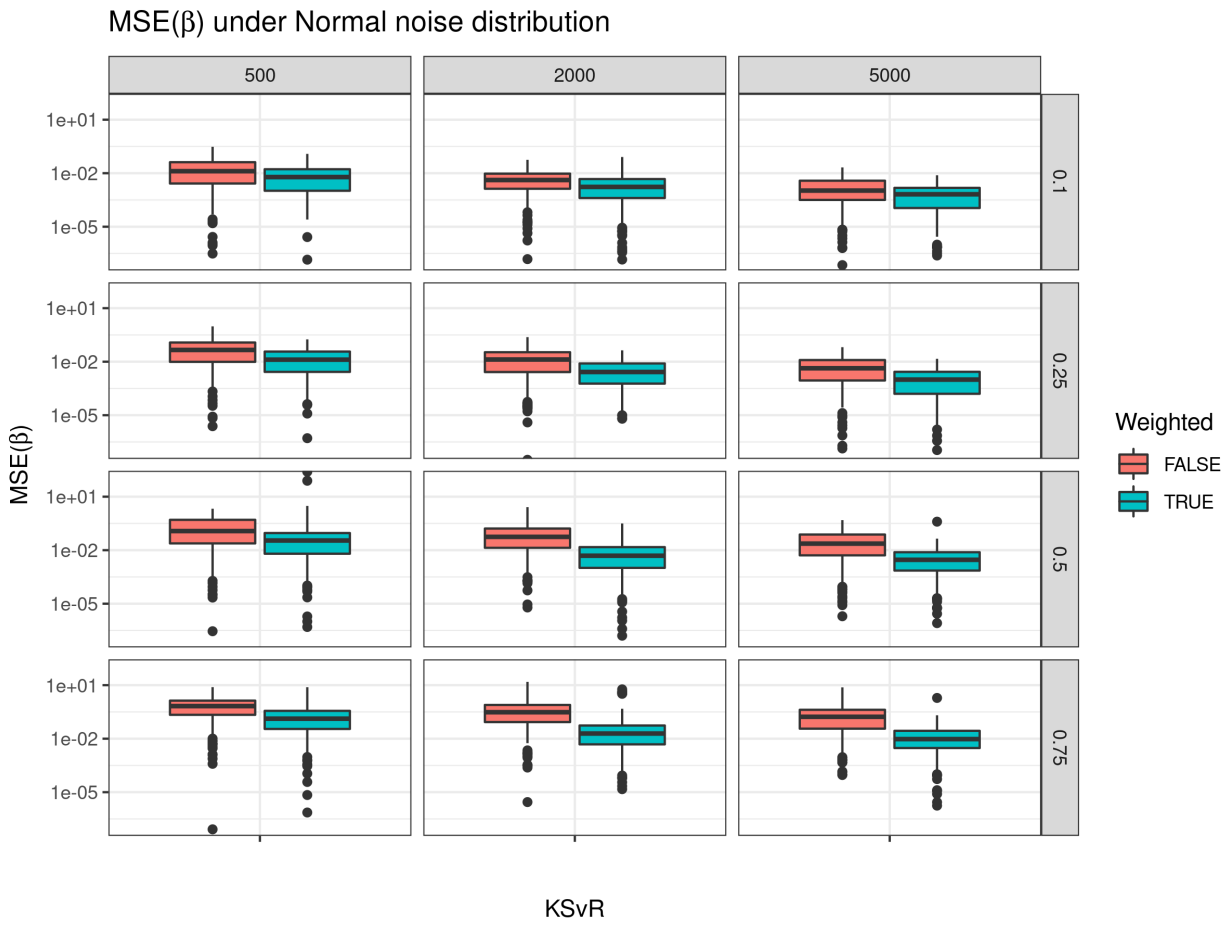


Figure 3.1: Mean squared error of mean parameters $\boldsymbol{\beta}$

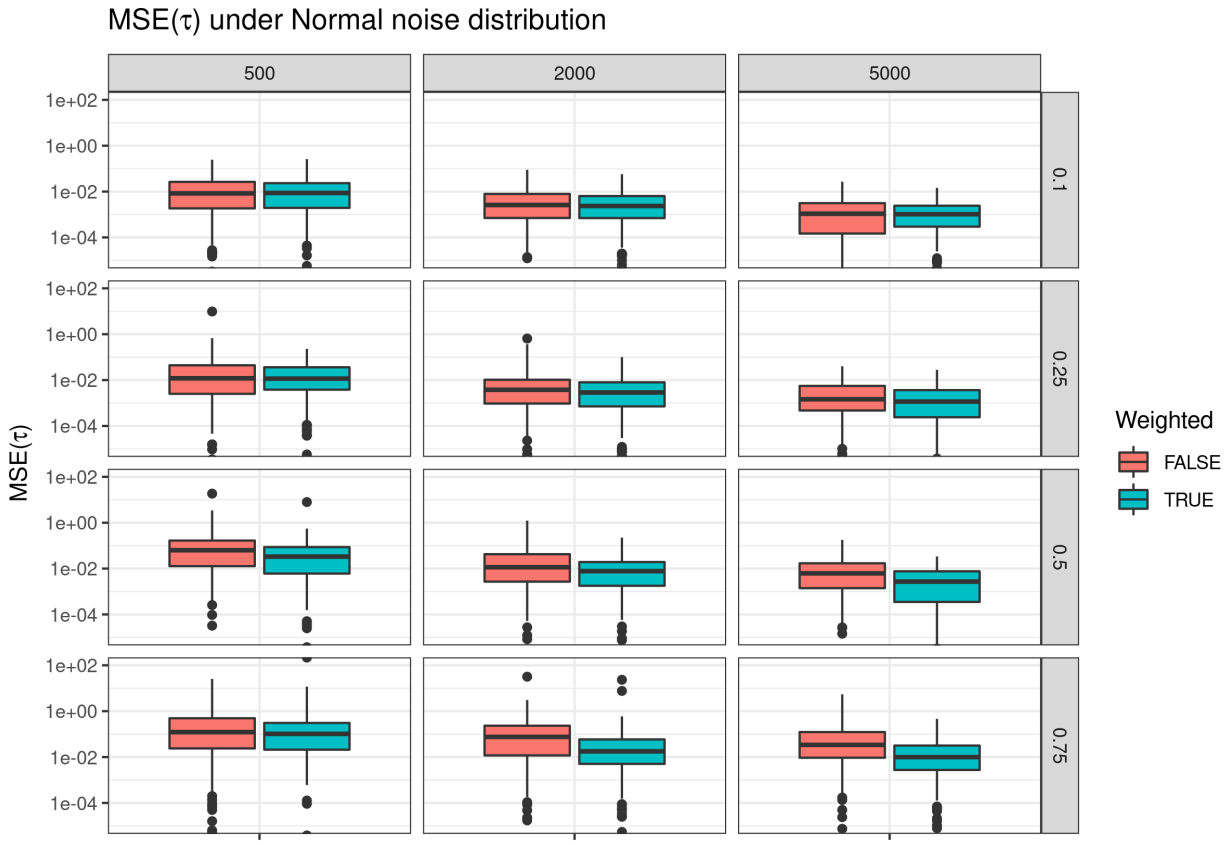


Figure 3.2: Mean squared error of variance parameters $\boldsymbol{\tau}$.

3.4 Conclusion

We have developed a synthetic variable based method for analyzing right censored data. Our method accommodates heteroscedasticity, allows the variance to be explicitly modeled, and uses iterative weighting to improve the estimation efficiency.

3.5 Supplementary Materials

3.5.1 Working variance for Leurgans Synthetic Variable

$$
\begin{aligned}
\mathrm{Var} y_1^* &= \mathrm{Var} y + \mathbb{E}(y_1^* - y_i)^2 \\
&= \mathrm{Var} y + \int_{-\infty}^{\infty} [1 - G(y_i)]\varphi^2(y_i)\, dF(y_i) \\
&\quad + \int_{-\infty}^{\infty} [1 - F(s)]\varphi^2(s)\, dG(s) - \mathbb{E}\, y^2 \\
&= \mathrm{Var} y + [1 - G(y_i)]\varphi^2(y_i)F(y_i) \mid_{-\infty}^{\infty} - \int_{-\infty}^{\infty} [1 - G(s)]2\varphi(s)\varphi'(s)F(s)\, ds \\
&\quad + \int_{-\infty}^{\infty} g(s)\varphi^2(s)F(s)\, ds + \int_{-\infty}^{\infty} \varphi^2(s)\, dG(s) - \int_{-\infty}^{\infty} g(s)\varphi^2(s)F(s)\, ds - \mathbb{E}\, y^2 \\
&= \mathrm{Var} y - \int_{-\infty}^{\infty} 2\varphi(s)F(s)\, ds + \int_{-\infty}^{\infty} \varphi^2(s)\, dG(s) - \mathbb{E}\, y^2 \\
&= \mathrm{Var} y - \int_{-\infty}^{0} 2\varphi(s)F(s)\, ds - \int_{0}^{\infty} 2\varphi(s)F(s)\, ds \\
&\quad + \int_{-\infty}^{0} \varphi^2(s)\, dG(s) - \int_{0}^{\infty} \varphi^2(s)\, d[1 - G(s)] \\
&\quad + \int_{-\infty}^{0} 2sF(s)\, ds - \int_{0}^{\infty} 2s[1 - F(s)]\, ds \\
&= \mathrm{Var} y - \int_{-\infty}^{0} 2\varphi(s)F(s)\, ds - \int_{0}^{\infty} 2\varphi(s)F(s)\, ds \\
&\quad + \varphi^2(s)G(s) \mid_{-\infty}^{0} - \int_{-\infty}^{0} 2\varphi(s)\frac{G(s)}{1 - G(s)}\, ds - \varphi^2(s)[1 - G(s)] \mid_{0}^{\infty} + \int_{0}^{\infty} 2\varphi(s)\, ds \\
&\quad + \int_{-\infty}^{0} 2sF(s)\, ds - \int_{0}^{\infty} 2s[1 - F(s)]\, ds \\
&= \mathrm{Var} y - \int_{-\infty}^{0} 2\varphi(s)F(s)\, ds - \int_{-\infty}^{0} 2\varphi(s)\frac{G(s)}{1 - G(s)}\, ds + \int_{-\infty}^{0} 2sF(s)\, ds \\
&\quad + \varphi^2(0) - \int_{0}^{\infty} 2\varphi(s)F(s)\, ds + \int_{0}^{\infty} 2\varphi(s)\, ds - \int_{0}^{\infty} 2s[1 - F(s)]\, ds \\
&= \mathrm{Var} y - \int_{G^{-1}(0)}^{0} 2\varphi(s)F(s)\, ds - \int_{G^{-1}(0)}^{0} 2\varphi(s)\left[\frac{1}{1 - G(s)} - 1\right] ds \\
&\quad + \int_{G^{-1}(0)}^{0} 2sF(s)\, ds + \varphi^2(0) - \int_{0}^{\infty} 2\varphi(s)F(s)\, ds + \int_{0}^{\infty} 2\varphi(s)\, ds \\
&\quad - \int_{0}^{\infty} 2s[1 - F(s)]\, ds
\end{aligned}
$$

$$
\begin{aligned}
&= \mathrm{Var}y + \int_{G^{-1}(0)}^{0} 2[1-F(s)]\varphi(s)\,ds - \int_{G^{-1}(0)}^{0} 2\varphi(s)\varphi'(s)\,ds + \int_{G^{-1}(0)}^{0} 2sF(s)\,ds \\
&\quad +\varphi^2(0) + 2\int_0^{\infty} [1-F(s)][\varphi(s)-s]\,ds \\
&= \mathrm{Var}y + \int_{G^{-1}(0)}^{0} 2[1-F(s)]\varphi(s)\,ds - \varphi^2(s)\,|_{G^{-1}(0)}^{0} + \int_{G^{-1}(0)}^{0} 2sF(s)\,ds \\
&\quad +\varphi^2(0) + 2\int_0^{\infty} [1-F(s)][\varphi(s)-s]\,ds \\
&= \mathrm{Var}y + \int_{G^{-1}(0)}^{0} 2[1-F(s)]\varphi(s)\,ds + \int_{G^{-1}(0)}^{0} 2sF(s)\,ds - \int_{G^{-1}(0)}^{0} 2s\,ds \\
&\quad +2\int_0^{\infty} [1-F(s)][\varphi(s)-s]\,ds \\
&= \mathrm{Var}y + 2\int_{G^{-1}(0)}^{\infty} [1-F(s)][\varphi(s)-s]\,ds
\end{aligned}
$$

When $y > 0$, we recover the variance formula given in [Zhe87, Example 3]: $\mathrm{Var}y + 2\int_0^{\infty}[1-F(s)][\varphi(s)-s]\,ds$.

Derivation for the working variance of the second moment synthetic variable.

$$
\begin{aligned}
\mathrm{Var}\epsilon_{i2}^{*} &= \mathrm{Var}\epsilon_i^2 + \int_{-\infty}^{\infty}[1-G(s+\boldsymbol{x}_i^T\boldsymbol{\beta})]\phi^2(s)dF(s) \\
&\quad + \int_{-\infty}^{\infty}\phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(1-F(t-\boldsymbol{x}_i^T\boldsymbol{\beta}))dG(t) - \mathbb{E}\,\epsilon_i^4 \\
&= \mathrm{Var}\epsilon_i^2 + [1-G(s+\boldsymbol{x}_i^T\boldsymbol{\beta})]\phi^2(s)F(s)|_{-\infty}^{\infty} \\
&\quad - \big[2\int_{-\infty}^{\infty}[1-G(s+\boldsymbol{x}_i^T\boldsymbol{\beta})]\phi(s)\phi'(s)F(s)ds - \int g(s+\boldsymbol{x}_i^T\boldsymbol{\beta})\phi^2(s)F(s)ds\big] \\
&\quad + \int_{-\infty}^{\infty}\phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dG(t) - \int_{-\infty}^{\infty}\phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})F(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dG(t) - \mathbb{E}\,\epsilon_i^4 \\
&\quad \text{Since } \phi'(s) = \frac{2s}{1-G(s+\boldsymbol{x}_i^T\boldsymbol{\beta})} \text{ and} \\
&\quad \int g(s+\boldsymbol{x}_i^T\boldsymbol{\beta})\phi^2(s)F(s)ds = \int_{-\infty}^{\infty} g(t)\phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})F(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
&= \mathrm{Var}\epsilon_i^2 - 4\int_{-\infty}^{\infty}\phi(s)sF(s)ds + \int_{-\infty}^{\infty}\phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dG(t) - \mathbb{E}\,\epsilon_i^4 \\
&= \mathrm{Var}\epsilon_i^2 - 4\int_{-\infty}^{0}\phi(s)sF(s)ds - 4\int_{0}^{\infty}\phi(s)sF(s)ds \\
&\quad + \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta}) - \int_{-\infty}^{0}2\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
&\quad + \int_{-\infty}^{0}4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt + \int_{0}^{\infty}4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
&\quad - \int_{0}^{\infty}4s^3[1-F(s)]\,ds + \int_{-\infty}^{0}4s^3F(s)\,ds \\
&\quad \text{(Arrage terms so those with the same integral limits are together.)} \\
&= \mathrm{Var}\epsilon_i^2 - 4\int_{-\infty}^{0}\phi(s)sF(s)ds - \int_{-\infty}^{0}2\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
&\quad + \int_{-\infty}^{0}4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt + \int_{-\infty}^{0}4s^3F(s)ds \\
&\quad - 4\int_{0}^{\infty}\phi(s)sF(s)ds + \int_{0}^{\infty}4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
&\quad - \int_{0}^{\infty}4s^3[1-F(s)]\,ds + \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta}) \\
&= \mathrm{Var}\epsilon_i^2 - 4\int_{-\infty}^{0}\phi(s)sF(s)ds - \int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}4s^3ds \\
&\quad + \int_{-\infty}^{0}4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt + \int_{-\infty}^{0}4s^3F(s)ds \\
&\quad - 4\int_{0}^{\infty}\phi(s)sF(s)ds + \int_{0}^{\infty}4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt - \int_{0}^{\infty}4s^3[1-F(s)]\,ds
\end{aligned}
$$

To simplify it, note that the first integral is

$$-4\int_{-\infty}^{0}\phi(s)sF(s)ds = -4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}\phi(s)sF(s)ds - 4\int_{-\infty}^{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}} s^3F(s)ds,$$

because $\phi(s) = s^2$ on the interval $(-\infty, G^{-1}(0) - \boldsymbol{x}_i^T\boldsymbol{\beta}]$. The second integral and the fourth integral combine to

$$\begin{aligned} & -4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0} s^3ds + 4\int_{-\infty}^{0} s^3F(s)ds \\ = \; & -4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0} s^3[1-F(s)]ds + 4\int_{-\infty}^{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}} s^3F(s)ds. \end{aligned}$$

The third integral is

$$\begin{aligned} & 4\int_{-\infty}^{0}\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\ = \; & 4\int_{G^{-1}(0)}^{0}\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\ = \; & 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{-\boldsymbol{x}_i^T\boldsymbol{\beta}}\phi(s)sds \\ = \; & \begin{cases} 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}\phi(s)sds - 4\int_{-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}\phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta}\geq 0 \\ 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}\phi(s)sds + 4\int_{0}^{-\boldsymbol{x}_i^T\boldsymbol{\beta}}\phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta}<0 \end{cases}. \end{aligned}$$

Combining the above, we have

$$\begin{aligned} & -4\int_{-\infty}^{0}\phi(s)sF(s)ds - \int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}4s^3ds \\ & + \int_{-\infty}^{0}4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt + \int_{-\infty}^{0}4s^3F(s)ds \\ = \; & \begin{cases} 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}[1-F(s)]s(\phi(s)-s^2)ds - 4\int_{-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}\phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta}\geq 0 \\ 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0}[1-F(s)]s(\phi(s)-s^2)ds + 4\int_{0}^{-\boldsymbol{x}_i^T\boldsymbol{\beta}}\phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta}<0 \end{cases}. \end{aligned}$$

Furthermore, since

$$\int_0^\infty 4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt = 4\int_{-\boldsymbol{x}_i^T\boldsymbol{\beta}}^\infty \phi(s)sds =$$
$$\begin{cases} 4\int_0^\infty \phi(s)sds + 4\int_{-\boldsymbol{x}_i^T\boldsymbol{\beta}}^0 \phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta} \geq 0 \\ 4\int_0^\infty \phi(s)sds - 4\int_0^{-\boldsymbol{x}_i^T\boldsymbol{\beta}} \phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta} < 0 \end{cases},$$

the last three integrals simplify to

$$\begin{aligned} & -4\int_0^\infty \phi(s)sF(s)ds + \int_0^\infty 4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt - \int_0^\infty 4s^3[1-F(s)]\,ds \\ = & \begin{cases} 4\int_0^\infty [1-F(s)]s(\phi(s)-s^2)ds + 4\int_{-\boldsymbol{x}_i^T\boldsymbol{\beta}}^0 \phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta} \geq 0 \\ 4\int_0^\infty [1-F(s)]s(\phi(s)-s^2)ds - 4\int_0^{-\boldsymbol{x}_i^T\boldsymbol{\beta}} \phi(s)sds & \boldsymbol{x}_i^T\boldsymbol{\beta} < 0 \end{cases}. \end{aligned}$$

All together, we get

$$\mathrm{Var}\epsilon_{i2}^* = \mathrm{Var}\epsilon_i^2 + 4\int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^\infty [1-F(s)]s(\phi(s)-s^2)ds.$$

Details for simplifying $\int_{-\infty}^\infty \phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dG(t)$:

$$\begin{aligned} & \int_{-\infty}^\infty \phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dG(t) \\ = & \int_{-\infty}^0 \phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dG(t) - \int_0^\infty \phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})d(1-G(t)) \\ = & \phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})G(t)|_{-\infty}^0 - \int_{-\infty}^0 2\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t-\boldsymbol{x}_i^T\boldsymbol{\beta})G(t)dt \\ & -\phi^2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(1-G(t))|_0^\infty + \int_0^\infty 2\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t-\boldsymbol{x}_i^T\boldsymbol{\beta})[1-G(t)]dt \\ = & \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta})G(0) - \int_{-\infty}^0 2\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t-\boldsymbol{x}_i^T\boldsymbol{\beta})[1-(1-G(t))]dt \\ & +\phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta})(1-G(0)) + \int_0^\infty 4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\ = & \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta}) - \int_{-\infty}^0 2\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \\ & + \int_{-\infty}^0 4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt + \int_0^\infty 4\phi(t-\boldsymbol{x}_i^T\boldsymbol{\beta})(t-\boldsymbol{x}_i^T\boldsymbol{\beta})dt \end{aligned}$$

where the last line uses $\phi'(t - \boldsymbol{x}_i^T\boldsymbol{\beta}) = \frac{2(t-\boldsymbol{x}_i^T\boldsymbol{\beta})}{1-G(t)}$. Furthermore,

$$\begin{aligned}
& \int_{-\infty}^{0} 2\phi(t - \boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t - \boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
= & \int_{G^{-1}(0)}^{0} 2\phi(t - \boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t - \boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
= & \ 2\phi^2(t - \boldsymbol{x}_i^T\boldsymbol{\beta})|_{G^{-1}(0)}^{0} - \int_{G^{-1}(0)}^{0} 2\phi(t - \boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t - \boldsymbol{x}_i^T\boldsymbol{\beta})dt.
\end{aligned}$$

So

$$\begin{aligned}
& \int_{-\infty}^{0} 2\phi(t - \boldsymbol{x}_i^T\boldsymbol{\beta})\phi'(t - \boldsymbol{x}_i^T\boldsymbol{\beta})dt \\
= & \ \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta}) - \phi^2(G^{-1}(0) - \boldsymbol{x}_i^T\boldsymbol{\beta}) \\
& \text{Since } \phi(\epsilon_i) = \epsilon_i^2 + \int_{-\infty}^{\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{2(t - \boldsymbol{x}_i^T\boldsymbol{\beta})G(t)}{1 - G(t)}dt \\
= & \ \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta}) - (G^{-1}(0) - \boldsymbol{x}_i^T\boldsymbol{\beta})^4 \\
= & \ \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta}) + \int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{0} 4s^3 ds \\
= & \ \phi^2(-\boldsymbol{x}_i^T\boldsymbol{\beta}) + \int_{G^{-1}(0)-\boldsymbol{x}_i^T\boldsymbol{\beta}}^{-\boldsymbol{x}_i^T\boldsymbol{\beta}} 4s^3 ds - (\boldsymbol{x}_i^T\boldsymbol{\beta})^4
\end{aligned}$$

To implement the second moment Leurgans

$$\begin{aligned}
\phi(\epsilon_i) &= \epsilon_i^2 + \int_{-\infty}^{\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{2(t - \boldsymbol{x}_i^T\boldsymbol{\beta})G(t)}{1 - G(t)}dt \\
&= \epsilon_i^2 + \int_{-\infty}^{\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{2tG(t)}{1 - G(t)}dt - 2\boldsymbol{x}_i^T\boldsymbol{\beta} \int_{-\infty}^{\epsilon_i + \boldsymbol{x}_i^T\boldsymbol{\beta}} \frac{G(t)}{1 - G(t)}dt
\end{aligned}$$

we cache both $\int \frac{2tG(t)}{1-G(t)}dt$ and $\int \frac{G(t)}{1-G(t)}dt$ so there is no need to re-evaluate the integral given a new $\boldsymbol{x}_i^T\boldsymbol{\beta}$.

CHAPTER 4

Concluding Remarks

This dissertation presents our recent effort in developing statistical inference tools for modern large and high-dimensional data sets. The first project provides a bag of little bootstrap (BLB) based method for conducting statistical inference of linear mixed models on massive and distributed longitudinal data sets. We provide theoretical guarantees for our algorithm and implement it as a Julia software package `MixedModelsBLB.jl`, which is freely available at `https://github.com/xinkai-zhou/MixedModelsBLB.jl`. A natural extension of this project is to develop similar algorithms for other types of outcomes such as binary or counts through generalized linear mixed models.

The second project provides a flexible and general statistical inference framework for constrained or regularized estimation problems. Our ProxMCMC method is fully Bayesian and can be easily adapted to handle various types of constraints and regularizations. Future research directions include developing ProxMCMC algorithms for more estimation problems and devising principled ways of choosing the Moreau-Yosida envelope parameter λ.

The third project provides tools for conducting estimation and inference of heteroscedastic linear models for analyzing censored data using synthetic variables. Our method allows for the explicit modeling of the heterogeneous variances and improves the estimation efficiency compared to the classical synthetic variable approaches. Further work in this area may extend the method to more general censoring mechanisms such as left or interval censoring.

REFERENCES

[ADL13] Artin Armagan, David Dunson, and J Lee. "Generalized double Pareto shrinkage." *Statistica Sinica*, **23**:119–143, 2013.

[Bar37] Maurice Stevenson Bartlett. "Properties of sufficiency and statistical tests." *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*, **160**(901):268–282, 1937.

[BBB13] Richard Berk, Lawrence Brown, Andreas Buja, Kai Zhang, and Linda Zhao. "Valid post-selection inference." *Ann. Statist.*, **41**(2):802–837, 2013.

[Bec17] Amir Beck. *First-Order Methods in Optimization*, volume 25 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA; Mathematical Optimization Society, Philadelphia, PA, 2017.

[BF74] Morton B Brown and Alan B Forsythe. "Robust tests for the equality of variances." *Journal of the American statistical association*, **69**(346):364–367, 1974.

[BG07] John B Buse, ACCORD Study Group, et al. "Action to Control Cardiovascular Risk in Diabetes (ACCORD) trial: design and methods." *The American journal of cardiology*, **99**(12):S21–S33, 2007.

[BG19] Amir Beck and Nili Guttmann-Beck. "FOM–A MATLAB toolbox of first-order methods for solving convex optimization problems." *Optimization Methods and Software*, **34**(1):172–193, 2019.

[BGZ97] P. J. Bickel, F. Götze, and W. R. van Zwet. "Resampling fewer than n observations: gains, losses, and remedies for losses." *Statistica Sinica*, **7**(1):1–31, 1997.

[BJ79] Jonathan Buckley and Ian James. "Linear regression with censored data." *Biometrika*, **66**(3):429–436, 1979.

[BMB15] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. "Fitting Linear Mixed-Effects Models Using lme4." *Journal of Statistical Software, Articles*, **67**(1):1–48, 2015.

[BNW06] Richard H Byrd, Jorge Nocedal, and Richard A Waltz. "Knitro: An integrated package for nonlinear optimization." In *Large-Scale Nonlinear Optimization*, pp. 35–59. Springer, 2006.

[BPP15] Anirban Bhattacharya, Debdeep Pati, Natesh S. Pillai, and David B. Dunson. "Dirichlet-Laplace priors for optimal shrinkage." *J. Amer. Statist. Assoc.*, **110**(512):1479–1490, 2015.

[BPS20] François Bachoc, David Preinerstorfer, and Lukas Steinberger. "Uniformly valid confidence intervals post-model-selection." *The Annals of Statistics*, **48**(1):440–463, 2020.

[BWT03] Gregory Belenky, Nancy J Wesensten, David R Thorne, Maria L Thomas, Helen C Sing, Daniel P Redmond, Michael B Russo, and Thomas J Balkin. "Patterns of performance degradation and restoration during sleep restriction and subsequent recovery: A sleep dose-response study." *Journal of Sleep Research*, **12**(1):1–12, 2003.

[Coo97] Nancy R Cook. "An imputation method for non-ignorable missing data in studies of blood pressure." *Statistics in medicine*, **16**(23):2713–2728, 1997.

[CP11] Patrick L. Combettes and Jean-Christophe Pesquet. *Proximal Splitting Methods in Signal Processing*, pp. 185–212. Springer New York, New York, NY, 2011.

[CPS10] Carlos M Carvalho, Nicholas G Polson, and James G Scott. "The horseshoe estimator for sparse signals." *Biometrika*, **97**(2):465–480, 2010.

[CR04] Ciprian M. Crainiceanu and David Ruppert. "Likelihood ratio tests in linear mixed models with one variance component." *J. R. Stat. Soc. Ser. B Stat. Methodol.*, **66**(1):165–185, 2004.

[Cra08] C.M. Crainiceanu. "Likelihood ratio testing for zero variance components in linear mixed models." In D.B. Dunson, editor, *Random Effect and Latent Variable Model Selection*, volume 192 of *Lecture Notes in Statistics*. Springer, 2008.

[CTT17] Yunjin Choi, Jonathan Taylor, and Robert Tibshirani. "Selecting the number of principal components: Estimation of the true rank of a noisy matrix." *The Annals of Statistics*, pp. 2590–2617, 2017.

[CW05] Patrick L. Combettes and Valérie R. Wajs. "Signal Recovery by Proximal Forward-Backward Splitting." *Multiscale Modeling & Simulation*, **4**(4):1168–1200, 2005.

[CWB14] Ying Cao, Peng Wei, Matthew Bailey, John SK Kauwe, Taylor J Maxwell, and Alzheimer's Disease Neuroimaging Initiative. "A versatile omnibus test for detecting mean and variance heterogeneity." *Genetic epidemiology*, **38**(1):51–59, 2014.

[CZL14] Eric C. Chi, Hua Zhou, and Kenneth Lange. "Distance majorization and its applications." *Mathematical Programming*, **146**(1):409–436, 2014.

[DMP18] Alain Durmus, Eric Moulines, and Marcelo Pereyra. "Efficient bayesian computation by proximal markov chain monte carlo: when langevin meets moreau." *SIAM Journal on Imaging Sciences*, **11**(1):473–506, 2018.

[Efr79] Bradley Efron. "Bootstrap methods: another look at the jackknife." *Ann. Statist.*, **7**(1):1–26, 1979.

[EHJ04] Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al. "Least angle regression." *Annals of Statistics*, **32**(2):407–499, 2004.

[FHT08] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. "Sparse inverse covariance estimation with the graphical lasso." *Biostatistics*, **9**(3):432–441, 2008.

[FK76] Michael A Fligner and Timothy J Killeen. "Distribution-free two-sample tests for scale." *Journal of the American Statistical Association*, **71**(353):210–213, 1976.

[GB12] Jim E Griffin and Philip J Brown. "Structuring shrinkage: some correlated priors for regression." *Biometrika*, **99**(2):481–487, 2012.

[GB13] Jim E. Griffin and Philip J. Brown. "Some priors for sparse regression modelling." *Bayesian Anal.*, **8**(3):691–702, 2013.

[GBR14] Sara van de Geer, Peter Bühlmann, Ya'acov Ritov, and Ruben Dezeure. "On asymptotically optimal confidence regions and tests for high-dimensional models." *Ann. Statist.*, **42**(3):1166–1202, 2014.

[GKZ18] Brian R Gaines, Juhyun Kim, and Hua Zhou. "Algorithms for fitting the constrained lasso." *Journal of Computational and Graphical Statistics*, **27**(4):861–871, 2018.

[GM93] Edward I. George and Robert E. McCulloch. "Variable selection via Gibbs sampling." *Journal of the American Statistical Association*, **88**(423):881–889, 1993.

[Gra19] Robert B. Gramacy. *monomvn: Estimation for MVN and Student-t Data with Monotone Missingness*, 2019. R package version 1.9-13.

[GSZ21] Christopher A German, Janet S Sinsheimer, Jin Zhou, and Hua Zhou. "WiSER: Robust and scalable estimation and inference of within-subject variances from intensive longitudinal data." *Biometrics*, 2021.

[GXG18] Hong Ge, Kai Xu, and Zoubin Ghahramani. "Turing: A language for flexible probabilistic inference." In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 1682–1690. PMLR, 09–11 Apr 2018.

[Han11] Chris Hans. "Elastic net regression modeling with the orthant normal prior." *Journal of the American Statistical Association*, **106**(496):1383–1393, 2011.

[HH14] Ulrich Halekoh, Søren Højsgaard, et al. "A kenward-roger approximation and parametric bootstrap methods for tests in linear mixed models–the R package pbkrtest." *Journal of Statistical Software*, **59**(9):1–30, 2014.

[ICB10] Faramarz Ismail-Beigi, Timothy Craven, Mary Ann Banerji, Jan Basile, Jorge Calles, Robert M Cohen, Robert Cuddihy, William C Cushman, Saul Genuth, Richard H Grimm Jr, et al. "Effect of intensive treatment of hyperglycaemia on microvascular outcomes in type 2 diabetes: an analysis of the ACCORD randomised trial." *The Lancet*, **376**(9739):419–430, 2010.

[Ioa05] John PA Ioannidis. "Why most published research findings are false." *PLoS Medicine*, **2**(8):e124, 2005.

[JLY06] Zhezhen Jin, D. Y. Lin, and Zhiliang Ying. "On least-squares regression with censored data." *Biometrika*, **93**(1):147–161, 2006.

[JM14] Adel Javanmard and Andrea Montanari. "Confidence intervals and hypothesis testing for high-dimensional regression." *J. Mach. Learn. Res.*, **15**:2869–2909, 2014.

[Joh20] Steven G. Johnson. "The NLopt nonlinear-optimization package.", 08 2020.

[JPR20] Gareth M. James, Courtney Paulson, and Paat Rusmevichientong. "Penalized and constrained optimization: an application to high-dimensional website advertising." *J. Amer. Statist. Assoc.*, **115**(529):107–122, 2020.

[KBB20] Arun K. Kuchibhotla, Lawrence D. Brown, Andreas Buja, Junhui Cai, Edward I. George, and Linda H. Zhao. "Valid post-selection inference in model-free linear regression." *Ann. Statist.*, **48**(5):2953–2981, 2020.

[KSV81] H. Koul, V. Susarla, and J. Van Ryzin. "Regression analysis with randomly right-censored data." *Ann. Statist.*, **9**(6):1276–1288, 1981.

[KTS14] Ariel Kleiner, Ameet Talwalkar, Purnamrita Sarkar, and Michael I Jordan. "A scalable bootstrap for massive data." *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pp. 795–816, 2014.

[KZL19] Kevin L. Keys, Hua Zhou, and Kenneth Lange. "Proximal Distance Algorithms: Theory and Practice." *Journal of Machine Learning Research*, **20**(66):1–38, 2019.

[Leu87] Sue Leurgans. "Linear models, random censoring and synthetic data." *Biometrika*, **74**(2):301–309, 1987.

[LL09] Wanrong Liu and Xuewen Lu. "Weighted least squares method for censored linear models." *J. Nonparametr. Stat.*, **21**(7):787–799, 2009.

[LL21] Alfonso Landeros and Kenneth Lange. "Algorithms for Sparse Support Vector Machines." arXiv:2110.07691 [stat.ME], 2021.

[LPZ22] Alfonso Landeros, Oscar Hernan Madrid Padilla, Hua Zhou, and Kenneth Lange. "Extensions to the Proximal Distance Method of Constrained Optimization." arXiv:2009.00801 [math.OC], 2022.

[LSS16] Jason D Lee, Dennis L Sun, Yuekai Sun, and Jonathan E Taylor. "Exact post-selection inference, with application to the lasso." *The Annals of Statistics*, **44**(3):907–927, 2016.

[LWL22] Alfonso Landeros, Tong Tong Wu, and Kenneth Lange. "Feature Selection for Vertex Discriminant Analysis." arXiv:2203.11168 [stat.CO], 2022.

[LYZ95] Tze Leung Lai, Zhiliang Ying, and Zu Kang Zheng. "Asymptotic normality of a class of adaptive statistics with applications to synthetic data methods for censored regression." *J. Multivariate Anal.*, **52**(2):259–279, 1995.

[MB88] T. J. Mitchell and J. J. Beauchamp. "Bayesian variable selection in linear regression." *J. Amer. Statist. Assoc.*, **83**(404):1023–1036, 1988. With comments by James Berger and C. L. Mallows and with a reply by the authors.

[MH82] Rupert Miller and Jerry Halpern. "Regression with censored data." *Biometrika*, **69**(3):521–531, 1982.

[MHT10] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. "Spectral regularization algorithms for learning large incomplete matrices." *Journal of Machine Learning Research*, **11**:2287–2322, 2010.

[MKH08] Robyn L McClelland, Richard A Kronmal, Jeffrey Haessler, Roger S Blumenthal, and David C Goff Jr. "Estimation of risk factor associations when the response is influenced by medication use: an imputation approach." *Statistics in Medicine*, **27**(24):5039–5053, 2008.

[MN99] Jan R. Magnus and Heinz Neudecker. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. Wiley Series in Probability and Statistics. John Wiley & Sons Ltd., Chichester, 1999.

[Nea11] Radford M Neal et al. "MCMC using Hamiltonian dynamics." *Handbook of Markov Chain Monte Carlo*, **2**(11):2, 2011.

[PC08] Trevor Park and George Casella. "The Bayesian lasso." *J. Amer. Statist. Assoc.*, **103**(482):681–686, 2008.

[Per16] Marcelo Pereyra. "Proximal markov chain monte carlo algorithms." *Statistics and Computing*, **26**(4):745–760, 2016.

[PRW99] Dimitris N Politis, Joseph P Romano, and Michael Wolf. *Subsampling*. Springer Science & Business Media, 1999.

[PS10] Nicholas G. Polson and James G. Scott. "Shrink globally, act locally: sparse Bayesian regularization and prediction." In *Bayesian Statistics 9*, pp. 501–538. Oxford Univ. Press, Oxford, 2010.

[PSW15] Nicholas G Polson, James G Scott, and Brandon T Willard. "Proximal algorithms in statistics and machine learning." *Statistical Science*, **30**(4):559–581, 2015.

[PV17] Juho Piironen and Aki Vehtari. "Sparsity information and regularization in the horseshoe and other shrinkage priors." *Electronic Journal of Statistics*, **11**(2):5018–5051, 2017.

[RFF10] Lars Rönnegård, Majbritt Felleki, Freddy Fikse, Herman A Mulder, and Erling Strandberg. "Genetic heterogeneity of residual variance-estimation of variance components using double hierarchical generalized linear models." *Genetics Selection Evolution*, **42**(1):1–10, 2010.

[ROF92] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. "Nonlinear total variation based noise removal algorithms." *Physica D: Nonlinear Phenomena*, **60**(1):259–268, 1992.

[RV11] Lars Rönnegård and William Valdar. "Detecting major genetic loci controlling phenotypic variability in experimental crosses." *Genetics*, **188**(2):435–447, 2011.

[RV12] Lars Rönnegård and William Valdar. "Recent developments in statistical methods for detecting genetic loci affecting phenotypic variability." *BMC genetics*, **13**(1):1–7, 2012.

[RW09] R Tyrrell Rockafellar and Roger J-B Wets. *Variational Analysis*, volume 317. Springer Science & Business Media, 2009.

[SL87] Steven G Self and Kung-Yee Liang. "Asymptotic properties of maximum likelihood estimators and likelihood ratio tests under nonstandard conditions." *Journal of the American Statistical Association*, **82**(398):605–610, 1987.

[Smy89] Gordon K Smyth. "Generalized linear models with varying dispersion." *Journal of the Royal Statistical Society: Series B (Methodological)*, **51**(1):47–60, 1989.

[SPP05] Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. "Causal protein-signaling networks derived from multiparameter single-cell data." *Science*, **308**(5721):523–529, 2005.

[SRR15] Elias S Siraj, Daniel J Rubin, Matthew C Riddle, Michael E Miller, Fang-Chi Hsu, Faramarz Ismail-Beigi, Shyh-Huei Chen, Walter T Ambrosius, Abraham Thomas, William Bestermann, et al. "Insulin dose and cardiovascular mortality in the ACCORD trial." *Diabetes Care*, **38**(11):2000–2008, 2015.

[Sta20] Stan Development Team. "Stan Modeling Language Users Guide and Reference Manual.", 2020.

[Tib96] Robert Tibshirani. "Regression shrinkage and selection via the lasso." *J. Roy. Statist. Soc. Ser. B*, **58**(1):267–288, 1996.

[TSS05] Martin D Tobin, Nuala A Sheehan, Katrina J Scurrah, and Paul R Burton. "Adjusting for treatment effects in studies of quantitative traits: antihypertensive therapy and systolic blood pressure." *Statistics in medicine*, **24**(19):2911–2935, 2005.

[TT18] Jonathan Taylor and Robert Tibshirani. "Post-selection inference for-penalized likelihood models." *Canadian Journal of Statistics*, **46**(1):41–61, 2018.

[TTT19] Ryan Tibshirani, Rob Tibshirani, Jonathan Taylor, Joshua Loftus, Stephen Reid, and Jelena Markovic. *selectiveInference: Tools for Post-Selection Inference*, 2019. R package version 1.2.5.

[Vaa00] Aad W Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

[VW13] A. Van Der Vaart and J. Wellner. *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer Series in Statistics. Springer New York, 2013.

[Wan12] Hao Wang. "Bayesian graphical lasso models and efficient posterior computation." *Bayesian Analysis*, **7**(4):867–886, 2012.

[WB06] Andreas Wächter and Lorenz T Biegler. "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming." *Mathematical Programming*, **106**(1):25–57, 2006.

[WCM94] Ian R White, Nishi Chaturvedi, and Paul M McKeigue. "Median analysis of blood pressure for a sample including treated hypertensives." *Statistics in medicine*, **13**(16):1635–1641, 1994.

[WKC03] Ian R White, Ilona Koupilova, and James Carpenter. "The use of regression models for medians when observed outcomes may be modified by interventions." *Statistics in medicine*, **22**(7):1083–1096, 2003.

[WMG22] Kenneth E Westerman, Timothy D Majarian, Franco Giulianini, Dong-Keun Jang, Jenkai Miao, Jose C Florez, Han Chen, Daniel I Chasman, Miriam S Udler, Alisa K Manning, et al. "Variance-quantitative trait loci enable systematic discovery of gene-environment interactions for cardiometabolic serum biomarkers." *Nature communications*, **13**(1):1–11, 2022.

[WZZ19] Huanwei Wang, Futao Zhang, Jian Zeng, Yang Wu, Kathryn E Kemper, Angli Xue, Min Zhang, Joseph E Powell, Michael E Goddard, Naomi R Wray, et al. "Genotype-by-environment interactions inferred from genetic effects on phenotypic variability in the UK Biobank." *Science advances*, **5**(8):eaaw3538, 2019.

[XCL17] Jason Xu, Eric Chi, and Kenneth Lange. "Generalized Linear Model Regression under Distance-to-set Penalties." In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[YLP12] Jian Yang, Ruth JF Loos, Joseph E Powell, Sarah E Medland, Elizabeth K Speliotes, Daniel I Chasman, Lynda M Rose, Gudmar Thorleifsson, Valgerdur Steinthorsdottir, Reedik Mägi, et al. "FTO genotype is associated with phenotypic variability of body mass index." *Nature*, **490**(7419):267–272, 2012.

[Zhe87] Zukang Zheng. "A class of estimators of the parameters in linear regression with censored data." *Acta Mathematicae Applicatae Sinica*, **3**(3):231–241, 1987.

[Zhe08] Zu Kang Zheng. "A class of estimators of the mean survival time from interval censored data with application to linear regression." *Appl. Math. J. Chinese Univ. Ser. B*, **23**(4):377–390, 2008.

[Zho92] Mai Zhou. "Asymptotic normality of the "synthetic data" regression estimator for censored survival data." *Ann. Statist.*, **20**(2):1002–1021, 1992.

[ZL14] Hua Zhou and Lexin Li. "Regularized matrix regression." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **76**(2):463–483, 2014.

[ZLZ13] Hua Zhou, Lexin Li, and Hongtu Zhu. "Tensor regression with applications in neuroimaging data analysis." *Journal of the American Statistical Association*, **108**:540–552, 2013.

[ZZ14] Cun-Hui Zhang and Stephanie S. Zhang. "Confidence intervals for low dimensional parameters in high dimensional linear models." *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, **76**(1):217–242, 2014.