Spring 2022

A Comparison of Inference Methods in High-Dimensional Linear Regression

Imtiaz Ebna Mannan

A Comparison of Inference Methods in High-dimensional Linear Regression

by

Imtiaz Ebna Mannan

Bachelor of Science
University of Dhaka, 2017

Master of Science
University of Dhaka, 2018

Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science in

Statistics

College of Arts and Sciences

University of South Carolina

2022

Accepted by:

Karl B. Gregory, Director of Thesis

David B. Hitchcock, Reader

Minsuk Shin, Reader

Tracey L. Weldon, Interim Vice Provost and Dean of the Graduate School

Dedication

To **my parents** and my aunt **Ripa** khalamoni

Acknowledgments

I am greatly indebted to my Thesis advisor **Dr. Karl Bruce Gregory**. Without his constant support, advice, and inspiration, I could not have completed my Master's journey. I would also like to thank **Dr. David Hitchcock** and **Dr. Minsuk Shin** for being on my committee. Dr. Gregory has worked tirelessly with me on the project, invested his time, and guided me in every stage of my graduate study. He is not only a knowledgeable person but also a wise empathetic human being. Having him as my thesis supervisor is one of the serendipitous events of my life.

If it weren't for my parents' (**Haruna Akhter Parul** and **M.A. Mannan**) constant inspiration, support, and love, I wouldn't be able to embark on this journey of pursuing my graduate education abroad. They and my younger brother **Intisar Ibn Mannan** have been ever-so-present bedrocks in my life and I am incredibly blessed to call them my family.

I would like to convey my gratitude to my undergrad professors from the University of Dhaka **Dr. Jafar Ahmed Khan** for piquing my interest in the LASSO models in the first place. I was able to espouse the early concepts of Statistics and Probability through his passionate introductory classes in the freshman year. I would also like to thank **Dr. Wasimul Bari** for some of the finest classes in robust statistics, generalized linear models, and survival analysis. I know I would be totally lost in some of those advanced topics if it were not for his kind yet involved way of teaching. Finally, my heart reaches out to belated **Dr. Taslim Sazzad Mallick** for his enormous efforts in teaching the fundamental concepts of hypothesis testing and introductory survival analysis. You will be missed, Sir. I am forever in debt to

these extraordinary human beings for their unparalleled support and encouragement which made my undergraduate studies interesting. I had some of the best one-to-one conversations with them which infused me with enthusiasm and fortitude towards pursuing the unknown in the realm of statistics, science, and life in general.

After coming to UofSC, I have met some fantastic professors and graduate students. I enjoyed interacting with **Dr. John Grego, Dr. Lianming Wang, Dr. Xianzheng Huang** and the department chair **Dr. Joshua Tebbs**. I want to especially thank my friends and colleagues from UofSc Statistics **Zichen, Ryan, Nubaira, Clover, Jackie, Kevin, Jihyun** and my roommate and friend from UofSC Epidemiology **Longgang**. Additionally, I had a great time teaching Statistics courses in the department and would like to thank **Mr. Muhammad Quasem, Ms. Wilma Sims** and **Ms. Kathryn Dobereiner** for their guidance and help in that regard. I would also like to thank my friends **Aupi, Arman, Sabbir, Dhiman, Shoumik, Saqib, Emi, Iqbal, Ifa, Nowrin, Sadia, Hridita, Tareq, Fahim, Mishma, Dipro, Eashna**, and **Samia** from the Bangladeshi community for their endless support in my life and stay at Columbia. Lastly, I would like to convey my earnest gratitude to **Dr. Tahmidul Islam**, a former graduate student of the Department of Statistics form both DU and UofSC whom I have come to know and admire in my stay at UofSC Statistics. He has been a tremendous support and guide to me and I will be grateful to him forever.

I have been using my ASUS X510U Vivobook since August 2018 and it has provided me with uninterrupted service ever since. If it could understand words, I would have wanted to thank it too. It has been a dear friend.

Abstract

Building confidence/credible intervals for the high-dimensional ($p >> n$) linear models have been the subject of exploration for many years. In this paper, we explore three specific setups. First, we look at the Bayesian paradigm for the LASSO model. A double-exponential prior has been applied to the regression coefficient and from that, a posterior distribution is derived to get the necessary quantiles to calculate the credible intervals for the regression coefficients. Second, we explore the *de-sparsified LASSO* estimates, and using its asymptotic normality, we calculate the confidence intervals for the model coefficients. Finally, we incorporate an *adaptive LASSO* model. To calculate the confidence intervals, we have used the residual and perturbation bootstrap methods and obtained the necessary quantiles. All three methods have been put through a simulation study to compare the interval coverage of the true coefficient values. The width of the intervals is also compared. We make n, the sample size fixed, and explore the cases where we put a set of correlated covariates as true values. The considered number of covariates, p includes 200, 500, and 1000. We also compare the time it takes to complete 10 runs of each setup on a personal computer. We assume two kinds of correlation structures for data. We call the first one AR-1 and the second one is known as *compound symmetry*. For AR-1 cases, when $p >> n$, the Bayesian LASSO provides better coverage for true non-zero coefficient, especially if the correlation is close to 0.9 and 0.5. For compound symmetry cases, the de-sparsified LASSO seems to provide closer to the nominal coverage regardless of the value of correlation coefficient, ρ. However, the coverage is around 0.90 for $p = 1000$. This better coverage comes with the cost of getting wider intervals for highly corre-

lated cases. For the moderate correlation, the intervals by de-sparsified LASSO are even smaller for the true non-zero coefficients. The bootstrap generated intervals for adaptive LASSO tend to provide coverage around 0.90, in the uncorrelated cases, regardless of the number of covariates in the model. But they appear to suffer when the predictors are highly correlated. If the correlation is low and the number of predictors is not too much greater than the sample size, perturbation bootstrap provides close to nominal coverage. In addition, the adaptive LASSO with perturbation bootstrap typically achieves faster calculation time in most cases of our simulation setup. In conjunction with the simulation study, we illustrate the aforementioned methods for building confidence/credible intervals on two real datasets.

Table of Contents

List of Tables

Chapter 1

Introduction

In the classical linear regression model, we have the response variable, e.g. Y regressed on one or more covariates or predictor variables denoted by $X_1, \ldots, X_p$. The corresponding regression coefficients for these variables are denoted by $\beta_1, \ldots, \beta_p$. We observe n realizations of Y and $X_1, \ldots, X_p$ based on which we wish to estimate $\beta_1, ..., \beta_p$. Let, Y_i represent the value of the response variable Y for realization i, and x_{ij} the value of X_j for realization i, for $j = 1, \ldots, p$ and $i = 1, \ldots, n$.

Define the response vector $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^T$, the matrix containing the predictors $\boldsymbol{X} = [\boldsymbol{x_1}, \ldots, \boldsymbol{x_p}]$ (where each $\boldsymbol{x_j} = (x_{1j}, \ldots, x_{nj})^T$ is a column entry), and the coefficient vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$. We assume that the columns of $\boldsymbol{X}$ and the response values are centered so that $\sum_{i=1}^n x_{ij} = 0$ and $\sum_{i=1}^n Y_i = 0$. Then, define the classical regression model:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{1.1}$$

where $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^T$ is the error term vector; they are independent and non-zero and each of them are identically distributed with zero mean and fixed variance. Therefore, the ordinary least squares (OLS) estimates are given by the estimator:

$$\underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left[||\boldsymbol{Y} - \sum_{j=1}^{p} \beta_j \boldsymbol{x_j}||^2 \right]$$

In the matrix notation, the estimator can be written as:

$$\underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} [(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^T (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})]$$

In the ordinary least square (OLS) case where the number of observations is greater than the number of predictors ($n > p$), the solution for the predictor vector $\boldsymbol{\beta}$ is $(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$.

The motivation for this endeavour stems from the difficulty in building the confidence interval for model coefficients (β_j's) in linear regression models when the number of predictors p is greater than the number of observations n i.e. $p > n$. The problem with this setup in classical linear regression is that we do not have unique solutions to the least squares estimates of β_j anymore. This is because if we apply the ordinary least square solution $\hat{\boldsymbol{\beta}}_{OLS} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$ in the $p > n$ setting, the matrix $(\boldsymbol{X}^T\boldsymbol{X})$ is not non-singular anymore.

In order to deal with this problem, we resort to some modern regression estimators. This class of estimators are called penalized regression estimators where we want to penalize the OLS estimators of $\beta_1, \ldots, \beta_p$. This way, we could address the problem of singularity of the matrix $(\boldsymbol{X}^T\boldsymbol{X})$. But in doing so, the estimators become biased as a side-effect.

LASSO (Least Absolute Shrinkage and Selection Operator) and Ridge are some of the most well-known estimators which operate in the aforementioned way. We will particularly focus on LASSO due to the fact that it has the property of setting some of the coefficient estimates to *exactly zero*. Thus, we have a way towards achieving a more interpretable model, especially in $p > n$ cases where we might want to know which covariates are significantly associated with the response. This leads to a *parsimonious* model. Simultaneously, it also aids us with the ability of selecting relevant variables in the model.

In this paper, we will look at some special ways for performing inference based on the LASSO estimators e.g. through Bayesian methods, de-sparsified LASSO methods, and using bootstrap methods for adaptive LASSO. But before going there, let's have a primer on the LASSO in general.

LASSO was first introduced by Robert Tibshirani in 1994 (Tibshirani (1994)).

We define the LASSO estimator $\hat{\boldsymbol{\beta}}^L$ of $\boldsymbol{\beta}$ as the solution to the minimization problem:

$$\text{minimize } \{\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p}\beta_j x_{ij})^2\}$$

$$\text{subject to } \textstyle\sum_{j=1}^{p}|\beta_j| \leq t$$

where $t \geq 0$ is a tuning parameter.

It can be equivalently written in *Lagrangian* form (Gill, Murray, and Wright (1981)) (Hastie, Tibshirani, and Friedman (2009)) as:

$$\hat{\boldsymbol{\beta}}^L(\lambda) = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left[\left\{ \frac{1}{2}\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p}\beta_j x_{ij})^2 \right\} + \lambda \sum_{j=1}^{p}|\beta_j| \right], \tag{1.2}$$

where $\lambda > 0$ is a shrinkage parameter. The value of λ is chosen using cross-validation for which the resulting λ gives out the lowest prediction error.

When we choose $\lambda = 0$, which we can only do in $p < n$ cases, $\hat{\boldsymbol{\beta}}^L$ becomes $\hat{\boldsymbol{\beta}}^{OLS}$ which is the ordinary least squares estimate.

We do not have a closed form solution of the LASSO estimates.

Chapter 2

LASSO Estimates and Building Their CIs

Since the LASSO minimization problem does not have a closed form solution, we have to resort to some iterative methods to minimize the objective function in (1.2) to obtain the LASSO estimates. Some of the most efficient algorithms to compute the estimates are LARS and Coordinate Descent. We will discuss them briefly about them before moving on to the question of building the confidence intervals based on the estimates.

2.1 LARS

LARS (Least Angle Regression) was proposed as a model selection algorithm by Efron, Hastie, Johnstone, and Tibshirani (2004). In that paper, they showed that this algorithm was able to calculate the LASSO estimates more efficiently than the previously popular forward selection methods. This algorithm requires only p steps to solve a LASSO problem. It starts by standardizing the predictors and then all the predictor coefficients β_js are set to zero. At this point, residuals are calculated and are kept track of. Then the predictor $\boldsymbol{x}_j$ most correlated with the response is calculated. Then, the largest step possible in the direction of this predictor is taken until some other predictor, say $\boldsymbol{x}_k$, is found which has as much correlation with the current residual. Afterwards, these two coefficients are moved towards the direction defined by their joint least squares coefficient of the current residuals on the model fitted by these two predictors $(\boldsymbol{x}_j, \boldsymbol{x}_k)$, until some other competing variable $\boldsymbol{x}_l$ comes in. In moving, if a non-zero coefficient hits zero, it is dropped and the current joint

least squares direction is recomputed. This is done for until p steps to arrive at the full LASSO solution.

2.2 Coordinate Descent

Coordinate wise descent is another fast popular method of calculating LASSO estimates. According to Friedman, Hastie, and Tibshirani (2010), this algorithm only updates one variable in each step while the rest of the predictors are regarded as constant. Since, the objective function in (1.2) is not differentiable with respect to β_j, this algorithm uses subdifferentials to calculate the solutions. For the j^{th} and the k^{th} covariates, we could re-write the cost function in equation (1.2) after differentiation with respect to β_j as:

$$-\sum_{i=1}^{n} x_{ij}\left[Y_i - \sum_{k\neq j}^{p} \beta_k x_{ik}\right] + \beta_j \sum_{i=1}^{n}(x_{ij})^2 + \partial_{\beta_j}\lambda|\beta_j|, \tag{2.1}$$

where ∂_{β_j} is the subdifferential with respect to β_j and thus

$$\partial_{\beta_j}\lambda|\beta_j| \in \begin{cases} \{-\lambda\} & \text{if } \beta_j < 0 \\ [-\lambda, \lambda] & \text{if } \beta_j = 0 \\ \{\lambda\} & \text{if } \beta_j > 0. \end{cases}$$

Let $\rho_j = \sum_{i=1}^{n} x_{ij}[Y_i - \sum_{k\neq j}^{p} \beta_k x_{ik}]$ and $z_j = \sum_{i=1}^{n}(x_{ij})^2$.

Finally, equating (2.1) to 0 and solving for β_j, we can obtain

$$\hat{\beta_j}^{L}(\lambda) = \begin{cases} \frac{\rho_j + \lambda}{z_j} & \text{if } \rho_j < -\lambda \\ 0 & \text{if } -\lambda \leq \rho_j \leq \lambda \\ \frac{\rho_j - \lambda}{z_j} & \text{if } \rho_j > \lambda. \end{cases}$$

This is called *soft thresholding* in the literature. Therefore, we could summarize the coordinate descent update rule as follows. We start with all p covariates. Then

we initialize with some value of $\boldsymbol{\beta}$ which we could call $\boldsymbol{\beta^0}$. Based on that, we compute the $\boldsymbol{\rho}, \boldsymbol{z_j}$ that will lead us to calculate $\hat{\beta}_j$. We repeat this for all p covariates. Thus we populate $\boldsymbol{\beta^1}$ for the first iteration. We repeat this process until t^{th} iteration until convergence. The explanation of this part is largely based on Sicotte (2018).

2.3 Problems with Building CIs

Now that we have an idea about the LASSO estimator and how it is computed, we shift our focus to some of the ways we could make inference about the regression coefficients based on the LASSO estimates.

We are particularly interested in exploring ways to construct confidence or credible intervals based on the LASSO estimates. The problem is LASSO-type estimators typically do not have a "nice looking" asymptotic distribution based on which confidence intervals could be made. In other words, we do not have a Normal distribution for these estimates in contrast to the least square estimates where we could have the property

$$\frac{\sqrt{n}(\hat{\beta_j}^{OLS} - \beta_j)}{\hat{\sigma}\sqrt{[(\boldsymbol{X}^T\boldsymbol{X})^{-1}]_{jj}}} \sim t_{n-p},$$

when $\hat{\sigma}^2 = \frac{1}{n-p}\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p}\hat{\beta_j}^{OLS}x_{ij})^2$, and when $p < n$ and $\hat{\beta_j}^{OLS}$ is the j^{th} entry of $(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$ and when $\epsilon_1, \ldots, \epsilon_n$ are iid $\mathcal{N}(0, \sigma^2)$.

Fu and Knight (2000) showed an asymptotic distribution for the LASSO estimates which is not Normal, and therefore it is difficult to compute intervals for the estimates based on that distribution.

We will discuss three methods mentioned earlier to construct confidence/credible intervals in high-dimensional setting and then set up a simulation study to compare the width and the coverage of those intervals generated by them. Then we illustrate the application of those methods on two real datasets.

Chapter 3

Bayesian LASSO

In Tibshirani (1994), there was a mention of LASSO as a Bayes estimate. He considered that the j^{th} covariate of the model, $|\beta_j|$ was proportional to the (minus) log-density of the double-exponential distribution. Exploiting this, the LASSO estimate could be derived as the Bayes posterior mode under independent double-exponential priors for the β_js, given by

$$f(\beta_j) = \frac{1}{2\tau} \exp\left(-\frac{|\beta_j|}{\tau}\right),$$

with $\tau = \frac{1}{\lambda}$.

In that article, the LASSO estimate was found to be the maximizer of the posterior mode with the penalty term being $2\tau\sigma^2$ where $\tau > 0$ and $\sigma^2 > 0$.

The Park and Casella (2008) article discussed about the possibility of Gibbs sampling from the posterior distribution of the regression parameters. In that paper, prior distributions on τ and σ^2 were considered as well as $\boldsymbol{\beta}$; for $\boldsymbol{\beta}$ the article also considered independent Laplace (double-exponential) prior. Similar to the Tibshirani (1994) setting, they used the median of the posterior distribution to calculate the point estimates of $\boldsymbol{\beta}$.

The Hans (2009) paper introduced a direct derivation of the posterior distribution $p(\boldsymbol{\beta}|\boldsymbol{Y}, \tau, \sigma^2)$. With the use of a Gibbs sampler, this method provided a process to provide a way to sample from the posterior distribution.

In this paper, we will recreate the derivation of the posterior for $\boldsymbol{\beta}$ and outline the details of the Gibbs sampler following Hans (2009).

3.1 LASSO Estimator of $\boldsymbol{\beta}$

In matrix notation, the Bayesian LASSO estimate can be defined as a minimizer of the following form.

$$\hat{\boldsymbol{\beta}}^L(\lambda) = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}) + \lambda||\boldsymbol{\beta}||_1 \tag{3.1}$$

where $\lambda > 0$

3.2 Bayesian Setup of Hans (2009)

We assume the response vector $\boldsymbol{Y}$ has the distribution such that

$$P(\boldsymbol{Y}|\boldsymbol{\beta},\tau,\sigma^2) = (\frac{1}{\sqrt{2\pi}})^n(\frac{1}{\sigma^2})^{n/2}\exp[-\frac{1}{2\sigma^2}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})],$$

and we put a double-exponential prior on $\boldsymbol{\beta}$, i.e.

$$P(\boldsymbol{\beta}|\tau,\sigma^2) = (\frac{\tau}{2\sigma})^p\exp(\frac{\tau}{2\sigma}||\boldsymbol{\beta}||_1)$$

The posterior distribution can then be derived from the conditional distribution property as follows. Using

$$f(x|y) = \frac{f(x,y)}{f(y)},$$

we can write

$$P(\boldsymbol{\beta}|\boldsymbol{Y},\sigma^2,\tau) = \frac{P(\boldsymbol{Y}|\boldsymbol{\beta},\sigma^2,\tau)P(\boldsymbol{\beta}|\sigma^2,\tau)}{\underbrace{\int P(\boldsymbol{Y}|\boldsymbol{\beta},\sigma^2,\tau)P(\boldsymbol{\beta}|\sigma^2,\tau)\,d\boldsymbol{\beta}}_{P(\boldsymbol{Y}|\sigma^2,\tau)\text{ which can be treated as a constant.}}}$$

$$\begin{aligned}
&\propto P(\boldsymbol{Y}|\boldsymbol{\beta},\tau,\sigma^2)P(\boldsymbol{\beta}|\tau,\sigma^2)\\
&= (\frac{1}{\sqrt{2\pi}})^n(\frac{1}{\sigma^2})^{n/2}\exp[-\frac{1}{2\sigma^2}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})](\frac{\tau}{2\sigma})^p\exp(\frac{\tau}{2\sigma}||\boldsymbol{\beta}||_1)\\
&\propto \exp[-\frac{1}{2\sigma^2}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}) - \frac{\tau}{\sigma}||\boldsymbol{\beta}||_1]
\end{aligned}$$

The mode of $P(\boldsymbol{\beta}|\boldsymbol{Y}, \tau, \sigma^2)$, that is the posterior mode, is $\hat{\boldsymbol{\beta}}^L(\lambda = 2\tau\sigma)$ for the Bayes rule under the zero-one loss function, according to the discussion by Hans (2009).

Using a Gibbs sampler Casella and George (1992), we sample $\boldsymbol{\beta}$ from this posterior distribution and eventually calculate the 2.5th and 97.5th percentiles of the posterior marginal distribution of each β_j to construct the credible intervals for each regression coefficient.

We created a Gibbs Sampler using the process mentioned in Hans (2009) and the codes are included in the Appendix.

Chapter 4

De-sparsified LASSO

Geer, Bühlmann, Ritov, and Dezeure (2014) proposed an estimator which they called the *de-sparsified LASSO* for the $p > n$ regression setting. Under some conditions, they showed that their proposed estimator of each regression coefficient attains asymptotically normal distribution allowing the construction of confidence intervals. At the same time, Zhang and Zhang (2014) proposed the same kind of estimator independently which was developed based on a relaxed covariance matrix; this estimators also had an asymptotic normal distribution. Here, we will discuss the development of the estimator proposed by Geer et al. (2014) briefly.

Considering the notations described for (1.1), let $\hat{\boldsymbol{\Theta}}$ be a "relaxed form" of an inverse of the observed covariance matrix $\hat{\boldsymbol{\Sigma}} = \frac{1}{n}\boldsymbol{X}^T\boldsymbol{X}$. First, they obtained the biased estimator $\tilde{\boldsymbol{\beta}}$ by replacing the inverse of $\frac{1}{n}\boldsymbol{X}^T\boldsymbol{X}$ with $\hat{\boldsymbol{\Theta}}$ in the OLS estimator. So the biased estimate can be obtained as follows.

$$\begin{aligned}
\tilde{\boldsymbol{\beta}} &= \frac{1}{n}\hat{\boldsymbol{\Theta}}\boldsymbol{X}^T\boldsymbol{Y} \\
&= \frac{1}{n}\hat{\boldsymbol{\Theta}}\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}) \\
&= \hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}}\boldsymbol{\beta} + \frac{1}{n}\hat{\boldsymbol{\Theta}}\boldsymbol{X}^T\boldsymbol{\epsilon} \\
&= \boldsymbol{\beta} + (\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})\boldsymbol{\beta} + \frac{1}{n}\hat{\boldsymbol{\Theta}}\boldsymbol{X}^T\boldsymbol{\epsilon}
\end{aligned}$$

Next, the de-sparsified estimate of $\boldsymbol{\beta}$, $\hat{\boldsymbol{\beta}}^{desp}$ is obtained by using $\hat{\boldsymbol{\beta}}^L$ to modify $\hat{\boldsymbol{\beta}}$ as follows. Here, the term $(\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})\hat{\boldsymbol{\beta}}^L$ is considered as an estimate of the bias.

$$\begin{aligned}\hat{\boldsymbol{\beta}}^{desp} &= \tilde{\boldsymbol{\beta}} - (\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})\hat{\boldsymbol{\beta}}^{L} \\ &= \boldsymbol{\beta} + (\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{L}) + \frac{1}{n}(\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})\boldsymbol{X}^{T}\boldsymbol{\epsilon}\end{aligned}$$

In this form at the right-hand side, $(\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{L})$ can be considered the bias term and $\frac{1}{n}(\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})\boldsymbol{X}^{T}\boldsymbol{\epsilon}$ can be considered as the variance term.

Geer et al. (2014) showed that the j^{th} component of $\hat{\boldsymbol{\beta}}^{desp}$, that is $\hat{\beta_j}^{desp}$ has an asymptotic normal distribution by writing

$$\sqrt{n}(\boldsymbol{e_j}^{T}\hat{\boldsymbol{\beta}}^{desp} - \boldsymbol{e_j}^{T}\boldsymbol{\beta}) = \sqrt{n}\boldsymbol{e_j}^{T}(\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}} - \boldsymbol{I})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{L}) + \frac{1}{\sqrt{n}}\boldsymbol{e_j}^{T}\hat{\boldsymbol{\Theta}}\boldsymbol{X}^{T}\boldsymbol{\epsilon}$$

and giving conditions under which the bias term becomes negligible as $n \to \infty$. Afterwards, the asymptotic distribution is derived to be:

$$\sqrt{n}(\hat{\beta_j}^{desp} - \beta_j) \dot{\sim} \mathcal{N}(0, \frac{1}{n}\boldsymbol{e_j}^{T}\hat{\boldsymbol{\Theta}}\boldsymbol{X}^{T}\boldsymbol{\sigma}^{2}\boldsymbol{I}_{n}\boldsymbol{X}\hat{\boldsymbol{\Theta}}\boldsymbol{e_j})$$

Using an estimate of this variance as $\hat{\sigma}^2(\hat{\boldsymbol{\Theta}}[\frac{1}{n}\boldsymbol{X}^{T}\boldsymbol{X}]\hat{\boldsymbol{\Theta}})_{jj}$, where $\hat{\sigma}$ is the estimate of σ, we could build a $(1-\alpha)100\%$ confidence interval based on this asymptotic normal distribution in the form:

$$\hat{\beta_j}^{desp} \pm Z_{\alpha/2}\frac{\hat{\sigma}}{\sqrt{n}}\sqrt{(\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{\Theta}})_{jj}}$$

where $Z_{\alpha/2}$ is the upper $\alpha/2$ quantile of the standard normal distribution and $(\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{\Theta}})_{jj}$ is the entry in the j^{th} row and j^{th} column of the $\hat{\boldsymbol{\Theta}}\hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{\Theta}}$ matrix.

This property allows us to build confidence intervals for each of the coefficents i.e. for each β_j.

Chapter 5

Adaptive LASSO with Bootstrap

The adaptive LASSO provides the estimates and active variable selection simultaneously. Zou (2006) showed that assigning different weights to penalize different coefficients leads to obtaining adaptive LASSO estimators. Considering $\hat{w}_j = \frac{1}{|\tilde{\beta}_j^L|}$, the estimator defined by

$$\underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left[||\boldsymbol{Y} - \sum_{j=1}^{p} \beta_j \boldsymbol{x}_j||^2 + \boldsymbol{\lambda}_n \sum_{j=1}^{p} \hat{w}_j |\beta_j| \right] \tag{5.1}$$

The adaptive LASSO estimates of the non-zero coefficients has the asymptotic property of converging to the normal distribution but usually the convergence is slow. Hence, we used the bootstrap method to derive the necessary intervals.

We could build the intervals for the adaptive LASSO estimates using (i) residual bootstrap and (ii) perturbation bootstrap methods using the following steps.

First, we calculate the adaptive LASSO estimate, $\tilde{\boldsymbol{\beta}}^{AL}$. Estimating adaptive LASSO estimate is a two-step process. We need an initial calculation of LASSO estimate. For that, we have to calculate λ through cross-validation. Thus, initial $\tilde{\beta}_j^L$s are obtained. This completes the first step. After that, using the weights, $\hat{w}_j = \frac{1}{|\tilde{\beta}_j^L|}$, the $\tilde{\beta}_j^{AL}$s are estimated by choosing appropriate λ again through cross-validation using the aforementioned objective function. This completes the second step and thus we obtain the adaptive LASSO estimate $\tilde{\beta}_j^{AL}$. Then we obtain the estimated residuals, $\hat{\epsilon}_i = Y_i - \sum_{j=1}^{p} x_{ij} \tilde{\beta}_j^{AL}$ where $i = 1, \ldots, n$. These estimated residuals are employed in two ways to obtain the bootstrap samples. For (i) the residual bootstrap samples, the estimated residuals $\hat{\epsilon}_i$s are resampled to generate $\hat{\epsilon}_i^*$s the bootstrapped residual

samples. For (ii) perturbation bootstrap samples, we set $\hat{\epsilon}_i^* = \hat{\epsilon}_i * 4(U_i - 1/4)$ where $U_i \sim Beta(1/2, 3/2); i = 1, \ldots, n$ according to Das, Gregory, and Lahiri (2019).

The next steps of calculating the intervals are same for both the residual and perturbation bootstrap methods.

Therefore, using $\hat{\epsilon}_i^*$s, we compute new bootstrapped residual generated response values such as: $Y_i^* = \sum_{j=1}^{p} x_{ij}\tilde{\beta}_j^{AL} + \hat{\epsilon}_i^*$. Finally, replacing $\boldsymbol{Y}$ with the vector of bootstraps response vectors $\boldsymbol{Y}^* = (\boldsymbol{Y}_1^*, \ldots, \boldsymbol{Y}_n^*)^T$, we can obtain $\tilde{\boldsymbol{\beta}}^{AL^*}$ the same way mentioned above. Repeating this, we get the necessary bootstrap samples $\tilde{\boldsymbol{\beta}}^{AL^*}$s which let us build the intervals using the 2.5th and 97.5th percentile.

Chapter 6

Simulation Study

6.1 Overview

The goal of this study is to use simulated scenarios to compare the performance of the three methods — Bayesian LASSO, de-sparsified LASSO, and the residual and perturbation bootstrap for the adaptive LASSO — for constructing confidence intervals of the regression coefficients of a high-dimensional linear regression model.

We consider two sets of true coefficient vectors for the simulation setups. In the first cases, the true coefficient vector $\boldsymbol{\beta}$ was set up having the first three values as 1 and the rest 0 i.e. $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$. In the second cases, the true coefficient vector $\boldsymbol{\beta}$ was set up having the first five values as $5, -4, 3, -2, 1$ and the rest 0 i.e. $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$. Apart from this, we needed three other components for our each run of the simulations for each method. We required the design matrix $\boldsymbol{X}$ and the error vector $\boldsymbol{\epsilon}$ to be generated first and then using the following formula, we would generate the response vector, $\boldsymbol{Y}$.

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{6.1}$$

Let, p be the number of independent covariates, and n be the number of observations.

For the design matrix $\boldsymbol{X}$, we randomly generated $n * p$ observations from a standard normal distribution to initially populate an $n \times p$ matrix. Then the matrix was column-centered, i.e. the centering on the matrix was done by subtracting the column means from their corresponding column elements. But we wanted to intro-

duce a specified correlation among the variables (columns) of our ultimate design matrix. To do so, we took the Cholesky root of the covariance matrix generated by (i) $\boldsymbol{\Sigma} = (\rho^{|i-j|})_{1\leq i,j\geq p}$, and by (ii) $\boldsymbol{\Sigma} = (\rho^{\mathbb{I}(i\neq j)})_{1\leq i,j\geq p}$ and matrix-multiplied $\boldsymbol{\Sigma}$ to the column-centered $\boldsymbol{X}$ matrix. For the rest of this paper, we will call setting (i) to have a covariance of type *AR-1*. In case of setting (ii), the data is assumed to have *compound symmetry* and we will call this setting to have *CS*. Setting up the covariance matrices in this way led us to generate an $n \times p$ design matrix where we could control the magnitude of correlation among the variables with just changing the value of ρ.

The error terms vector was generated following $\boldsymbol{\epsilon} \sim \boldsymbol{\mathcal{N}(0_{n\times 1}, \mathcal{I}_n)}$ and were centered by subtracting the mean.

We went through 500 simulated datasets under each combination of p and ρ setting $p = 200, 500, 1000$ and setting $\rho = 0, 0.5, 0.9$. These settings were considered for the true values $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$ and $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$ with them having AR-1 structure. This process was done for Bayesian LASSO, de-sparsified LASSO, and adaptive LASSO methods. Then, for true value of $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$ having CS structure, we went through 500 simulated datasets under each combinations of $p - \rho$ setting $p = 200, 500, 1000$ and setting $\rho = 0.5, 0.9$. $\rho = 0$ under CS was not considered here because it would be a similar setup to $\rho = 0$ under AR-1.

In terms of correlations, $\rho = 0$ signified that the variables for the simulations were uncorrelated and $\rho = 0.9$ indicated a high correlation. The number of observations n was fixed to be 100 for every dataset.

6.2 Bayesian LASSO

Although we had developed a Gibbs Sampler with our own R-scripts to calculate the *Bayesian LASSO*-generated credible intervals, for efficiency, we implemented the **blasso** function from the **monomvn** R package (Gramacy and Fortran contributions from Cleve Moler as updated by Berwin A. Turlach (qpgen2/quadprog) (2019)). The

package-default for the penalty parameter (λ) was 1 and with that, 1000 MCMC (Markov Chain Monte Carlo) samples were generated. Calculating the 2.5th and 97.5th percentile of these samples gave us a CI for the underlying simulated dataset in this iteration.

6.3 De-sparsified LASSO

For *de-sparsified LASSO*, we considered the **lasso.proj** function from the **hdi** R package (Dezeure, Bühlmann, Meier, and Meinshausen (2015)). Since this estimates of $\boldsymbol{\beta}$ derived by this method had an asymptotic normal distribution, we were able to obtain a 95% CI for the regression coefficients for the underlying simulated dataset.

6.4 Adaptive LASSO

For the *adaptive LASSO* method, we used the **hidbootreg** R package (Gregory (2020)). We calcualted 1000 Monte Carlo resamples and used the residual and perturbation bootstrap methods to generate 95% confidence intervals from the resamples for the underlying simulated dataset. The **hidbootreg** package used another package called **ncvreg** which provided relatively faster result for the cross-validation procedure to calculate the initial values of the penalty parameters.

6.5 Calculation of the Coverage

The datasets comprising $\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{\beta}, \boldsymbol{\epsilon}$ — where $\boldsymbol{Y}$ was formulated by $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ — were simulated 500 times for each setting of $p - \rho$ combination mentioned at the end of chapter 6.1 and each time the width of the coverage was measured along with whether it was able to capture the values of the true regression coefficients under each method. We then calculated in how many (fraction of) simulated datasets, the confidence/credible intervals were able to capture the true values of regression

coefficients. We also measured the average width of the intervals as a metric to gauge how good the CIs were in comparison with the other methods.

6.6 Computation Hardware

The main bulk of the computation for Bayesian LASSO and de-sparsified LASSO was performed using University of South Carolina's Hyperion supercomputer. The computation for adaptive LASSO was done using a personal laptop computer which ran a 8th generation Intel processor (i5-8250U) with 8GB memory capacity.

Furthermore, to compare the computation time, 10 runs of each setting were performed with the same personal computer setup and the elapsed time (in minutes) have been reported.

Chapter 7

Simulation Study Results

7.1 AR-1 with True $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$

Under AR-1 for $\boldsymbol{\Sigma}$, Table 7.1 and Table 7.2 were generated by considering sample size $n = 100$ and number of covariates, $p = 200$. Table 7.1 gives the simulated coverage probabilities and Table 7.2 gives the average width of the intervals.

Table 7.1. Coverage Comparison for 200 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
1	0.77	0.94	0.93	0.90	0.90	0.92	0.92	0.92	0.95	0.87	0.53	0.59
1	0.79	0.95	0.95	0.94	0.95	0.80	0.96	0.96	0.97	0.75	0.40	0.43
1	0.77	0.95	0.94	0.91	0.91	0.91	0.93	0.92	0.94	0.86	0.52	0.54
0	0.99	0.97	0.99	0.99	1.00	0.93	1.00	1.00	0.98	0.89	0.99	1.00
0	1.00	0.97	0.99	1.00	1.00	0.96	1.00	1.00	1.00	0.94	1.00	1.00
0	1.00	0.95	0.99	1.00	0.99	0.94	1.00	1.00	1.00	0.93	1.00	1.00
0	1.00	0.97	0.99	0.99	1.00	0.96	1.00	1.00	1.00	0.92	1.00	1.00
0	1.00	0.96	0.99	1.00	1.00	0.95	1.00	1.00	1.00	0.96	1.00	1.00
0	1.00	0.95	0.99	1.00	1.00	0.97	1.00	1.00	1.00	0.94	1.00	1.00
0	0.99	0.96	1.00	1.00	1.00	0.97	1.00	1.00	1.00	0.97	1.00	1.00

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.2. Average Interval Width Comparison for 200 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
1	0.51	0.45	0.49	0.45	0.57	0.47	0.61	0.57	1.07	0.77	0.80	0.87
1	0.51	0.45	0.48	0.46	0.63	0.48	0.75	0.69	1.42	0.85	1.00	1.04
1	0.51	0.44	0.49	0.45	0.59	0.49	0.62	0.58	1.20	0.85	0.82	0.83
0	0.27	0.44	0.00	0.00	0.31	0.48	0.00	0.00	0.57	0.85	0.00	0.01
0	0.26	0.45	0.00	0.00	0.29	0.48	0.00	0.00	0.46	0.85	0.00	0.00
0	0.27	0.45	0.00	0.00	0.28	0.48	0.00	0.00	0.41	0.85	0.00	0.00
0	0.27	0.45	0.00	0.00	0.28	0.48	0.00	0.00	0.40	0.85	0.00	0.00
0	0.26	0.45	0.00	0.00	0.28	0.48	0.00	0.00	0.38	0.85	0.00	0.00
0	0.26	0.45	0.00	0.00	0.28	0.48	0.00	0.00	0.38	0.85	0.00	0.00
0	0.26	0.45	0.00	0.00	0.28	0.48	0.00	0.00	0.38	0.85	0.00	0.00

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Similarly, for AR-1, Table 7.3 and Table 7.4 give the simulated coverage probabilities and the average width of the intervals respectively under $n = 100$ and $p = 500$ for true coefficients $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$.

Table 7.3. Coverage Comparison for 500 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
1	0.76	0.93	0.89	0.88	0.90	0.92	0.92	0.91	0.95	0.87	0.53	0.59
1	0.77	0.95	0.92	0.90	0.97	0.78	0.94	0.94	0.97	0.69	0.44	0.46
1	0.77	0.93	0.91	0.88	0.92	0.90	0.89	0.90	0.92	0.80	0.54	0.51
0	0.97	0.96	0.99	1.00	0.95	0.92	1.00	1.00	0.90	0.87	0.98	0.99
0	0.97	0.95	1.00	1.00	0.97	0.94	1.00	1.00	0.97	0.93	1.00	0.99
0	0.98	0.94	1.00	1.00	0.97	0.96	1.00	1.00	0.98	0.92	1.00	1.00
0	0.96	0.94	1.00	0.99	0.98	0.96	1.00	1.00	0.98	0.92	1.00	1.00
0	0.96	0.96	1.00	1.00	0.98	0.95	1.00	1.00	1.00	0.93	1.00	1.00
0	0.97	0.96	1.00	1.00	0.98	0.96	1.00	1.00	1.00	0.93	1.00	1.00
0	0.98	0.96	1.00	1.00	0.97	0.96	1.00	1.00	1.00	0.94	1.00	1.00

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.4. Average Interval Width Comparison for 500 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
1	0.77	0.45	0.47	0.43	0.84	0.47	0.60	0.56	1.30	0.74	0.78	0.86
1	0.80	0.45	0.48	0.44	0.93	0.48	0.72	0.67	1.69	0.81	1.05	1.05
1	0.79	0.46	0.48	0.44	0.85	0.48	0.61	0.56	1.35	0.81	0.81	0.80
0	0.17	0.46	0.00	0.00	0.24	0.48	0.00	0.00	0.54	0.81	0.02	0.01
0	0.17	0.46	0.00	0.00	0.19	0.48	0.00	0.00	0.33	0.80	0.00	0.01
0	0.17	0.45	0.00	0.00	0.18	0.48	0.00	0.00	0.28	0.81	0.00	0.00
0	0.17	0.46	0.00	0.00	0.17	0.48	0.00	0.00	0.25	0.81	0.00	0.00
0	0.18	0.46	0.00	0.00	0.18	0.48	0.00	0.00	0.24	0.80	0.00	0.00
0	0.17	0.45	0.00	0.00	0.18	0.48	0.00	0.00	0.23	0.80	0.00	0.00
0	0.17	0.46	0.00	0.00	0.18	0.48	0.00	0.00	0.23	0.80	0.00	0.00

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Lastly, Table 7.5 and Table 7.6 give the simulated coverage probabilities and the average width of the intervals respectively under $n = 100$ and $p = 1000$ for true coefficients $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$.

Table 7.5. Coverage Comparison for 1000 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
1	0.74	0.94	0.90	0.88	0.93	0.91	0.91	0.89	0.94	0.87	0.55	0.50
1	0.74	0.94	0.90	0.85	0.96	0.70	0.94	0.91	0.98	0.69	0.43	0.41
1	0.70	0.95	0.90	0.87	0.93	0.89	0.90	0.90	0.95	0.85	0.53	0.48
0	0.56	0.97	1.00	1.00	0.54	0.88	1.00	0.99	0.42	0.85	0.99	0.99
0	0.57	0.95	1.00	1.00	0.51	0.96	1.00	1.00	0.51	0.89	0.99	0.99
0	0.54	0.96	1.00	0.99	0.54	0.95	1.00	1.00	0.60	0.90	1.00	1.00
0	0.58	0.95	1.00	1.00	0.59	0.94	1.00	1.00	0.56	0.93	1.00	1.00
0	0.56	0.97	1.00	1.00	0.59	0.95	1.00	1.00	0.63	0.90	1.00	1.00
0	0.57	0.95	1.00	1.00	0.50	0.94	1.00	1.00	0.63	0.94	1.00	1.00
0	0.59	0.94	1.00	1.00	0.50	0.96	1.00	1.00	0.61	0.93	1.00	1.00

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.6. Average Interval Width Comparison for 1000 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
1	0.99	0.46	0.49	0.44	1.13	0.47	0.58	0.56	1.54	0.72	0.83	0.74
1	1.01	0.46	0.47	0.43	1.31	0.48	0.72	0.66	1.95	0.78	1.04	0.95
1	1.01	0.46	0.47	0.43	1.14	0.48	0.60	0.56	1.64	0.78	0.81	0.72
0	0.09	0.46	0.00	0.00	0.20	0.48	0.00	0.00	0.75	0.78	0.01	0.01
0	0.09	0.46	0.00	0.00	0.10	0.48	0.00	0.00	0.40	0.78	0.00	0.00
0	0.09	0.46	0.00	0.00	0.09	0.48	0.00	0.00	0.25	0.78	0.00	0.00
0	0.09	0.46	0.00	0.00	0.10	0.48	0.00	0.00	0.19	0.78	0.00	0.00
0	0.09	0.46	0.00	0.00	0.10	0.48	0.00	0.00	0.15	0.78	0.00	0.00
0	0.09	0.46	0.00	0.00	0.10	0.48	0.00	0.00	0.15	0.78	0.00	0.00
0	0.10	0.46	0.00	0.00	0.09	0.48	0.00	0.00	0.13	0.78	0.00	0.00

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.7 shows the time it takes to complete 10 runs of simulations under each setup.

Table 7.7. Time Comparison of 10 runs (in minutes)

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
#p $\downarrow$	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
200	34.89	6.45	0.48	0.21	34.91	5.64	0.50	0.18	35.13	3.38	0.71	0.25
500	43.61	24.29	0.67	0.33	43.63	22.55	0.77	0.34	44.33	15.04	0.73	0.37
1000	45.94	51.20	1.40	0.63	45.69	48.98	1.56	0.65	44.16	42.98	1.46	0.62

*Time comparison for for true $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

7.2 AR-1 with True $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$

Under AR-1 for $\boldsymbol{\Sigma}$, and for true coefficients $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$, Table 7.8 and Table 7.9 were generated by considering sample size $n = 100$ and number of covariates, $p = 200$. Table 7.8 gives the simulated coverage probabilities and Table 7.9 gives the average width of the intervals.

Table 7.8. Coverage Comparison for 200 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
5	0.94	0.92	0.92	0.89	0.88	0.71	0.89	0.84	0.26	0.39	0.73	0.71
-4	0.93	0.93	0.93	0.90	0.81	0.45	0.85	0.82	0.11	0.03	0.68	0.67
3	0.94	0.94	0.95	0.88	0.82	0.44	0.86	0.83	0.10	0.04	0.51	0.46
-2	0.94	0.94	0.93	0.89	0.80	0.41	0.86	0.86	0.08	0.08	0.24	0.24
1	0.94	0.94	0.92	0.92	0.89	0.70	0.87	0.88	0.18	0.49	0.10	0.12
0	0.99	0.95	0.99	0.99	0.98	0.95	0.98	0.99	1.00	0.94	0.92	0.92
0	0.98	0.96	0.99	1.00	0.99	0.95	0.98	0.97	1.00	0.94	0.97	0.96
0	0.98	0.96	0.99	1.00	0.99	0.96	0.98	0.98	1.00	0.95	0.97	0.96
0	0.99	0.97	0.99	1.00	0.98	0.97	0.98	0.99	1.00	0.96	0.98	0.97
0	0.98	0.97	0.99	0.99	1.00	0.95	0.99	0.97	1.00	0.95	0.98	0.96

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.9. Average Interval Width Comparison for 200 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
5	0.59	0.46	0.39	0.36	0.72	0.51	0.46	0.41	2.05	0.95	1.13	0.98
-4	0.59	0.46	0.39	0.37	0.81	0.53	0.51	0.46	2.68	1.05	1.46	1.26
3	0.59	0.46	0.39	0.37	0.81	0.53	0.52	0.47	1.93	1.05	1.17	1.01
-2	0.59	0.46	0.41	0.38	0.82	0.53	0.57	0.51	1.23	1.05	0.45	0.43
1	0.60	0.46	0.47	0.44	0.77	0.53	0.55	0.50	0.86	1.05	0.16	0.17
0	0.37	0.46	0.00	0.00	0.45	0.53	0.02	0.02	0.69	1.05	0.06	0.05
0	0.36	0.46	0.00	0.00	0.42	0.53	0.01	0.01	0.63	1.05	0.03	0.02
0	0.37	0.46	0.00	0.00	0.41	0.53	0.01	0.01	0.62	1.05	0.01	0.03
0	0.36	0.46	0.00	0.00	0.41	0.53	0.00	0.00	0.61	1.05	0.01	0.02
0	0.37	0.46	0.00	0.00	0.42	0.53	0.01	0.01	0.60	1.05	0.02	0.02

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

And under AR-1 for $\mathbf{\Sigma}$, and for true coefficients $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$, Table 7.10 and Table 7.11 were generated by considering sample size $n = 100$ and number of covariates, $p = 500$. Table 7.10 gives the simulated coverage probabilities and Table 7.11 gives the average width of the intervals.

Table 7.10. Coverage Comparison for 500 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
5	0.95	0.93	0.89	0.88	0.86	0.51	0.76	0.66	0.03	0.31	0.08	0.05
-4	0.96	0.90	0.86	0.88	0.73	0.15	0.68	0.61	0.01	0.00	0.05	0.03
3	0.95	0.91	0.92	0.85	0.72	0.17	0.65	0.58	0.00	0.05	0.02	0.00
-2	0.97	0.93	0.89	0.87	0.70	0.19	0.66	0.59	0.00	0.09	0.00	0.00
1	0.96	0.92	0.88	0.86	0.84	0.47	0.73	0.69	0.02	0.81	0.01	0.01
0	0.97	0.96	1.00	1.00	0.98	0.95	0.99	0.96	0.98	0.94	0.99	0.99
0	0.98	0.97	0.99	0.99	0.98	0.97	0.99	0.98	0.99	0.93	1.00	0.99
0	0.98	0.96	1.00	1.00	0.99	0.96	0.99	0.99	0.99	0.93	1.00	1.00
0	0.98	0.96	1.00	0.99	0.98	0.96	0.99	0.98	1.00	0.94	1.00	1.00
0	0.96	0.96	0.99	0.99	0.99	0.97	0.99	1.00	0.99	0.95	0.99	0.99

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.11. Average Interval Width Comparison for 500 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
5	2.78	0.47	0.39	0.36	2.99	0.55	0.46	0.40	2.56	1.26	0.99	0.94
-4	2.55	0.47	0.39	0.36	3.28	0.57	0.53	0.47	1.63	1.36	0.19	0.11
3	2.18	0.47	0.39	0.36	2.74	0.57	0.54	0.48	1.11	1.36	0.11	0.05
-2	1.79	0.47	0.40	0.37	1.98	0.57	0.59	0.51	0.48	1.37	0.00	0.00
1	1.13	0.47	0.46	0.42	1.12	0.57	0.53	0.44	0.47	1.37	0.02	0.01
0	0.28	0.47	0.00	0.00	0.35	0.57	0.02	0.02	0.40	1.37	0.01	0.00
0	0.28	0.47	0.00	0.00	0.31	0.57	0.01	0.01	0.37	1.37	0.00	0.00
0	0.28	0.47	0.00	0.00	0.30	0.57	0.00	0.00	0.34	1.37	0.00	0.00
0	0.28	0.47	0.00	0.00	0.29	0.57	0.01	0.01	0.33	1.37	0.00	0.00
0	0.28	0.47	0.00	0.00	0.30	0.57	0.01	0.00	0.33	1.37	0.00	0.00

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Lastly, under AR-1 for $\mathbf{\Sigma}$, and for true coefficients $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$, Table 7.12 and Table 7.13 were generated by considering sample size $n = 100$ and number of covariates, $p = 1000$. Table 7.12 gives the simulated coverage probabilities and Table 7.13 gives the average width of the intervals.

Table 7.12. Coverage Comparison for 1000 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
5	0.93	0.91	0.86	0.82	0.71	0.38	0.56	0.50	0.02	0.29	0.00	0.00
-4	0.93	0.89	0.85	0.84	0.49	0.08	0.46	0.42	0.00	0.00	0.00	0.00
3	0.93	0.92	0.85	0.83	0.46	0.07	0.45	0.44	0.00	0.05	0.00	0.00
-2	0.91	0.90	0.85	0.84	0.40	0.09	0.40	0.36	0.00	0.07	0.00	0.00
1	0.86	0.92	0.85	0.83	0.55	0.35	0.42	0.42	0.07	0.91	0.00	0.00
0	0.54	0.95	1.00	1.00	0.56	0.95	0.98	0.97	0.59	0.92	0.99	1.00
0	0.62	0.97	0.99	1.00	0.57	0.98	0.98	1.00	0.61	0.93	1.00	1.00
0	0.58	0.96	1.00	1.00	0.58	0.98	1.00	1.00	0.65	0.94	1.00	1.00
0	0.61	0.98	1.00	0.99	0.58	0.99	0.99	0.99	0.65	0.94	1.00	1.00
0	0.59	0.95	1.00	0.99	0.60	0.98	0.99	0.99	0.65	0.95	1.00	1.00

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.13. Average Interval Width Comparison for 1000 Covariates

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
5	4.49	0.48	0.38	0.36	4.51	0.64	0.50	0.42	2.94	1.26	0.98	0.95
-4	3.85	0.48	0.38	0.36	3.83	0.66	0.60	0.51	1.24	1.36	0.00	0.00
3	3.05	0.48	0.38	0.36	2.85	0.66	0.63	0.52	1.07	1.36	0.00	0.01
-2	2.17	0.49	0.40	0.36	1.81	0.66	0.63	0.52	0.34	1.36	0.00	0.00
1	1.16	0.49	0.43	0.41	0.95	0.66	0.37	0.34	0.38	1.36	0.00	0.00
0	0.20	0.49	0.00	0.00	0.26	0.66	0.02	0.02	0.27	1.36	0.00	0.00
0	0.21	0.48	0.00	0.00	0.19	0.66	0.01	0.00	0.22	1.36	0.00	0.00
0	0.18	0.49	0.00	0.00	0.20	0.66	0.00	0.00	0.20	1.36	0.00	0.00
0	0.20	0.49	0.00	0.00	0.19	0.66	0.00	0.00	0.19	1.36	0.00	0.00
0	0.19	0.49	0.00	0.00	0.18	0.66	0.00	0.01	0.17	1.36	0.00	0.00

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.14 shows the time it takes to complete 10 runs of simulations under each setup.

Table 7.14. Time Comparison of 10 runs (in minutes)

	$\rho = 0$				$\rho = 0.5$				$\rho = 0.9$			
#p $\downarrow$	BL	DL	AB	APB	BL	DL	AB	APB	BL	DL	AB	APB
200	22.92	6.20	0.18	0.18	25.02	5.60	0.29	0.27	38.20	3.37	0.62	0.82
500	44.71	23.92	0.33	0.33	43.37	22.16	0.61	0.56	43.74	15.00	0.46	0.36
1000	44.22	50.25	0.66	0.62	44.30	48.73	1.17	1.14	44.36	43.23	0.64	0.61

*Time comparison for for true $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

7.3 Compound Symmetry

Under compound symmetry (CS) for $\boldsymbol{\Sigma}$, we got the following tables. We note that we don't have to consider the case where $\rho = 0$ under this setting, since it has already been considered in the aforementioned tables. First, for $p = 200$, we have Tables 7.15 and 7.16 respectively for the coverage and width.

Table 7.15. Coverage Comparison for 200 Covariates under Compound Symmetry

	$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB
1	0.71	0.94	0.94	0.92	0.25	0.92	0.37	0.36
1	0.73	0.94	0.93	0.92	0.25	0.92	0.35	0.38
1	0.69	0.93	0.93	0.94	0.27	0.91	0.37	0.35
0	1.00	0.94	0.99	0.99	1.00	0.97	1.00	0.99
0	0.99	0.96	1.00	1.00	1.00	0.97	1.00	1.00
0	0.99	0.96	1.00	1.00	1.00	0.96	1.00	1.00
0	1.00	0.95	1.00	0.99	1.00	0.97	0.99	1.00
0	0.99	0.96	0.99	1.00	1.00	0.96	1.00	0.99
0	1.00	0.95	1.00	1.00	1.00	0.96	1.00	0.99
0	1.00	0.96	1.00	1.00	1.00	0.96	0.99	1.00

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.16. Average Interval Width Comparison for 200 Covariates under Compound Symmetry

	$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB
1	0.70	0.58	0.73	0.69	0.86	1.23	0.80	0.81
1	0.70	0.58	0.73	0.68	0.85	1.24	0.75	0.83
1	0.70	0.58	0.73	0.70	0.86	1.24	0.82	0.76
0	0.33	0.58	0.00	0.00	0.46	1.23	0.00	0.00
0	0.33	0.58	0.00	0.00	0.46	1.23	0.01	0.01
0	0.33	0.58	0.00	0.00	0.46	1.24	0.02	0.00
0	0.33	0.58	0.00	0.00	0.46	1.24	0.00	0.01
0	0.33	0.58	0.00	0.00	0.47	1.24	0.01	0.01
0	0.33	0.58	0.00	0.00	0.46	1.24	0.00	0.00
0	0.33	0.58	0.00	0.00	0.46	1.24	0.00	0.00

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Under compound symmetry, for $p = 500$, we have Tables 7.17 and 7.18 respectively for the coverage and width.

Table 7.17. Coverage Comparison for 500 Covariates under Compound Symmetry

	$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB
1	0.68	0.92	0.91	0.91	0.19	0.90	0.32	0.28
1	0.71	0.92	0.91	0.92	0.21	0.89	0.34	0.29
1	0.71	0.93	0.94	0.92	0.19	0.89	0.29	0.30
0	0.98	0.96	1.00	1.00	1.00	0.98	0.99	1.00
0	0.97	0.94	1.00	1.00	1.00	0.98	1.00	1.00
0	0.98	0.97	1.00	1.00	1.00	0.98	1.00	1.00
0	0.98	0.95	1.00	1.00	0.99	0.97	0.99	1.00
0	0.98	0.98	1.00	1.00	1.00	0.97	1.00	1.00
0	0.97	0.95	1.00	1.00	1.00	0.97	1.00	1.00
0	0.98	0.96	1.00	1.00	1.00	0.97	1.00	1.00

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.18. Average Interval Width Comparison for 500 Covariates under Compound Symmetry

	$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB
1	0.88	0.58	0.74	0.68	0.73	1.22	0.75	0.59
1	0.89	0.58	0.72	0.68	0.74	1.23	0.75	0.68
1	0.88	0.58	0.75	0.69	0.71	1.23	0.65	0.66
0	0.22	0.58	0.00	0.00	0.29	1.23	0.01	0.00
0	0.22	0.58	0.00	0.00	0.28	1.22	0.00	0.00
0	0.21	0.58	0.00	0.00	0.28	1.22	0.00	0.00
0	0.22	0.58	0.00	0.00	0.27	1.22	0.01	0.00
0	0.22	0.58	0.00	0.00	0.29	1.23	0.00	0.01
0	0.22	0.58	0.00	0.00	0.28	1.22	0.00	0.01
0	0.21	0.58	0.00	0.00	0.28	1.22	0.00	0.01

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Finally, considering $p = 1000$ we get the following tables (Tables 7.19 and 7.20) under compound symmetry.

Table 7.19. Coverage Comparison for 1000 Covariates under Compound Symmetry

	$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB
1	0.71	0.90	0.94	0.91	0.15	0.90	0.28	0.26
1	0.70	0.90	0.90	0.90	0.14	0.89	0.27	0.25
1	0.70	0.90	0.92	0.91	0.15	0.89	0.26	0.25
0	0.59	0.96	1.00	1.00	0.71	0.98	1.00	1.00
0	0.57	0.96	1.00	1.00	0.68	0.97	1.00	1.00
0	0.59	0.95	1.00	1.00	0.69	0.97	1.00	1.00
0	0.54	0.95	1.00	1.00	0.69	0.98	1.00	1.00
0	0.54	0.96	1.00	1.00	0.65	0.99	1.00	0.99
0	0.56	0.96	1.00	1.00	0.68	0.97	1.00	1.00
0	0.52	0.97	1.00	1.00	0.67	0.96	1.00	1.00

*Coverages for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.20. Average Interval Width Comparison for 1000 Covariates under Compound Symmetry

	$\rho = 0.5$				$\rho = 0.9$			
β_j	BL	DL	AB	APB	BL	DL	AB	APB
1	1.02	0.58	0.75	0.69	0.57	1.21	0.65	0.61
1	1.02	0.58	0.74	0.69	0.53	1.21	0.60	0.54
1	1.02	0.58	0.75	0.68	0.59	1.21	0.57	0.57
0	0.11	0.58	0.00	0.00	0.13	1.21	0.00	0.00
0	0.11	0.58	0.00	0.00	0.13	1.21	0.01	0.00
0	0.11	0.58	0.00	0.00	0.13	1.21	0.00	0.00
0	0.10	0.58	0.00	0.00	0.14	1.21	0.00	0.01
0	0.12	0.58	0.00	0.00	0.12	1.21	0.00	0.00
0	0.11	0.58	0.00	0.00	0.13	1.21	0.00	0.01
0	0.10	0.58	0.00	0.00	0.14	1.21	0.00	0.00

*Average widths for the first 10 regression coefficients are shown here.
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Table 7.14 shows the time it takes to complete 10 runs of simulations under each setup for compound symmetry.

Table 7.21. Time Comparison of 10 runs (in minutes)

	$\rho = 0.5$				$\rho = 0.9$			
#p ↓	BL	DL	AB	APB	BL	DL	AB	APB
200	38.34	6.79	0.26	0.30	39.65	4.01	0.56	0.70
500	46.20	29.09	0.45	0.47	44.10	22.32	1.05	0.87
1000	44.64	69.85	0.82	0.85	44.63	97.29	1.43	1.40

*Time comparison for for true $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$ under Compund Symmetry
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Chapter 8

Discussion

The simulation study of the three methods, Bayesian LASSO, de-sparsified LASSO, and the residual/perturbation bootstrap for the adaptive LASSO, under high dimensional datasets yielded some interesting results. The discussion is divided into a few segments.

8.1 Setting: True $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$ with AR-1

When the covariates are highly correlated with $\rho = 0.9$, the Bayesian LASSO-generated credible intervals are providing coverage almost close to 0.95 for $\beta_j = 1$ (last four columns of Tables 7.3 and 7.1). This is a somewhat desirable result since it is the *nominal coverage*. But they show bigger width according to the first three rows of the last four columns of Tables 7.4 and 7.2. In these cases, adaptive LASSO generated intervals seem to be doing worse (but with smaller intervals) while de-sparsified LASSO coverage is around 0.8 with smaller average width than Bayesian LASSO. In the cases of $\beta_j = 0$, Bayesian LASSO is providing coverages close to 1 (over-coverage) with very narrow intervals in $p = 200 and 500$ cases. For $p = 1000$ cases, however, the coverage is poor with approximately 0.50-0.60. The de-sparsified LASSO-generated intervals are doing moderately fine having captured the true coefficients around 0.90-0.95 fractions of the datasets.

The adaptive LASSO generated bootstrap intervals whether they are generated by residual bootstrap or perturbation bootstrap, exhibit similar kind of results throughout this section.

For adaptive LASSO, the coverage for the intervals for zero-coefficients are (almost) always 1 due to the fact that the method throws out those coefficients in each of the iterations and hence we did not make any intervals to begin with. This is a prevalent pattern in Tables 7.3 and 7.1 for any correlation structure. Sometimes, we see 0.99 coverage but that could be attributed to misidentification of those covariates as non-zeros. For the same reason, the average width for $\beta_j = 0$ is almost always 0, as we can see in rows 4-10 of columns 3,4,7,8,11 and 12 of Tables 7.2, 7.4, and 7.6.

When the covariates are moderately correlated with $\rho = 0.5$, for non-zero coefficients (in the middle four columns of Tables 7.5, 7.3, and 7.1), the coverage is — almost of the same nature for three methods — revolving around 0.9. The de-sparsified LASSO has the smallest average interval width. In zero coefficient cases, the interval widths for de-sparsified LASSO remain the same but for Bayesian, they get very narrow while getting better coverage (around 0.97) than de-sparsified LASSO in $p = 200 and 500$ cases. For $p = 1000$ cases, again, the coverage is poor with approximately 0.50-0.60. As usual, adaptive LASSO has coverage 1 having thrown out those coefficients in the beginning.

In the cases where the predictors were uncorrelated i.e $\rho = 0$, for non-zero coefficients, the coverage is around 0.77 which is lower (with wider intervals too) than both the other methods, according to the first three rows of the first four columns of Tables 7.5, 7.3, and 7.1. De-sparsified LASSO is having coverage close to nominal 0.95 while adaptive LASSO coverages are around 0.9. Both these methods have almost the same average width. For zero coefficients under $\rho = 0$, the Bayesian is producing higher coverage (close to 0.98) with narrower intervals in $p = 200 and 500$ cases. For $p = 1000$ cases, again, the coverage is poor with approximately 0.50-0.60. The de-sparsified LASSO is hitting the nominal coverage with average width the same as the ones from $\beta_j = 0$.

In both the uncorrelated and the moderately correlated cases as in $\rho = 0, 0.5$ (from

Tables 7.2, 7.4 and 7.6, columns 2 and 6), we could also see that the average width for de-sparsified LASSO revolves around 0.45 to 0.50, regardless of the coefficients' true values.

In summary, the de-sparsified LASSO is producing coverage close to 0.95 overall regardless of the correlation. The same kind of coverage is achieved by Bayesian LASSO under moderate and high correlation although the width is larger than de-sparsified LASSO. In a particular setup, the average width for zero and non-zero coefficients under de-sparsified LASSO remain almost the same. Regardless of the correlation, the Bayesian LASSO is getting narrower intervals for the zero-coefficients. For $p = 1000$ cases though, the coverage for the zero true coeffcients under Bayesian setup is poor (0.50-0.60). Adaptive LASSO provides close to nominal coverage in moderately correlated cases for non-zero coefficients with wider intervals than de-sparsified LASSO.

As for time comparison, after running 10 iterations or each setup, adaptive LASSO with perturbation bootstrap achieves the fastest time - slightly faster than adaptive LASSO with residual bootstrap. Compared to these two, Bayesian LASSO and de-sparsified LASSO are much slower. While most of the time, de-sparsified LASSO takes less time, under no correlation and moderate correlation cases for $p = 1000$, de-sparsified LASSO is actually slower, according to Table 7.7.

8.2 Setting: True $\boldsymbol{\beta}^T = [5, -4, 3, -2, 1, 0, \ldots, 0]$ with AR-1

First, we look at the high correlation i.e. $\rho = 0.9$ settings for the predictors. For true zero coefficients, the Bayesian LASSO-generated intervals are getting coverage close to 1 for $p = 200, 500$ settings in Tables 7.8 and 7.10. But when $p = 1000$, the intervals are getting coverage around 0.59-0.65 in the last 5 rows of column 9 of Table 7.12. Regardless of the size of p, the width of these intervals are smaller than those from the de-sparsified LASSO (Tables 7.9, 7.11, 7.13). The de-sparsified

LASSO-generated intervals for these cases are getting close to nominal coverage 0.95. Also, regardless of the size of p, the adaptive LASSO-generated intervals have width almost close to zero if not zero. That means they did not generate an interval to begin with similar to the previous AR-1 setups. The result is validated by the fact that coverage for the zero coefficients is almost close to one if not one. This result for adaptive LASSO is even more pronounced for $p = 1000$ cases (Table 7.13).

For the non-zero coefficients under high correlation setting, in bigger p i.e $p = 500, 1000$ setting, the coverage for the adaptive LASSO-generated intervals is very close to zero or exactly zero (Tables 7.10, 7.12). But in $p = 200$ cases, the coverage seems to be more if the size of the true coefficient value is bigger and vice-versa (Columns 11 and 12 in Table 7.8). For the de-sparsified LASSO-generated intervals, the coverage for the true coefficients 1 seem to increase from 0.49 to 0.91 with the increase in size for p. For $\beta = 5$, the coverage lurks between 0.29-039. For the rest of the true non-zero coefficients $(-4, 3, -2)$ the coverage seemed to be small and very close to zero under each p. The average width of the covariates for each p for this method is almost the same regardless of the true coefficient value, except for true coefficient value 5, but whose width turned out to be slightly different. The coverage gets poorer and gets to almost zero for the non-zero coefficents under the Bayesian LASSO-generated intervals with the increase in the size of p. For $p = 200$, the coverage was small but by the time we got to $p = 1000$, the coverage for four of the five non-zero true coefficients got to zero.In all three setting of p, the average width gets smaller with the size of the absolute true coefficient value getting smaller and for true zero coefficients, the average width appear to be almost the same under each p (Tables 7.13, 7.11, 7.9).

For $\rho = 0.5$ setting, the coverage for the adaptive LASSO-generated intervals for the true zero coefficients were close to 1 with average width close to 0, according to columns 7 and 8 in Tables 7.13, 7.11, 7.9. For the non-zero true coefficients,

the coverage seemed to get smaller as the size of p increased. In case for $p = 200$, the coverage is moderately good at 0.80-0.89 but for $p = 1000$, it gets poor and ranges among 0.42-.56. The perturbation bootstrap generated smaller intervals with smaller coverage than those of residual bootstrap regardless of the size of p. For the true zero coefficients under de-sparsified LASSO, the intervals were very close to zero and sometimes zero and for the zero coefficients, the intervals give almost close to nominal coverage. For the non-zero true coefficients, the pattern follows the same as the highly correlated cases. The coverage gets poorer with the increase of the size of p. The average width remained almost the same regardless of the the true coefficient under each p. Under Bayesian LASSO, the coverage for the non-zero true coefficient is actually decent, above 0.80. It got a poorer with the increase of the size of p although the coverage for true coefficient values $5, 1$ remain decent for $p = 1000$. Again, the magnitude of the true β affected the width of the intervals with the bigger absolute value of β got bigger intervals.

Under the uncorrelated cases i.e. $\rho = 0$, (column 1 in Tables 7.12, 7.10, 7.8) the Bayesian LASSO-generated intervals for true non-zero coefficients produce good coverage, close to the nominal 0.95 mark although for $p = 1000$ cases. The average width increase with the size of p for these true coefficients. True zero coefficients enjoy coverage close to 1 for $p = 200, 500$ cases but decline dramatically for $p = 1000$ cases to around 0.60. Their width seem to be decreasing with the increase of p. For Bayesian LASSO, the average width is the biggest among the four methods for true non-zero coefficients but appear to be smaller than de-sparsified LASSO-generated intervals for true zero coefficient values. Under $\rho = 0$, for the adaptive LASSO methods, the residual bootstrap generated intervals for the true non-zero coefficients provide coverage closer to 0.95 for $p = 200$, closer to 0.9 for $p = 500$, and closer to 0.85 for $p = 1000$; the perturbation bootstrap-generated coverage is very slightly smaller than that by residual bootstrap. For true zero coefficients, the coverage is as

usual close to one since this method do not select zero coefficients in its estimation and selection. For the de-sparsified LASSO, and for the true zero coefficients, the intervals range from 0.95-0.98 which is more than our nominal coverage in most cases. For true non-zero coefficients, the intervals tend to give less than nominal coverage but it is still greater than 0.90 in most cases. The average width remained almost the same regardless of the the true coefficient under each p. This is a result similar to the one from $rho = 0.5$ for de-sparsified LASSO.

According to Table 7.14, the time comparison results are similar to the previous AR-1 setup. The Bayesian LASSO is the slowest succeeded by de-sparsified LASSO, adaptive LASSO with residual bootstrap, and perturbation bootstrap with the exception as follows. Under $p = 1000$ cases for no correlation and moderate correlation, Bayesian LASSO is faster than the de-sparsified LASSO.

8.3 Setting: True $\beta^T = [1, 1, 1, 0, \ldots, 0]$ with Compound Symmetry

For $\rho = 0.9$, the adaptive LASSO generated intervals provided coverage close to zero for the true zero coefficients for the reasons stated in the previous sections. The coverage seemed to get worse with the increase in the size of p. The coverage in these cases ranged from 0.38 to 0.25 (columns 7 and 8 in Tables 7.15, 7.17, and 7.19). The true zero coefficients got coverage close to one. There seemed to be no distinguishable differences for the coverage provided by residual and perturbation bootstrap methods;however, the width were smaller for the intervals generated by the perturbation bootstrap in $p = 500, 100$ settings (Tables 7.18, and 7.20).

For $\rho = 0.9$, the de-sparsified LASSO-generated intervals for the true zero coefficients are getting coverage greater than the nominal coverage consistently. For the true non-zero coefficients, the coverage was found to be greater than 0.9 for $p = 200$ (in column 6 in Table 7.15), but ranged from 0.89-0.9 for $p = 500, 1000$ (in column 6 in Tables 7.17 and 7.19). Regardless of the true coefficient size, these intervals are

the widest among the four kinds of intervals.

For For $\rho = 0.9$, under Bayesian LASSO, the non-zero true coefficients provdied very poor coverage, around 0.26 for $p = 200$ and seemed to get worse as the size of p increased. For the true zero coefficients, this method yielded coverage ranging from 0.67 to 0.71 for $p = 1000$ setting (Table 7.19). Interestingly, for $p = 200, 500$, the coverage seemed to be close to one for this method (column 5 in (Tables 7.17, 7.15). The width were smaller for the true zero coefficients.

For $\rho = 0.5$, the adaptive LASSO generated intervals provided coverage around 0.90-0.94 (in columns 3 and 4 in Tables 7.15, 7.17, and 7.19) . We could not see clear distinction among the coverage by residual and perturbation bootstrap methods; however, the perturbation bootstrap generated intervals were narrower (in columns 3 and 4 in Tables 7.16, 7.18, and 7.20).

For this moderate correlation, the de-sparsified LASSO-generated intervals for the true zero coefficients were close to the nominal coverage while the coverage for the true non-zero coefficients were between 0.90-0.94. The coverage seemed to be around 0.90 under $p = 1000$ (in columns 2 in Table 7.5). The average width remains the same regardless of the size of the true coefficients across different p for this method. These width were, however, greater than those from Bayesian LASSO-generated intervals.

For Bayesian LASSO, the coverage for the true non-zero coefficients were between 0.69-0.71 (column 1 in Tables 7.15, 7.17, and 7.19) with wider intervals than the true zero coefficient values. Interestingly, for the true zero coefficients, the intervals provided coverage close to one for $p = 200$, around 0.97-0.98 for $p = 500$, but around 0.52-0.59 for $p = 1000$.

According to Table 7.21, the time comparison results are slightly different. The Bayesian LASSO is the slowest succeeded by de-sparsified LASSO with the exception for $p = 1000$ cases, where Bayesian LASSO is significantly faster. In this compund symmetry setup, it is found that adaptive LASSO with perturbation bootstrap is

typically slower than the adaptive LASSO with residual bootstrap.

8.4 Recommendation

For AR-1 cases, when $p >> n$, we should consider Bayesian LASSO, as it provided better coverage for true non-zero coefficient, specially if we could somehow determine that the correlation is close to 0.9 and 0.5. But for uncorrelated cases, we should use de-sparsified LASSO when the true $\boldsymbol{\beta}^T = [1, 1, 1, 0, \ldots, 0]$.

For compound symmetry cases, the de-sparsified LASSO seemed to provide coverage closer to the nominal coverage regardless of the value of $\rho = 0$. However, the coverage was around 0.90 for $p = 1000$ that is truly high-dimensional case. This better coverage comes with the cost of getting wider intervals for highly correlated cases. For the moderate correlation, the intervals by de-sparsified LASSO is even smaller for the true non-zero coefficients. So, in compound symmetry cases, de-sparsified LASSO would be my recommendation.

But we need to be mindful of the fact that typically Bayesian LASSO and de-sparsified LASSO take significantly longer to compute.

The bootstrap generated intervals tend to provide decent coverage, around 0.90, in the uncorrelated cases, regardless of the number of covariates in the model. But they appear to suffer when the predictors are highly correlated. For $p = 200$ though, they did well under moderate correlation with the perturbation bootstrap providing narrower intervals leading us to the conclusion that with low correlation and number of predictors not too much greater than the sample size, perturbation bootstrap could provide close to nominal coverage.

If we take into account that the adaptive LASSO with perturbation bootstrap typically achieves faster time in most cases, then the trade-off for time with accuracy makes it an appealing choice.

Chapter 9

Application on Real Data

The aforementioned three methods were applied on two real datasets. The first one is the **riboflavin** dataset and the second one is the **Eyedata**.

While analyzing the datasets, for the Bayesian LASSO method, we will proceed on selecting predictor if the posterior median of the predictor is non-zero. In this way, for large values of τ i.e. putting strong penalization towards sparsity, the posterior median tends to be exactly zero.

9.1 Riboflavin Data

This dataset is publicly available in the **hdi** R package. This is a dataset of riboflavin production by Bacillus subtilis containing n = 71 observations of p = 4088 predictors (gene expressions) and a one-dimensional response (riboflavin production). The responses as well as the predictors are numeric.

Since the de-sparsified LASSO is not a sparse estimator, we only get variable selection through the Bayesian LASSO and the adaptive LASSO methods. Also, it appears that the Bayesian LASSO put strong penalization on the covariates to induce sparsity. Hence a large number of covariates gets their posterior means as exactly zero and do not get selected through this method.

In total, 17 variables were selected by these two methods where Bayesian LASSO methods selected only 5 of them whereas adaptive LASSO chose 11 predictors. We used residual and perturbation bootstrap on two occasions to build confidence intervals for the estimates of the selected variables by adaptive LASSO. Interestingly,

while performing the calculation for estimating and subsequently building CI using residual bootstrap, the 11 chosen variables would not quite match with the ones that later got generated by using perturbation bootstrap. Two variables were mismatched for these two methods of estimation. Table 9.2 shows the CIs of these 17 variables.

But before that, we will look at the empirical correlation structure of the selected covariates by Bayesian LASSO.

Table 9.1. Observed Correlation of the Selected Predictors from Eyedata by Bayesian LASSO Estimator

Index $\downarrow \rightarrow$	825	1285	3024	3026	3995
825	1.00	-0.22	0.56	0.51	-0.27
1285	-0.22	1.00	0.05	-0.27	-0.25
3024	0.56	0.05	1.00	0.27	-0.42
3026	0.51	-0.27	0.27	1.00	0.30
3995	-0.27	-0.25	-0.42	0.30	1.00

Table 9.1 suggests that the predictors selected by Bayesian LASSO have mild to moderate correlations.

Table 9.2. CIs of the Selected Predictors from Riboflavin Dataset by Bayesian and Adaptive LASSO methods

Methods →	BL		DL		AB		APB	
Variable # ↓	L	U	L	U	L	U	L	U
825	-0.71	0.04	-0.36	0.28	0.00	0.00	0.00	0.00
1285	-0.00	0.49	-0.18	0.32	0.00	0.00	0.00	0.00
3024	-0.72	0.10	-0.69	0.12	0.00	0.00	0.00	0.00
3026	-0.72	0.11	-0.43	0.29	0.00	0.00	0.00	0.00
3995	-0.63	0.17	-0.53	0.40	0.00	0.00	0.00	0.00
69	0.00	0.00	-0.37	0.05	**-0.50**	**-0.06**	-0.44	-0.10
87	0.00	0.00	-0.41	0.22	**-0.46**	**-0.03**	-0.50	-0.04
974	0.00	0.00	-0.27	0.11	-0.34	0.10	-0.29	0.09
1278	0.00	0.00	-0.21	0.21	-0.32	0.08	0.00	0.00
1303	0.00	0.00	-0.10	0.28	**0.09**	**0.46**	0.12	0.48
1436	0.00	0.00	-0.30	0.20	-0.27	0.26	-0.18	0.29
1478	0.00	0.00	-0.19	0.05	**-0.37**	**-0.06**	-0.36	-0.09
1502	0.00	0.00	-0.23	0.13	-0.35	0.04	-0.38	0.06
3288	0.00	0.00	-0.08	0.29	**0.06**	**0.50**	0.08	0.53
3310	0.00	0.00	-0.15	0.20	**-0.07**	**-0.06**	-0.02	-0.01
4002	0.00	0.00	-0.17	0.19	**-0.45**	**-0.02**	-0.44	-0.09
69	0.00	0.00	-0.37	0.05	-0.50	-0.06	**-0.44**	**-0.10**
87	0.00	0.00	-0.41	0.22	-0.46	-0.03	**-0.50**	**-0.04**
974	0.00	0.00	-0.27	0.11	-0.34	0.10	-0.29	0.09
1303	0.00	0.00	-0.10	0.28	0.09	0.46	**0.12**	**0.48**
1436	0.00	0.00	-0.30	0.20	-0.27	0.26	-0.18	0.29
1478	0.00	0.00	-0.19	0.05	-0.37	-0.06	**-0.36**	**-0.09**
1502	0.00	0.00	-0.23	0.13	-0.35	0.04	-0.38	0.06
3288	0.00	0.00	-0.08	0.29	0.06	0.50	**0.08**	**0.53**
3310	0.00	0.00	-0.15	0.20	-0.07	-0.06	**-0.02**	**-0.01**
4002	0.00	0.00	-0.17	0.19	-0.45	-0.02	**-0.44**	**-0.09**
4003	0.00	0.00	**-0.52**	**-0.17**	0.00	0.00	-0.03	0.18

*Showing only 17 credible/confidence intervals selected by Bayesian and adaptive LASSO methods
*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

We could see from Table 9.2 that all of the confidence intervals generated via de-sparsified LASSO estimates contained zero except for covariate #4003. The de-sparsified LASSO method did generate coefficients for all the covariates and 47 of them had p-value less than 0.05. But only one (#4003) of these 47 predictors matched

the ones selected by the Bayesian and adaptive LASSO methods.

Also, we would like to look at the average width of these intervals through Table 9.3.

Table 9.3. Average Width of the CIs of the Selected Predictors from Riboflavin Dataset by Bayesian and Adaptive LASSO methods

Variable # ↓	BL	DL	AB	APB
825	0.75	0.64	0.00	0.00
1285	0.49	0.50	0.00	0.00
3024	0.82	0.81	0.00	0.00
3026	0.83	0.73	0.00	0.00
3995	0.80	0.93	0.00	0.00
69	0.00	0.43	0.44	0.34
87	0.00	0.63	0.43	0.46
974	0.00	0.37	0.44	0.38
1278	0.00	0.42	0.40	0.00
1303	0.00	0.38	0.38	0.36
1436	0.00	0.50	0.53	0.47
1478	0.00	0.24	0.31	0.27
1502	0.00	0.37	0.39	0.44
3288	0.00	0.37	0.44	0.44
3310	0.00	0.36	0.02	0.00
4002	0.00	0.35	0.43	0.34
69	0.00	0.43	0.44	0.34
87	0.00	0.63	0.43	0.46
974	0.00	0.37	0.44	0.38
1303	0.00	0.38	0.38	0.36
1436	0.00	0.50	0.53	0.47
1478	0.00	0.24	0.31	0.27
1502	0.00	0.37	0.39	0.44
3288	0.00	0.37	0.44	0.44
3310	0.00	0.36	0.02	0.00
4002	0.00	0.35	0.43	0.34
4003	0.00	0.35	0.00	0.20

*Showing the width of the 17 credible/confidence intervals selected by Bayesian and adaptive LASSO methods respectively
**BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO with Residual Bootstrap, APB = Adaptive LASSO with Perturbation Bootstrap

Another important thing to notice is the runtimes of three methods.

The reason as to why de-sparsified LASSO took this long might be due to the fact

Table 9.4. Runtime Comparison for the Methods (in minutes)

Bayesian	De-Sparsified	ALASSO-RB	ALASSO-PB
2.08	51.63	4.54	9.36

that this method had to conduct node-wise regression where it ran cross-validation to tune p number of penalty parameters in order to calculate $\hat{\mathbf{\Theta}}$ as well as it had to tune the penalty parameter to calculate the $\hat{\boldsymbol{\beta}}^L$. And it had to do this for each covariate.

9.2 Eyedata

This dataset is available in **picasso** R package (Ge, Li, Jiang, Wang, Zhang, Liu, and Zhao (2019)). This is a dataset of gene expression data from the microarray experiments of mammalian eye tissue samples. There are 200 predictors of 120 observations containing the data of 120 rats with 200 gene probes and the response represents the expression level of TRIM32 gene of 120 rats. Therefore, it is a $n = 120, p = 200$ setting. The responses and predictors are both numeric.

The analysis shows that Bayesian LASSO selected 11 predictors while adaptive LASSO while using residual bootstrap selected 14, but while using perturbation bootstrap, it selected 16. There is one match from the Bayesian LASSO and adaptive LASSO (with residual bootstrap) selected covariates, that is covariate #62. For adaptive LASSO (with perturbation bootstrap) and Bayesian LASSO, we also have the same match – Covariate #62. Except for covariate #38, all the covariates selected by adaptive LASSO while using residual bootstrap would match the covariates selected by adaptive LASSO while using perturbation bootstrap. Only covariate #62 is selected in the three processes. So in total, 26 covariates are selected.

Since de-sparsified LASSO is not a sparse estimator, we did not get any variable selection from it. But through this estimator, we obtained eight predictors which were statistically significant with p-value $< 0{:}05$. From those eight, 6 predictors have been matched with the ones selected by Bayesian LASSO and adaptive LASSO. They

are covariates #50, #62, #76, #87, #174, and #185.

First we look at the empirical correlation structure of the covariates. We will observe the covariates selected by the Bayesian LASSO in this case.

Table 9.5. Observed Correlation of the Selected Predictors from Eyedata by Bayesian LASSO Estimator

Index ↓ →	50	62	76	87	140	153	155	180	185	187	200
50	1.00	0.54	-0.73	0.66	-0.43	-0.58	-0.52	-0.53	0.54	0.63	0.55
62	0.54	1.00	-0.57	0.50	-0.53	-0.60	-0.53	-0.58	0.36	0.38	0.33
76	-0.73	-0.57	1.00	-0.60	0.63	0.65	0.70	0.73	-0.47	-0.46	-0.50
87	0.66	0.50	-0.60	1.00	-0.50	-0.63	-0.42	-0.50	0.55	0.53	0.51
140	-0.43	-0.53	0.63	-0.50	1.00	0.72	0.57	0.57	-0.31	-0.27	-0.41
153	-0.58	-0.60	0.65	-0.63	0.72	1.00	0.70	0.64	-0.40	-0.42	-0.45
155	-0.52	-0.53	0.70	-0.42	0.57	0.70	1.00	0.72	-0.36	-0.44	-0.39
180	-0.53	-0.58	0.73	-0.50	0.57	0.64	0.72	1.00	-0.29	-0.40	-0.42
185	0.54	0.36	-0.47	0.55	-0.31	-0.40	-0.36	-0.29	1.00	0.60	0.52
187	0.63	0.38	-0.46	0.53	-0.27	-0.42	-0.44	-0.40	0.60	1.00	0.56
200	0.55	0.33	-0.50	0.51	-0.41	-0.45	-0.39	-0.42	0.52	0.56	1.00

It appears that from Table 9.5 the empirical correlation among these variables are mostly of moderate nature.

Table 9.6. CIs of the Selected Predictors from Eyedata by Bayesian LASSO Estimator

Methods →	BL		DL		AB		APB	
Variable # ↓	L	U	L	U	L	U	L	U
50	-0.01	0.09	**0.04**	**0.21**	0.00	0.00	0.00	0.00
62	-0.09	0.01	**-0.14**	**-0.01**	**-0.25**	**-0.05**	**-0.31**	**-0.08**
76	-0.10	0.01	**-0.17**	**-0.02**	0.00	0.00	0.00	0.00
87	-0.15	0.01	**-0.23**	**-0.03**	0.00	0.00	0.00	0.00
140	-0.02	0.19	-0.01	0.28	0.00	0.00	0.00	0.00
153	-0.00	0.21	**0.10**	**0.35**	0.00	0.00	0.00	0.00
155	-0.01	0.09	-0.01	0.15	0.00	0.00	0.00	0.00
180	-0.01	0.14	**0.01**	**0.22**	0.00	0.00	0.00	0.00
185	-0.16	0.01	**-0.28**	**-0.00**	0.00	0.00	0.00	0.00
187	-0.13	0.03	-0.23	0.02	0.00	0.00	0.00	0.00
200	-0.13	0.02	-0.20	0.01	0.00	0.00	0.00	0.00

*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO w/ Residual Bootstrap, APB = Adaptive LASSO w/ Perturbation Bootstrap

Similar to the *Riboflavin* dataset, the Bayesian LASSO put strong penalization on the covariates to induce sparsity. Hence a large number of covariates gets their posterior means as exactly zero and do not get selected through this method.

Table 9.7. CIs of the Selected Predictors from Eyedata by Adaptive LASSO Estimator with Residual Bootstrap Process

Methods →	BL		DL		AB		APB	
Variable # ↓	L	U	L	U	L	U	L	U
2	-0.05	0.02	-0.12	0.05	-0.14	0.29	-0.15	0.35
3	-0.02	0.04	-0.04	0.10	**0.21**	**0.50**	**0.35**	**0.57**
4	-0.03	0.04	-0.05	0.11	-0.00	0.16	**0.05**	**0.24**
5	-0.05	0.03	-0.11	0.07	**0.39**	**0.64**	**0.37**	**0.60**
7	-0.04	0.02	-0.11	0.04	**0.18**	**0.45**	**0.15**	**0.39**
9	-0.03	0.04	-0.09	0.08	**0.03**	**0.04**	**0.04**	**0.05**
10	-0.03	0.03	-0.10	0.05	**0.10**	**0.37**	**0.10**	**0.27**
17	-0.02	0.03	-0.08	0.06	**-0.35**	**-0.05**	**-0.37**	**-0.12**
58	-0.02	0.04	-0.04	0.09	**0.07**	**0.47**	**0.12**	**0.41**
62 65	-0.02	0.04	-0.06	0.08	**0.02**	**0.20**	**0.06**	**0.26**
83	-0.03	0.04	-0.05	0.12	-0.35	0.04	**-0.42**	**-0.04**
85	-0.02	0.05	-0.09	0.09	-0.11	0.15	-0.10	0.13
109	-0.06	0.02	-0.11	0.10	-0.12	0.03	**-0.12**	**-0.01**

*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO w/ Residual Bootstrap, APB = Adaptive LASSO w/ Perturbation Bootstrap

Table 9.8. CIs of the Selected Predictors from Eyedata by Adaptive LASSO Estimator with Perturbation Bootstrap Process

Methods →	BL		DL		AB		APB	
Variable # ↓	L	U	L	U	L	U	L	U
2	-0.05	0.02	-0.12	0.05	-0.14	0.29	-0.15	0.35
3	-0.02	0.04	-0.04	0.10	**0.21**	**0.50**	**0.35**	**0.57**
4	-0.03	0.04	-0.05	0.11	-0.00	0.16	**0.05**	**0.24**
5	-0.05	0.03	-0.11	0.07	**0.39**	**0.64**	**0.37**	**0.60**
7	-0.04	0.02	-0.11	0.04	**0.18**	**0.45**	**0.15**	**0.39**
9	-0.03	0.04	-0.09	0.08	**0.03**	**0.04**	**0.04**	**0.05**
10	-0.03	0.03	-0.10	0.05	**0.10**	**0.37**	**0.10**	**0.27**
17	-0.02	0.03	-0.08	0.06	**-0.35**	**-0.05**	**-0.37**	**-0.12**
38	-0.03	0.04	-0.08	0.09	0.00	0.00	**0.04**	**0.04**
58	-0.02	0.04	-0.04	0.09	**0.07**	**0.47**	**0.12**	**0.41**
62	-0.09	0.01	**-0.14**	**-0.01**	**-0.25**	**-0.05**	**-0.31**	**-0.08**
65	-0.02	0.04	-0.06	0.08	**0.02**	**0.20**	**0.06**	**0.26**
83	-0.03	0.04	-0.05	0.12	-0.35	0.04	**-0.42**	**-0.04**
85	-0.02	0.05	-0.09	0.09	-0.11	0.15	-0.10	0.13
109	-0.06	0.02	-0.11	0.10	-0.12	0.03	**-0.12**	**-0.01**
169	-0.03	0.05	-0.06	0.10	0.00	0.00	**-0.00**	**-0.00**

*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO w/ Residual Bootstrap, APB = Adaptive LASSO w/ Perturbation Bootstrap

Tables 9.6, 9.7, and 9.8 show that covariate #62 is selected by both estimators in all three occasions. Interestingly, Bayesian LASSO provided with the intervals for all the predictors regardless of whether the predictor was selected by this method or not. But all of these intervals seem to contain 0 inside them.

In Table 9.6, 7 of the predictors selected by Bayesian LASSO are significant for the de-sparsified LASSO according to the confidence intervals while for adaptive LASSO, only one is significant for both the residual bootstrap-generated intervals and the perturbation bootstrap-generated intervals.

In Table 9.7, one variable is significant by de-sparsified LASSO generated intervals among the predictors selected by adaptive LASSO (with residual bootstrap method). Among those selected variables, 9 variables are found significant according to the residual bootstrap-generated intervals and 12, according to the perturbation

bootstrap intervals.

Table 9.7 shows the intervals for the predictors selected by adaptive LASSO estimators while calculating perturbation bootstrap intervals. According to those perturbation bootstrap-generated intervals, 14 of the selected predictors are significant while de-sparsified LASSO-generated intervals indicate only one and adaptive LASSO (in calculating residual bootstrap intervals) indicate 9 significant predictors.

Table 9.9. Average Width of the CIs of the Selected Predictors from Eyedata by Bayesian and Adaptive LASSO methods

Variable # ↓	BL	DL	AB	APB
50	0.10	0.17	0.00	0.00
62	0.10	0.13	0.20	0.23
76	0.11	0.15	0.00	0.00
87	0.16	0.19	0.00	0.00
140	0.20	0.29	0.00	0.00
153	0.22	0.25	0.00	0.00
155	0.10	0.16	0.00	0.00
180	0.15	0.21	0.00	0.00
185	0.17	0.28	0.00	0.00
187	0.16	0.25	0.00	0.00
200	0.14	0.21	0.00	0.00
2	0.07	0.16	0.43	0.50
3	0.07	0.14	0.29	0.22
4	0.06	0.16	0.16	0.19
5	0.07	0.18	0.24	0.22
7	0.06	0.16	0.27	0.24
9	0.07	0.17	0.00	0.02
10	0.06	0.15	0.27	0.16
17	0.06	0.14	0.31	0.25
58	0.06	0.14	0.40	0.29
62	0.10	0.13	0.20	0.23
65	0.06	0.14	0.19	0.19
83	0.07	0.17	0.39	0.38
85	0.07	0.18	0.26	0.23
109	0.09	0.21	0.15	0.10
2	0.07	0.16	0.43	0.50
3	0.07	0.14	0.29	0.22
4	0.06	0.16	0.16	0.19
5	0.07	0.18	0.24	0.22
7	0.06	0.16	0.27	0.24
9	0.07	0.17	0.00	0.02
10	0.06	0.15	0.27	0.16
17	0.06	0.14	0.31	0.25
38	0.07	0.17	0.00	0.01
58	0.06	0.14	0.40	0.29
62	0.10	0.13	0.20	0.23
65	0.06	0.14	0.19	0.19
83	0.07	0.17	0.39	0.38
85	0.07	0.18	0.26	0.23
109	0.09	0.21	0.15	0.10
169	0.08	0.16	0.00	0.00

*BL = Bayesian LASSO, DL = De-sparsified LASSO, AB = Adaptive LASSO w/ Residual Bootstrap, APB = Adaptive LASSO w/ Perturbation Bootstrap

Table 9.9 shows the average width of the intervals of the predictors. It seems that the Bayesian LASSO generated intervals are smaller than the other three types. The perturbation bootstrap generated intervals are narrower than the ones generated by the residual bootstrap generated ones.

Table 9.10 shows the runtime to estimate and calculate the intervals under these methods.

Table 9.10. Runtime Comparison for the Methods (in minutes)

Bayesian	De-Sparsified	ALASSO-RB	ALASSO-PB
6.06	0.74	1.12	0.94

In contrast to the Riboflavin dataset analysis results (Table 9.4), Bayesian LASSO took the longest to complete the calculation for this dataset.

Bibliography

Casella, George, and Edward I. George. 1992. "Explaining the Gibbs Sampler." *The American Statistician* 46 (3): 167–174. ISSN: 00031305. http://www.jstor.org/stable/2685208.

Das, Debraj, Karl Gregory, and S. N. Lahiri. 2019. "Perturbation bootstrap in adaptive Lasso." *The Annals of Statistics* 47 (4): 2080–2116. https://doi.org/10.1214/18-AOS1741. https://doi.org/10.1214/18-AOS1741.

Dezeure, Ruben, Peter Bühlmann, Lukas Meier, and Nicolai Meinshausen. 2015. "High-Dimensional Inference: Confidence Intervals, p-values and R-Software hdi." *Statistical Science* 30 (4): 533–558.

Efron, Bradley, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. 2004. "Least angle regression." *The Annals of Statistics* 32, no. 2 (April). ISSN: 0090-5364. https://doi.org/10.1214/009053604000000067. http://dx.doi.org/10.1214/009053604000000067.

Friedman, Jerome H., Trevor Hastie, and Rob Tibshirani. 2010. "Regularization Paths for Generalized Linear Models via Coordinate Descent." *Journal of Statistical Software* 33 (1): 1–22. https://doi.org/10.18637/jss.v033.i01. https://www.jstatsoft.org/index.php/jss/article/view/v033i01.

Fu, Wenjiang, and Keith Knight. 2000. "Asymptotics for lasso-type estimators." *The Annals of Statistics* 28 (5): 1356–1378. https://doi.org/10.1214/aos/1015957397. https://doi.org/10.1214/aos/1015957397.

Ge, Jason, Xingguo Li, Haoming Jiang, Mengdi Wang, Tong Zhang, Han Liu, and Tuo Zhao. 2019. *picasso: Pathwise Calibrated Sparse Shooting Algorithm.* R package version 1.3.1. https://CRAN.R-project.org/package=picasso.

Geer, Sara van de, Peter Bühlmann, Ya'acov Ritov, and Ruben Dezeure. 2014. "On asymptotically optimal confidence regions and tests for high-dimensional models." *The Annals of Statistics* 42, no. 3 (June). ISSN: 0090-5364. https://doi.org/10.1214/14-aos1221. http://dx.doi.org/10.1214/14-AOS1221.

Gill, P.E., W. Murray, and M.H. Wright. 1981. "Practical Optimization." Chap. 5. Academic Press.

Gramacy, Robert B., and with Fortran contributions from Cleve Moler as updated by Berwin A. Turlach (qpgen2/quadprog). 2019. *monomvn: Estimation for MVN and Student-t Data with Monotone Missingness.* R package version 1.9-13. https://CRAN.R-project.org/package=monomvn.

Gregory, Karl. 2020. *hidbootreg: Bootstrap methods for sparsity-penalized linear regression estimators.* R package version 1.0.

Hans, Chris. 2009. "Bayesian lasso regression." *Biometrika* 96 (4): 835–845. https://EconPapers.repec.org/RePEc:oup:biomet:v:96:y:2009:i:4:p:835-845.

Hastie, T., R. Tibshirani, and J. Friedman. 2009. "The elements of statistical learning: data mining, inference, and prediction.," 2nd, 68–73. New York: Springer.

Park, Trevor, and George Casella. 2008. "The Bayesian Lasso." *Journal of the American Statistical Association* 103 (482): 681–686. https://doi.org/10.1198/016214508000000337. eprint: https://doi.org/10.1198/016214508000000337. https://doi.org/10.1198/016214508000000337.

Sicotte, Xavier Bourret. 2018. "Lasso regression: derivation of the coordinate descent update rule." Accessed: 2021-11-10. https://xavierbourretsicotte.github.io/lasso_derivation.html.

Tibshirani, Robert. 1994. "Regression Shrinkage and Selection Via the Lasso." *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B* 58:267–288.

Zhang, Cun-Hui, and Stephanie S. Zhang. 2014. "Confidence intervals for low dimensional parameters in high dimensional linear models." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 76 (1): 217–242. https://doi.org/https://doi.org/10.1111/rssb.12026. eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/rssb.12026. https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12026.

Zou, Hui. 2006. "The Adaptive Lasso and Its Oracle Properties." *Journal of the American Statistical Association* 101 (476): 1418–1429. https://doi.org/10.1198/016214506000000735. eprint: https://doi.org/10.1198/016214506000000735. https://doi.org/10.1198/016214506000000735.

Appendix A

R Codes for the Gibbs Sampler

R codes to build the Gibbs Sampler based on Hans (2009) without imposing any prior on τ and σ.

```
sigma <- 1

tau <- 0.5

p <- 10

s <- 3

n <- 100

beta0 <- c(rep(1,s),rep(0,p-s))

Sigma <- .5^abs(outer(1:p,1:p,"-"))

Z <- scale(matrix(rnorm(n*p),n,p),center=TRUE,scale= FALSE)

x <- Z %*% chol(Sigma)

error <- rnorm(n,0,sigma)

y <- as.numeric(x %*% beta0 + error - mean(error))


ols <- lm(y ~ 0 + x)

summary(ols) #checkin the OLS


#beta_ols<-sapply(1:p,

            function(x) {ols$coefficients[[x]]})


beta_ols<- ols$coefficients
```

```
#note on this ols$coefficients


tau = 0.5
sigma = 1


set.seed(2020)
#initial beta_j;
#beta = beta_ols*runif(5,min=0, max=5)


iter=10000


#Initializing beta-storage matrix for ith batch
beta_store = matrix( NA , iter, p)
beta = rep(0, p)


for(i in 1:iter){


  for(j in 1:p){


    #square root of L-2 norm
    sxj = sqrt(sum(x[,j]^2))


    #positive truncated
    mu_j_plus = beta_ols[j] + (1/ (sxj^2) ) *
                as.numeric( t(x[,j]) %*% (x[,-j] %*%
                (beta_ols[-j] - beta[-j])) ) - tau*sigma/(sxj^2)
    #negative truncated
```

```
mu_j_minus = beta_ols[j] + (1/ (sxj^2)) *
            as.numeric( t(x[,j]) %*% (x[,-j] %*%
            (beta_ols[-j] - beta[-j])) ) + tau*sigma/(sxj^2)

a_j = ( 1 - pnorm( - mu_j_plus/ (sigma/sxj) ) )/
        dnorm(  - mu_j_plus/ ( sigma/sxj ) )
b_j =  pnorm(  - mu_j_minus/ (sigma/sxj) )/
        dnorm( - mu_j_minus/ (sigma/sxj) )

w_j = a_j/(a_j + b_j)

z_j = sample( c(-1,1),size=1, prob = c(w_j, 1-w_j) )
#We're taking "-1" for the first statement in the next line.

if(z_j == -1){
  #the positive truncated normal
  beta[j] = mu_j_plus + (sigma/sxj)  *
            qnorm( (1-pnorm(- mu_j_plus/ (sigma/sxj) ) ) *
            runif(1) )

} else if(z_j == 1) {
  #negative truncated normal
  beta[j] = mu_j_minus + (sigma/sxj) *
            qnorm( (pnorm(-mu_j_minus/ (sigma/sxj) ) ) *
            runif(1) )
}
#beta_j changed to beta[j].
```

```
  }
  beta_store[i,] = beta
}

beta_store = beta_store[-(1:2000),]
#discarding burn-in period

head(beta_store)
tail(beta_store)
#beta_post_mean = sapply(1:p,
                function(x){mean(beta_store[,x])} )

apply(beta_store,2,mean)
apply(beta_store,2, median)
```