On Ridge Regression and Least Absolute Shrinkage and Selection Operator

by

Hassan AlNasser
B.Sc., University of Victoria, 2014

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

MASTER OF SCIENCE

in the Department of Mathematics and Statistics

On Ridge Regression and Least Absolute Shrinkage and Selection Operator

by

Hassan AlNasser
B.Sc., University of Victoria, 2014

Supervisory Committee

Dr. Jane Ye, Co-Supervisor
(Department of Mathematics and Statistics)

Dr. Julie Zhou, Co-Supervisor
(Department of Mathematics and Statistics)

Supervisory Committee

Dr. Jane Ye, Co-Supervisor
(Department of Mathematics and Statistics)

Dr. Julie Zhou, Co-Supervisor
(Department of Mathematics and Statistics)

ABSTRACT

This thesis focuses on ridge regression (RR) and least absolute shrinkage and selection operator $(lasso)$. Ridge properties are being investigated in great detail which include studying the bias, the variance and the mean squared error as a function of the tuning parameter. We also study the convexity of the trace of the mean squared error in terms of the tuning parameter. In addition, we examined some special properties of RR for factorial experiments. Not only do we review ridge properties, we also review $lasso$ properties because they are somewhat similar. Rather than shrinking the estimates toward zero in RR, the $lasso$ is able to provide a sparse solution, setting many coefficient estimates exaclty to zero. Furthermore, we try a new approach to solve the $lasso$ problem by formulating it as a bilevel problem and implementing a new algorithm to solve this bilevel program.

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisors Dr. Jane Ye and Dr. Julie Zhou for their immense help and support not only through my graduate study but also for their guidance. Without them, this thesis would not be possible. I feel privileged to have had the opportunity to work and learn from them. I would also like to thank Dr. Jin Zhang for his help with the algorithm. A special thanks goes to everyone who has helped me directly or indirectly to make this thesis achievable.

DEDICATION

I would like to dedicate this thesis to my family because without them I would not be where I am today.

Chapter 1

INTRODUCTION

This chapter introduces linear regression model and ordinary least squares method. We also talk about the properties of least squares estimator (LSE) and use an example to demostrate the use of ordinary least squares approach to a real data set. However, there are several issues with LSE that we will explore in depth and try to overcome in later chapters.

1.1 Linear Regression

A methodology known as regression studies the relationship between variables. The relations between variables are usually approximated by functions. Francis Galton, in the late 1880s, wondered if he could predict men's height based on their fathers' height . He hypothesized that men's heights would depend on their fathers' heights, i.e. the taller the father is, the taller the son would be. Galton (1889) plotted the heights of 14 fathers and their sons' heights and tried to fit a straight line through the data, see Figure 1.1.

Let us denote the son's height by y and the father's height by x, so the

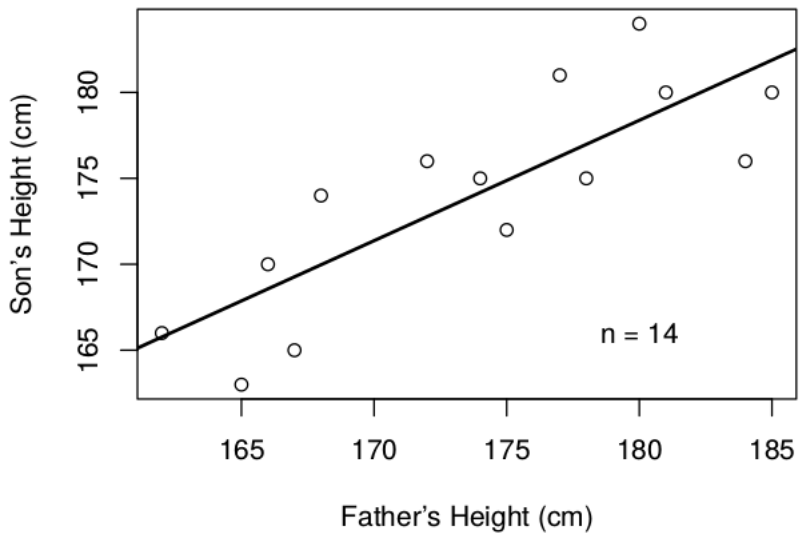

Figure 1.1: Fathers and Sons Heights for 14 Pairs of Father and Son. The straight line is the least square fitted line.

relation between y and x can be written as:

$$y = \alpha + \beta x + \epsilon.$$

This equation refers to the simple linear regression model where y is called a dependent variable, x is called a predictor variable, and ϵ is called a random error. The other symbols α and β are called the regression coefficients (parameters).

Now, we introduce a multiple linear regression model, in which many statistical learning approaches are based on. A mutliple linear regression model, in general, studies the relationship between a response variable y_i and several predictors or features, $\mathbf{x}_i = (x_{i1}, ..., x_{ip})$, for a given n samples, $(\mathbf{x}_i, y_i)_{i=1}^n$. This model can be written as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \epsilon_i, \qquad \text{for } i = 1, 2, ..., n. \tag{1.1}$$

The coefficients, $\beta_0, \beta_1, ..., \beta_p$, are unknown, and the random error terms, $\epsilon_1, ..., \epsilon_n$, are often assumed independent and identically distributed (i.i.d.) with mean zero and variance σ^2.

Also, model (1.1) can be alternatively written in a matrix form as:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{1.2}$$

where

$$\boldsymbol{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \boldsymbol{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1p} \\ 1 & x_{21} & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix}, \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}, \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}.$$

From the model assumptions, we have

$$E(\boldsymbol{\epsilon}) = \boldsymbol{0} \quad \text{and} \quad Cov(\boldsymbol{\epsilon}) = \sigma^2 \boldsymbol{I}_n, \tag{1.3}$$

where $\boldsymbol{I}_n$ is the identity matrix of order n, i.e.,

$$\boldsymbol{I}_n = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}_{n \times n}.$$

As seen above, the parameter vector $\boldsymbol{\beta}$ is unknown and must be estimated. A very well-studied approach is to use LSE.

1.2 Least Squares Estimation

The LSE $\hat{\boldsymbol{\beta}}$ of $\boldsymbol{\beta}$ is defined by

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \quad RSS(\boldsymbol{\beta}),$$

where RSS is the sume of squared residuals and is defined as follows

$$RSS(\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip}))^2. \tag{1.4}$$

For $n > p$ and a nonsingular matrix $\boldsymbol{X}^T\boldsymbol{X}$, we have

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}.$$

The fitted (predicted) values of the response variable are given by

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2} + ... + \hat{\beta}_p x_{ip}, \quad \text{for } i = 1, 2, ..., n.$$

We try, here, to demonstrate the LSE by fitting the following data set. The data was obtained from the Office of Instituational Research at the Uinversity of New Mexico in Albuquerque, NM. The objective is to estimate fall undergraduate enrollment at the university, given 29 years worth of data with five variables (YEAR, ROLL, UNEM, HGRAD, INC) described in Table 1.1. The first few rows of the data set are shown in Table 1.2 and a summary of the data set is given in Table 1.3. We call the data set UE in the analysis below.

Table 1.1: Undergraduate Enrollments (UE).

Variables	Description
YEAR	1961 = 1, $\cdots$, 1989 = 29
ROLL	Fall Undergraduate Enrollment
UNEM	January Unemployment Rate for New Mexico
HGRAD	Number of Spring High School Graduates in New Mexico
INC	Per Capita Income in Albuquerque (1961 US dollars)

Table 1.2: The First Few Rows of the UE Data Set.

	YEAR	ROLL	UNEM	HGRAD	INC
1	1	5,501	8.100	9,552	1,923
2	2	5,945	7	9,680	1,961
3	3	6,629	7.300	9,731	1,979
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.

Table 1.3: Summary of the UE Data Set.

Statistic	Obs.	Mean	St. Dev.	Min	Max
YEAR	29	15.000	8.515	1	29
ROLL	29	12,707.030	3,254.077	5,501	16,081
UNEM	29	7.717	1.123	5.700	10.100
HGRAD	29	16,528.140	2,926.927	9,552	19,800
INC	29	2,729.483	461.429	1,923	3,345

Suppose we want to fit a multiple linear regression model with ROLL as the response variable and YEAR, UNEM, HGRAD and INC as the explantory variables. Table 1.4 gives the estimated coefficients and the standard errors of the estimates.

Table 1.4: LSE of UE.

	Dependent variable:ROLL
	LSE
	(St. Error)
YEAR	144.173**
	(51.568)
UNEM	336.778***
	(112.320)
HGRAD	0.533***
	(0.081)
INC	1.184
	(1.189)
Constant	−4,090.590*
	(2,037.305)
Observations	29
R^2	0.971
Adjusted R^2	0.967
Residual Std. Error	594.300 (df = 24)
F Statistic	203.866*** (df = 4; 24)
Note:	*p-value<0.1; **p-value<0.05; ***p-value<0.01

Therefore, we can write the fitted model as

$$\hat{y} = -4090.590 + 144.173\ \textit{Year} + 336.778\ \textit{UNEM} + 0.533\ \textit{HGRAD} + 1.184\ \textit{INC}.$$

Properties of LSE

Some of the well-known properties of LSE are listed below. For further explanation, see Montgomery et al. (2013).

- The LSE is unbiased, $E[\hat{\boldsymbol{\beta}}] = \boldsymbol{\beta}$.
- The covariance of $\hat{\boldsymbol{\beta}}$ is given by $Var[\hat{\boldsymbol{\beta}}] = \sigma^2(\boldsymbol{X}^T\boldsymbol{X})^{-1}$.
- The residuals sum up to zero, $\sum_{i=1}^{n} e_i = 0$, where $e_i = y_i - \hat{y}_i$.
- The residuals and predictors are uncorrelated, $\sum_{i=1}^{n} \mathbf{x}_i e_i = \mathbf{0}$.
- The predicted values and residuals are uncorrelated, $\sum_{i=1}^{n} \hat{y}_i e_i = 0$.

1.3 Research Problems

Although there are various good properties of LSE, there are issues with the least squares estimates: (i) there is more variability in the least squares fit as n, the number of observations, gets closer to p, the number of regressors. See, for example, James et al. (2013); (ii) there is no unique solution for LSE if $p > n$ because the matrix $(\boldsymbol{X}^T\boldsymbol{X})$ is not invertible in this case, Hastie et al. (2015); and (iii) there may be some irrelevant variables included in the multiple linear regression model. There are several methods that deal with these issues.

In this thesis, we study and review two estimation methods: ridge regression, RR, and least absolute shrinkage and selection operator, $lasso$. We particularly focus on studying various properties of each method and derive new ones as well as proposing a new algorthim to compute estimates for the $lasso$. We will compare our results with those already obtained from proposed algorthims in the literature.

1.4 Main Contributions

The main contributions in this thesis can be summarized as follows:

- reviewing ridge regression properties, which includes the bias, the variance, the mean squared error and the convexity of the trace of the mean squared error as a function of the tuning parameter,
- adding new properties of RR applied to factorial experiments,
- reviewing the $lasso$ properties for regression models with many covariates,
- formulating the $lasso$ into a bilevel problem,
- implementing an algorithm to solve the $lasso$ bilevel problem,
- applying various data sets to both RR and the $lasso$.

Chapter 2

RIDGE REGRESSION

In this chapter, we review ridge regression, explore its properties and derive some new results, which are in Sections 2.1 and 2.2. In Section 2.3, we present several real data sets, which will be analyzed in Section 2.4 using RR. In Section 2.4, a K-fold cross-validation procedure is also given for RR. In Section 2.5, we apply RR to factorial designs and study its properties.

2.1 Introduction

Recently, high-dimensional shrinkage and parameter selection techniques have been of great importance. Tikhonov regularization, another name for RR, deals with many predictors and an ill-conditioned model matrix $\boldsymbol{X}$ (i.e. $\boldsymbol{X}^T\boldsymbol{X}$ is not invertible or near singular). Fitting a model with many predictors and no regularization results in a non-unique LSE's solution. Furthermore, LSE depends on $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ where if the $rank(\boldsymbol{X}) < p$, then $(\boldsymbol{X}^T\boldsymbol{X})$ doesn't have an inverse. However, RR has the ability to overcome these hurdles by constraining the coefficient estimates; hence, it can reduce the estimator's variance and introduce some bias, James et al. (2013).

Similar to LSE, RR coefficients are estimated by minimizing

$$S(\boldsymbol{\beta}, \lambda) = RSS(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} \beta_j^2, \tag{2.1}$$

over $\boldsymbol{\beta}$ for a given λ, where λ is called a tuning parameter and $\lambda \geq 0$. Notice that $RSS(\boldsymbol{\beta})$ is defined in (1.4). The ridge regression estimator (RRE) is denoted by $\hat{\boldsymbol{\beta}}^R(\lambda)$.

Tikhonov regularization tries to find estimates that fit the data reasonably well, by making the $RSS(\boldsymbol{\beta})$ small, while the term, $\lambda \sum_{j=1}^{p} \beta_j^2$, is also small when the estimates, $(\hat{\beta}_1, ..., \hat{\beta}_p)$ are shrunken approximately to zero (James et al. (2013)). Figure 2.1 gives an interpretation of LSE and RRE in 2-dimensional space with $\boldsymbol{\beta} = (\beta_1, \beta_2)^T$, where $\hat{\boldsymbol{\beta}}$ represents the LSE and the contour plots are for $RSS(\boldsymbol{\beta})$. The RRE, $\hat{\boldsymbol{\beta}}^R(\lambda)$, is the point where the contour of $RSS(\boldsymbol{\beta})$ meets the circle defined by $\beta_1^2 + \beta_2^2 = r^2$. If we define (2.1) in vector and vector norm representation, the representation is as follows:

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \| \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} \|^2 \\ \text{subject to} \quad & \| \boldsymbol{\beta} \|_2^2 \leq r^2, \end{aligned} \tag{2.2}$$

for $r^2 > 0$ and $\| \boldsymbol{\beta} \|_2^2 = \sum_{j=1}^{p} \beta_j^2$, and r^2 is related to λ.

The tuning parameter, λ, plays a very crucial part here. It controls the relative impact of the two terms in $S(\boldsymbol{\beta}, \lambda)$. For example, when $\lambda = 0$, RR becomes LSE, but when λ gets larger, $\| \hat{\boldsymbol{\beta}}^R(\lambda) \|_2^2$ shrinks; hence, the coefficient estimates of RR approach zero. Different from LSE, for every value of λ, we get a different set of coefficient estimates, $\hat{\boldsymbol{\beta}}^R(\lambda)$.

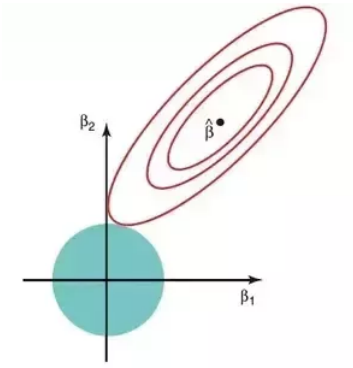

Figure 2.1: Geometric Interpretation of RRE in 2-Dimensional Space.

Looking at $S(\boldsymbol{\beta}, \lambda)$, we notice that the shrinkage penalty, $\lambda \sum_{j=1}^{p} \beta_j^2$, is only applied to $\beta_1, \beta_2, ..., \beta_p$ and not to the intercept β_0. The intercept term is a measure of the mean value of the response if $x_{i1} = x_{i2} = ... = x_{ip} = 0$. To omit β_0, we standardize the predictors, meaning that each column is centered so that $\frac{1}{n} \sum_{i=1}^{n} x_{ij} = 0$ and $\frac{1}{n} \sum_{i=1}^{n} x_{ij}^2 = 1$, for $j = 1, 2, ..., p$. Note here that standardization is done if the features are in different units. For convenience, we also center the outcome values such that $\frac{1}{n} \sum_{i=1}^{n} y_i = 0$. Once the RR coefficient estimates are obtained, we can recover the intercept term by

$$\hat{\beta}_0 = \bar{y} - \sum_{j=1}^{p} \bar{x}_j \hat{\beta}_j^R(\lambda),$$

where $\bar{y}$ and $\bar{x}_j$, for $j = 1, 2, ..., p$, are the original means. Using vector and vector norm notation, we can write the RR problem as

$$\min_{\beta \in \mathbb{R}^p} S(\boldsymbol{\beta}, \lambda) \quad = \min_{\beta \in \mathbb{R}^p} \quad \| \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} \|_2^2 + \lambda \| \boldsymbol{\beta} \|_2^2 \quad \text{for some} \quad \lambda \geq 0,$$

where $\boldsymbol{\beta} = (\beta_1, \beta_2, ..., \beta_p)$ and $\| \boldsymbol{\beta} \|_2^2 = \sum_{j=1}^{p} \beta_j^2$. Here the matrix $\boldsymbol{X}$ does not include the column of ones. This equation (2.1) is usually refered to as the Lagrangian form of RR.

Another format of RR problem is given as follows:

$$S(\boldsymbol{\beta}, \lambda) = \sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \sum_{j=1}^{p}(0 - \sqrt{\lambda}\beta_j)^2. \tag{2.3}$$

This approach reconstructs another LSE problem for an expanded data set by adding p observations, which leads to an expanded model matrix $\boldsymbol{X}_\lambda$ and vector $\boldsymbol{Y}_\lambda$ below:

$$\boldsymbol{X}_\lambda = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \\ \sqrt{\lambda} & 0 & \dots & 0 \\ 0 & \sqrt{\lambda} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sqrt{\lambda} \end{bmatrix}, \quad \boldsymbol{Y}_\lambda = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

It is clear that

$$\boldsymbol{X}_\lambda = \begin{bmatrix} \boldsymbol{X} \\ \sqrt{\lambda}\boldsymbol{I}_p \end{bmatrix}, \quad \boldsymbol{Y}_\lambda = \begin{bmatrix} \boldsymbol{Y} \\ \boldsymbol{0} \end{bmatrix},$$

which eventually yields the RR solution, Hoerl and Kennard (1970),

$$\hat{\boldsymbol{\beta}}^R(\lambda) = (\boldsymbol{X}_\lambda^T\boldsymbol{X}_\lambda)^{-1}\boldsymbol{X}_\lambda^T\boldsymbol{Y}_\lambda = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T\boldsymbol{Y}, \tag{2.4}$$

in the form of the LSE.

2.2 Properties of Ridge Regression

Here we review several properties of RRE. They are studied in the literature. See, for example, van Wieringen (2015), Hoerl and Kennard (1970), Marquardt and Snee (1975), Montgomery et al.(2013).

2.2.1 Bias of RRE

From (1.2), (1.3) and (2.4), we have

$$E[\hat{\boldsymbol{\beta}}^R(\lambda)] = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta}. \tag{2.5}$$

Proof. From (2.4),

$$\begin{aligned} E[\hat{\boldsymbol{\beta}}^R(\lambda)] &= E[(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T\boldsymbol{Y}] \\ &= (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T E[\boldsymbol{Y}] \\ &= (\lambda\boldsymbol{I}_p + \boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta}, \quad \text{from (1.2) and (1.3).} \quad \square \end{aligned}$$

Therefore, the ridge estimator is biased since $E[\hat{\boldsymbol{\beta}}^R(\lambda)] \neq \boldsymbol{\beta}$ and, in general, $E[\hat{\boldsymbol{\beta}}^R(\lambda)]$ goes to zero as $\lambda \longrightarrow \infty$:

$$\lim_{\lambda\longrightarrow\infty} E[\hat{\boldsymbol{\beta}}^R(\lambda)] = \lim_{\lambda\longrightarrow\infty} (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{0}_p,$$

for any $\boldsymbol{\beta}$ and $\boldsymbol{X}$. As a special case, when $\lambda = 0$, we get the LSE, and it is unbiased.

Here we introduce the bias of RRE, $bias(\hat{\boldsymbol{\beta}}^R(\lambda))$. First, we define $A(\lambda) = \lambda\boldsymbol{I}_p + \boldsymbol{X}^T\boldsymbol{X}$ and from there, the calculation goes as follows:

$$
\begin{aligned}
bias(\hat{\boldsymbol{\beta}}^R(\lambda)) &= E[\hat{\boldsymbol{\beta}}^R(\lambda) - \boldsymbol{\beta}] \\
&= A^{-1}(\lambda)\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{\beta}, \qquad \text{from (2.5)} \\
&= A^{-1}(\lambda)(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p - \lambda\boldsymbol{I}_p)\boldsymbol{\beta} - \boldsymbol{\beta} \\
&= (\boldsymbol{I}_p - \lambda A^{-1}(\lambda))\boldsymbol{\beta} - \boldsymbol{\beta} \\
&= -\lambda A^{-1}(\lambda)\boldsymbol{\beta}.
\end{aligned}
$$

2.2.2 Variance of RRE

The variance of RRE is given by

$$
Var[\hat{\boldsymbol{\beta}}^R(\lambda)] = \sigma^2 A^{-1}(\lambda)\boldsymbol{X}^T\boldsymbol{X}A^{-1}(\lambda).
$$

Proof. From (2.4),

$$
\begin{aligned}
Var[\hat{\boldsymbol{\beta}}^R(\lambda)] &= Var[(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T\boldsymbol{Y}] \\
&= Var[A^{-1}(\lambda)\boldsymbol{X}^T\boldsymbol{Y}] \\
&= A^{-1}(\lambda)\boldsymbol{X}^T Var[\boldsymbol{Y}](A^{-1}(\lambda)\boldsymbol{X}^T)^T \\
&= A^{-1}(\lambda)\boldsymbol{X}^T Var[\boldsymbol{\epsilon}]\boldsymbol{X}A^{-1}(\lambda) \\
&= \sigma^2 A^{-1}(\lambda)\boldsymbol{X}^T\boldsymbol{X}A^{-1}(\lambda), \qquad \text{from (1.3).} \quad \square
\end{aligned}
$$

Notice, the variance of ridge estimator, $Var[\hat{\boldsymbol{\beta}}^R(\lambda)]$, goes to zero as $\lambda \longrightarrow \infty$:

$$\begin{aligned} \lim_{\lambda \longrightarrow \infty} Var[\hat{\boldsymbol{\beta}}^R(\lambda)] &= \lim_{\lambda \longrightarrow \infty} \sigma^2 A^{-1}(\lambda) \boldsymbol{X}^T \boldsymbol{X} A^{-1}(\lambda) \\ &= \mathbf{0}_{p \times p}. \end{aligned}$$

2.2.3 Mean Squared Error of RRE

The mean squared error of RRE, $MSE(\hat{\boldsymbol{\beta}}^R(\lambda))$, is given by:

$$MSE(\hat{\boldsymbol{\beta}}^R(\lambda)) = \sigma^2 A^{-1}(\lambda) \boldsymbol{X}^T \boldsymbol{X} A^{-1}(\lambda) + \lambda^2 A^{-1}(\lambda) \boldsymbol{\beta} \boldsymbol{\beta}^T A^{-1}(\lambda).$$

Proof.

$$\begin{aligned} MSE(\hat{\boldsymbol{\beta}}^R(\lambda)) &= Var[\hat{\boldsymbol{\beta}}^R(\lambda)] + bias(\hat{\boldsymbol{\beta}}^R(\lambda)) bias^T(\hat{\boldsymbol{\beta}}^R(\lambda)) \\ &= \sigma^2 A^{-1}(\lambda) \boldsymbol{X}^T \boldsymbol{X} A^{-1}(\lambda) + (-\lambda A^{-1}(\lambda) \boldsymbol{\beta})(-\lambda A^{-1}(\lambda) \boldsymbol{\beta})^T \\ &= \sigma^2 A^{-1}(\lambda) \boldsymbol{X}^T \boldsymbol{X} A^{-1}(\lambda) + \lambda^2 A^{-1}(\lambda) \boldsymbol{\beta} \boldsymbol{\beta}^T A^{-1}(\lambda). \quad \square \end{aligned}$$

2.2.4 Trace of the Mean Squared Error of RRE

From Section 2.2.3, we can get the trace of the mean squared error of RRE, $m(\lambda)$, as follows:

$$\begin{aligned} m(\lambda) &= tr(MSE(\hat{\boldsymbol{\beta}}^R(\lambda))) \\ &= tr(\sigma^2 A^{-1}(\lambda)\boldsymbol{X}^T\boldsymbol{X}A^{-1}(\lambda) + \lambda^2 A^{-1}(\lambda)\boldsymbol{\beta}\boldsymbol{\beta}^T A^{-1}(\lambda)) \\ &= \sigma^2 tr[A^{-1}(\lambda)(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p - \lambda\boldsymbol{I}_p)A^{-1}(\lambda)] + \lambda^2\boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta} \\ &= \sigma^2[tr(A^{-1}(\lambda)) - \lambda tr(A^{-2}(\lambda))] + \lambda^2\boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta}. \end{aligned}$$

Let $m_v(\lambda) = \sigma^2[tr(A^{-1}(\lambda)) - \lambda tr(A^{-2}(\lambda))]$ and $m_b(\lambda) = \lambda^2\boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta}$, then $m(\lambda) = m_v(\lambda) + m_b(\lambda)$. Note that here $m_v(\lambda)$ stands for the trace of the variance term and $m_b(\lambda)$ stands for the trace of the bias term.

Differentiating $m_v(\lambda)$ and $m_b(\lambda)$ gives the following results:

- (i) : $\frac{dm_v(\lambda)}{d\lambda} < 0$ for all $\lambda > 0$.
- (ii) : $\frac{d^2m_v(\lambda)}{d\lambda^2} > 0$ for all $\lambda > 0$.
- (iii): $\frac{dm_b(\lambda)}{d\lambda} \geq 0$ for all $\lambda > 0$.

Here are the derivations for the above results. Let $u_1 \geq u_2 \geq ... \geq u_p$ be the eigenvalues of $\boldsymbol{X}^T\boldsymbol{X}$. Since $\boldsymbol{X}^T\boldsymbol{X}$ is positive semidefinite, we have $u_i \geq 0$ for all $i = 1, 2, ..., p$. From $A(\lambda) = \boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p$, $A(\lambda)$ has eigenvalues $(u_1 + \lambda), (u_2 + \lambda), ..., (u_p + \lambda)$ and $A^{-j}(\lambda)$ has eigenvalues $(u_1 + \lambda)^{-j}, (u_2 + \lambda)^{-j}, ..., (u_p + \lambda)^{-j}$, for $j = 1, 2$.

Proof of (i). From

$$\begin{aligned} m_v(\lambda) &= \sigma^2[tr(A^{-1}(\lambda)) - \lambda tr(A^{-2}(\lambda))] \\ &= \sigma^2 \left[\sum_{i=1}^{p} \frac{1}{u_i+\lambda} - \lambda \sum_{i=1}^{p} \frac{1}{(u_i+\lambda)^2}\right] \\ &= \sigma^2 \sum_{i=1}^{p} \frac{u_i}{(u_i+\lambda)^2}, \end{aligned}$$

$$\text{we get} \quad \frac{dm_v(\lambda)}{d\lambda} = \sigma^2 \sum_{i=1}^{p} \frac{-2u_i}{(u_i+\lambda)^3} < 0, \quad \text{for all } \lambda > 0.$$

Proof of (ii). From (i), we get

$$\frac{d^2 m_v(\lambda)}{d\lambda^2} = \sigma^2 \sum_{i=1}^{p} \frac{6u_i}{(u_i+\lambda)^4} > 0, \quad \text{for all } \lambda > 0.$$

Proof of (iii). From

$$\begin{aligned} m_b(\lambda) &= \lambda^2 \boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta}, \\ \text{we get} \quad \frac{dm_b(\lambda)}{d\lambda} &= 2\lambda \boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta} + \lambda^2 \frac{d}{d\lambda}(\boldsymbol{\beta}^T A^{-1}(\lambda) A^{-1}(\lambda)\boldsymbol{\beta}) \\ &= 2\lambda \boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta} + \lambda^2 \boldsymbol{\beta}^T \frac{dA^{-1}(\lambda)}{d\lambda} A^{-1}(\lambda)\boldsymbol{\beta} \\ &\quad + \lambda^2 \boldsymbol{\beta}^T A^{-1}(\lambda) \frac{dA^{-1}(\lambda)}{d\lambda}\boldsymbol{\beta} \\ &= 2\lambda \boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta} - 2\lambda^2 \boldsymbol{\beta}^T A^{-3}(\lambda)\boldsymbol{\beta} \\ &= 2\lambda \boldsymbol{\beta}^T [A^{-2}(\lambda) - \lambda A^{-3}(\lambda)]\boldsymbol{\beta} \\ &= 2\lambda \boldsymbol{\beta}^T B(\lambda)\boldsymbol{\beta}, \quad \text{with } B(\lambda) = A^{-2}(\lambda) - \lambda A^{-3}(\lambda). \end{aligned}$$

Since $B(\lambda)$ has eigenvalues: $\dfrac{1}{(u_i+\lambda)^2} - \dfrac{\lambda}{(u_i+\lambda)^3}$

$$= \frac{u_i}{(u_i+\lambda)^3} \geq 0, \quad i = 1, 2, ..., p, \text{for } \lambda > 0.$$

$$\text{Thus,} \quad \frac{dm_b(\lambda)}{d\lambda} \geq 0, \quad \text{for all } \lambda > 0 \text{ and } \boldsymbol{\beta}.$$

We used the fact that $A(\lambda)A^{-1}(\lambda) = \boldsymbol{I}$ in proving (iii). Given $A(\lambda)A^{-1}(\lambda) = \boldsymbol{I}_p$, we obtain an expression for $\frac{dA^{-1}(\lambda)}{d\lambda}$ as given below:

$$\begin{aligned} \frac{dA(\lambda)}{d\lambda}\, \mathrm{A}^{-1}(\lambda) + A(\lambda)\tfrac{dA^{-1}(\lambda)}{d\lambda} &= \mathbf{0}, \\ \boldsymbol{I}_p A^{-1}(\lambda) + A(\lambda)\frac{dA^{-1}(\lambda)}{d\lambda} &= \mathbf{0}, \\ \text{which gives,} \quad \frac{dA^{-1}(\lambda)}{d\lambda} &= -A^{-1}(\lambda)A^{-1}(\lambda) = -A^{-2}(\lambda). \end{aligned}$$

These results show that $m_v(\lambda)$ is a strictly decreasing and convex function of λ, while $m_b(\lambda)$ is an increasing function.

We use a linear regression model to illustrate the functions $m_v(\lambda)$, $m_b(\lambda)$ and $m(\lambda)$ for $\lambda > 0$. The linear regression model is

$$y = 0.9 \times x_1 + 0.3 \times x_2 + 2.8 \times x_3 + \epsilon,$$

and 30 observations are generated as follows:

- x_1 has a normal distribution with mean zero and standard deviation two;
- x_2 has a uniform distribution on the interval $(3, 10)$;
- x_3 has a normal distribution with mean zero and standard deviation 1.2; and
- ϵ has a standard normal distribution with mean zero and standard deviation one.

For this model, we have $\beta = (0.9, 0.3, 2.8)^T$, $p = 3$, $\sigma^2 = 1$. Figure 2.2 shows the plots of $m_v(\lambda)$ vs. λ, $m_b(\lambda)$ vs. λ and $m(\lambda)$ vs. λ.

Figure 2.3 shows RRE for different values of λ and the results are consistent with the properties discussed in this section.

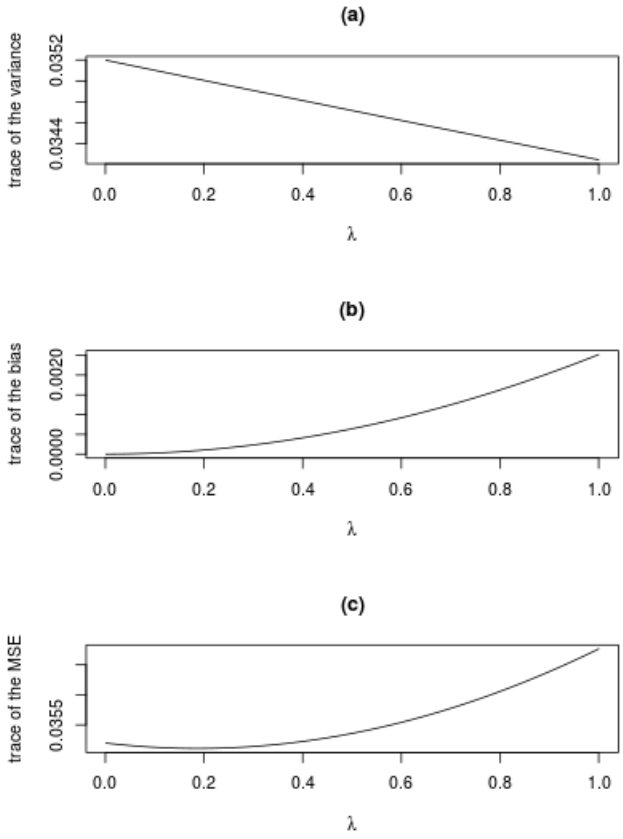

Figure 2.2: The Plots of Variance, Bias and the MSE: (a) $m_v(\lambda)$ vs. λ, (b) $m_b(\lambda)$ vs. λ, (c) $m(\lambda)$ vs. λ.

More Properties about the Trace of the Mean Squared Error of RRE

(M1) $m_v(\lambda) \longrightarrow 0$ as $\lambda \longrightarrow \infty$.

(M2) $m_b(\lambda) = \lambda^2 \boldsymbol{\beta}^T A^{-2}(\lambda) \boldsymbol{\beta} \longrightarrow \boldsymbol{\beta}^T \boldsymbol{\beta}$ as $\lambda \longrightarrow \infty$.

(M3) $m_b(\lambda)$ is a concave function for $\lambda \in (\frac{u_1}{2}, \infty)$.

Proof of $(\boldsymbol{M1})$

$$m_v(\lambda) = \sigma^2 \sum_{i=1}^{p} \frac{u_i}{(u_i + \lambda)^2} \longrightarrow 0, \quad \text{as } \lambda \longrightarrow \infty.$$

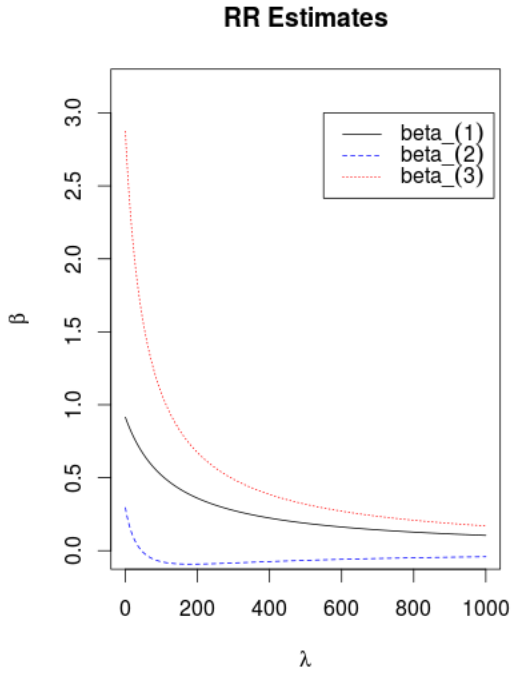

Figure 2.3: RRE for Various Values of λ.

Proof of $(\boldsymbol{M2})$

$$m_b(\lambda) = \lambda^2 \boldsymbol{\beta}^T A^{-2}(\lambda) \boldsymbol{\beta},$$

$\lambda^2 A^{-2}(\lambda)$ has eigenvalues $\frac{\lambda^2}{(u_i+\lambda)^2} \longrightarrow 1$, as $\lambda \longrightarrow \infty$ for all $i = 1, 2, ..., p$.

Therefore, $m_b(\lambda) \longrightarrow \boldsymbol{\beta}^T \boldsymbol{\beta}$, as $\lambda \longrightarrow \infty$.

Proof of $(\boldsymbol{M3})$. Similar to the proof in (iii), we have

$$\begin{aligned}
\frac{d^2 m_b(\lambda)}{d\lambda^2} &= 2\boldsymbol{\beta}^T A^{-2}(\lambda)\boldsymbol{\beta} - 4\lambda\boldsymbol{\beta}^T A^{-3}(\lambda)\boldsymbol{\beta} - 4\lambda\boldsymbol{\beta}^T A^{-3}(\lambda)\boldsymbol{\beta} + 6\lambda^2\boldsymbol{\beta}^T A^{-4}(\lambda)\boldsymbol{\beta} \\
&= 2\boldsymbol{\beta}^T[A^{-2}(\lambda) - 2\lambda A^{-3}(\lambda) - 2\lambda A^{-3}(\lambda) + 3\lambda^2 A^{-4}(\lambda)]\boldsymbol{\beta} \\
&= 2\boldsymbol{\beta}^T[A^{-2}(\lambda) - 4\lambda A^{-3}(\lambda) + 3\lambda^2 A^{-4}(\lambda)]\boldsymbol{\beta} \\
&= 2\boldsymbol{\beta}^T \boldsymbol{c}(\lambda)\boldsymbol{\beta},
\end{aligned}$$

where $\boldsymbol{c}(\lambda) = A^{-2}(\lambda) - 4\lambda A^{-3}(\lambda) + 3\lambda^2 A^{-4}(\lambda)$, and it has eigenvalues:

$$\begin{aligned}
\text{eig}(\boldsymbol{c}(\lambda)) &= \frac{1}{(u_i+\lambda)^2} - \frac{4\lambda}{(u_i+\lambda)^3} + \frac{3\lambda^2}{(u_i+\lambda)^4} \\
&= \frac{1}{(u_i+\lambda)^4}[(u_i+\lambda)^2 - 4\lambda(u_i+\lambda) + 3\lambda^2] \\
&= \frac{1}{(u_i+\lambda)^4}u_i^2 + 2\lambda u_i + \lambda^2 - 4\lambda u_i - 4\lambda^2 + 3\lambda^2 \\
&= \frac{1}{(u_i+\lambda)^4}u_i^2 - 2\lambda u_i \\
&= \frac{u_i(u_i-2\lambda)}{(u_i+\lambda)^4} < 0, \qquad \text{if } \lambda > \frac{u_1}{2} \text{ for all } i = 1, 2, ..., p.
\end{aligned}$$

This means $\boldsymbol{c}(\lambda)$ is negative definite for $\lambda > \frac{u_1}{2}$. Thus, $\frac{d^2 m_b(\lambda)}{d\lambda^2} = 2\boldsymbol{\beta}^T \boldsymbol{c}(\lambda)\boldsymbol{\beta} \leq 0$ for all $\boldsymbol{\beta}$, when $\lambda > \frac{u_1}{2}$, which implies $m_b(\lambda)$ is a concave function of λ.

2.2.5 Degrees of Freedom

In general, the degrees of freedom for LSE is $tr(H)$, where $H = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T$. Analogous to LSE, the hat matrix for RR is $H_\lambda = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T$. Hence, the degrees of freedom is given by the trace of the ridge hat matrix.

$$tr[\boldsymbol{H}_\lambda] = tr[\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}_p)^{-1}\boldsymbol{X}^T].$$

We can show that $tr[H_\lambda] = \sum_{i=1}^{p} \frac{u_i}{u_i+\lambda}$, which is a decreasing function of λ. When $\lambda = 0, tr[H_\lambda] = tr[H] = p$; however, when $\lambda \longrightarrow \infty$, $tr[H_\lambda] \longrightarrow 0$.

2.3 Data Sets

In this section, three data sets will be fully explained, so it will be easier to refer to them later. We use three different data sets, depending on the number of observations, n, and the number of features (covariates), p, in our analysis when we apply RR and the $lasso$.

2.3.1 Prostate Cancer Data (PCD)

This data set consists of 97 observations (n = 97) from patients each with 9 features (p = 9). The study is done by Stamey et al. (1989) over the period of 38 months with a mean of 12 months. The data set measures the correlation between the level of prostate-specific atigen (the response variable) and some covariates described in Table 2.1. Table 2.2 and Table 2.3 display the first five rows of the **PCD** and the summary of the **PCD**, respectively.

Table 2.1: Prostate Cancer Data Descripition

Variables	Description	Units
lcavol	log(cancer volume)	log(cm^3)
lweight	log(prostate weight)	log($gram$)
age	age	years
lbph	log(benign prostatic hyperplasia amount)	
svi	seminal vesicle invasion	0/1
lcp	log(capsular penetration)	log(cm)
gleason	Gleason score	2-10
pgg45	percentage Gleason scores	0-100
lpsa	log(prostate specific antigen)	nano-gram/ milli-liter

Gleason score: it is a measure(2-10) of how progressive someone's cancer is. The higher the number, the more aggresive it is.

Table 2.2: First 5 Rows of **PCD**.

Patient	lcavol	lweight	age	lbph	svi	lcp	gleason	pgg45	lpsa
1	-0.580	2.769	50	-1.386	0	-1.386	6	0	-0.431
2	-0.994	3.320	58	-1.386	0	-1.386	6	0	-0.163
3	-0.511	2.691	74	-1.386	0	-1.386	7	20	-0.163
4	-1.204	3.283	58	-1.386	0	-1.386	6	0	-0.163
5	0.751	3.432	62	-1.386	0	-1.386	6	0	0.372
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.

Since the response variable in the following data sets (**BCNKID** and **GD**) is binary, logistic regression model is used to model the probability, $p_1 = P(y_i = 1)$, related to the features,

$$\log(\frac{p_1}{1-p_1}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p, \tag{2.6}$$

where y is the response variable. In logistic regression, RR minimizes

$$S_1(\boldsymbol{\beta}, \lambda) = -l(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} \beta_j^2,$$

Table 2.3: Summary of **PCD**.

Statistics	Obs.	Mean	St. Dev.	Min	Max
lcavol	97	1.350	1.179	−1.347	3.821
lweight	97	3.629	0.428	2.375	4.780
age	97	63.866	7.445	41	79
lbph	97	0.100	1.451	−1.386	2.326
svi	97	0.216	0.414	0	1
lcp	97	−0.179	1.398	−1.386	2.904
gleason	97	6.753	0.722	6	9
pgg45	97	24.381	28.204	0	100
lpsa	97	2.478	1.154	−0.431	5.583

lpsa is the response variable.

where $l(\beta)$ is the log-likelihood function. See Le Cessie and van Houwelingen (1992) for various properties of RR in logistic regression.

2.3.2 Breast Cancer NKI Dataset (BCNKID)

This data set contains the expression profiles of 337 breast cancer patients (n = 337), and each profile comprises expression levels of 24481 genes (p = 24481). **BCNKID** was obtained from Schroeder et al.(2017). Sample observations of the data set are displayed in Table 2.4; a summary of the data set is provided as well in Table 2.5. Before we start our analysis, we remove the genes with missing values; thus, the number of genes is reduced to 14318, i.e., $p = 14318$. For this data set, we try to study the relationship between estrogen receptor status (the response variable), which is an important prognostic indicator for breast cancer, and patients' gene expressions. The response variable takes on value 0 (no risk of recurrence or death from breast cancer) and 1 (there is risk of recurrence or death from breast cancer), and there are 88 zeros and 249 1's in the data set.

Table 2.4: Sample Observations of **BCNKID**.

Profile	Estrogen Receptor	Gene_1	Gene_2	. . .	Gene_14318
NKI_4	1	-0.267	0.059	. . .	-0.394
NKI_6	1	-0.310	-0.135	. . .	0.043
.	.	.	.	. . .	.
.	.	.	.	. . .	.
.	.	.	.	. . .	.
NKI_403	1	0.242	-0.220	. . .	-0.053
NKI_404	1	0.156	-0.585	. . .	-0.292

Table 2.5: Summary of **BCNKID**.

Statistics	Obs.	Mean	St. Dev.	Min	Max
Gene_1	337	-0.099	0.325	-2.0	2.0
Gene_2	337	-0.012	0.136	-0.585	0.310
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
Gene_14318	337	-0.255	0.478	-2.0	0.988

2.3.3 Golub Dataset (GD)

The Golub data set is a microarray data set which consists of 3051 features (p = 3051) and 38 samples (n = 38). This data set contains samples of acute lymphoblastic leukemia (ALL/0) and acute myeloid leukemia (AML/1) and was recovered from Pollard et al. (2017). Table 2.6 shows some obsevations from **GD** and Table 2.7 gives a summary of some features from **GD**. We try to study the relationship between an indicator variable for the leukemia type (the response variable) and gene expressions obtained from patients. The response variable takes 0 or 1 with 27 zeros and 11 ones.

Table 2.6: Sample Observations of **GD**.

Patient	leuk.type	Gene_1	Gene_2	. . .	Gene_3051
1	0	-1.458	-0.752	. . .	-0.861
2	0	-1.394	-1.263	. . .	-1.394
.	.	.	.	. . .	.
.	.	.	.	. . .	.
.	.	.	.	. . .	.
37	1	0.849	0.451	. . .	1.630
38	1	-0.665	-0.458	. . .	1.600

Table 2.7: Summary of **GD**.

Statistics	Obs.	Mean	St. Dev.	Min	Max
Gene_1	38	-1.129	0.588	-1.608	1.101
Gene_2	38	-0.847	0.529	-1.374	0.978
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
Gene_3051	38	-0.360	0.869	-1.427	1.905

2.4 Analysis of the Data Sets Using Ridge Regression

In this section, we apply RR to the data sets described in Section 2.3. We will neglect the UE data set because of the small sample size (n = 29). The sample has to be large enough to be devided into a training set and a test set (Hastie et al.(2015)). To perform RR, we will use the **glmnet** package in R. **Glmnet** package fits a generalized linear model via penalized maximum likelihood, and the algorthim in this package uses the cyclical coordinate descent method to find the optimal solution. Also, advantages of this package includes; the methods for prediction, plotting and the built in function that performs K-fold cross-validation. This function is used to select the tuning parameter λ that minimzies the mean squared error. Note that before we start our analysis, we try to expalin how K-fold cross-validation is performed.

2.4.1 K-Fold Cross-Validation

Phase 1

Take a data set and divide it into K different blocks: [| | ... |]. Each block has $\frac{n}{K}$ observations. The observations are randomly assigned, and if $\frac{n}{K}$ is not an integer, each block will have approximately equal size.

Phase 2

Cycle through these K different blocks treating each block as the test set, whereas on all the remaining blocks (observations), fit the model for every value of λ, for example [test set | obs. | ... | obs.]. For simplicity, assume we are dealing with a specific value for λ out of a set of possible values. We fit our model using all the remaining blocks for this λ while treating the first block as the test set. This

will enable us to assess λ's performance on the test set, thus, obtaining $error_1(\lambda)$, meaning the error on the first block:

$$error_1(\lambda) = \sum_{i \in 1^{st} block} (y_i - \hat{y}_i^{(1)}(\lambda))^2.$$

where $\hat{y}_i^{(1)}(\lambda)$ is the fitted value for y_i with RRE by fitting all observations without block 1.

Phase 3

Keep track of the error for the value of λ for all the blocks, i.e.

$$error_k(\lambda) = \sum_{i \in k^{th} block} (y_i - \hat{y}_i^{(k)}(\lambda))^2, \quad k = 1, 2, ..., K,$$

where $\hat{y}_i^{(k)}(\lambda)$ is the fitted values for y_i with RRE by fitting all the observations without block k.

Phase 4

We compute the cross-validation error of λ by

$$CV(\lambda) = \frac{1}{K} \sum_{k=1}^{K} error_k(\lambda).$$

Phase 5

Repeat this process for every value of λ. At the end, choose the λ that minimizes the mean cross-validated error, $CV(\lambda)$.

The algorithm

Step 1. Divide a data set T into K seperate sets of equal size

•T=(T$_1, T_2, ..., T_K$).

Step 2. For $k = 1, 2, ..., K$, use T_k as the test set and the rest of the $(K-1)$ blocks as the training set, and compute the fitted values $\hat{y}_i^{(k)}(\lambda)$.

Step 3. Compute the mean-squared-error on the observations in T_k, i.e., $error_k(\lambda) = \sum_{i \in k^{th} block}(y_i - \hat{y}_i^{(k)}(\lambda))^2$, and compute

$$CV(\lambda) = \frac{1}{K}\sum_{k=1}^{K} error_k(\lambda).$$

Step 4. Repeat **Steps 2 and 3** for various values of $\lambda > 0$.

Step 5. Find λ^* that minimizes $CV(\lambda)$.

In **Section 2.2**, we have derived and summarized various properties of RRE. Ideally, we want to choose λ to minimize $m(\lambda)$, the trace of the mean squared error of RRE. However, there are unknown parameters in $m(\lambda)$, such as $\boldsymbol{\beta}$ and σ^2. It is not possible to minimize $m(\lambda)$ in practical applications. Thus, the prediction error (cross-validation error) $CV(\lambda)$ is used to find a tuning parameter λ^* in RRE.

General Guideline for Selecting the Number of Folds K

According to Kohavi (1995), the best choice for K is 10. The choice of K will impact how much you introduce bias and variance into your analysis. A lower K results in less variance but more bias, whereas higher K results in more variance but less bias. In the following applications, we use $K = 5$ and $K = 10$ if possible.

2.4.2 PCD

Cross-Validation with 5-folds is first performed on **PCD** to get λ which yeilds the smallest mean cross-validated error. Figure 2.4 displays a range of tuning parameter values associated with their cross-validation errors. The value with the smallest mean error is $\lambda = 0.1501$ and ($\log(\lambda)$ = -1.8966). The test MSE with this tuning parameter is 0.8158. After choosing $\lambda = 0.1501$, we refit our RR model on the full data set to examine the coefficient estimates. Table 2.8 shows the estimates, and some estimates are shrinking towards zero.

We also perform the same analysis but with 10-folds instead. This gives $\lambda = 0.1246$ with the smallest mean cross-validated error. Figure 2.5 shows a range of tuning parameter values associated with their cross-validation errors. The test MSE with this tuning parameter is 0.8143. The coefficient estimates with $\lambda = 0.1246$ are included in Table 2.8. The same behavior is present here, some estimates are shrinking towards zero.

Note that, here we are able to get LS estimates because $p < n$. The model matrix $\boldsymbol{X}$ has full rank; hence, the LS coefficient estimates are also presented in Table 2.8. We note that, RR is able to shrink the estimates by introducing the penatly parameter λ as in (2.1). Since $p << n$ and p is small, the estimation from LSE and RR are similar in this application.

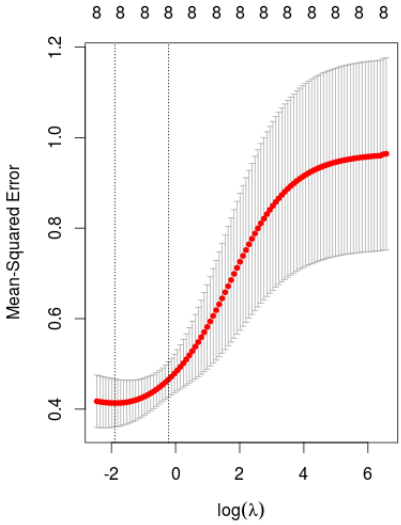

Figure 2.4: 5-Fold Cross-Validation of **PCD**. The corss-validation curve is the red-dotted line with upper and lower standard deviation curves along the λ sequence. The two vertical dotted lines indicate the selected λ. The vertical dotted line on the left gives the minimum mean cross-validated error, and the right vertical dotted line gives a regularized model such that the error is within one standard error of the minimum.

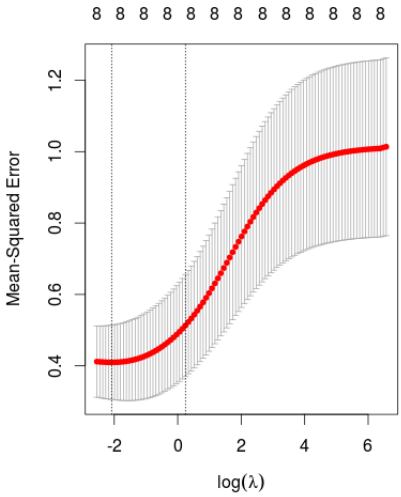

Figure 2.5: 10-Fold Cross-Validation of **PCD**. The corss-validation curve is the red-dotted line with upper and lower standard deviation curves along the λ sequence. The two vertical dotted lines indicate the selected λ. The vertical dotted line on the left gives the minimum mean cross-validated error, and the right vertical dotted line gives a regularized model such that the error is within one standard error of the minimum.

Table 2.8: Coefficient Estimates of **PCD**.

Variable	RR Estimates with 5-Folds	RR Estimates with 10-Folds	LS Estimates
(Intercept)	-0.0481 (-)	-0.0294 (-)	0.1816 (1.3206)
lcavol	0.4526 (0.0870)	0.4668 (0.0870)	0.5643 (0.0873)
lweight	0.5874 (0.1972)	0.5940 (0.1977)	0.6220 (0.1998)
age	-0.0142 (0.0110)	-0.0151 (0.0110)	-0.0212 (0.0110)
lbph	0.0798 (0.0574)	0.0820 (0.0574)	0.0967 (0.0576)
svi	0.6458 (0.2355)	0.6598 (0.2362)	0.7617 (0.2398)
lcp	-0.0087 (0.0888)	-0.0193 (0.0889)	-0.1061 (0.0894)
gleason	0.0690 (0.1533)	0.0673 (0.1535)	0.0492 (0.1545)
pgg45	0.0030 (0.0043)	0.0032 (0.0043)	0.0045 (0.0043)

2.4.3 BCNKID

We, first, perform a 5-fold cross-validation on this data set. Figure 2.6 displays a range of tuning parameter values, we select $\lambda = 3.8076$ ($\log(\lambda) = 1.3370$), which yeilds the smallest mean cross-validated error. The test MSE for this tuning parameter is 0.0648. Since there are many covariates, $p = 14318$, it is very hard to interpret the results from RR. We will analyze this data set by $lasso$ in Chapter 3, which will give a better explanation.

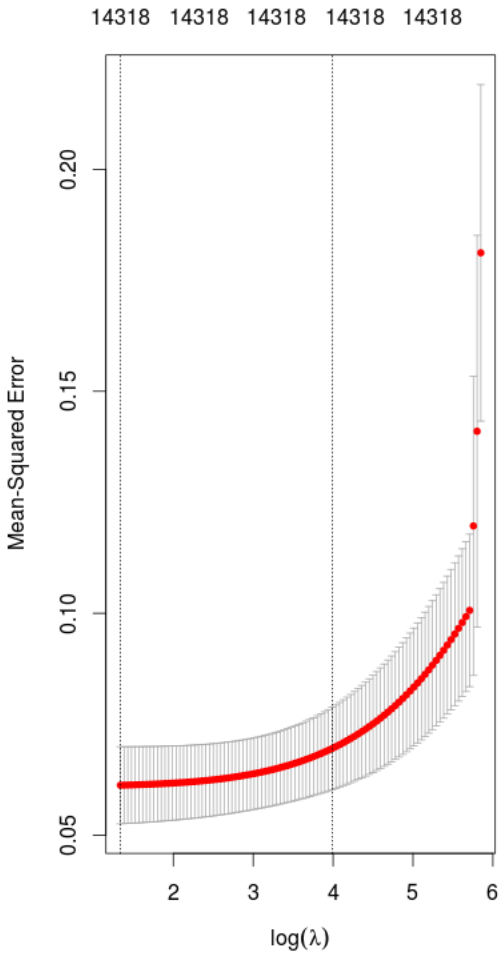

Figure 2.6: 5-Fold Cross-Validation of **BCNKID**. The corss-validation curve is the red-dotted line with upper and lower standard deviation curves along the λ sequence. The two vertical dotted lines indicate the selected λ. The vertical dotted line on the left gives the minimum mean cross-validated error, and the right vertical dotted line gives a regularized model such that the error is within one standard error of the minimum.

A 10-fold cross-validation is also performed on this data set, shown in Figure 2.7. The tuning parameter we select is $\lambda = 17.3483$ $(\log(\lambda) = 2.8535)$, which gives the smallest mean cross-validated error. The test MSE for this λ is 0.0449.

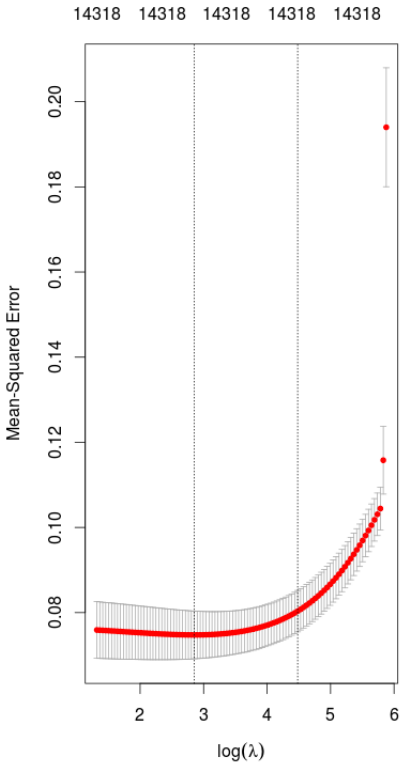

Figure 2.7: 10-Fold Cross-Validation of **BCNKID**. The corss-validation curve is the red-dotted line with upper and lower standard deviation curves along the λ sequence. The two vertical dotted lines indicate the selected λ. The vertical dotted line on the left gives the minimum mean cross-validated error, and the right vertical dotted line gives a regularized model such that the error is within one standard error of the minimum.

2.4.4 GD

Since n is small ($n = 38$) for **GD**, we only perform a 5-fold cross-validation in Figure 2.8. We choose $\lambda = 5.4112$ ($\log(\lambda) = 1.6885$), which corresponds to the smallest mean cross-validated error. The test MSE for this λ is 0.0733.

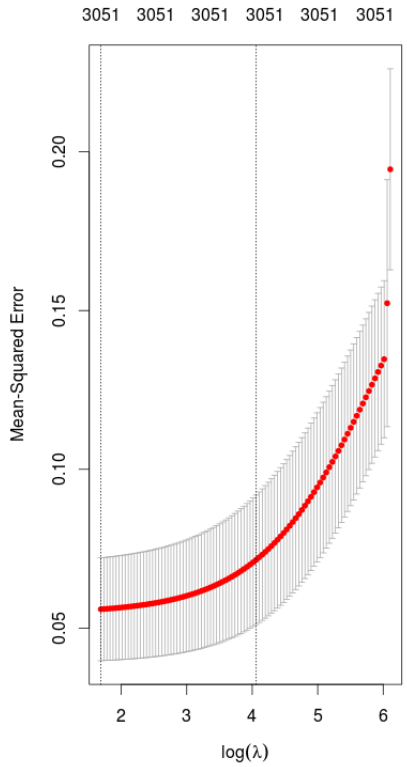

Figure 2.8: 5-Fold Cross-Validation of **GD**. The corss-validation curve is the red-dotted line with upper and lower standard deviation curves along the λ sequence. The two vertical dotted lines indicate the selected λ. The vertical dotted line on the left gives the minimum mean cross-validated error, and the right vertical dotted line gives a regularized model such that the error is within one standard error of the minimum.

Because of the large number of regressors, $p = 3051$, we will leave the results and interpretation for this data set to Chapter 3. The $lasso$ provides a sparse solution, with many coefficients being exactly to zero.

2.5 Factorial Design and Ridge Regression

In this section, we examine some properties of RR for factorial designs. Consider a factorial experiment with k factors and each factor has two levels, coded as $+1$ and -1. A regular fractional factorial experiment with $n = 2^{k-q}$ runs, $q < k$, has $\boldsymbol{X}^T\boldsymbol{X} = n\boldsymbol{I}$, where $\frac{1}{2^q}$ fraction is used in the design. This means that the columns of $\boldsymbol{X}$ are orthogonal. For RR, the results in Section 2.2 can be simplified for fractional factorial designs. Matrix $A(\lambda)$ becomes

$$A(\lambda) = \boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I} = (n+\lambda)\boldsymbol{I}_p.$$

The traces of the mean squared error, the variance, $m_v(\lambda)$, and the bias, $m_b(\lambda)$, are:

$$\begin{aligned}
m_v(\lambda) &= \sigma^2 \sum_{i=1}^{p} \frac{n}{(n+\lambda)^2} = \sigma^2 \frac{np}{(n+\lambda)^2}, \\
m_b(\lambda) &= \lambda^2 \boldsymbol{\beta}^T A(\lambda)^{-2} \boldsymbol{\beta} = \frac{\lambda^2}{(n+\lambda)^2} \boldsymbol{\beta}^T\boldsymbol{\beta}, \\
m(\lambda) &= m_v(\lambda) + m_b(\lambda) \\
&= \frac{np\sigma^2}{(n+\lambda)^2} + \frac{\boldsymbol{\beta}^T\boldsymbol{\beta}\lambda^2}{(n+\lambda)^2} \\
&= \frac{a + b\lambda^2}{(n+\lambda)^2},
\end{aligned}$$

where $a = np\sigma^2$ and $b = \boldsymbol{\beta}^T\boldsymbol{\beta}$.

Differentiating $m(\lambda)$ gives the following results:

$$
\begin{aligned}
(i)\text{If } \lambda < \frac{p\sigma^2}{\boldsymbol{\beta}^T\boldsymbol{\beta}}, \text{ then } \frac{dm(\lambda)}{d\lambda} < 0,\\
(ii) \text{ If } \lambda = \frac{p\sigma^2}{\boldsymbol{\beta}^T\boldsymbol{\beta}}, \text{ then } \frac{dm(\lambda)}{d\lambda} = 0,\\
(iii)\text{If } \lambda > \frac{p\sigma^2}{\boldsymbol{\beta}^T\boldsymbol{\beta}}, \text{ then } \frac{dm(\lambda)}{d\lambda} > 0.
\end{aligned}
$$

Proof of (i),

$$
\begin{aligned}
\text{since } \frac{dm(\lambda)}{d\lambda} &= \frac{2b\lambda(n+\lambda)^2 - 2(n+\lambda)(a+b\lambda^2)}{(n+\lambda)^4}\\
&= \frac{2b\lambda n + 2b\lambda^2 - 2a - 2b\lambda^2}{(n+\lambda)^3}\\
&= \frac{2(b\lambda n - a)}{(n+\lambda)^3}.
\end{aligned}
$$

If $\frac{dm(\lambda)}{d\lambda} < 0$, then $b\lambda n - a < 0$. This gives $\lambda < \frac{a}{bn} = \frac{np\sigma^2}{n\boldsymbol{\beta}^T\boldsymbol{\beta}} = \frac{p\sigma^2}{\boldsymbol{\beta}^T\boldsymbol{\beta}}$, which shows the result of (i). The results for (ii) and (iii) are obvious as well. This indicates that $m(\lambda)$ is minimized at $\lambda = \frac{p\sigma^2}{\boldsymbol{\beta}^T\boldsymbol{\beta}}$.

For regular 2-level fractional factorial design (ffd), some effects are fully aliased. If two effects, say X_r and X_s, are fully aliased and both are included in the linear model, then the LSE doesn't exist. However, the ridge estimate will give the same coefficient estimate for the two effects, i.e., for the following model,

$$y = \beta_0 + \beta_r x_r + \beta_s x_s + ... + \beta_l x_l + \epsilon,$$

we get $\hat{\beta}_r^R(\lambda) = \hat{\beta}_s^R(\lambda)$ for all $\lambda > 0$.

The reason is given in the following. We can write the model matrix as:

$$\boldsymbol{X} = \begin{pmatrix} 1 & x_{1r} & x_{1s} & \cdots & x_{1l} \\ 1 & x_{2r} & x_{2s} & \cdots & x_{2l} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{nr} & x_{ns} & \cdots & x_{nl} \end{pmatrix}_{n \times p}$$

Suppose X_r and X_s are fully aliased, and they are orthogonal to other effects in the model. Then

$$\boldsymbol{X^T X} = \left(\begin{array}{c|c} \begin{smallmatrix} n & 0 & 0 \\ 0 & n & n \\ 0 & n & n \end{smallmatrix} & \boldsymbol{0} \\ \hline \boldsymbol{0} & n\boldsymbol{I}_{p-3} \end{array}\right), \qquad \boldsymbol{X^T Y} = \begin{pmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_{ir} y_i \\ \sum_{i=1}^n x_{is} y_i \\ \vdots \end{pmatrix}$$

$$(\boldsymbol{X^T X} + \lambda \boldsymbol{I})^{-1} = \left(\begin{array}{c|c} \begin{smallmatrix} \frac{1}{n+\lambda} & 0 & 0 \\ 0 & a & b \\ 0 & b & a \end{smallmatrix} & \boldsymbol{0} \\ \hline \boldsymbol{0} & (\frac{1}{n+\lambda})\boldsymbol{I}_{p-3} \end{array}\right), \text{ with } a = \frac{n+\lambda}{2n\lambda+\lambda^2}, \text{ b} = \frac{-n}{2n\lambda+\lambda^2}.$$

$$\text{Denote } \boldsymbol{X^T Y} = \begin{pmatrix} \sum_{i=1}^n y_i \\ c \\ c \\ \vdots \end{pmatrix}, \text{with c} = \sum_{i=1}^n x_{ir} y_i = \sum_{i=1}^n x_{is} y_i.$$

Thus $\hat{\beta}_r^R(\lambda) = \hat{\beta}_s^R(\lambda) = (a+b)c = \frac{\lambda}{2n\lambda+\lambda^2}$ c = $\frac{c}{2n+\lambda}$ for all $\lambda > 0$. In addition, as $\lambda \longrightarrow 0$, we get $\hat{\beta}_r^R(\lambda) = \hat{\beta}_s^R(\lambda) \longrightarrow \frac{c}{2n}$ = $(\frac{1}{2})\tilde{\beta}_r(0)$, where $\tilde{\beta}_r$ is the LSE estimate if $\beta_r x_r$ is the only term included in the model.

If there are m effects that are fully aliased, then

$\hat{\beta}_{r1}(\lambda) = \hat{\beta}_{r2}(\lambda) = ... = \hat{\beta}_{rm}(\lambda) \longrightarrow \frac{1}{m}\tilde{\beta}_r(0)$, as $\lambda \longrightarrow 0$.

The example below shows the property we just proved. Consider a 2^{4-1} design with the defining relation $I = \boldsymbol{ABCD}$, where $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ are the four factors in the experiment. The design is constructed and outlined in Table 2.9.

Table 2.9: 2^{4-1} Design with $I = \boldsymbol{ABCD}$.

Run	$\boldsymbol{A}$	$\boldsymbol{B}$	$\boldsymbol{C}$	$\boldsymbol{D} = \boldsymbol{ABC}$	Treatment Combination
1	-	-	-	-	(1)
2	+	-	-	+	ad
3	-	+	-	+	bd
4	+	+	-	-	ab
5	-	-	+	+	cd
6	+	-	+	-	ac
7	-	+	+	-	bc
8	+	+	+	+	abcd

Notice that here each main effect is aliased with a three-factor interaction and that each two-factor interaction is aliased with another two-factor interaction. See Table 2.10.

Table 2.10: Alias Structure.

Alias Structure
[A]$\longrightarrow A + BCD$
[B]$\longrightarrow B + ACD$
[C]$\longrightarrow C + ABD$
[D]$\longrightarrow D + ABC$
[AB]$\longrightarrow AB + CD$
[AC]$\longrightarrow AC + BD$
[AD]$\longrightarrow AD + BC$

As an example, to get an estimate for the effects, we use the eight observed filtration rates used in Example 6.2 in Montgomery (2012). Table 2.11 gives the filtration rate and the estimates if the model is

$$y = \beta_0 + \beta_1 A + \beta_2 B + \beta_3 C + \beta_4 D + \beta_5 AB + \beta_6 AC + \beta_7 AD + \epsilon.$$

Table 2.11: Filtration Rate and Estimates

Run	Filtration Rate
1	45
2	100
3	45
4	65
5	75
6	60
7	80
8	96

Estimate
$[\boldsymbol{A}] = 19.00$
$[B] = 1.50$
$[\boldsymbol{C}] = 14.00$
$[\boldsymbol{D}] = 16.50$
$[AB] = -1.00$
$[\boldsymbol{AC}] = -18.50$
$[\boldsymbol{AD}] = 19.00$

Now suppose we include both $\boldsymbol{AC}$ and $\boldsymbol{BD}$ in the linear model,

$$y = \beta_0 + \beta_1 A + \beta_2 B + \beta_3 C + \beta_4 D + \beta_5 AB + \underline{\beta_6 AC} + \beta_7 AD + \underline{\beta_8 BD} + \epsilon.$$

Since AC and BD are fully aliased, we cannot get the LSE. But the RRE can be computed for $\lambda > 0,$ and they are given in Table 2.12. It is clear that, as λ increases, all the estimates shrink towards zero. For β_6 and β_8, we have $\hat{\beta}_6^R(\lambda) = \hat{\beta}_8^R(\lambda)$ for all $\lambda > 0$. Note also that here, Table 2.12, we consider only half of the effects, whereas Table 2.11 presents the full effects.

Table 2.12: LSE and RRE Estimates for Various Values of λ.

Estimate	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.5$	$\lambda = 1.5$	$\lambda = 5$	$\lambda = 20$	$\lambda = 200$
$\hat{\beta}_0$	70.75	70.75	70.75	70.75	70.75	70.75	70.75
$\hat{\beta}_1$	9.50	9.38	8.94	8.00	5.85	2.71	0.37
$\hat{\beta}_2$	0.75	0.74	0.71	0.63	0.46	0.21	0.03
$\hat{\beta}_3$	7.00	6.91	6.59	5.89	4.31	2.00	0.27
$\hat{\beta}_4$	8.25	8.15	7.76	6.95	5.08	2.36	0.32
$\hat{\beta}_5$	-0.50	-0.49	-0.47	-0.42	-0.31	-0.14	-0.02
$\hat{\beta}_6$	-9.25	-4.60	-4.48	-4.23	-3.52	-2.06	-0.34
$\hat{\beta}_7$	9.50	9.38	8.94	8.00	5.85	2.71	0.37
$\hat{\beta}_8$	*NA*	-4.60	-4.48	-4.23	-3.52	-2.06	-0.34

Chapter 3

The Lasso

3.1 Introduction

Similar to RR, $lasso$ (least absolute shrinkage and selection operator) deals with many predictors, $p \gg n$, and with an ill-conditioned model matrix $\boldsymbol{X}$. However, it is different from RR, in that the $lasso$ sets many coefficient estimates exactly to zero, making interpretation of the statistical model more plausible. The $lasso$ is introduced by Robert Tibshirani in 1996 and is defined as follows.

Let

$$L(\boldsymbol{\beta}, \lambda) = RSS(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|, \tag{3.1}$$

where $RSS(\boldsymbol{\beta})$ is defined in (1.4) and $\lambda \geq 0$ is the tuning parameter.

The $lasso$ estimator, denoted by $\hat{\boldsymbol{\beta}}^L(\lambda)$, minimizes (3.1) over $\boldsymbol{\beta}$ for a given λ. The tuning parameter λ decides whether $\hat{\boldsymbol{\beta}}^L(\lambda)$ is sparse or not (setting some coefficient estimates excatly to zero). When λ gets larger, $\| \hat{\boldsymbol{\beta}}^L(\lambda) \|_1$ gets smaller, which results in a sparser solution. For a given $\lambda > 0$, $lasso$ estimates may not be

unique, Ryan J. Tibshirani (2012).

As in RR, we also standardize the predictors of $\boldsymbol{X}$ if they have different units. The intercept term can be recovered after standardization and obtaining the $lasso$ coefficient estimates from

$$\hat{\beta}_0 = \bar{y} - \sum_{j=1}^{p} \bar{x}_j \hat{\beta}_j^L(\lambda),$$

where $\bar{y}$ and $\bar{x}_j$, for $j = 1, \cdots, p$, are the original means and $\hat{\beta}_j^L(\lambda)$ is the $lasso$ coefficient estimate for $j = 1, \cdots, p$.

In vector and vector norm representation, we can write the $lasso$ estimator $\hat{\boldsymbol{\beta}}^L(\lambda)$ as the solution to the following problem:

$$\min_{\beta \in \mathbb{R}^p} L(\boldsymbol{\beta}, \lambda) \quad = \min_{\beta \in \mathbb{R}^p} \quad \| \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} \|_2^2 + \lambda \| \boldsymbol{\beta} \|_1 \qquad \text{for some} \quad \lambda \geq 0, \tag{3.2}$$

where $\| \boldsymbol{\beta} \|_1 = \sum_{j=1}^{p} |\beta_j|$ and the matrix $\boldsymbol{X}$ does not include the column of ones after standardization. When $\lambda = 0$, we get LSE in (1.4).

In fact, problem (3.2) is equivalent to

$$\begin{aligned} \min_{\beta \in \mathcal{R}^p} \quad & \| \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} \|^2 \\ \text{subject to} \quad & \| \boldsymbol{\beta} \|_1 \leq r, \end{aligned} \tag{3.3}$$

for certain $r > 0$ (see Tibishirani (1996)), which is related to λ. Figure 3.1 gives a geometric interpretation of the $lasso$ and LSE in 2-dimensional space with $\boldsymbol{\beta} = (\beta_1, \beta_2)^T$. Here, $\hat{\boldsymbol{\beta}}$ represents the LS solution and the contour plots are for $RRS(\boldsymbol{\beta})$ as it moves away from the LS solution. The $lasso$ estimates, $\hat{\boldsymbol{\beta}}^L(\lambda)$, is the point

where the contour of $RRS(\boldsymbol{\beta})$ meets the square defined by $|\beta_1| + |\beta_2| = r$.

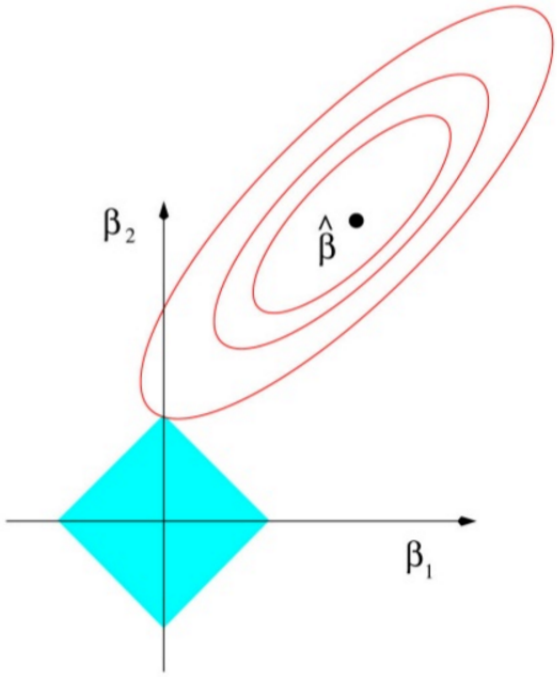

Figure 3.1: Geometric Interpretation of the Lasso for $p = 2$. Here $\hat{\boldsymbol{\beta}}$ refers to the LSE of $\boldsymbol{\beta}$.

3.2 Properties of the Lasso

Since there is no analytical solution to problem (3.1), there is no explicit formulas for the bias and the variance of the $lasso$ estimator, Hastie et al.(2015). However, there are asymptotic results in Hastie et al.(2015). Here we use an example to illustrate some prperties of the bias and variance of the $lasso$. We generate a linear regression model with $n = 50$ and $p = 30$ (20 of the true coefficients are zero). The plots of the bias vs. different values of the tuning parameter λ and the variance vs. different values of the tuning parameter λ are displayed in Figure 3.2, where $m_v(\lambda)$ and $m_b(\lambda)$ are defined the same as in RR.

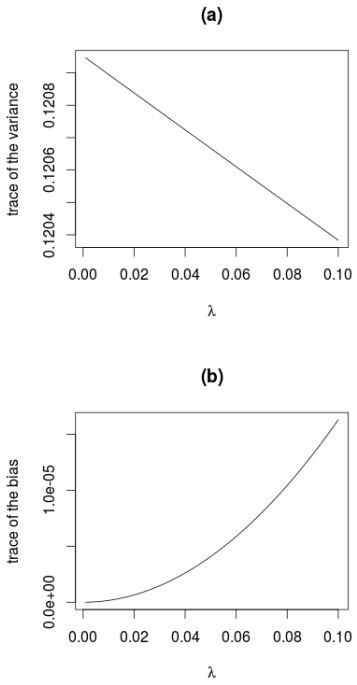

Figure 3.2: The Plots of Variance and Bias: (a) $m_v(\lambda)$ vs. λ, (b) $m_b(\lambda)$ vs. λ.

We can observe the following from the plots:

- The bias increases as λ increases.
- The variance decreases as λ increases.
- The mean squared error (MSE) decreases initially but increases later as λ increases, as seen in Figure 3.3.

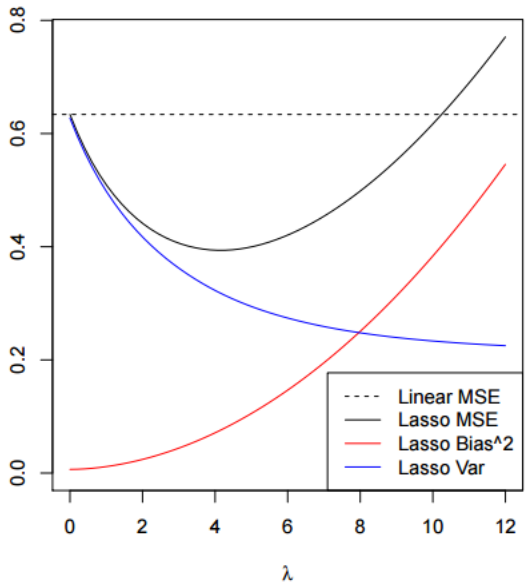

Figure 3.3: The Plots of the Bias, Variance and MSE.

3.3 Bilevel Optimization

Bilevel optimization was first introduced by Stackelberg in 1934 on market economy, von Stackelberg (1952). The bilevel programming problem ($BLPP$) can be written as follows :

$$\begin{aligned} &\min_{x,y} \quad F(x,y) \\ &\text{subject to} \begin{cases} G(x,y) \leq 0, \\ y \in \underset{y}{\text{argmin}} \quad \{f(x,y) : g(x,y) \leq 0\}, \end{cases} \end{aligned} \tag{3.4}$$

where $x \in \mathbb{R}^{n_1}$ and $y \in \mathbb{R}^{n_2}$. The above problem ($BLPP$) is divided into an upper-level (the leader) with variables $x \in \mathbb{R}^{n_1}$ and a lower-level (the follower) with variables $y \in \mathbb{R}^{n_2}$. The functions $F : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}$ and $f : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}$ are the objective functions for the upper-level and lower-level respectively, whereas $G : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}^{m_1}$ and $g : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}^{m_2}$ are the constraints for the

upper-level and lower-level respectively.

3.3.1 The Lasso as a BLPP

The use of bilevel optimization in solving problems such as choosing appropriate tuning parameters, model selection and classification has attracted researchers in recent years. Rosset (2009) used bilevel optimization to follow the path of cross validated solutions of kernel quantile regression and Pedregosa (2016) used it to find an appropriate set of hyperparameters.

Before we formulate the $lasso$ as a $BLPP$, we discuss the advanatges of formulating it as a $BLPP$. Instead of specifying a grid search to find a suitable tuning parameter λ, the bilevel approach identifies one tuning parameter at once via optimization methods. The bilevel approach can handle several goals at once, such as selecting the optimal choice of λ, providing a sparse solution, and performing inexact cross-validation, Kunapuli (2008).

We now formualte the $lasso$ in (3.2) and cross-validation for estimating the error for a given tuning parameter as a $BLPP$ as follows:

$$\begin{aligned} \min_{\lambda,\boldsymbol{\beta}_k} \quad & \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\Omega_{val}^k|}\sum_{j\in\Omega_{val}^k}(\boldsymbol{\beta}_k^T\boldsymbol{x}_j - y_j)^2 \\ \text{subject to} \begin{cases} & \lambda \geq 0, \text{ and for each } k = 1, ..., K, \\ & \boldsymbol{\beta}_k \in \underset{\boldsymbol{\beta}_k}{\text{argmin}} \quad \{\sum_{j\in\Omega_{trn}^k}(\boldsymbol{\beta}_k^T\boldsymbol{x}_j - y_j)^2 + \lambda \parallel \boldsymbol{\beta}_k \parallel_1\}, \end{cases} \end{aligned} \tag{3.5}$$

where $\Omega = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n \subset \mathbb{R}^{p+1}$ is partitioned into a testing set Ω_{val}^k and a training set Ω_{trn}^k, so that $\Omega = \Omega_{trn}^k \cup \Omega_{val}^k$ for each fold k. $|\Omega|$ is the cardinality of the set Ω which is defined by the number of points or elements in the set.

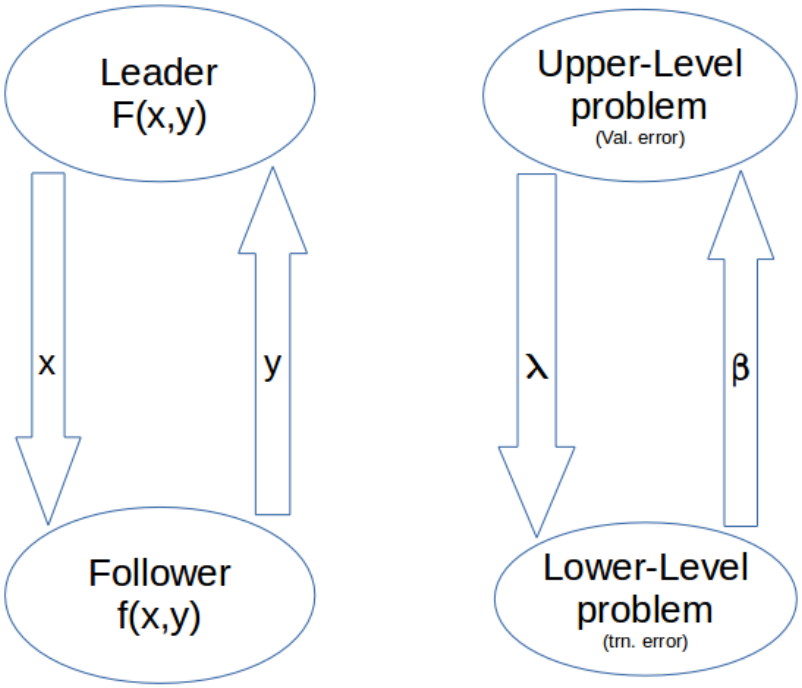

Figure 3.4: Bilevel Plot Displaying the Upper-Level Problem and the Lower-Level Problem.

To solve the $lasso$, we formulate it into a mathematical program with equilibrium constraints ($MPEC$) by replacing the lower level of (3.5) by its KKT conditions. Before we proceed, we consider the equivalent convex quadratic programming model:

$$\begin{aligned} \underset{\boldsymbol{\beta}_k, \boldsymbol{\omega}^{k+}, \boldsymbol{\omega}^{k-}}{min} \quad & \sum_{j \in \Omega^k_{trn}} (\boldsymbol{\beta}_k^T \boldsymbol{x}_j - y_j)^2 + \lambda(\boldsymbol{\omega}^{k+} + \boldsymbol{\omega}^{k-})\} \\ \text{subject to} \quad & \boldsymbol{\omega}^{k+} - \boldsymbol{\omega}^{k-} = \boldsymbol{\beta}_k, \\ & \boldsymbol{\omega}^{k+}, \boldsymbol{\omega}^{k-} \geq \mathbf{0}, \end{aligned} \tag{3.6}$$

where $\boldsymbol{\omega}^{k+}$, $\boldsymbol{\omega}^{k-} \in \mathbb{R}^p$.

Because the problem of (3.6) is a linear convex quadratic problem, the KKT con-

ditions are necessary and sufficient for optimality, Boyd and Vandenberghe (2004). Hence, we can write the $MPEC$ as follows:

$$\min_{\lambda,\boldsymbol{\beta}_k,\boldsymbol{\omega}^{k+},\boldsymbol{\omega}^{k-},\boldsymbol{\mu}^k,\boldsymbol{\xi}^{k+},\boldsymbol{\xi}^{k-}} \quad \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\Omega_{val}^k|}\sum_{j\in\Omega_{val}^k}(\boldsymbol{\beta}_k^T\boldsymbol{x}_j - y_j)^2$$

$$\text{subject to} \quad \begin{cases} \sum_{j\in\Omega_{trn}^k} 2(\boldsymbol{\beta}_k^T\boldsymbol{x}_j - y_j)\boldsymbol{x}_j - \boldsymbol{\mu}^k = \mathbf{0}, \quad k = 1,2,...,K, \\ \lambda\boldsymbol{e} + \boldsymbol{\mu}^k - \boldsymbol{\xi}^{k+} = \mathbf{0}, \quad k = 1,2,...,K, \\ \lambda\boldsymbol{e} - \boldsymbol{\mu}^k - \boldsymbol{\xi}^{k-} = \mathbf{0}, \quad k = 1,2,...,K, \\ \boldsymbol{\omega}^{k+} - \boldsymbol{\omega}^{k-} = \boldsymbol{\beta}_k, \quad k = 1,2,...,K, \\ \boldsymbol{\omega}^{k+} \geq \mathbf{0}, \quad \boldsymbol{\xi}^{k+} \geq \mathbf{0}, \quad \boldsymbol{\omega}^{k+^T}\boldsymbol{\xi}^{k+} = \mathbf{0}, \quad k = 1,2,...,K, \\ \boldsymbol{\omega}^{k-} \geq \mathbf{0}, \quad \boldsymbol{\xi}^{k-} \geq \mathbf{0}, \quad \boldsymbol{\omega}^{k-^T}\boldsymbol{\xi}^{k-} = \mathbf{0}, \quad k = 1,2,...,K, \\ \lambda \geq \mathbf{0}, \end{cases} \tag{3.7}$$

where $\boldsymbol{\mu}^k, \boldsymbol{e}, \boldsymbol{\xi}^{k+}$ and $\boldsymbol{\xi}^{k-} \in \mathbb{R}^p$.

3.3.2 The Algorithm

Globally solving the $BLPP$ in (3.5) is equivalent to globally solving the $MPEC$ problem in (3.7), Luo et al. (1996). There have been many algorithms developed to solve $MPEC$ problems; however, we are considering the Proximal Alternating Direction Method of Multipliers ($PADMM$) algorithm proposed recently by Chen et al. (2017) because it seems potentially suitable for the $lasso$ with big data. The corresponding $PADMM$ algorithm for the $MPEC$ model in (3.7) is as follows.

We first rewrite the model in (3.7) as the following:

$$\begin{aligned}
\min_{\lambda,\boldsymbol{\beta}_k,\boldsymbol{\omega}^{k+},\boldsymbol{\omega}^{k-},\boldsymbol{\mu}^k,\boldsymbol{\xi}^{k+},\boldsymbol{\xi}^{k-}} \quad & \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\Omega_{val}^k|}[\sum_{j\in\Omega_{val}^k}\boldsymbol{\beta}_k^T\boldsymbol{x}_j\boldsymbol{x}_j^T\boldsymbol{\beta}_k - 2\sum_{j\in\Omega_{val}^k}(\boldsymbol{x}_j y_j)^T\boldsymbol{\beta}_k] \\
\text{subject to} \quad & \begin{cases}
2\sum_{j\in\Omega_{trn}^k}(\boldsymbol{x}_j\boldsymbol{x}_j^T)\boldsymbol{\beta}_k - \boldsymbol{\mu}^k = 2\sum_{j\in\Omega_{trn}^k} y_j\boldsymbol{x}_j, \quad k=1,2,...,K, \\
-\boldsymbol{\beta}_k + \boldsymbol{\omega}^{k+} - \boldsymbol{\omega}^{k-} = \mathbf{0}, \quad k=1,2,...,K, \\
\boldsymbol{\mu}^k - \boldsymbol{\xi}^{k+} + \lambda\boldsymbol{e} = \mathbf{0}, \quad k=1,2,...,K, \\
-\boldsymbol{\mu}^k - \boldsymbol{\xi}^{k-} + \lambda\boldsymbol{e} = \mathbf{0}, \quad k=1,2,...,K, \\
(\boldsymbol{\omega}^{k+},\boldsymbol{\xi}^{k+}) \in \mathcal{D}_p, \quad k=1,2,...,K, \\
(\boldsymbol{\omega}^{k-},\boldsymbol{\xi}^{k-}) \in \mathcal{D}_p, \quad k=1,2,...,K, \\
\lambda \geq \mathbf{0},
\end{cases}
\end{aligned} \tag{3.8}$$

where $\mathcal{D}_p = \{(\boldsymbol{\omega},\boldsymbol{\xi}) \in \mathbb{R}^{4p} : \boldsymbol{\omega} \geq \mathbf{0}, \boldsymbol{\xi} \geq \mathbf{0}, \boldsymbol{\omega}^T\boldsymbol{\xi} = \mathbf{0}\}$.

Now we are easily able to define the quadratic program with equilibrium constraints ($QPEC$) model:

$$\begin{aligned}
\min_{w,y,z,l} \quad & \frac{1}{2}w^TQw + c^Tw \\
\text{subject to} \quad & Cw = d, \\
& Ew - \begin{bmatrix} y \\ z \\ l \end{bmatrix} = 0, \\
& (y,z) \in \mathcal{D}, \\
& l \in [l_1, l_2],
\end{aligned} \tag{3.9}$$

where

- $\mathcal{D} = \{(y, z) \in \mathbb{R}^{4Kp} : y \geq 0, z \geq 0, y^T z = 0\}$;

- $w = (\beta_1, ..., \beta_K, \mu^1, ..., \mu^K, \omega^{1+}, ..., \omega^{K+}, \omega^{1-}, ..., \omega^{K-}, \xi^{1+}, ..., \xi^{K+}, \xi^{1-}, ..., \xi^{K-}, \lambda)$, where $\beta_k \in \mathbb{R}^p, \mu^k \in \mathbb{R}^p, \omega^{k+} \in \mathbb{R}^p, \omega^{k-} \in \mathbb{R}^p, \xi^{k+} \in \mathbb{R}^p, \xi^{k-} \in \mathbb{R}^p, \lambda \in \mathbb{R}_+$, hence $w \in \mathbb{R}^{6Kp+1}$;

- $Q = \frac{2}{K}(\overbrace{\boldsymbol{B}^{(1)} \oplus \boldsymbol{B}^{(2)} \oplus \cdots \oplus \boldsymbol{B}^{(K)}}^{Kp} \oplus \mathbf{0}_{Kp})$, where $\boldsymbol{B} = \sum_{j=1}^{n} \boldsymbol{x}_j \boldsymbol{x}_j^T$, $\boldsymbol{B}^{(k)} = \sum_{j \in \Omega_{val}^k} \boldsymbol{x}_j \boldsymbol{x}_j^T$, $\oplus$ is the direct sum, and $\mathbf{0}_{Kp}$ is the zero matrix with dimension Kp.

- $c = \frac{-2}{K}\left(\frac{1}{|\Omega_{val}^1|}\boldsymbol{T}^{(1)}, \cdots, \frac{1}{|\Omega_{val}^K|}\boldsymbol{T}^{(K)}\right) \in \mathbb{R}^{(6Kp+1)\times 1}$, where $\boldsymbol{T}^{(k)} = \sum_{j \in \Omega_{val}^k} y_j \boldsymbol{x}_j$;

- $C =$

$$\left[\begin{array}{c|c|c|c|c|c|c}
\overbrace{\begin{array}{ccc} 2\boldsymbol{G}^{(1)} & & \\ & \ddots & \\ & & 2\boldsymbol{G}^{(K)} \end{array}}^{Kp} & \overbrace{\begin{array}{ccc} -I_p & & \\ & \ddots & \\ & & -I_p \end{array}}^{Kp} & & & & & \\
\hline
\begin{array}{ccc} -I_p & & \\ & \ddots & \\ & & -I_p \end{array} & & I_{Kp} & -I_{Kp} & & & \\
\hline
 & I_{Kp} & & & -I_{Kp} & & \boldsymbol{e}_{Kp} \\
\hline
 & -I_{Kp} & & & & -I_{Kp} & \boldsymbol{e}_{Kp}
\end{array}\right],$$

where $\boldsymbol{G}^{(k)} = \sum_{j \in \Omega_{trn}^k} \boldsymbol{x}_j \boldsymbol{x}_j^T$ and $C \in \mathbb{R}^{4Kp \times (6Kp+1)}$ and the blank blocks represent zero matrices of appropriate size;

- $d = (2\hat{\boldsymbol{T}}^{(1)}, \cdots, 2\hat{\boldsymbol{T}}^{(K)}, 0, \cdots, 0)^T \in \mathbb{R}^{4Kp \times 1}$, where $\hat{\boldsymbol{T}}^{(k)} = \sum_{j \in \Omega_{trn}^k} y_j \boldsymbol{x}_j$;

- $E = \begin{bmatrix} E_{yz} \\ E_l \end{bmatrix} \in \mathbb{R}^{(4Kp+1)\times(6Kp+1)}$, where $E_{yz} = [\mathbf{0}_{4Kp\times 2Kp} \quad I_{4Kp} \quad \mathbf{0}_{4Kp\times 1}] \in \mathbb{R}^{(4Kp)\times(6Kp+1)}$, $E_l = [\mathbf{0}_{1\times 6Kp} \quad \mathbf{1}] \in \mathbb{R}^{1\times(6Kp+1)}$, and

$$\mathbf{0}_{n\times m} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}_{n\times m}.$$

- the interval $[l_1, l_2]$ represents the bounds for the optimal λ.

Now, we define

$$\boldsymbol{\mathcal{L}}(w, y, z, l, \lambda, \mu; \theta_1, \theta_2) = \tfrac{1}{2} w^T Q w + c^T w + \delta_{\mathcal{D}}(y, z) + \delta_{[l_1,l_2]}(l) + \big\langle \lambda_C, Cw - d \big\rangle$$
$$+ \big\langle \lambda_l, E_l w - l \big\rangle + \big\langle \mu, E_{yz} w - \begin{bmatrix} y \\ z \end{bmatrix} \big\rangle + \tfrac{\theta_1}{2} \parallel Cw - d \parallel^2 + \tfrac{\theta_1}{2} \parallel E_l w - l \parallel^2 + \tfrac{\theta_2}{2} \parallel E_{yz} w - \begin{bmatrix} y \\ z \end{bmatrix} \parallel^2$$

where

$$\delta_T(x) = \begin{cases} 0 & \text{if } \mathrm{x} \in T, \\ +\infty & \text{if } \mathrm{x} \notin T, \end{cases}$$

is the indicator function of set T at x; θ_1 and θ_2 are exact penalty parameters. This now will enable us to write the $PADMM$ algorithm.

$PADMM$ Algorithm

Set $\theta_1, \theta_2, \sigma > 0$, and choose inital values $w^0 \in \mathbb{R}^{11Kp+1}, (y^0, z^0) \in \mathcal{D}, l^0 \in [l_1, l_2], \lambda^0, \mu^0$.

for k = 0 , 1, ... **do**

$$
\begin{cases}
w^{k+1} = \underset{w}{\text{argmin}} \quad \boldsymbol{\mathcal{L}}(w, y^k, z^k, l^k, \lambda^k, \mu^k; \theta_1, \theta_2) + \frac{\sigma}{2} \parallel w - w^k \parallel^2 \\
\begin{bmatrix} y^{k+1} \\ z^{k+1} \\ l^{k+1} \end{bmatrix} = \underset{y,z,l}{\text{argmin}} \quad \boldsymbol{\mathcal{L}}(w^{k+1}, y, z, l, \lambda^k, \mu^k; \theta_1, \theta_2), \\
\lambda_C^{k+1} = \lambda_C^k + \theta_1(Cw^{k+1} - d), \\
\lambda_l^{k+1} = \lambda_l^k + \theta_1(E_l w^{k+1} - l^{k+1}), \\
\mu^{k+1} = \mu^k + \theta_2(E_{yz} w^{k+1} - \begin{bmatrix} y^{k+1} \\ z^{k+1} \end{bmatrix}).
\end{cases}
$$

end

For the w sub-problem, we have

$$(Q + \theta_1 C^T C + \theta_1 E_l^T E_l + \theta_2 E_{yz}^T E_{yz} + \sigma I) w^{k+1}$$
$$= \sigma w^k - c - C^T(\lambda_C^k - \theta_1 d) - E_l^T(\lambda_l^k - \theta_1 l^k) - E_{yz}^T(\mu^k - \theta_2 \begin{bmatrix} y^k \\ z^k \end{bmatrix}),$$

where σ is a very small positive number. Similarly, we have

$$\begin{bmatrix} y^{k+1} \\ z^{k+1} \end{bmatrix} = \text{Proj}_{\mathcal{D}}(E_{yz} w^{k+1} + \lambda_{yz}^k / \theta_2),$$
$$l^{k+1} = \text{Proj}_{[l_1, l_2]}(E_l w^{k+1} + \lambda_l^k / \theta_1),$$

for the y, z, l sub-problem, where $\text{Proj}_{\mathcal{D}}(x)$ is the projection of x on the set $\mathcal{D}$. We consider this algorithm is convergent if the relative change is less than a defined tolerance:

$$\textit{relchg} = \max(\| y^{k+1} - y^k \|^2, \| z^{k+1} - z^k \|^2, \| l^{k+1} - l^k \|^2, \| \lambda^{k+1} - \lambda^k \|^2, \| \mu^{k+1} - \mu^k \|^2) < \textit{tol}.$$

3.4 Analysis of the Data Sets Using the Lasso

In this section, we apply the $lasso$ to the data sets described in **Section 2.3** and to a simulated data set. Both **glmnet** package in R and $PADMM$ are used to analyze the appropriate data sets. Package **glmnet** is recommended for the $lasso$ in James et al. (2013). We compare the results obtained from both algorithms for $n > p$. Note that here, **glmnet** performs K-fold cross-validation over a search grid, whereas $PADMM$ performs it inexactly by setting an upper and a lower bound for the tuning parameter λ. The results are presented in the following subsections.

For the losgistic regression model (2.6), the $lasso$ minimizes

$$S_2(\boldsymbol{\beta}, \lambda) = -l(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta|.$$

More developments on the $lasso$ for logistic regression can be found in Meier et al. (2008).

3.4.1 Simulated Data Set

We simulate a random data set with $n = 50$ and $p = 11$ and five coefficients are set to be zero. We use 5-folds and 10-folds cross-validation on this data set. Table 3.1 shows the results obtained from our algorithm, $PADMM$, they closely match the corresponding results obtained from a widely used algorithm, i.e. **glmnet**. There are some differences in some of the coefficient estimates, but, overall, we are able to identify the variables with zero coefficients. A 10-fold is also performed on the same data set, and the results are displayed in Table 3.2.

Table 3.1: Lasso Results of Simulated Data Set with 5-Folds.

Variable	**glmnet**	$PADMM$	True Coefficients
λ	1.3775	0.7667	-
x_1	0.2531	1.2239	0.9
x_2	0	1.8784	0.3
x_3	0.9517	0.2937	2.8
x_4	0	0	0
x_5	0	0	0
x_6	1.8471	2	2
x_7	0	0	0
x_8	1.2740	0.3463	2.1
x_9	0	0.4527	0
x_{10}	1.6188	10	10
x_{11}	0	1.1498	0

Table 3.2: Lasso Results of Simulated Data Set with 10-Folds.

Variable	**glmnet**	$PADMM$	True Coefficients
λ	1.3596	0.4623	-
x_1	0.1906	0.9	0.9
x_2	0	0.6140	0.3
x_3	2.0094	-1.3764	2.8
x_4	0	0	0
x_5	0	-1.5622	0
x_6	1.8848	-1.7322	2
x_7	0	1.2348	0
x_8	1.2822	2.1	2.1
x_9	0	0	0
x_{10}	1.4493	0.0220	10
x_{11}	0	0	0

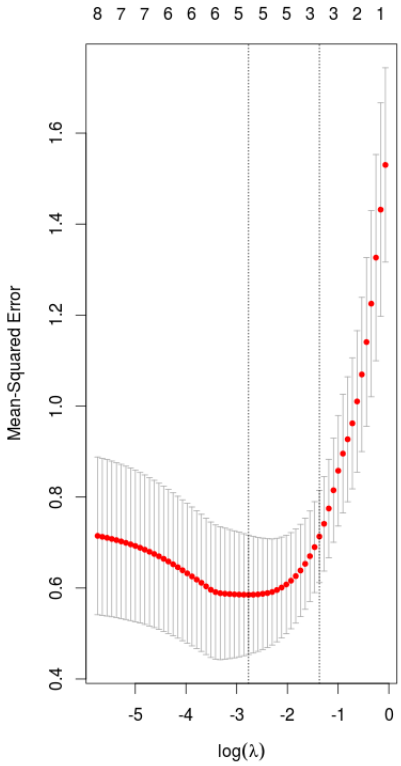

Figure 3.5: 5-Fold Cross-Validation of the **PCD** Data Set. The corss-validation curve is the red-dotted line with upper and lower standard deviation curves along the λ sequence. The two vertical dotted lines indicate the selected λ. The vertical dotted line on the left gives the minimum mean cross-validated error, and the right vertical dotted line gives a regularized model such that the error is within one standard error of the minimum.

3.4.2 PCD

A 5-fold cross-validation is first performed on this data set. Figure 3.5 displays a range of the tuning parameter values associated with their cross-validation errors. We select $\lambda = 0.0629, log(\lambda) = -2.7659,$ with the smallest mean cross-validated error, which implies that only five coefficient estimates are significant. However, if we select ($\lambda = 0.2540$), this would imply that three of the coefficient estimates are significant, as displayed in Figure 3.5.

We also perform a 10-fold cross-validation via **glmnet**. Figure 3.6 gives

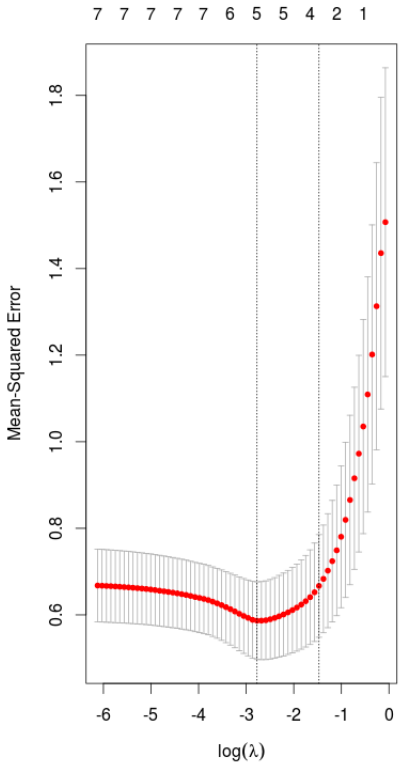

Figure 3.6: 10-Fold Cross-Validation of the **PCD** Data Set. The corss-validation curve is the red-dotted line with upper and lower standard deviation curves along the λ sequence. The two vertical dotted lines indicate the selected λ. The vertical dotted line on the left gives the minimum mean cross-validated error, and the right vertical dotted line gives a regularized model such that the error is within one standard error of the minimum.

$\lambda = 0.0625, \log(\lambda) = -2.7728,$ with the smallest mean cross-validated error. This λ yields only 5 significant coefficient estimates; however, if we select $\lambda = 0.2298, \log(\lambda) = -1.4703,$ with the error is within one standard error of the minimum, there will be only 3 significant coefficient estimates, see Figure 3.6.

Table 3.3 gives the coefficient estimates for 5 and 10-folds from both algorithms, **glmnet** and $PADMM$. Note that here we select λ with the smallest mean cross-validated error.

Table 3.3: **Glmnet** and $PADMM$ Results for **PCD**.

Variables	5-Folds **glmnet**	5-Fold $PADMM$	10-Folds **glmnet**	10-Folds $PADMM$
λ	0.0629	0.9663	0.0625	1
lcavol	0.4950	-1.4748	0.4951	0.5563
lweight	0.4927	0	0.4932	0
age	-0.0004	0	-0.0004	0.2078
lbph	0.0363	1.1849	0.0366	0
svi	0.5549	0	0.5555	0
lcp	0	0	0	0
gleason	0	0.3727	0	-0.2011
pgg45	0.0014	0	0.0014	0

3.4.3 BCNKID

We only apply **glmnet** for this data set because the $PADMM$ algorithm does not work for generalized linear models. Table 3.4 outlines the results obtained if 5 and 10-folds are performed for various values of λ. From Table 3.4, only 28 coefficient estimates are significant if 5-fold is performed with $\lambda = 0.0494$ among 14318 variables. This computation is carried out with the implementation of the smallest mean cross-validated error, λ.

Table 3.4: Results for **BCNKID** from **glmnet**.

	5-Folds	10-Folds	5-Folds	5-Folds
λ	0.0494	0.0382	0.2	0.25
number of significant coef.	28	39	7	5

As λ increases, the number of significant coefficients decreases. When $\lambda = 0.25$ there are only 4 significant coefficients excluding the intercept term and the coefficient estimates are given in Table 3.5.

Table 3.5: Significant Coefficients for **BCNKID** with 5-Folds.

Variable	Coefficient Estimate from $lasso$ with $\lambda = 0.25$
Gene_19	0.0070
Gene_8762	0.0208
Gene_12754	0.0565
Gene_13226	0.1650

Hopefully these genes identified by the $lasso$ are the important ones related to breast cancer.

3.4.4 GD

Similar to the analysis done to **BCNKID**, we only apply the algorithm in the **glmnet** package for the $lasso$. We provide a summary of the results obtained for various values of λ for 5-folds, see Table 3.6.

Table 3.6: Results for **GB** from **glmnet**.

	5-Folds	5-Folds	5-Folds
λ	0.0536	0.2	0.3
number of significant coef.	17	8	3

As λ increases, the number of significant coefficients decreases. When $\lambda = 0.2$, there are only 7 significant coefficients excluding the intercept, and the coefficient estimates are given in Table 3.7.

Table 3.7: Significant Coefficients for **GD** with 5-Folds.

Variable	Coefficient Estimate from $lasso$ with $\lambda = 0.2$
Gene_395	-0.0054
Gene_524	-0.0168
Gene_809	0.0290
Gene_830	0.0808
Gene_1996	-0.0371
Gene_2125	0.0328
Gene_2199	0.0043

Hopefully these genes identified from the $lasso$ are the ones that will help identify the leukemia type (ALL/0 or AML/1).

Chapter 4

Conclusion

We start this thesis by introducing a multiple linear regression model. Multiple linear regression is the basis for the two topics we discuss in great detail. We introduce RR, discuss its properties, and prove the trace of the MSE is a convex function of the tuning parameter λ over a region. We also analyze some real data sets using RR and display some of the properties discussed in the results we obtained. In the field of fractional factorial designs, LSE cannot estimate aliased affects; however, we are able to estimate aliased effects via RR in Section 2.5. A proof for a general case, m fully aliased effects, is also provided.

In Chapter 3, we introduce the $lasso$ and discuss some of its properties. Because we can write the $lasso$ problem as a bilevel problem, we develop an algorithm to solve the $lasso$ bilevel problem. Formulating the bilevel problem into $MPEC$ and $QPEC$, we were able to develop an algorithm called $PADMM$. We applied this algorithm to two data sets, a simulated data set and a real data set. The results obtained are comparable to those from an algorithm in R, from **glmnet** package. However, we are not able to apply $PADMM$ to other data sets because

the two applications given using RR and the $lasso$ are for logistic regression. We show how powerful the $lasso$ is in obtaining a sparse solution, hence, making interpretation and variable selection a lot easier.

We recommend the usage of RR when dealing with an ill-conditioned model matrix $\boldsymbol{X}$ and if p is not very large. We also recommend it for variable selection because RR is able to shrink coefficient estimates towards zero by constraining the sum of the estimates squared. For fully aliased effects in fractional factorial designs, we were able to prove that RR can actually provide coefficient estimates for fully aliased effects.

We suggest the $lasso$ when dealing with an ill-conditioned model matrix $\boldsymbol{X}$, for $p > n$. When $p > n$, the $lasso$ provides a better variable selection method than RR. The $lasso$ provides a sparse solution by penalizing the sum of the absolute values of the estimates. As λ increases, the number of significant coefficients decreases. Hence, this makes the $lasso$ for variable selection and intrepretation of the results a more plausible method than RR.

Through the analysis of the data sets using the $lasso$, we have noted that we are not able to obtain results for the case $p > n$ using $PADMM$. We think that there is some improvement needed in the $PADMM$ algorithm in order to successfully analyze data sets with $p > n$.

References

Bard, J. (1998). *Practical Bilevel Optimization: Algorithms and Applications*. Kluwer Academic Publishers, The Netherlands.

Boyd, S., and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, New York, USA.

Chen, C., Yuan, X., Zeng, S., and Zhang, J. (2017). Splitting methods based on patrial penalty for mathematical program with equilibrium constraints. *preprint*.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, *33*(1), 1-22.

Galton, F. (1889). *Natural Inheritance*. Macmillan And Co., London, U.K.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. Taylor Francis Group, Forida, USA.

Hoerl, A. E., and Kennard, R. W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, *12*(1), 55-67.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning: With Applications in R*. Springer Publishing Company, Incorporated, New York.

Kohavi, R. (1995). A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection. In *Proceedings of the 14th international joint conference on artificial intelligence* (Vol. 2, p. 1137-1143). Morgan Kaufmann Publishers Inc.

Kunapuli, G. (2008). *A Bilevel Optimization Approach to Machine Learning* (Unpub-

lished doctoral dissertation). Rensselaer Polytechnic Institute, New York, USA.

Le Cessie, S., and van Houwelingen, J. C. (1992). Ridge Estimators in Logistic Regression. *Applied Statistics*, *41*(1), 191-201.

Luo, Z.Q., Pang, J.S., and Ralph, D. (1996). *Mathematical Programs with Equilibrium Constraints*. Cambridge University Press, U. K.

Marquardt, D. W., and Snee, R. D. (1975). Ridge Regression in Practice. *The American Statistician*, *29*(1), 3-20.

Meier, L., van de Geer, S., and Bühlmann, P. (2008). The Group Lasso for Logistic Regression. *Journal of the Royal Statistical Society, Series B*, *70*(1), 53-71.

Montgomery, D. (2012). *Design and Analysis of Experiments, 8th Edition*. John Wiley & Sons, Inc., New Jersey, USA.

Montgomery, D.C., Peck, E.A., and Vining, G.G. (2013). *Introduction to Linear Regression Analysis*. John Wiley & Sons, Inc., New Jersey, USA.

Office of Institutional Research at the University of New Mexico. (1990). *Enrollment Forecast*. `http:https://www3.nd.edu/~busiforc/handouts/Data%20and%20Stories/regression/enrollment%20forecast/enrollment.html`.

Pedregosa, F. (2016). Hyperparameter Optimization with Approximate Gradient. In *International conference on machine learning* (Vol. 48, p. 737-746).

Pollard, K. S., Gilbert, H. N., Ge, Y., Taylor, S., and Dudoit, S. (2017). *Resampling-Based Multiple Hypothesis Testing*. `https://www.bioconductor.org/packages/devel/bioc/manuals/multtest/man/multtest.pdf`.

Rosset, S. (2009). Bi-Level Path Following for Cross Validated Solution of Kernel Quantile Regression. *Journal of Machine Learning Research*, *10*, 2473-2505.

Schroeder, M., Haibe Kains, B., Culhane, A., Sotiriou, C., Bontempi, G., and Quack-

enbush, J. (2017). *Genexpression Dataset Published by van't Veer et al. [2002] and van de Vijver et al. [2002] (NKI)*. `https://www.bioconductor.org/packages/devel/data/experiment/manuals/breastCancerNKI/man/breastCancerNKI.pdf`.

Stamey, T.A., Kabalin, J.N., McNeal, J.E., Johnstone, I.M., Freiha, F., Redwine, E.A., and Yang, N. (1989). Prostate Specific Antigen in the Diagnosis and Treatment of Adenocarcinoma of the Prostate: II. Radical Prostatectomy Treated Patients. *Journal of Urology*, *141*(5), 1076-1083.

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, *58*(1), 267-288.

Tibshirani, Ryan J. (2012). The Lasso Problem and Uniqueness. *Electronic Journal of Statistics*, *7*(0), 1456-1490.

van Wieringen, W.N. (2015). Lecture Notes on Ridge Regression. *ArXiv e-prints*.

von Stackelberg, H. (1952). *The Theory of the Market Economy*. Oxford University Press, London, U.K.