Regression Models and Feature Selection for High-Dimensional Genomics Data

Master Thesis

Regitze Kuhr Skals

AALBORG UNIVERSITY
DENMARK

Department of Mathematical Sciences

Frontpage illustrations are from http://en.wikipedia.org/wiki/DNA_methylation#/media/File:DNA_methylation.jpg and https://parc.pop.upenn.edu/about-population-aging-research-center-parc

AALBORG UNIVERSITY
DEPARTMENT OF MATHEMATICAL SCIENCES
FREDRIK BAJERS VEJ 7G, 9220 AALBORG

TITLE:

Regression Models and Feature Selection for High-Dimensional Genomics Data

PROJECT PERIOD:

From: September 3, 2014

To: June 10, 2015

AUTHOR:

Regitze Kuhr Skals

SUPERVISOR:

Torben Tvedebrink

CIRCULATION: 6

NUMBER of PAGES: 89

ABSTRACT:

DNA-methylation is a process that happens in connection with gene expression. This process has shown to be a promising predictor of age. The relation is interesting in the field of forensic science. If the age of a suspect could be predicted on the basis of DNA, a group of suspects could be narrowed down or it could form a lead for the police, if they had no other leads.

In this thesis regression models usable for handling high dimensional genomics data of DNA-methylation has been studied. The purpose was to find few good predictors of age among hundreds of thousands, and to determine consistency of those.

The methods which were studied for the purpose were Ridge regression, Elastic net and Lasso. Especially Elastic net and Lasso were relevant methods, as they performed variable selection. The consistency of predictors was determined for the Lasso and Elastic net method by Stability selection. Moreover Partial least squares was applied to the data.

The final result was a Ridge regression model found by Elastic net combined with Stability selection. It contained 18 stable predictors, and resulted in an RMSE at 2.43 on the validation data.

Preface

This master thesis was written during the third and fourth semester of the Master of Science in Mathematics at Aalborg University. The title of the project is "Regression Models and Feature Selection for High-Dimensional Genomics Data".

The target group of the thesis is fellow students and others with an interest in statistics and forensic genetics.

Citations will be specified by the Harvard Method, so citations appear with [last name, year] e.g. [Bartholomew et al., 2012]. In the bibliography the books and articles will be specified by author, title, edition, year, and publishing. Tables and figures are numbered referring to the chapter in question, which means that the first figure in Chapter 3 has the number 3.1, the next 3.2 etc. Abbreviations of chosen terms can occur in the thesis. These are defined in parentheses when first mentioned. After the term has been introduced the first time, the abbreviation is used. Vectors will be column vectors, unless otherwise stated, they will together with matrices be denoted by bold letters or symbols, and matrices will furthermore be denoted by capital letters or symbols.

A special thanks to my supervisor assistant professor Torben Tvedebrink for being extremely dedicated and helpful during the project. I would also like to thank Section of Forensic Genetics University of Copenhagen and Head of Department Niels Morling for the thesis of the project and for providing me with data.

Regitze Kuhr Skals

Resumé

I dette speciale er der arbejdet med regressions modeller anvendelige til genomisk data af høj dimension. Mere specifikt er der arbejdet på at finde en model, der kan forudsige en persons alder ud fra dennes DNA. Problemstillingen er særligt interessant indenfor retsgenetik, idet at det vil være en fordel i forbindelse med efterforskningen af kriminalsager at kunne forudsige alderen på en mistænkt ud fra et DNA spor. Man vil så kunne reducere antallet af mistænkte. Derudover vil en bestemt alder på en person være et godt spor at gå ud fra, hvis politiet ikke har andre spor af gerningsmanden.

Det er en bestemt process i forbindelse med gen-ekspressionen kaldet DNA metylering, som har vist en sammenhæng med alder. Denne process kan måles ved hjælp af en teknologi fra Illumina. 485.000 forskellige steder på DNA'et i forskellige gener kaldet markører, kan med denne teknologi måles for raten af DNA-metylering ved hjælp af et micro array. I dette speciale er DNA-metylering blevet målt i 50 blodprøver fra 45 personer i alderen $15-82$ år. Formålet var så at finde nogle få markører ud af de 485.000 målte markører, der bedst kunne forklare en persons alder. Dette var ønskeligt, da det ikke er muligt at indsamle nok DNA fra et gerningssted til at kunne analysere mere end omkring $12-24$ markører. Udover at finde nogle gode markører til at forudsige alderen, var det også vigtigt at disse markører ville kunne bruges gentagne gange til at prædiktere alderen på en mistænkt. Det skulle derfor også være nogle stabile markører, der ville kunne bruges på enhver persons DNA.

Det, at der var 50 observationer til rådighed og flere hundrede tusinde variable, der skulle undersøges for hver observation, klassificerede data som værende høj dimensionelt. Det er ikke muligt at anvende standard regressions metoder på høj dimensionelt data, da disse metoder kræver, at data indeholder flere observationer end variable. Metoderne der er blevet anvendt i forbindelse med at løse problemstillingen, er derfor shrinkage-metoderne Ridge regression, Elastic net og Lasso. Metoderne er baseret på least squares metoden, hvor der er tilføjet en straf parameter. De egner sig derfor til høj dimensionelt data, idet at denne straf parameter tillægger modellen en smule bias, og dermed opnås et bias-variance trade-off for modellen, som medfører, at det er muligt at anvende flere variable end observationer i modellen. Derudover er dimensionsreduktions metoden Partial least squares også blevet anvendt. Denne metode laver et reduceret antal af nye variable ud fra lineære kombinationer af de oprindelige variable, og på denne måde kan de reducerede nye variable benyttes i en standard lineær model.

Fordelen ved metoderne Elastisk net og Lasso var, at disse også udførte selektion af variable. Med disse metoder var det muligt at få udvalgt et udsnit af de 485.000 markører. For at finde stabile markører i blandt de udvalgte, blev metoderne kombineret med Stability selection.

Efter at have anvendt de forskellige metoder på data, blev en Ridge regression model med 18 markører fundet som den bedste ud fra RMSE til at forudsige alderen ud fra metyleret DNA i blodet. Modellen var et resultat af at anvende Elastisk net kombineret med Stability selection.

Contents

1 Introduction

In the field of forensic science, association between human aging and DNA has been studied, since it would be helpful in the investigation of a criminal offence to be able to predict the age of a suspect on the basis of DNA evidence. By this opportunity, the amount of suspects could be narrowed down, when no match with evidence DNA is available. Moreover it would be a great lead for the police to know an approximate age of the perpetrator, if no other leads are available. In the field of epigenetics especially DNA methylation has attracted much attention, as several studies have shown that DNA methylation changes with age [Fraga and Esteller, 2007, Florath et al., 2014, Yi et al., 2014, Hannum et al., 2013], and thereby it might be a promising predictor of age.

DNA-methylation as a predictor of age is not only relevant in forensic science, the relation has also shown interest for age-related diseases such as diabetes mellitus (type 2), cancer and cardiovascular disease [Florath et al., 2014].

In this thesis methylated DNA is measured in 50 blood samples from subjects in the age 15 to 82, and association between methylated DNA in the blood and age is studied. The methylated DNA is measured by the Infinium Human Methylation450 BeadChip Kit from Illumina, a technology which is capable of measuring 485.000 different positions on sequences of DNA per sample. The biology of DNA methylation and the technique for measuring it is explained in Chapter 2, followed by a description of the preprocessing and a first analysis of the data in Chapter 3.

As not much DNA evidence is available from a scene of crime, the purpose of this thesis is to determine the 485.000 positions available from the blood samples, and find a combination of $12 - 24$ of those, that gives the best prediction of a humans age, on the basis of the level of methylated DNA in the blood. The purpose is not only to build a model with a good prediction performance, but also to find some reliable predictors that can be used to consistently predict the age of suspects.

The data to determine in this thesis is genomic data measured by a micro array, which is generally high dimensional data, as we are able to measure several thousand predictors and typically only have few observations available. Because of the high dimension, regular methods for fitting the data will not be applicable since for these methods there must be more observations than predictors available. Issues like overfitting and collinearity in the data, explained in Chapter 4, will typically be a problem. One way to deal with such problems in the regression setting is by shrinkage models like Ridge regression, Elastic net and Lasso. They add a regularization term to the standard linear regression method which makes it possible to fit data with more predictors than observations. An advantage of Elastic net and Lasso is that they perform variable selection, and they are therefore relevant methods for the purpose of reducing the 485.000 measured predictors to an amount between 12 and 24. The methods are explained in Chapter 5.

As the purpose not only is to find some predictive variables, but also to ensure to some

extent reliability of these variables, the method of Stability selection will be combined with Elastic net and Lasso, to investigate the randomness of their selected variables. This method will also be explained in Chapter 5.

Another type of methods useful for handling data of high dimension, is dimension reduction methods. As the name suggests the procedure reduces the dimension of a problem, by making a reduced number of new predictors, where each one is based on linear combinations of all of the original predictors. A method of this type which will be applied in this thesis, is the Partial least squares method, it will be explained in Chapter 6.

Results of applying the methods will be presented in Chapter 7, where performance of some simulated data of methylation levels for different ages is also tested. The performance of data for a subject at the age 18 is especially interesting, as their are different rules for sentences of persons above or below this age. A model with more precision around this age will hence be preferred.

Lastly the found results will be discussed in Chapter 8.

2 Biology and Study of DNA Methylation

DNA methylation is a process that happens in connection with gene expression. In the following biology of and a method for studying DNA methylation will be described.

2.1 DNA and RNA

DNA (deoxiribonucleic acid) and RNA (ribonucleic acid) are to classes of nucleic acids, they store and process information inside cells. DNA determines characteristics such as eye color, hair color and blood type. It makes the encoding of information for proteins, and in this way DNA directs the protein synthesis, and controls shape and physical characteristics of our bodies. RNA uses the information from DNA to build proteins.

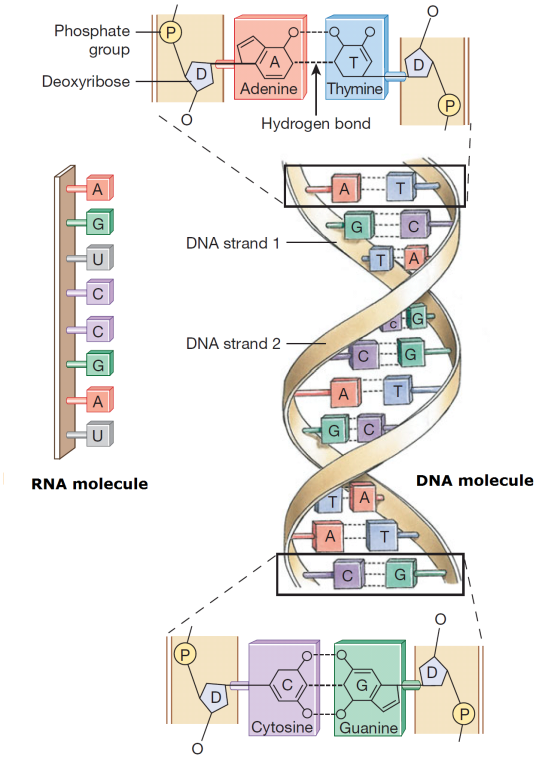


Figure 2.1: RNA and DNA molecules, modified from [Bartholomew et al., 2012].

A nucleic acid is one or two long chains, and the subunits of these chains are called nucleotides. RNA only consists of one chain of nucleotides, whereas DNA consists of two (a pair), see Figure 2.1. Each nucleotide consists of three components. It contains a five-carbon sugar, either ribose (in RNA) or deoxyribose (in DNA), which is attached to a phosphate group and a nitrogenous base. The nitrogenous bases that occur in nucleic acids are adenine (A), guanine (G), cytosine (C), thymine (T) and uracil (U). Thymine occurs only in DNA and Uracil only in RNA, [Bartholomew et al., 2012].

2.2 Gene Expression

The units that contain DNA are genes, they contain the DNA needed to produce particular proteins. The process of building a protein, starting with a gene, is called gene expression. This process can be divided into two subprocesses, transcription and translation. The main function of these processes is that DNA is transcribed into messenger RNA (mRNA), so that it can leave the nucleus and be translated into proteins. In the process of transcription, elements such as the enzyme RNA polymerase and transcription

factors play an important role. General transcription factors are proteins required in the transcription process. They bind to both RNA polymerase and certain DNA sequences in the promoter region of a gene [1]. A sequence called enhancers can help to increase the process of transcription of specific genes. The enhancers bind to special transcription factors called activators which bind to another special class of transcription factors called co-activators, and they finally bind to the general transcription factors which binds to the gene [Jorde, 2006].

The expression of genes may be tissue specific. That is, almost all cells contain the exact same sequence of DNA, and hence only some of the genes in a cell are transcribed depending on the type of tissue, and it happens at specific points in time. To prevent genes from being expressed in a wrong tissue, silencing of these genes is necessary. Silencers are DNA sequences that helps to repress the transcription of genes [Jorde, 2006]. This can happen by DNA methylation

2.3 DNA Methylation

DNA methylation is silencing of genes by modification of DNA. It happens at CpG-sites of the nitrogenous bases. The notation is a shorthand for C - phosphate - G, that is cytosine is in front of a guanine in the DNA sequence, and they are connected by this phosphate group only. The notation is to avoid confusion with the CG base pairing. A methyl group is in this process added to the fifth carbon atom of the cytosine, thus it becomes 5-methylcytosine (methylated C), see Figure 2.2. This transferring of a methyl group to the cytosine, is done by the enzymes DNA methyltransferases. The effect of the methylated C's, is that they block the binding of activators to the enhancers, and thus the gene cannot be transcribed, as illustrated in Figure 2.2. Methylation is an epigenetic phenomena, which means that gene expression is altered without changing the nucleotide sequence [Ginder and Singal, 1999].

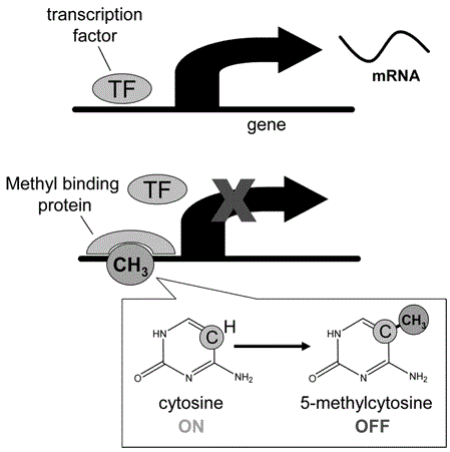


Figure 2.2: The effect of DNA methylation on gene expression [Zeisel, 2007].

[1] a nucleotide sequence laying in the upper part of a gene

CpG-islands

DNA methylation is in particular found in regions called CpG-islands. These regions are DNA-sequences consisting of several CpG-sites next to each other. They are often found in the promoter region of a gene. CpG-islands can be divided into two classes, depending on whether the frequency of CpG-sites is low or high. CpG-islands with low frequency of CpG-sites are typically seen in genes that are tissue-specific, and are therefore often methylated. If the frequency of CpG-sites is high, CpG-islands are generally unmethylated [Armstrong, 2014].

2.4 Illumina 450k Methylation Array

To determine the methylation pattern of DNA, it can be treated with bisulfite. By this process unmethylated cytosines are converted into uracil (U) and the methylated cytosines stay unchanged [Ginder and Singal, 1999]. After this the DNA is whole genome amplified, and uracil (U) has turned into thymine (T), while the methylated cytosines become cytosines (C) again. The conversions make an analysis of the single nucleotide polymorphisms (SNP's) possible in the search for T's and C's. This analysis can be done by micro arrays. There are different types of micro arrays, they can be used to determine the expression of genes or like in this case, to determine variation in genomic DNA by analyzing SNP's.

A micro array is a solid surface with probes attached to it. A probe is a dissolution of a smaller DNA- or RNA-strand, that is supposed to capture a specific target. The targets are complementary DNA - or RNA strands labeled with reporter molecules, typically fluorescent dyes, which makes it possible to measure the frequency of targets captured by the probes. The process where a target binds to a probe is called hybridization. Weak hybridizations are subsequently washed away, and the array is ready to be scanned. When a target labeled with a fluorescent dye is captured by a probe, it emits light which is measured during the scan [Welle, 2013].

One type of micro arrays for the study of DNA methylation is called Infinium HumanMethylation450 BeadChip Kit. It allows more than 485,000 methylation sites per sample, which should be understood as 485,000 different probes. These methylation sites or probes will further on be referred to as CpG-positions or CpG-markers. Two chemistry technologies are used in this array, these are called infinium I - and infinium II assays. Infinium I allows 135,000 probes whereas infinium II allows 350,000.

Infinium I Assay

This assay uses two different probes, one to detect the 'methylated' CpG-sites and one for detection of the unmethylated CpG-sites. As in Figure 2.3 (A), the target for the 'methylated' probe is the nucleotide polymorphism C and for the 'unmethylated' probe T. The signals in both probes emitted from the fluorescent dyes are generated in the same color channel [Dedeuwaerder et al., 2011].

Infinium II Assay

Only one probe detects if the CpG-sites are 'methylated' or not, see Figure 2.3 (B). The targets are either the nucleotide polymorphisms A or G, since they have the complementary bases T and C respectively. Methylated and unmethylated signals are generated in green and red color channels respectively.

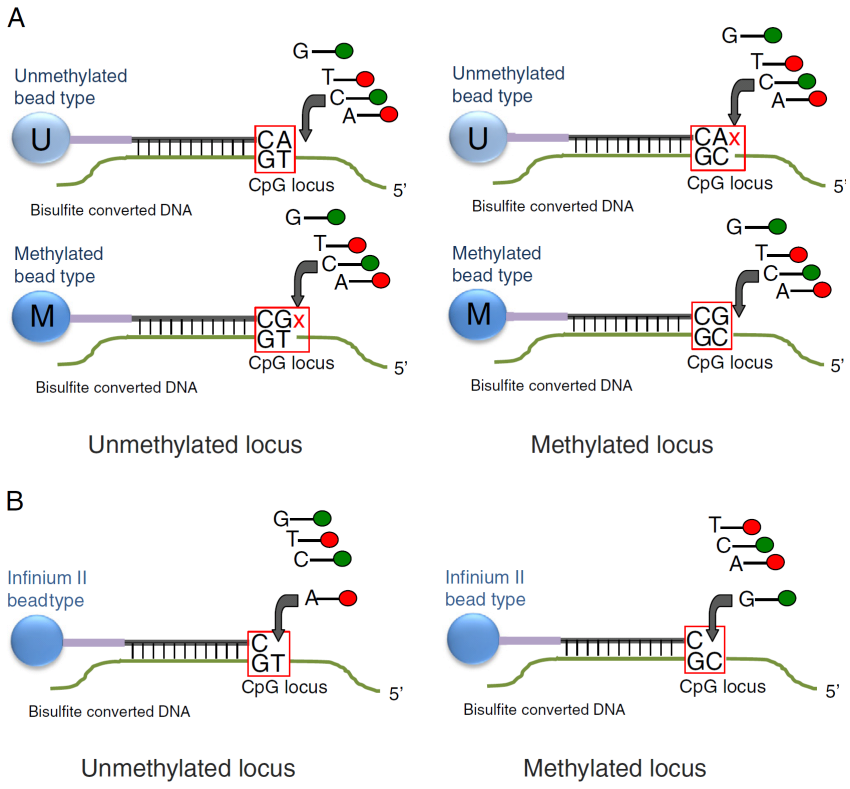


Figure 2.3: Infinium I assay (A) and Infinium II assay (B) [Bibikova et al., 2011].

The methylation can be measured as

$$\beta = \frac{\text{Methylated CpG-sites}}{\text{Unmethylated CpG-sites} + \text{Methylated CpG-sites} + \text{offset}}, \tag{2.1}$$

where the offset is an arbitrary value. It is added to avoid division with small values. In stead of using β as the measure for the level of methylation, it is rather preferred to use M-values. They are computed as logit(β) [Hansen and Aryee, 2013], the definition is

$$M = \text{logit}(\beta) = \log\left(\frac{\beta}{1-\beta}\right) = \log\left(\frac{\text{Methylated CpG-sites}}{\text{Unmethylated CpG-sites} + \text{offset}}\right). \tag{2.2}$$

The data is then closer at fulfilling the assumption of normality, since $M \in (-\infty, \infty)$. In this thesis offset = 0, and a threshold is set for β, to avoid β to be equal to 0 and 1. The threshold is set to 0.001, and hence β will be in the range $[0.001, 0.999]$.

3 Preliminary Data Analysis

For analyzing Illumina's 450k Methylation arrays, the package `minfi` can be used, together with the software `R` [Hansen and Aryee, 2013].

Section of Forensic Genetics, University of Copenhagen, has provided data in the form of 50 blood samples from 45 subjects in the age range 15 – 82 years, random across gender. In the following structure, preprocessing and a first analysis of the data will be described.

3.1 Structure of The Data

When receiving data generated by Illumina's Infinium HumanMethylation450 BeadChip Kit, relevant files for analysis by the `minfi` package are the IDAT files. These represent red and green color channels before normalization.

The overall dataset is called a plate, it contains at most 8 slides, which again contains 12 arrays per slide. The arrays are subdivided into a 6 by 2 grid which are further divided into red and green color channels. A single array contains only one sample, and it measures approximately 450,000 CpG-positions [Hansen and Aryee, 2013], in this particular data set 485,512 CpG-positions are measured. The content of the plate is shown below, arranged into the eight slides.

```
> baseDir <- file.path("D:", "UNI", "Speciale", "Metylering", "Raw_data")

> list.files(baseDir)
[1] "8918692116"      "8918692117"      "8918692128"      "9297949068"
[5] "9297949069"      "9297949106"      "9297949109"      "9297949127"
[9] "A2003_01-96.csv"
```

The csv-file also contained in this overall filepath is called the samplesheet. This file makes the data more readable, and contains information such as array, slide, sample id and for this purpose also age of the subjects.

If we look inside slide 8918692116 we get 12 arrays, each with a green (Grn) and a red (Red) IDAT file .

```
> list.files(file.path(baseDir, "8918692116"), pattern=".idat")
 [1] "8918692116_R01C01_Grn.idat" "8918692116_R01C01_Red.idat"
 [3] "8918692116_R01C02_Grn.idat" "8918692116_R01C02_Red.idat"
 [5] "8918692116_R02C01_Grn.idat" "8918692116_R02C01_Red.idat"
 [7] "8918692116_R02C02_Grn.idat" "8918692116_R02C02_Red.idat"
 [9] "8918692116_R03C01_Grn.idat" "8918692116_R03C01_Red.idat"
[11] "8918692116_R03C02_Grn.idat" "8918692116_R03C02_Red.idat"
[13] "8918692116_R04C01_Grn.idat" "8918692116_R04C01_Red.idat"
[15] "8918692116_R04C02_Grn.idat" "8918692116_R04C02_Red.idat"
[17] "8918692116_R05C01_Grn.idat" "8918692116_R05C01_Red.idat"
[19] "8918692116_R05C02_Grn.idat" "8918692116_R05C02_Red.idat"
[21] "8918692116_R06C01_Grn.idat" "8918692116_R06C01_Red.idat"
[23] "8918692116_R06C02_Grn.idat" "8918692116_R06C02_Red.idat"
```

3.2 Preprocessing

In order to process the data, all the IDAT files are brought together with the samplesheet to an `RGChannelSet` that contains the raw unprocessed data. It contains measures of red and green channels from the different CpG-positions for each sample, which we wish to convert to methylation levels of the CpG-positions. The raw data needs to be normalized before any analysis can take place, since it is important that differences in intensities are in fact due to difference in level of methylation and not due to experimental artefacts [Dudoit and Yang, 2003, Ch. 3]. A function called `preprocessIllumina` in the `minfi` package is able to make this background normalization besides from converting the signals. It takes an `RGChannelSet` as input and returns a `MethylSet` denoted `Mset.norm` below, which is the data converted from green and red channels to methylated and unmethylated signal respectively [Hansen and Aryee, 2013].

It is now possible to observe the number of methylated and unmethylated CpG-sites at individual CpG-positions for each sample. The following shows the number of methylated CpG-sites at five of the 485,512 CpG-positions for three of the 50 samples.

```
> getMeth(Mset.norm)[1:5,1:3]
           9297949127_R03C01 9297949068_R05C02 8918692128_R05C01
cg00050873         933.05278        10053.66467         5916.90935
cg00212031          47.06794          119.45793          156.45378
cg00213748          53.98970         2202.50557          242.68814
cg00214611          22.14962           45.86331           70.21941
cg00455876         184.11872         3381.08604         1294.74739
```

In the same way the number of unmethylated CpG-sites can be observed, and β (as defined in (2.1)) can be computed for every CpG-position, where offset=0. The values of β can be obtained by the following function

```
> getBeta(Mset.norm,offset=0)[1:5,1:3]
           9297949127_R03C01 9297949068_R05C02 8918692128_R05C01
cg00050873         0.5145038        0.889664936         0.70924395
cg00212031         0.2500000        0.025151583         0.15894869
cg00213748         0.2867647        0.952051637         0.64590164
cg00214611         0.2807018        0.008477918         0.08662614
cg00455876         0.3333333        0.839735099         0.50071463
```

Furthermore the M-values (defined in (2.2)) can be obtained by

```
> getM(Mset.norm, type = "", betaThreshold = 0.001)[1:5,1:3]
           9297949127_R03C01 9297949068_R05C02 8918692128_R05C01
cg00050873        0.08372183          3.011371        1.286472697
cg00212031       -1.58496250         -5.276457       -2.403632736
cg00213748       -1.31451062          4.311486        0.867164317
cg00214611       -1.35755200         -6.869791       -3.398331167
cg00455876       -1.00000000          2.389476        0.004123952
```

where `type=""` indicates that the values are computed without any offset, and `betaThreshold=0.001` is the threshold mentioned in Section 2.4, to avoid values of β at 0 and 1.

3.3 Differentially Methylated Positions

The derived `Methylset` is applicable for analysis of the individual CpG-positions. We are now capable of examining the correlation of the methylation level (β) for each CpG-position with specific phenotypes. Where an individual correlation between a phenotype and the methylation level for a CpG-position occurs, is defined as a differentially methylated position (dmp) [Hansen and Aryee, 2013]. In this thesis we are as mentioned looking at age, a continuous phenotype.

As age is a continuous phenotype, we are dealing with a regression problem. The most simple type of regression is linear regression, and hence a first analysis of the data will be to use linear regression to determine the dmp's. The dmp's can be identified by the `dmpFinder` function, it performs univariate linear regressions of age by each of the CpG-positions. It is then tested by multiple testing if the regression coefficient is equal to zero, i.e. if the regression coefficient is significant.

The `Methylset` is turned into the M-values as computed in (2.2) for use in these tests, with a threshold for β (computed in (2.1)) at 0.001. Since the data is from both males and females, the sex chromosomes might influence the analysis. Females have the double amount of the X-chromosome compared to males, and hence in order to remove a potential false gender effect, the CpG-positions appearing from this chromosome are removed. From the 485,512 CpG-positions, 11,232 appearing from the X-chromosome are removed, which leaves 474,280 CpG-positions for analysis. This data set of M-values, where the positions from the X-chromosome are removed, will be the data applied in all later analysis, unless otherwise stated.

The output of `dmpFinder` is seen below for the six most significantly differentially methylated CpG-positions, where `beta` is the change in mean age per unit increase in level of methylation. The variable `qval`, gives the q-value, which is the false discovery rate (FDR) when performing multiple testing. It is the expected proportion of CpG-positions which is incorrectly called significant. The object `M.noX` denotes the data, where CpG-positions from the X-chromosome are removed.

```
> dmp <- dmpFinder(M.noX, pheno=age, type="continuous")
> dmp[1:6,]
            intercept        beta          t         pval         qval
cg10501210  3.8050619 -0.05224796 -12.914126 2.147430e-16 9.998429e-11
cg16867657 -0.7246028  0.02805765  12.000476 2.576239e-15 5.997481e-10
cg22454769 -1.3324585  0.02815874  11.213349 2.386375e-14 3.703653e-09
cg06639320 -1.3242045  0.01790768  10.503227 1.904581e-13 2.216931e-08
cg04875128 -5.0814385  0.04998958  10.007979 8.425975e-13 7.846264e-08
cg08128734  1.5105158 -0.02304459  -9.624023 2.726463e-12 1.990679e-07
```

In Figure 3.1 linear regression with the six most significant differentially methylated CpG-positions are illustrated one at a time. Linear trends are clear, although some outliers are appearing. By visually inspecting 220 of these most significant dmp's, it is more clear that the slope of the regression line and the goodness of fit indicates whether the CpG-position is a good predictor of age or not. The closer the slope is to zero, the more difficult it gets to measure if the subject is young or old. Moreover a high variance among the samples makes a prediction uncertain. A slope that indicates a great change in the level of methylation during age would hence be preferable together with a good fit to the regression line.

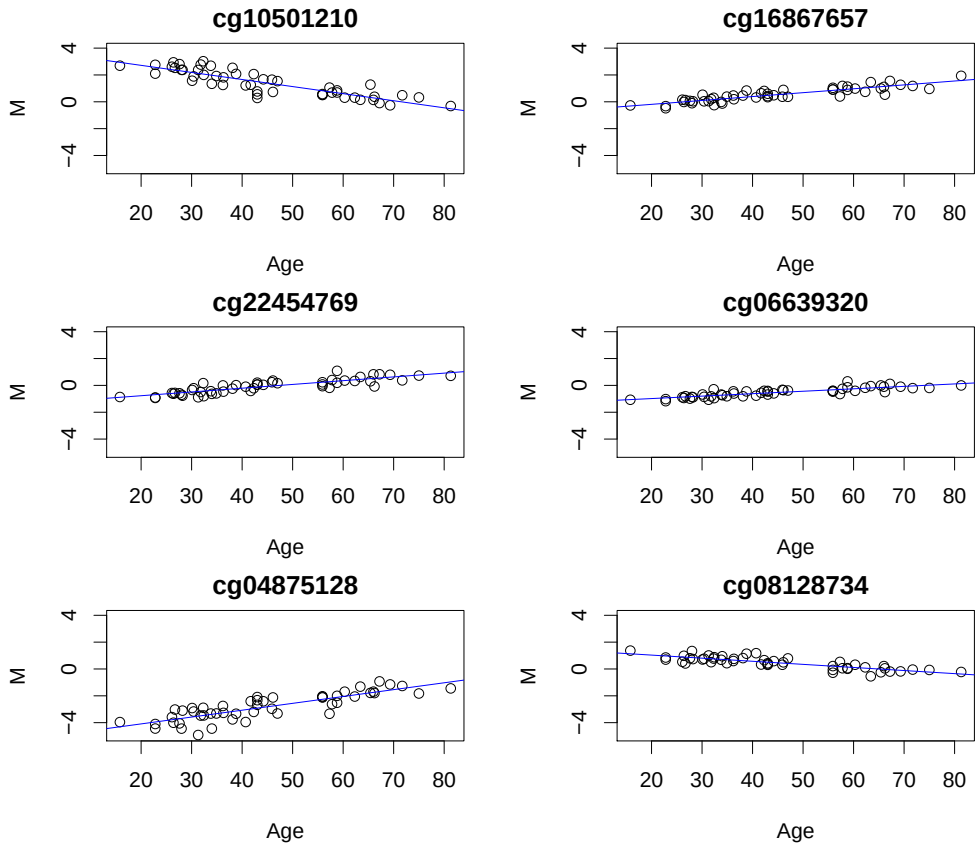


Figure 3.1: Linear trends with age of the six most significant differentially methylated CpG-positions.

4 High-Dimensional Regression

In the preliminary analysis a linear trend between a persons age and the level of methylated DNA in the blood was seen. It is then the purpose to build a model which is able to explain this linear trend in the best possible way, and hence to pick those CpG-markers from the data with the most predictive value in relation to a persons age. The most classical method used for fitting linear models is the least squares method. However, this method is not feasible due to singularity if dimensions of the data is high, i.e. when the data contains more predictors than observations, and hence other methods should be considered. Some of the major problems with high dimensional data are the risk of overfitting a model and the concept of collinearity.

As we in this particular case work with a data set with 50 observations and 474,280 predictors, we are dealing with a high dimensional regression problem, in which the above mentioned issues needs to be considered. In this chapter the problems of overfitting and collinearity will be deeper explained, and appropriate methods for dealing with these issues will be introduced to make the foundation of the methods determined in the following chapters.

4.1 Overfitting

When the number of predictors is high compared to the number of observations in a data set, care should be taken when estimating the prediction error. The complexity of the model plays a key role in this connection. A model tends to fit a set of training data more and more accurate, the more complex it gets, that is when more and more predictors are added to the model, no matter if they truly are associated with the response. This is the concept of overfitting, as the model follows the errors or the noise in the data to closely. In this case the model will not be able to predict new data well, and hence the error of independent test data will increase with the complexity, unless the predictors truly are associated with the response. The error of a model fitted to training data is hence misleading, and conclusions should always be made from the error of a model fitted to a test set which has not been used to train the model. Cross-validation can also be used as a valid method for estimating the prediction error [James et al., 2013].

The reason why the prediction errors increases when a model becomes more flexible by adding more variables, is due to an increase of the variance of the model. The variance should be understood as the variability of estimating the model by other sets of training data. Not much change in the data would cause big changes to a model with high flexibility, and hence the variance will get high. Another term of interest when estimating the accuracy of a model is the bias. The bias is the difference between the predicted and the observed values. This term will generally decrease with the complexity, as the model will fit the data more well the more flexible it gets. A trade off between a low bias and a low variance will give the best prediction as illustrated in Figure 4.1, where the total error

of a model will be minimal around the point where the bias and the variance intersect. Overfitting will then occur after passing this point [James et al., 2013].

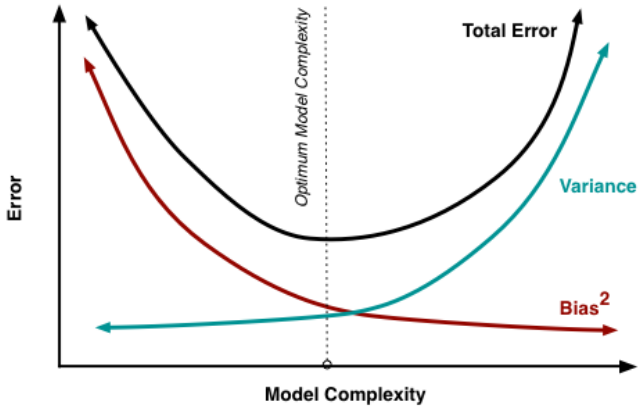


Figure 4.1: The bias-variance trade-off [Fortmann-Roe].

To illustrate the problem of overfitting, the methylation data is fitted by linear regression models, where the number of predictors is increased by one each time. Cross validation is used to estimate the prediction error. As we cannot use more predictors than observations in a linear model to obtain a unique solution, a number between 1 – 48 CpG-positions can be used in the model (due to the intercept and the degrees of freedom). One CpG-marker at a time of the 48 most significant ones found by the `dmpFinder`-function in Section 3.3 is added in a linear model, and each time leave-one-out cross validation is performed. The prediction error for each model computed as the Akaike Information Criterion (AIC) is plotted in Figure 4.2.

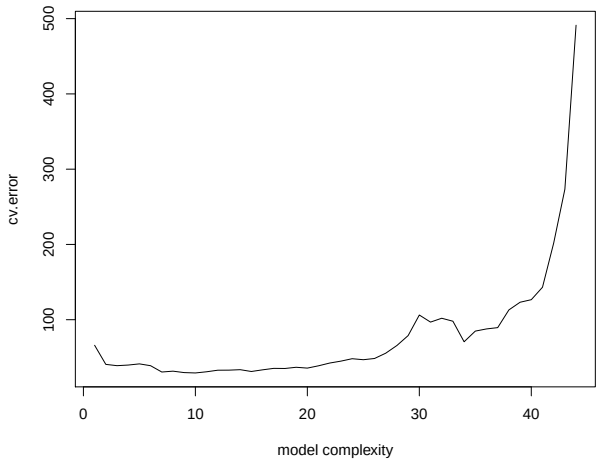


Figure 4.2: The cross validation error of linear models of age by DNA methylation with different numbers of CpG-positions.

Figure 4.2 shows that the error as expected increases with the complexity of the model, even though all of the 48 CpG-positions should be associated with the response. A model with around ten CpG-positions would yield the most reasonable model, as the error is minimal here. This is one way to find and select some significant CpG-positions for prediction of age by the level of DNA methylation and avoid overfitting. We are though

by this method constrained to use 10 CpG-markers, and we do not know whether some of the less significant positions would contribute to a better prediction in combination with more significant positions. A deeper investigation of all of the 474,280 CpG-positions should therefore be made, to find the best combination of CpG-positions for prediction of age.

As the full data set contains 474,280 predictors and only 50 observations, it is not possible to fit a linear model by all of the predictors. A linear regression model is normally fitted by the least squares method, which corresponds to minimizing the residual sum of squares (RSS) given by

$$\text{RSS}(\boldsymbol{\beta}) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2, \tag{4.1}$$

with N being the number of observations and p being the number of predictors. The solution of the minimization is

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y},$$

where $\boldsymbol{X} = [\mathbf{1}, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_p]$ and $\boldsymbol{x}_j = (x_{1j}, \ldots, x_{Nj})^T$.

The term $\boldsymbol{X}^T \boldsymbol{X}$ will be singular when the number of predictors are higher than the number of observations, due to $\boldsymbol{X}$ not having full rank. Because of this the variance of the regression coefficient estimates will be infinite, causing the variance of the model to be infinite. Shrinkage methods like Ridge regression, Elastic net and Lasso are methods appropriate for handling high dimensional problems like this. As we also saw in Figure 4.1, the variance of a model will increase as the number of predictors increases. The shrinkage methods are capable of using all of the 474,280 predictors, as they are linear regression models with a shrinkage penalty. This penalty term adds some bias to the model and in this way a bias-variance tradeoff is achieved as it causes a reduction of the variance of the model. Elastic net and Lasso does even perform feature selection, and a less complex model is obtained by these methods. Partial least squares is another useful method in the high dimensional setting. It reduces the dimension of the original problem by using a reduced number of transformed predictors in stead of the original predictors in a linear model. Another reason why standard linear regression should not be used, is due to collinearity.

4.2 Collinearity

A problem in regression models, especially for high dimensional data, is that two or more predictors might remind of each other, in which case we call them collinear or multicollinear if a group of predictors can be formed by linear combinations of each other. For this data set the problem is essential, the predictors are variations of each other, as they all contain the same measure, namely the level of methylated DNA in a persons blood. For this reason the predictors actually associated with the response age all remind of each other, and can to some extent be described as linear combinations of each other.

When collinearity occurs, some of the predictors $\boldsymbol{x}_j$ can be described as linear combinations of other predictors, which causes the term $\boldsymbol{X}^T \boldsymbol{X}$ to become nearly singular. Com-

plete singularity would cause no unique solution of $\hat{\boldsymbol{\beta}}$. Where it is close to singular, results will be unstable, and small changes to the data will have a big influence on $\hat{\boldsymbol{\beta}}$ [Dorman et al., 2013], causing an increased variance.

It becomes a problem in a linear regression model, when testing the significance of predictors included in the model. Standard errors of the regression coefficients will increase due to the increased variance, and induce a low t-statistic as the t-statistic for each predictor is given by $(\hat{\beta}_j - \beta_0)/\text{SE}_{\beta_j}$. A low t-statistic may cause that collinear predictors incorrectly will be discarded from the model, since they will not be stated statistically significant. This is even though they are truly associated with the response [James et al., 2013, Madsen and Thyregod, 2011].

Collinearity can be detected by determining the correlation between predictors. A rule of thumb or a threshold for detection of collinearity is a pairwise absolute correlation above $0.5 - 0.7$ [Dorman et al., 2013]. As an example, take a look at the pairwise correlations between the four most significant CpG-markers found by `dmpFinder`

	cg10501210	cg16867657	cg22454769	cg06639320
cg10501210	1.00	-0.81	-0.80	-0.77
cg16867657	-0.81	1.00	0.87	0.86
cg22454769	-0.80	0.87	1.00	0.94
cg06639320	-0.77	0.86	0.94	1.00

As seen, all of the pairwise absolute correlations are above 0.7, and are therefore highly correlated and indicates collinearity. In Figure 4.3 the CpG-markers are plotted against each other and age to show their linear relation. They all have a linear relation to the response age, but they do also have a linear relation to each other as the correlation coefficients also show.

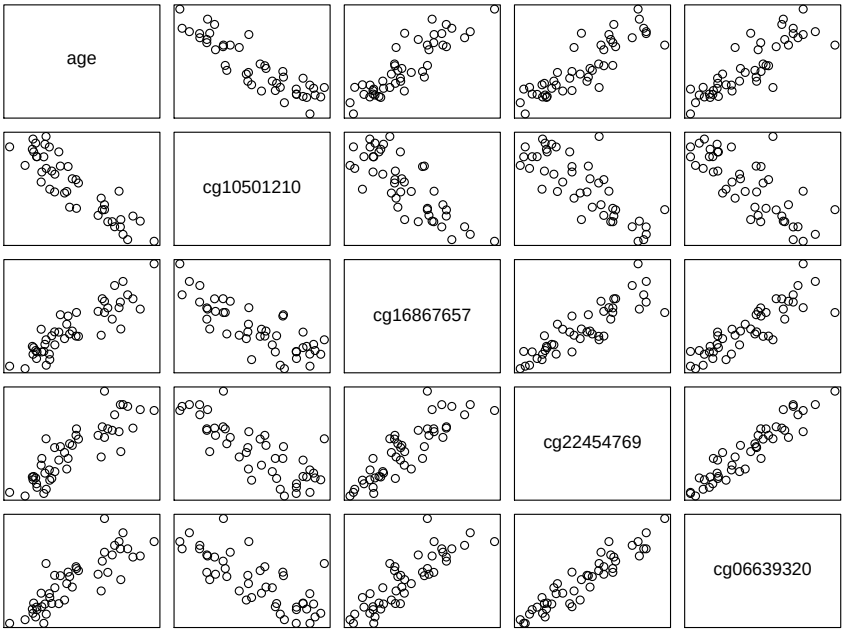


Figure 4.3: Relation between the four most significant CpG-markers and age.

To show how collinearity influence the significance of predictors, a linear model of age is fit by the CpG-markers cg22454769 and cg06639320 as they showed to be most collinear by a correlation coefficient at 0.94. The result is shown in Figure 4.4, where the data `M` contains the response age and the data of the two CpG-markers.

Figure 4.4: Linear regression of age by the CpG-markers cg22454769 and cg06639320.

```
> fit <- lm(age ~ cg22454769 + cg06639320,data=M)
> summary(fit)

Call:
lm(formula = age ~ cg22454769 + cg06639320, data = M)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   53.830      5.285  10.186 6.46e-13 ***
cg22454769    17.451      6.944   2.513   0.0159 *
cg06639320    14.787     10.730   1.378   0.1755
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Only one of the two CpG-markers is stated statistically significant, and the significance level is only just reached with a p-value at 0.0159. Moreover the t-value is low for both of the CpG-markers, due to the large standard error. Removing one of the CpG-markers changes a lot, as shown in the outputs in Figures 4.5 and 4.6. A great decrease in the standard error is seen, which causes the t-value to increase, and the CpG-marker becomes much more statistically significant. A linear model with both of these CpG-markers would thus not be optimal, even though they both are significant predictors of age.

Figure 4.5: Linear regression of age by the CpG-marker cg06639320.

```
> fit <- lm(age ~ cg06639320, data=M)
> summary(fit)

Call:
lm(formula = age ~ cg06639320, data = M)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   65.856      2.376   27.71  < 2e-16 ***
cg06639320    40.180      3.826   10.50  1.9e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 4.6: Linear regression of age by the CpG-marker cg22454769.

```
> fit <- lm(age ~ cg22454769, data = M)
> summary(fit)

Call:
lm(formula = age ~ cg22454769, data = M)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   46.754      1.267   36.91  < 2e-16 ***
cg22454769    26.463      2.360   11.21 2.39e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We are interested in using collinear predictors in the same model, as some of them might explain something that the others do not, and hence other methods accounting for this problem should be applied. As the problem with collinearity is high variance, the shrinkage methods also accounts for this, by the addition of some bias in the model, leading to a reduction of variance of the model. Especially Ridge regression is appropriate for solving this problem, as this method does not perform feature selection. In the Partial least squares method, the transformed variables are uncorrelated, and collinearity will hence not exist.

5 Shrinkage Methods

Techniques similar to least squares but where the coefficient estimates become regularized can be used for fitting regression models on high dimensional data. Such techniques are called shrinkage methods, since they shrink the regression coefficients towards zero [James et al., 2013].

In the following three regularized variants of the least squares method; Ridge regression, Lasso and Elastic net, will be presented. The theory is from [Hastie et al., 2009, James et al., 2013] unless otherwise stated.

Let N be the number of observations and let p be the number of predictors also known as features in a data set $(\boldsymbol{X}, \boldsymbol{y})$. The design matrix $\boldsymbol{X}$ is of dimension $N \times p$, where $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p]$ and $\boldsymbol{x}_j = (x_{1j}, \ldots, x_{Nj})^T$ for $j = 1, \ldots, p$. The response $\boldsymbol{y}$ is of dimension $N \times 1$ and is defined as $\boldsymbol{y} = (y_1, \ldots, y_N)^T$.

5.1 Ridge Regression

Recalling that the Least squares method corresponds to minimizing the RSS given in (4.1), the Ridge regression coefficient estimates are given by

$$\begin{aligned}\hat{\boldsymbol{\beta}}^{\text{Ridge}} &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \\ &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \text{RSS} + \lambda \sum_{j=1}^{p} \beta_j^2 \right\},\end{aligned}$$

here $\lambda \geq 0$ is a tuning parameter, and the term $\lambda \sum_{j=1}^{p} \beta_j^2$ that makes the difference between least squares and Ridge regression, is called the shrinkage penalty. It is in fact an L_2-penalty, since the L_2-norm of $\boldsymbol{\beta}$ is $\left\|\beta\right\|_2 = \sum_{j=1}^{p} \beta_j^2$. The tuning parameter controls the amount of shrinkage of the coefficient estimates β_j, i.e. when $\lambda \to \infty$, $\beta_j \to 0$. When $\lambda = 0$, the Ridge regression coefficient estimates become the least squares coefficient estimates.

It is a constrained optimization problem which also can be written as

$$\hat{\boldsymbol{\beta}}^{\text{Ridge}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \qquad \text{subject to} \sum_{j=1}^{p} \beta_j^2 \leq t.$$

It should be noticed that shrinkage is not applied to the intercept β_0 in the shrinkage penalty term, since when the features $\boldsymbol{x}_j = 0$, the intercept is a measure of the mean of the response, which should not be shrunken. If the feature data has been centered to have mean zero before Ridge regression is performed, then $\hat{\beta}_0 = \bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$.

In matrix form the Ridge regression would be the minimization of

$$\text{RSS}(\lambda) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda\boldsymbol{\beta}^T\boldsymbol{\beta},$$

where $\boldsymbol{X} = [\mathbf{1}, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_p]$ and $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_p)^T$.

The solution is then

$$\begin{aligned}
\frac{\partial \text{RSS}(\lambda)}{\partial \boldsymbol{\beta}} &= -2\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + 2\lambda\boldsymbol{\beta} = -2\boldsymbol{X}^T\boldsymbol{y} + 2(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})\boldsymbol{\beta}, \\
\frac{\partial \text{RSS}(\lambda)}{\partial \boldsymbol{\beta}} &= 0 \Leftrightarrow \\
\hat{\boldsymbol{\beta}}^{\text{Ridge}} &= (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y},
\end{aligned} \tag{5.1}$$

with $\boldsymbol{I}$ being the identity matrix. The addition of λ to the term $\boldsymbol{X}^T\boldsymbol{X}$, solves the problem of singularity.

5.1.1 Singular Value Decomposition

Another way of expressing Ridge regression can be done by making a singular value decomposition (SVD) of the $N \times p$ input matrix $\boldsymbol{X}$. In this way, the understanding of the shrinkage effect of the method can be enhanced. The usual SVD when $N > p$ is given by

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T, \tag{5.2}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal matrices of dimensions $N \times p$ and $p \times p$ respectively. The columns of $\boldsymbol{U}$ are spanning the column space of $\boldsymbol{X}$ whereas columns of $\boldsymbol{V}$ are spanning the row space of $\boldsymbol{X}$. $\boldsymbol{D}$ is a diagonal matrix of dimension $p \times p$, where the diagonal entries $d_1 \geq d_2 \geq \ldots \geq d_p \geq 0$, these are called the singular values of $\boldsymbol{X}$. $\boldsymbol{X}$ is singular if one or more of the diagonal entries are equal to zero. The situation where $p >> N$ will be explained in Section 5.1.3.

By the singular value decomposition of $\boldsymbol{X}$, the Ridge regression fitted vector is computed as

$$\begin{aligned}
\boldsymbol{X}\hat{\boldsymbol{\beta}}^{\text{Ridge}} &= \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y} \\
&= \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y} \\
&= \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y} \\
&= \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T + \lambda\boldsymbol{V}\boldsymbol{V}^T)^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y} \\
&= \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}(\boldsymbol{D}^2 + \lambda\boldsymbol{I})\boldsymbol{V}^T)^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y} \\
&= \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}^T)^{-1}(\boldsymbol{D}^2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y} \\
&= \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T\boldsymbol{V}(\boldsymbol{D}^2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y} \\
&= \boldsymbol{U}\boldsymbol{D}(\boldsymbol{D}^2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y}.
\end{aligned}$$

Writing out the expression yields

$$\begin{aligned}\boldsymbol{X}\hat{\beta}^{\text{Ridge}} =& \boldsymbol{U}\boldsymbol{D}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y}\\ =& \sum_{j=1}^{p}\boldsymbol{u}_j\frac{d_j^2}{d_j^2+\lambda}\boldsymbol{u}_j^T\boldsymbol{y},\end{aligned} \tag{5.3}$$

with $\boldsymbol{u}_j$ indicating the j'th column of $\boldsymbol{U}$. Since $\lambda \geq 0$, the term $\frac{d_j^2}{d_j^2+\lambda} \leq 1$, and hence as $\lambda \to \infty$, it has the effect of shrinking the values of the fitted vector. Shrinkage where d_j^2 is small, would then be greater than for bigger values of d_j^2.

In the same way we can compute the variance of the Ridge regression coefficient estimate

$$\begin{aligned}\text{Var}\left(\hat{\beta}^{\text{Ridge}}\right) =& \text{Var}\left((\boldsymbol{X}^T\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\boldsymbol{X}\boldsymbol{y}\right)\\ =&(\boldsymbol{X}^T\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\text{Var}(\boldsymbol{y})\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\\ =&(\boldsymbol{X}^T\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\sigma^2\boldsymbol{I}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\\ =&(\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\sigma^2\boldsymbol{I}\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T+\lambda\boldsymbol{I})^{-1}\\ =&(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\sigma^2\boldsymbol{I}\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T+\lambda\boldsymbol{I})^{-1}\\ =&(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T+\lambda\boldsymbol{V}\boldsymbol{V}^T)^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\sigma^2\boldsymbol{I}\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T+\lambda\boldsymbol{V}\boldsymbol{V}^T)^{-1}\\ =&(\boldsymbol{V}(\boldsymbol{D}^2+\lambda\boldsymbol{I})\boldsymbol{V}^T)^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\sigma^2\boldsymbol{I}\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}(\boldsymbol{D}^2+\lambda\boldsymbol{I})\boldsymbol{V}^T)^{-1}\\ =&(\boldsymbol{V}^T)^{-1}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^{-1}\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\sigma^2\boldsymbol{I}\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}^T)^{-1}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^{-1}\\ =&\boldsymbol{V}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\sigma^2\boldsymbol{I}\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T\boldsymbol{V}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\\ =&\boldsymbol{V}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{D}\boldsymbol{U}^T\sigma^2\boldsymbol{I}\boldsymbol{U}\boldsymbol{D}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\\ =&\sigma^2\boldsymbol{V}(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{D}^2(\boldsymbol{D}^2+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T.\end{aligned}$$

Writing it out gives

$$\text{Var}\left(\hat{\beta}^{\text{Ridge}}\right) = \sigma^2\sum_{j=1}^{p}\boldsymbol{v}_j\frac{d_j^2}{(d_j^2+\lambda)^2}\boldsymbol{v}_j^T.$$

As λ increases, it is seen that the variance of the coefficient estimates decreases.

Singular value decomposition can be used for expression of principal components, and in this way the meaning of the size of the values d_j^2 can be shown. The sample covariance matrix is

$$\boldsymbol{S} = \frac{\boldsymbol{X}^T\boldsymbol{X}}{N},$$

and by (5.2)

$$\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T = \boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T.$$

This is the eigen decomposition of $\boldsymbol{X}^T\boldsymbol{X}$, where $\boldsymbol{v}_j$ are the eigenvectors. The eigenvectors are the principal component directions of $\boldsymbol{X}$. The first principal component of $\boldsymbol{X}$ is

given by $\boldsymbol{z_1} = \boldsymbol{X}\boldsymbol{v_1}$ and has the largest sample variance comparing all normalized linear combinations of the columns in $\boldsymbol{X}$. The sample variance is computed as

$$\begin{aligned}\frac{\boldsymbol{z}_j^T\boldsymbol{z}_j}{N} &= \frac{\boldsymbol{v}_j^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{v}_j}{N} = \frac{\boldsymbol{v}_j^T\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^T\boldsymbol{v}_j}{N} \\ &= \frac{\boldsymbol{e}_j\boldsymbol{D}^2\boldsymbol{e}_j^T}{N} = \frac{d_j^2}{N},\end{aligned}$$

where $\boldsymbol{e}_j$ is the j'th column of the identity matrix. The normalized principal components are $\boldsymbol{u}_j$, since

$$\boldsymbol{z}_j = \boldsymbol{X}\boldsymbol{v}_j = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T\boldsymbol{v}_j = \boldsymbol{U}\boldsymbol{D}\boldsymbol{e}_j^T = \boldsymbol{u}_j d_j.$$

The principal components are orthogonal to the earlier ones and the last one has the lowest variance. That is, small values of d_j means a low variance of the principal components of $\boldsymbol{X}$. Connecting this with (5.3), $\boldsymbol{U}$ is spanning the column space of $\boldsymbol{X}$, and the columns of $\boldsymbol{U}$ are the normalized principal components, thus Ridge regression shrinks the directions with low variance in the column space of $\boldsymbol{X}$ the most.

5.1.2 Ridge regression and bias

An advantage of Ridge regression is as mentioned, that by adding some bias to the model in the form of λ, it reduces the variance of the estimates which yields a lower prediction error for the model.

The Ridge regression model is biased when $\lambda > 0$. We know that an estimator is unbiased if $\mathbb{E}\left[\hat{\boldsymbol{\beta}}\right] = \boldsymbol{\beta}$, the bias is hence given by $\text{Bias}(\hat{\boldsymbol{\beta}}) = \mathbb{E}\left[\hat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}$. The bias of the Ridge regression estimator is

$$\begin{aligned}\text{Bias}(\hat{\boldsymbol{\beta}}^{\text{ridge}}) &= \mathbb{E}\left[\hat{\boldsymbol{\beta}}^{\text{ridge}}\right] - \boldsymbol{\beta} \\ &= \mathbb{E}\left[(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}\right] - \boldsymbol{\beta} \\ &= (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\mathbb{E}[\boldsymbol{y}] - \boldsymbol{\beta} \\ &= (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{\beta} \\ &= (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I} - \lambda\boldsymbol{I})\boldsymbol{\beta} - \boldsymbol{\beta} \\ &= (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})(\boldsymbol{I} - \lambda\boldsymbol{I}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1})\boldsymbol{\beta} - \boldsymbol{\beta} \\ &= (\boldsymbol{I} - \lambda\boldsymbol{I}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1})\boldsymbol{\beta} - \boldsymbol{\beta} \\ &= -\lambda\boldsymbol{I}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{\beta}.\end{aligned}$$

Notice that we get $\text{Bias}(\hat{\boldsymbol{\beta}}^{\text{ridge}}) = 0$ when $\lambda = 0$, since this is just the Ordinary Least Squares estimator, which is an unbiased estimator. As seen the bias increases with λ.

5.1.3 When $p >> N$

When we are dealing with data matrices $\boldsymbol{X}$, where the dimension of the columns (p) is much bigger than the dimension of the rows (N), we can use the SVD to reduce the computations to N dimensions rather than p. This method is usable only for methods fitting linear models with a quadratic regularization on the coefficients. It results in a reduction of computational cost from $O(p^3)$ to $O(pN^2)$ [Hastie et al., 2009].

Making SVD on a matrix where $p >> N$ corresponds to making usual SVD as in (5.2), but at the transpose of $\boldsymbol{X}$, where the dimension of the columns of $\boldsymbol{X}$ is N and the dimension of the rows is p. This matrix with more rows than columns has at most rank N. We will then obtain the matrix $\tilde{\boldsymbol{X}}$, with dimension $N \times p$, where $p >> N$ as follows

$$\tilde{\boldsymbol{X}} = \boldsymbol{X}^T = (\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T)^T = \boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^T = \tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}\tilde{\boldsymbol{V}}^T, \tag{5.4}$$

where $\tilde{\boldsymbol{U}}$ is orthogonal and $\tilde{\boldsymbol{V}}$ has orthonormal columns, the matrices are of dimensions $N \times N$ and $p \times N$ respectively. The diagonal matrix $\tilde{\boldsymbol{D}}$ is of dimension $N \times N$, since the rank of $\boldsymbol{X}$ is at most N when $p >> N$.

The singular value decomposition can hence be defined as follows

$$\begin{aligned} \tilde{\boldsymbol{X}} =& \tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}\tilde{\boldsymbol{V}}^T \\ =& \boldsymbol{R}\tilde{\boldsymbol{V}}^T. \end{aligned} \tag{5.5}$$

This makes $\boldsymbol{R}$ an $N \times N$ matrix. The estimates of Ridge regression can then be expressed as

$$\hat{\beta}^{\text{Ridge}} = \underbrace{\tilde{\boldsymbol{V}}(\boldsymbol{R}^T\boldsymbol{R} + \lambda\boldsymbol{I})^{-1}}_{*}\boldsymbol{R}^T\boldsymbol{y}, \tag{5.6}$$

since using (5.5) in (5.1) we get

$$\begin{aligned} \hat{\boldsymbol{\beta}}^{\text{Ridge}} =& (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y} \\ =& \left((\boldsymbol{R}\tilde{\boldsymbol{V}}^T)^T\boldsymbol{R}\tilde{\boldsymbol{V}}^T + \lambda\boldsymbol{I}\right)^{-1}(\boldsymbol{R}\tilde{\boldsymbol{V}}^T)^T\boldsymbol{y} \\ =& \underbrace{(\tilde{\boldsymbol{V}}\boldsymbol{R}^T\boldsymbol{R}\tilde{\boldsymbol{V}}^T + \lambda\boldsymbol{I})^{-1}\tilde{\boldsymbol{V}}}_{**}\boldsymbol{R}^T\boldsymbol{y}. \end{aligned} \tag{5.7}$$

We can then show that (5.6) is equal to (5.7), by showing that $*$ is equal to $**$

$$
\begin{aligned}
&(\tilde{\boldsymbol{V}}\boldsymbol{R}^T\boldsymbol{R}\tilde{\boldsymbol{V}}^T + \lambda\boldsymbol{I})^{-1}\tilde{\boldsymbol{V}} = \tilde{\boldsymbol{V}}(\boldsymbol{R}^T\boldsymbol{R} + \lambda\boldsymbol{I})^{-1} \Leftrightarrow \\
&\tilde{\boldsymbol{V}} = (\tilde{\boldsymbol{V}}\boldsymbol{R}^T\boldsymbol{R}\tilde{\boldsymbol{V}}^T + \lambda\boldsymbol{I})\tilde{\boldsymbol{V}}(\boldsymbol{R}^T\boldsymbol{R} + \lambda\boldsymbol{I})^{-1} \Leftrightarrow \\
&\tilde{\boldsymbol{V}} = (\tilde{\boldsymbol{V}}\boldsymbol{R}^T\boldsymbol{R}\tilde{\boldsymbol{V}}^T\tilde{\boldsymbol{V}} + \lambda\tilde{\boldsymbol{V}})(\boldsymbol{R}^T\boldsymbol{R} + \lambda\boldsymbol{I})^{-1} \Leftrightarrow \\
&\tilde{\boldsymbol{V}} = (\tilde{\boldsymbol{V}}\boldsymbol{R}^T\boldsymbol{R} + \lambda\tilde{\boldsymbol{V}})(\boldsymbol{R}^T\boldsymbol{R} + \lambda\boldsymbol{I})^{-1} \Leftrightarrow \\
&\tilde{\boldsymbol{V}} = \tilde{\boldsymbol{V}}(\boldsymbol{R}^T\boldsymbol{R} + \lambda\boldsymbol{I})(\boldsymbol{R}^T\boldsymbol{R} + \lambda\boldsymbol{I})^{-1} \Leftrightarrow \\
&\tilde{\boldsymbol{V}} = \tilde{\boldsymbol{V}}.
\end{aligned}
$$

The Ridge regression model can hence be expressed as $\hat{\boldsymbol{\beta}}^{\text{Ridge}} = \tilde{\boldsymbol{V}}\hat{\theta}$, with $\hat{\theta}$ being the Ridge regression estimates of (r_i, y_i) for $i = 1,\ldots,N$. In a high dimensional setting the data matrix $\tilde{\boldsymbol{X}}$ can be reduced to $\boldsymbol{R}$, and the penalized fit is then made on the rows of $\boldsymbol{R}$, which has fewer predictors. Afterward the result of the fit is transformed back to the p-dimensional vector solution, by multiplying it with the matrix $\tilde{\boldsymbol{V}}$.

5.1.4 Example

A small example of how to use Ridge regression will now be presented, where some simulated data will be used. A data matrix $\boldsymbol{X}$ is generated by drawing independently from the standard normal distribution, where the dimensions are chosen to be 30×100, i.e. a fictive data set with 100 features and 30 observations. The response $\boldsymbol{y}$ is randomly chosen to be the addition of the first and sixth column of $\boldsymbol{X}$ with some noise.

```
n = 30
p = 100

set.seed(3177)
X = matrix(rnorm(n*p), ncol=p)
y = X[,1] + X[,6]+ rnorm(n)
```

The Ridge regression solutions can then be computed by the function `glmnet()` in the `R`-package `glmnet`. It is a function capable of fitting generalized linear models by different kinds of penalized least squares methods. It optimizes

$$
\text{RSS}(\boldsymbol{\beta}) + \lambda\left[\frac{(1-\alpha)}{2}\|\boldsymbol{\beta}\|_2^2 + \alpha\|\boldsymbol{\beta}\|_1\right], \tag{5.8}
$$

which is a shrinkage method called Elastic net, it is a combination of Lasso and Ridge regression described in Section 5.4. To obtain the Ridge regression shrinkage method, `alpha` is set to 0. As default the function performs regression on 100 automatically chosen values of λ. The beta coefficient estimates for these λ's can be reached by `coef()`, which is a matrix, in this case with 101 rows (one for each feature and an intercept) and 100 columns (one for every value of λ).

```
> ridge.mod <- glmnet (X,y,alpha =0)

> dim(coef(ridge.mod))
[1] 101 100

#One of the lambdas
```

```
> ridge.mod$lambda[50]
[1] 160.5806

#The first 10 regression coefficients for lambda=50
> coef(ridge.mod)[,50][1:10]
 (Intercept)           X1           X2           X3           X4           X5
 0.270847903  0.019302946  0.008606355 -0.007749915  0.002828386  0.008156471
          X6           X7           X8           X9          X10
 0.016153727  0.003055189 -0.001431342  0.003970974  0.001077364
```

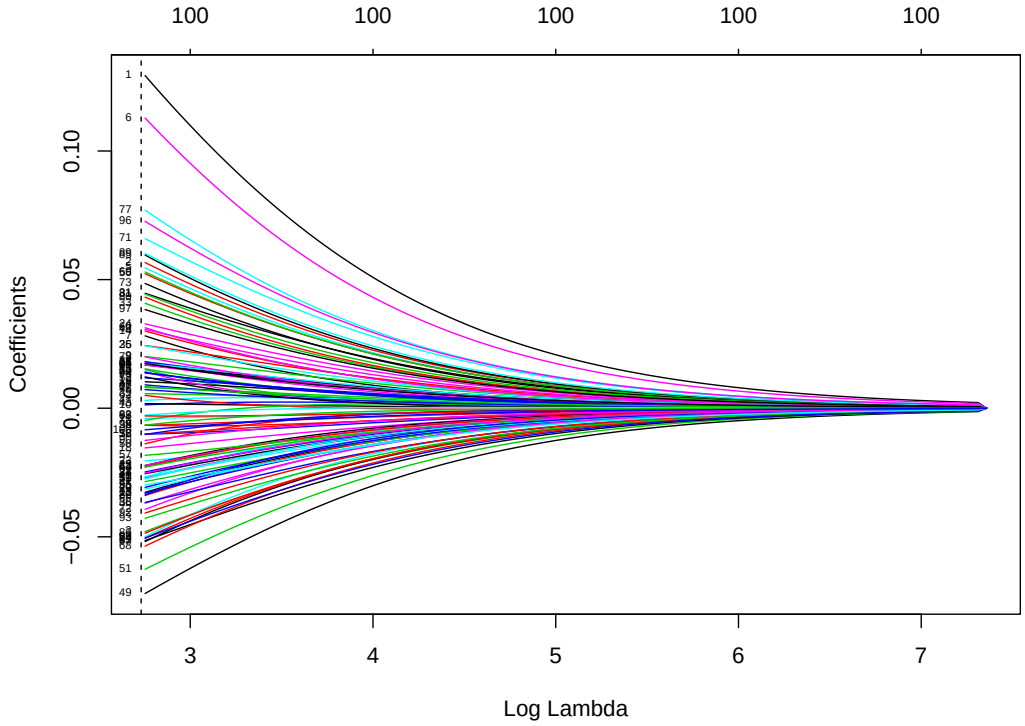


Figure 5.1: The Ridge regression coefficient estimates for different values of $\log(\lambda)$. The dashed vertical line indicates the best value of λ, found by leave-one-out cross-validation. The numbers in the top of the plot, illustrates non-zero coefficients.

The Ridge regression coefficient estimates are plotted as a function of the 100 values of $\log(\lambda)$ in Figure 5.1, where the numbers in the top of the plot indicates the number of non-zero coefficient estimates as λ gets larger. It is seen that as λ gets larger the coefficient estimates shrink towards zero, but non actually hits zero. The optimal value of λ is chosen by cross-validation. The data is divided into a training - and a test set for the purpose, and the function `cv.glmnet()` performs as default 10-fold cross validation. Setting `nfolds` in the function equal to the number of observations of the training data, the method performs instead leave one out cross validation. The optimal value of λ is then the one resulting in the lowest mean squared error (MSE) on the training data.

```
> set.seed (317)
> train=sample(1:n, n/2, replace=FALSE)
> test=(- train )
> y.test <- y[test]
```

```
> set.seed (317)
> cv.out <- cv.glmnet (X[train ,], y[train], nfolds=15, alpha=0)
> bestlam <- cv.out$lambda.min
> bestlam
[1] 15.35349

# MSE
> ridge.pred <- predict(ridge.mod, s=bestlam, newx=X[test,])
> mean((ridge.pred-y.test)^2)
[1] 2.192798
```

By cross validation on the training data $\lambda_{\text{best}} = 15.35$ gave the minimal MSE = 2.19 on the test data. λ_{best} is plotted in Figure 5.1 as the dashed vertical line, at $\log(15.35) = 2.73$. Predicting the response with λ_{best}, by the function `predict()` and `type ="coefficients"`, gave the Ridge regression coefficient estimates seen below (of the first twenty). None of these coefficients are shrunken directly to zero as the minimal value of the 100 coefficients is 0.0013, and hence all features are used in the model. Notice however that shrinkage of feature `X1` and `X6` is less than the other features. They are as seen in Figure 5.1 the most important features, since they have been used to form the response. Moreover the maximal value of the coefficients is the value of `X1`.

```
> ridge.coef=predict (ridge.mod, type ="coefficients", s=bestlam)
> ridge.coef[1:21]
                       1
(Intercept)  0.337695179
X1           0.129320036
X2           0.056627091
X3          -0.047963240
X4           0.017414684
X5           0.054591182
X6           0.112815127
X7           0.028119017
X8          -0.006663196
X9           0.020135052
X10          0.001301733
X11         -0.003958387
X12          0.017978184
X13         -0.032709595
X14          0.030024584
X15          0.015043737
X16         -0.010108345
X17         -0.026801563
X18         -0.015428539
X19          0.010281764
X20         -0.033930709

> max(abs(ridge.coef[-1]))
[1] 0.12932

> min(abs(ridge.coef[-1]))
[1] 0.001301733
```

5.2 The Lasso

Lasso stands for least absolute shrinkage and selection operator. It is similar to Ridge regression, but as the name suggests it does not only shrink the coefficient estimates towards zero, it actually sets some of them directly to zero. In this way it also "selects"

the coefficient estimates. Unless otherwise stated, theory in this section is from [Murphy, 2012].

The Lasso is defined in its lagrangian form by

$$\hat{\beta}^{\text{Lasso}} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\},$$

where as for Ridge regression, λ is a tuning parameter. The constrained optimization problem can also be written as

$$\hat{\boldsymbol{\beta}}^{\text{Lasso}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \qquad \text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t.$$

Since the objective function $\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$ is quadratic, and the constrains $\sum_{j=1}^{p} |\beta_j|$ are linear, it is a quadratic programming problem.

The shrinkage penalty $\sum_{j=1}^{p} |\beta_j|$ is an L_1 penalty, since the L_1 norm of $\boldsymbol{\beta}$ is given by $\|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{p} |\beta_j|$. Taking a deeper look at this term, it is seen that it is not differentiable when $\beta_j = 0$, and therefore the derivative of the absolute value $f(\theta) = |\theta|$ is undefined.

The shrinkage penalty makes the optimization problem non-smooth, and for this term the standard definition of a derivative is not usable. A subderivative [Rockafellar, 1997] of such a function $f : I \to \mathbb{R}$ at a point θ_0 can then be defined as a scalar g such that

$$f(\theta) - f(\theta_0) \geq g(\theta - \theta_0) \qquad \forall \theta \in I,$$

with I being the interval containing θ_0. The set of subderivatives is then defined as the interval $[a, b]$, where a and b are the one-sided limits

$$a = \lim_{\theta \to \theta_0^-} \frac{f(\theta) - f(\theta_0)}{(\theta - \theta_0)}, \qquad b = \lim_{\theta \to \theta_0^+} \frac{f(\theta) - f(\theta_0)}{(\theta - \theta_0)}. \tag{5.9}$$

The subdifferential of the function f at θ_0 is denoted $\partial f(\theta)|_{\theta_0}$, it contains the subderivatives of the set $[a, b]$. Using (5.9), we see that for the absolute value function $f(\theta) = |\theta|$, when $\theta > 0$, $f'(\theta) = \frac{\theta}{|\theta|} = 1$. When $\theta < 0$, $f'(\theta) = \frac{\theta}{|\theta|} = -1$. The subderivative of the absolute value is then given by

$$\partial f(\theta) = \begin{cases} \{-1\} & \text{if } \theta < 0 \\ [-1, 1] & \text{if } \theta = 0 \\ \{+1\} & \text{if } \theta > 0. \end{cases}$$

Returning to the Lasso problem, we can find the partial derivatives in the standard way, if we leave the penalty term out

$$\begin{aligned}
\frac{\partial}{\partial \beta_k} RSS(\boldsymbol{\beta}) =& \frac{\partial}{\partial \beta_k} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \\
=& \frac{\partial}{\partial \beta_k} \sum_{i=1}^{N} \left(y_i - \beta_0 - x_{ik}\beta_k - \sum_{j=1, j\neq k}^{p} x_{ij}\beta_j \right)^2 \\
=& -2 \sum_{i=1}^{N} x_{ik} \left(y_i - \beta_0 - x_{ik}\beta_k - \sum_{j=1, j\neq k}^{p} x_{ij}\beta_j \right) \\
=& -2 \sum_{i=1}^{N} x_{ik} \left(y_i - \beta_0 - \sum_{j=1, j\neq k}^{p} x_{ij}\beta_j \right) + 2 \sum_{i=1}^{N} x_{ik}^2 \beta_k \\
=& a_k \beta_k - c_k
\end{aligned}$$

where

$$\begin{aligned}
a_k =& 2 \sum_{i=1}^{N} x_{ik}^2 = 2 \boldsymbol{x}_k^T \boldsymbol{x}_k \\
c_k =& 2 \sum_{i=1}^{N} x_{ik} \left(y_i - \beta_0 - \sum_{j=1, j\neq k}^{p} x_{ij}\beta_j \right) = 2 \boldsymbol{x}_k^T \boldsymbol{r}_k.
\end{aligned} \tag{5.10}$$

The residual $\boldsymbol{r}_k$ is the residual computed with all features $\boldsymbol{x}_j$ except $\boldsymbol{x}_k$. Thus, c_k can be interpreted as being proportional to the correlation between the k'th feature $\boldsymbol{x}_k$ and the residual with all the other features, since the data is assumed to have been centered to have mean zero and unit norm. Hence, c_k is then a measure of how important the k'th feature is in the prediction of the response $\boldsymbol{y}$, in relation to the other features.

The subderivative of the Lasso estimate is by the penalty term added back in, given as

$$\begin{aligned}
\partial_{\beta_k} f(\beta) =& a_k \beta_k - c_k + \lambda \partial_{\beta_k} \left\| \beta \right\|_1 \\
=& \begin{cases} \{a_k \beta_k - c_k - \lambda\} & \text{if } \beta_k < 0 \\ [-c_k - \lambda, -c_k + \lambda] & \text{if } \beta_k = 0 \\ \{a_k \beta_k - c_k + \lambda\} & \text{if } \beta_k > 0. \end{cases}
\end{aligned}$$

When f is a convex function, it can be shown that for a subdifferential, the point $\hat{\theta}$ is a local minimum if and only if, $0 \in \partial f(\theta)|_{\hat{\theta}}$ [Rockafellar, 1997, Thm. 28.3]. We have that in vector form the solution of the partial derivative of the RSS with respect to β_k is $\partial_{\beta_k} RSS(\boldsymbol{\beta}) = 2\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y})_k$.

It is then only true that $0 \in \partial f(\theta)|_{\hat{\theta}}$ if the following holds

$$2\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y})_k \in \begin{cases} \{+\lambda\} & \text{if } \beta_k < 0 \\ [-\lambda, \lambda] & \text{if } \beta_k = 0 \\ \{-\lambda\} & \text{if } \beta_k > 0. \end{cases} \tag{5.11}$$

There is then three possible solutions of $\partial_{\beta_k} f(\beta) = 0$:

1. $\hat{\beta}_k = \frac{c_k+\lambda}{a_k} < 0$, which holds when $c_k < -\lambda$. This corresponds to a strongly negative correlation between the k'th feature and the residual.

2. $\hat{\beta}_k = 0$, which holds when $c_k \in [-\lambda, \lambda]$. In this case there is only a weak correlation between the k'th feature and the residual.

3. $\hat{\beta}_k = \frac{c_k-\lambda}{a_k} > 0$, which holds when $c_k > \lambda$. This corresponds to a strongly positive correlation between the k'th feature and the residual.

To sum up we have

$$\hat{\beta}_k(c_k) = \begin{cases} \frac{c_k+\lambda}{a_k} & \text{if } c_k < -\lambda \\ 0 & \text{if } c_k \in [-\lambda, \lambda] \\ \frac{c_k-\lambda}{a_k} & \text{if } c_k > \lambda. \end{cases}$$

The solution can be written as soft thresholding

$$\hat{\beta}_k^{\text{Lasso}} = \text{soft}\left(\frac{c_k}{a_k}; \frac{\lambda}{a_k}\right) = \text{sign}\left(\frac{c_k}{a_k}\right)\left(\left|\frac{c_k}{a_k}\right| - \frac{\lambda}{a_k}\right)_+,$$

where $x_+ = \max(x, 0)$, i.e. the positive part of x. Setting $\lambda = 0$ corresponds to the ordinary Least Squares (OLS) estimate and hence

$$\hat{\beta}_k^{\text{OLS}} = \frac{c_k}{a_k} \tag{5.12}$$

The maximum value of λ needed to be considered is

$$\lambda_{\max} = \max_k |2\boldsymbol{x}_k^T \boldsymbol{y}|,$$

since for $\beta_k = 0$ we have that $2\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y})_k \in [-\lambda, \lambda]$ from (5.11). That is when $\beta_k = 0$ for all k, we get $2\boldsymbol{x}_k^T \boldsymbol{y} \in [-\lambda, \lambda]$. This means that $\left\|2\boldsymbol{x}_k^T \boldsymbol{y}\right\| < \lambda$ for all k, and in particular the biggest value of $2\boldsymbol{x}_k^T \boldsymbol{y}$ also needs to be less than λ. We then get $\left\|2\boldsymbol{X}^T \boldsymbol{y}\right\|_\infty = \max_k |2\boldsymbol{x}_k^T \boldsymbol{y}| = \lambda_{\max}$.

5.2.1 Why does Lasso perform Feature Selection?

If we compare Lasso and Ridge regression, a geometric interpretation of the two constraint optimization problems with $p = 2$ can be seen in Figure 5.2. For the Lasso the constraint region $|\beta_1| + |\beta_2| \leq t$ is formed like a diamond, with sharp corners on the axes, and for Ridge regression $\beta_1^2 + \beta_2^2 \leq t$ is a circle. The red ellipses on the figure, around the Least Squares estimate $\hat{\boldsymbol{\beta}}$, represents the RSS.

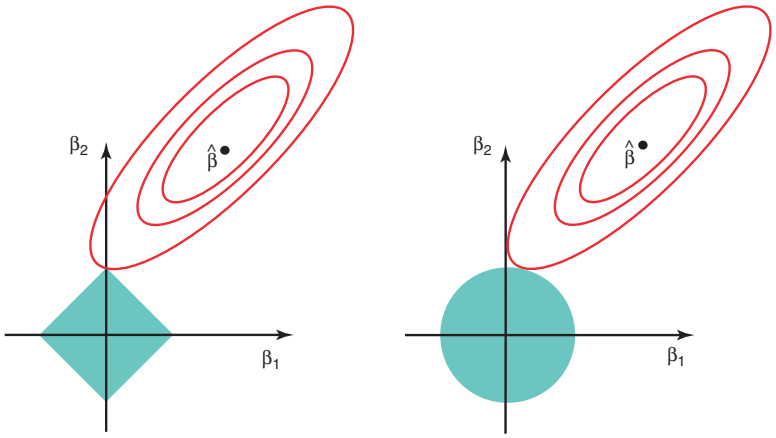


Figure 5.2: Geometric interpretation of Lasso (left) and Ridge regression (right) when $p = 2$. The blue diamond and circle are the constraint regions of the methods, and the red ellipses around the Least Squares estimate $\hat{\beta}$ are the RSS [Hastie et al., 2009, Fig. 3.11].

We know that when $\lambda = 0$ in Lasso or Ridge regression we get the Least Squares estimates, and hence on the figure the constraint regions would contain $\hat{\beta}$ if this was the case. As this is not the case, we get the Lasso or Ridge regression estimates at the first point in which the ellipses of the RSS hits the diamond or circle respectively. Since the diamond has sharp corners on the axes, the ellipse will more likely hit one of these first, rather than edges between the axes. For Ridge regression it is the other way around, since the constraint region is a circle and thus has no corners. For this reason if the solution of the Lasso is at a corner of the diamond, one of the coefficient estimates is equal to zero, and hence the Lasso performs feature selection [James et al., 2013].

5.2.2 Example

The same simulated data as used in Section 5.1.4 is here applied in an example of the Lasso method. Again the function `glmnet()` is used, where `alpha` is set to 1, to obtain the Lasso fit.

```
> n = 30
> p = 100
> set.seed(3177)
> X = matrix(rnorm(n*p), ncol=p)
> y = X[,1] + X[,6]+ rnorm(n)

> lasso.mod <- glmnet(X, y, alpha=1)
```

A plot of the result can be seen in Figure 5.3, where most of the coefficient estimates are shrunken directly to zero as $\log(\lambda)$ gets bigger. The numbers in the top of the plot indicates the number of non-zero coefficient estimates as $\log(\lambda)$ gets bigger.

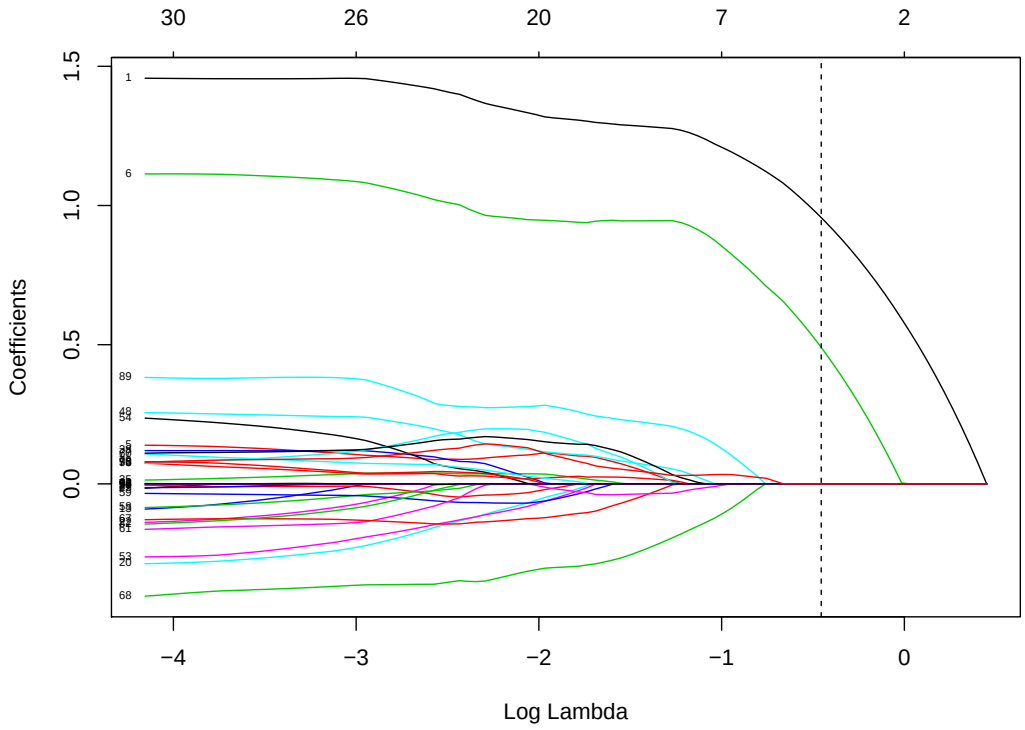


Figure 5.3: The Lasso coefficient estimates for different values of $\log(\lambda)$, where the dashed vertical line indicates the best value of λ found by leave one out cross-validation. The numbers in the top of the plot represent non-zero coefficients.

Using the same split of the data into a training - and a testset, as for Ridge regression, the best value of λ is chosen by leave one out cross validation on the training set, with the function `cv.glmnet()`. The MSE is afterward computed on the test set using the function `predict()`, to get the prediction of the response with the chosen λ.

```
>set.seed(317)
>cv.out=cv.glmnet(X[train,], y[train], alpha=1, nfolds=15)
>bestlam=cv.out$lambda.min
> bestlam
[1] 0.6344081

# MSE
> lasso.pred=predict(lasso.mod, s=bestlam, newx=X[test,])
> mean((lasso.pred-y.test)^2)
[1] 2.068049
```

The best value of λ is plotted as the dashed vertical line in Figure 5.3. This value resulted in an MSE at 2.07, which is a bit lower than the one obtained by Ridge regression at 2.19. Compared to Ridge regression, the Lasso method has the advantage of making feature selection. The non-zero coefficient estimates associated with the chosen λ are

```
> lasso.coef=predict(Lasso.mod, type="coefficients", s=bestlam)
> lasso.coef[which(lasso.coef!=0),]
(Intercept)          X1          X6
  0.1077382   0.9566929   0.4902700
```

All of the features are eliminated from the model, except for `X1` and `X6`, which is as expected, since they are the only features truly associated with the response. In Figure 5.3 these coefficients deviate from the others with much higher values.

5.3 Least Squares versus Ridge Regression and Lasso

If we assume that the columns of the data matrix $\boldsymbol{X}$ are orthonormal, we have that $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}$.

We then get the following results for the three methods

- **Ordinary Least Squares**

$$\hat{\beta}_k^{\text{OLS}} = (\boldsymbol{x}_k^T\boldsymbol{x}_k)^{-1}\boldsymbol{x}_k^T\boldsymbol{y} = \boldsymbol{x}_k^T\boldsymbol{y}.$$

- **Ridge Regression**

$$\hat{\beta}_k^{\text{Ridge}} = (\boldsymbol{x}_k^T\boldsymbol{x}_k + \lambda)^{-1}\boldsymbol{x}_k^T\boldsymbol{y} = \frac{\hat{\beta}_k^{\text{OLS}}}{1+\lambda}.$$

- **Lasso**
  From (5.10) we get that $a_k = 2$, if the columns of $\boldsymbol{X}$ are orthonormal. Then $\hat{\beta}_k^{\text{OLS}} = \frac{c_k}{2}$ from (5.12), and the solution is

$$\hat{\beta}_k^{\text{Lasso}} = \text{sign}\left(\hat{\beta}_k^{\text{OLS}}\right)\left(\left|\hat{\beta}_k^{\text{OLS}}\right| - \frac{\lambda}{2}\right)_+.$$

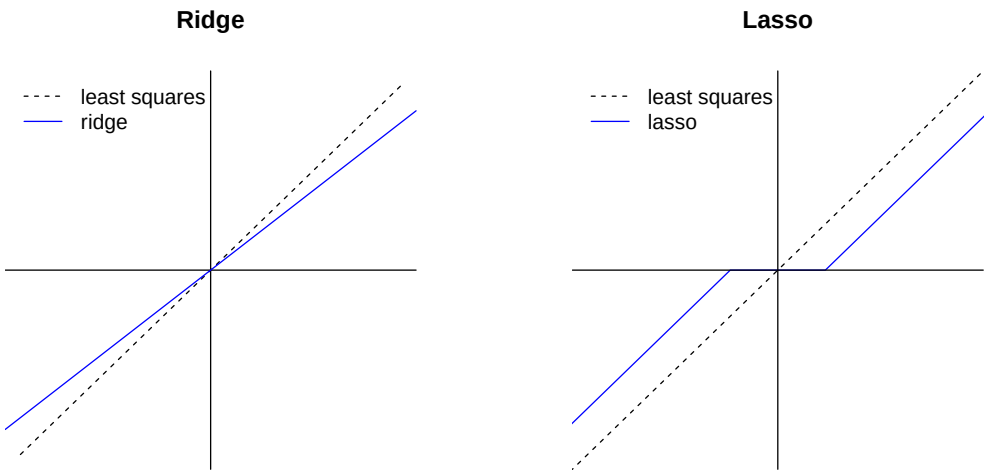


Figure 5.4: Comparison of Ridge regression and Lasso coefficient estimates as a function of the response in relation to the Least Squares. Inspired from [Hastie et al., 2009, Tab. 3.4].

By these expressions of the methods, it is more clear that Lasso and Ridge regression are regularized variants of the Ordinary Least Squares. The estimators are plotted in Figure 5.4 as a function of the response, where it also appears that Lasso and Ridge regression are biased estimators, since they shrink all of the coefficients even when this is not desired, that is when the coefficient estimates have a high value.

5.4 Elastic Net

A combination of Ridge regression and Lasso is the Elastic net, it uses a compromise of the L_1 and L_2 penalties [Zou and Hastie, 2005]. It is given by

$$\hat{\boldsymbol{\beta}}^{\text{Elastic net}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \alpha|\beta_j| + (1-\alpha)\beta_j^2 \right\},$$

where $0 \leq \alpha \leq 1$ controls the mix of the two shrinkage penalties. Setting $\alpha = 0$ yields the Ridge regression solution, and $\alpha = 1$ yields the Lasso solution.

An advantage of Elastic net over Lasso when p>>N is that it is able to select more than N non-zero coefficients. Moreover if a group of features are highly correlated with each other, Lasso would only choose one of these features, whereas Elastic net would choose the whole group, for details see [Zou and Hastie, 2005]. A contour plot of the constrain of Elastic net in two dimensional space compared with the constrains of Ridge regression and Lasso is illustrated in Figure 5.5.

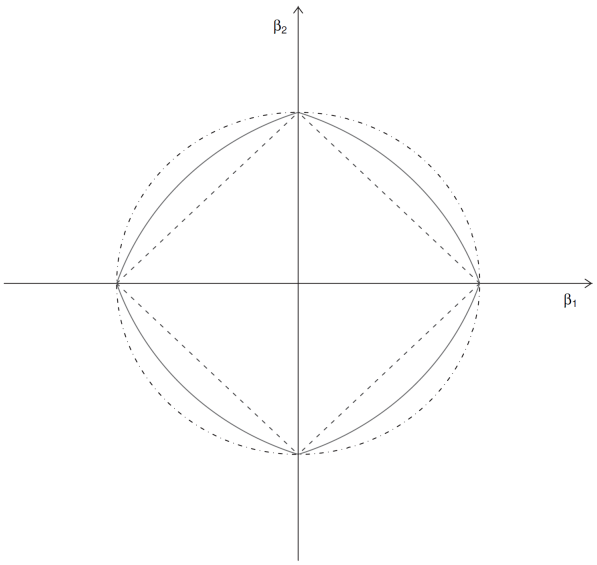


Figure 5.5: Contourplot of the constrains of Ridge regression (the dashed-dotted line), Elastic net (the solid line) with $\alpha = 0.5$ and Lasso (the dashed line)[Zou and Hastie, 2005, Fig. 1].

5.5 A Bayesian Viewpoint of Ridge Regression and Lasso

The two shrinkage methods Ridge regression and Lasso can also be expressed from a Bayesian point of view [Murphy, 2012, James et al., 2013]. From Bayes' theorem it follows that the posterior distribution is given by

$$p(\boldsymbol{\beta}|\boldsymbol{X},\boldsymbol{y}) \propto f(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\beta})p(\boldsymbol{\beta}|\boldsymbol{X}) = f(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\beta})p(\boldsymbol{\beta}), \tag{5.13}$$

where $f(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\beta})$ is the likelihood of the data, and $p(\boldsymbol{\beta})$ is the prior distribution of $\boldsymbol{\beta}$. The equality in (5.13) holds since it is assumed that the feature data $\boldsymbol{X}$ is fixed. If we assume the linear model given as usual with independent errors drawn from a normal distribution, we compute the likelihood of the data as follows,

$$\begin{aligned}
f(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\beta}) &= \prod_{i=1}^{N} \mathcal{N}\left(y_i \middle| \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j, \sigma^2\right) \\
&= \prod_{i=1}^{N} \left(\frac{1}{2\pi\sigma^2}\right)^{\frac{1}{2}} \exp\left(-\frac{1}{2\sigma^2}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2\right) \\
&= \left(\frac{1}{2\pi\sigma^2}\right)^{\frac{N}{2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2\right) \\
&\propto \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2\right).
\end{aligned}$$

Furthermore assume that the prior distribution of $\boldsymbol{\beta}$ is $p(\boldsymbol{\beta}) = \prod_{j=1}^{p} g(\beta_j)$, where g is a density function, then the Lasso and Ridge regression follows from two special choices of g.

5.5.1 Ridge Regression

If the prior of $\boldsymbol{\beta}$ is gaussian distributed with mean zero, and standard deviation $\tau^2 = \sigma^2/\lambda$, where $\lambda > 0$, we get the following

$$\begin{aligned}
p(\boldsymbol{\beta}) &= \prod_{j=1}^{p} \mathcal{N}\left(\beta_j \middle| 0, \tau^2\right) \\
&= \prod_{j=1}^{p} \left(\frac{1}{2\pi\tau^2}\right)^{\frac{1}{2}} \exp\left(-\frac{1}{2\tau^2}\beta_j^2\right) \\
&= \left(\frac{1}{2\pi\tau^2}\right)^{\frac{p}{2}} \exp\left(-\frac{1}{2\tau^2}\sum_{j=1}^{p}\beta_j^2\right) \\
&\propto \exp\left(-\frac{1}{2\tau^2}\sum_{j=1}^{p}\beta_j^2\right),
\end{aligned}$$

and the posterior is

$$\begin{aligned} p(\boldsymbol{\beta}|\boldsymbol{X},\boldsymbol{y}) &\propto f(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\beta})p(\boldsymbol{\beta}) \\ &\propto \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2-\frac{1}{2\tau^2}\sum_{j=1}^{p}\beta_j^2\right). \end{aligned}$$

By performing maximum a posteriori probability (MAP) estimation, we get the posterior mode of $\boldsymbol{\beta}$. In this case, the MAP estimate is similar to the maximum likelihood estimate, and we get the Ridge regression solution

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{\text{MAP}}^{\text{Ridge}} &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\left\{\frac{1}{2\sigma^2}\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\frac{1}{2\tau^2}\sum_{j=1}^{p}\beta_j^2\right\} \\ &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\left\{\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda\sum_{j=1}^{p}\beta_j^2\right\}. \end{aligned} \tag{5.14}$$

The expression in (5.14) is multiplied with $2\sigma^2$ to get the Ridge solution, and $\lambda = \frac{\sigma^2}{\tau^2}$.

5.5.2 The Lasso

If we use a laplacian prior, that is, if g has a Laplace distribution, the posterior mode of $\boldsymbol{\beta}$ will yield the Lasso solution.

The prior is given by

$$\begin{aligned} p(\boldsymbol{\beta}) &= \prod_{j=1}^{p}\text{Lap}\left(\beta_j\middle|0,\frac{1}{\lambda}\right) \propto \prod_{j=1}^{p}\exp\left(\frac{1}{(1/\lambda)}|0-\beta_j|\right) \\ &\propto \prod_{j=1}^{p}\exp\left(-\lambda|\beta_j|\right) = \exp\left(-\lambda\sum_{j=1}^{p}|\beta_j|\right), \end{aligned}$$

and the posterior is hence

$$p(\boldsymbol{\beta}|\boldsymbol{X},\boldsymbol{y}) \propto \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2-\lambda\sum_{j=1}^{p}|\beta_j|\right).$$

Using MAP-estimation we obtain the Lasso solution

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{\text{MAP}}^{\text{Lasso}} &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\left\{\frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j)^2+\lambda\sum_{j=1}^{p}|\beta_j|\right\} \\ &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\left\{\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda'\sum_{j=1}^{p}|\beta_j|\right\}, \end{aligned}$$

with $\lambda' = 2\sigma^2\lambda$, when the expression is multiplied with $2\sigma^2$.

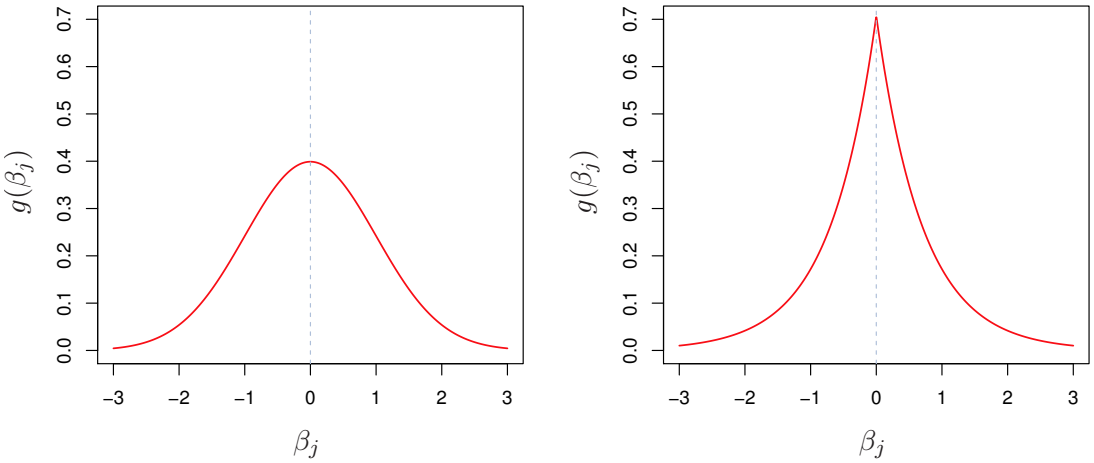


Figure 5.6: A gaussian prior (left) can be used to obtain the Ridge regression solution as the posterior mode, whereas a laplacian prior (right) can be used to obtain the Lasso solution as the posterior mode [James et al., 2013, Fig. 6.11].

The two different priors can be seen in Figure 5.6, where the Lasso prior is not differentiable when the coefficient estimates are equal to zero, since the peak of the function is more like a spike compared to the prior of Ridge regression which has the famous bell shape. Hence the Lasso expects a priori that more of the coefficients will be equal to zero.

5.6 Stability Selection

When performing variable selection as is the case with the Lasso and Elastic net method, it is of great interest to determine the consistency of the selected variables. A method called Stability selection is capable of doing this. The method was first described by [Meinshausen and Bühlmann, 2010]. Assume a tuning parameter $\lambda \in \Lambda \subseteq \mathbb{R}^+$ as for the Lasso, which controls the regularization of the model. For each $\lambda \in \Lambda$ we get an estimate with a subset $\hat{S}^\lambda \subseteq \{1,\dots,p\}$ of non-zero β-coefficients, also known as the selected set. The aim is then to determine how to choose λ to obtain the right amount of regularization, such that $\hat{S}^\lambda$ is identical to the true set S of relevant non-zero β-coefficients with a high probability [Meinshausen and Bühlmann, 2010].

To determine this consistency of selected variables, a concept called stability path needs to be introduced. It is defined in Definition 5.6.1 to be, when randomly resampling from the data, the probability of selecting each variable. In this section the number of observations will be denoted by n in stead of N.

Definition 5.6.1 (selection probabilities)**.**
Let I be a random subsample of $\{1,\dots,n\}$ of size $\lfloor n/2 \rfloor$, drawn without replacement. For

every set $K \subseteq \{1,\dots,p\}$, the probability of being in the selected set $\hat{S}^{\lambda}(I)$ is

$$\hat{\Pi}_K^{\lambda} = P^*\{K \subseteq \hat{S}^{\lambda}(I)\}.$$

The probability P^* is with regard to the subsampling being random, and if the algorithm for finding $\hat{S}^{\lambda}$ contains other sources of randomness.

The procedure in Stability selection is that a shrinkage model like the Lasso is fitted to many subsamples of the data, where each fit gives a resulting selected set $\hat{S}^{\lambda}(I)$. The consistent or stable variables are then the ones occurring in a great part of the selected sets, see Definition 5.6.2.

Definition 5.6.2 (stable variables)**.**
For a cut-off π_{thr} with $0 < \pi_{thr} < 1$ and a set of regularization parameters Λ, the set of stable variables is defined as

$$\hat{S}^{stable} = \{k : \max_{\lambda \in \Lambda} (\hat{\Pi}_k^{\lambda}) \geq \pi_{thr}\}.$$

The selection probabilities make up the stability path, and it is hence only the variables with a high selection probability, higher than a certain threshold, which are selected as the stable variables.

It is desired to obtain the true set S, and that carry to select a set without as many noise variables, denoted by N, as possible. In that connection the regularization parameter plays an important role. As an advantage in Stability selection it is stated in [Meinshausen and Bühlmann, 2010], that the initial set of regularization parameters in a reasonable range, will not be very essential for the result. Moreover another advantage is that exact error control is possible [Meinshausen and Bühlmann, 2010].

Definition 5.6.3 (additional notation)**.**
Let $\hat{S}^{\Lambda} = \cup_{\lambda \in \Lambda} \hat{S}^{\Lambda}$ be the set of selected structures or variables if varying the regularization λ in the set Λ. Let q_{Λ} be the average number of selected variables, $q_{\Lambda} = E(|\hat{S}^{\Lambda}|)$. Define V to be the number of falsely selected variables with stability selection, $V = |N \cap \hat{S}^{Stable}|$.

It is only possible to get exact error control i.e. to estimate $\mathrm{E}(V)$ if some simple assumptions are made.

Theorem 5.6.1 (error control)**.**
Assume that the distribution of $\{\mathbf{1}_{\{k \in \hat{S}^{\Lambda}\}}, k \in N\}$ is exchangeable for all $\lambda \in \Lambda$. Also, assume

that the original procedure is not worse than random guessing,

$$\frac{E(|S \cap \hat{S}^{\Lambda}|)}{E(|N \cap \hat{S}^{\Lambda}|)} \geq \frac{|S|}{|N|}. \tag{5.15}$$

The expected number of V of falsely selected variables is then bounded for $\pi_{thr} \in (\frac{1}{2}, 1)$ by

$$E(V) \leq \frac{1}{2\pi_{thr} - 1} \frac{q_{\Lambda}^2}{p}.$$

Proof.
The idea is first to show that $P(k \in \hat{S}^{\Lambda}) \leq \frac{q_{\Lambda}}{p}$ for all $k \in N$, and then use Lemmas A.1.1 and A.1.2 (in section A.1, page 77) to complete the proof.

Let $\hat{S}^{\Lambda} = \cup_{\lambda \in \Lambda} \hat{S}^{\lambda}$ and $q_{\Lambda} = \mathrm{E}(|\hat{S}^{\Lambda}|)$. Furthermore let $N_{\Lambda} = N \cap \hat{S}^{\Lambda}$, the set of noise variables in the set $\hat{S}^{\Lambda}$ and $U_{\Lambda} = S \cap \hat{S}^{\Lambda}$ the set of true variables in $\hat{S}^{\Lambda}$. We can then write the expected number of falsely selected variables as

$$\mathrm{E}(|N_{\Lambda}|) = \mathrm{E}(|\hat{S}^{\Lambda}|) - \mathrm{E}(|U_{\Lambda}|) = q_{\Lambda} - \mathrm{E}(|U_{\Lambda}|). \tag{5.16}$$

From the assumption in (5.15), we get

$$\frac{\mathrm{E}(|U_{\Lambda}|)}{\mathrm{E}(|N_{\Lambda}|)} \geq \frac{|S|}{|N|} \Leftrightarrow \mathrm{E}(|U_{\Lambda}|) \geq \mathrm{E}(|N_{\Lambda}|) \frac{|S|}{|N|}.$$

Combining this with (5.16) yields

$$\mathrm{E}(|U_{\Lambda}|) = q_{\Lambda} - \mathrm{E}(|N_{\Lambda}|) \geq \mathrm{E}(|N_{\Lambda}|) \frac{|S|}{|N|}.$$

Therefore we have that q_{Λ} is bounded below by

$$q_{\Lambda} \geq \mathrm{E}(|N_{\Lambda}|) \left(\frac{|S|}{|N|} + 1 \right).$$

The overall number of variables is p, which includes the number of noise variables and the true variables, hence $p = |N| + |S|$ and we get

$$q_{\Lambda} \geq \mathrm{E}(|N_{\Lambda}|) \left(\frac{|S|}{|N|} + \frac{|N|}{|N|} \right) = \mathrm{E}(|N_{\Lambda}|) \frac{p}{|N|},$$

which is equivalent to

$$\frac{\mathrm{E}(|N_{\Lambda}|)}{|N|} \leq \frac{q_{\Lambda}}{p}.$$

We have that $\mathrm{E}\left[\mathbf{1}_{\{k \in \hat{S}^{\lambda}\}}, k \in N\right] = P\left(\mathbf{1}_{\{k \in \hat{S}^{\lambda}\}} = 1, k \in N\right) = P\left(k \in \hat{S}^{\Lambda}\right)$. Due to the exchangeability assumption we end up with a distribution closely related to the binomial distribution and hence we get $P\left(k \in \hat{S}^{\Lambda}\right) = \frac{\mathrm{E}(|N_{\Lambda}|)}{|N|}$ for all $k \in N$. It then holds that $P\left(k \in \hat{S}^{\Lambda}\right) \leq \frac{q_{\Lambda}}{p}$ for all $k \in N$, which was what we wanted to show.

From Lemma A.1.2 we can write

$$P\left[\max_{\lambda\in\Lambda}\left(\hat{\Pi}_k^{\text{simult},\lambda}\right)\geq\xi\right]\leq\frac{(q_\Lambda/p)^2}{\xi},$$

for all $0<\xi<1$ and $k\in N$.

Using Lemma A.1.1 we get the following

$$\begin{aligned}&\max_{\lambda\in\Lambda}\left(\hat{\Pi}_K^{\text{simult},\lambda}\right)\geq 2\max_{\lambda\in\Lambda}\left(\hat{\Pi}_K^{\lambda}\right)-1\Leftrightarrow\\&\max_{\lambda\in\Lambda}\left(\hat{\Pi}_K^{\lambda}\right)\leq\frac{\max_{\lambda\in\Lambda}\left(\hat{\Pi}_K^{\text{simult},\lambda}\right)+1}{2}.\end{aligned}$$

Let $\xi=\pi_{\text{thr}}$, we may then have

$$P\left[\max_{\lambda\in\Lambda}\left(\hat{\Pi}_k^{\lambda}\right)\geq\pi_{\text{thr}}\right]\leq P\left[\frac{\left\{\max_{\lambda\in\Lambda}\left(\hat{\Pi}^{\text{simult},\lambda}\right)+1\right\}}{2}\geq\pi_{\text{thr}}\right]\leq\frac{(q_\Lambda/p)^2}{2\pi_{\text{thr}}-1}.$$

The expected number of falsely selected variables is then bounded above by

$$\mathrm{E}(V)=\sum_{k\in N}P\left[\max_{\lambda\in\Lambda}\left(\hat{\Pi}_k^{\lambda}\right)\geq\pi_{\text{thr}}\right]\leq\sum_{k\in N}\frac{1}{2\pi_{\text{thr}}-1}\frac{q_\Lambda^2}{p^2}=p\frac{1}{2\pi_{\text{thr}}-1}\frac{q_\Lambda^2}{p^2}=\frac{1}{2\pi_{\text{thr}}-1}\frac{q_\Lambda^2}{p}.$$

□

The expected number of falsely selected variables is also named the per-family-error rate (`pfer`), used in the following examples. The error controlled in this way, will then correspond to the expected number of type 1 errors, that is the probability that at least one variable has been falsely selected in the set $\hat{S}^{\text{stable}}$ [Sill et al., 2014].

5.6.1 Example 1

In the following an example will illustrate the method of Stability selection by the Lasso. The data used is randomly generated from the normal distribution with mean zero and standard deviation one.

```
set.seed(3)
x <- matrix(rnorm(80*500,0,1),80,500)
y <- x[1:80,1:500]%*% c(rep(3,2),rep(-3,3),rep(.1,495))
```

The data consists of 80 observations with 500 features, and the response `y` is constructed such that the last 495 features should be less important than the first 5 features. The functions used in `R` to perform Stability selection is to be found in the library `c060`. The stability path of the data is constructed by the function `stabpath()`, which is applied to the Lasso path as default. The actual stability selection is performed by `stabsel()`, with a type I error level `err=0.05` of the type per-family-error rate (`pfer`). The output is shown below, where the variables `V2`,`V3`,`V4` and `V5` are selected as stable.

```
res <- stabpath(y,x)
sel <- stabsel(res,error=0.05,type="pfer",pi_thr=0.6)
sel
$stable
V2 V3 V4 V5
 2  3  4  5

$lambda
[1] 3.429283

$lpos
[1] 7

$error
[1] 0.05

$type
[1] "pfer"
```

In Figure 5.7 the penalization path and the stability path are plotted, where four variables marked with red have been selected as stable. Notice, as expected it is four of the first five variables which have been selected as stable. The fact that `V1` has not been selected as stable, corresponds to a false negative, that is the variable has falsely not been detected as stable. This is a consequence of the method accounting for false positives and not false negatives.

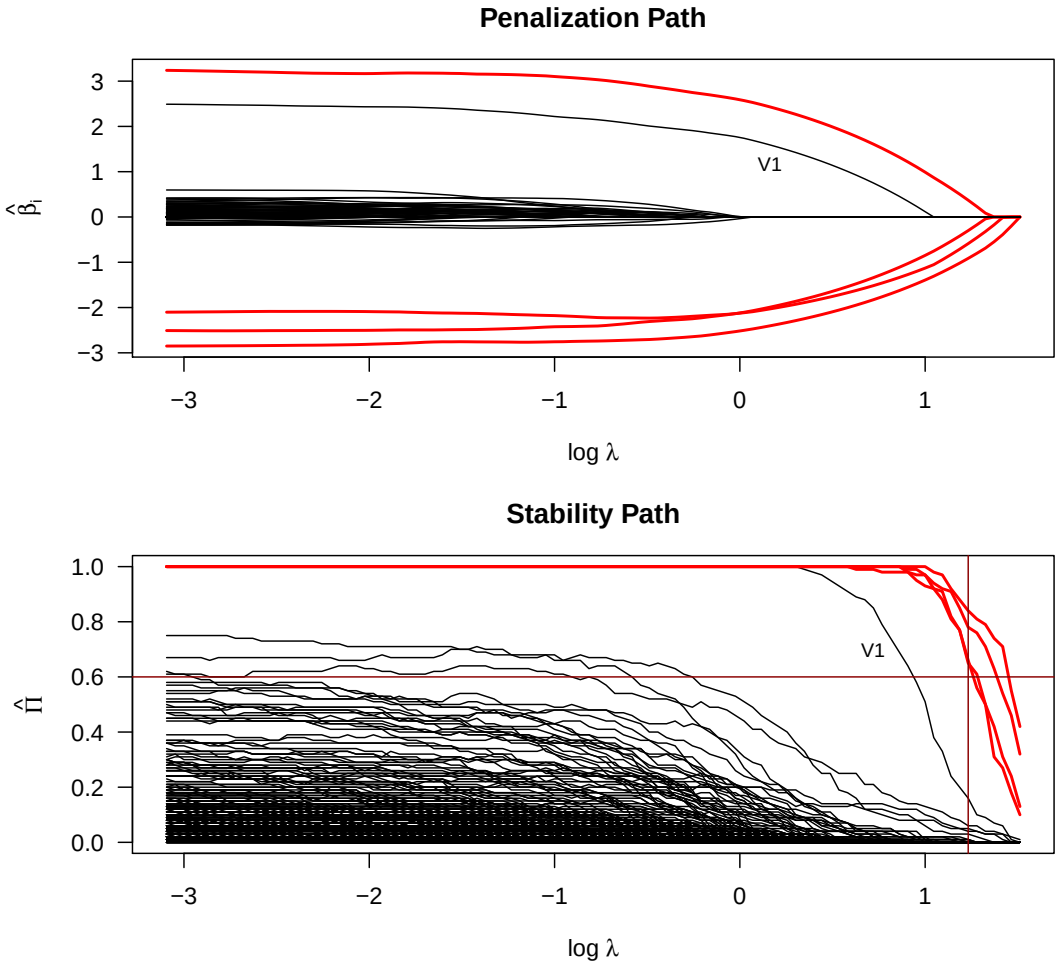


Figure 5.7: Penalization- and stability path, where the selected stable variables are marked with red. The logarithm of the chosen lambda value is the vertical red line, whereas the horizontal red line represents the value of π_{thr}.

5.6.2 Example 2

When collinearity or multicollinearity exists in a data set, the Lasso method will as the penalty parameter λ increases only select the variables which are most correlated with the response, in a group of multicollinear variables. This property is not desirable if we are interested in all of the relevant variables, and we can in stead make use of Elastic net, which selects variables in groups by their multicollinearity. The following example will demonstrate how Stability selection perform when variables are collinear, using Lasso and Elastic net.

The data is simulated from a multivariate normal distribution, with a mean vector of zeroes. The covariance matrix is constructed as the identity matrix, except for the entries of the variables which we want to be collinear and multicollinear. As shown in the output below, we construct the covariance matrix such that `x1`, `x2` and `x3` forms a group of multicollinear variables, and `x7` and `x8` are collinear. The response is constructed as a weighted sum of the 500 variables, where a higher weight is on the first eight variables.

```
> p=500
> n=80
```

```
# The covariance matrix
> Sigma=diag(p)
> Sigma[1,2] <- Sigma[2,3] <- Sigma[1,3] <- Sigma[7,8]  <- 0.8
> Sigma[2,1] <- Sigma[3,2] <- Sigma[3,1] <- Sigma[8,7] <- 0.8

# Data is simulated
> set.seed(211)
> x <- rmvnorm(n=n, mean=rep(0,p),sigma=Sigma)

# The response is constructed
> y <- x\%*\%c(rep(1,8),rep(.1,492))
```

The following output shows how correlated the first eight relevant variables are with the response, and the high correlations between the collinear variables are also given in the correlation matrix of these.

```
# Correlation between the response and relevant variables
> cor(y,x[,1:8])
         x1        x2        x3        x4        x5        x6        x7        x8
y 0.5390047 0.4452463 0.5594276 0.1694115 0.230595 0.3955168 0.2510914 0.3437712

# Correlation between collinear variables
> cor(x[,c(1:3,7:8)])
           x1         x2         x3         x7         x8
x1  1.0000000  0.7688409  0.8016038 -0.2426516 -0.2290382
x2  0.7688409  1.0000000  0.7773380 -0.2713801 -0.2581502
x3  0.8016038  0.7773380  1.0000000 -0.2782402 -0.2562329
x7 -0.2426516 -0.2713801 -0.2782402  1.0000000  0.8751749
x8 -0.2290382 -0.2581502 -0.2562329  0.8751749  1.0000000
```

Stability paths are computed with tuning parameter $\alpha = 1$ corresponding to the Lasso method, and with $\alpha = 0.1$ for the Elastic net method. The paths are plotted in Figure 5.8, where the Stability selection with per-family-error-rate equal to 0.05 is shown by the black vertical line indicating the value of $\log(\lambda)$ for the selected variables x1 and x3, and the black horizontal line indicating the threshold at 0.6 for the selection probabilities.

It is seen that for the Lasso method, the collinear variables with the lowest correlation to the response x2 and x7, have a low selection probability, and will not be considered as stable variables as they never passes the line of π_{thr}, even though they are relevant and higher correlated with the response compared to other variables. When using the Lasso, the factor influencing the stability path, is the individual subsamples. If some observations have an influence on the correlation to the response, it will not be consistent which variable in a group of multicollinear variables the method will select, and hence the selection probability will be lowered, causing that some relevant variables will not be detected as stable.

Setting $\alpha = 0.1$ results in an increase of the selection probability for the collinear and multicollinear variables, due to the property that Elastic net selects variables in groups by their collinearity. The fact that only x1 and x3 are selected as stable, might be due to their higher correlation to the response, compared to the six other relevant variables, and thereby they might be selected by Lasso or Elastic net more often than the others. Even though Elastic net has the property of selecting variables in groups, it is a compromise between Lasso and Ridge regression, and the part from the Lasso will shrink most of the regression coefficients to zero. However, none of the eight relevant variables have a considerably high correlation to the response, and as seen some of the 492 irrelevant

variables in Figure 5.8 have selection probabilities just as high as some of the relevant ones. A clearer difference between relevant and irrelevant variables, as their would be in real data, with some variables being truly associated with the response, would cause a more clear distinction of stable variables.

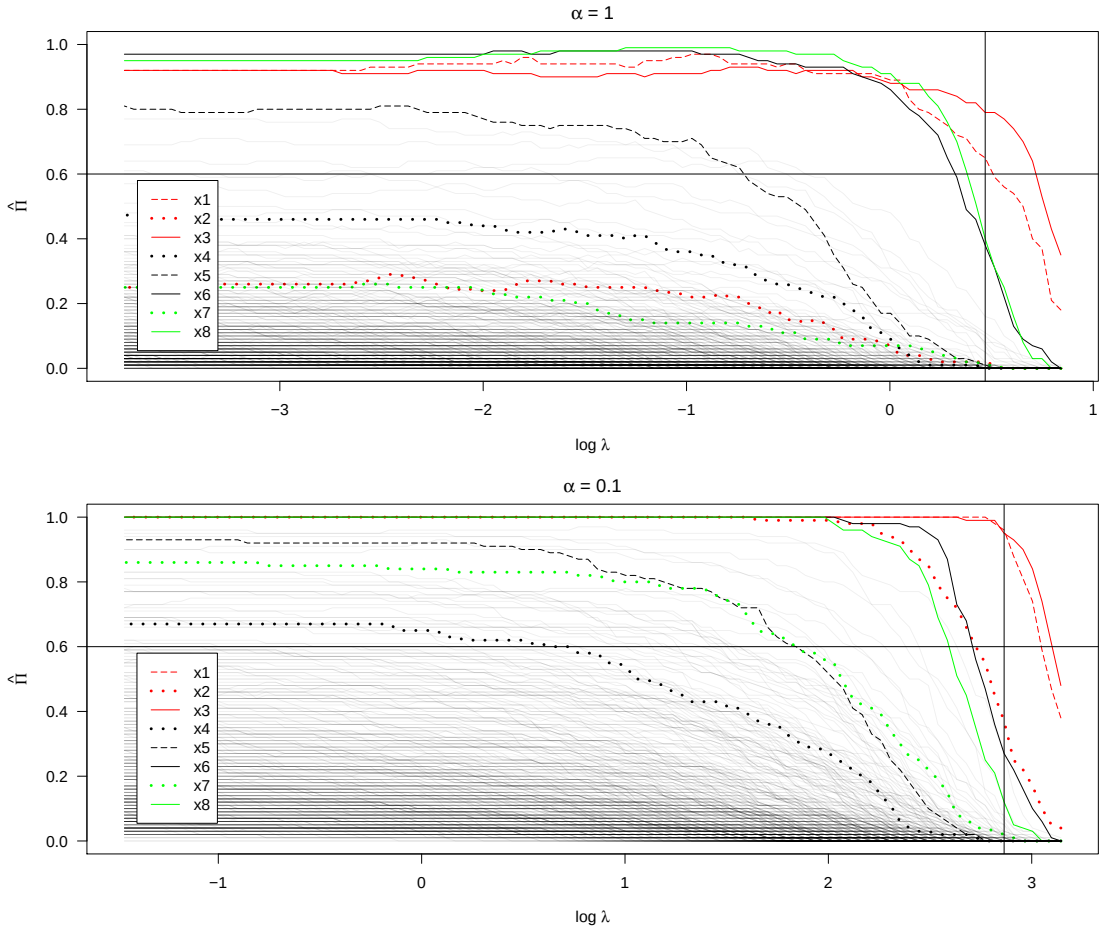


Figure 5.8: Stability paths of Lasso ($\alpha = 1$) and Elastic net ($\alpha = 0.1$), where some of the variables are collinear and multicollinear. The horizontal black line marks π_{thr}, and the vertical line marks the value of $\log(\lambda)$ for the selected variables x1 and x3. The faded lines in the background are the 492 irrelevant variables.

6 Partial Least Squares

Partial least squares (PLS) is another useful technique for high dimensional regression problems. It transforms the predictors and uses those in stead of the original predictors to fit a least squares model. The method is appropriate for high dimensional data, since one of the advantages is that a problem can be reduced to a lower dimension, by using fewer transformed variables in stead of all the original predictors. The theory is from [Garthwaite, 1994], unless otherwise stated.

The partial least squares regression equation is given by

$$\hat{\boldsymbol{y}} = \beta_0 + \beta_1 \boldsymbol{t}_1 + \beta_2 \boldsymbol{t}_2 + \ldots + \beta_m \boldsymbol{t}_m, \tag{6.1}$$

where the transformed variables are $\boldsymbol{t}_k$ for $k = 1, \ldots, m$, they will also be referred to as the PLS-components in the following. The response and predictors are denoted in their centered form by $\boldsymbol{u}_1 = \boldsymbol{y} - \bar{y}$ and $\boldsymbol{v}_{1j} = \boldsymbol{x}_j - \bar{x}_j$, for $j = 1, \ldots, p$. The index of $\boldsymbol{u}_1$ and the first index of $\boldsymbol{v}_{1j}$ is due to PLS being an iterative procedure.

The first transformed variable $\boldsymbol{t}_1$ is a linear combination of $\boldsymbol{v}_{1j}$, it is obtained by regressing $\boldsymbol{u}_1$ on each of the j predictors $\boldsymbol{v}_{1j}$ in turn. Since the sample means are zero due to the centering, we end up with the regression equation

$$\hat{\boldsymbol{u}}_{1j} = b_{1j} \boldsymbol{v}_{1j}, \tag{6.2}$$

where solving for the least squares solution yields $b_{1j} = \boldsymbol{v}_{1j}^T \boldsymbol{u}_1 / \boldsymbol{v}_{1j}^T \boldsymbol{v}_{1j}$. As we would like an estimate of $\boldsymbol{u}_1$, and $\hat{\boldsymbol{u}}_{1j}$ for $j = 1, \ldots, p$ give estimates of $\boldsymbol{u}_1$, an average of those would be reasonable, hence the first PLS-component $\boldsymbol{t}_1$ is computed as the weighted average of (6.2)

$$\boldsymbol{t}_1 = \sum_{j=1}^{p} w_{1j} b_{1j} \boldsymbol{v}_{1j}, \qquad \sum_{j=1}^{p} w_{1j} = 1.$$

Since $\boldsymbol{t}_1$ is an average, there might still be some useful information left, which is not explained by $\boldsymbol{t}_1$. This information can possibly be obtained for the predictors by the residuals from regression of $\boldsymbol{v}_{1j}$ on $\boldsymbol{t}_1$, and for the response by the residuals from regressing $\boldsymbol{u}_1$ on $\boldsymbol{t}_1$. The residuals are named $\boldsymbol{v}_{2j}$ and $\boldsymbol{u}_2$ respectively, and the next PLS-component $\boldsymbol{t}_2$ is hence computed in the same way as $\boldsymbol{t}_1$, though using $\boldsymbol{v}_{2j}$ and $\boldsymbol{u}_2$ in stead of $\boldsymbol{v}_{1j}$ and $\boldsymbol{u}_1$. This procedure continues, and the residuals are in general computed as follows

$$\boldsymbol{v}_{i+1,j} = \boldsymbol{v}_{ij} - \frac{\boldsymbol{t}_i^T \boldsymbol{v}_{ij}}{\boldsymbol{t}_i^T \boldsymbol{t}_i} \boldsymbol{t}_i,$$

and

$$\boldsymbol{u}_{i+1} = \boldsymbol{u}_i - \frac{\boldsymbol{t}_i^T \boldsymbol{u}_i}{\boldsymbol{t}_i^T \boldsymbol{t}_i} \boldsymbol{t}_i.$$

The PLS-components are then

$$\boldsymbol{t}_{i+1} = \sum_{j=1}^{p} w_{i+1,j} b_{i+1,j} \boldsymbol{v}_{i+1,j}, \qquad \text{where } b_{i+1,j} = \frac{\boldsymbol{v}_{i+1,j}^T \boldsymbol{u}_{i+1}}{\boldsymbol{v}_{i+1,j}^T \boldsymbol{v}_{i+1,j}}. \tag{6.3}$$

These PLS-components are used in stead of the original predictors $\boldsymbol{x}_j$ in (6.1) to fit a linear model by the least squares method. The choice of m, the number of components $\boldsymbol{t}_i$ to use in the model, is typically found by cross-validation.

The weights w_{ij} are for all i, j computed as $w_{ij} = \boldsymbol{v}_{ij}^T \boldsymbol{v}_{ij}$. This is approximately equal to the sample covariance of $\boldsymbol{v}_{ij}$ which is $\boldsymbol{v}_{ij}^T \boldsymbol{v}_{ij}/n - 1$. Inserting the value of w_{ij} into (6.3) gives

$$\boldsymbol{t}_i = \sum_{j=1}^{p} \boldsymbol{v}_{ij}^T \boldsymbol{v}_{ij} \frac{\boldsymbol{v}_{ij}^T \boldsymbol{u}_i}{\boldsymbol{v}_{ij}^T \boldsymbol{v}_{ij}} \boldsymbol{v}_{ij} = \sum_{j=1}^{p} (\boldsymbol{v}_{ij}^T \boldsymbol{u}_i) \boldsymbol{v}_{ij}. \tag{6.4}$$

The term $\boldsymbol{v}_{ij}^T \boldsymbol{u}_i$ can be shown to be proportional to the correlation between the two variables [James et al., 2013], and hence the centered predictors or residuals with the highest correlation to the centered response or residual will contribute the most to the construction of the particular transformed variable.

An advantage of the method is that a regressor is uncorrelated with the residual, and $\boldsymbol{t}_i$ is thereby uncorrelated with $\boldsymbol{v}_{(i+1)j}$. Because of this, the components $\boldsymbol{t}_i$ in (6.1) are uncorrelated, and we would be able to fit a linear model by these, where collinearity is not a problem.

6.1 Example

The method of Partial least squares will in the following be performed using the exact same simulated data as used in the example of Ridge regression and the Lasso method (see subsection 5.1.4). The data consists of 100 features with 30 observations, and the response is constructed by some noise and the features `X1` and `X6`. The data is divided into 15 training - and 15 test-samples as in subsection 5.1.4. PLS is performed by the function `plsr()` in the `R`-package `pls`. The output of the fitted model to the training data is shown in Figure 6.1, where `data` is a matrix containing both the response and the features. The `validation` argument is a specification of whether the validation should be regular cross validation (`"CV"`), with default 10 folds, or leave one out cross validation (`"LOO"`).

Figure 6.1: Output of the Partial least squares model fitted to some simulated data.

```
> set.seed(4013)
> pls.fit=plsr(y~., data=data[train,], validation ="LOO")
> summary(pls.fit)
Data:   X dimension: 15 100
        Y dimension: 15 1
Fit method: kernelpls
Number of components considered: 13

VALIDATION: RMSEP
Cross-validated using 15 leave-one-out segments.
       (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
CV            1.98    1.799    1.726    1.710    1.711    1.714    1.714
adjCV         1.98    1.745    1.669    1.653    1.653    1.656    1.656
          7 comps  8 comps 9 comps  10 comps  11 comps  12 comps  13 comps
CV          1.714   1.714   1.714     1.714    1.714    1.714    1.714
adjCV       1.656   1.656   1.656     1.656    1.656    1.656    1.656

TRAINING: % variance explained
   1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
X     9.11    17.07    24.58    31.92    39.96    46.99    55.45
y    92.21    99.24    99.88    99.99   100.00   100.00   100.00
   8 comps  9 comps 10 comps 11 comps  12 comps  13 comps
X    61.58    69.04   74.12     82.75    88.97    94.43
y   100.00   100.00  100.00    100.00   100.00   100.00
```

The term `adjCV` is the adjusted cross validation error. It adjusts for bias associated with training the model on segments of the data, in stead of using all the data [Mevik and Cederkvist, 2004].

As seen the adjusted cross validation root mean squared error is lowest when using three or four PLS-components. A prediction of the test data with the model containing three PLS-components results in a mean squared error equal to 4.14, which is around the double of the one obtained by Ridge regression and Lasso, with an MSE equal to 2.19 and 2.07 respectively.

Fitting the model with three PLS-components to the full data set yields the explained variance in the response at 99.25% as seen in the output in Figure 6.2. Most of the information in the response seems though to be explained by the first PLS-component with an explained variance at 84.91, but the three PLS-components together does a reasonable

job of explaining the response.

Figure 6.2: Output of the PLS-model with three PLS-components fitted to the full data set.

```
> pls.full=plsr(y~., data=data, validation ="LOO",ncomp=3)
> summary(pls.full)
Data:   X dimension: 30 100
        Y dimension: 30 1
Fit method: kernelpls
Number of components considered: 3

VALIDATION: RMSEP
Cross-validated using 30 leave-one-out segments.
       (Intercept)  1 comps  2 comps  3 comps
CV           2.146    1.793    1.758    1.742
adjCV        2.146    1.772    1.734    1.714

TRAINING: % variance explained
   1 comps  2 comps  3 comps
X    5.835    11.08    14.90
y   84.906    95.51    99.25
```

The term $\boldsymbol{v}_{ij}^T \boldsymbol{u}_i$ in (6.4) is typically denoted as a loading. It tells how important a given feature is in the construction of a PLS-component. In Figure 6.3 the absolute value of the loadings of the features in the first PLS-component are sorted in decreasing order, and the two features with the highest and lowest values are shown. To illustrate their importance in the first PLS-component, their associated features are plotted against the first PLS-component in Figure 6.4.

Figure 6.3: The two highest and the two lowest loadings in the first PLS-component.

```
> loadings <- sort(abs(pls.full$loadings[,1]),decreasing=T)
> loadings[1:2]
       X1       X96
0.3266672 0.2425969
> loadings[99:100]
        X43         X46
0.006319321 0.000736374
```

As seen in Figure 6.4 feature `X1` and `X96` have a stronger relation to the first PLS-component, compared to the two features `X43` and `X46`, where almost no relation is indicated. As feature `X1` has been used in the construction of the response, it has as expected a high loading, and is thereby also highly associated with the first PLS-component. We would also expect `X6` to have a loading close to that of `X1`, since it also was used to construct the response. The reason why this is not the case, could be that all of the features are random simulated, and other features might accidentally have a higher association with the response due to the addition of noise to the response.

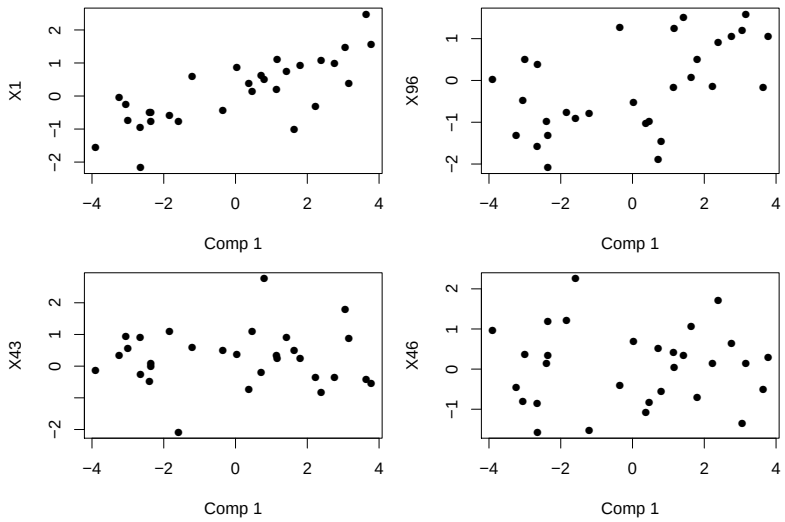


Figure 6.4: Relation between the first PLS-component and features X1,X96, X43 and X46.

The correlation matrix in Figure 6.5 shows that the PLS-components as expected are uncorrelated, with correlation coefficients effectively zero.

Figure 6.5: The correlation matrix of the three PLS-components.

```
> cor(pls.full$scores)
              Comp 1        Comp 2        Comp 3
Comp 1  1.000000e+00  1.243253e-16  7.836004e-17
Comp 2  1.243253e-16  1.000000e+00 -6.132732e-17
Comp 3  7.836004e-17 -6.132732e-17  1.000000e+00
```

7 Results

For analysis of the correlation between age and DNA methylation, 50 blood samples were available. Of these samples, there were replicates for three subjects, two subjects had two replicates and one subject had one. The five replicates were removed from the data used for the analysis, so that they could be used for later validation. The 45 samples left were divided into a training and a test data set, 29 samples were used for training and 16 samples were used as test data.

The training data were randomly sampled with 2/3 from each of the four age groups in Table 7.1, the rest was used as test data.

	Age groups			
	15-30	31-45	46-60	61-82
Subjects	8	18	8	11

Table 7.1: Number of subjects in four different age-groups.

In this chapter five different methods described in the previous chapters, will be applied to the data just mentioned, and the results will be presented. At first the three shrinkage models Ridge regression, Elastic net and Lasso will be fitted to the data with different values of the tuning parameter α. Afterward Stability selection will be applied to Elastic net models, with the purpose to obtain some consistent CpG-markers. To see if other methods does a better job of selecting relevant CpG-markers and predicting age, Partial least squares will also be applied to the data. At last prediction performance of achieved models from Stability selection will be tested on some simulated data with specific ages, constructed from knowledge of the observed data, based on Ridge regression to compensate for collinearity of predictors. The validation data will subsequently be used to asses the model performance.

7.1 Shrinkage Models

The training data was fitted to three different types of shrinkage-models, Ridge regression, Elastic net and Lasso with the `glmnet`-package in `R` by a grid of 20 α-values in the range $[0,1]$. Leave-one-out cross-validation was used on the 29 training samples to choose the best λ among 100 values of λ for each model. In Table 7.2 the prediction error on the test data can be seen as the root mean squared error (RMSE) together with the chosen λ and the number of CpG-markers for each model.

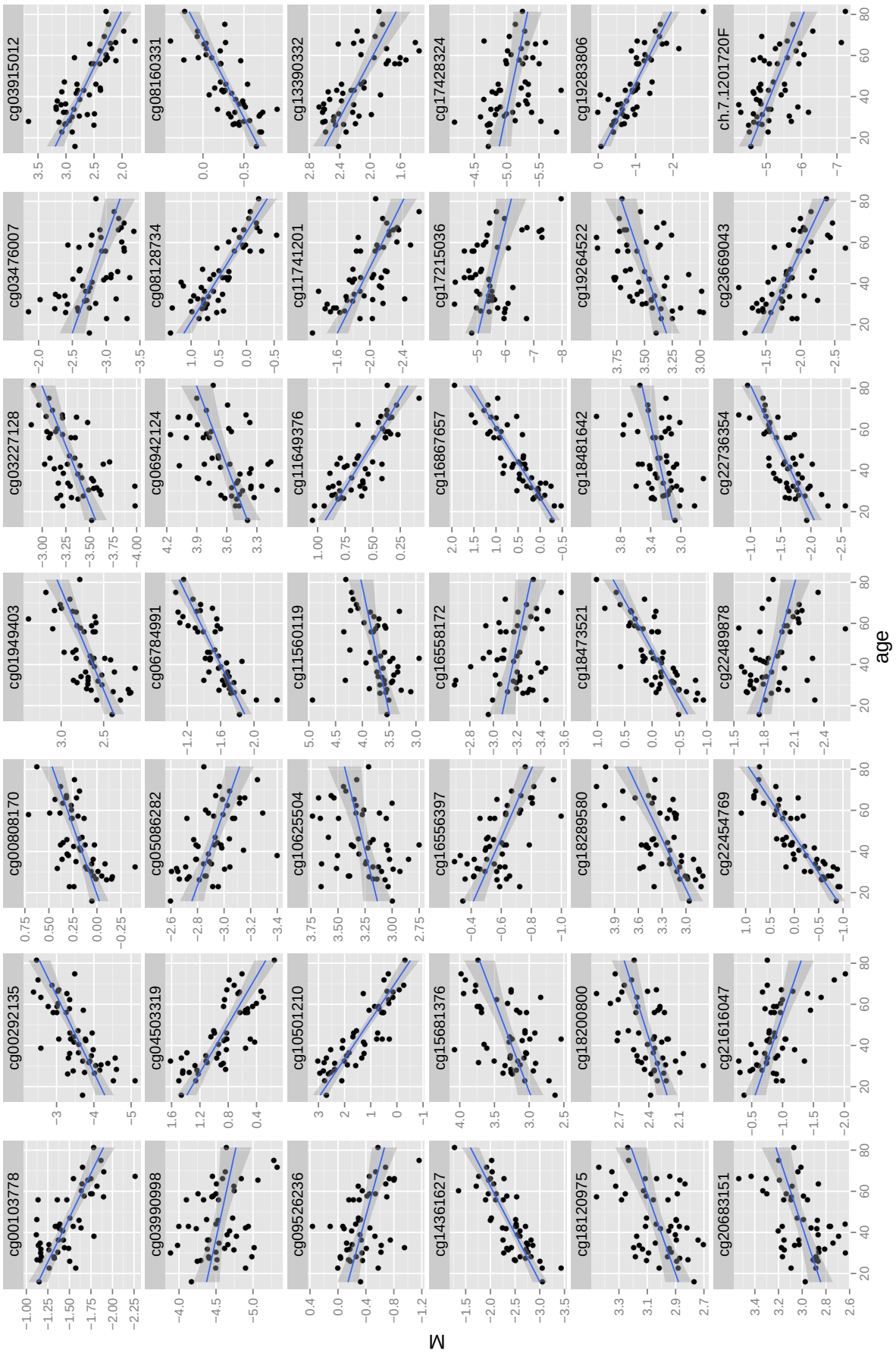
As seen in Table 7.2 the model with the lowest prediction error that performed most feature selection was the Elastic net model with $\alpha = 0.95$, it resulted in a model with 42 CpG-markers shown in Figure 7.1.

α	λ	CpG-markers	RMSE
0.00	13586.13	474280	0.26
0.05	5.96	378	0.57
0.11	3.12	221	0.59
0.16	2.08	158	0.58
0.21	1.64	130	0.60
0.26	1.31	111	0.60
0.32	1.09	98	0.60
0.37	0.93	84	0.59
0.42	0.82	78	0.57
0.47	0.73	68	0.57
0.53	0.95	62	0.82
0.58	0.72	60	0.67
0.63	1.05	53	1.07
0.68	0.88	49	0.98
0.74	0.49	51	0.58
0.79	0.63	50	0.80
0.84	0.43	45	0.58
0.89	0.77	41	1.12
0.95	0.67	42	1.02
1.00	0.83	42	1.37

Table 7.2: Results of Ridge regression, Elastic net and Lasso.

Section of Forensic Genetics wanted to limit down the number of CpG-markers to approximately $12 - 24$, since otherwise they would need to much DNA from a scene of crime, in order to consistently be able to estimate the age of the person of interest. Due to this limitation, further reduction of the CpG-markers was necessary. However, we were also interested in finding some reliable CpG-markers, and for this reason we would like to determine the randomness associated with the selected CpG-markers by Lasso and Elastic net. This was done by Stability selection.

Figure 7.1: The 42 CpG-markers selected by the Elastic net-model with $\alpha = 0.95$.



7.2 Stability Selection

As Section of Forensic Genetics wanted some reliable CpG-markers to predict the age of a suspect, it was very important that the chosen CpG-markers for a potential model were consistent, and thus had a predictive value for any given data set of DNA evidence.

We were interested in the best combination of CpG-markers with a high correlation to age, since some CpG-markers might explain something that the others did not. As we saw in the example in Section 5.6.2, Elastic net selects groups of predictors that are multicollinear, and Lasso would only select the predictor with the highest correlation to age in such a group. This causes instability in which predictors the Lasso method selects for each subsample in Stability selection, and only few if any will be selected as stable. It was experienced that only one CpG-marker was selected as stable when applying Stability selection to the Lasso method. To ensure that all relevant CpG-markers were considered, the method of Stability selection was applied on 20 Elastic net-models where α was varied in the range $[0.01; 0.99]$. Notice that these α's are different from the ones in Table 7.2, as the methods of Ridge regression and Lasso are excluded. The error was the "per-family-error-rate" and the level was 0.05, where $\pi_{thr} = 0.6$.

The union of the consistent CpG-markers selected from the 20 different α-models yielded a set of 32 CpG-markers. They are plotted against age in Figure 7.3. Figure 7.4 provides an overview of which markers appeared as consistent to each of the 20 values of α.

It seemed reasonable for prediction of age to consider a model with a combination of CpG-markers corresponding to the result of Stability selection for one of the 20 α-values, as it was 20 suggestions of stable CpG-markers related to age, with different levels, corresponding to α, of the number of multicollinear CpG-markers.

To estimate the prediction error of the models, a linear model would not be suitable, due to the multicollinearity between the CpG-markers. As an example, the five consistent markers chosen by Stability selection with $\alpha = 0.63$ are plotted against each other in Figure 7.2. They show a clear linear trend, and together with the correlation coefficients given in the correlation matrix in Table 7.3 with values around 0.7 and above it indicates collinearity.

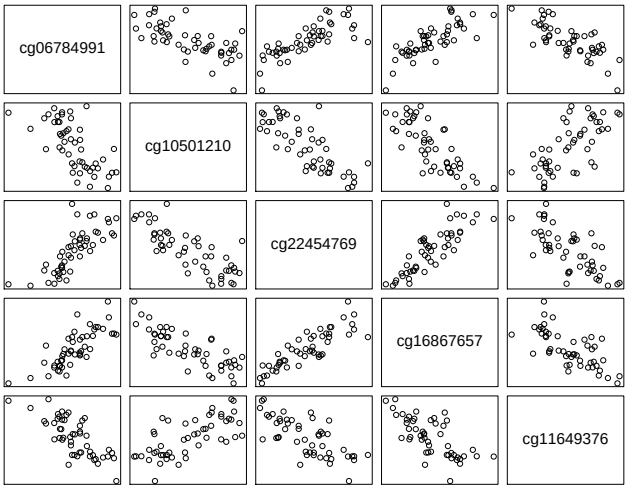


Figure 7.2: The linear relationship between CpG-markers chosen by stability selection with $\alpha = 0.63$.

Figure 7.3: The 32 consistent markers found by stability selection for the different values of α.

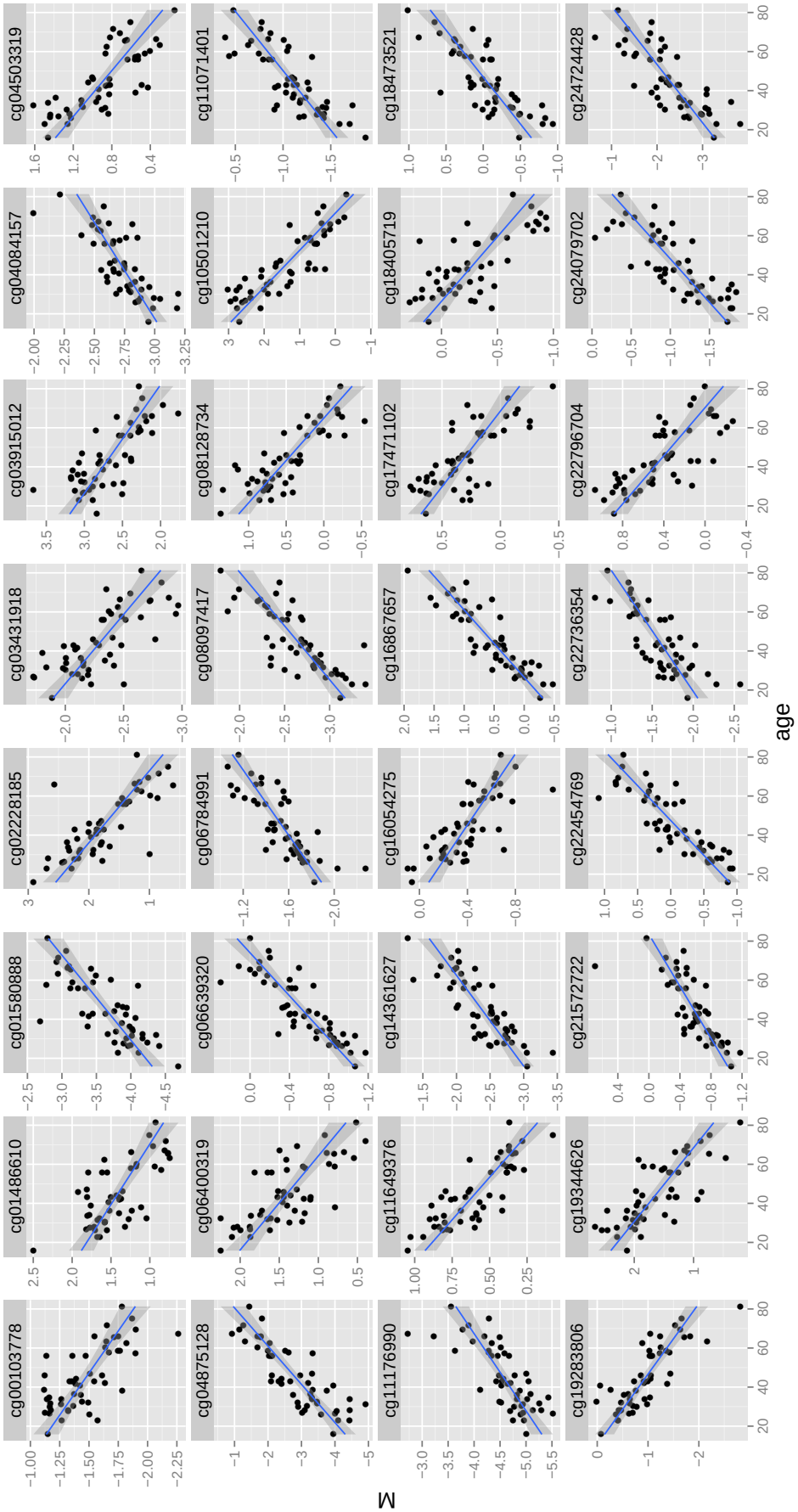
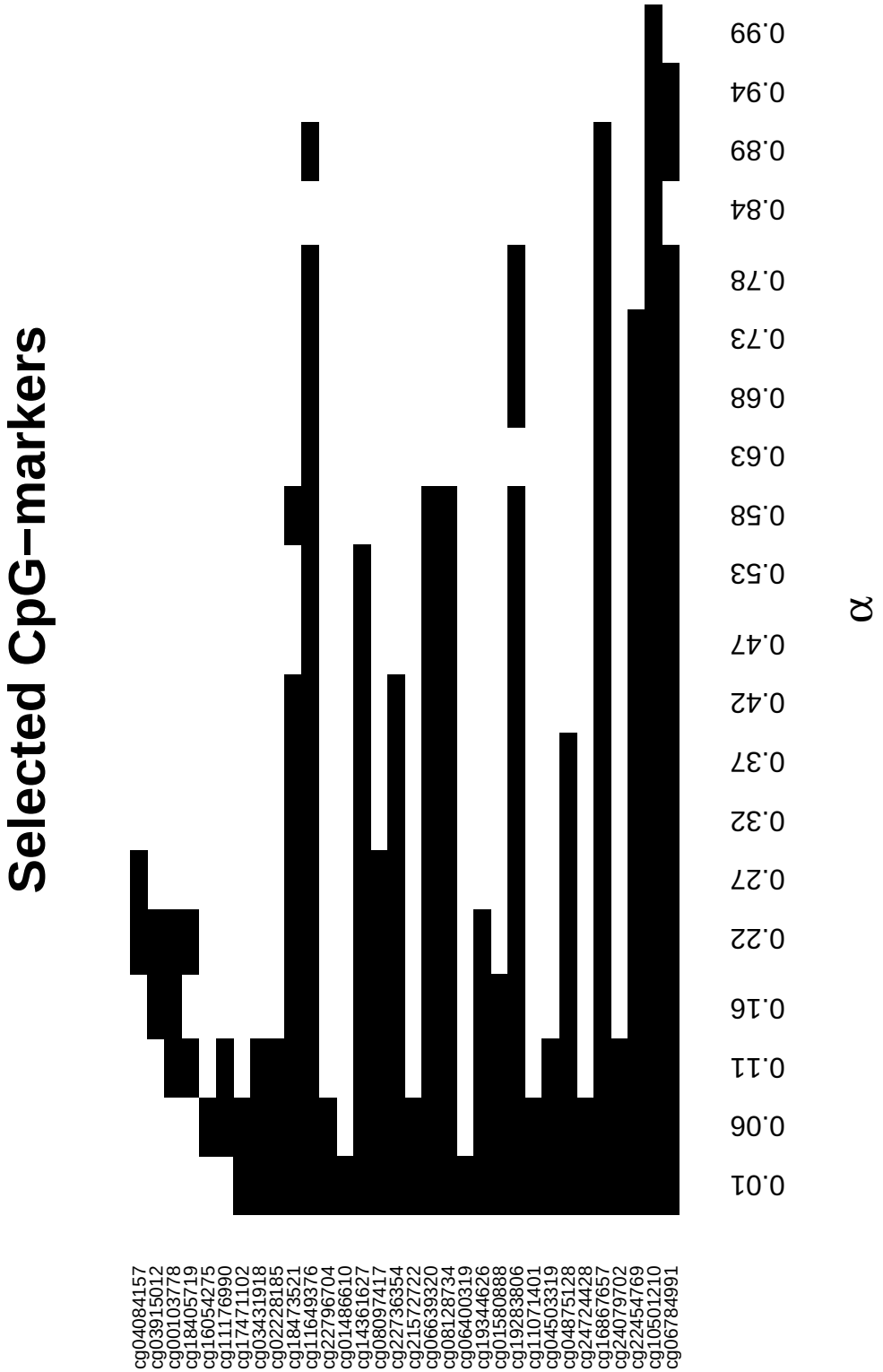


Figure 7.4: Presence of the 32 CpG-markers at the different α-values in Stability selection.



```
> cor(M.noXY[,res[[13]]])
           cg06784991 cg10501210 cg22454769 cg16867657 cg11649376
cg06784991  1.0000000 -0.6766791  0.7747738  0.7875159 -0.7554610
cg10501210 -0.6766791  1.0000000 -0.7961520 -0.7954466  0.6971878
cg22454769  0.7747738 -0.7961520  1.0000000  0.8580314 -0.7198260
cg16867657  0.7875159 -0.7954466  0.8580314  1.0000000 -0.7026890
cg11649376 -0.7554610  0.6971878 -0.7198260 -0.7026890  1.0000000
```

Table 7.3: The correlation matrix for the CpG-markers (contained in `res[[13]]`) chosen by stability selection with $\alpha = 0.63$.

Ridge regression accommodated the problem of multicollinearity without performing any variable selection, and hence the 20 models found by Stability selection was fitted by Ridge regression to estimate the prediction error on the test data. A boxplot of the prediction errors for the different models corresponding to the 20 α-values can be seen in Figure 7.5. Furthermore Figure 7.6 illustrates a path of the estimated Ridge regression coefficients for each of the 32 CpG-markers plotted against their corresponding value of α.

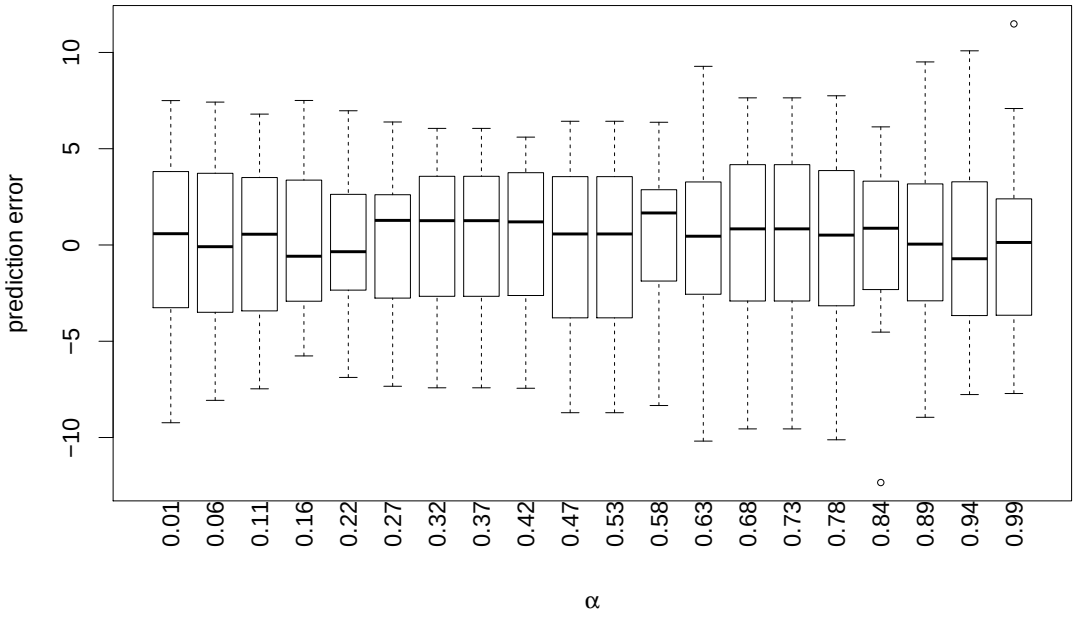


Figure 7.5: Distribution of prediction errors with the test data when Ridge regression was performed on the models found by Stability selection.

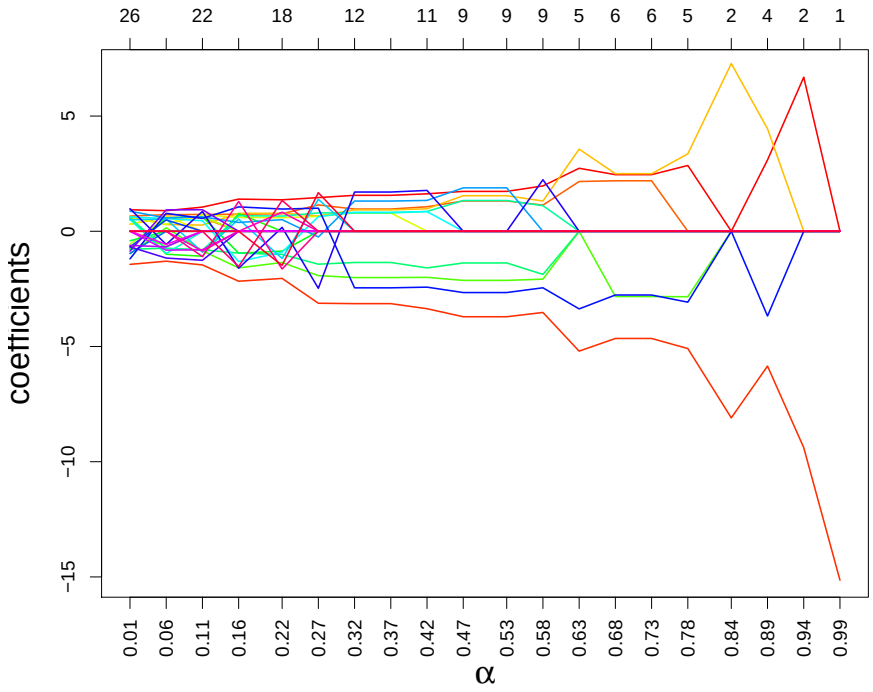


Figure 7.6: Ridge regression coefficient path for the different values of α. Each line segment refers to the CpG-markers in Figure 7.4, where e.g. cg10501210 is the lower red line segment.

The root mean squared error for each model is given in Table 7.4, where the model ID refers to the α-value from Stability selection. The lowest RMSE was obtained for the model with model ID $\alpha = 0.22$, this model contained 18 CpG-markers, which can be seen in Figure 7.8.

Model ID (α)	CpG-markers	RMSE	λ
0.01	26	4.380	14.315
0.06	26	4.369	17.242
0.11	22	4.046	14.315
0.16	17	3.924	6.663
0.22	18	3.751	7.313
0.27	14	3.984	3.813
0.32	12	4.145	4.185
0.37	12	4.145	4.185
0.42	11	4.147	3.813
0.47	9	4.367	2.884
0.53	9	4.367	2.884
0.58	9	4.412	3.813
0.63	5	4.910	1.988
0.68	6	5.044	2.182
0.73	6	5.044	2.182
0.78	5	4.971	2.395
0.84	2	4.535	1.504
0.89	4	4.934	1.504
0.94	2	5.194	1.462
0.99	1	4.760	0.000

Table 7.4: Results of Ridge regression on the 20 models, where the model ID is the α-values from Stability selection.

In Figure 7.7 it was determined if there was any trend associated with the age among the prediction errors for the model with model ID $\alpha = 0.22$. As seen there were no clear relation.

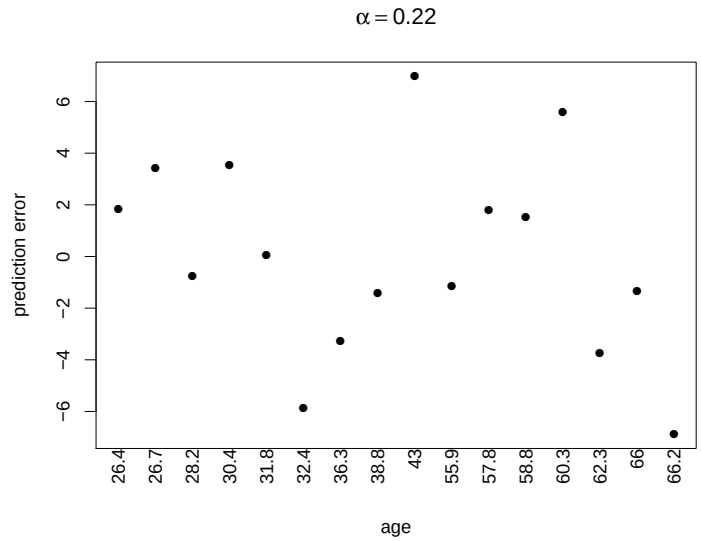
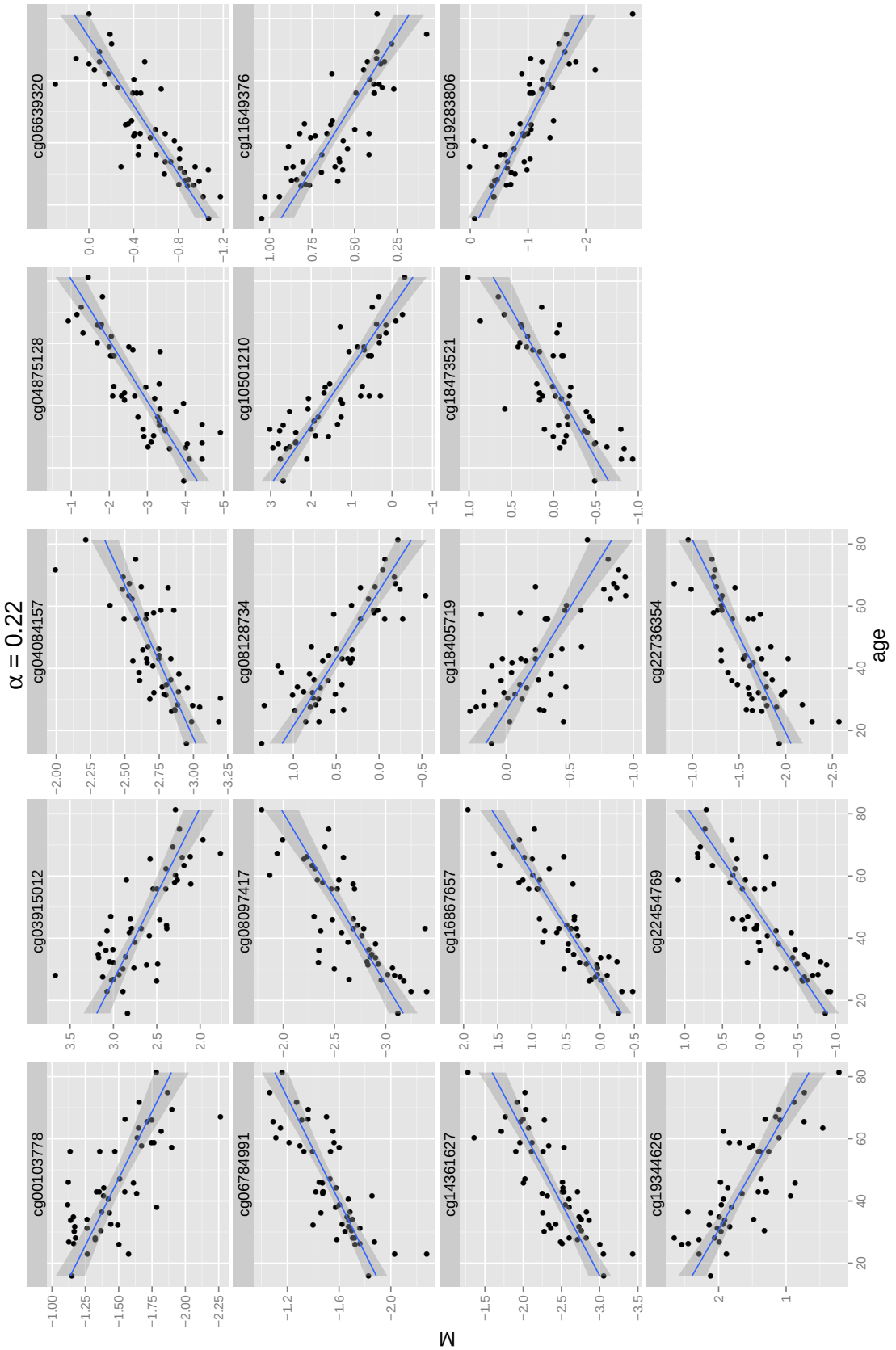


Figure 7.7: Prediction errors plotted against age for the 16 test observations, for the model with model ID $\alpha = 0.22$.

Figure 7.8: The 18 CpG-markers of the model with model ID $\alpha = 0.22$.



7.3 Partial Least Squares

The software used to fit a Partial least squares model to the data was the C++-implementation of partial least squares made by Thomas Hladish and Eugene Melamud [Hladish and Melamud]. This method was applied, since the implementation of pls in R was not applicable on such high dimensional data. The method was performed on the training data, and the number of PLS-components was set to 50. In Table 7.5 the explained variance of the first 10 PLS-components is shown together with the corresponding sum of squared errors (SSE). The optimal number of PLS-components was found to be 25 and 3, by leave-one-out (loo) and leave-some-out (lso) cross validation respectively. The amount of data for testing in the lso method was 30% and the number of trials was 450 (10 times the number of observations). The loadings for the CpG-markers were extracted for the first PLS-component, with the purpose to find the CpG-markers with the highest weights. The 30 CpG-markers with the highest weight on the first PLS-component were used to fit a new model on the training data, where only the optimal number of PLS-components was computed. Prediction was then made on the test data. The RMSE for the full model with all CpG-markers included and for the reduced model with 30 CpG-markers included can be seen in Table 7.6 where both 25 and 3 PLS-components were used for prediction.

No. of PLS-Comp.	Explained variance	SSE
1	0.22258	22.55
2	0.91352	2.51
3	0.98381	0.47
4	0.99632	0.11
5	0.99893	0.03
6	0.99978	0.01
7	0.99995	1.32e-03
8	0.99999	2.15e-04
9	1.00000	9.84e-06
10	1.00000	8.62e-07

Table 7.5: Explained variance associated with the number of PLS-components.

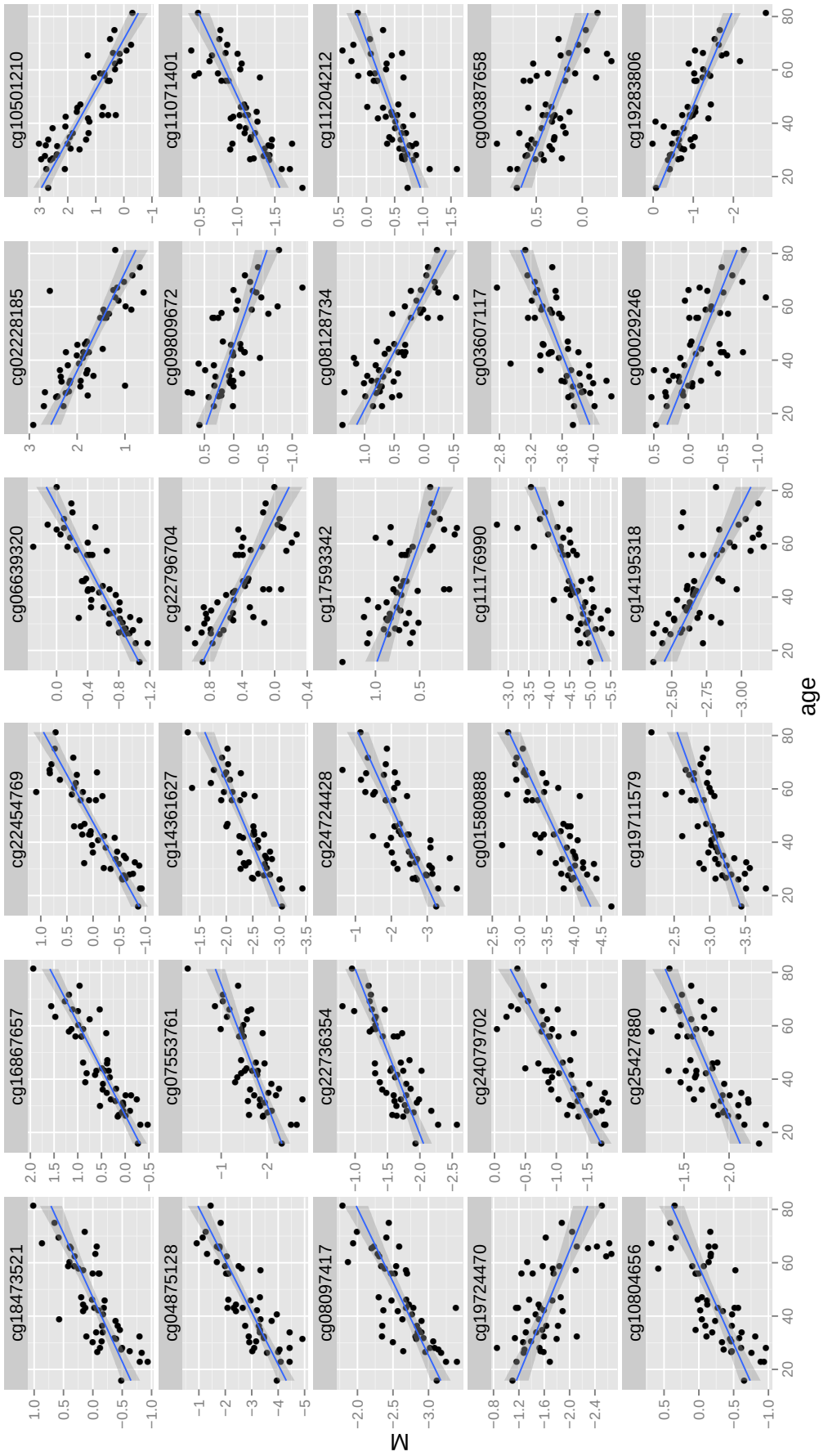
	RMSE, loo	RMSE, lso
All CpG-markers	11.65	11.67
30 CpG-markers	19.27	7.24

Table 7.6: Results of the Partial least squares method with 25 (loo) and 3 (lso) PLS-components used.

The 30 CpG-markers with most influence on the first PLS-component are plotted in Figure 7.9. Compared to the 32 CpG-markers found by stability selection, they have 18 in common, and 11 of them are also in common with the model with $\alpha = 0.22$ from stability selection.

The RMSE was determined for the model with 3 PLS-components, where the 10 and 100 most influential CpG-markers were included, to see if further reduction of the RMSE could be obtained by removing more markers. For these models the RMSE was 8.66 and 6.09 respectively.

Figure 7.9: The 30 most influential CpG-markers in the first PLS-component.



7.4 Prediction Performance with Simulated Data

The final model should be used to exclude suspects from a scene of crime, or to generate leads for the police to narrow down the group of potential perpetrators. The reliability and the prediction accuracy of the model was for this reason extremely important, as a bad model with low prediction accuracy could lead to wrong exclusions from a group of suspects. Another important issue was that especially the age 18 needed to have a high prediction accuracy, as there are different rules for sentences of persons above and below the age 18. To determine how the 20 Ridge regression models from stability selection performed at different ages, new data was simulated from a multivariate normal distribution. Let the observed ages be denoted as the vector **age**, and the observed data of the methylation levels for the different CpG-markers be denoted as the matrix **meth**. It was then assumed that **age** and **meth** followed a multivariate normal distribution, as their marginal distributions with good approximation followed a normal distribution, as illustrated by the qq-plots in Appendix A.2 at page 79. The mean and covariance matrix were then computed as follows [Madsen and Thyregod, 2011]

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_{\mathbf{age}} \\ \boldsymbol{\mu}_{\mathbf{meth}} \end{bmatrix}, \qquad \Sigma = \begin{bmatrix} \Sigma_{\mathbf{age}} & \Sigma_{\mathbf{age,meth}} \\ \Sigma_{\mathbf{meth,age}} & \Sigma_{\mathbf{meth}} \end{bmatrix},$$

the conditional distribution was given by

$$\mathbf{meth}|\mathbf{age} = \text{age} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{meth}|\mathbf{age}}, \Sigma_{\mathbf{meth}|\mathbf{age}}),$$

with

$$\begin{aligned} \boldsymbol{\mu}_{\mathbf{meth}|\mathbf{age}} &= \boldsymbol{\mu}_{\mathbf{meth}} + \Sigma_{\mathbf{meth,age}}\Sigma_{\mathbf{age}}^{-1}(\text{age} - \boldsymbol{\mu}_{\mathbf{age}}) \\ \Sigma_{\mathbf{meth}|\mathbf{age}} &= \Sigma_{\mathbf{meth}} - \Sigma_{\mathbf{meth,age}}\Sigma_{\mathbf{age}}^{-1}\Sigma_{\mathbf{age,meth}}. \end{aligned}$$

From this conditional multivariate normal distribution, 100 simulated observations were made for each of the ages 18, 25, 30, 45, 60 and 75, from knowledge of the observed data. This was done for each of the 20 models in section 7.2. In Figure 7.10 a parallel coordinates plot is made, for the model with $\alpha = 0.01$ to illustrate how the simulated data is distributed compared to the observed data. Each of the simulated data sets with 100 observations for each of the six ages are plotted with the transparent colors in addition to the 45 observations from the observed data which are plotted in black. The three youngest ages follow nearly the same pattern, whereas the ages 60 and 75 follow an exact opposite pattern, the age 45 follows a straight pattern in between these two groupings.

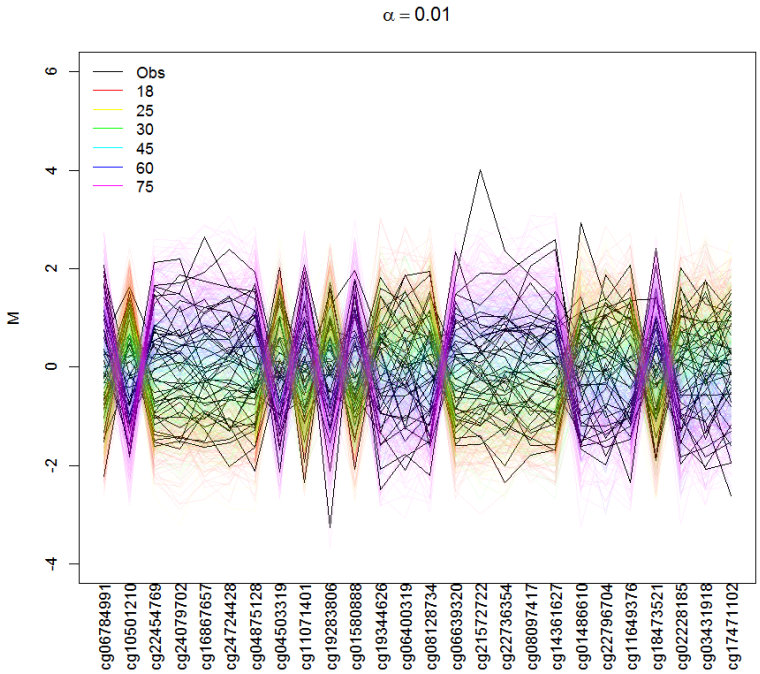


Figure 7.10: Parallel coordinates plot of the simulations compared to the observed data.

Another way of inspecting the behavior of the simulated data in comparison to the observed data was by looking at how the simulated methylation levels for the individual ages were distributed for one single CpG-marker. This is shown in Figure 7.11 for one of the CpG-markers cg10501210 by plotting the observed methylation levels for this marker against age, where the distributions of the simulations are shown by boxplots for the individual ages. The simulations are based on the model with $\alpha = 0.01$.

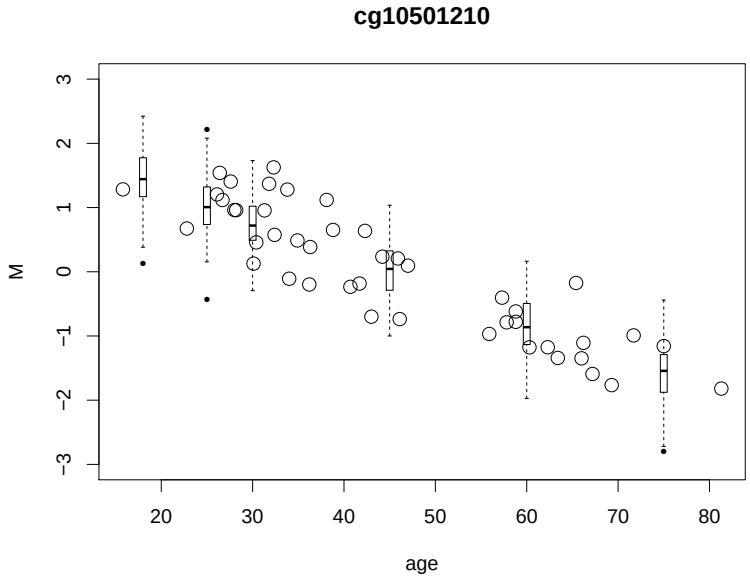


Figure 7.11: Boxplots of the simulated data in addition to the observed data for one of the CpG-markers in the model with $\alpha = 0.01$.

For every of the 20 models from section 7.2, Ridge regression was applied to these using all of the observed data except for the five observations left out for validation. Prediction

was then performed six times for a particular model using the six different simulated data sets, one for each of the six ages. The distribution of the prediction errors age−$\widehat{\mathbf{age}}$ for each of the 20 models are shown as boxplots in Figure 7.12.

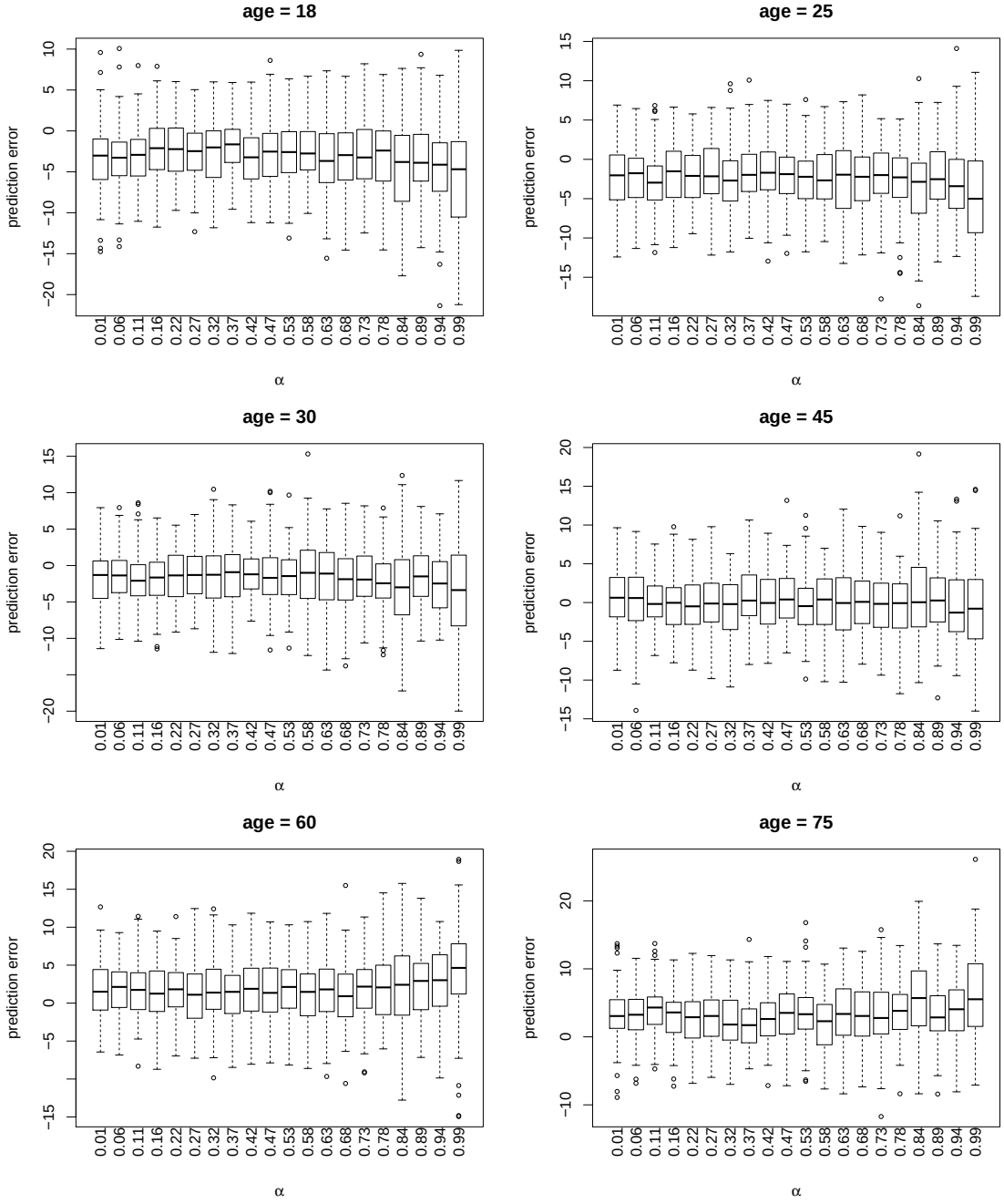


Figure 7.12: Distribution of the prediction errors for the different models with the simulated data.

In Table 7.7 the models with the lowest RMSE on the simulated data are shown for the investigated ages, together with the RMSE for the simulated data with the model which had the lowest RMSE on the observed data (model ID $\alpha = 0.22$).

The model with the lowest averaged RMSE across all ages was the model from stability selection with model ID $\alpha = 0.37$, the averaged RMSE was at 3.98, slightly lower than the one obtained for the model with model ID $\alpha = 0.22$, which was at 4.00.

age	RMSE	Model ID (α)	λ	$\text{RMSE}_{\alpha=0.22}$
18	3.68	0.37	4.01	4.17
25	3.95	0.37	4.01	3.99
30	3.27	0.42	4.01	3.79
45	3.30	0.11	13.44	3.63
60	3.83	0.16	7.69	3.88
75	4.29	0.42	4.01	4.54

Table 7.7: The models with the minimal RMSE at the simulated data of the different ages. The RMSE for the model that gave the lowest RMSE on the observed data ($\alpha = 0.22$), is also shown for comparison.

If we further inspect the boxplots in Figure 7.12, it seems that some systematic bias exists, as the youngest ages seems to be overestimated, according to the majority of the prediction errors being negative, whereas the oldest ages are generally underestimated as the majority of the prediction errors are positive. The prediction errors seems though to be centered around zero as expected for the age 45. There is an explanation for this bias, which is easiest to show for the model ($\alpha = 0.99$) where only the CpG-marker cg10501210 was selected as stable, the bias can here be explained as follows.

In this model, the simulated methylation levels were from the conditional bivariate normal distribution. Let the simulated methylation levels for the CpG-marker cg10501210 be given by the vector **meth***, they were then simulated from the distribution given by [Olofsson, 2005]

$$\mathbf{meth}^*|\mathbf{age} \sim \mathcal{N}\left(\mu_{\mathbf{meth}} + \rho\frac{\sigma_{\mathbf{meth}}}{\sigma_{\mathbf{age}}}(\mathbf{age} - \mu_{\mathbf{age}}), \sigma^2_{\mathbf{meth}}(1-\rho^2)\right),$$

where **meth** is the observed methylation levels for the CpG-marker cg10501210.

The fitted model for the simulated data **meth*** then had the distribution

$$\widehat{\mathbf{age}}|\mathbf{meth}^* \sim \mathcal{N}\left(\mu_{\mathbf{age}} + \rho\frac{\sigma_{\mathbf{age}}}{\sigma_{\mathbf{meth}}}(\mathbf{meth}^* - \mu_{\mathbf{meth}}), \sigma^2_{\mathbf{age}}(1-\rho^2)\right).$$

The mean of these distributions are regression lines, and by those the best predicted values are obtained. Writing out the mean of the fitted model yields an expression of the best predicted ages using the simulated data

$$\begin{aligned}
\mathbb{E}[\widehat{\mathbf{age}}|\mathbf{meth}^*] =& \mu_{\mathbf{age}} + \rho\frac{\sigma_{\mathbf{age}}}{\sigma_{\mathbf{meth}}}\left(\mathbb{E}[\mathbf{meth}^*|\mathbf{age}] - \mu_{\mathbf{meth}}\right) \\
=& \mu_{\mathbf{age}} + \rho\frac{\sigma_{\mathbf{age}}}{\sigma_{\mathbf{meth}}}\left(\mu_{\mathbf{meth}} + \rho\frac{\sigma_{\mathbf{meth}}}{\sigma_{\mathbf{age}}}(\mathbf{age} - \mu_{\mathbf{age}})\right) - \rho\frac{\sigma_{\mathbf{age}}}{\sigma_{\mathbf{meth}}}\mu_{\mathbf{meth}} \\
=& \mu_{\mathbf{age}} + \rho^2(\mathbf{age} - \mu_{\mathbf{age}}) = \rho^2\mathbf{age} + \mu_{\mathbf{age}}(1-\rho^2).
\end{aligned}$$

We would expect this expression to be equal to **age** if the method was unbiased, hence the bias of the method is

$$\begin{aligned}\text{Bias}(\widehat{\mathbf{age}}|\mathbf{meth}^*) &= \rho^2\mathbf{age} + \mu_{\mathbf{age}}(1-\rho^2) - \mathbf{age} \\ &= -\mathbf{age}(1-\rho^2) + \mu_{\mathbf{age}}(1-\rho^2) = (1-\rho^2)(\mu_{\mathbf{age}} - \mathbf{age}).\end{aligned}$$

The bias for this model is controlled by the mean of the observed ages which is $\mu_{\mathbf{age}} = 45.10$. It explains why almost no bias is observed for the simulated data of the age 45. A similar bias occurs for the other models as well.

7.5 Validation

Two different types of methods had been applied to the data. From Stability selection the model with the minimal RMSE was the model with model ID $\alpha = 0.22$ containing 18 CpG-markers. When Partial least squares was applied to the data, a model with 3 PLS-components gave the minimal RMSE when 30 of the most influential CpG-markers were included. For validation of these models, the five samples of replicates were available.

Predicting the validation data using the Ridge regression model from section 7.2 with model ID $\alpha = 0.22$ yielded an RMSE at 2.43. If we instead used the partial least squares model for prediction, we got RMSE= 6.56. Based on these predictions the Ridge regression model with model ID $\alpha = 0.22$ would be preferable, as it resulted in the lowest RMSE.

The validation data was replicates, and we had two samples from the same subject at age 55.90 and two samples from the subject at age 43. The actual predictions and the prediction errors of these observations are shown in Table 7.8 for the Ridge regression model with model ID $\alpha = 0.22$, to show the difference between the replicates.

Observed age	Predicted age	Prediction error
55.90	53.41	2.49
55.90	57.61	-1.71
43.00	45.25	-2.25
43.00	46.88	-3.88
22.80	22.36	0.44

Table 7.8: Prediction of the validation data by the Ridge regression model with model ID $\alpha = 0.22$.

In section 7.4 we investigated simulated data of methylation levels for different ages, and found that the model with model ID $\alpha = 0.37$ had the lowest RMSE for the age 18 and 25, it also turned out to be the model with the averaged minimal RMSE. As the validation data contained one observation at the age 22.8, this model was applied on the validation data to see if it performed better than the model with model ID $\alpha = 0.22$ on this observation. The predictions and prediction errors are shown in Table 7.9.

Observed age	Predicted age	Prediction error
55.90	54.72	1.18
55.90	57.05	-1.15
43.00	48.56	-5.56
43.00	49.52	-6.52
22.80	15.66	7.14

Table 7.9: Prediction of the validation data by the Ridge regression model with model ID $\alpha = 0.37$ and $\lambda = 4.01$ from Table 7.7.

As seen the prediction error of the observation with age 22.8 is highest among the five observations. The model results in an RMSE at 5.04.

8 Discussion and Conclusion

High dimensional regression methods has been applied to genomic micro array data of methylated DNA in this thesis, and the relation to a humans age was investigated. The methods used for the study was the shrinkage models Ridge regression, Elastic net and Lasso. Moreover Partial least squares was determined for comparison with another type of model available for high dimensional data. The purpose was most importantly to find some reliable predictors among hundreds of thousands predictors, usable for consistent age prediction of suspects in crime cases. The interest was to find few predictors with a high predictive performance, as not much DNA material is available at a scene of crime. To ensure that the predictors found by the mentioned shrinkage models were reliable, Stability selection was performed on these. The multicollinearity of the predictors was found to influence their stability for the Lasso method, and hence 20 different Elastic net models was used for the final investigation of these methods, where a model with 18 stable CpG-markers showed the lowest root mean squared error when Ridge regression was applied on it. This model was fitted to the validation data which contained five observations and it yielded an RMSE at 2.43.

The Partial least squares method resulted in a model with 3 PLS-components as the best performing model when only using the 30 most influential CpG-markers from the first PLS-component. On the validation data, this model yielded an RMSE at 6.56. It was seen that the prediction error decreased, when several of the CpG-markers were removed from the model, and hence including all of the CpG-markers would just add noise to the model. The optimal number of CpG-markers could be determined for obtaining the minimal RMSE, but as we needed as few CpG-markers as possible, and the prediction error was seen to increase when more than 30 CpG-markers were removed, and it decreased when more were added, the model from Stability selection would be preferred as a lower prediction error and a lower amount of CpG-markers was obtained with this model. However the two models had 11 CpG-markers in common.

In addition to the results of the observed data, data was simulated for different ages, to see how the 20 models found by Stability selection performed on these by Ridge regression. This was, to get an idea of a more general performance of the models, as the available observed data was limited. Special focus was on the age 18, as there are different rules for sentences depending on whether a person is below or above 18 years. Moreover the available data did not contain any observations at the age 18, or close to it, and hence it was necessary to simulate data for this age, to determine model performance of it. The model from Stability selection with $\alpha = 0.37$ showed the averaged best performance across the different ages, and in particular of predicting the age 18 on the simulated data, with RMSE=3.98. This model was also fitted on the validation data with an RMSE at 5.04. As the validation data contained an observation with the age 22.8, we would expect from the results on the simulated data, that the prediction error of this observation would be the smallest, as this model performed best for both the age 18 and 25 on the simulated data (see Table 7.7). The prediction error for this observation was actually highest among the five validation data points. It shows that no reasonable suggestions for the data can

be made from the assumption of the data following a multivariate normal distribution, even though we saw that their marginal distributions nearly followed a normal distribution. Moreover we saw a systematic bias when predicting the simulations, which caused the results to be less straight forward to interpret. However, regardless of the bias, the minimal RMSE for each of the ages in Table 7.7 was still only around 4, and reminded of the RMSE's of the observed data predicted by the models. The model that was found to perform best on the observed data was the model with model ID $\alpha = 0.22$, and it would then be most reliable to look at the results for the simulated data with this model. For the age 18 the RMSE was 4.17, and averaged across all ages the RMSE was 4.00. These results are as close as we can get of a more general picture of the performance of this model, and of the accuracy of predicting the age 18. Due to the systematic bias and what we saw on the validation data, a deeper study of data from subjects at the age around 18 should be made, to be able to build a model with more accuracy in this area.

In regression models we assume that the predictors are measured without noise, but as we saw for the validation data, the prediction error was different for those two observations who had a replicate. If the data was measured without noise, the measurements would have been exactly the same, and the prediction errors would have been identical. Measurements for genomic data like this is extremely sensitive, due to this it is difficult to make an accurate model, and it is possible that other sets of observed data would result in another model with the best performance. Even though Stability selection was performed, to ensure selection of stable markers, we can not provide for noise in the data which could cause that other CpG-markers would be selected as stable for other observed data sets. The analysis should therefore be performed on several independent observed data sets, to see if the same results would be obtained for these.

In the field of forensic science, two recent studies of the relation between age and DNA methylation [Zbieć-Piekarska et al., 2015, Yi et al., 2015] came up with linear models including two and three CpG-markers respectively, resulting in prediction errors at 6.58 and 4 years. These studies used different technologies for measuring the methylation levels. The study of [Zbieć-Piekarska et al., 2015] investigated the ELOVL2 gene, which has been found interesting in relation to age in more studies as well [Florath et al., 2014, Hannum et al., 2013, Garagnani et al., 2012]. Three out of the seven selected CpG-markers from the ELOVL2 gene in the study of [Zbieć-Piekarska et al., 2015], was among the 32 stable CpG-markers chosen by Stability selection in this thesis, these are cg21572722, cg24724428 and cg16867657 (see Figure 7.3). Moreover cg16867657 is among the 18 CpG-markers from the model with the best performance (model ID $\alpha = 0.22$), see Figure 7.8. However in their study these three CpG-markers were not chosen for the final model, as two other were more significantly correlated with age. Non of the two studies accounted for collinearity or multicollinearity in their models. We experienced that multicollinearity was a great issue, and not accounting for it if it exists, might result in bad predictions on new data, as small changes in data will have big influence on the model.

Two other studies outside the field of forensic science used the same technology by Illumina as was used in this thesis, for examination of age and DNA methylation. One of the studies by [Florath et al., 2014] found a linear regression model with 17 CpG-markers and an average accuracy at 2.6 years, where only 10 of the CpG-markers were detected as significant. Also here, no accounting of collinearity was made. However, more of their 10 significant CpG-markers were among the 18 found by Stability selec-

tion in the model with model ID $\alpha = 0.22$ in this thesis. The other study with the same technology was made by [Hannum et al., 2013]. They used the Elastic net method and bootstrapping for selection of CpG-markers. In their model they also included gender and body mass index (BMI) and yielded an optimal model with 71 CpG-markers and an accuracy at 3.9 years. Also with this model, several CpG-markers were in common with the CpG-markers found stable in this thesis.

Some of the other studies ensured that their subjects did not have any serious diseases. This is something that was not accounted for in this thesis.

For improvement of the analysis, it would be of interest, like in the study of [Hannum et al., 2013], to examine if there were any connection with the gender in relation to age and DNA methylation. Other tissues could also be of interest to determine, such as saliva and semen. It would be preferable to be able to use other tissues than blood for prediction of age, in case if the only available DNA material was saliva or semen. Furthermore DNA-methylation patterns for people with different ethnicity could be studied. Maybe people from Africa or China has other patterns of DNA methylation than people coming from Europe or Scandinavia.

As DNA-methylation is an epigenetic phenomenon, it would not be expected that it would be affected by things having an impact on a persons biological age, such as physical shape and BMI. But these are potential subjects for investigations as well.

Another method which could be considered for high dimensional genomic data, is Supervised principal components. It is a method where the univariate regression coefficients are computed, and based on a threshold found by cross validation, the regression coefficients that exceeds this threshold are chosen. Principal components are then computed of these regression coefficients and are used in a regression model. It reminds of Partial least squares, but an advantage is that Supervised principal components perform feature selection, and hence sort out potential noise from the model [Hastie et al., 2009]. As we performed feature selection of the Partial least squares model, a similar result might be expected of the Supervised principal components method.

All in all, the method of Elastic net combined with Stability selection seems to be a great combination for selecting stable variables in genomic data of high dimension, with its advantage of selecting groups of multicollinear variables. Moreover Ridge regression is a usable method for estimating prediction accuracy of models with highly multicollinear variables. The final best performing Ridge regression model with the 18 CpG-markers seems to be a reasonable suggestion for a model usable for age prediction by its low prediction error and by accounting for collinearity in the data. The methods should though be applied to other independent data sets, to determine the influence of noise in the data. Furthermore a study with more measurements from subjects at the age 18 should be performed, to be able to make a more accurate model of persons at this age.

Bibliography

Lyle Armstrong. *Epigenetics*. Garland Science, 2014.

Edwin F. Bartholomew, Judi L. Nath, and Frederic H. Martini. *Fundamentals of Anatomy and Physiology*. Benjamin Cummings, 2012.

Marina Bibikova, Bret Barnes, Chan Tsan, Vincent Ho, Brandy Klotzle, Jennie M. Le, David Delano, Lu Zhang, Gary P. Schroth, Kevin L. Gunderson, Jian-Bing Fan, and Richard Shen. High density DNA methylation array with single CpG site resolution. *Genomics*, 98:288–295, 2011.

Sarah Dedeuwaerder, Matthieu Defrance, Emilie Calonne, Helene Denis, Christos Sotiriou, and Francois Fuks. Evaluation of the Infinium Methylation 450k technology. *Epigenomics*, 3:771–784, 2011.

Carsten F. Dorman, Jane Elith, Sven Bacher, Carsten Buchman, Gudrun Carl, Gabriel Carre, and Jaime R. Garcia Marquez. Collinearity. *Ecography*, 36:27–46, 2013.

Sandrine Dudoit and Jean Yee Hwa Yang. *The Analysis of Gene Expression Data: Methods and Software*. Springer, 2003.

Ines Florath, Katja Butterbach, Heiko Müller, Melanie Bewerunge-Hudler, and Hermann Brenner. Cross-sectional and longitudinal changes in dna methylation with age: an epigenome-wide analysis revealing over 60 novel age-associated cpg-sites. *Human Molecular Genetics*, 23(5), 2014.

Scott Fortmann-Roe. Illustration of bias-variance trade-off. URL `http://scott.fortmann-roe.com/docs/BiasVariance.html`.

Mario F. Fraga and Manel Esteller. Epigenetics and aging: the targets and the marks. *Trends in Genetics*, 23(8), 2007.

Paolo Garagnani, Maria G. Bacalini, Chiara Pirazzini, Davide Gori, Cristina Giuliani, Daniela Mari, Anna M. Di Blasio, Davide Gentilini, Giovanni Vitale, Sebastiano Collino, Serge Rezzi, Gastone Castellani, Miriam Capri, Stefano Salvioli, and Claudio Franceschi. Methylation of ELOVL2 gene as a new epigenetic marker of age. *Aging Cell*, 11:1132–1134, 2012.

Paul H. Garthwaite. An interpretation of partial least squares. *Journal of the American Statistical Association*, 89(425), 1994.

Gordon D. Ginder and Rakesh Singal. DNA Methylation. *Blood*, 12:4059–4070, 1999.

Gregory Hannum, Justin Guinney, Ling Zhang, Guy Hughes, SriniVas Sadda, Brandy Klotzle, and Marina Bibikova. Genome-wide methylation profiles reveal quantitative views of human aging rates. *Molecular Cell*, 49:359–367, 2013.

Kasper D. Hansen and Martin J. Aryee. *The minfi User's Guide Analyzing Illumina 450k Methylation Arrays*, April 2013.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2009.

Thomas Hladish and Eugene Melamud. C++ implementation of partial least squares. URL `https://github.com/tjhladish/PLS`.

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction To Statistical Learning*. Springer, 2013.

Lynn B. Jorde. *Medical Genetics*. St. Louis, Mo. : Mosby, 2006.

Henrik Madsen and Poul Thyregod. *Introduction to General and Generalized Linear Models*. CRC Press, 2011.

Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society Series B*, 72:417–473, 2010.

Bjørn-Helge Mevik and Henrik René Cederkvist. Mean squared error of prediction (MSEP) estimates for principal component regression (PCR) and partial least squares regression (PLSR). *Journal of Chemometrics*, 18(9):422–429, 2004. ISSN 1099-128X.

Kevin Patrick Murphy. *Machine Learning: a Probabilistic Perspective*. The MIT Press, 2012.

Peter Olofsson. *Probability, Statistics and Stochastic Processes*. WILEY-INTERSCIENCE, 2005.

R. Tyrrell Rockafellar. *Convex analysis*. Princeton Landmarks in Mathematics. Princeton University Press, Princeton, NJ, 1997. ISBN 0-691-01586-4. Reprint of the 1970 original, Princeton Paperbacks.

Martin Sill, Thomas Hielscher, Natalia Becker, and Manuela Zucknick. c060: Extended inference with lasso and elastic-net regularized cox and generalized linear models. *Journal of Statistical Software*, 62(5), 12 2014. ISSN 1548-7660. URL `http://www.jstatsoft.org/v62/i05`.

Stephen Welle. *Statistical Methods for Microarray Data Analysis, Chapter 1*. Humana Press, 2013.

Shao Hua Yi, Long Chang Xu, Kun Mei, Rong Zhi Yang, and Dai Xin Huang. Isolation and identification of age-related dna methylation markers for forensic age-prediction. *Forensic Science International: Genetics*, 11:117–125, 2014.

Shao Hua Yi, Yun Shu Jia, Kun Mei, Rong Zhi Yang, and Dai Xin Huang. Age-related DNA methylation changes for forensic age-prediction. *International Journal of Legal Medicine*, 129(2):237–244, 2015. ISSN 0937-9827. doi: 10.1007/s00414-014-1100-3. URL `http://dx.doi.org/10.1007/s00414-014-1100-3`.

Renata Zbieć-Piekarska, Magdalena Spólnicka, Tomasz Kupiec, Żanetta Makowska, Anna Spas, Agnieszka Parys-Proszek, Krzysztof Kucharczyk, and Rafal Ploski Wojciech Branicki. Examination of DNA methylation status of the ELOVL2 marker may be useful for human age prediction in forensic science. *Forensic Science International: Genetics*, 14:161–167, 2015.

Steven H. Zeisel. Nutrigenomics and metabolomics will change clinical nutrition and public health practice: insights from studies on dietary requirements for choline. *The American Journal of Clinical Nutrition*, 86:542–548, 2007.

Hui Zou and Trevor Hastie. Regularization and variable selection via elastic net. *Journal of Royal Statistical Society B*, 67:301–320, 2005.

Appendix

A.1 Lemmas for Theorem 5.6.1

In section 5.6 we define selection probabilities on the basis of different subsamples of the data. In stead we can split the data randomly into two samples of size $\lfloor n/2 \rfloor$ which have no overlap. In this way we can determine if a variable occur in both samples simultaneously. Define the two random subsets to be I_1 and I_2 of $\{1,\dots,n\}$ where $|I_i| = \lfloor n/2 \rfloor$, $i = 1, 2$ and $I_1 \cap I_2 = \emptyset$. The simultaneously selected set is then defined as

$$\hat{S}^{\text{simult},\lambda} = \hat{S}^{\lambda}(I_1) \cap \hat{S}^{\lambda}(I_2). \tag{A.1}$$

Definition A.1.1 (simultaneous selection probability)**.**
Define the simultaneous selection probabilities $\hat{\Pi}$ *for any set* $K \subseteq \{1,\dots,p\}$ *as*

$$\hat{\Pi}_K^{simult,\lambda} = P^*(K \subseteq \hat{S}^{simult,\lambda}), \tag{A.2}$$

the probability P^* *is with regard to the random sample splitting.*

The following two lemmas are used in the proof of Theorem 5.6.1 in section 5.6.

Lemma A.1.1 (lower bound for simultaneous selection probabilities)**.**
For any set $K \subseteq \{1,\dots,p\}$*, a lower bound for the simultaneous selection probabilities is, for every* $\omega \in \Omega$*, given by*

$$\hat{\Pi}_K^{simult,\lambda} \geq 2\hat{\Pi}_K^{\lambda} - 1.$$

Proof.
Let I_1 and I_2 be defined as above. Denote then the probability $P^*\left[\{K \subseteq \hat{S}^{\lambda}(I_1) \cap K \subseteq \hat{S}^{\lambda}(I_2)\}\right]$ by $s_K(\{1,1\})$. In the same way $s_K(\{1,0\})$, $s_K(\{0,1\})$ and $s_K(\{0,0\})$ are defined to be the probabilities $P^*\left[\{K \subseteq \hat{S}^{\lambda}(I_1) \cap K \not\subseteq \hat{S}^{\lambda}(I_2)\}\right]$, $P^*\left[\{K \not\subseteq \hat{S}^{\lambda}(I_1) \cap K \subseteq \hat{S}^{\lambda}(I_2)\}\right]$ and $P^*\left[\{K \not\subseteq \hat{S}^{\lambda}(I_1) \cap K \not\subseteq \hat{S}^{\lambda}(I_2)\}\right]$ respectively. From (A.1) and (A.2) we have that $\hat{\Pi}_K^{\text{simult},\lambda} = s_K(\{1,1\})$. We can express the selection probabilities based on subsampling by the simultaneous selection probabilities based on sample splitting as follows

$$\begin{aligned}
\hat{\Pi}_K^{\lambda} &= s_K(\{1,0\}) + s_K(\{1,1\}) = s_K(\{0,1\}) + s_K(\{1,1\}),\\
1 - \hat{\Pi}_K^{\lambda} &= s_K(\{1,0\}) + s_K(\{0,0\}) = s_K(\{0,1\}) + s_K(\{0,0\}).
\end{aligned}$$

It follows that $s_K(\{1,0\}) \leq 1 - \hat{\Pi}_K^{\lambda}$ since $s_K(\{0,0\}) \geq 0$, and we get that

$$\hat{\Pi}_K^{\text{simult},\lambda} = s_K(\{1,1\}) = \hat{\Pi}_K^{\lambda} - s_K(\{1,0\}) \geq 2\hat{\Pi}_K^{\lambda} - 1,$$

as desired. □

Lemma A.1.2.
Let $K \subset \{1,\ldots,p\}$ and $\hat{S}^{\lambda}$ the set of selected variables based on a sample size of $\lfloor n/2 \rfloor$.

If $P\left(K \subseteq \hat{S}^{\lambda}\right) \leq \varepsilon$, then

$$P\left(\hat{\Pi}_K^{simult,\lambda} \geq \xi\right) \leq \varepsilon^2/\xi.$$

If $P\left(K \subseteq \cup_{\lambda\in\Lambda} \hat{S}^{\lambda}\right) \leq \varepsilon$ for some $\Lambda \in \mathbb{R}^+$, then

$$P\left\{\max_{\lambda\in\Lambda}\left(\hat{\Pi}_K^{simult,\lambda}\right) \geq \xi\right\} \leq \varepsilon^2/\xi.$$

The proof of A.1.2 can be found in [Meinshausen and Bühlmann, 2010].

A.2 Normality assumptions

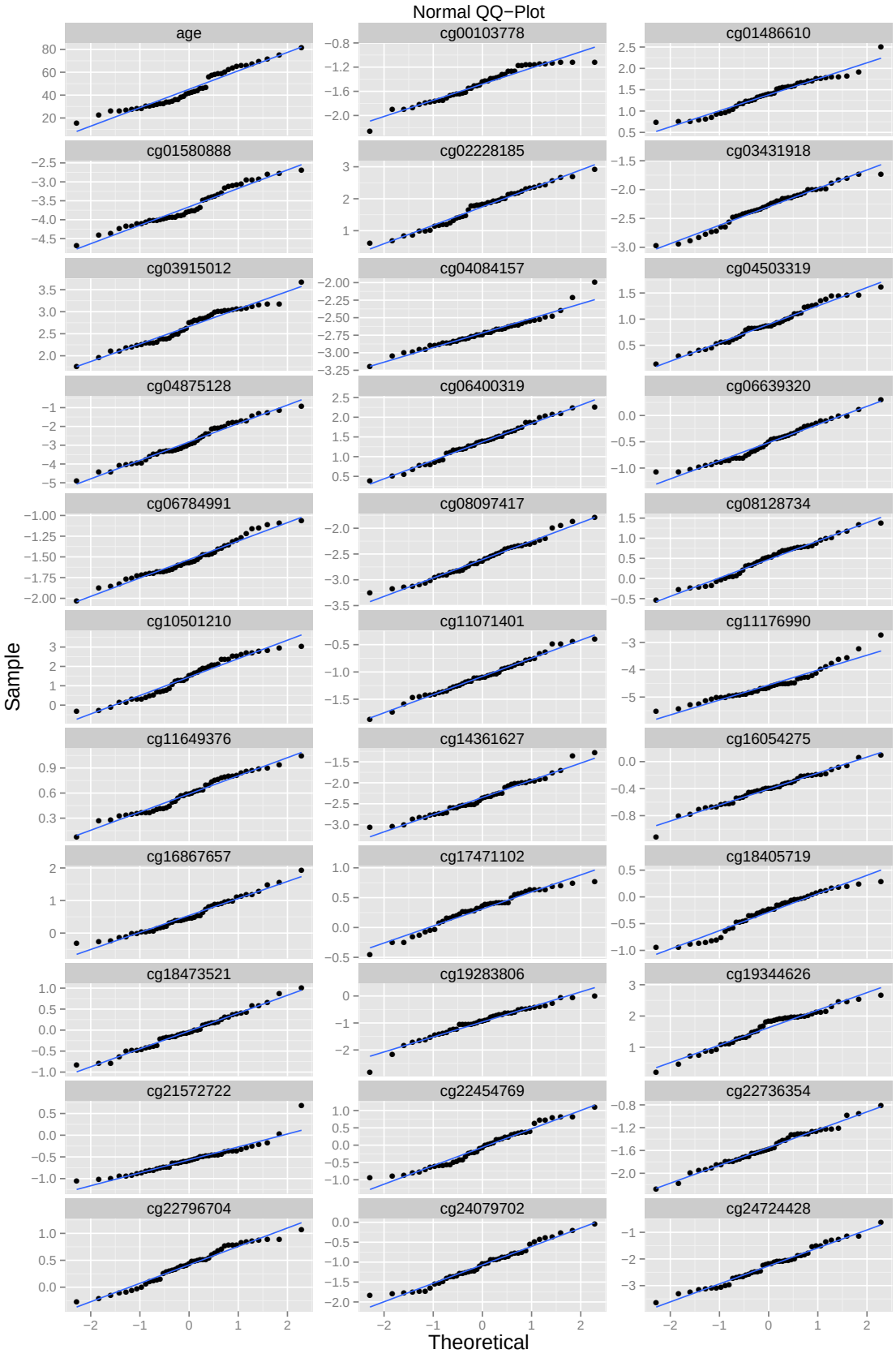


Figure A.1: QQ-plots of the distribution of the ages and the methylation levels for the 32 CpG-markers found by stability selection, against the normal distribution.