Lecture notes on ridge regression

Version 0.02, September 21, 2015.

Wessel N. van Wieringen[1,2]

[1] Department of Epidemiology and Biostatistics, VU University Medical Center
P.O. Box 7057, 1007 MB Amsterdam, The Netherlands
[2] Department of Mathematics, VU University Amsterdam
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
Email: w.vanwieringen@vumc.nl

Disclaimer

This document is a collection of many well-known results on ridge regression. The current status of the document is 'work-in-progress' as it is incomplete (more results from literature will be included) and it may contain inconsistencies and errors. Hence, reading and believing at own risk. Finally, proper reference to the original source may sometimes be lacking. This is regrettable and these references (if ever known to the author) will be included in later versions.

Contents

1 Ridge regression

High-throughput techniques measure many characteristics of a single sample simultaneously. The number of characteristics p measured may easily exceed ten thousand. In most medical studies the number of samples n involved often falls behind the number of characteristics measured, i.e: $p > n$. The resulting $(n \times p)$-dimensional data matrix $\mathbf{X}$:

$$\mathbf{X} = (X_{*,1} \,|\, \ldots \,|\, X_{*,p}) = \begin{pmatrix} X_{1,*} \\ \vdots \\ X_{n,*} \end{pmatrix} = \begin{pmatrix} X_{1,1} & \ldots & X_{1,p} \\ \vdots & \ddots & \vdots \\ X_{n,1} & \ldots & X_{n,p} \end{pmatrix}$$

from such a study contains a larger number of covariates than samples. When $p > n$ the data matrix $\mathbf{X}$ is said to be *high-dimensional*.

In these notes we adopt the traditional statistical notation of the data matrix. An alternative notation would be $\mathbf{X}^\top$ (rather than $\mathbf{X}$), which is employed in the field of (statistical) bioinformatics. In $\mathbf{X}^\top$ the rows comprise the samples rather than the covariates. The case for the bioinformatics notation stems from practical arguments. A spreadsheet is designed to have more rows than columns. In case $p > n$ the traditional notation yields a spreadsheet with more columns than rows. When $p > 10000$ the conventional display is impractical. In these notes we stick to the conventional statistical notation of the data matrix as all mathematical expressions involving $\mathbf{X}$ are then in line with those of standard textbooks on regression.

The information contained in $\mathbf{X}$ is often used to explain a particular property of the samples involved. In applications in molecular biology $\mathbf{X}$ may contain microRNA expression data from which the expression levels of a gene are to be described. When the gene's expression levels are denoted by $\mathbf{Y} = (Y_1, \ldots, Y_n)^\top$, the aim is to find the linear relation $Y_i = \mathbf{X}_{i,*}\boldsymbol{\beta}$ from the data at hand by means of regression analysis. Regression is however frustrated by the high-dimensionality of $\mathbf{X}$ (illustrated in Section 1.1 and at the end of Section 1.4). These notes discuss how regression may be modified to accommodate the high-dimensionality of $\mathbf{X}$.

1.1 Ridge regression

When the design matrix is high-dimensional, the covariates (the columns of $\mathbf{X}$) are super-collinear. Recall *collinearity* in regression analysis refers to the event of two (or multiple) covariates being highly linearly related. Consequently, the subspace spanned by collinear covariates may not be (or close to not being) of full rank. When the subspace (onto which $\mathbf{Y}$ is projected) is (close to) rank deficient, it is (almost) impossible to separate the contribution of the individual covariates. The uncertainty with respect to the covariate responsible for the variation explained in $\mathbf{Y}$ is often reflected in the fit of the linear regression model to data by a large error of the estimates of the regression parameters corresponding to the collinear covariates.

Example 1.1 The flotillins (the FLOT-1 and FLOT-2 genes) have been observed to regulate the proto-oncogene ERBB2 *in vitro* (Pust *et al.*, 2013). One may wish to corroborate this *in vivo*. To this end we use gene expression data of a breast cancer study, available as a Bioconductor package: `breastCancerVDX`. From this study the expression levels of probes interrogating the FLOT-1 and ERBB2 genes are retrieved. For clarity of the illustration the FLOT-2 gene is ignored. After centering, the expression levels of the first ERBB2 probe are regressed on those of the four FLOT-1 probes. The R-code below carries out the data retrieval and analysis.

Listing 1.1 R code

```
# load packages
```

```
library(Biobase)
library(breastCancerVDX)

# ids of genes FLOT1
idFLOT1 <- which(fData(vdx)[,5] == 10211)

# ids of ERBB2
idERBB2 <- which(fData(vdx)[,5] == 2064)

# get expression levels of probes mapping to FLOT genes
X <- t(exprs(vdx)[idFLOT1,])
X <- sweep(X, 2, colMeans(X))

# get expression levels of probes mapping to FLOT genes
Y <- t(exprs(vdx)[idERBB2,])
Y <- sweep(Y, 2, colMeans(Y))

# regression analysis
summary(lm(formula = Y[,1] ~ X[,1] + X[,2] + X[,3] + X[,4]))

# correlation among the covariates
cor(X)
```

Prior to the regression analysis, we first assess whether there is collinearity among the FLOT-1 probes through evaluation of the correlation matrix. This reveals a strong correlation ($\hat{\rho} = 0.91$) between the second and third probe. All other cross-correlations do not exceed the 0.20 (in an absolute sense). Hence, there is collinearity among the columns of the design matrix in the to-be-performed regression analysis.

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.0000     0.0633  0.0000   1.0000
X[, 1]        0.1641     0.0616  2.6637   0.0081 **
X[, 2]        0.3203     0.3773  0.8490   0.3965
X[, 3]        0.0393     0.2974  0.1321   0.8949
X[, 4]        0.1117     0.0773  1.4444   0.1496
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.175 on 339 degrees of freedom
Multiple R-squared:  0.04834,Adjusted R-squared:  0.03711
F-statistic: 4.305 on 4 and 339 DF,  p-value: 0.002072
```

The output of the regression analysis above shows the first probe to be significantly associated to the expression levels of ERBB2. The collinearity of the second and third probe reveals itself in the standard errors of the effect size: for these probes the standard error is much larger than those of the other two probes. This reflects the uncertainty in the estimates. Regression analysis has difficulty to decide to which covariate the explained proportion of variation in the response should be attributed. The large standard error of these effect sizes propagates to the testing as the Wald test statistic is the ratio of the estimated effect size and its standard error. Collinear covariates are thus less likely pass the significance threshold. □

The case of two (or multiple) covariates being perfectly linearly dependent is referred as *super-collinearity*. The rank of a high-dimensional design matrix is maximally equal to n: $\text{rank}(\mathbf{X}) \leq n$. Consequently, the dimension of subspace spanned by the columns of $\mathbf{X}$ is smaller than or equal to n. As $p > n$, this implies that columns of $\mathbf{X}$ are linearly dependent. Put differently, a high-dimensional $\mathbf{X}$ suffers from super-collinearity.

Example 1.2 *Super-collinearity*

Consider the design matrix:

$$\mathbf{X} = \begin{pmatrix} 1 & -1 & 2 \\ 1 & 0 & 1 \\ 1 & 2 & -1 \\ 1 & 1 & 0 \end{pmatrix}$$

The columns of $\mathbf{X}$ are linearly dependent: the first column is the row-wise sum of the other two columns. The rank (more correct, the column rank) of a matrix is the dimension of space spanned by the column vectors. Hence, the rank of $\mathbf{X}$ is equal to the number of linearly independent columns: $\text{rank}(\mathbf{X}) = 2$. □

Super-collinearity of an $(n \times p)$-dimensional design matrix $\mathbf{X}$ implies* that the rank of the $(p \times p)$-dimensional matrix $\mathbf{X}^\top\mathbf{X}$ is smaller than p, and, consequently, it is singular. A square matrix that does not have an inverse is called *singular*. A matrix $\mathbf{A}$ is singular if and only if its determinant is zero: $\det(\mathbf{A}) = 0$.

Example 1.3 *Singularity*
Consider the matrix $\mathbf{A}$ given by:

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}$$

Clearly, $\det(\mathbf{A}) = a_{11}a_{22} - a_{12}a_{21} = 1 \times 4 - 2 \times 2 = 0$. Hence, $\mathbf{A}$ is singular and its inverse is undefined.□

As $\det(\mathbf{A})$ is equal to the product of the eigenvalues λ_j of $\mathbf{A}$, the matrix $\mathbf{A}$ is singular if one (or more) of the eigenvalues of $\mathbf{A}$ is zero. To see this, consider the spectral decomposition of $\mathbf{A}$:

$$\mathbf{A} = \sum_{j=1}^{p} \lambda_j \, \mathbf{v}_j \, \mathbf{v}_j^\top,$$

where $\mathbf{v}_j$ is the eigenvector corresponding to λ_j. The inverse of $\mathbf{A}$ is then:

$$\mathbf{A}^{-1} = \sum_{j=1}^{p} \lambda_j^{-1} \, \mathbf{v}_j \, \mathbf{v}_j^\top.$$

The right-hand side is undefined if $\lambda_j = 0$ for any j.

Example 1.3 *Singularity (continued)*
Revisit Example 1.3. Matrix $\mathbf{A}$ has eigenvalues $\lambda_1 = 5$ and $\lambda_2 = 0$. According to the spectral decomposition, the inverse of $\mathbf{A}$ is:

$$\mathbf{A}^{-1} = \frac{1}{5} \, \mathbf{v}_1 \, \mathbf{v}_1^\top + \frac{1}{0} \, \mathbf{v}_2 \, \mathbf{v}_2^\top.$$

This expression is undefined as we divide by zero in the second summand on the right-hand side. □

In summary, the columns of a high-dimensional design matrix $\mathbf{X}$ are linearly dependent and this super-collinearity causes $\mathbf{X}^\top\mathbf{X}$ to be singular. Now recall the ML estimator of the parameter of the linear regression model:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y}. \tag{1.1}$$

This estimator is only well-defined if $(\mathbf{X}^\top\mathbf{X})^{-1}$ exits. Hence, when $\mathbf{X}$ is high-dimensional the regression parameter $\boldsymbol{\beta}$ cannot be estimated.

Above only the practical consequence of high-dimensionality is presented: the expression $(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y}$

*If the (column) rank of $\mathbf{X}$ is smaller than p, there exists a non-trivial $\mathbf{v} \in \mathbb{R}^p$ such that $\mathbf{X}\mathbf{v} = \mathbf{0}_{p\times 1}$. Multiplication of this inequality by $\mathbf{X}^\top$ yields $\mathbf{X}^\top\mathbf{X}\mathbf{v} = \mathbf{0}_{p\times 1}$. As $\mathbf{v} \neq \mathbf{0}_{p\times 1}$, this implies that $\mathbf{X}^\top\mathbf{X}$ is not invertible.

cannot be evaluated numerically. But the problem arising from the high-dimensionality of the data is more fundamental. To appreciate this, consider the normal equations:

$$\mathbf{X}^\top\mathbf{X}\boldsymbol{\beta} \;=\; \mathbf{X}^\top\mathbf{Y}.$$

The matrix $\mathbf{X}^\top\mathbf{X}$ is of rank n, while $\boldsymbol{\beta}$ is a vector of length p. Hence, while there are p unknowns, the system of linear equations from which these are to be solved effectively comprises n degrees of freedom. If $p > n$, the vector $\boldsymbol{\beta}$ cannot uniquely be determined from this system of equations. To make this more specific let U be the n-dimensional space spanned by the columns of $\mathbf{X}$ and the $p-n$-dimensional space V be orthogonal complement of U, i.e. $V = U^\perp$. Then, $\mathbf{X}\mathbf{v} = \mathbf{0}_{p\times 1}$ for all $\mathbf{v} \in V$. So, V is the non-trivial null space of $\mathbf{X}$. Consequently, as $\mathbf{X}^\top\mathbf{X}\mathbf{v} = \mathbf{X}^\top\mathbf{0}_{p\times 1} = \mathbf{0}_{n\times 1}$, the solution of the normal equations is:

$$\hat{\boldsymbol{\beta}} \;=\; (\mathbf{X}^\top\mathbf{X})^-\mathbf{X}^\top\mathbf{Y} + \mathbf{v} \qquad \text{for all } \mathbf{v} \in V,$$

where $\mathbf{A}^-$ denotes the Moore-Penrose inverse of the matrix $\mathbf{A}$, which is defined as:

$$\mathbf{A}^- \;=\; \sum_{j=1}^{p} \lambda_j^{-1}\, I_{\{\lambda_j \neq 0\}}\, \mathbf{v}_j\, \mathbf{v}_j^\top.$$

The solution of the normal equations is thus only determined up to an element from a non-trivial space V, and there is no unique estimator of the regression parameter.

To obtain an estimate of the regression parameter $\boldsymbol{\beta}$ when $\mathbf{X}$ is (close to) super-collinearity, Hoerl and Kennard (1970) proposed an ad-hoc fix to resolve the (almost) singularity of $\mathbf{X}^\top\mathbf{X}$. Simply replace $\mathbf{X}^\top\mathbf{X}$ by $\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}$ with $\lambda \in [0,\infty)$. The scalar λ is a tuning parameter, henceforth called the *penalty parameter*.

Example 1.2 *Super-collinearity (continued)*
Recall the super-collinear design matrix $\mathbf{X}$ of Example 1.2. Then, for (say) $\lambda = 1$:

$$\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p} \;=\; \begin{pmatrix} 5 & 2 & 2 \\ 2 & 7 & -4 \\ 2 & -4 & 7 \end{pmatrix}.$$

The eigenvalues of this matrix are 11, 7, and 1. Hence, $\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}$ has no zero eigenvalue and its inverse is well-defined. □

With the ad-hoc fix for the singularity of $\mathbf{X}^\top\mathbf{X}$, Hoerl and Kennard (1970) proceed to define the *ridge regression estimator*:

$$\hat{\boldsymbol{\beta}}(\lambda) \;=\; (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y}, \tag{1.2}$$

for $\lambda \in [0,\infty)$. Clearly, this is a well-defined estimator, even if $\mathbf{X}$ is high-dimensional (for λ strictly positive). However, each choice of λ leads to a different ridge regression estimate. The set of all ridge regression estimates $\{\hat{\boldsymbol{\beta}}(\lambda) : \lambda \in [0,\infty)\}$ is called the *solution path* of the ridge estimator.

Example 1.2 *Super-collinearity (continued)*
Recall the super-collinear design matrix $\mathbf{X}$ of Example 1.2. Suppose that the corresponding response vector is $\mathbf{Y} = (1.3, -0.5, 2.6, 0.9)^\top$. The ridge regression estimates for, e.g. $\lambda = 1, 2$, and 10 are then:

$$\begin{aligned} \hat{\boldsymbol{\beta}}(1) &= (0.614, 0.548, 0.066)^\top, \\ \hat{\boldsymbol{\beta}}(2) &= (0.537, 0.490, 0.048)^\top, \\ \hat{\boldsymbol{\beta}}(10) &= (0.269, 0.267, 0.002)^\top. \end{aligned}$$

The full solution path of the ridge estimator is plotted in Figure 1.1.

Having obtained an estimate of the regression parameter $\boldsymbol{\beta}$, one can define the fit $\hat{\mathbf{Y}}$. It is defined analogous to the standard case:

$$\hat{\mathbf{Y}}(\lambda) \;=\; \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda) \;=\; \mathbf{X}(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y} \;:=\; \mathbf{H}(\lambda)\mathbf{Y}.$$

Previously, when using the ML estimator, the fit could be understood as a projection of $\mathbf{Y}$ onto the subspace spanned by the columns of $\mathbf{X}$. The fit $\hat{\mathbf{Y}}(\lambda)$ corresponding to the ridge estimator is not a projection of $\mathbf{Y}$ onto $\mathbf{X}$ (confer Exercise 1.3 a). Consequently, the 'ridge residuals' $\mathbf{Y} - \hat{\mathbf{Y}}(\lambda)$ are not orthogonal to the fit $\hat{\mathbf{Y}}(\lambda)$ (confer Exercise 1.3 b).

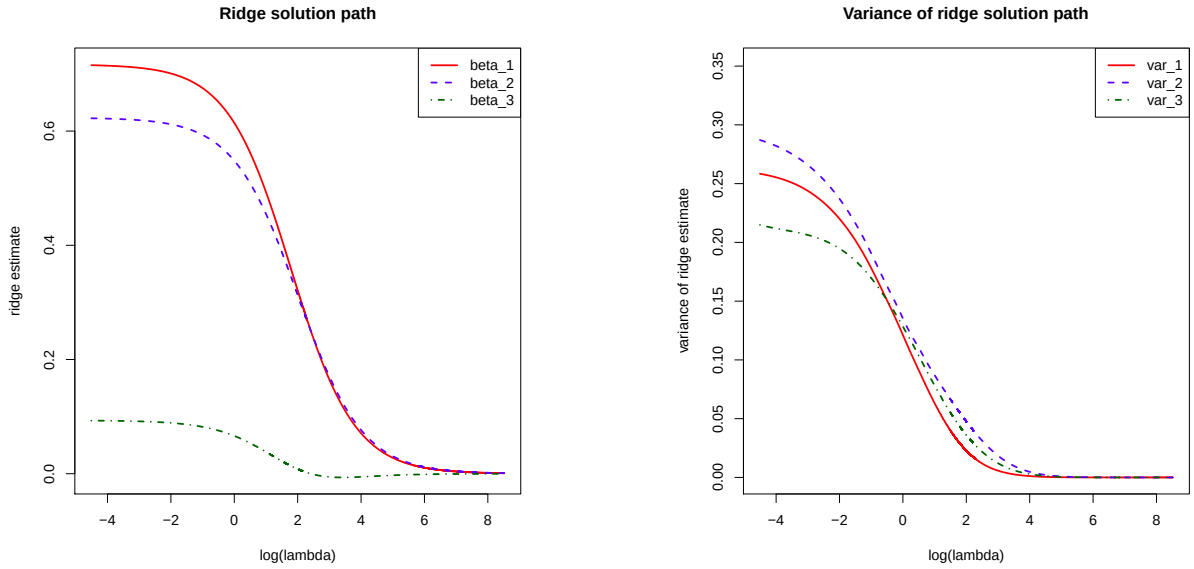

Figure 1.1: Solution path of the ridge estimator and its variance. The left panel shows the solution path of the ridge estimator for the data of Example 1.2. In the right panel the corresponding variance of the ridge estimator is plotted against the (logarithm of the) penalty parameter.

1.2 Expectation

The left panel of Figure 1.1 shows ridge estimates of the regression parameters converging to zero as the penalty parameter tends to infinity. This behaviour of the ridge estimator does not depend on the specifics of the data set. To see this study the expectation of the ridge estimator:

$$\begin{aligned} E\big[\hat{\boldsymbol{\beta}}(\lambda)\big] &= E\big[(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y}\big] \\ &= E\big\{[\mathbf{I}_{p\times p} + \lambda\,(\mathbf{X}^\top\mathbf{X})^{-1}]^{-1}(\mathbf{X}^\top\mathbf{X})^{-1}\,\mathbf{X}^\top\mathbf{Y}\big\} \\ &= E\big\{[\mathbf{I}_{p\times p} + \lambda\,(\mathbf{X}^\top\mathbf{X})^{-1}]^{-1}\,\hat{\boldsymbol{\beta}}\big\} \\ &= [\mathbf{I}_{p\times p} + \lambda\,(\mathbf{X}^\top\mathbf{X})^{-1}]^{-1}\,E\big(\hat{\boldsymbol{\beta}}\big) \\ &= [\mathbf{I}_{p\times p} + \lambda\,(\mathbf{X}^\top\mathbf{X})^{-1}]^{-1}\,\boldsymbol{\beta} \\ &= \mathbf{X}^\top\mathbf{X}(\lambda\mathbf{I}_{p\times p} + \mathbf{X}^\top\mathbf{X})^{-1}\,\boldsymbol{\beta}. \end{aligned}$$

Clearly, $E\big[\hat{\boldsymbol{\beta}}(\lambda)\big] \neq \boldsymbol{\beta}$ for any $\lambda > 0$. Hence, the ridge estimator is biased.

From the expression above it is clear that the expectation of the ridge estimator vanishes as λ tends to infinity:

$$\lim_{\lambda\to\infty} E\big[\hat{\boldsymbol{\beta}}(\lambda)\big] \;=\; \lim_{\lambda\to\infty} \mathbf{X}^\top\mathbf{X}(\lambda\mathbf{I}_{p\times p} + \mathbf{X}^\top\mathbf{X})^{-1}\,\boldsymbol{\beta} \;=\; \mathbf{0}_{p\times 1}.$$

Hence, all regression coefficients are shrunken towards zero as the penalty parameter increases. This also holds for $\mathbf{X}$ with $p > n$. Furthermore, this behaviour is not strictly monotone in λ: $\lambda_a > \lambda_b$ does not necessarily imply $|\hat{\beta}_j(\lambda_a)| < |\hat{\beta}_j(\lambda_b)|$. Upon close inspection this can witnessed from the ridge solution path of β_3 in Figure 1.1.

Example 1.4 *Orthonormal design matrix*
Consider an orthonormal design matrix $\mathbf{X}$, i.e.:

$$\mathbf{X}^\top\mathbf{X} \;=\; \mathbf{I}_{p\times p} \;=\; (\mathbf{X}^\top\mathbf{X})^{-1}.$$

An example of an orthonormal design matrix would be:

$$\mathbf{X} \;=\; \frac{1}{2}\begin{pmatrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{pmatrix}.$$

This design matrix is orthonormal as $\mathbf{X}^\top\mathbf{X} = \mathbf{I}_{2\times 2}$, which is easily verified:

$$\mathbf{X}^\top\mathbf{X} = \frac{1}{4}\begin{pmatrix} -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \end{pmatrix}\begin{pmatrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{pmatrix} = \frac{1}{4}\begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}.$$

In case of an orthonormal design matrix the relation between the OLS and ridge estimator is:

$$\begin{aligned} \hat{\boldsymbol{\beta}}(\lambda) &= (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &= (\mathbf{I}_{p\times p} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &= (1+\lambda)^{-1}\mathbf{I}_{p\times p}\mathbf{X}^\top\mathbf{Y} \\ &= (1+\lambda)^{-1}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &= (1+\lambda)^{-1}\hat{\boldsymbol{\beta}}. \end{aligned}$$

Hence, the ridge estimator scales the OLS estimator by a factor. When taking the expectation on both sides, it is evident that the ridge estimator converges to zero as $\lambda \to \infty$. □

1.3 Variance

As for the ML estimate of the regression parameter $\boldsymbol{\beta}$ of the linear regression model we may derive the second moment of the ridge estimator. Hereto define:

$$\mathbf{W}_\lambda = [\mathbf{I} + \lambda(\mathbf{X}^\top\mathbf{X})^{-1}]^{-1}.$$

Using $\mathbf{W}_\lambda$ the ridge estimator $\hat{\boldsymbol{\beta}}(\lambda)$ can be expressed as $\mathbf{W}_\lambda\hat{\boldsymbol{\beta}}$ for:

$$\begin{aligned} \mathbf{W}_\lambda\hat{\boldsymbol{\beta}} &= \mathbf{W}_\lambda(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &= \{(\mathbf{X}^\top\mathbf{X})^{-1}[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]\}^{-1}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &= [\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\mathbf{X}^\top\mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &= [\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\mathbf{X}^\top\mathbf{Y} \\ &= \hat{\boldsymbol{\beta}}(\lambda). \end{aligned}$$

The linear operator $\mathbf{W}_\lambda$ thus transforms the ML estimator of the regression parameter into the ridge estimator.

It is now easily seen that:

$$\begin{aligned} \mathrm{Var}[\hat{\boldsymbol{\beta}}(\lambda)] &= \mathrm{Var}[\mathbf{W}_\lambda\hat{\boldsymbol{\beta}}] \\ &= \mathbf{W}_\lambda\mathrm{Var}[\hat{\boldsymbol{\beta}}]\mathbf{W}_\lambda^\top \\ &= \sigma^2\mathbf{W}_\lambda(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{W}_\lambda^\top, \\ &= \sigma^2[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\mathbf{X}^\top\mathbf{X}\{[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\}^\top, \end{aligned}$$

in which we have used $\mathrm{Var}(\mathbf{AY}) = \mathbf{A}\mathrm{Var}(\mathbf{Y})\mathbf{A}^\top$ for a non-random matrix $\mathbf{A}$, the fact that $\mathbf{W}_\lambda$ is non-random, and $\mathrm{Var}[\hat{\boldsymbol{\beta}}] = \sigma^2(\mathbf{X}^\top\mathbf{X})^{-1}$.

Like the expectation the variance of the ridge estimator vanishes as λ tends to infinity:

$$\lim_{\lambda\to\infty}\mathrm{Var}\big[\hat{\boldsymbol{\beta}}(\lambda)\big] = \lim_{\lambda\to\infty}\sigma^2\mathbf{W}_\lambda(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{W}_\lambda^\top = \mathbf{0}_{p\times p}.$$

Hence, the variance of the ridge regression coefficient estimates decreases towards zero as the penalty parameter becomes large. This is illustrated in the right panel of Figure 1.1 for the data of Example 1.2.

With an explicit expression of the variance of the ridge estimator at hand, we can compare it to that of the OLS estimator:

$$\begin{aligned} \mathrm{Var}[\hat{\boldsymbol{\beta}}] - \mathrm{Var}[\hat{\boldsymbol{\beta}}(\lambda)] &= \sigma^2[(\mathbf{X}^\top\mathbf{X})^{-1} - \mathbf{W}_\lambda(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{W}_\lambda^\top] \\ &= \sigma^2\mathbf{W}_\lambda\{[\mathbf{I} + \lambda(\mathbf{X}^\top\mathbf{X})^{-1}](\mathbf{X}^\top\mathbf{X})^{-1}[\mathbf{I} + \lambda(\mathbf{X}^\top\mathbf{X})^{-1}]^\top - (\mathbf{X}^\top\mathbf{X})^{-1}\}\mathbf{W}_\lambda^\top \\ &= \sigma^2\mathbf{W}_\lambda[2\,\lambda\,(\mathbf{X}^\top\mathbf{X})^{-2} + \lambda^2(\mathbf{X}^\top\mathbf{X})^{-3}]\mathbf{W}_\lambda^\top \\ &= \sigma^2[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}[2\,\lambda\,\mathbf{I}_{p\times p} + \lambda^2(\mathbf{X}^\top\mathbf{X})^{-1}]\{[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\}^\top. \end{aligned}$$

The difference is non-negative definite as each component in the matrix product is non-negative definite. Hence,

$$\text{Var}[\hat{\boldsymbol{\beta}}] \quad \succeq \quad \text{Var}[\hat{\boldsymbol{\beta}}(\lambda)]. \tag{1.3}$$

In words, the variance of the ML estimator is larger than that of the ridge estimator (in the sense that their difference is non-negative definite). The variance inequality (1.3) can be interpreted in terms of the uncertainty of the estimate. This is illustrated by the next example.

Example 1.5 *Variance comparison*
Consider the design matrix:

$$\mathbf{X} \quad = \quad \begin{pmatrix} -1 & 2 \\ 0 & 1 \\ 2 & -1 \\ 1 & 0 \end{pmatrix}.$$

The variances of the ML and ridge (with $\lambda = 1$) estimates of the regression coefficients then are:

$$\text{Var}(\hat{\boldsymbol{\beta}}) \quad = \quad \sigma^2 \begin{pmatrix} 0.3 & 0.2 \\ 0.2 & 0.3 \end{pmatrix} \qquad \text{and} \qquad \text{Var}(\hat{\boldsymbol{\beta}}) \quad = \quad \sigma^2 \begin{pmatrix} 0.1524 & 0.0698 \\ 0.0698 & 0.1524 \end{pmatrix}.$$

These variance can be used to construct confidence intervals of the estimates. The 50%, 75% and 95% confidence intervals for the ML and ridge estimates are plotted in Figure 1.2. In line with inequality (1.3) the confidence intervals of the ridge estimate are smaller than that of the ML estimate. □

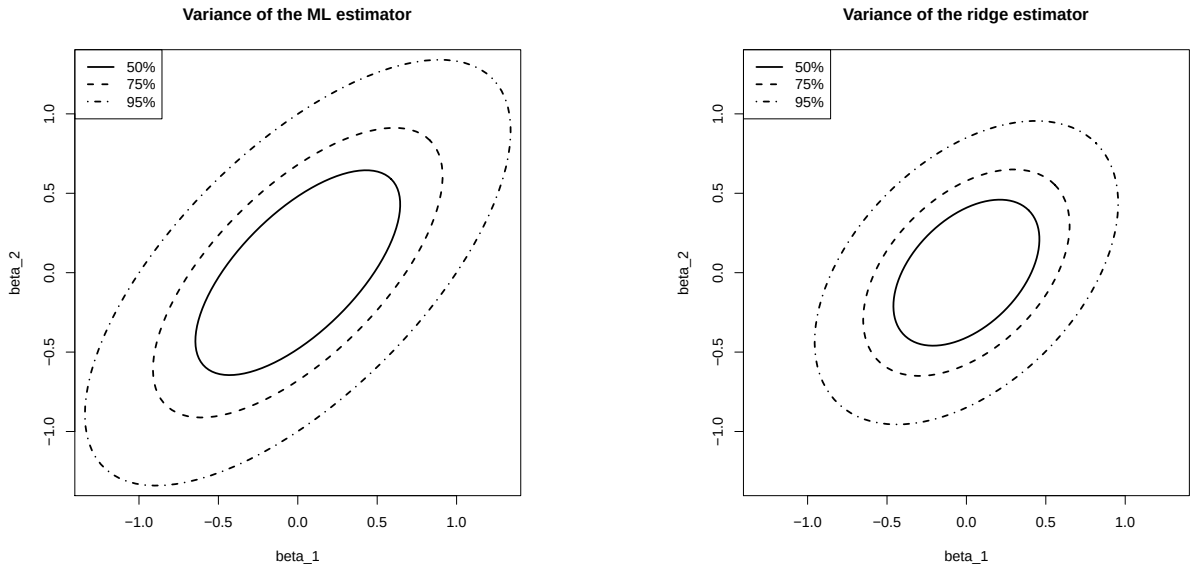

Figure 1.2: Solution path of the ridge estimator and its variance. The left panel shows the solution path of the ridge estimator for the data of Example 1.2. In the right panel the corresponding variance of the ridge estimator is plotted against the (logarithm of the) penalty parameter.

Example 1.4 *Orthonormal design matrix (continued)*
Assume the design matrix $\mathbf{X}$ is orthonormal. Then, $\text{Var}[\hat{\boldsymbol{\beta}}] = \sigma^2 \mathbf{I}_{p\times p}$ and

$$\begin{aligned} \text{Var}[\hat{\boldsymbol{\beta}}(\lambda)] &= \sigma^2 \mathbf{W}_\lambda (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{W}_\lambda^\top \\ &= \sigma^2 [\mathbf{I}_{p\times p} + \lambda \mathbf{I}_{p\times p}]^{-1} \mathbf{I}_{p\times p} \{[\mathbf{I}_{p\times p} + \lambda \mathbf{I}_{p\times p}]^{-1}\}^\top \\ &= \sigma^2 (1+\lambda)^{-2} \mathbf{I}_{p\times p}. \end{aligned}$$

As the penalty parameter λ is non-negative the former exceeds the latter. In particular, this expression vanishes as $\lambda \to \infty$. □

1.4 Constrained estimation

The ad-hoc fix of Hoerl and Kennard (1970) to super-collinearity of the design matrix (and, consequently the singularity of the matrix $\mathbf{X}^\top\mathbf{X}$) has been motivated post-hoc. The ridge estimator minimizes the *ridge loss function*, which is defined as:

$$\begin{aligned}\mathcal{L}_{\text{ridge}}(\boldsymbol{\beta};\lambda) &= \|\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\lambda\|\boldsymbol{\beta}\|_2^2\\ &= \sum_{i=1}^{n}(Y_i-\mathbf{X}_{i*}\boldsymbol{\beta})^2+\lambda\sum_{j=1}^{p}\beta_j^2.\end{aligned}$$

This loss function is the traditional sum-of-squares augmented with a *penalty*. The particular form of the penalty, $\lambda\|\boldsymbol{\beta}\|_2^2$ is referred to as the *ridge penalty* and λ as the *penalty parameter*. For $\lambda=0$, minimization of the ridge loss function yields the ML estimator. For any $\lambda>0$, the ridge penalty contributes to the loss function, affecting its minimum and its location. The minimum of the sum-of-squares is well-known. The minimum of the ridge penalty is attained at $\boldsymbol{\beta}=\mathbf{0}_{p\times 1}$ whenever $\lambda>0$. The $\boldsymbol{\beta}$ that minimizes $\mathcal{L}_{\text{ridge}}(\boldsymbol{\beta};\lambda)$ then balances the sum-of-squares and the penalty. The effect of the penalty in this balancing act is to shrink the regression coefficients towards zero, its minimum. In particular, the larger λ, the larger the contribution of the penalty to the loss function, the stronger the tendency to shrink non-zero regression coefficients to zero (and decrease the contribution of the penalty to the loss function). This motivates the name 'penalty' as non-zero elements of $\boldsymbol{\beta}$ increase (or penalize) the loss function.

To verify that the ridge estimator indeed minimizes the ridge loss function, proceed as usual. Take the derivative with respect to $\boldsymbol{\beta}$:

$$\begin{aligned}\frac{\partial}{\partial\boldsymbol{\beta}}\mathcal{L}_{\text{ridge}}(\boldsymbol{\beta};\lambda) &= -2\,\mathbf{X}^\top(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})+2\,\lambda\,\mathbf{I}_{p\times p}\,\boldsymbol{\beta}\\ &= -2\,\mathbf{X}^\top\mathbf{Y}+2\,(\mathbf{X}^\top\mathbf{X}+\lambda\,\mathbf{I}_{p\times p})\boldsymbol{\beta}.\end{aligned}$$

Equate the derivative to zero and solve for $\boldsymbol{\beta}$. This yields the ridge regression estimator.

The ridge estimator is thus a stationary point of the ridge loss function. A stationary point corresponds to a minimum if the Hessian matrix with second order partial derivatives is positive definite. The Hessian of the ridge loss function is

$$\frac{\partial^2}{\partial\boldsymbol{\beta}\,\partial\boldsymbol{\beta}^\top}\mathcal{L}_{\text{ridge}}(\boldsymbol{\beta};\lambda) = 2\,(\mathbf{X}^\top\mathbf{X}+\lambda\,\mathbf{I}_{p\times p}).$$

This Hessian is the sum of the semi-positive definite matrix $\mathbf{X}^\top\mathbf{X}$ and the positive definite matrix $\lambda\,\mathbf{I}_{p\times p}$. Lemma 14.2.4 of Harville (2008) then states that the sum of these matrices is itself a positive definite matrix. Hence, the Hessian is positive definite and the ridge loss function has a stationary point at the ridge estimator, which is a minimum.

The ridge regression estimator minimizes the ridge loss function. It rests to verify that is a global minimum. To this end of we introduce the concept of a convex function. As a prerequisite, a set $\mathcal{S}\subset\mathbb{R}^p$ is called *convex* if for all $\boldsymbol{\beta}_1,\boldsymbol{\beta}_2\in\mathcal{S}$ their weighted average $\boldsymbol{\beta}_\theta=(1-\theta)\boldsymbol{\beta}_1+\theta\boldsymbol{\beta}_2$ for all $\theta\in[0,1]$ is itself an element of $\mathcal{S}$, thus $\boldsymbol{\beta}_\theta\in\mathcal{S}$. Examples of convex and nonconvex sets are depicted in Figure 1.3. A function $f(\cdot)$ is *convex* if the set $\{y : y\geq f(\boldsymbol{\beta})$ for all $\boldsymbol{\beta}\in\mathcal{S}$ for any convex $\mathcal{S}\}$, called the epigraph of $f(\cdot)$, is convex. Examples of convex and nonconvex functions are depicted in Figure 1.3. The ridge loss function is the sum of two parabola's, both convex functions in $\boldsymbol{\beta}$. The sum of two convex functions is itself convex (confer Lemma 9.4.2 of Fletcher 2008). The ridge loss function is thus convex. Theorem 9.4.1 of Fletcher 2008 then warrants, by the convexity of the ridge loss function, that the ridge estimator is a global minimum.

From the ridge loss function the limiting behavior of the variance of the ridge regression estimator can be understood. The ridge penalty with its minimum $\boldsymbol{\beta}=\mathbf{0}_{p\times 1}$ does not involve data and, consequently, the variance of its minimum equals zero. With the ridge regression being a compromise between the ML estimator and the minimum of the penalty, so is its variance a compromise of their variances. As λ tends to infinity, the ridge estimator and its variance converge to the minimum and the variance of the minimum, respectively. Hence, in the limit (large λ) the variance of the ridge regression estimator vanishes. Understandably, as the penalty now fully dominates the loss function and, consequently, it

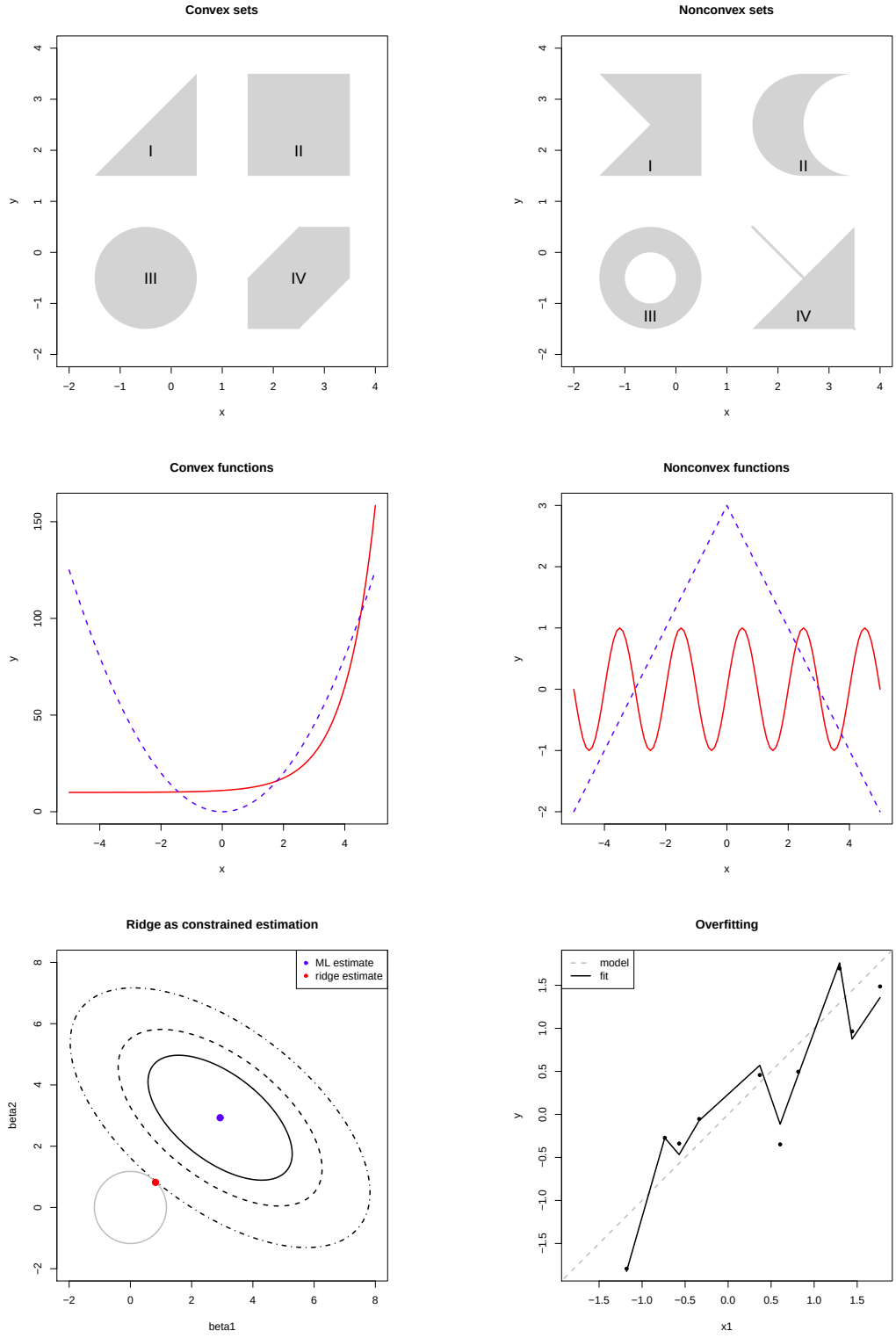

Figure 1.3: Top panels show examples of convex (left) and nonconvex (right) sets. Middle panels show examples of convex (left) and nonconvex (right) functions. The left bottom left illustrates the ridge estimation as a constrained estimation problem. The ellipses represent the contours of the ML loss function, with the blue dot at the center the ML estimate. The circle is the ridge parameter constraint. The red dot is the ridge estimate. It is at the intersection of the ridge constraint and the smallest contour with a non-empty intersection with the constraint. The right bottom panel shows the data corresponding to Example 1.6. The grey line represents the 'true' relationship, while the black line the fitted one.

does no longer involve data (i.e. randomness).

Above it has been shown that the ridge estimator can be defined as:

$$\hat{\boldsymbol{\beta}}(\lambda) \quad = \quad \arg\min_{\boldsymbol{\beta}} \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_2^2. \tag{1.4}$$

This minimization problem can be reformulated into the following constrained optimization problem (illustrated in Figure 1.3):

$$\hat{\boldsymbol{\beta}}(\lambda) \quad = \quad \arg\min_{\|\boldsymbol{\beta}\|_2^2 \leq c} \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2, \tag{1.5}$$

for some suitable $c > 0$. The constrained optimization problem (1.5) can be solved by means of the Karush-Kuhn-Tucker (KKT) multiplier method, which minimizes a function subject to inequality constraints. The KKT multiplier method states that, under some regularity conditions (all met here), there exists a constant $\nu \geq 0$, called the *multiplier*, such that the solution $\hat{\boldsymbol{\beta}}(\nu)$ of the constrained minimization problem (1.5) satisfies the so-called KKT conditions. The first KKT condition (referred to as the stationarity condition) demands that the gradient (with respect to $\boldsymbol{\beta}$) of the Lagrangian associated with the minimization problem equals zero at the solution $\hat{\boldsymbol{\beta}}(\nu)$. The Lagrangian for problem (1.5) is:

$$\|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \nu(\|\boldsymbol{\beta}\|_2^2 - c).$$

The second KKT condition (the complementarity condition) requires that $\nu(\|\hat{\boldsymbol{\beta}}(\nu)\|_2^2 - c) = 0$. If $\nu = \lambda$ and $c = \|\hat{\boldsymbol{\beta}}(\lambda)\|_2^2$, the ridge estimator $\boldsymbol{\beta}(\lambda)$ satisfies both KKT conditions. Hence, both problems have the same solution when $c = \|\hat{\boldsymbol{\beta}}(\lambda)\|_2^2$.

The relevance of viewing the ridge regression estimator as the solution to a constrained estimation problem becomes obvious when considering a typical threat to high-dimensional data analysis: overfitting. *Overfitting* refers to the phenomenon of modelling the noise rather than the signal. In case the true model is parsimonious (few covariates driving the response) and data on many covariates are available, it is likely that a linear combination of all covariates yields a higher likelihood than a combination of the few that are actually related to the response. As only the few covariates related to the response contain the signal, the model involving all covariates then cannot but explain more than the signal alone: it also models the error. Hence, it overfits the data. In high-dimensional settings overfitting is a real threat. The number of explanatory variables exceeds the number of observations. It is thus possible to form a linear combination of the covariates that perfectly explains the response, including the noise.

Large estimates of regression coefficients are often an indication of overfitting. Augmentation of the estimation procedure with a constraint on the regression coefficients is a simple remedy to large parameter estimates. As a conseuence it decreases the probability of overfitting. Overfitting is illustrated in the next Example.

Example 1.6 *(Overfitting)*
Consider an artificial data set comprising of ten observations on a response Y_i and nine covariates $X_{i,j}$. All covariate data are sampled from the standard normal distribution: $X_{i,j} \sim \mathcal{N}(0,1)$. The response is generated by $Y_i = X_{i,1} + \varepsilon_i$ with $\varepsilon_i \sim \mathcal{N}(0, 1/4)$. Hence, only the first covariate contributes to the response.

The regression model :

$$Y_i \quad = \quad \sum_{j=1}^{9} X_{i,j}\beta_j + \varepsilon_i$$

is fitted to the artificial data using `R`. This yields the regression parameter estimates:

$$\hat{\boldsymbol{\beta}}^\top \quad = \quad (0.048, -2.386, -5.528, 6.243, -4.819, 0.760, -3.345, -4.748, 2.136).$$

As $\boldsymbol{\beta}^\top = (1, 0, \ldots, 0)$, many regression coefficient are clearly over-estimated.

The fitted values $\hat{Y}_i = \mathbf{X}_i\hat{\boldsymbol{\beta}}$ are plotted against the values of the first covariates in the right bottom panel of Figure 1.3. As a reference the line $x = y$ is added, which represents the 'true' model. The fitted model follows the 'true' relationship. But it also captures the deviations from this line that represent the errors. □

1.5 Mean squared error

Previously, we motivated the ridge estimator as *i)* an ad hoc solution to collinearity, *ii)* a minimizer of a penalized sum of squares. An alternative motivation comes from studying the Mean Squared Error (MSE) of the ridge regression estimator: for a suitable choice of λ the ridge regression estimator may outperform the ML regression estimator in terms of the MSE. Before we prove this, we first derive the MSE of the ridge estimator and quote some auxiliary results.

Recall that (in general) for any estimator of a parameter θ:

$$\text{MSE}(\hat{\theta}) \;=\; E[(\hat{\theta}-\theta)^2] \;=\; \text{Var}(\hat{\theta}) + [\text{Bias}(\hat{\theta})]^2.$$

Hence, the MSE is a measure of the quality of the estimator.

The MSE of the ridge estimator is:

$$\begin{aligned}
\text{MSE}[\hat{\boldsymbol{\beta}}(\lambda)] &= E[(\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top(\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}-\boldsymbol{\beta})] \\
&= E(\hat{\boldsymbol{\beta}}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}) - E(\boldsymbol{\beta}^\top\,\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}) - E(\hat{\boldsymbol{\beta}}^\top\mathbf{W}_\lambda^\top\,\boldsymbol{\beta}) + E(\boldsymbol{\beta}^\top\boldsymbol{\beta}) \\
&= E(\hat{\boldsymbol{\beta}}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}) - E(\boldsymbol{\beta}^\top\,\mathbf{W}_\lambda^\top\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}) - E(\hat{\boldsymbol{\beta}}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\boldsymbol{\beta}) + E(\boldsymbol{\beta}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\,\boldsymbol{\beta}) \\
&\quad -E(\boldsymbol{\beta}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\,\boldsymbol{\beta}) + E(\boldsymbol{\beta}^\top\,\mathbf{W}_\lambda^\top\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}) + E(\hat{\boldsymbol{\beta}}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\boldsymbol{\beta}) \\
&\quad -E(\boldsymbol{\beta}^\top\,\mathbf{W}_\lambda\,\hat{\boldsymbol{\beta}}) - E(\hat{\boldsymbol{\beta}}^\top\mathbf{W}_\lambda^\top\,\boldsymbol{\beta}) + E(\boldsymbol{\beta}^\top\boldsymbol{\beta}) \\
&= E[(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\,(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta})] \\
&\quad -\boldsymbol{\beta}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\,\boldsymbol{\beta} + \boldsymbol{\beta}^\top\,\mathbf{W}_\lambda^\top\mathbf{W}_\lambda\,\boldsymbol{\beta} + \boldsymbol{\beta}^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\boldsymbol{\beta} \\
&\quad -\boldsymbol{\beta}^\top\,\mathbf{W}_\lambda\,\boldsymbol{\beta} - \boldsymbol{\beta}^\top\mathbf{W}_\lambda^\top\,\boldsymbol{\beta} + \boldsymbol{\beta}^\top\boldsymbol{\beta} \\
&= E\big\{(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top\mathbf{W}_\lambda^\top\,\mathbf{W}_\lambda\,(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta})\big\} + \boldsymbol{\beta}^\top(\mathbf{W}_\lambda-\mathbf{I}_{p\times p})^\top(\mathbf{W}_\lambda-\mathbf{I}_{p\times p})\,\boldsymbol{\beta} \\
&= \sigma^2\,\text{tr}\big\{\mathbf{W}_\lambda\,(\mathbf{X}^\top\mathbf{X})^{-1}\,\mathbf{W}_\lambda^\top\big\} + \boldsymbol{\beta}^\top(\mathbf{W}_\lambda-\mathbf{I}_{p\times p})^\top(\mathbf{W}_\lambda-\mathbf{I}_{p\times p})\,\boldsymbol{\beta}.
\end{aligned}$$

In the last step we have used $\hat{\boldsymbol{\beta}} \sim \mathcal{N}(\boldsymbol{\beta}, \sigma^2\,[\mathbf{X}^\top\mathbf{X}]^{-1})$ and the expectation of the quadratic form of a multivariate random variable $\boldsymbol{\varepsilon} \sim \mathcal{N}(\boldsymbol{\mu}_\varepsilon, \boldsymbol{\Sigma}_\varepsilon)$ is:

$$E(\boldsymbol{\varepsilon}^\top\,\boldsymbol{\Lambda}\,\boldsymbol{\varepsilon}) \;=\; \text{tr}(\boldsymbol{\Lambda}\,\boldsymbol{\Sigma}_\varepsilon) + \boldsymbol{\mu}_\varepsilon^\top\,\boldsymbol{\Lambda}\,\boldsymbol{\mu}_\varepsilon,$$

of course replacing $\boldsymbol{\varepsilon}$ by $\hat{\boldsymbol{\beta}}$ in this expectation. The first summand in the final derived expression for $\text{MSE}[\hat{\boldsymbol{\beta}}(\lambda)]$ is the sum of the variances of the ridge estimator, while the second summand can be thought of the "squared bias" of the ridge estimator. In particular, $\lim_{\lambda\to\infty}\text{MSE}[\hat{\boldsymbol{\beta}}(\lambda)] = \boldsymbol{\beta}^\top\boldsymbol{\beta}$, which is the squared biased for an estimator that equals zero (as does the ridge estimator in the limit).

Example 1.7 *Orthonormal design matrix*
Assume the design matrix $\mathbf{X}$ is orthonormal. Then, $\text{MSE}[\hat{\boldsymbol{\beta}}] = p\,\sigma^2$ and

$$\text{MSE}[\hat{\boldsymbol{\beta}}(\lambda)] \;=\; \frac{p\,\sigma^2}{(1+\lambda)^2} + \frac{\lambda^2}{(1+\lambda)^2}\boldsymbol{\beta}^\top\boldsymbol{\beta}.$$

The latter achieves its minimum at: $\lambda = p\sigma^2/\boldsymbol{\beta}^\top\boldsymbol{\beta}$. □

The following theorem and proposition are required for the proof of the main result.

Theorem 1.1 *(Theorem 1 of Theobald, 1974)*
Let $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ be (different) estimators of $\boldsymbol{\theta}$ with second order moments:

$$\mathbf{M}_k \;=\; E[(\hat{\boldsymbol{\theta}}_k-\boldsymbol{\theta})(\hat{\boldsymbol{\theta}}_k-\boldsymbol{\theta})^\top] \qquad \text{for } k=1,2,$$

and

$$\text{MSE}(\hat{\boldsymbol{\theta}}_k) \;=\; E[(\hat{\boldsymbol{\theta}}_k-\boldsymbol{\theta})^\top\mathbf{A}(\hat{\boldsymbol{\theta}}_k-\boldsymbol{\theta})] \qquad \text{for } k=1,2,$$

where $\mathbf{A} \succeq 0$. Then, $\mathbf{M}_1 - \mathbf{M}_2 \succeq 0$ if and only if $\text{MSE}(\hat{\boldsymbol{\theta}}_1) - \text{MSE}(\hat{\boldsymbol{\theta}}_2) \geq 0$ for all $\mathbf{A} \succeq 0$.

Proposition 1.1 *(Farebrother, 1976)*
Let $\mathbf{A}$ be a $p \times p$ dimensional, positive definite matrix, $\mathbf{b}$ be a nonzero p dimensional vector, and $c \in \mathbb{R}_+$. Then, $c\mathbf{A} - \mathbf{b}\mathbf{b}^\top \succ 0$ if and only if $\mathbf{b}^\top \mathbf{A}^{-1}\mathbf{b} > c$.

We are now ready to proof the main result, formalized as Theorem 1.2, that for the some λ the ridge regression estimator yields a lower MSE than the ML regression estimator.

Theorem 1.2 *(Theorem 2 of Theobald, 1974)*
There exists $\lambda > 0$ such that $\text{MSE}[\hat{\boldsymbol{\beta}}(\lambda)] < \text{MSE}[\hat{\boldsymbol{\beta}}(0)] = \text{MSE}[\hat{\boldsymbol{\beta}}]$.

Proof The second order moment matrix of the ridge estimator is:

$$\begin{aligned}
\mathbf{M}(\lambda) &:= E[(\hat{\boldsymbol{\beta}}(\lambda) - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}}(\lambda) - \boldsymbol{\beta})^\top] \\
&= E\{\hat{\boldsymbol{\beta}}(\lambda)[\hat{\boldsymbol{\beta}}(\lambda)]^\top\} - E[\hat{\boldsymbol{\beta}}(\lambda)]\{E[\hat{\boldsymbol{\beta}}(\lambda)]\}^\top + E[\hat{\boldsymbol{\beta}}(\lambda) - \boldsymbol{\beta})]\{E[\hat{\boldsymbol{\beta}}(\lambda) - \boldsymbol{\beta})]\}^\top \\
&= \text{Var}[\hat{\boldsymbol{\beta}}(\lambda)] + E[\hat{\boldsymbol{\beta}}(\lambda) - \boldsymbol{\beta})]\{E[\hat{\boldsymbol{\beta}}(\lambda) - \boldsymbol{\beta})]\}^\top.
\end{aligned}$$

Then:

$$\begin{aligned}
\mathbf{M}(0) - \mathbf{M}(\lambda) &= \sigma^2(\mathbf{X}^\top\mathbf{X})^{-1} - \sigma^2\mathbf{W}_\lambda(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{W}_\lambda^\top] \\
&\quad -(\mathbf{W}_\lambda - \mathbf{I}_{p\times p})\boldsymbol{\beta}\boldsymbol{\beta}^\top(\mathbf{W}_\lambda - \mathbf{I}_{p\times p})^\top \\
&= \sigma^2\mathbf{W}_\lambda[2\,\lambda\,(\mathbf{X}^\top\mathbf{X})^{-2} + \lambda^2(\mathbf{X}^\top\mathbf{X})^{-3}]\mathbf{W}_\lambda^\top \\
&\quad -\lambda^2[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\boldsymbol{\beta}\boldsymbol{\beta}^\top\{[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\}^\top \\
&= \sigma^2[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}[2\,\lambda\,\mathbf{I}_{p\times p} + \lambda^2(\mathbf{X}^\top\mathbf{X})^{-1}]\{[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\}^\top \\
&\quad -\lambda^2[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\boldsymbol{\beta}\boldsymbol{\beta}^\top\{[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\}^\top \\
&= \lambda[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}[2\,\sigma^2\,\mathbf{I}_{p\times p} + \lambda\sigma^2(\mathbf{X}^\top\mathbf{X})^{-1} - \lambda\boldsymbol{\beta}\boldsymbol{\beta}^\top]\{[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}]^{-1}\}^\top
\end{aligned}$$

This is positive definite if and only if $2\,\sigma^2\,\mathbf{I}_{p\times p} + \lambda\sigma^2(\mathbf{X}^\top\mathbf{X})^{-1} - \lambda\boldsymbol{\beta}\boldsymbol{\beta}^\top \succ 0$. Hereto is suffices to show that $2\,\sigma^2\,\mathbf{I}_{p\times p} - \lambda\boldsymbol{\beta}\boldsymbol{\beta}^\top \succ 0$. By Proposition 1.1 this holds for λ such that $2\sigma^2(\boldsymbol{\beta}^\top\boldsymbol{\beta})^{-1} > \lambda$. For these λ, we thus have $\mathbf{M}(0) - \mathbf{M}(\lambda)$. Application of Theorem 1.1 now concludes the proof. ■

This result of Theobald (1974) is generalized by Farebrother (1976) to the class of design matrices $\mathbf{X}$ with $\text{rank}(\mathbf{X}) < p$.

Theorem 1.2 can be used to illustrate that the ridge regression estimator strikes a balance between the variance and bias. This is illustrated in the left panel of Figure 1.4. For small λ, the variance of the ridge estimator dominates the MSE. This may be understood when realizing that in this domain of λ the ridge estimator is close to the unbiased ML regression estimator. For large λ, the variance vanishes and the bias dominates the MSE. For small enough values of λ, the decrease in variance of the ridge regression estimator exceeds the increase in its bias. As the MSE is the sum of these two, the MSE first decreases as λ moves away from zero. In particular, as $\lambda = 0$ corresponds to the ML regression estimator, the ridge regression estimator yields a lower MSE for these values of λ. In the right panel of Figure 1.4 $\text{MSE}[\hat{\boldsymbol{\beta}}(\lambda)] < \text{MSE}[\hat{\boldsymbol{\beta}}(0)]$ for $\lambda < 7$ (roughly) and the ridge estimator outperforms the ML estimator.
Besides another motivation behind the ridge regression estimator, the use of Theorem 1.2 is limited. The optimal choice of λ depends on the quantities $\boldsymbol{\beta}$ and σ^2. These are unknown in practice. Then, the penalty parameter is chosen in a data-driven fashion by means of cross-validation (see Section 1.10.2).

Remark 1.1
Theorem 1.2 can also be used to conclude on the biasedness of the ridge regression estimator. The Gauss-Markov theorem (Rao, 1973) states (under some assumptions) that the ML regression estimator is the best linear unbiased estimator (BLUE) with the smallest MSE. As the ridge regression estimator is a linear estimator and outperforms (in terms of MSE) this ML estimator, it must be biased (for it would otherwise refute the Gauss-Markov theorem).

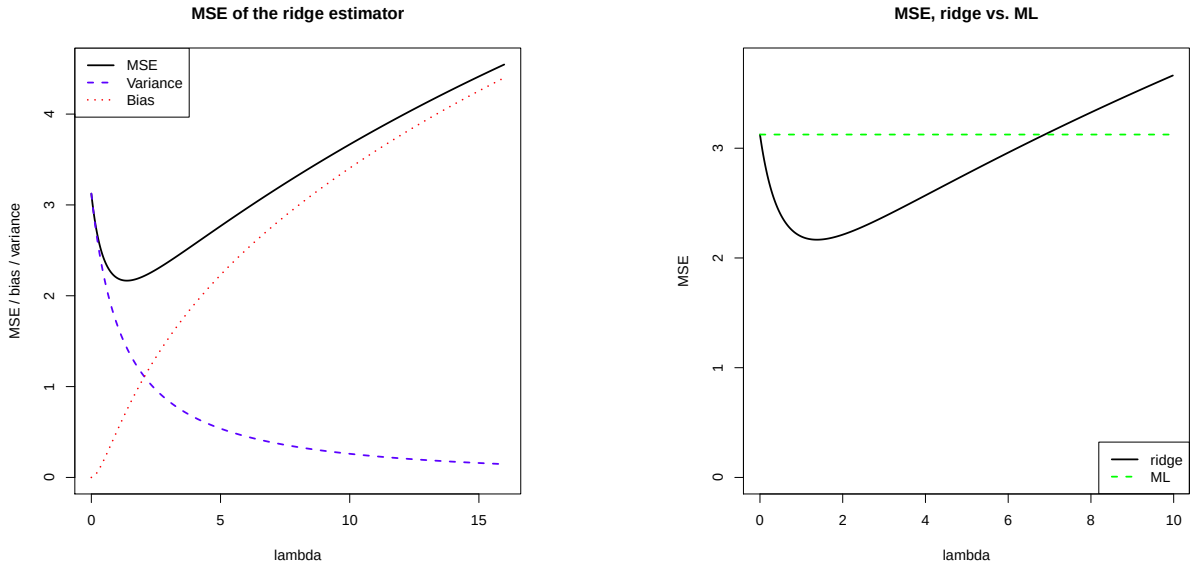

Figure 1.4: Left panel: mean squared error, and its 'bias' and 'variance' parts, of the ridge regression estimator (for artificial data). Right panel: mean squared error of the ridge and ML estimator of the regression coefficient vector (for the same artificial data).

1.6 Bayesian regression

Ridge regression has a close connection to Bayesian linear regression. Bayesian linear regression assumes the parameters $\boldsymbol{\beta}$ and σ^2 to be the random variables, while at the same time considering $\mathbf{X}$ and $\mathbf{Y}$ as fixed. Within the regression context, the conjugate priors of $\boldsymbol{\beta}$ and σ^2 are:

$$\boldsymbol{\beta} \,|\, \sigma^2 \sim \mathcal{N}(\mathbf{0}, \frac{\sigma^2}{\lambda}\mathbf{I}_{p\times p}) \qquad \text{and} \qquad \sigma^2 \sim \mathcal{IG}(\alpha_0, \beta_0),$$

where $\mathcal{IG}$ denotes the inverse Gamma distribution with shape parameter α_0 and scale parameter β_0. The penalty parameter can be interpreted as the precision of the prior, determining how informative the prior should be. A smaller penalty (i.e. precision) corresponds to a wider prior, and a larger penalty to a more informative, concentrated prior (Figure 1.5).

Under the assumption of the conjugate priors above, the joint posterior distribution of $\boldsymbol{\beta}$ and σ^2 is then:

$$\begin{aligned} & f_{\boldsymbol{\beta},\sigma^2}(\boldsymbol{\beta}, \sigma^2 \,|\, \mathbf{Y}, \mathbf{X}) \\ &\quad = \; f_Y(\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2)\, f_\beta(\boldsymbol{\beta}|\sigma^2)\, f_\sigma(\sigma^2) \\ &\quad \propto \; \sigma^{-n} \exp\Big[-\frac{1}{2\sigma^2}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})^\top(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})\Big] \\ &\qquad\quad \times\, \sigma^{-p} \exp\Big[-\frac{1}{2\sigma^2}\lambda\boldsymbol{\beta}^\top\boldsymbol{\beta}\Big] \;\times\; [\sigma^2]^{-\alpha_0-1} \exp\Big[-\frac{\beta_0}{2\sigma^2}\Big]. \end{aligned}$$

As

$$\begin{aligned} & (\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})^\top(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \lambda\boldsymbol{\beta}^\top\boldsymbol{\beta} \\ &\quad = \; \mathbf{Y}^\top\mathbf{Y} - \boldsymbol{\beta}^\top\mathbf{X}^\top\mathbf{Y} - \mathbf{Y}^\top\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}^\top\mathbf{X}^\top\mathbf{X}\boldsymbol{\beta} + \lambda\boldsymbol{\beta}^\top\boldsymbol{\beta} \\ &\quad = \; \mathbf{Y}^\top\mathbf{Y} - \boldsymbol{\beta}^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &\qquad - \mathbf{Y}^\top\mathbf{X}(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})^{-1}(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})\boldsymbol{\beta} + \boldsymbol{\beta}^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})\boldsymbol{\beta} \\ &\quad = \; \mathbf{Y}^\top\mathbf{Y} - \boldsymbol{\beta}^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})\hat{\boldsymbol{\beta}}(\lambda) \\ &\qquad - [\hat{\boldsymbol{\beta}}(\lambda)]^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})\boldsymbol{\beta} + \boldsymbol{\beta}^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})\boldsymbol{\beta} \\ &\quad = \; \mathbf{Y}^\top\mathbf{Y} - \mathbf{Y}^\top\mathbf{X}(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y} \\ &\qquad + \big[\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}(\lambda)\big]^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})\big[\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}(\lambda)\big], \end{aligned}$$

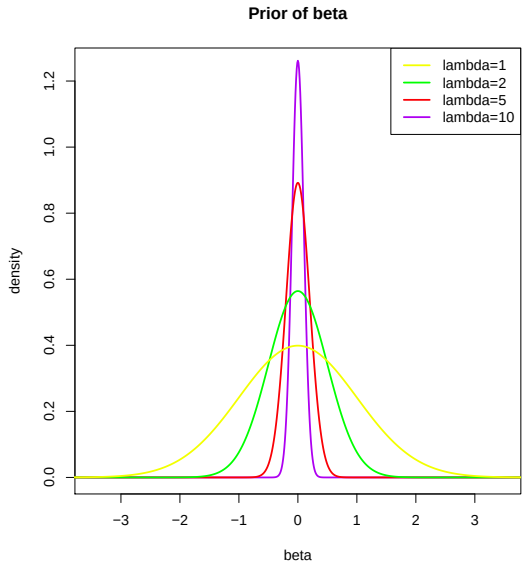

Figure 1.5: Conjugate prior of the regression parameter $\boldsymbol{\beta}$ for various choices of λ, the penalty parameters c.q. precision.

the posterior distribution can be rewritten to:

$$f_{\boldsymbol{\beta},\sigma^2}(\boldsymbol{\beta},\sigma^2 \,|\, \mathbf{Y},\mathbf{X}) \quad \propto \quad g_{\boldsymbol{\beta}}(\boldsymbol{\beta} \,|\, \sigma^2,\mathbf{Y},\mathbf{X})\, g_{\sigma^2}(\sigma^2 \,|\, \mathbf{Y},\mathbf{X})$$

with

$$g_{\boldsymbol{\beta}}(\boldsymbol{\beta} \,|\, \sigma^2,\mathbf{Y},\mathbf{X}) \quad \propto \quad \exp\Big\{ -\frac{1}{2\sigma^2}\big[\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}(\lambda)\big]^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})\big[\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}(\lambda)\big]\Big\}.$$

Then, clearly the posterior mean of $\boldsymbol{\beta}$ is $E(\boldsymbol{\beta}) = \hat{\boldsymbol{\beta}}(\lambda)$. Hence, the ridge regression estimator can be viewed as the Bayesian posterior mean estimator of $\boldsymbol{\beta}$ when imposing a Gaussian prior on the regression parameter.

1.7 Degrees of freedom

The degrees of freedom consumed by ridge regression is calculated. The degrees of freedom may be used in combination with an information criterion to decide on the value of the penalty parameter. Recall from ordinary regression that:

$$\hat{\mathbf{Y}} \quad = \quad \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} \quad = \quad \mathbf{H}\mathbf{Y},$$

where $\mathbf{H}$ is the hat matrix. The degrees of freedom used in the regression is then equal to $\mathrm{tr}(\mathbf{H})$, the trace of $\mathbf{H}$. In particular, if $\mathbf{X}$ is of full rank, i.e. $\mathrm{rank}(\mathbf{X}) = p$, then $\mathrm{tr}(\mathbf{H}) = p$.

By analogy, the ridge-version of the hat matrix is:

$$\mathbf{H}(\lambda) \quad = \quad \mathbf{X}(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top.$$

Continuing this analogy, the degrees of freedom of ridge regression is given by the trace of the ridge hat matrix $\mathbf{H}(\lambda)$:

$$\begin{aligned}\mathrm{tr}[\mathbf{H}(\lambda)] \quad &= \quad \mathrm{tr}[\mathbf{X}(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top] \\ &= \quad \sum_{j=1}^{p}\frac{d_{jj}^2}{d_{jj}^2+\lambda}.\end{aligned}$$

The degrees of freedom consumed by ridge regression is monotone decreasing in λ. In particular:

$$\lim_{\lambda \to \infty} \text{tr}[\mathbf{H}(\lambda)] \quad = \quad 0.$$

That is, in the limit no information from $\mathbf{X}$ is used. Indeed, $\boldsymbol{\beta}$ is forced to equal $\mathbf{0}_{p\times 1}$ which is not derived from data.

1.8 Eigenvalue shrinkage

The effect of the ridge penalty may also studied from the perspective of singular values. Let the singular value decomposition of the $n \times p$ design matrix $\mathbf{X}$ be:

$$\mathbf{X} \quad = \quad \mathbf{U}\mathbf{D}\mathbf{V}^\top,$$

where $\mathbf{D}$ an $n \times n$-diagonal matrix with the singular values, $\mathbf{U}$ an $n \times n$ dimensional matrix with columns containing the left singular vectors (denoted $\mathbf{u}_i$), and $\mathbf{V}$ a $p \times n$ dimensional matrix with columns containing the right singular vectors (denoted $\mathbf{v}_i$). The columns of $\mathbf{U}$ and $\mathbf{V}$ are orthogonal: $\mathbf{U}^\top\mathbf{U} = \mathbf{I}_{n\times n} = \mathbf{V}^\top\mathbf{V}$.

The OLS estimator can then be rewritten in terms of the SVD-matrices as:

$$\begin{aligned}
\hat{\boldsymbol{\beta}} &= (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} \\
&= (\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{U}\mathbf{D}\mathbf{V}^\top)^{-1}\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{Y} \\
&= (\mathbf{V}\mathbf{D}^2\mathbf{V}^\top)^{-1}\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{Y} \\
&= \mathbf{V}\mathbf{D}^{-2}\mathbf{V}^\top\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{Y} \\
&= \mathbf{V}\mathbf{D}^{-2}\mathbf{D}\mathbf{U}^\top\mathbf{Y},
\end{aligned}$$

where $\mathbf{D}^{-2}\mathbf{D}$ is not simplified further to emphasize the effect of the ridge penalty. Similarly, the ridge estimator can be rewritten in terms of the SVD-matrices as:

$$\begin{aligned}
\hat{\boldsymbol{\beta}} &= (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y} \\
&= (\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{U}\mathbf{D}\mathbf{V}^\top + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{Y} \\
&= (\mathbf{V}\mathbf{D}^2\mathbf{V}^\top + \lambda\mathbf{V}\mathbf{V}^\top)^{-1}\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{Y} \\
&= \mathbf{V}(\mathbf{D}^2 + \lambda\mathbf{I}_{n\times n})^{-1}\mathbf{V}^\top\mathbf{V}\mathbf{D}\mathbf{U}^\top\mathbf{Y} \\
&= \mathbf{V}(\mathbf{D}^2 + \lambda\mathbf{I}_{n\times n})^{-1}\mathbf{D}\mathbf{U}^\top\mathbf{Y}.
\end{aligned}$$

Combining the two results and writing $(\mathbf{D})_{jj} = d_{jj}$ we have:

$$d_{jj} \quad \geq \quad \frac{d_{jj}}{d_{jj}^2 + \lambda} \qquad \text{for all } \lambda > 0.$$

Thus, the ridge penalty shrinks the singular values.

Return to the problem of the super-collinearity of $\mathbf{X}$ in the high-dimensional setting ($p > n$). The super-collinearity implies the singularity of $\mathbf{X}^\top\mathbf{X}$ and prevents the calculation of the OLS estimator of the regression coefficients. However, $\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p}$ is non-singular, with inverse:

$$(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1} \quad = \quad \sum_{j=1}^{p}(d_{jj}^2 + \lambda)^{-1}\mathbf{v}_j\mathbf{v}_j^\top.$$

The right-hand side is well-defined for $\lambda > 0$.

1.9 Efficient calculation

In the high-dimensional setting the number of covariates p is large compared to the number of samples n. In a microarray experiment $p = 40000$ and $n = 100$ is not uncommon. To perform ridge regression in this context, the following expression needs to be evaluated numerically:

$$(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y}.$$

For $p = 40000$ this requires the inversion of a 40000×40000 dimensional matrix. This is not feasible on most desktop computers. However, there is a workaround.

Revisit the singular value decomposition of $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ and write $\mathbf{R} = \mathbf{U}\mathbf{D}$. As both $\mathbf{U}$ and $\mathbf{D}$ are $(n \times n)$-dimensional matrices, so is $\mathbf{R}$. Consequently, $\mathbf{X}$ is now decomposed as $\mathbf{X} = \mathbf{R}\mathbf{V}^\top$. The ridge estimator can be rewritten in terms of $\mathbf{R}$ and $\mathbf{V}$:

$$\begin{aligned}
\hat{\boldsymbol{\beta}} &= (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}^\top\mathbf{Y} \\
&= (\mathbf{V}\mathbf{R}^\top\mathbf{R}\mathbf{V}^\top + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{V}\mathbf{R}^\top\mathbf{Y} \\
&= (\mathbf{V}\mathbf{R}^\top\mathbf{R}\mathbf{V}^\top + \lambda\mathbf{V}\mathbf{V}^\top)^{-1}\mathbf{V}\mathbf{R}^\top\mathbf{Y} \\
&= \mathbf{V}(\mathbf{R}^\top\mathbf{R} + \lambda\mathbf{I}_{n\times n})^{-1}\mathbf{V}^\top\mathbf{V}\mathbf{R}^\top\mathbf{Y} \\
&= \mathbf{V}(\mathbf{R}^\top\mathbf{R} + \lambda\mathbf{I}_{n\times n})^{-1}\mathbf{R}^\top\mathbf{Y}.
\end{aligned}$$

Hence, the reformulated ridge estimator involves the inversion of an $(n \times n)$-dimensional matrix. With $n = 100$ this is feasible on most standard computer.

Hastie and Tibshirani (2004) point out that the number of computation operations reduces from $\mathcal{O}(p^3)$ to $\mathcal{O}(pn^2)$. In addition, they point out that this computational short-cut can be used in combination with other loss functions, for instance that of a GLM.

Avoidance of the inversion of the $p \times p$ matrix may be achieved in an other way. Hereto one needs the Woodbury identity. Let $\mathbf{A}$, $\mathbf{U}$ and $\mathbf{V}$ be $p \times p$, $p \times n$ and $n \times p$ dimensional matrices, respectively. The (simplified form of the) Woodbury identity then is:

$$(\mathbf{A} + \mathbf{U}\mathbf{V})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{I}_{n\times n} + \mathbf{V}\mathbf{A}^{-1}\mathbf{U})^{-1}\mathbf{V}\mathbf{A}^{-1}.$$

Application of the Woodbury identity to the matrix inverse in the ridge estimator of the regression parameter gives:

$$(\lambda\mathbf{I}_{p\times p} + \mathbf{X}^\top\mathbf{X})^{-1} = \frac{1}{\lambda}\mathbf{I}_{p\times p} - \frac{1}{\lambda^2}\mathbf{X}^\top(\mathbf{I}_{n\times n} + \frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}.$$

This gives:

$$\begin{aligned}
(\lambda\mathbf{I}_{p\times p} + \mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} &= \frac{1}{\lambda}\mathbf{X}^\top\mathbf{Y} - \frac{1}{\lambda^2}\mathbf{X}^\top(\mathbf{I}_{n\times n} + \frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{X}^\top\mathbf{Y} \\
&= \frac{1}{\lambda}\mathbf{X}^\top\left[\mathbf{Y} - \frac{1}{\lambda}\mathbf{X}^\top(\mathbf{I}_{n\times n} + \frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{Y}\right].
\end{aligned}$$

The inversion of the $p \times p$ dimensional matrix $\lambda\mathbf{I}_{p\times p} + \mathbf{X}^\top\mathbf{X}$ is thus replaced by that of the $n \times n$ dimensional matrix $\mathbf{I}_{n\times n} + \frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top$. In addition, this expression of the ridge regression estimator avoids the singular value decomposition of $\mathbf{X}$, which may in some cases introduce additional numerical errors (e.g. at the level of machine precision).

1.10 Choice of the penalty parameter

Throughout the introduction of ridge regression and the subsequent discussion of its properties the penalty parameter is considered known or 'given'. In practice, it is unknown and the user needs to make an informed decision on its value. Several strategies to facilitate such a decision are presented.

1.10.1 Information criterion

A popular strategy is to choose a penalty parameter that yields a good but parsimonious model. Information criteria measure the balance between model fit and model complexity. Here we present the Aikaike's information criterion (AIC), but may other criteria have been presented in the literature (e.g. Akaike, 1974, Schwarz, 1978). The AIC measures model fit by the log-likelihood and model complexity is measured by the number of parameters used by the model. The number of model parameters in regular regression simply corresponds to the number of covariates in the model. Or, by the degrees of freedom consumed by the model, which is equivalent to the trace of the hat matrix. For ridge regression it thus

seems natural to define model complexity analogously by the trace of the hat matrix. This yields the AIC for the linear regression model with ridge estimates:

$$\begin{aligned}\mathrm{AIC}(\lambda) &= 2\,p - 2\log(\hat{L}) \\ &= 2\,\mathrm{tr}[\mathbf{H}(\lambda)] - 2\log\{L[\hat{\boldsymbol{\beta}}(\lambda), \hat{\sigma}^2(\lambda)]\} \\ &= 2\sum_{j=1}^{p}\frac{d_{jj}^2}{d_{jj}^2+\lambda} + 2n\,\log[\sqrt{2\,\pi}\,\hat{\sigma}(\lambda)] + \frac{1}{\hat{\sigma}^2(\lambda)}\sum_{i=1}^{n}[y_i - \mathbf{X}_{i,*}\,\hat{\boldsymbol{\beta}}(\lambda)]^2.\end{aligned}$$

The value of λ which minimizes $\mathrm{AIC}(\lambda)$ corresponds to the 'optimal' balance of model complexity and overfitting.

Information criteria guide the decision process when having to decide among various different models. Different models use different sets of explanatory variables to explain the behaviour of the response variable. In that sense, the use of information criteria for the deciding on the ridge penalty parameter may be considered inappropriate: ridge regression uses the same set of explanatory variables irrespective of the value of the penalty parameter. Moreover, often ridge regression is employed to predict a response and not to provide an insightful explanatory model. The latter need not yield the best predictions. Finally, empirically we observe that the AIC often does not show an optimum *inside* the domain of the ridge penalty parameter. Henceforth, we refrain from the use of the AIC (or any of its relatives) in determining the optimal ridge penalty parameter.

1.10.2 Cross-validation

Instead of choosing the penalty parameter to balance model fit with model complexity, cross-validation requires it (i.e. the penalty parameter) to yield a model with good prediction performance. Commonly, this performance is evaluated on novel data. Novel data need not be easy to come by and one has to make do with the data at hand. The setting of 'original' and novel data is then mimicked by sample splitting: the data set is divided into two (groups of samples). One of these two data sets, called the *training set*, plays the role of 'original' data on which the model is build. The second of these data sets, called the *test set*, plays the role of the 'novel' data and is used to evaluate the prediction performance (often operationalized as the log-likelihood or the prediction error) of the model built on the training data set. This procedure (model building and prediction evaluation on training and test set, respectively) is done for a collection of possible penalty parameter choices. The penalty parameter that yields the model with the best prediction performance is to be preferred. The thus obtained performance evaluation depends on actual split of the data set. To remove this dependence the data set is split many times into a training and test set. For each split the model parameters are estimated for all choices of λ using the training data and estimated parameters are evaluated on the corresponding test set. The penalty parameter that on average over the test sets performs best (in some sense) is then selected.

When the repetitive splitting of the data set is done randomly, samples may accidently end up in a fast majority of the splits in either training or test set. Such samples may have an unbalanced influence on either model building or prediction evaluation. To avoid this K-fold cross-validation structures the data splitting. The samples are divided into K more or less equally sized subsets. In turn (at each split) one of these subsets plays the role of the test set while the union of the remaining subsets constitute the training set. Such a splitting warrants a balanced representation of each sample in both training and test set over the splits. Still the division into the K subsets involves a degree of randomness. This may be fully excluded when choosing $K = n$. This particular case is referred to as leave-one-out cross-validation (LOOCV). For illustration purposes the LOOCV procedure is detailed fully below:

0) Define a range of interest for the penalty parameter.
1) Divide the data set into training and test set comprising samples $\{1, \ldots, n\} \setminus i$ and $\{i\}$, respectively.
2) Fit the linear regression model by means of ridge estimation for each λ in the grid using the training set. This yields:

$$\hat{\boldsymbol{\beta}}_{-i}(\lambda) = (\mathbf{X}_{-i,*}^{\top}\mathbf{X}_{-i,*} + \lambda\mathbf{I}_{p\times p})^{-1}\mathbf{X}_{-i,*}^{\top}\mathbf{Y}_{-i}$$

   and the corresponding estimate of the error variance $\hat{\sigma}^2_{-i}(\lambda)$.
3) Evaluate the prediction performance of these models on the test set by $\log\{L[Y_i, \mathbf{X}_{i,*}; \hat{\boldsymbol{\beta}}_{-i}(\lambda), \hat{\sigma}^2_{-i}(\lambda)]\}$. Or, by the prediction error $|Y_i - \mathbf{X}_{i,*}\hat{\boldsymbol{\beta}}_{-i}(\lambda)|$, possibly squared.

4) Repeat steps 1) to 3) such that each sample plays the role of the test set once.
5) Average the prediction performances of the test sets at each grid point of the penalty parameter:

$$\frac{1}{n}\sum_{i=1}^{n}\log\{L[Y_i,\mathbf{X}_{i,*};\hat{\boldsymbol{\beta}}_{-i}(\lambda),\hat{\sigma}^2_{-i}(\lambda)]\}.$$

   The quantity above is called the *cross-validated log-likelihood*. It is an estimate of the prediction performance of the model corresponding to this value of the penalty parameter on novel data.
6) The value of the penalty parameter that maximizes the cross-validated log-likelihood is the value of choice.

Finally, we refer to Meijer and Goeman (2013) for efficient calculations of the LOOCV scheme in a ridge penalized context.

1.11 Simulations

Simulations are presented that illustrate properties of the ridge estimator not discussed explicitly in the previous sections of these notes.

1.11.1 Role of the variance of the covariates

In many applications of high-dimensional data the covariates are standardized prior to the execution of the ridge regression. Before we discuss whether this is appropriate, we first illustrate the effect of ridge penalization on covariates with distinct variances using simulated data.

The simulation involves one response to be (ridge) regressed on fifty covariates. Data (with $n = 1000$) for the covariates, denoted $\mathbf{X}$, are drawn from a multivariate normal distribution: $\mathbf{X} \sim \mathcal{N}(\mathbf{0}_{50\times 1}, \boldsymbol{\Sigma})$ with $\boldsymbol{\Sigma}$ diagonal and $(\boldsymbol{\Sigma})_{jj} = j/10$. From this the response is generated through $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ with $\boldsymbol{\beta} = \mathbf{1}_{50\times 1}$ and $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}_{50\times 1}, \mathbf{I}_{50\times 50})$.

With the simulated data at hand the ridge regression estimates of $\boldsymbol{\beta}$ are evaluated for a large grid of the penalty parameter λ. The resulting ridge regularization paths of the regression coefficients are plotted (Figure 1.6). All paths start ($\lambda = 0$) close to one and vanish as $\lambda \to \infty$. However, ridge regularization paths of regression coefficients corresponding to covariates with a large variance dominate those with a low variance.

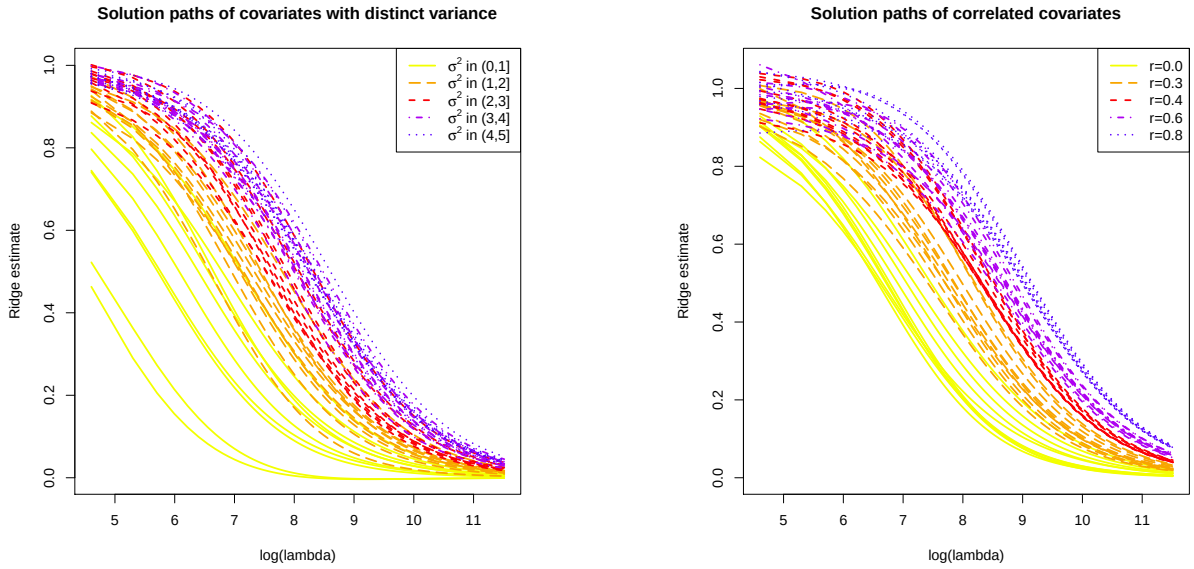

Figure 1.6: Ridge regularization paths for coefficients of the 50 covariates. Left panel: uncorrelated covariates with distinct variances. Color and line type indicated the grouping of the covariates by their variance. Right panel: correlated covariates with equal variance. Color and line type correspond to the five blocks of the covariate matrix $\boldsymbol{\Sigma}$.

Ridge regression's preference of covariates with a large variance can be understood as follows. First note that the ridge regression estimator now can be written as:

$$\begin{aligned}\boldsymbol{\beta}(\lambda) &= [\text{Var}(\mathbf{X}) + \lambda \mathbf{I}_{50\times 50}]^{-1}\text{Cov}(\mathbf{X}, \mathbf{Y}) \\ &= (\boldsymbol{\Sigma} + \lambda \mathbf{I}_{50\times 50})^{-1}\boldsymbol{\Sigma}[\text{Var}(\mathbf{X})]^{-1}\text{Cov}(\mathbf{X}, \mathbf{Y}) \\ &= (\boldsymbol{\Sigma} + \lambda \mathbf{I}_{50\times 50})^{-1}\boldsymbol{\Sigma}\boldsymbol{\beta}.\end{aligned}$$

Plug in the employed parametrization of $\boldsymbol{\Sigma}$, which gives:

$$[\boldsymbol{\beta}(\lambda)]_j = \frac{j}{j + 50\lambda}(\boldsymbol{\beta})_j.$$

Hence, the larger the covariate's variance (corresponding to the larger j), the larger its ridge regression coefficient estimate. Ridge regression thus prefers (among a set of covariates with comparable effect sizes) those with larger variances.

Should one thus standardize the covariates prior to ridge regression analysis? When dealing with gene expression data from microarrays, the data have been subjected to a series of pre-processing steps (e.g. quality control, background correction, within- and between-normalization). The purpose of these steps is to make the expression levels of genes comparable both within and between hybridizations. The preprocessing should thus be considered an inherent part of the measurement. As such it is to be done independently of whatever down-stream analysis is to follow and further tinkering with the data is preferably to be avoided (as it may mess up the 'comparable-ness' of the expression levels as achieved by the preprocessing). For other data types different considerations may apply.

Among the considerations to decide on standardization of the covariates, one should also include the fact ridge estimates prior and posterior to scaling do not simply differ by a factor. To see this assume that the covariates have been centered. Scaling of the covariates amounts to post-multiplication of the design matrix by a $p \times p$ diagonal matrix $\mathbf{A}$ with the reciprocals of the covariates' scale estimates on its diagonal (Sardy, 2008). Hence, the ridge estimator (for the rescaled data) is then given by:

$$\min_{\boldsymbol{\beta}} \|\mathbf{Y} - \mathbf{X}\mathbf{A}\boldsymbol{\beta}\|_2^2 + \lambda\|\beta\|_2^2.$$

Apply the change-of-variable $\boldsymbol{\gamma} = \mathbf{A}\boldsymbol{\beta}$ and obtain:

$$\min_{\boldsymbol{\gamma}} \|\mathbf{Y} - \mathbf{X}\gamma\|_2^2 + \lambda\|\mathbf{A}^{-1}\boldsymbol{\gamma}\|_2^2 = \min_{\boldsymbol{\beta}} \|\mathbf{Y} - \mathbf{X}\gamma\|_2^2 + \sum_{j=1}^{p} \lambda[(\mathbf{A})_{jj}]^{-2}\gamma_j^2.$$

Effectively, the scaling is equivalent to covariate-wise penalization. The 'scaled' ridge estimator may then be derived along the same lines as before in Section 1.4:

$$\hat{\boldsymbol{\beta}}^{(\text{scaled})}(\lambda) = \mathbf{A}^{-1}\hat{\boldsymbol{\gamma}}(\lambda) = \mathbf{A}^{-1}(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{A}^{-2})^{-1}\mathbf{X}^\top\mathbf{Y}.$$

In general, this is unequal to the ridge estimator without the rescaling of the columns of the design matrix. Moreover, it should be clear that $\hat{\boldsymbol{\beta}}^{(\text{scaled})}(\lambda) \neq \mathbf{A}\hat{\boldsymbol{\beta}}(\lambda)$.

1.11.2 Ridge regression and collinearity

Initially, ridge regression was motivated as an ad-hoc fix of (super)-collinear covariates in order to obtain a well-defined estimator. We now study the effect of this ad-hoc fix on the regression coefficient estimates of collinear covariates. In particular, their ridge regularization paths are contrasted to those of 'non-collinear' covariates.

To this end, we consider a simulation in which one response is regressed on 50 covariates. The data of these covariates, stored in a design matrix denoted $\mathbf{X}$, are sampled from a multivariate normal distribution, with mean zero and a 5×5 blocked covariance matrix:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} \\ \mathbf{0}_{10\times 10} & \boldsymbol{\Sigma}_{22} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} \\ \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \boldsymbol{\Sigma}_{33} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} \\ \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \boldsymbol{\Sigma}_{44} & \mathbf{0}_{10\times 10} \\ \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \mathbf{0}_{10\times 10} & \boldsymbol{\Sigma}_{55} \end{pmatrix}$$

with

$$\boldsymbol{\Sigma}_{kk} = \frac{k-1}{5}\,\mathbf{1}_{10\times 10} + \frac{6-k}{5}\,\mathbf{I}_{10\times 10}.$$

The data of the response variable $\mathbf{Y}$ are then obtained through: $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}_n, \mathbf{I}_{n\times n})$ and $\boldsymbol{\beta} = \mathbf{1}_{50\times 1}$. Hence, all covariates contribute equally to the response. Would the columns of $\mathbf{X}$ be orthogonal, little difference in the ridge estimates of the regression coefficients is expected.

The results of this simulation study with sample size $n = 1000$ are presented in Figure 1.6. All 50 regularization paths start close to one as λ is small and converge to zero as $\lambda \to \infty$. But the paths of covariates of the same block of the covariance matrix $\boldsymbol{\Sigma}$ quickly group, with those corresponding to a block with larger off-diagonal elements above those with smaller ones. Thus, ridge regression prefers (i.e. shrinks less) coefficient estimates of strongly positively correlated covariates.

Intuitive understanding of the observed behaviour may be obtained from the $p = 2$ case. Let U, V and ε be independent random variables with zero mean. Define $X_1 = U + V$, $X_2 = U - V$, and $Y = \beta_1 X_1 + \beta_2 X_2 + \varepsilon$ with β_1 and β_2 constants. Hence, $E(Y) = 0$. Then:

$$\begin{aligned} Y &= (\beta_1 + \beta_2)U + (\beta_1 - \beta_2)V + \varepsilon \\ &= \gamma_u U + \gamma_v V + \varepsilon \end{aligned}$$

and $\mathrm{Cor}(X_1, X_2) = [\mathrm{Var}(U) - \mathrm{Var}(V)]/[\mathrm{Var}(U) + \mathrm{Var}(V)]$. The random variables X_1 and X_2 are strongly positively correlated if $\mathrm{Var}(U) \gg \mathrm{Var}(V)$.

The ridge regression estimator associated with regression of Y on U and V is:

$$\boldsymbol{\gamma}(\lambda) = \begin{pmatrix} \mathrm{Var}(U) + \lambda & 0 \\ 0 & \mathrm{Var}(V) + \lambda \end{pmatrix}^{-1} \begin{pmatrix} \mathrm{Cov}(U, Y) \\ \mathrm{Cov}(V, Y) \end{pmatrix}.$$

For large enough λ

$$\boldsymbol{\gamma}(\lambda) \approx \frac{1}{\lambda} \begin{pmatrix} \mathrm{Var}(U) & 0 \\ 0 & \mathrm{Var}(V) \end{pmatrix} \begin{pmatrix} \beta_1 + \beta_2 \\ \beta_1 - \beta_2 \end{pmatrix}.$$

When $\mathrm{Var}(U) \gg \mathrm{Var}(V)$ and $\beta_1 \approx \beta_2$, the ridge estimate of γ_v vanishes for large λ. Hence, ridge regression prefers positively covariates with similar effect sizes.

The above needs some attenuation. Among others it depends on: *i)* the number of covariates in each block, *ii)* the size of the effects, i.e. regression coefficients of each covariate, and *iii)* the degree of collinearity. Possibly, there are more factors influencing the behaviour of the ridge estimator presented in this subsection.

This behaviour of ridge regression is to be understood when using (say) gene expression data to predict a certain clinical outcome. Genes work in concert to fulfil a certain function in the cell. Consequently, one expects their expression levels to be correlated. Indeed, gene expression studies exhibit many co-expressed genes, that is, genes with correlating transcript levels.

1.12 Illustration

The application of ridge regression to actual data aims to illustrates its use in practice.

1.12.1 MCM7 expression regulation by microRNAs

Recently, a new class of RNA was discovered, referred to as microRNA. MicroRNAs are non-coding, single stranded RNAs of approximately 22 nucleotides. Like mRNAs, microRNAs are encoded in and transcribed from the DNA. MicroRNAs play an important role in the regulatory mechanism of the cell. MicroRNAs down-regulate gene expression by either of two post-transcriptional mechanisms: mRNA cleavage or transcriptional repression. This depends on the degree of complementarity between the microRNA and the target. Perfect or nearly perfect complementarity of the mRNA to the microRNA will lead to cleavage and degradation of the target mRNA. Imperfect complementarity will repress the productive translation and reduction in protein levels without affecting the mRNA levels. A single microRNA can bind to and regulate many different mRNA targets. Conversely, several microRNAs can

bind to and cooperatively control a single mRNA target (Bartel, 2004; Esquela-Kerscher and Slack, 2006; Kim and Nam, 2006).

In this illustration we wish to confirm the regulation of mRNA expression by microRNAs in an independent data set. We cherry pick an arbitrary finding from literature reported in Ambs *et al.* (2008), which focusses on the microRNA regulation of the MCM7 gene in prostate cancer. The MCM7 gene is involved in DNA replication (Tye, 1999), a cellular process often derailed in cancer. Furthermore, MCM7 interacts with the tumor-suppressor gene RB1 (Sterner *et al.*, 1998). Several studies indeed confirm the involvement of MCM7 in prostate cancer (Padmanabhan *et al.*, 2004). And recently, it has been reported that in prostate cancer MCM7 may be regulated by microRNAs (Ambs *et al.*, 2008).

We here assess whether the MCM7 down-regulation by microRNAs can be observed in a data set other than the one upon which the microRNA-regulation of MCM7 claim has been based. To this end we download from the Gene Expression Omnibus (GEO) a prostate cancer data set (presented by Wang *et al.*, 2009). This data set (with GEO identifier: GSE20161) has both mRNA and microRNA profiles for all samples available. The preprocessed (as detailed in Wang *et al.*, 2009) data are downloaded and require only minor further manipulations to suit our purpose. These manipulations comprise *i)* averaging of duplicated profiles of several samples, *ii)* gene- and mir-wise zero-centering of the expression data, *iii)* averaging the expression levels of the probes that interrogate MCM7. Eventually, this leaves 90 profiles each comprising of 735 microRNA expression measurements.

Listing 1.2 R code

```
# load libraries
library(GEOquery)
library(RmiR.hsa)
library(penalized)

# extract data
slh <- getGEO("GSE20161", GSEMatrix=TRUE)
GEdata <- slh[1][[1]]
MIRdata <- slh[2][[1]]

# average duplicate profiles
Yge <- numeric()
Xmir <- numeric()
for (sName in 1:90){
  Yge <- cbind(Yge, apply(exprs(GEdata)[,sName,drop=FALSE], 1, mean))
  Xmir <- cbind(Xmir, apply(exprs(MIRdata)[,sName,drop=FALSE], 1, mean))
}
colnames(Yge) <- paste("S", 1:90, sep="")
colnames(Xmir) <- paste("S", 1:90, sep="")

# extact mRNA expression of the MCM7N tumor suppressor gene
entrezID <- c("4176")
geneName <- "MCM7"
Y <- Yge[which(levels(fData(GEdata)[,6])[fData(GEdata)[,6]] == geneName),]

# average gene expression levels over probes
Y <- apply(Y, 2, mean)

# mir-wise centering mir expression data
X <- t(sweep(Xmir, 1, rowMeans(Xmir)))

# generate cross-validated likelihood profile
profL2(Y, penalized=X, minlambda2=1, maxlambda2=20000, plot=TRUE)

# decide on the optimal penalty value directly
optLambda <- optL2(Y, penalized=X)$lambda

# obtain the ridge regression estimages
ridgeFit <- penalized(Y, penalized=X, lambda2=optLambda)
```

```
# plot them as histogram
hist(coef(ridgeFit, "penalized"), n=50, col="blue", border="lightblue", xlab="
    ridge regression estimates with optimal lambda", main="Histogram of ridge
    estimates")

#  linear prediction from ridge
Yhat <- predict(ridgeFit, X)[,1]
plot(Y ~ Yhat, pch=20, xlab="pred. MCM7 expression", ylab="obs. MCM7 expression
    ")
```

With this prostate data set at hand we now investigate whether MCM7 is regulated by microRNAs. Hereto we fit a linear regression model regressing the expression levels of MCM7 onto those of the microRNAs. As the number of microRNAs exceeds the number of samples, ordinary least squares fails and we resort to the ridge estimator of the regression coefficients. First, an informed choice of the penalty parameter is made through maximization of the LOOCV log-likelihood, resulting in $\lambda_{\text{opt}} = 1812.826$. Having decided on the value of the to-be-employed penalty parameter, the ridge regression estimator can now readily be evaluated. The thus fitted model allows for the evaluation of microRNA-regulation of MCM7. E.g., by the proportion of variation of the MCM7 expression levels by the microRNAs as expressed in coefficient of determination: $R^2 = 0.4492$. Alternatively, but closely related, observed expression levels may be related to the linear predictor of the MCM7 expression levels: $\hat{\mathbf{Y}}(\lambda_{\text{opt}}) = \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda_{\text{opt}})$. The Spearman correlation of response and predictor equals 0.6295. A visual inspection is provided by the top-left panel of Figure 1.7. Note the difference in scale of the x- and y-axes. This is due to the fact that the regression coefficients have been estimated in penalized fashion, consequently shrinking estimates of the regression coefficients towards zero leading to small estimates and in turn compressing the range of the linear prediction. The above suggests there is indeed association between the microRNA expression levels and those of MCM7.

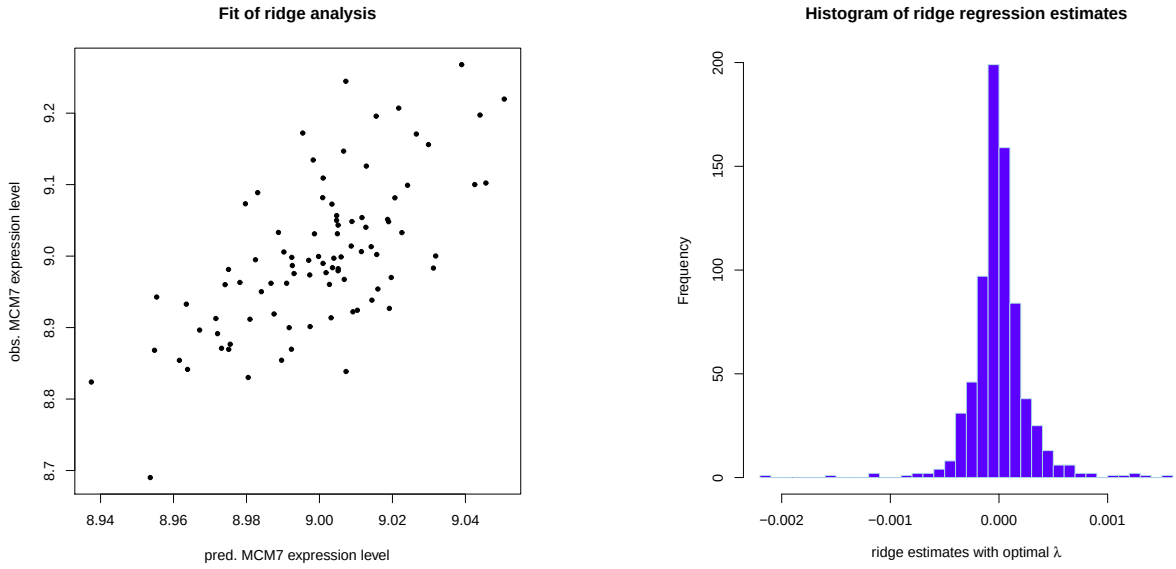

Figure 1.7: Left panel: Observed vs. (ridge) fitted MCM7 expression values. Right panel: Histogram of the ridge regression coefficient estimates.

The overall aim of this illustration was to assess whether microRNA-regulation of MCM7 could also be observed in this prostate cancer data set. In this endeavour the dogma (stating this regulation should be negative) has nowhere been used. A first simple assessment of the validity of this dogma studies the signs of the estimated regression coefficients. The ridge regression estimate has 394 out of the 735 microRNA probes with a negative coefficient. Hence, a small majority has a sign in line with the 'microRNA $\downarrow$ mRNA' dogma. When, in addition, taking the size of these coefficients into account (Figure 1.7, right panel), the negative regression coefficient estimates do not substantially differ from their positive counterparts (as can be witnessed from their almost symmetrical distribution around zero).

Hence, the value of the 'microRNA ↓ mRNA' dogma is not confirmed by this ridge regression analysis of the MCM7-regulation by microRNAs. Nor is it refuted.

The implementation of ridge regression in the `penalized`-package offers the possibility to fully obey the dogma on negative regulation of mRNA expression by microRNAs. This requires all regression coefficients to be negative. Incorporation of the requirement into the ridge estimation augments the constrained estimation problem with an additional constraint:

$$\hat{\boldsymbol{\beta}}(\lambda) \quad = \quad \arg \min_{\substack{\|\boldsymbol{\beta}\|_2^2 \leq c(\lambda) \\ \beta_j \leq 0 \text{ for all } j}} \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2.$$

With the additional non-positivity constraint on the parameters, there is no explicit solution for the estimator. The ridge estimate of the regression parameters is then found by numerical optimization using e.g. the Newton-Raphson algorithm or a gradient descent approach. The next listing gives the R-code for ridge estimation with the non-positivity constraint of the linear regression model.

Listing 1.3 R code

```
# decide on the optimal penalty value with sign constraint on paramers
optLambda <- optL2(Y, penalized=-X, positive=rep(TRUE, ncol(X)))$lambda

# obtain the ridge regression estimages
ridgeFit <- penalized(Y, penalized=-X, lambda2=optLambda, positive=rep(TRUE,
    ncol(X)))

#  linear prediction from ridge
Yhat <- predict(ridgeFit, -X)[,1]
plot(Y ~ Yhat, pch=20, xlab="predicted␣MCM7␣expression␣level", ylab="observed␣
    MCM7␣expression␣level")
cor(Y, Yhat, m="s")
summary(lm(Y ~ Yhat))[8]
```

The linear regression model linking MCM7 expression to that of the microRNAs is fitted by ridge regression while simultaneously obeying the 'negative regulation of mRNA by microRNA'-dogma to the prostate cancer data. In the resulting model 401 out of 735 microRNA probes have a nonzero (and negative) coefficient. There is a large overlap in microRNAs with a negative coefficient between those from this and the previous fit. The models are also compared in terms of their fit to the data. The Spearman rank correlation coefficient between response and predictor for the model without positive regression coefficients equals 0.679 and its coefficient of determination 0.524 (confer the left panel of 1.8 for a visualization). This is a slight improvement upon the unconstrained ridge estimated model. The improvement may be small but it should be kept in mind that the number of parameters used by both models is 401 (for the model without positive regression coefficients) vs. 735. Hence, with close to half the number of parameters the dogma-obeying model gives a somewhat better description of the data. This may suggest that there is some value in the dogma as inclusion of this prior information leads to a more parsimonious model without any loss in fit.

The dogma-obeying model selects 401 microRNAs that aid in the explanation of the variation in the gene expression levels of MCM7. There is an active field of research, called *target prediction*, trying to identify which microRNAs target the mRNA of which genes. Within `R` there is a collection of packages that provide the target prediction of known microRNAs. The packages differ on the method (e.g. experimental or sequence comparison) that has been used to arrive at the prediction. These target predictions may be used to evaluate the value of the found 401 microRNAs. Ideally, there would be a substantial amount of overlap. The `R`-script that loads the target predictions and does the comparison is below.

Listing 1.4 R code

```
# extract mir names and their (hypothesized) mrna target
mir2target <- numeric()
mirPredProgram <- c("targetscan", "miranda", "mirbase", "pictar", "mirtarget2")
for (program in mirPredProgram){
    slh <- dbReadTable(RmiR.hsa_dbconn(), program)
    slh <- cbind(program, slh[,1:2])
    colnames(slh) <- c("method", "mir", "target")
```

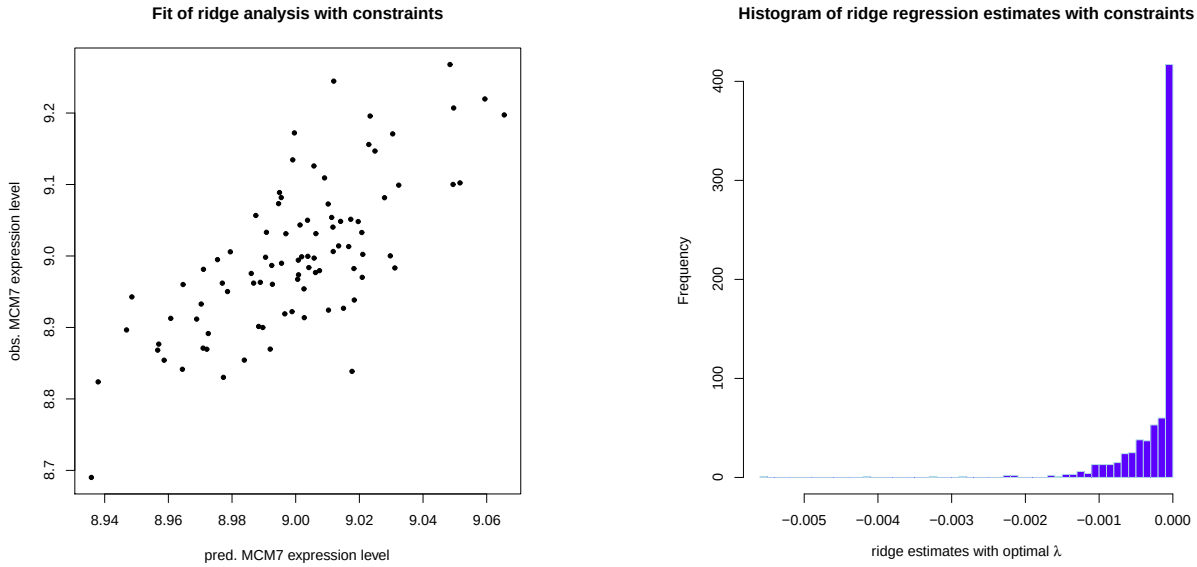

Figure 1.8: Left panel: Observed vs. (ridge) fitted MCM7 expression values (with the non-positive constraint on the parameters in place). Right panel: Histogram of the ridge regression coefficient estimates (from the non-positivity constrained analysis).

```
    mir2target <- rbind(mir2target, slh)
}
mir2target <- unique(mir2target)
mir2target <- mir2target[which(mir2target[,3] == entrezID),]
uniqMirs <- tolower(unique(mir2target[,2]))

# extract names of mir-probe on array
arrayMirs <- tolower(levels(fData(MIRdata)[,3])[fData(MIRdata)[,3]])

# which mir-probes are predicted to down-regulate MCM7
selMirs <- intersect(arrayMirs, uniqMirs)
ids <- which(arrayMirs %in% selMirs)

# which ridge estimates are non-zero
nonzeroBetas <- (coef(ridgeFit, "penalized") != 0)

# which mirs are predicted to
nonzeroPred <- 0 * betas
nonzeroPred[ids] <- 1

# contingency table and chi-square test
table(nonzeroBetas, nonzeroPred)
chisq.test(table(nonzeroBetas, nonzeroPred))
```

	$\hat{\beta}_j = 0$	$\hat{\beta}_j < 0$
microRNA not target	323	390
microRNA target	11	11

Table 1.1: Cross-tabulation of the microRNAs being a potential target of MCM7 vs. the value of its regression coefficient in the dogma-obeying model.

With knowledge available on each microRNA whether it is predicted (by at least one target prediction

package) to be a potential target of MCM7, it may be cross-tabulated against its corresponding regression coefficient estimate in the dogma-obeying model being equal to zero or not. Table 1.1 contains the result. Somewhat superfluous considering the data, we may test whether the targets of MCM7 are overrepresented in the group of strictly negatively estimated regression coefficients. The corresponding chi-squared test (with Yates' continuity correction) yields the test statistic $\chi^2 = 0.0478$ with a p-value equal to 0.827. Hence, there is no enrichment among the 401 microRNAS of those that have been predicted to target MCM7. This may seem worrisome. However, the microRNAs have been selected for their predictive power of the expression levels of MCM7. Variable selection has not been a criterion (although the sign constraint implies selection). Moreover, criticism on the value of the microRNA target prediction has been accumulating in recent years.

1.13 Conclusion

We discussed ridge regression as a modification of linear regression to overcome the empirical non-identifiability of the latter when confronted with high-dimensional data. The means to this end was the addition of a (ridge) penalty to the sum-of-squares loss function of the linear regression model, which turned out to be equivalent to constraining the parameter domain. This warranted the identification of the regression coefficients, but came at the cost of introducing bias in the estimates. Several properties of ridge regression like moments, MSE, and its Bayesian interpretation have been reviewed. Finally, its behaviour and use have been illustrated in simulation and omics data. In all, these notes hopefully present a useful overview of the potential and limitations of ridge regression.

Throughout the response variable has been assumed to be continuous. Often, this is not the case and it is e.g. binary (assuming only two values) or a survival outcome. When linking such a response to a high-dimensional set of covariates, the empirical identifiability problem is encountered. Again, penalization comes to the rescue: the ridge penalty may be combined with other link functions.

1.14 Exercises

Question 1.1 †
Find the ridge regression solution for the data below for a general value of λ and for the straight line model $Y = \beta_0 + \beta_1 X + \varepsilon$ (only apply the ridge penalty to the slope parameter, not to the intercept). Show that when λ is chosen as 0.4, the ridge solution fit is $\hat{Y} = 40 + 1.75X$. Data: $\mathbf{X}^\top = (X_1, X_2, \ldots, X_8)^\top = (-2, -1, -1, -1, 0, 1, 2, 2)^\top$, and $\mathbf{Y}^\top = (Y_1, Y_2, \ldots, Y_8)^\top = (35, 40, 36, 38, 40, 43, 45, 43)^\top$.

Question 1.2 ‡
Show that the ridge regression estimates can be obtained by ordinary least squares regression on an augmented data set. We augment the centered matrix $\mathbf{X}$ with p additional row $\sqrt{\lambda}\mathbf{I}$, and augment $\mathbf{y}$ with p zeros.

Question 1.3
The coefficients $\boldsymbol{\beta}$ of a linear regression model, $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, are estimated by $\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y}$. The associated fitted values then given by $\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y} = \mathbf{P}\mathbf{Y}$, where $\mathbf{P} = \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top$. The matrix $\mathbf{P}$ is a projection matrix and satisfies $\mathbf{P} = \mathbf{P}^2$. Hence, linear regression projects the response $\mathbf{Y}$ onto the vector space spanned by the columns of $\mathbf{Y}$. Consequently, the residuals $\hat{\boldsymbol{\varepsilon}}$ and $\hat{\mathbf{Y}}$ are orthogonal.

Now consider the ridge estimator of the regression coefficients: $\hat{\boldsymbol{\beta}}(\lambda) = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^\top\mathbf{Y}$. Let $\hat{\mathbf{Y}}(\lambda) = \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)$ be the vector of associated fitted values.

Question 1.3 a
Show that the matrix $\mathbf{Q} = \mathbf{X}(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^\top$, associated with ridge regression, is not a projection matrix (for any $\lambda > 0$).

Question 1.3 b
Show that the 'ridge fit' $\hat{\mathbf{Y}}(\lambda)$ is not orthogonal to the associated 'ridge residuals' $\hat{\boldsymbol{\varepsilon}}(\lambda)$ (for any $\lambda > 0$).

Question 1.4

† This exercise is freely rendered from Draper and Smith (1998)

‡ This exercise is freely rendered from Hastie *et al.* (2009), but can be found in many other places. The original source is unknown to the author.

The ridge penalty may be interpreted as a multivariate normal prior on the regression coefficients: $\boldsymbol{\beta} \sim \mathcal{N}(\mathbf{0}, \lambda^{-1}\mathbf{I}_{pp})$. Different priors may be considered. In case the covariates are spatially related in some sense (e.g. genomically), it may of interest to assume a first-order autoregressive prior: $\boldsymbol{\beta} \sim \mathcal{N}(\mathbf{0}, \lambda^{-1}\boldsymbol{\Sigma}_A)$, in which $\boldsymbol{\Sigma}_A$ is a $p \times p$-correlation matrix with $(\boldsymbol{\Sigma}_A)_{j_1,j_2} = \rho^{|j_1-j_2|}$ for some correlation coefficient $\rho \in [0, 1)$. Hence,

$$\boldsymbol{\Sigma}_A = \begin{pmatrix} 1 & \rho & \dots & \rho^{p-1} \\ \rho & 1 & \dots & \rho^{p-2} \\ \vdots & \vdots & \ddots & \vdots \\ \rho^{p-1} & \rho^{p-2} & \dots & 1 \end{pmatrix}.$$

Question 1.4 a
The penalized loss function associated with this AR(1) prior is:

$$\mathcal{L}(\boldsymbol{\beta}; \lambda, \boldsymbol{\Sigma}_A) = \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \boldsymbol{\beta}^\top \boldsymbol{\Sigma}_A^{-1} \boldsymbol{\beta}.$$

Find the minimizer of this loss function.

Question 1.4 b
What is the effect of ρ on the ridge estimates? Contrast this to the effect of λ. Illustrate this on (simulated) data.

Question 1.4 c
Instead of an AR(1) prior assume a prior with a uniform correlation between the elements of $\boldsymbol{\beta}$. That is, replace $\boldsymbol{\Sigma}_A$ by $\boldsymbol{\Sigma}_U$, given by:

$$\boldsymbol{\Sigma}_U = \begin{pmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{pmatrix}$$

Investigate (again on data) the effect of changing from the AR(1) to the uniform prior on the ridge regression estimates.

Question 1.5
Consider an experiment involving n cancer samples. For each sample i the transcriptome of its tumor has been profiled and is denoted $\mathbf{X}_i = (X_{i1}, \dots, X_{ip})^\top$ where X_{ij} represents the gene $j = 1, \dots, p$ in sample i. Additionally, the tumors of the samples are classified as benign and malignant.

Question 1.5 a
Write down the logistic regression model that links tumor status (as the response variable) to the expression levels.

Question 1.5 b
Specify its loss function for penalized maximum likelihood estimation of the parameters. Penalization is via the ridge penalty.

Question 1.5 c
From this loss function, derive the estimation equation for the logistic regression coefficients.

Question 1.5 d
Describe (in words) how you would find the 'ridge ML estimate'.

Question 1.6
Download the `multtest` package from BioConductor:

```
> source("http://www.bioconductor.org/biocLite.R")
```

```
> biocLite("multtest")
```
Activate the library and load leukemia data from the package:
```
> library(multtest)
> data(golub)
```
The objects `golub` and `golub.cl` are now available. The matrix-object `golub` contains the expression profiles of 38 leukemia patients. Each profile comprises expression levels of 3051 genes. The numeric-object `golub.cl` is an indicator variable for the leukemia type (AML or ALL) of the patient.

Question 1.6 a
Relate the leukemia subtype and the gene expression levels by a logistic regression model. Fit this model by means of penalized maximum likelihood, employing the ridge penalty with penalty parameter $\lambda = 1$. This is implemented in the `penalized`-packages available from `CRAN`. *Note:* center (gene-wise) the expression levels around zero.

Question 1.6 b
Obtain the fits from the regression model. The fit is almost perfect. Could this be due to overfitting the data? Alternatively, could it be that the biological information in the gene expression levels indeed determines the leukemia subtype almost perfectly?

Question 1.6 c
To discern between the two explanations for the almost perfect fit, randomly shuffle the subtypes. Refit the logistic regression model and obtain the fits. On the basis of this and the previous fit, which explanation is more plausible?

Question 1.6 d
Compare the fit of the logistic model with different penalty parameters, say $\lambda = 1$ and $\lambda = 1000$. How does λ influence the possibility of overfitting the data?

Question 1.6 e
Describe what you would do to prevent overfitting.

Question 1.7
Download the `breastCancerNKI` package from BioConductor:
```
> source("http://www.bioconductor.org/biocLite.R")
> biocLite("breastCancerNKI")
```
Activate the library and load leukemia data from the package:
```
> library(breastCancerNKI)
> data(nki)
```
The eset-object `nki` is now available. It contains the expression profiles of 337 breast cancer patients. Each profile comprises expression levels of 24481 genes. Extract the expression data from the object, remove all genes with missing values, center the gene expression gene-wise around zero, and limit the data set to the first thousand genes. The reduction of the gene dimensionality is only for computational speed.
```
X <- exprs(nki)
X <- X[-which(rowSums(is.na(X)) > 0),]
X <- apply(X[1:1000,], 1, function(X) X - mean(X) ) .
```
Furthermore, extract the estrogen receptor status (short: ER status), an important prognostic indicator for breast cancer.
```
Y <- pData(nki)[,8]
```

Question 1.7 a
Relate the ER status and the gene expression levels by a logistic regression model, which is fitted by means of ridge penalized maximum likelihood. First, find the optimal value of the penalty parameter of λ by means of cross-validation. This is implemented in `optL2`-function of the `penalized`-package available from `CRAN`.

Question 1.7 b

Evaluate whether the cross-validated likelihood indeed attains a maximum at the optimal value of λ. This can be done with the `profL2`-function of the `penalized`-package available from `CRAN`.

Question 1.7 c
Investigate the sensitivity of the penalty parameter selection with respect to the choice of the cross-validation fold.

Question 1.7 d
Does the optimal lambda produce a reasonable fit?

Bibliography

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, **19**(6), 716–723.

Ambs, S., Prueitt, R. L., Yi, M., Hudson, R. S., Howe, T. M., Petrocca, F., Wallace, T. A., Liu, C.-G., Volinia, S., Calin, G. A., Yfantis, H. G., Stephens, R. M., and Croce, C. M. (2008). Genomic profiling of microrna and messenger RNA reveals deregulated microrna expression in prostate cancer. *Cancer Research*, **68**(15), 6162–6170.

Bartel, D. P. (2004). MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell*, **116**(2), 281–297.

Draper, N. R. and Smith, H. (1998). *Applied Regression Analysis (3rd edition)*. John Wiley & Sons.

Esquela-Kerscher, A. and Slack, F. J. (2006). Oncomirs: microRNAs with a role in cancer. *Nature Reviews Cancer*, **6**(4), 259–269.

Farebrother, R. W. (1976). Further results on the mean square error of ridge regression. *Journal of the Royal Statistical Society, Series B (Methodological)*, pages 248–250.

Fletcher, R. (2008). *Practical Methods of Optimization, 2nd Edition*. John Wiley, New York.

Harville, D. A. (2008). *Matrix Algebra From a Statistician's Perspective*. Springer, New York.

Hastie, T. and Tibshirani, R. (2004). Efficient quadratic regularization for expression arrays. *Biostatistics*, **5**(3), 329–340.

Hastie, T., Friedman, J., and Tibshirani, R. (2009). *The elements of statistical learning*. Springer.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, **12**(1), 55–67.

Kim, V. N. and Nam, J.-W. (2006). Genomics of microRNA. *TRENDS in Genetics*, **22**(3), 165–173.

Meijer, R. J. and Goeman, J. J. (2013). Efficient approximate k-fold and leave-one-out cross-validation for ridge regression. *Biometrical Journal*, **55**(2), 141–155.

Padmanabhan, V., Callas, P., Philips, G., Trainer, T., and Beatty, B. (2004). DNA replication regulation protein MCM7 as a marker of proliferation in prostate cancer. *Journal of Clinical Pathology*, **57**(10), 1057–1062.

Pust, S., Klokk, T., Musa, N., Jenstad, M., Risberg, B., Erikstein, B., Tcatchoff, L., Liestøl, K., Danielsen, H., Van Deurs, B., and K, S. (2013). Flotillins as regulators of ErbB2 levels in breast cancer. *Oncogene*, **32**(29), 3443–3451.

Rao, C. R. (1973). *Linear statistical inference and its applications*. John Wiley & Sons.

Sardy, S. (2008). On the practice of rescaling covariates. *International Statistical Review*, **76**(2), 285–297.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, **6**(2), 461–464.

Sterner, J. M., Dew-Knight, S., Musahl, C., Kornbluth, S., and Horowitz, J. M. (1998). Negative regulation of DNA replication by the retinoblastoma protein is mediated by its association with MCM7. *Molecular and Cellular Biology*, **18**(5), 2748–2757.

Theobald, C. M. (1974). Generalizations of mean square error applied to ridge regression. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 103–106.

Tye, B. K. (1999). MCM proteins in DNA replication. *Annual Review of Biochemistry*, **68**(1), 649–686.

Wang, L., Tang, H., Thayanithy, V., Subramanian, S., Oberg, L., Cunningham, J. M., Cerhan, J. R., Steer, C. J., and Thibodeau, S. N. (2009). Gene networks and microRNAs implicated in aggressive prostate cancer. *Cancer research*, **69**(24), 9490–9497.