High Dimensional Data Analysis Course Project

Luca Galli – Enrico Mannarino – Christian Persico

Master's Degree in Data Science

Introduction to ridge regression

Linear regression is a conditional model that estimates the linear relationship between a scalar response variable and one or more explanatory variables.

There are some specific cases in which linear regression is not used:

- $rank(\mathbf{X}) < p$ which makes $\mathbf{X}^T\mathbf{X}$ not invertible, producing infinite solutions that perfectly fit the data;
- Multicollinearity between predictors, which can make $\mathbf{X}^T\mathbf{X}$ ill-conditioned (close to singularity);
- $n \simeq p$ which causes high variability and overfitting, leading to poor predictions.

The phenomenon of multicollinearity can be found in high-dimensional settings, where the number of columns or features is much greater than the number of rows or instances. A common solution is Ridge Regression, which abandons the unbiased estimator requirement of the OLS regression model and adds a penalization parameter λ to our estimator.

The Ridge Regression solution involves shrinking the estimated coefficients towards zero, proportionally to the magnitude of λ. This causes an increase of the error bias, but leads to a lower variance, generally implying a lower error with respect to the OLS estimator. The optimal value for λ is commonly obtained through Cross Validation.

The ridge regression estimator minimizes the ridge loss function:

$$\hat{\beta}(\lambda) = arg\min_{\beta} \sum_{i=1}^{n} (y_i - x_i\beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

Its solution in matrix form is $\hat{\boldsymbol{\beta}}_\lambda = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y}$, which is analogous to the OLS estimator when $\lambda = 0$: $\hat{\boldsymbol{\beta}}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$.

The $\hat{\beta}$ parameters computed with the ridge method are called *penalized parameters*, due to the penalization that is affecting them. The intercept, instead, does not need to be penalized, because this would bias our model's predictions away from the $\bar{y}$ (sample mean) in extreme cases when all model parameters are shrunk towards 0. Additionally, being ridge regression not scale-invariant, centering or standardizing the variables is recommended to ensure accurate coefficient estimates.

Note: the R function lm.ridge(), available in the MASS package, does not accept a single covariate as input. Consequently, for the exercises, we opted to perform individual matrix multiplications step by step instead of relying on that function.

Linear Regression	Ridge Regression
$arg\min_{\beta} \sum_{i=1}^{n}(y_i - x_i\beta)^2$	$arg\min_{\beta} \sum_{i=1}^{n}(y_i - x_i\beta)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$
$\hat{\boldsymbol{\beta}}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$	$\hat{\boldsymbol{\beta}}_{\lambda} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y}$
Unbiased	Biased
Higher Variance	Lower Variance

Ridge regression exercises

The following exercises are taken from section 1.12 of *van Wieringen (2023). Lecture notes on ridge regression.*

Question 1.2

Consider the simple linear regression model $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$ with $\epsilon_i \sim N(0, \sigma^2)$. The data on the covariate and response are: $\mathbf{X}^T = (X_1, X_2, ..., X_8)^T = (-2, -1, -1, -1, 0, 1, 2, 2)^T$ and $\mathbf{Y}^T = (Y_1, Y_2, ..., Y_8)^T = (35, 40, 36, 38, 40, 43, 45, 43)^T$, with corresponding elements in the same order.

a) Find the ridge regression estimator for the data above for a general value of λ.

b) Evaluate the fit, i.e. $\hat{Y}_i(\lambda)$ for $\lambda = 10$. Would you judge the fit as good? If not, what is the most striking feature that you find unsatisfactory?

c) Now zero center the covariate and response data, denote it by $\tilde{X}_i$ and $\tilde{Y}_i$, and evaluate the ridge regression estimator of $\tilde{Y}_i = \beta_1\tilde{X}_i + \epsilon_i$ at $\lambda = 4$. Verify that in terms of original data the resulting predictor now is: $\hat{Y}_i(\lambda) = 40 + 1.75X$.

Note that the employed estimate in the predictor found in part *c)* is effectively a combination of a maximum likelihood and ridge regression one for intercept and slope, respectively. Put differently, only the slope has been shrunken.

Solution

a)

First we calculate $\mathbf{X}^T\mathbf{X}$ (also taking into account the intercept in the result) and $\mathbf{X}^T\mathbf{Y}$. These are given by:

$$\mathbf{X}^T\mathbf{X} = \begin{pmatrix} 8 & 0 \\ 0 & 16 \end{pmatrix} \qquad \mathbf{X}^T\mathbf{Y} = \begin{pmatrix} 320 \\ 35 \end{pmatrix}.$$

The ridge estimator for a general value of λ - with an unpenalized intercept - is as follows:

$$
\begin{aligned}
\hat{\boldsymbol{\beta}}_\lambda &= (\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y} \\
&= \begin{pmatrix} 8 & 0 \\ 0 & 16+\lambda \end{pmatrix}^{-1} \begin{pmatrix} 320 \\ 35 \end{pmatrix} \\
&= \begin{pmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{16+\lambda} \end{pmatrix} \begin{pmatrix} 320 \\ 35 \end{pmatrix} \\
&= \begin{pmatrix} 40 \\ \\ \frac{35}{16+\lambda} \end{pmatrix}.
\end{aligned} \tag{1}
$$

b)

If $\lambda = 10$: $\hat{\boldsymbol{\beta}}_\lambda = \begin{pmatrix} 40 \\ \\ \frac{35}{16+\lambda} \end{pmatrix} = \begin{pmatrix} 40 \\ \\ \frac{35}{16+\mathbf{10}} \end{pmatrix} = \begin{pmatrix} 40 \\ 1.35 \end{pmatrix}$

Below are the calculations performed using R:

```
x <- matrix(c(-2,-1,-1,-1,0,1,2,2))
y <- matrix(c(35,40,36,38,40,43,45,43))

n <- nrow(x)
p <- ncol(x)

lambda <- 10

intercept <- matrix(rep(1,8))

(hatbetas.without.intercept <- c(solve(t(intercept) %*% intercept * diag(p)) %*%
        t(intercept) %*%y, solve(t(x) %*% x + lambda * diag(p)) %*% t(x) %*% y))
```

```
## [1] 40.000000  1.346154
```

We see that the results match.

To compare with the model estimate with penalized intercept, we show the following:

```
(hatbetas.with.intercept <- c(solve(t(intercept) %*% intercept + lambda * diag(p)) %*%
           t(intercept) %*%y, solve(t(x) %*% x + lambda * diag(p)) %*% t(x) %*% y))
```

```
## [1] 17.777778  1.346154
```

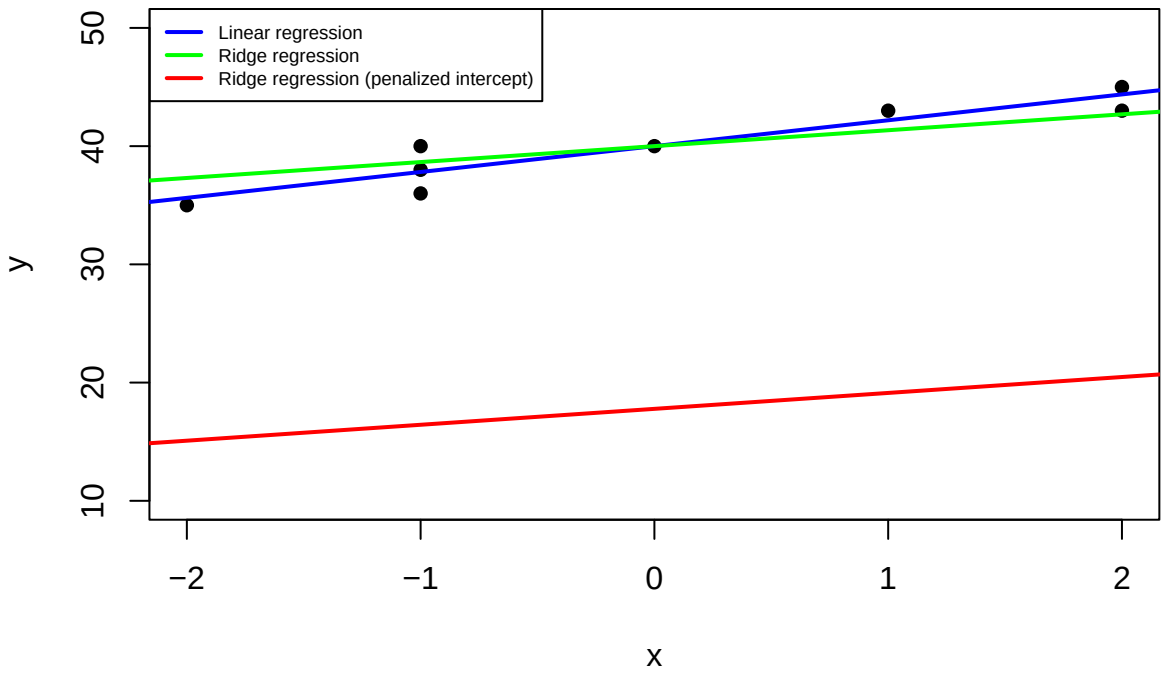
The intercept estimate changed considerably after the ridge penalty (from 40 to 17.78).

We visualise the estimated functions using the classical linear model as a benchmark:

```r
plot(x, y, pch=16, ylim = c(10,50))
abline(hatbetas.with.intercept[1], hatbetas.with.intercept[2], col="red", lwd=2)

mod <- lm(y ~ x)
(hatbetas.lm <- coef(mod))
```

```
## (Intercept)           x
##     40.0000      2.1875
```

```r
abline(hatbetas.lm[1], hatbetas.lm[2], col="blue", lwd=2)

abline(hatbetas.without.intercept[1], hatbetas.without.intercept[2], col="green", lwd=2)

legend("topleft", legend = c("Linear regression", "Ridge regression",
             "Ridge regression (penalized intercept)"),
             col = c("blue", "green", "red"), lty = 1, lwd = 2, cex = 0.6)
```



The fit of the ridge regression model with $\lambda = 10$ and non-penalised intercept is good. In fact, it deviates little from the estimated linear model, which minimises the sum of squares of the residuals. This is not the case for the estimated ridge model that penalises the intercept as the fit is far removed from the real function representing our sample data.

It is important to note that we are not in a high dimensionality context - only 8 observations and 1 covariate.

Had we been in a context with more data and covariates, one could have assessed the fit of the model by considering different values of λ and evaluating the MSE.

c)

Centring the covariate and response data on zero and evaluating the ridge regression model for $\lambda = 4$, we have that:

```
x.center <- as.matrix(scale(x, scale = F)[,])
y.center <- scale(y, scale = F)[,]

lambda <- 4

(hatbeta.1.center <- solve(t(x.center) %*% x.center + lambda * diag(ncol(x.center))) %*%
                          t(x.center) %*% y.center)
```

```
##       [,1]
## [1,] 1.75
```

The estimated ridge regression coefficient $\hat{\beta}_1(\lambda)$ is 1.75.

Let us now see the case considering the original data:

```
(hatbeta.1 <- solve(t(x) %*% x + lambda * diag(ncol(x))) %*% t(x) %*% y)
```

```
##       [,1]
## [1,] 1.75
```

In terms of original data the resulting predictor now is: $\hat{Y}_i(\lambda) = 40 + 1.75X$ (the intercept is estimated via OLS and remains the same as before). This is because the values of the covariate were already centred on zero, so the ridge regression model, which is not scale invariant, did not change.

Question 1.3

Consider the simple linear regression model $Y_i = \beta_0 + X_i\boldsymbol{\beta} + \epsilon_i$ for $i = 1, ..., n$ and with $\epsilon_i \sim_{i.i.d.} N(0, \sigma^2)$ The model comprises a single covariate and an intercept. Response and covariate data are: $\{(y_i, x_i)\}_{i=1}^4 = \{(1.4, 0.0), (1.4, -2.0), (0.8, 0.0), (0.4, 2.0)\}$. Find the value of λ that yields the ridge regression estimate (with an unregularized/unpenalized intercept as is done in part *c)* of Question 1.2) equal to $(1, -\frac{1}{8})^T$.

Solution

As before, also estimating the intercept via OLS, we first calculate $\mathbf{X}^T\mathbf{X}$ and $\mathbf{X}^T\mathbf{Y}$:

$$\mathbf{X}^T\mathbf{X} = \begin{pmatrix} 4 & 0 \\ 0 & 8 \end{pmatrix} \qquad \mathbf{X}^T\mathbf{Y} = \begin{pmatrix} 4 \\ -2 \end{pmatrix}.$$

Applying the penalty only to the covariate and solving by keeping λ unknown, the ridge estimator is:

$$
\begin{aligned}
\hat{\beta}_\lambda &= (\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y} \\
&= \begin{pmatrix} 4 & 0 \\ 0 & 8+\lambda \end{pmatrix}^{-1} \begin{pmatrix} 4 \\ -2 \end{pmatrix} \\
&= \begin{pmatrix} \frac{1}{4} & 0 \\ 0 & \frac{1}{8+\lambda} \end{pmatrix} \begin{pmatrix} 4 \\ -2 \end{pmatrix} \\
&= \begin{pmatrix} 1 \\ -\frac{2}{8+\lambda} \end{pmatrix}.
\end{aligned} \tag{2}
$$

Since we want the ridge estimate to be equal to $(1, -\frac{1}{8})^T$, we equate and solve for λ: $-\frac{2}{8+\lambda} = -\frac{1}{8}$. Then $\lambda = 8$.

Question 1.4

Plot the regularization path of the ridge regression estimator over the range $\lambda \in (0, 20.000]$ using the data of Example 1.2.

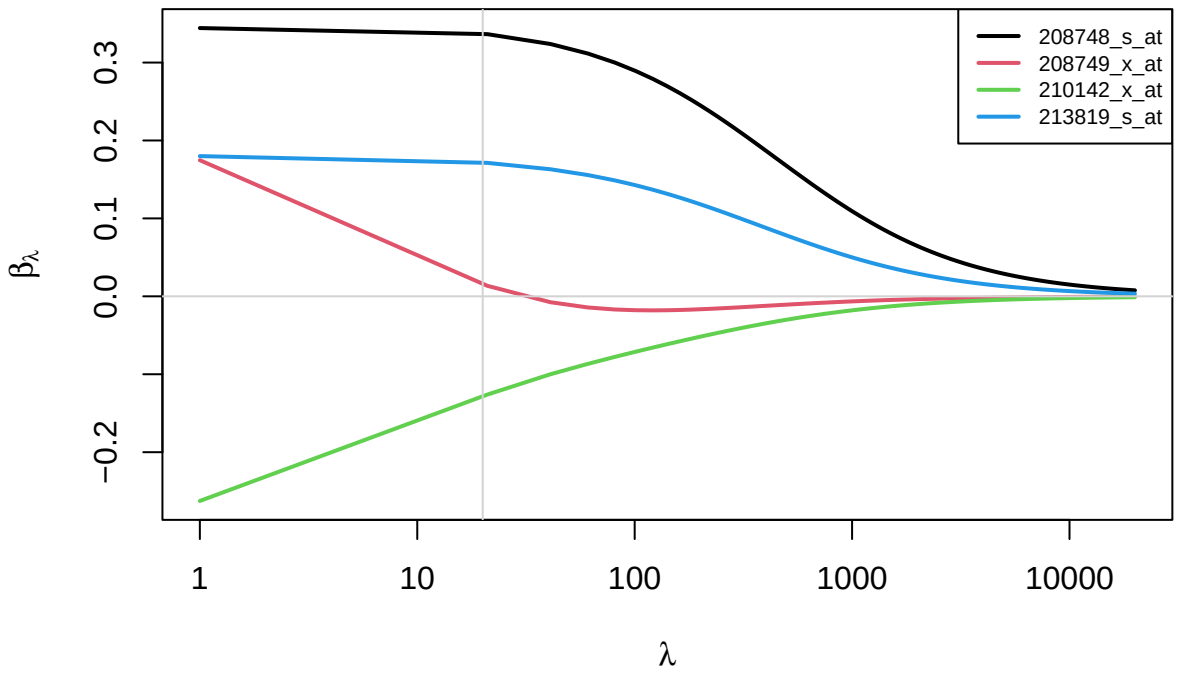
Solution

The dataset is about gene expression 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. After centering, the expression levels of the first ERBB2 probe are regressed on those of the four FLOT-1 probes (then we consider only 344 observations and 4 covariates).

So, we plot the regularization path of the ridge regression estimator over the range $\lambda \in (0, 20.000]$, using a logarithmic scale for the λ values to highlight the behaviour of the curves:

```
# if (!require("BiocManager", quietly = TRUE))
#   install.packages("BiocManager")

# BiocManager::install("breastCancerVDX")
# BiocManager::install("Biobase")

library(Biobase)
library(breastCancerVDX)

# import data
data(vdx)

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

```r
# 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))

Y <- t(exprs(vdx)[idERBB2,])
Y <- sweep(Y, 2, colMeans(Y))


lambdas <- c(seq(1, 20000, length.out = 1000))
hatbetas <- sapply(lambdas, function(lambda) solve(t(X) %*% X + lambda * diag(ncol(X)))
                                    %*% t(X) %*% Y[,1])

matplot(lambdas, t(hatbetas), log="x", type = "l", lty = 1, lwd = 2,
                ylab = expression(widehat(beta)[lambda]), xlab=expression(lambda))
legend("topright", legend = colnames(X), cex = 0.7, lty = 1, lwd = 2, col = 1:4)

abline(h=0, v=20, col = "lightgray")
```




```r
# ALTERNATIVE WAY
# mod <- lm.ridge(Y[,1] ~ X, lambda = seq(1, 20000, length.out = 1000))
# plot(mod)
```

The ridge penalty has a concrete effect starting from $\lambda > 20$ or so. In particular, as λ increases (from left to right along the x-axis), the ridge regression coefficient estimates shrink towards zero. When λ is extremely large, then all of the ridge coefficient estimates are basically zero. The variable 208748_*s_at* (black curve) is arguably the most important, which is the one receiving less shrinkage compared to the others.

The coefficient of 208749_*x_at* (red curve) is positive at the beginning and then becomes negative for large values of λ. This can probably be due to correlation with other variables, causing instability in the estimated coefficients.

Let us check this:

```
cor(X)
```

```
##             208748_s_at 208749_x_at 210142_x_at 213819_s_at
## 208748_s_at  1.00000000  0.02617216 -0.08468458   0.2093071
## 208749_x_at  0.02617216  1.00000000  0.91034538  -0.0921865
## 210142_x_at -0.08468458  0.91034538  1.00000000  -0.1222137
## 213819_s_at  0.20930712 -0.09218650 -0.12221375   1.0000000
```

As hypothesised, there is a very high positive correlation (0.91) between the variables 208749_*x_at* (red curve) and 210142_*x_at* (green curve). Therefore, the ridge regression may try to distribute the weight between them during the estimation process. This balancing could cause the coefficient estimates of the linearly associated variables to become concordant in sign (as it's noticeable on the graph from $\lambda > 50$ or so).

Question 1.6

Show that the ridge regression estimator can be obtained by ordinary least squares regression on an augmented data set. Hereto augment the matrix $\mathbf{X}$ with p additional rows $\sqrt{\lambda}\mathbf{I}_{pp}$, and augment the response vector $\mathbf{Y}$ with p zeros.

Solution

Denote by $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ the augmented data, i.e.,

$$\tilde{\mathbf{X}} = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda}\mathbf{I}_{pp} \end{pmatrix}, \qquad \tilde{\mathbf{Y}} = \begin{pmatrix} \mathbf{Y} \\ \mathbf{0}_{p1} \end{pmatrix}.$$

The least squares estimates of the regression parameters for the augmented data are given by:

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

Using the definition of $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ we have:

$$\tilde{\mathbf{X}}^T\tilde{\mathbf{X}} = \begin{pmatrix} \mathbf{X}^T & \sqrt{\lambda}\mathbf{I}_{pp} \end{pmatrix} \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda}\mathbf{I}_{pp} \end{pmatrix} = \mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp}$$

and

$$\tilde{\mathbf{X}}^T\tilde{\mathbf{Y}} = \begin{pmatrix}\mathbf{X}^T & \sqrt{\lambda}\mathbf{I}_{pp}\end{pmatrix}\begin{pmatrix}\mathbf{Y}\\ \mathbf{0}_{p1}\end{pmatrix} = \mathbf{X}^T\mathbf{Y}$$

So, $\hat{\boldsymbol{\beta}}_{new} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y}$.

Let's make an example in R (using data from Exercise 1.2).

```
x <- matrix(c(0.0, -2.0, 0.0, 2.0))
y <- matrix(c(1.4, 1.4, 0.8, 0.4))
```

Let's perform a ridge regression with $\lambda = 8$ and print the coefficient $\hat{\beta}_1$:

```
lambda <- 8
hatbetas.rid <- c(solve(t(x) %*% x + lambda * diag(ncol(x))) %*% t(x) %*% y)
hatbetas.rid
```

```
## [1] -0.125
```

Now let's see if we obtain the same coefficient by performing a linear regression on augmented data:

```
x_aug <- rbind(x, sqrt(lambda))
y_aug <- rbind(y, 0)

hatbetas.lin <- c(solve(t(x_aug) %*% x_aug) %*% t(x_aug) %*% y_aug)
hatbetas.lin
```

```
## [1] -0.125
```

Question 1.8

The coefficients $\boldsymbol{\beta}$ of a linear regression model, $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \epsilon$, are estimated by $\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$. The associated fitted values then given by $\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} = \mathbf{H}\mathbf{Y}$, where $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ referred to as the hat matrix. The hat matrix $\mathbf{H}$ is a projection matrix as it satisfies $\mathbf{H} = \mathbf{H}^2$. Hence, linear regression projects the response $\mathbf{Y}$ onto the vector space spanned by the columns of $\mathbf{Y}$. Consequently, the residuals $\hat{\epsilon}$ and $\hat{\mathbf{Y}}$ are orthogonal. Now consider the ridge estimator of the regression coefficients: $\hat{\boldsymbol{\beta}}(\lambda) = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y}$. Let $\hat{\mathbf{Y}}(\lambda) = \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)$ be the vector of associated fitted values.

a) Show that the ridge hat matrix $\mathbf{H}(\lambda) = \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T$, associated with ridge regression, is not a projection matrix (for any $\lambda > 0$), i.e. $\mathbf{H}(\lambda) = [\mathbf{H}(\lambda)]^2$.

b) Show that for any $\lambda > 0$ the 'ridge fit' $\hat{\mathbf{Y}}(\lambda)$ is not orthogonal to the associated 'ridge residuals' $\hat{\epsilon}(\lambda)$, defined as $\epsilon(\lambda) = \mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)$.

Solution

a)

A projection matrix $\mathbf{H}(\lambda)$ should be idempotent and satisfy: $\mathbf{H}(\lambda) = [\mathbf{H}(\lambda)]^2$. Let's check it out:

$$\begin{aligned}
[\mathbf{H}(\lambda)]^2 &= \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T \\
&= \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T - \lambda\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T \\
&= \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T - \lambda\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-2}\mathbf{X}^T \\
&= \mathbf{H}(\lambda) - \lambda\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-2}\mathbf{X}^T \\
&\neq \mathbf{H}(\lambda).
\end{aligned} \tag{3}$$

Hence $\mathbf{H}(\lambda)$ is not a projection matrix (unless $\lambda = 0$).

Note that for the calculation we have added and subtracted λ in the middle term.

b)

The ridge fit is given by $\hat{\mathbf{Y}}(\lambda) = \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y} = \mathbf{H}\mathbf{Y}$ and the associated residuals by: $\hat{\boldsymbol{\epsilon}}(\lambda) = \mathbf{Y} - \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y} = [\mathbf{I}_{pp} - \mathbf{H}]\mathbf{Y}$. If the residual and the fit were orthogonal, their inner product would vanish: $\langle\hat{\mathbf{Y}}(\lambda), \hat{\boldsymbol{\epsilon}}(\lambda)\rangle = 0$. Let's check it out:

$$\begin{aligned}
\langle\hat{\mathbf{Y}}(\lambda), \hat{\boldsymbol{\epsilon}}(\lambda)\rangle &= [\hat{\mathbf{Y}}(\lambda)]^T\hat{\boldsymbol{\epsilon}}(\lambda) \\
&= [\mathbf{H}\mathbf{Y}]^T[\mathbf{I}_{pp} - \mathbf{H}]\mathbf{Y} \\
&= \mathbf{Y}^T\mathbf{H}^T(\mathbf{I}_{pp} - \mathbf{H})\mathbf{Y} \\
&= \mathbf{Y}^T(\mathbf{H}^T - \mathbf{H}^T\mathbf{H})\mathbf{Y} \\
&= \mathbf{Y}^T(\mathbf{H} - \mathbf{H}^2)\mathbf{Y},
\end{aligned} \tag{4}$$

where we have used the symmetry property of $\mathbf{H}(\lambda)$ but, since it is not idempotent ($\mathbf{H}(\lambda) \neq [\mathbf{H}(\lambda)]^2$) as demonstrated in the previous point a), then: $\langle\hat{\mathbf{Y}}(\lambda), \hat{\boldsymbol{\epsilon}}(\lambda)\rangle \neq 0$ for any $\lambda > 0$.

Question 1.20

Consider the standard linear regression model $Y_i = \mathbf{X}_{i,*}\boldsymbol{\beta} + \epsilon_i$ for $i = 1, ..., n$ and with the ϵ_i i.i.d. normally distributed with zero mean and a common but unknown variance. Information on the response, design matrix and relevant summary statistics are:

$$\mathbf{X}^T = (2 \quad 1 - 2), \quad \mathbf{Y}^T = (-1 - 1 \quad 1), \quad \mathbf{X}^T\mathbf{X} = (9), \quad \text{and} \quad \mathbf{X}^T\mathbf{Y} = (-5),$$

from which the sample size and dimension of the covariate space are immediate.

a) Evaluate the ridge regression estimator $\hat{\boldsymbol{\beta}}(\lambda)$ with $\lambda = 1$.

b) Evaluate the variance of the ridge regression estimator, i.e. $\hat{Var}[\hat{\boldsymbol{\beta}}(\lambda)]$, for $\lambda = 1$. In this the error variance σ^2 is estimated by $n^{-1}||\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)||_2^2$.

c) Recall that the ridge regression estimator $\hat{\boldsymbol{\beta}}(\lambda)$ is normally distributed. Consider the interval

$$C = (\hat{\boldsymbol{\beta}}(\lambda) - 2\{\hat{Var}[\hat{\beta}(\lambda)]\}^{1/2}, \hat{\boldsymbol{\beta}}(\lambda) + 2\{\hat{Var}[\hat{\beta}(\lambda)]\}^{1/2}).$$

Is this a genuine (approximate) 95% confidence interval for $\boldsymbol{\beta}$? If so, motivate. If not, what is the interpretation of this interval?

d) Suppose the design matrix is augmented with an extra column identical to the first one. Moreover, assume λ to be fixed. Is the estimate of the error variance unaffected, or not? Motivate.

Solution

a)

Since $\hat{\boldsymbol{\beta}}(\lambda) = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y}$, for $\lambda = 1$ the ridge regression estimator is:

$$\hat{\boldsymbol{\beta}}(\lambda) = (9 + 1 \cdot 1)^{-1}(-5) = \frac{1}{10} \cdot (-5) = -0.5 \tag{5}$$

b)

Given that $\hat{\sigma}^2 = n^{-1}||\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)||_2^2$, the estimated error variance is:

$$\begin{aligned}
\hat{\sigma}^2 &= 3^{-1}[(-1 + 2 \cdot 0.5)^2 + (-1 + 1 \cdot 0.5)^2 + (1 - 2 \cdot 0.5)^2] \\
&= 3^{-1}[0 + 0.25 + 0] \\
&= \frac{1}{3} \cdot \frac{1}{4} \\
&= 0.083
\end{aligned} \tag{6}$$

From this result we can compute $\hat{Var}[\hat{\boldsymbol{\beta}}(\lambda)]$:

$$\begin{aligned}
\hat{Var}[\hat{\boldsymbol{\beta}}(\lambda)] &= \hat{\sigma}^2(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{X}[(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}]^T \\
&= \frac{1}{12} \cdot (9 + 1)^{-1} \cdot 9 \cdot [(9 + 1)^{-1}]^T \\
&= \frac{1}{12} \cdot \frac{1}{10} \cdot 9 \cdot \frac{1}{10} \\
&= 0.0075
\end{aligned} \tag{7}$$

c)

The 95% approximate confidence interval proposed for $\boldsymbol{\beta}$ is unreliable because the ridge $\hat{\boldsymbol{\beta}}(\lambda)$ estimator is biased. In fact:

$$\begin{aligned}\mathbb{E}[\hat{\boldsymbol{\beta}}(\lambda)] &= \mathbb{E}[(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{Y}] \\ &= (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbb{E}[\mathbf{Y}] \\ &= (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} \\ &= \boldsymbol{\beta} - \lambda(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\boldsymbol{\beta} \neq \boldsymbol{\beta}.\end{aligned} \tag{8}$$

Therefore, the values of the $\boldsymbol{\beta}$ parameters could fall within the reported interval with a significance of less than 95%.

d)

Given $X = \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \end{pmatrix}$, $X' = \begin{pmatrix} x_{11} & x_{11} \\ x_{12} & x_{12} \\ x_{13} & x_{13} \end{pmatrix}$ and a generic fixed λ value.

The second column of the matrix X' is a linear combination of the first one, in our case the coefficient of the linear transformation is 1.

So, X' is a singular matrix ($\det(X') = 0$). It's the case of super-collinearity.

In our case, by adding a new column identical to the first one, we have that:

$$\tilde{\mathbf{X}}^T = \begin{pmatrix} 2 & 1 & -2 \\ 2 & 1 & -2 \end{pmatrix},$$

so,

$$\tilde{\mathbf{X}}^T\tilde{\mathbf{X}} = \begin{pmatrix} 9 & 9 \\ 9 & 9 \end{pmatrix}, \qquad \tilde{\mathbf{X}}^T\mathbf{Y} = \begin{pmatrix} -5 \\ -5 \end{pmatrix}$$

and

$$\begin{aligned}
\hat{\boldsymbol{\beta}}(\lambda) &= \begin{pmatrix} 9+\lambda & 9 \\ 9 & 9+\lambda \end{pmatrix}^{-1} \begin{pmatrix} -5 \\ -5 \end{pmatrix} \\
&= \begin{pmatrix} 10 & 9 \\ 9 & 10 \end{pmatrix}^{-1} \begin{pmatrix} -5 \\ -5 \end{pmatrix} \\
&= \begin{pmatrix} \frac{10}{19} & -\frac{9}{19} \\ -\frac{9}{19} & \frac{10}{19} \end{pmatrix} \begin{pmatrix} -5 \\ -5 \end{pmatrix} \\
&= \begin{pmatrix} -\frac{5}{19} \\ -\frac{5}{19} \end{pmatrix} = \begin{pmatrix} -0.263 \\ -0.263 \end{pmatrix}.
\end{aligned} \tag{9}$$

We can now calculate the new estimated error variance:

$$\begin{aligned}
\hat{\sigma}^2 &= \frac{1}{3} \cdot \left\| \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix} - \begin{pmatrix} 2 & 2 \\ 1 & 1 \\ -2 & -2 \end{pmatrix} \begin{pmatrix} -\frac{5}{19} \\ -\frac{5}{19} \end{pmatrix} \right\|_2^2 \\
&= \frac{1}{3} \cdot \left\| \begin{pmatrix} \frac{1}{19} \\ -\frac{9}{19} \\ -\frac{1}{19} \end{pmatrix} \right\|_2^2 \\
&= \frac{1}{3} \cdot [(\frac{1}{19})^2 + (-\frac{9}{19})^2 + (-\frac{1}{19})^2] \\
&= \frac{1}{3} \cdot \frac{83}{361} \\
&= \frac{83}{1083} = 0.077.
\end{aligned} \tag{10}$$

As a further feedback, we apply SVD to $\mathbf{X}$ to demonstrate the consistency of the results reported by ridge. Specifically, SVD involves decomposing a matrix into $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ where the diagonal matrix $\mathbf{D}$, uniquely determined by $\mathbf{X}$, are known as the singular values of $\mathbf{X}$. The matrix $\mathbf{U}$ and the columns of $\mathbf{V}$ are called left-singular vectors and right-singular vectors of $\mathbf{X}$, respectively.
In our case we have:

$$\hat{\boldsymbol{\beta}}^{ridge} = \mathbf{V}(\mathbf{D}^2 + \lambda \mathbf{I}_{pp})^{-1}\mathbf{D}\mathbf{U}^T\mathbf{Y}$$

So, the predictions of our model are:

$$\mathbf{X}\hat{\boldsymbol{\beta}}^{ridge} = \mathbf{U}\mathbf{D}(\mathbf{D}^2 + \lambda \mathbf{I}_{pp})^{-1}\mathbf{D}\mathbf{U}^T\mathbf{Y}$$

```r
x <- matrix(c(2,1,-2))
y <- matrix(c(-1,-1,1))
lambda <- 1

svd_data <- svd(x)
v <- svd_data$v
u <- svd_data$u
d <- svd_data$d
```

```r
beta_svd <- v %*% solve(d %*% t(d) + lambda) %*% d %*% t(u) %*% y
beta_svd
```

```
##      [,1]
## [1,] -0.5
```

```r
x_beta_svd <- u %*% d %*% solve(d %*% t(d) + lambda) %*% d %*% t(u) %*% y
x_beta_svd
```

```
##      [,1]
## [1,] -1.0
## [2,] -0.5
## [3,]  1.0
```

```r
MSE <- mean((y-x_beta_svd)^2)
MSE
```

```
## [1] 0.08333333
```

Augmented data case:

```r
x <- matrix(c(2,1,-2))
x <- cbind(x,x)

y <- matrix(c(-1,-1,1))


svd_data <- svd(x)
v <- svd_data$v
u <- svd_data$u
d <- diag(svd_data$d)
```

```r
beta_svd <- v %*% solve(d %*% t(d) + diag(lambda,2)) %*% d %*% t(u) %*% y
beta_svd
```

```
##             [,1]
## [1,] -0.2631579
## [2,] -0.2631579
```

```
x_beta_svd <- u %*% d %*% solve(d %*% t(d) + diag(lambda,2)) %*% d %*% t(u) %*% y
x_beta_svd
```

```
##             [,1]
## [1,] -1.0526316
## [2,] -0.5263158
## [3,]  1.0526316
```

```
MSE <- mean((y-x_beta_svd)^2)
MSE
```

```
## [1] 0.07663897
```

Using the SVD decomposition method for estimating the β^{SVD}, it is possible to show that the value of the MSE turns out to be lower for augmented data, obtaining results in agreement with those obtained by the ridge regression method.