Assignment 7

STAT 32950

Ki Hyun

Due: 09:00 (CT) 2023-05-16

```
library(dplyr)
library(ggplot2)
library(MASS)
library(glmnet)
```

Problem 1.

```
x1 = rnorm(30)
x2 = x1 + rnorm(30, sd = 0.01)
Y = rnorm(30, mean = 3 + x1 + x2)
```

(a)

```
OLS_model <- lm(Y ~ x1 + x2)
betas <- OLS_model$coefficients
summary(OLS_model)
```

```
##
## Call:
## lm(formula = Y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7345 -0.6456 -0.1179  0.6083  2.0093
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.7448     0.1895  14.482    3e-14 ***
## x1           33.0493    19.9179   1.659    0.109
## x2          -31.1073    19.8991  -1.563    0.130
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9764 on 27 degrees of freedom
## Multiple R-squared:  0.825,  Adjusted R-squared:  0.812
## F-statistic: 63.63 on 2 and 27 DF,  p-value: 6.054e-11
```

From the Least Square method, the fitted model with estimated parameters is as below:

$$\begin{aligned}\mathbf{E}[\hat{Y}] &= \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 \\ &\approx 2.74 + (33.05)x_1 + (-31.11)x_2\end{aligned}$$

(b)

From the given code, the true model with the true β_i's are:

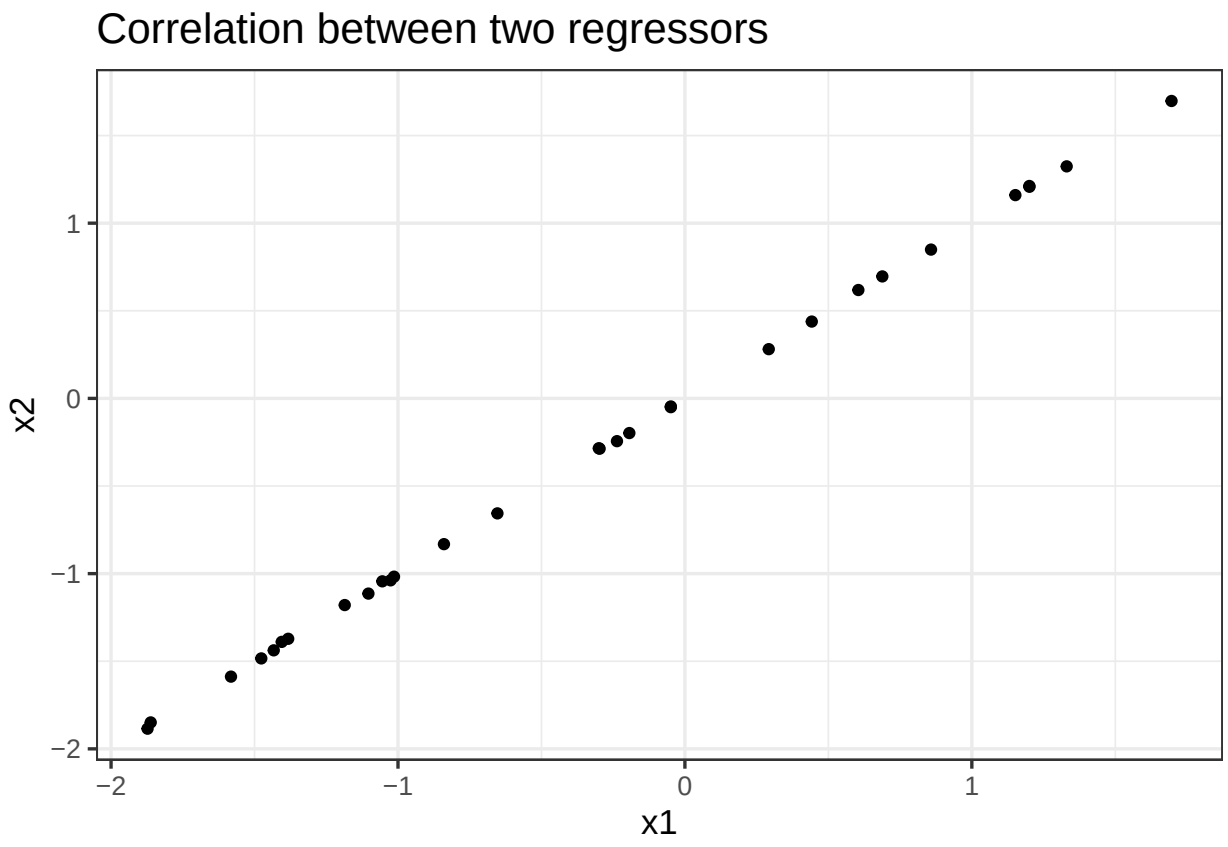
$$Y = 3 + x_1 + x_2 + \epsilon$$

where $\beta_0 = 3$, $\beta_1 = 1$, and $\beta_2 = 1$.

Compared to this true value, the LS model in (a) is **very bad**.

From the code, it appears that x_1 and x_2 are highly correlated. If we actually plot the two values:

```
ggplot(data = tibble(x1 = x1, x2 = x2)) +
  geom_point(mapping = aes(x = x1, y = x2)) +
  labs(title = "Correlation between two regressors") +
  theme_bw(base_size = 13)
```



We can clearly see that the two independent variables are highly correlated. This would result in coefficient estimates that are far away from the true values.

(c)

```
RSS_true = sum((Y - 3 - x1 - x2)^2)
print(paste0("The RSS of the true model: ", RSS_true))
```

```
## [1] "The RSS of the true model: 30.8654998613122"
```

```
RSS_LS = sum(OLS_model$residuals^2)
print(paste0("The RSS of the LS model: ", RSS_LS))
```

```
## [1] "The RSS of the LS model: 25.7407762877548"
```

The two RSS are indeed comparable. In fact, the RSS of the "bad" LS model is lower than the true model.

This is the case since the LS parameter values are chosen to minimize the RSS value. Therefore, the optimized LS coefficients will result in not only comparable, but also the lowest in-sample RSS value.

(d)

```
Ridge_model <- lm.ridge(Y ~ x1 + x2, lambda = 1, model = TRUE)
betas_ridge <- coef(Ridge_model)
summary(Ridge_model)
```

```
##        Length Class  Mode
## coef   2      -none- numeric
## scales 2      -none- numeric
## Inter  1      -none- numeric
## lambda 1      -none- numeric
## ym     1      -none- numeric
## xm     2      -none- numeric
## GCV    1      -none- numeric
## kHKB   1      -none- numeric
## kLW    1      -none- numeric
```

The fitted Ridge model is:

$$\begin{aligned}\mathbf{E}[\hat{Y}] &= \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 \\ &\approx 2.68 + (0.98)x_1 + (0.9)x_2\end{aligned}$$

The parameter estimates are much closer to the true model.

(e)

The criterion of the LS method is the RSS. In mathematical expression:

$$\min_{\boldsymbol{\beta}} \sum_{j=1}^{n} \left[y_j - (\hat{\beta}_0 + \hat{\beta}_1 x_{1,j} + \hat{\beta}_2 x_{2,j}) \right]^2$$

The criterion of the Ridge method is the RSS and a l_2 penalty term on the coefficients. In mathematical expression:

$$\min_{\boldsymbol{\beta}} \left(\sum_{j=1}^{n} \left[y_j - (\hat{\beta}_0 + \hat{\beta}_1 x_{1,j} + \hat{\beta}_2 x_{2,j}) \right]^2 + \sum_{k=1}^{3} |\beta_k|^2 \right)$$

To recap, the result in (a) was:

```
OLS_model$coefficients
```

```
## (Intercept)          x1          x2
##    2.744768   33.049256  -31.107286
```

The result in (d) was:

```
Ridge_model
```

```
##                   x1        x2
## 2.6794508 0.9761712 0.9041853
```

As shown by comparing the absolute values of the coefficients for the results of (a) and (d), the Ridge regression reduces the magnitude of the coefficients.

Problem 2.

```
data(Boston)
colnames(Boston)
```

```
##  [1] "crim"    "zn"      "indus"   "chas"    "nox"     "rm"      "age"
##  [8] "dis"     "rad"     "tax"     "ptratio" "black"   "lstat"   "medv"
```

(a)

```
Tdata = Boston[1:300,]
Cdata = Boston[301:506,]
X=as.matrix(Tdata[,1:13])
Y=Tdata[,14]
```

```
trainfit = glmnet(X, Y)
nx = as.matrix(Cdata[, 1:13])
ny = Cdata[, 14]
calibrate_mse = colMeans((predict(trainfit, newx = nx) - ny)^2)
lambda_star <- trainfit$lambda[which.min(calibrate_mse)]
```

```
betas_LASSO <- coef(trainfit, s = lambda_star)
betas_LASSO
```

```
## 14 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept) -19.190331147
## crim          .
## zn            .
## indus         .
## chas          0.131996136
## nox           .
## rm            9.193505503
## age          -0.024883019
## dis          -0.358735094
## rad           .
## tax          -0.009113364
## ptratio      -0.586462537
## black         0.008388966
## lstat        -0.119856133
```

The optimal model after calibration is:

$$\begin{aligned}\mathbf{E}[Y_{MEDV}] \approx &-19.19+ \\ &(0.13)X_{chas} + (9.19)X_{rm} + (-0.02)X_{age} + (-0.36)X_{dis}+ \\ &(-0.01)X_{tax} + (-0.59)X_{ptratio} + (0.01)X_{black} + (-0.12)X_{lstat}\end{aligned}$$

The independent variables crim, zn, indus, nox, rad were excluded from the model as their coefficients were optimized at 0 after the l_1 penalty.

(b)

```
OLS_model2 <- lm(medv ~ ., data = Boston)
betas2 <- OLS_model2$coefficients
summary(OLS_model2)
```

```
##
## Call:
## lm(formula = medv ~ ., data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.595  -2.730  -0.518   1.777  26.199
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
## crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
## zn           4.642e-02  1.373e-02   3.382 0.000778 ***
## indus        2.056e-02  6.150e-02   0.334 0.738288
## chas         2.687e+00  8.616e-01   3.118 0.001925 **
## nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
## rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
## age          6.922e-04  1.321e-02   0.052 0.958229
## dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
## rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
## tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
## ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
## black        9.312e-03  2.686e-03   3.467 0.000573 ***
## lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.745 on 492 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7338
## F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

The OLS model result is:

$$\begin{aligned}\mathbf{E}[Y_{MEDV}] \approx & 36.46+ \\ & (-0.11)X_{crim} + (0.046)X_{zn} + (0.021)X_{indus} + (2.69)X_{chas}+ \\ & (-17.77)X_{nox} + (3.81)X_{rm} + (6.9 \times 10^{-4})X_{age} + (-1.48)X_{dis}+ \\ & (0.31)X_{rad} + (-0.012)X_{tax} + (-0.95)X_{ptratio} + (0.009)X_{black}+ \\ & (-0.52)X_{lstat}\end{aligned}$$