Assignment 6

Brian Morales

October 11, 2022

```
library(ISLR)
library(MASS)
head(College)
```

```
##                              Private Apps Accept Enroll Top10perc Top25perc
## Abilene Christian University     Yes 1660   1232    721        23        52
## Adelphi University               Yes 2186   1924    512        16        29
## Adrian College                   Yes 1428   1097    336        22        50
## Agnes Scott College              Yes  417    349    137        60        89
## Alaska Pacific University        Yes  193    146     55        16        44
## Albertson College                Yes  587    479    158        38        62
##                              F.Undergrad P.Undergrad Outstate Room.Board Books
## Abilene Christian University        2885         537     7440       3300   450
## Adelphi University                  2683        1227    12280       6450   750
## Adrian College                      1036          99    11250       3750   400
## Agnes Scott College                  510          63    12960       5450   450
## Alaska Pacific University            249         869     7560       4120   800
## Albertson College                    678          41    13500       3335   500
##                              Personal PhD Terminal S.F.Ratio perc.alumni Expend
## Abilene Christian University     2200  70       78      18.1          12   7041
## Adelphi University               1500  29       30      12.2          16  10527
## Adrian College                   1165  53       66      12.9          30   8735
## Agnes Scott College               875  92       97       7.7          37  19016
## Alaska Pacific University        1500  76       72      11.9           2  10922
## Albertson College                 675  67       73       9.4          11   9727
##                              Grad.Rate
## Abilene Christian University        60
## Adelphi University                  56
## Adrian College                      54
## Agnes Scott College                 59
## Alaska Pacific University           15
## Albertson College                   55
```

Part A

A usual train and test split is %80 train and %20 test. Therefore I follow that convention below.

```
set.seed(001)
split = sort(sample(nrow(College), nrow(College)*0.8))
```

```
train = College[split, ]
test = College[-split, ]
```

Part B: Linear Model

```
lm.fit = lm(Apps ~., data=train)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Apps ~ ., data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5555.2  -404.6    19.9   310.3  7577.7
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -630.58238  435.56266  -1.448 0.148209
## PrivateYes  -388.97393  148.87623  -2.613 0.009206 **
## Accept         1.69123    0.04433  38.153  < 2e-16 ***
## Enroll        -1.21543    0.20873  -5.823 9.41e-09 ***
## Top10perc     50.45622    5.88174   8.578  < 2e-16 ***
## Top25perc    -13.62655    4.67321  -2.916 0.003679 **
## F.Undergrad    0.08271    0.03632   2.277 0.023111 *
## P.Undergrad    0.06555    0.03367   1.947 0.052008 .
## Outstate      -0.07562    0.01987  -3.805 0.000156 ***
## Room.Board     0.14161    0.05130   2.760 0.005947 **
## Books          0.21161    0.25184   0.840 0.401102
## Personal       0.01873    0.06604   0.284 0.776803
## PhD           -9.72551    4.91228  -1.980 0.048176 *
## Terminal      -0.48690    5.43302  -0.090 0.928620
## S.F.Ratio     18.26146   13.83984   1.319 0.187508
## perc.alumni    1.39008    4.39572   0.316 0.751934
## Expend         0.05764    0.01254   4.595 5.26e-06 ***
## Grad.Rate      5.89480    3.11185   1.894 0.058662 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 993.8 on 603 degrees of freedom
## Multiple R-squared:  0.9347, Adjusted R-squared:  0.9328
## F-statistic: 507.5 on 17 and 603 DF,  p-value: < 2.2e-16
```

```
lm.pred = predict(lm.fit, newdata = test)
mse = mean((test$Apps - lm.pred)^2)
print(mse)
```

```
## [1] 1567324
```

Our base case MSE is $1,567,324$. A pretty large MSE.

Part C: Ridge Regression

We choose our best lambda using cross validation

```
library(glmnet)
```

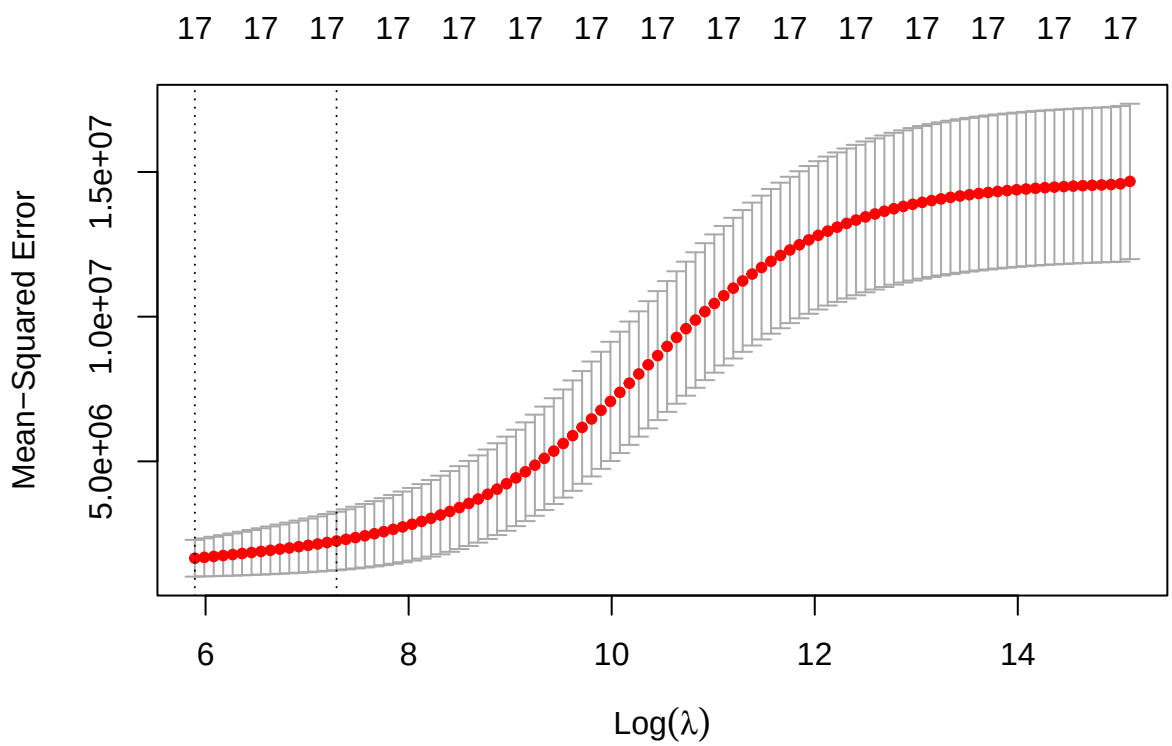
```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-4
```

```
x_train = model.matrix(Apps ~., train)[,-1]
y_train = train$Apps

x_test = model.matrix(Apps ~., test)[,-1]
y_test = test$Apps

# cross validation selection here
set.seed(0101)
cross_valid <- cv.glmnet(x_train, y_train, alpha = 0)
plot(cross_valid)
```




```
bestlam <- cross_valid$lambda.min
ridge.fit = glmnet(x_train, y_train, alpha = 0, lambda = bestlam)
```

```
ridge.pred = predict(ridge.fit, s = bestlam, newx = x_test)
mse = mean((ridge.pred - y_test)^2)
print(mse)
```

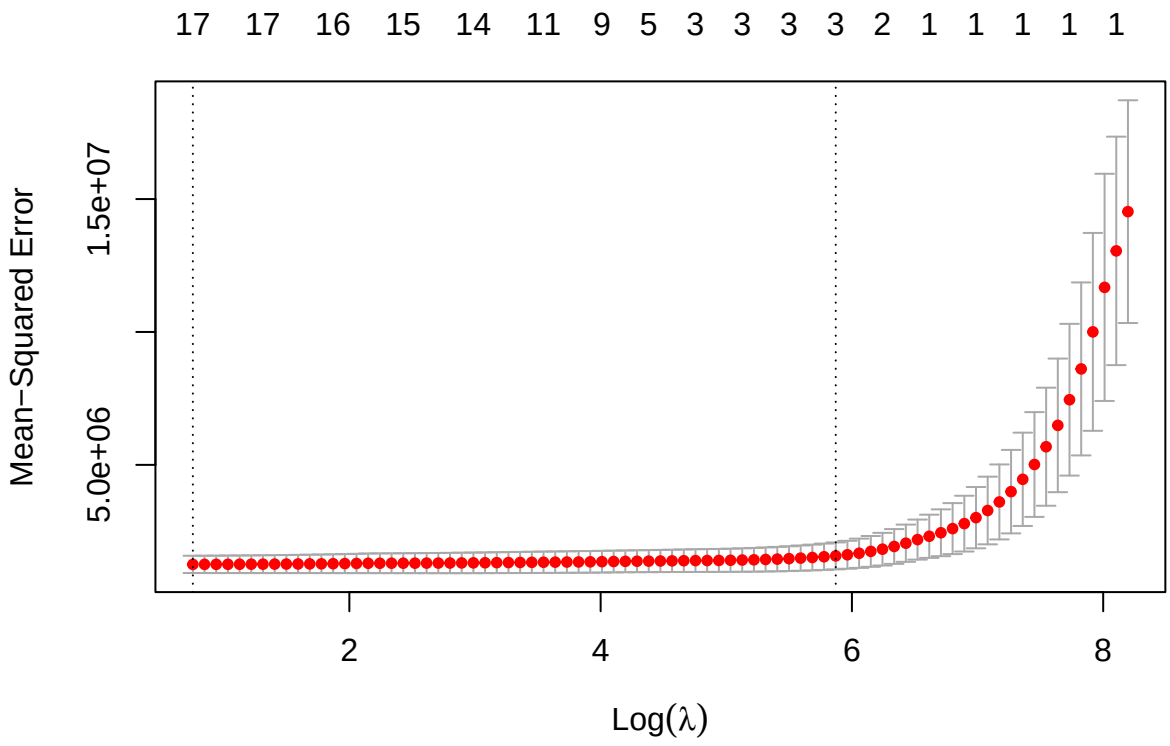
```
## [1] 1442487
```

The MSE for ridge regression is $1,442,487$ which is significantly lower than linear regressions MSE.

Part D: Lasso Regression

Similarly, we choose best lambda using cross validation

```
set.seed(010)
cross_valid <- cv.glmnet(x_train, y_train, alpha = 1)
plot(cross_valid)
```




```
bestlam <- cross_valid$lambda.min
lasso.fit = glmnet(x_train, y_train, alpha = 1, lambda = bestlam)
lasso.pred = predict(lasso.fit, s = bestlam, newx = x_test)
mse = mean((lasso.pred - y_test)^2)
print(mse)
```

```
## [1] 1553527
```

Here we have our $MSE = 1,553,527$ which is greater than ridge regression and linear regression. I'm kind of surprised because I thought lasso would do the best.

```
variables = length(lasso.fit$beta)
lasso_coef = predict(lasso.fit, type="coefficients", s=bestlam)[1:variables,]
lasso_coef[lasso_coef != 0]
```

```
##   (Intercept)    PrivateYes        Accept        Enroll     Top10perc
## -637.25517217 -383.69774027    1.67464667   -1.09263805   48.97995040
##     Top25perc   F.Undergrad   P.Undergrad      Outstate    Room.Board
##  -12.50269893    0.06750937    0.06452051   -0.07273262    0.13921176
##         Books      Personal           PhD      Terminal     S.F.Ratio
##    0.19910184    0.01572526   -9.50800866   -0.33786629   16.97883017
##   perc.alumni        Expend
##    0.58183816    0.05690592
```

An advantage of the lasso regression is that resulting coefficients estimates can be sparse. In our fit we choose the best λ from cross-validation. In our case use all 17 coefficients but in some cases the resulting coefficients can be half or less of all the coefficients.