Homework 6

Chapter 6, Exercise 9

In this exercise, we will predict the number of applications received using the other variables in the College data set.

```r
library(ISLR)
set.seed(123)

# recommended (but optional):
# check if there are any missing observations:
print(sum(is.na(College)))
```

(a) Split the data set into a training set and a test set.a

```
## [1] 0
```

```r
n = dim(College)[1]
train.size = n / 2
train = sample(1:n, train.size)
test = -train
College.train = College[train, ]
College.test = College[test, ]
```

```r
lm.fit = lm(Apps~., data=College.train)
lm.pred = predict(lm.fit, College.test)
mean((College.test[, "Apps"] - lm.pred)^2)
```

b Fit a linear model using least squares on the training set, and report the test error obtained.

```
## [1] 1373995
```

Test MSE is 939038.3

c Fit a ridge regression model on the training set, with λ chosen by cross-validation. Report the test error obtained. (Pick λ using College.train and report error on College.test)

```r
library(glmnet)
```

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

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

```r
train.mat = model.matrix(Apps~., data=College.train)
test.mat = model.matrix(Apps~., data=College.test)
grid = 10 ^ seq(4, -2, length=100)
mod.ridge = cv.glmnet(train.mat, College.train[, "Apps"], alpha=0, lambda=grid, thresh=1e-12)
lambda.best = mod.ridge$lambda.min
lambda.best
```

```
## [1] 18.73817
```

```r
ridge.pred = predict(mod.ridge, newx=test.mat, s=lambda.best)
mean((College.test[, "Apps"] - ridge.pred)^2)
```

```
## [1] 1431537
```

Test MSE is 939008.4.

d Fit a lasso model on the training set, with λ chosen by cross-validation. Report the test error obtained, along with the number of non-zero coefficient estimates. (Pick λ using College.train and report error on College.test)

```
mod.lasso = cv.glmnet(train.mat, College.train[, "Apps"], alpha=1, lambda=grid, thresh=1e-12)

lambda.best = mod.lasso$lambda.min
lambda.best
```

```
## [1] 21.54435
```

```
lasso.pred = predict(mod.lasso, newx=test.mat, s=lambda.best)
mean((College.test[, "Apps"] - lasso.pred)^2)
```

```
## [1] 1397303
```

Test MSE 918893.7.

The coefficients look like

```
mod.lasso = glmnet(model.matrix(Apps~., data=College), College[, "Apps"], alpha=1)
lasso.coefs = predict(mod.lasso, s=lambda.best, type="coefficients")

print(lasso.coefs)
```

```
## 19 x 1 sparse Matrix of class "dgCMatrix"
##                        s1
## (Intercept) -6.038452e+02
## (Intercept)  .           
## PrivateYes  -4.235413e+02
## Accept       1.455236e+00
## Enroll      -2.003696e-01
## Top10perc    3.367640e+01
## Top25perc   -2.403036e+00
## F.Undergrad  .           
## P.Undergrad  2.086035e-02
## Outstate    -5.781855e-02
## Room.Board   1.246462e-01
## Books        .           
## Personal     1.832912e-05
## PhD         -5.601313e+00
## Terminal    -3.313824e+00
## S.F.Ratio    4.478684e+00
## perc.alumni -9.796600e-01
## Expend       6.967693e-02
## Grad.Rate    5.159652e+00
```

```
sum(lasso.coefs != 0)
```

```
## [1] 16
```

There are 16 non-zero lasso coefficients.