Project 3

Achraf cherkaoui

We will use the `College` data set in the package `ISLR`. It will be a objective-oriented project. The questions are modified based on # 9 on page 263.

The college data set contains information of 777 US college and university with 18 variables. We will use `Apps` as our response y. And we would like to consider both prediction and inference.

Please plug in your own R code chunks.

Linar model selection and regularization

9 (pg 263) We will predict the number of applications received (y) using the other variables in the `College` data set.

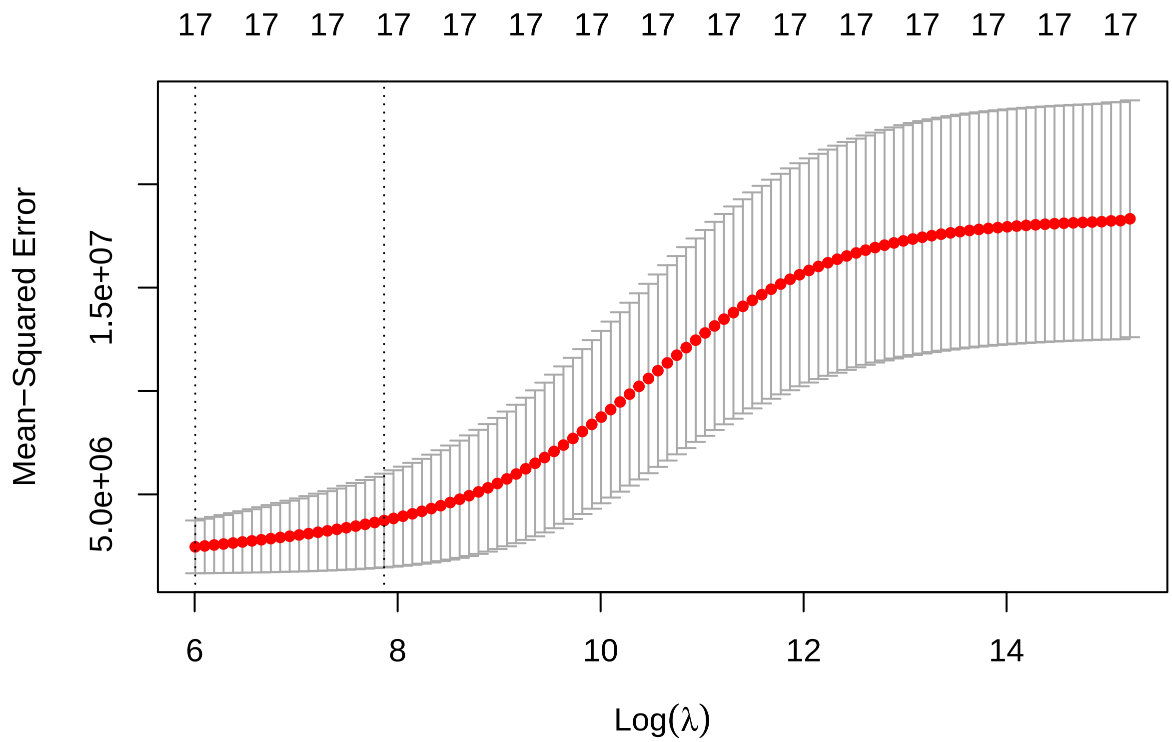
(a) Split the data into a training set and a test set.

```
library(ISLR)
library(glmnet)
set.seed(1)
train = sample(length(College$Apps), length(College$Apps)/2)
test = - train
#train = College[train.rows, ]
#test = College[test.rows, ]
X <- model.matrix(Apps~.,College)[,-1]
Y <- College$Apps
y.test<- Y[test]
```

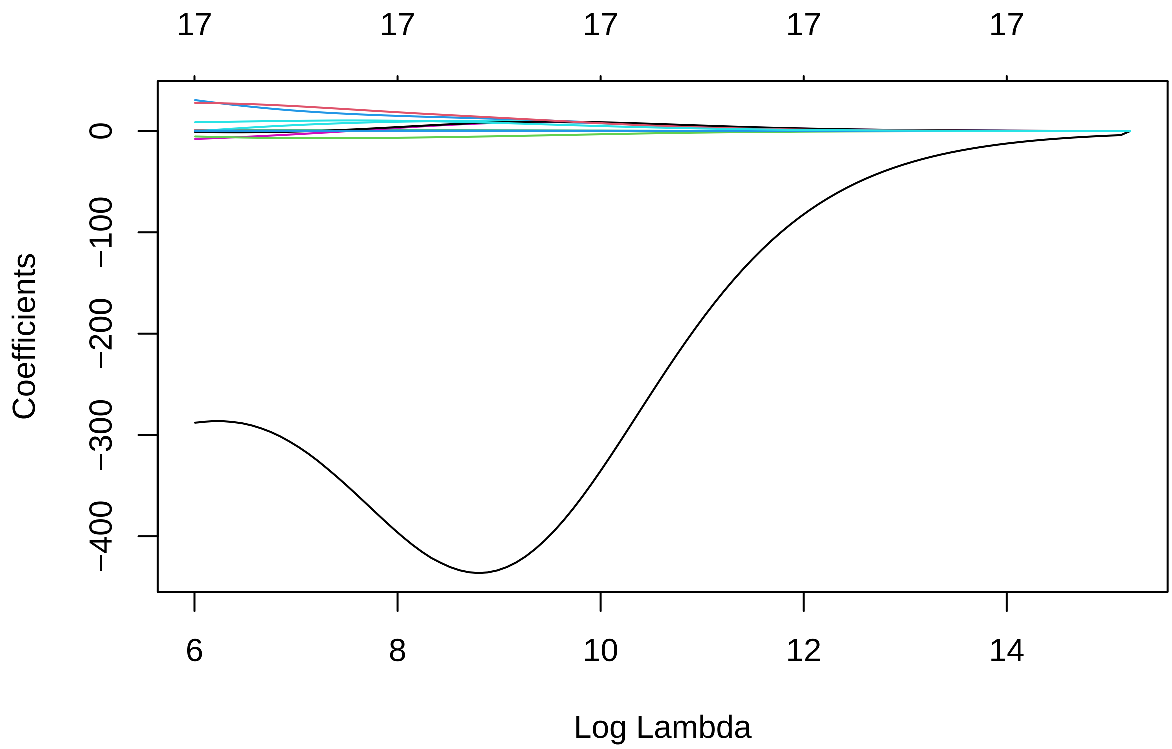
Question (b). Ridge regression (hint: refer to the lab on lecture notes from page 22 to page 25)

- Use `cv.glmnet()` to fit ridge regression models on the training set.
- Make the plot for λ (or $\log(\lambda)$) versus Mean-squared Error. Also, make the plot for λ (or $\log(\lambda)$) (x-axis) versus coefficient estimates (y-axis). Also
- Find the best λ based on the cross-validation performed by `cv.glmnet()`, and show the value of the best λ.
- Make prediction on the test set using the ridge regression with the best λ, report the test error.

```
set.seed(1)
cv.out <- cv.glmnet(X[train,],Y[train],alpha=0)
plot(cv.out)
```




```
plot(cv.out$glmnet.fit, xvar="lambda")
```




```
bestlambda <- cv.out$lambda.min
bestlambda
```

```
## [1] 405.8404
```

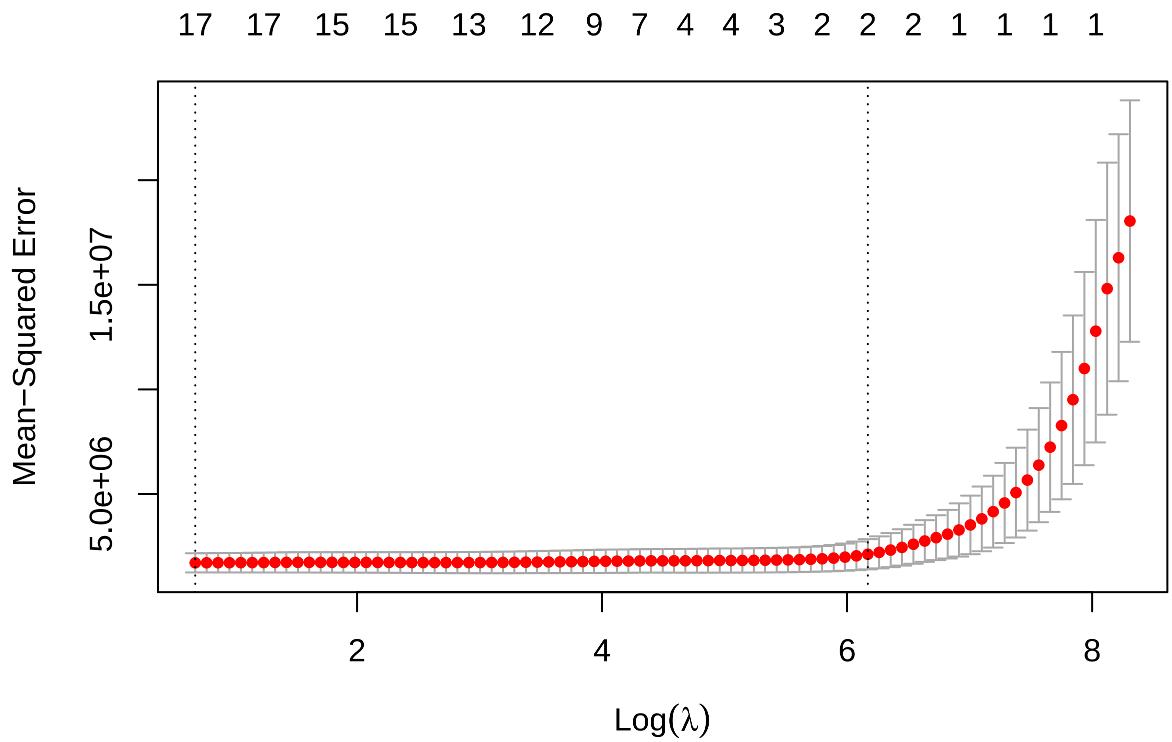
```
ridge.pred <- predict(cv.out,s=bestlambda,newx=X[test,])
mean((ridge.pred-y.test)^2)
```

```
## [1] 976261.5
```

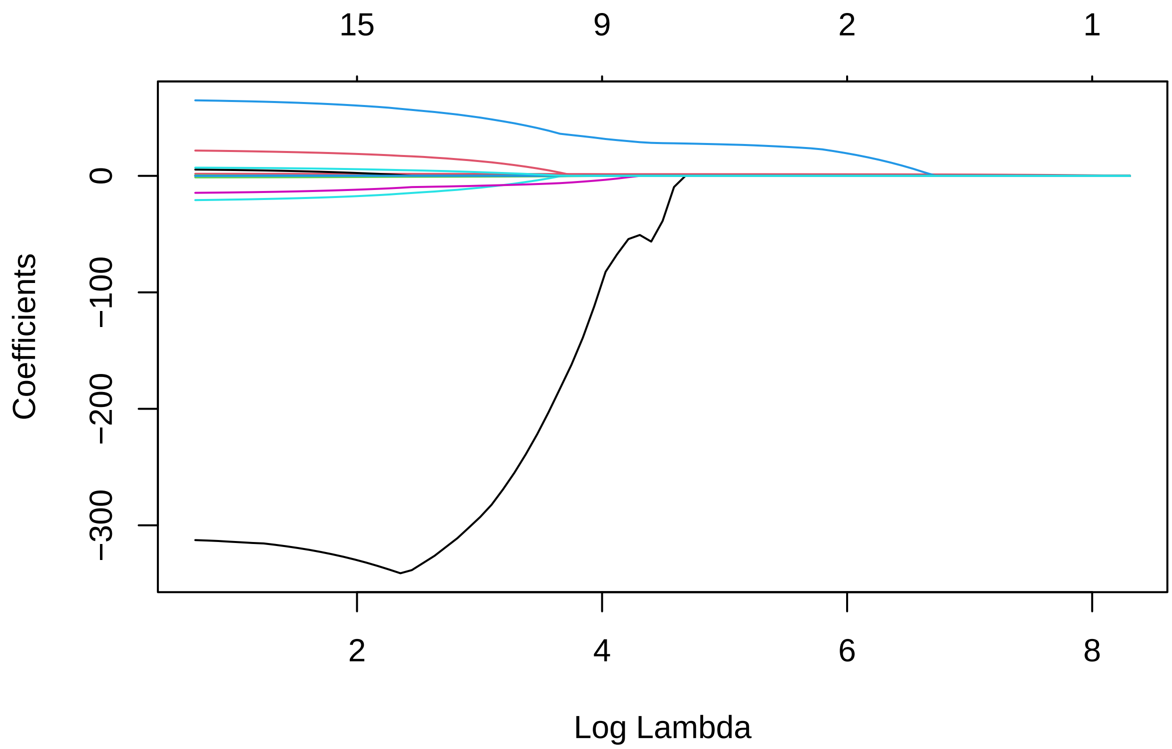
Question (c). Lasso regression (hint: refer to the lab on lecture notes from page 22 to page 25)

- Use `cv.glmnet()` to fit Lasso regression models on the training set.
- Make the plot for λ (or $\log(\lambda)$) versus Mean-squared Error. Also, make the plot for λ (or $\log(\lambda)$) (x-axis) versus coefficient estimates (y-axis).
- Find the best λ based on the cross-validation performed by `cv.glmnet()`, and show the value of the best λ.
- Make prediction on the test set using the lasso regression with the best λ, report the test error.
- Re-fit the Lasso using the best selected λ for the whole data and show the predicted coefficient estimates.
- From Lasso regression, what are the most important predictors (whose coefficients are NOT forced to be 0) for Apps for US colleges and universities? Do they positively or negatively explain the response?

```
set.seed(1)
cv.outL <- cv.glmnet(X[train,],Y[train],alpha=1)
plot(cv.outL)
```




```
plot(cv.outL$glmnet.fit, xvar="lambda")
```




```
bestlambda <- cv.outL$lambda.min
bestlambda
```

```
## [1] 1.97344
```

```
lasso.pred <- predict(cv.outL,s=bestlambda,newx=X[test,])
mean((lasso.pred-y.test)^2)
```

```
## [1] 1115901
```

```
out <- glmnet(X,Y,alpha=1)
predict(out,type="coefficients",s= bestlambda)
```

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept) -471.39372069
## PrivateYes  -491.04485135
## Accept         1.57033288
## Enroll        -0.75961467
## Top10perc     48.14698891
## Top25perc    -12.84690694
## F.Undergrad    0.04149116
## P.Undergrad    0.04438973
## Outstate      -0.08328388
## Room.Board     0.14943472
## Books          0.01532293
## Personal       0.02909954
## PhD           -8.39597537
```

```
## Terminal      -3.26800340
## S.F.Ratio     14.59298267
## perc.alumni   -0.04404771
## Expend         0.07712632
## Grad.Rate      8.28950241
```

answer:The most important predictors are PrivateYes,accept, Top10perc ,Top25perc , PHD,Terminal , S.F.Ratio , Grad.Rate. these predictors negatively explain the response because most of coefficients are negative and with a big negative value.

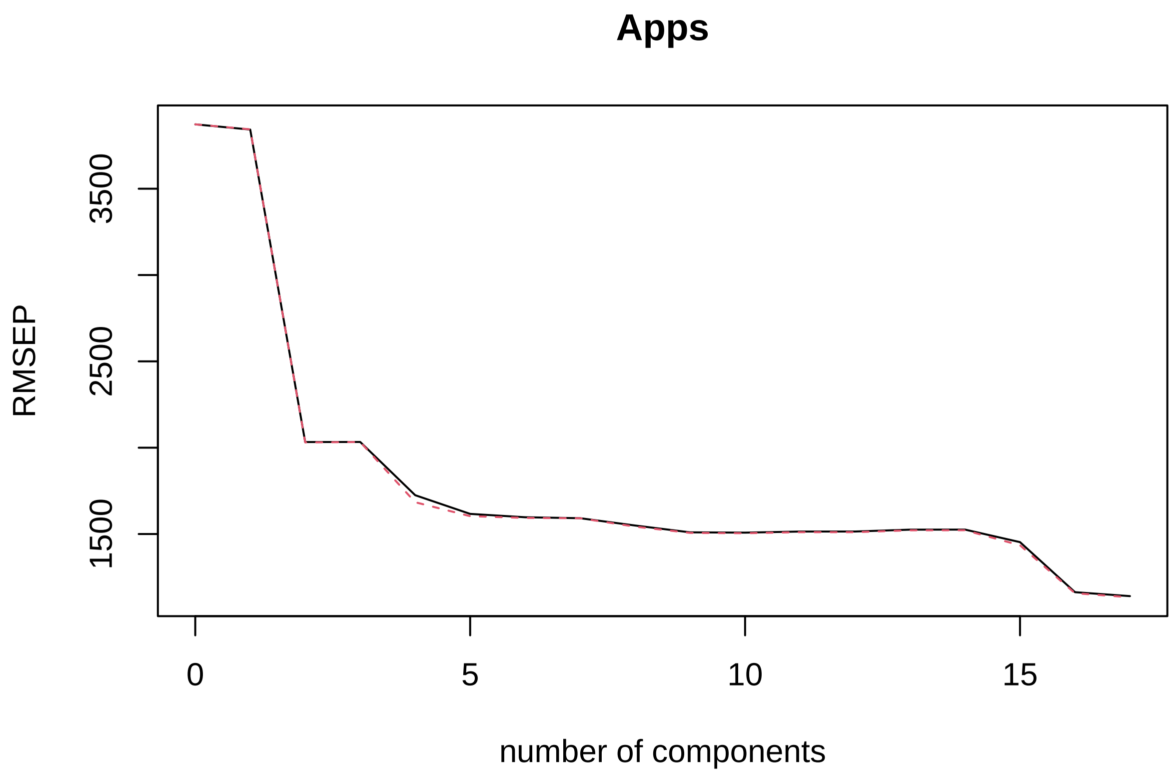
Question (d). PCR model. (hint: refer to the lab on lecture notes from page 29 to page 33)

- Use `pcr()` in the `pls()` package to fit PCR models on the training set using cross-validation. Show the summary.
- Find the best M chosen by cross-validation.
- Show the plot for the number of components versus validation MSE.
- Make prediction on the test set using the best M, and report the test error obtained.

```
library(pls)
set.seed(1)
pcr.fit=pcr(Apps ~., data= College,scale=TRUE,validation="CV")
summary(pcr.fit)
```

```
## Data:    X dimension: 777 17
##  Y dimension: 777 1
## Fit method: svdpc
## Number of components considered: 17
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV            3873     3842     2033     2033     1725     1617     1597
## adjCV         3873     3844     2031     2033     1684     1604     1593
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV        1592     1549     1510      1508      1514      1515      1525
## adjCV     1592     1543     1507      1505      1511      1511      1522
##        14 comps  15 comps  16 comps  17 comps
## CV         1526      1453      1163      1140
## adjCV      1522      1435      1157      1134
##
## TRAINING: % variance explained
##       1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X      31.670    57.30    64.30    69.90    75.39    80.38    83.99    87.40
## Apps    2.316    73.06    73.07    82.08    84.08    84.11    84.32    85.18
##       9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
## X       90.50     92.91     95.01     96.81      97.9     98.75     99.36
## Apps    85.88     86.06     86.06     86.10      86.1     86.13     90.32
##       16 comps  17 comps
## X        99.84    100.00
## Apps     92.52     92.92
```

```
validationplot(pcr.fit,val.type="RMSEP")
```




```
pcr.pred=predict(pcr.fit,X[test,],ncomp=17)# predict on test data
mean((pcr.pred-y.test)^2) # calculate the test MSE
```

```
## [1] 928953.7
```

Answer : The best M is 17.

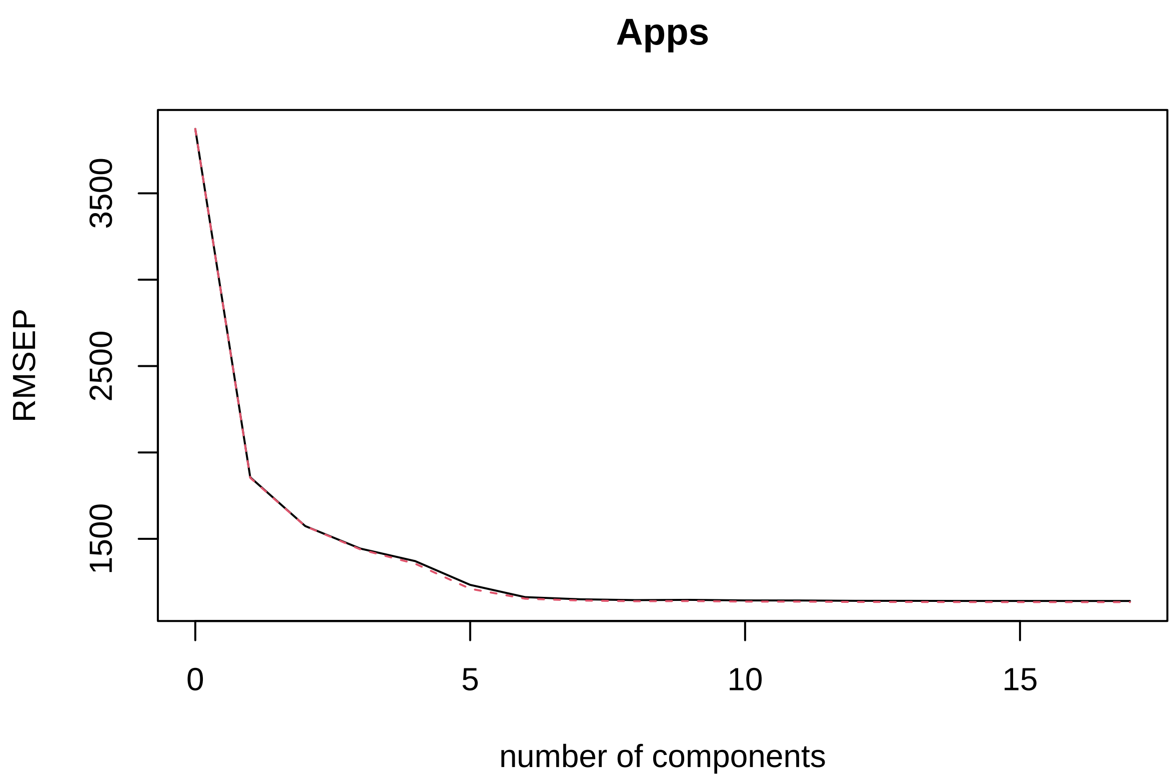
Question (e). (**Bonus**) Repeat steps in (d) for the PLS method. (hint: refer to the lab on lecture notes from page 29 to page 33)

```
set.seed(1)
pls.fit=plsr(Apps~., data=College,scale=TRUE,validation="CV")
summary(pls.fit)
```

```
## Data:    X dimension: 777 17
##  Y dimension: 777 1
## Fit method: kernelpls
## Number of components considered: 17
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV            3873     1857     1574     1444     1371     1234     1163
## adjCV         3873     1853     1574     1440     1356     1211     1154
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV        1150     1145     1147      1144      1143      1141      1141
## adjCV     1143     1139     1140      1137      1137      1135      1135
##        14 comps  15 comps  16 comps  17 comps
```

```
## CV          1141       1140       1140       1140
## adjCV       1134       1134       1134       1134
##
## TRAINING: % variance explained
##         1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X         25.76    40.33    62.59    64.97    66.87    71.33    75.39    79.37
## Apps      78.01    85.14    87.67    90.73    92.63    92.72    92.77    92.82
##         9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
## X         82.36     85.04     87.92     90.65     92.69     95.50     96.87
## Apps      92.87     92.89     92.90     92.91     92.92     92.92     92.92
##         16 comps  17 comps
## X          98.65    100.00
## Apps       92.92     92.92
```

```
validationplot(pls.fit,val.type="RMSEP")
```




```
pls.pred=predict(pls.fit,X[test,],ncomp = 6)# predict on test data. Best M=6
mean((pls.pred-y.test)^2) # calculate the test MSE
```

```
## [1] 929380.1
```

Answer : The best M is 6.

Question (f). Comment on the results obtained: is there much difference among the test errors resulting from these three (or four if you work on (e)) approaches? Or which is the best method for the data?

Answer: Lasso regression has the biggest test error .Yet, there is no big difference between the test errors obtained from Ridge regression , PLS, PCR .since the principal component analysis regression (PCAR) tent to have the smallest test error therefore it is going to be the best model for the data.