

Q2.

```
library(ISLR)
data("College")
College$Accept.Apps<-College$Accept/College$Apps
College<-College[,c(-2,-3)]
#Split the dataset
set.seed(234)
inTrain <- createDataPartition(College$Accept.Apps, p = 0.7, list = FALSE)
training <- College[inTrain,]
testing <- College[-inTrain,]
```

(a)

```
X <- model.matrix(Accept.Apps ~ ., data = College)[-1]
# Run PCA
collegePCA <- prcomp(x = X, center = T, scale = T)
```

We Choose to standardize our data, Since the scales of different variables vary a lot.

```
collegePCA

## Standard deviations (1, ..., p=16):
## [1] 2.3202063 1.9099163 1.0903953 0.9665203 0.9209106 0.9014974 0.7834930
## [8] 0.7596985 0.7038782 0.6333919 0.5883741 0.5535871 0.4293728 0.3795948
## [15] 0.2925402 0.1750902
##
## Rotation (n x k) = (16 x 16):
##
##      PC1      PC2      PC3      PC4      PC5
## PrivateYes -0.20681302  0.356455041 -0.1591399019  0.03669239 -0.03923585
## Enroll      0.03464939 -0.462454913  0.0246197785 -0.07571059 -0.06136725
## Top10perc   -0.34312918 -0.157492803  0.0002373487 -0.38473317 -0.05895401
## Top25perc   -0.31666718 -0.190913384  0.0644941854 -0.40669327  0.02206220
## F.Undergrad 0.06037906 -0.473283545  0.0155847650 -0.05457372 -0.06157156
## P.Undergrad 0.12071330 -0.332095123 -0.1454697842  0.32845546 -0.20131954
## Outstate    -0.37484355  0.073682797 -0.0709249583  0.19779008 -0.02606275
## Room.Board  -0.28237524 -0.002834249 -0.1876472860  0.51936358  0.17450305
## Books       -0.03323706 -0.098129260 -0.6600537366 -0.16214229  0.65009874
## Personal     0.13546497 -0.203608582 -0.4614837522 -0.18011244 -0.33460409
## PhD         -0.24369433 -0.300157933  0.2191702188  0.21858223  0.09364174
## Terminal    -0.24860292 -0.287163985  0.1569234220  0.27737483  0.16018974
## S.F.Ratio    0.26977520 -0.134839801  0.2949613466 -0.05055637  0.46993927
## perc.alumni -0.29132731  0.102080574  0.1587564355 -0.21746668 -0.05517069
## Expend      -0.33637777 -0.072088937 -0.2167806971  0.05856710 -0.28415926
## Grad.Rate    -0.29682521  0.025669164  0.1666259249 -0.15184996  0.20403454
##
##      PC6      PC7      PC8      PC9      PC10
## PrivateYes  0.12977308 -0.16685240  0.046025151 -0.210730381  0.11463677
## Enroll      0.37968193  0.04806950  0.009264097  0.270836882  0.18228282
## Top10perc   -0.03828945  0.14658931 -0.207437186 -0.308024087  0.02285775
## Top25perc   -0.05309190  0.08900245 -0.193962229 -0.403774469 -0.03674098
## F.Undergrad 0.34030917  0.04446552  0.017143427  0.199925132  0.15102496
## P.Undergrad 0.18895873 -0.02573806  0.404669987 -0.605723298 -0.34496102
## Outstate    0.12389422 -0.08847727  0.001696581 -0.002664277  0.19633570
## Room.Board  0.25514858 -0.15582031 -0.312821881 -0.109745295  0.25526994
## Books       -0.05613622  0.16618403  0.234332295  0.080892272 -0.07551044
```

```

## Personal    -0.30706177 -0.67260614 -0.131755103  0.047793743  0.11121565
## PhD         -0.42798888 -0.07499474  0.027832793  0.073616941 -0.09598987
## Terminal    -0.42734017 -0.04113989  0.105461368  0.125249977 -0.05160145
## S.F.Ratio    0.02766480 -0.27618136 -0.111151688 -0.329572850  0.42582930
## perc.alumni  0.03380466 -0.19571483  0.741129392  0.042582398  0.36289505
## Expend      -0.00163687  0.27121084 -0.075119992  0.143535476  0.13524297
## Grad.Rate    0.37095330 -0.47981591 -0.061594994  0.210800288 -0.58735027
##              PC11      PC12      PC13      PC14      PC15
## PrivateYes  -0.692632136  0.25873200 -0.38698443 -0.023188042  0.014246848
## Enroll      -0.249028057  0.00355242 -0.03198605  0.028758893 -0.020862258
## Top10perc    0.046221362 -0.01736028 -0.04481966  0.104186948 -0.727257149
## Top25perc   -0.088902323 -0.23347958  0.08855632 -0.145456913  0.622734534
## F.Undergrad -0.211799028 -0.02657383 -0.03857620 -0.012894604  0.021249495
## P.Undergrad  0.079513281  0.10106581  0.01339449 -0.005876036 -0.027600024
## Outstate    -0.100271062  0.26705152  0.81437712  0.045249482 -0.014967200
## Room.Board  0.214915208 -0.48079461 -0.20189361  0.045088043 -0.012466482
## Books       -0.008701055  0.03588007  0.02435974  0.068552934  0.009707638
## Personal    0.041863946 -0.02325922  0.03523255 -0.021561698  0.001521452
## PhD         -0.145412245  0.08781640 -0.11929824  0.696090060  0.118155335
## Terminal    -0.173066481 -0.01072878 -0.04630350 -0.676592742 -0.156069951
## S.F.Ratio    0.182527711  0.42380583 -0.06167845 -0.047221417  0.016532343
## perc.alumni  0.212784181 -0.22561467 -0.09581960  0.038676644  0.006200652
## Expend      0.427127703  0.54926490 -0.31140597 -0.094508901  0.205746389
## Grad.Rate    0.163783749  0.15155896 -0.07456633 -0.046905892 -0.003064506
##              PC16
## PrivateYes   0.0207312563
## Enroll       -0.6763011086
## Top10perc    0.0193041993
## Top25perc   -0.0356159248
## F.Undergrad  0.7326758367
## P.Undergrad -0.0450587928
## Outstate     0.0253870978
## Room.Board  -0.0104115884
## Books        0.0009363273
## Personal    -0.0111575157
## PhD          0.0117603412
## Terminal    -0.0194870584
## S.F.Ratio   -0.0076363267
## perc.alumni  0.0069337649
## Expend      0.0046295077
## Grad.Rate    0.0133250183

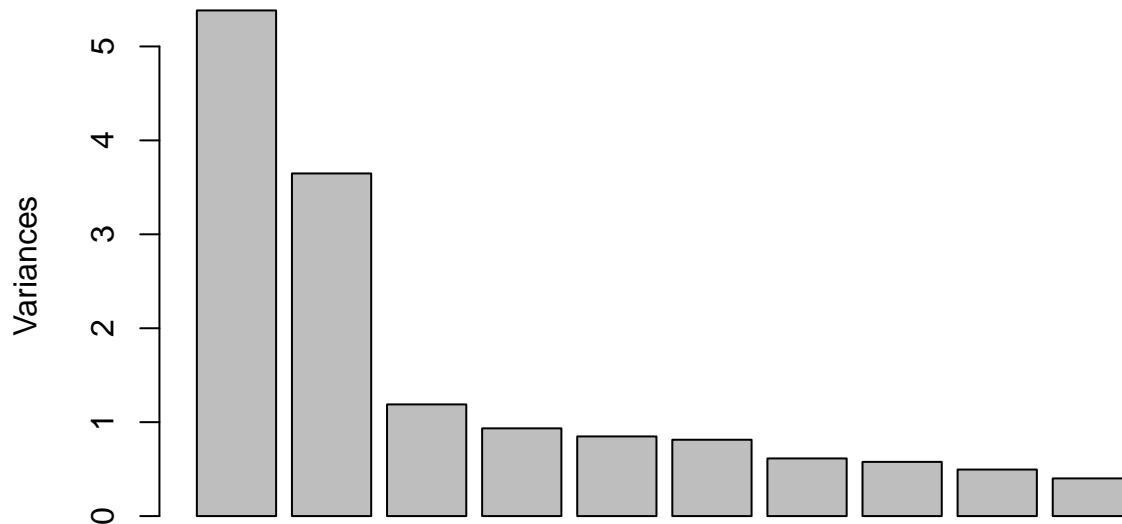
```

```

#Scree Plot
plot(collegePCA)

```

collegePCA



We need 14 eigenvalues for us to explain 95% of the variances in the data.

#loadings for the first and second PCs

```
collegePCA$rotation[,1]
```

```
## PrivateYes      Enroll    Top10perc    Top25perc F.Undergrad P.Undergrad
## -0.20681302  0.03464939 -0.34312918 -0.31666718  0.06037906  0.12071330
## Outstate    Room.Board      Books      Personal      PhD      Terminal
## -0.37484355 -0.28237524 -0.03323706  0.13546497 -0.24369433 -0.24860292
## S.F.Ratio  perc.alumni      Expend    Grad.Rate
## 0.26977520 -0.29132731 -0.33637777 -0.29682521
```

```
collegePCA$rotation[,2]
```

```
## PrivateYes      Enroll    Top10perc    Top25perc F.Undergrad P.Undergrad
## 0.356455041 -0.462454913 -0.157492803 -0.190913384 -0.473283545 -0.332095123
## Outstate    Room.Board      Books      Personal      PhD      Terminal
## 0.073682797 -0.002834249 -0.098129260 -0.203608582 -0.300157933 -0.287163985
## S.F.Ratio  perc.alumni      Expend    Grad.Rate
## -0.134839801  0.102080574 -0.072088937  0.025669164
```

The first component is proportional to each variable, and the second component measures the difference between the first pair of variables and the second pair of variables.

(b)

```
library(pls)
```

```
## Warning: package 'pls' was built under R version 4.0.3
```

```
##
## Attaching package: 'pls'

## The following object is masked from 'package:caret':
##
##      R2

## The following object is masked from 'package:corrplot':
##
##      corrplot

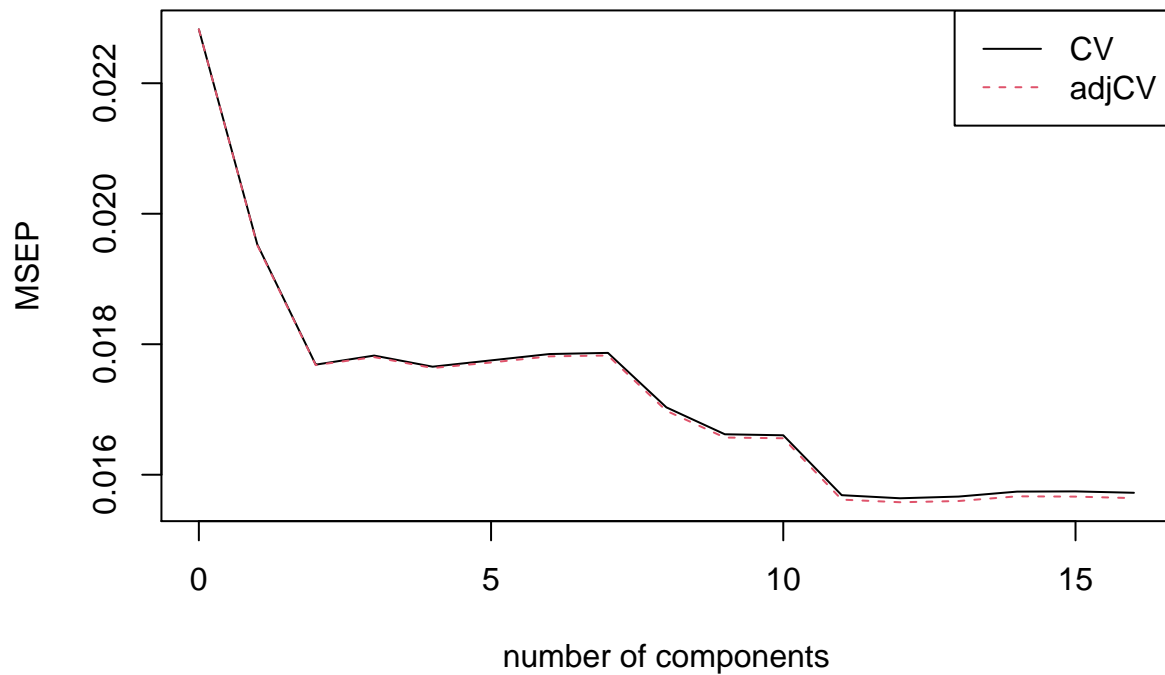
## The following object is masked from 'package:stats':
##
##      loadings

collegePCR <- pcr(Accept.Apps ~ ., data = training, scale = TRUE, validation = "CV")
summary(collegePCR)

## Data:      X dimension: 545 16
## Y dimension: 545 1
## Fit method: svdpc
## Number of components considered: 16
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##      (Intercept) 1 comps 2 comps 3 comps 4 comps 5 comps 6 comps
## CV      0.1511  0.1398  0.133  0.1335  0.1329  0.1332  0.1336
## adjCV    0.1511  0.1397  0.133  0.1334  0.1328  0.1331  0.1335
##      7 comps 8 comps 9 comps 10 comps 11 comps 12 comps 13 comps
## CV      0.1337  0.1305  0.1289  0.1289  0.1252  0.1251  0.1252
## adjCV    0.1335  0.1303  0.1287  0.1287  0.1250  0.1248  0.1249
##      14 comps 15 comps 16 comps
## CV      0.1255  0.1255  0.1254
## adjCV    0.1252  0.1252  0.1251
##
## TRAINING: % variance explained
##      1 comps 2 comps 3 comps 4 comps 5 comps 6 comps 7 comps
## X      33.66  56.40  63.62  69.64  75.27  80.14  83.90
## Accept.Apps 14.84  22.91  23.75  24.30  25.05  25.22  25.98
##      8 comps 9 comps 10 comps 11 comps 12 comps 13 comps 14 comps
## X      87.37  90.47  93.07  95.33  97.30  98.46  99.29
## Accept.Apps 29.73  31.19  31.26  35.38  36.12  36.65  36.66
##      15 comps 16 comps
## X      99.81  100.00
## Accept.Apps 37.21  37.46

validationplot(collegePCR, val.type = "MSEP", legendpos = "topright")
```

Accept.Apps



Hence the value m is chosen with 14

```
collegePCR.pred <- predict(collegePCR, College[-inTrain, names(College) != 'Salary'], ncomp = 14)
PCRTTestMSE <- mean((collegePCR.pred - College[-inTrain, "Accept.Apps"])^2)
PCRTTestMSE #test error obtained
```

```
## [1] 0.01333012
```

(C)

```
PCRTTestMSE
```

```
## [1] 0.01333012
```

#	TestMSE	TrainMSE
#Best reduced OLS	0.01441521	0.01315906
#Ridge Regression	0.01432566	0.01334943
#Lasso	0.01554418	0.01405796

The PC regression performs the lowest testMSE for this dataset. Although it requires a lot of calculation, the differences of the testMSEs are significant. So PCR should be chosen as the approach to analyze this dataset.