STAT223 Final exam-ZihaoHuang

Zihao Huang

May 8, 2018

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
##Final Exam for STAT223
##Zihao Huang
##

#1.
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 3.4.3
```

```
firms <- read_excel("C:/Users/simon/Desktop/STAT223/firms.xlsx")

firm.b<-subset(firms, Status=="B", select=c(x1,x2,x3,x4))
firm.s<-subset(firms, Status=="S", select=c(x1,x2,x3,x4))
n1<-nrow(firm.b)
n2<-nrow(firm.s)
s1<-var(firm.b)
s2<-var(firm.s)
meany1<- apply(firm.b,2,mean)
meany2<- apply(firm.s,2,mean)
meandiff<-(meany1-meany2)
p<-4
k<-2
sp<-((n1-1)*s1+(n2-1)*s2)/(n1+n2-2)
##A)
(T2<-(n1*n2/(n1+n2))*t(meandiff)%*%solve(sp)%*%(meandiff))
```

```
##          [,1]
## [1,] 41.41408
```

```
p*(n1+n2-2)/(n1+n2-p-1)*qf(.95,p, n1+n2-p-1)
```

```
## [1] 11.16084
```

```
##B)
(a<-solve(sp)%*%(meany1-meany2))
```

```
##         [,1]
## x1 -1.197074
## x2 -8.517403
## x3 -1.689613
## x4  2.247780
```

```
(a.star <- sqrt(diag(sp))*solve(sp)%*%meandiff)
```

```
##          [,1]
## x1 -0.2565645
## x2 -0.8838825
## x3 -1.3572974
## x4  0.4173275
```

```
###The ranking for variables is x2,x4,x3,x1
###
##C)
(zc<-0.5*t(a)%*%(meany1+meany2))
```

```
##           [,1]
## [1,] -2.360438
```

```
###The rule is -2.360438
z<-as.matrix(firms[,1:4])%*%a
class.1<-c(rep("B",n1+n2))
class.1[which(z>=-2.360438)]<-"S"

library(MASS)
m1 <- lda(Status~x1+x2+x3+x4, data=firms, prior=rep(1,k)/k)
pred1 <- predict(m1)$class #Predicting each state
pre<-data.frame(firms$Status,pred1)
table(pre)
```

```
##             pred1
## firms.Status  B  S
##            B 18  3
##            S  1 24
```

```
1-sum(diag(table(pre)))/sum(table(pre))
```

```
## [1] 0.08695652
```

```
###apparent error rate is 0.0870
##D)



##E)
m.cv <- lda(Status~x1+x2+x3+x4, data=firms, prior=rep(1,k)/k,CV=T)
pred2<-m.cv$class
pre.cv<-data.frame(firms$Status,pred2)#Comparing our predictions and R predictions
table(pre.cv)
```

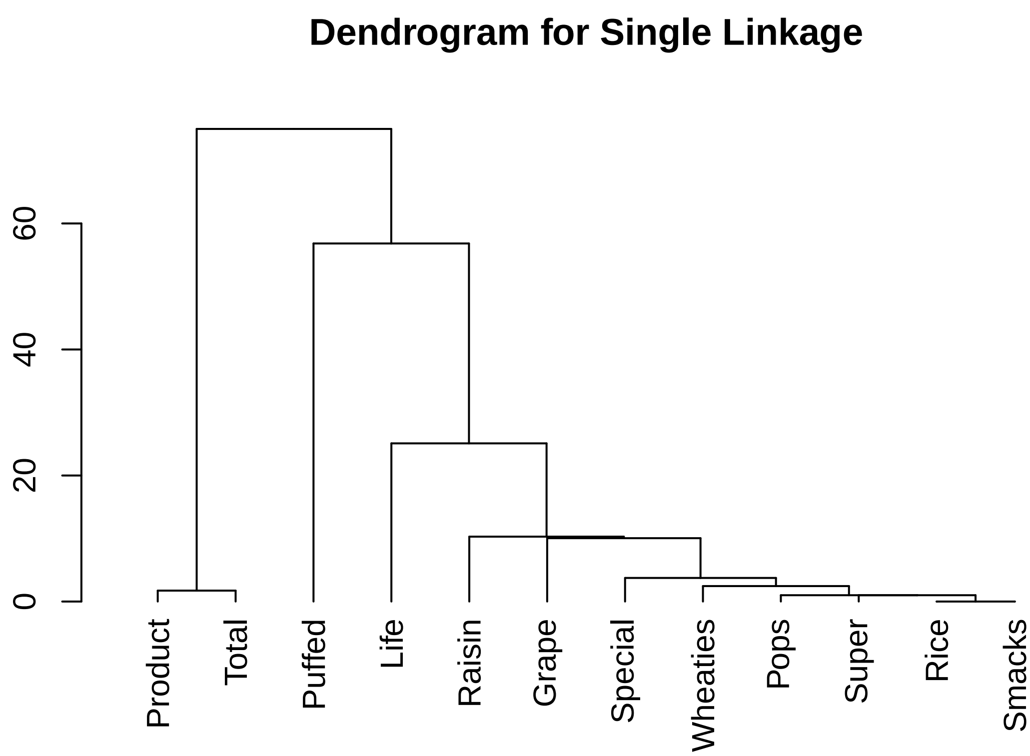
```
##             pred2
## firms.Status  B  S
##            B 18  3
##            S  2 23
```

```
1-sum(diag(table(pre.cv)))/sum(table(pre.cv))
```

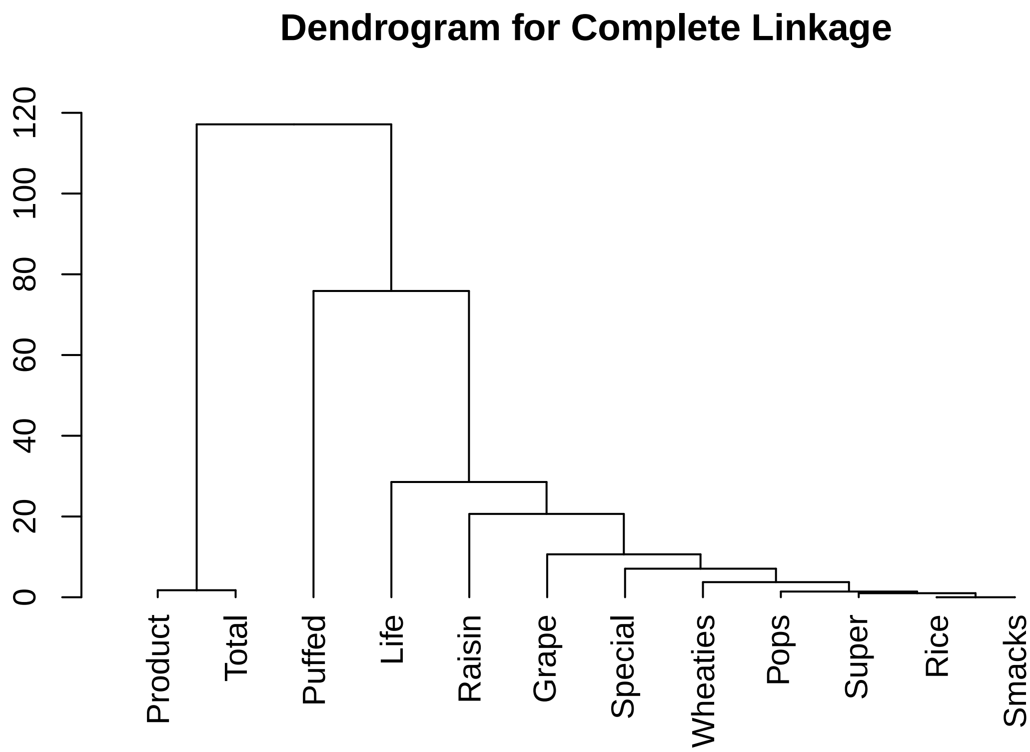
```
## [1] 0.1086957
```

```
###apparent rate for CV is 0.1087

#2.
```

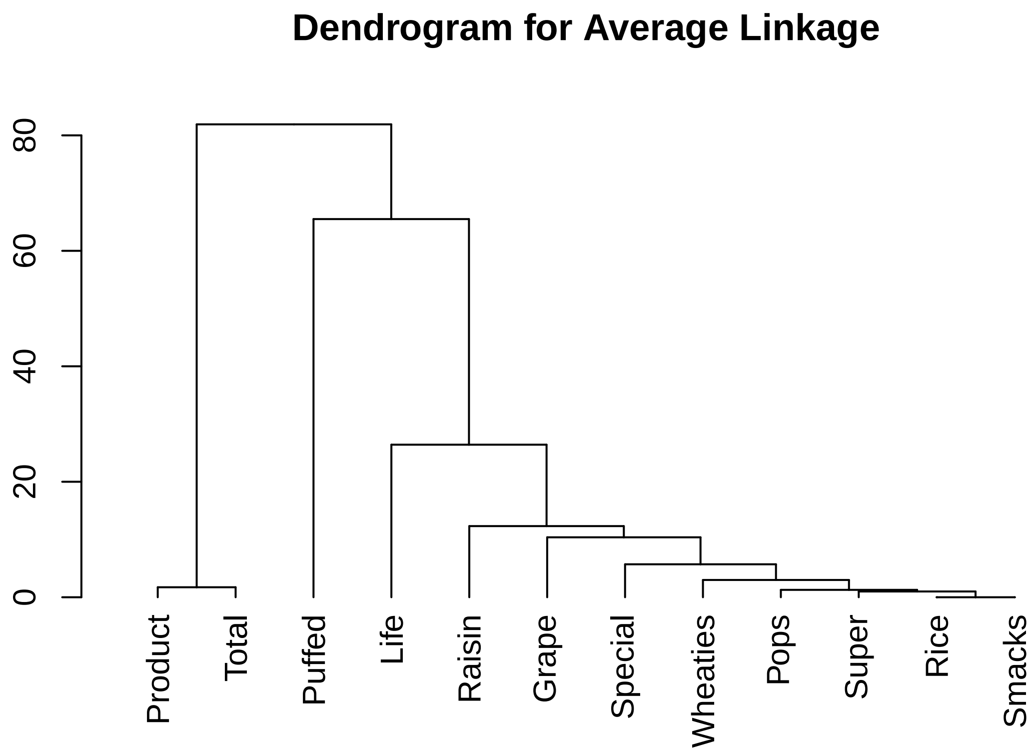
```
library(readxl)
cereal <- read_excel("C:/Users/simon/Desktop/STAT223/cereal.xlsx")
ce<-as.data.frame(cereal[,-1])
rownames(ce)<-cereal$Name
##A)
D <- (dist(ce,diag=T, upper=T))
cl.sin<-hclust(D,method="single")
plot(as.dendrogram(cl.sin),main="Dendrogram for Single Linkage")
```



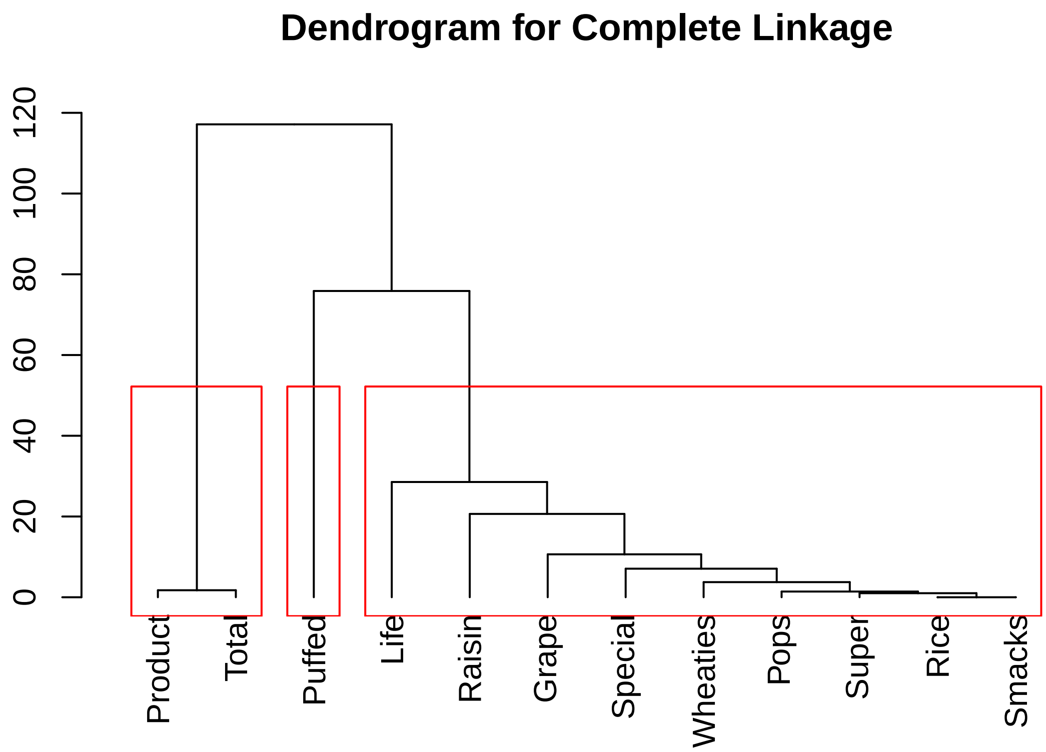

```
cl.com<-hclust(D,method="complete")
plot(as.dendrogram(cl.com),main="Dendrogram for Complete Linkage")
```




```
cl.ave<-hclust(D,method="average")
plot(as.dendrogram(cl.ave),main="Dendrogram for Average Linkage")
```




```
##B,C
cl.com<-hclust(D,method="complete")
plot(as.dendrogram(cl.com),main="Dendrogram for Complete Linkage")
rect.hclust(cl.com,k=3,border="red")
```




```
###We will choose the clusters before large gaps are shown. Hence, it is suggested to use k=3.
##D
###The method that clustering the cereals are determined by the furthest distances
###from each cluster. Because Complete linkage tends to increase the distances
###between clusters, to find compact clusters.
group1<-apply(ce[c(7,9),],2,mean)
group2<-ce[10,]
group3<-apply(ce[c(-7,-9,-10)],2,mean)
rbind(group1,group2,group3)
```

```
##         Protein Carbs       Fat Calories   Vitamin
## 1      2.500000  23.5 0.5000000      110 100.00000
## Puffed 1.000000  13.0 0.0000000       50   0.00000
## 3      2.833333  23.0 0.3333333      105  33.33333
```

```
###Comparing to the group 1 (total and product) and puffed, they have different vitamin, crabs
###protein, calories. group 1 and group 3 have different vitamin.
###pUffed and group 3 have different vitamin, carbs, calories and protein.
###

##E
###The next step is gathering puffed to group 3
n1<-1
n2<-9
c1<-ce[10,]
c2<-ce[c(-7,-9,-10),]
s22<-var(c2)
```

```r
s21<-matrix(rep(0,25),nrow=5)
meany21<- apply(c1,2,mean)
meany22<- apply(c2,2,mean)
meandiff2<-(meany21-meany22)
p2<-5
k2<-2
sp2<-((n1-1)*s21+(n2-1)*s22)/(n1+n2-2)
(T2<-(n1*n2/(n1+n2))*t(meandiff2)%*%solve(sp2)%*%(meandiff2))
```

```
##         [,1]
## [1,] 208.65
```

```r
p2*(n1+n2-2)/(n1+n2-p2-1)*qf(.95,p2, n1+n2-p2-1)
```

```
## [1] 62.56057
```

```r
###Given T2 is 208.65>62.561,we reject the null hypothesis that the group 3
###is significant different from puffed. We believe puffed should not be
##combined.

#3.
house1<- read.table("C:/Users/simon/Desktop/STAT223/housdat.txt",header=T)
house<-house1[,-14]
R<-cor(house)
(S<-cov(house))
```

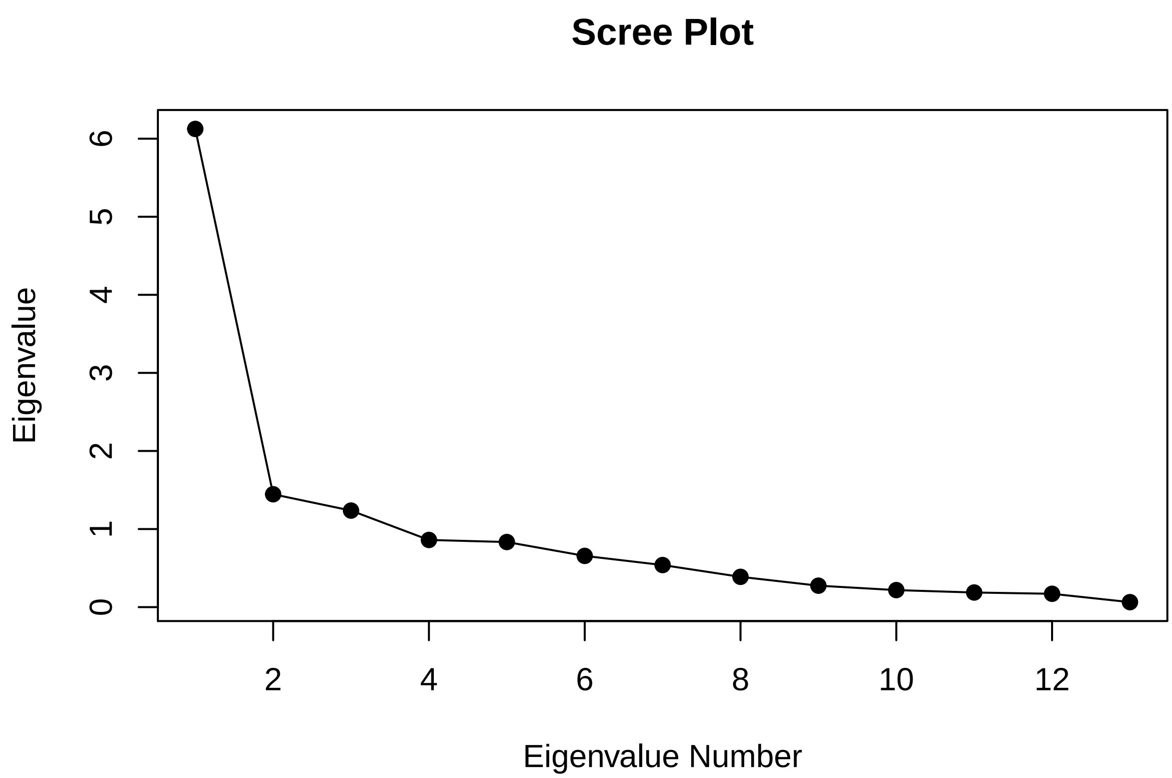
```
##              CRIM         PLAND         PBUS          OCE          NOC
## CRIM    75.6202780   -40.7010889   24.3172316 -0.131767774  0.423911451
## PLAND  -40.7010889   536.6461394  -84.5794680 -0.249468164 -1.394393433
## PBUS    24.3172316   -84.5794680   47.5564906  0.106425568  0.613862543
## OCE     -0.1317678    -0.2494682    0.1064256  0.066087831  0.002622183
## NOC      0.4239115    -1.3943934    0.6138625  0.002622183  0.013518673
## ARM     -1.3337745     4.9564037   -1.8842788  0.017220036 -0.024630519
## PAGE    85.9501263  -369.5565651  125.5117628  0.602827801  2.355132102
## WDIS    -6.9965642    32.7842043  -10.3453476 -0.053202865 -0.189219432
## INDEX   47.4280227   -63.8008872   35.8339158 -0.029877637  0.620733651
## FTAX   856.0301629 -1240.3587769  842.9365819 -1.786280282 13.167866216
## PTR      5.5384673   -18.8077419    5.6161946 -0.068458006  0.045929876
## BK    -307.6120299   377.4457947 -227.0080018  1.214016475 -4.069288884
## LSP     28.2121160   -68.3591947   29.7780481 -0.110223083  0.489055483
##               ARM         PAGE         WDIS        INDEX        FTAX
## CRIM    -1.33377449   85.9501263   -6.99656420  47.42802268   856.03016
## PLAND    4.95640372 -369.5565651   32.78420430 -63.80088722 -1240.35878
## PBUS    -1.88427878  125.5117628  -10.34534763  35.83391576   842.93658
## OCE      0.01722004    0.6028278   -0.05320286  -0.02987764    -1.78628
## NOC     -0.02463052    2.3551321   -0.18921943   0.62073365    13.16787
## ARM      0.49863053   -4.7447021    0.30486781  -1.26373953   -34.42451
## PAGE    -4.74470214  770.3394058  -44.30780148 111.95731955  2422.06911
## WDIS     0.30486781  -44.3078015    4.49299575  -9.19873285  -192.45056
## INDEX   -1.26373953  111.9573195   -9.19873285  76.42315177  1347.50629
## FTAX   -34.42451001 2422.0691069 -192.45056073 1347.50629133 28735.85477
## PTR     -0.53942010   15.1129105   -1.03013163   8.99175118   172.19750
## BK       8.12546138 -704.8166207   57.13471678 -356.95435133 -6883.63659
## LSP     -3.09776467  119.9937631   -7.55775359  30.13938781   654.35369
##               PTR           BK          LSP
```

```
## CRIM    5.53846728  -307.612030   28.2121160
## PLAND -18.80774192   377.445795  -68.3591947
## PBUS    5.61619463  -227.008002   29.7780481
## OCE    -0.06845801     1.214016   -0.1102231
## NOC     0.04592988    -4.069289    0.4890555
## ARM    -0.53942010     8.125461   -3.0977647
## PAGE   15.11291050  -704.816621  119.9937631
## WDIS   -1.03013163    57.134717   -7.5577536
## INDEX   8.99175118  -356.954351   30.1393878
## FTAX  172.19750161 -6883.636586  654.3536878
## PTR     4.65497065   -35.646525    5.7994388
## BK    -35.64652516  8518.766774 -239.2574486
## LSP     5.79943882  -239.257449   51.2365898
```

```r
##A)
###choose R than S.Because covariance have larger variances rather than others. These would
###lead to the variable with high variance explains most of the data.

##B)
E <- eigen(R)$vectors
eval<-eigen(R)$values
Lambda <- diag(eigen(R)$values)
diag(Lambda)
```

```
##  [1] 6.12486829 1.44581485 1.23668623 0.86010904 0.83401244 0.65616893
##  [7] 0.53900091 0.38800683 0.27489869 0.21807399 0.18740270 0.17092471
## [13] 0.06403238
```

```r
plot(1:13,diag(Lambda), xlab="Eigenvalue Number", ylab = "Eigenvalue",
     main= "Scree Plot", pch=19); lines(1:13, diag(Lambda))
```



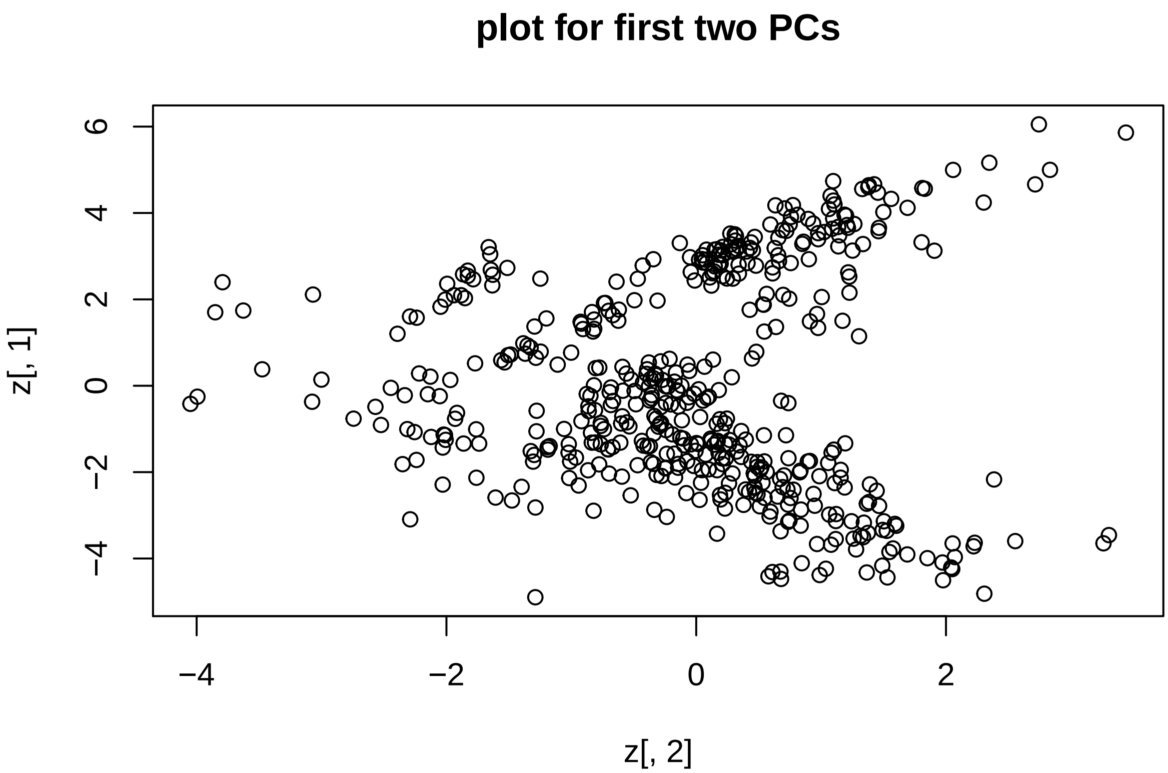

```
percentage <- rep(0,14)
for (i in 1:13){
  percentage[i] <- sum(diag(Lambda)[1:i])/sum(diag(Lambda))
}
percentage
```

```
##  [1] 0.4711437 0.5823602 0.6774900 0.7436522 0.8078070 0.8582815 0.8997431
##  [8] 0.9295898 0.9507359 0.9675108 0.9819264 0.9950744 1.0000000 0.0000000
```

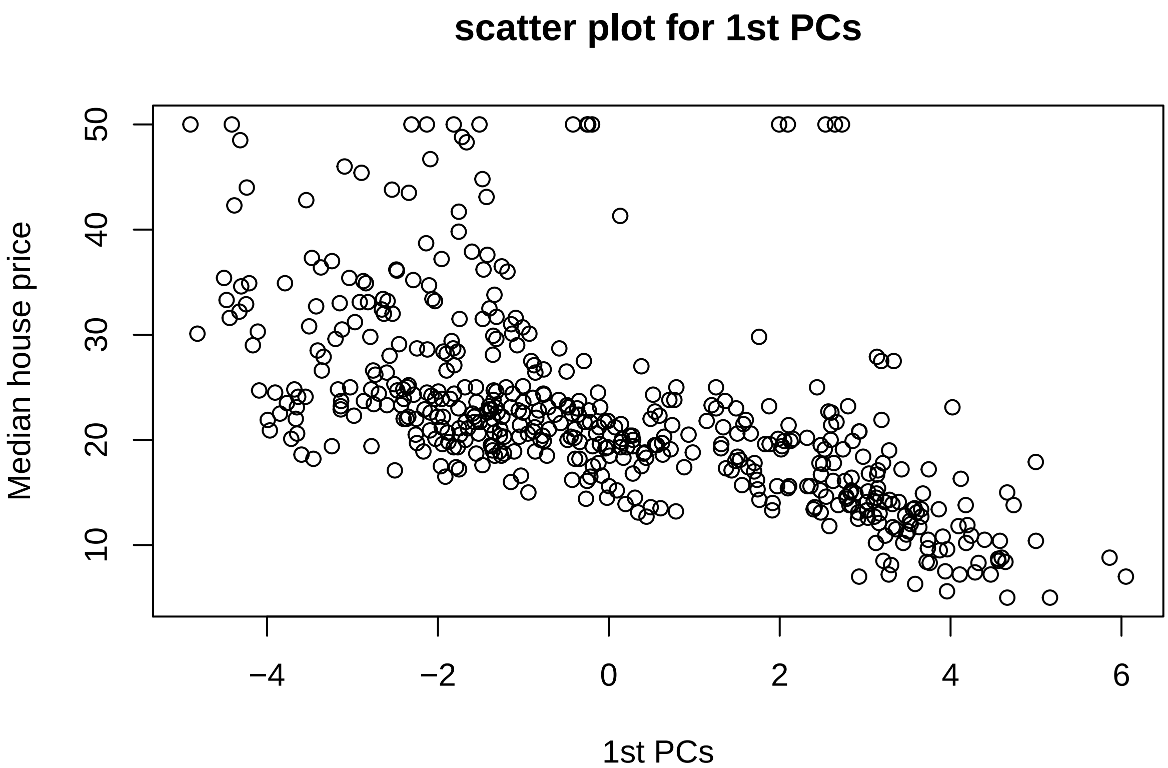
```
length(eval[eval>mean(eval)])
```

```
## [1] 3
```

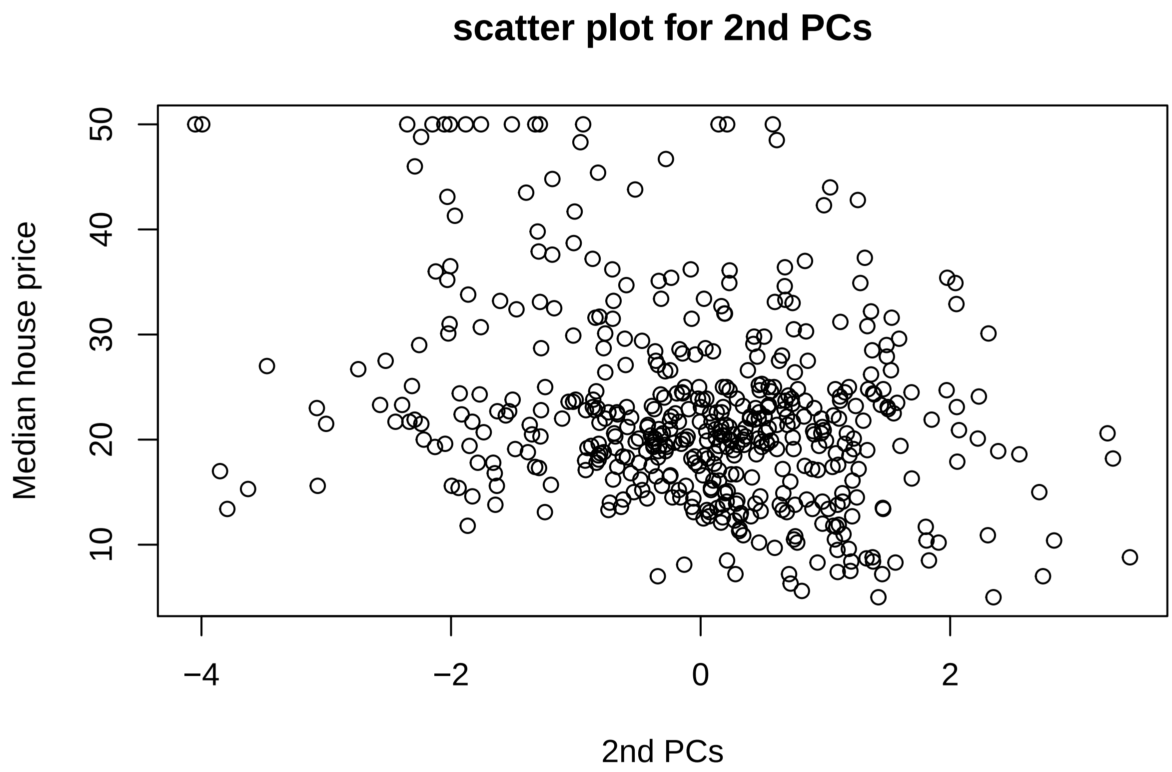
```
###There are 3 eigenvalues larger than mean of eigenvalues.
###The scree plot shows that a huge decrease before m=4, and the percentage of
###egienvalues explained the total variance shows that m=5 explains 80.78%
###Hence, we choose m=5

##C)
###m=5, 5 eigenvalues explains 80.78%

##D)
h.sc <- scale(house, center=T, scale=apply(house,2,sd))
z<-h.sc%*%E[,1:2]
plot(z[,1]~z[,2],main="plot for first two PCs")
```



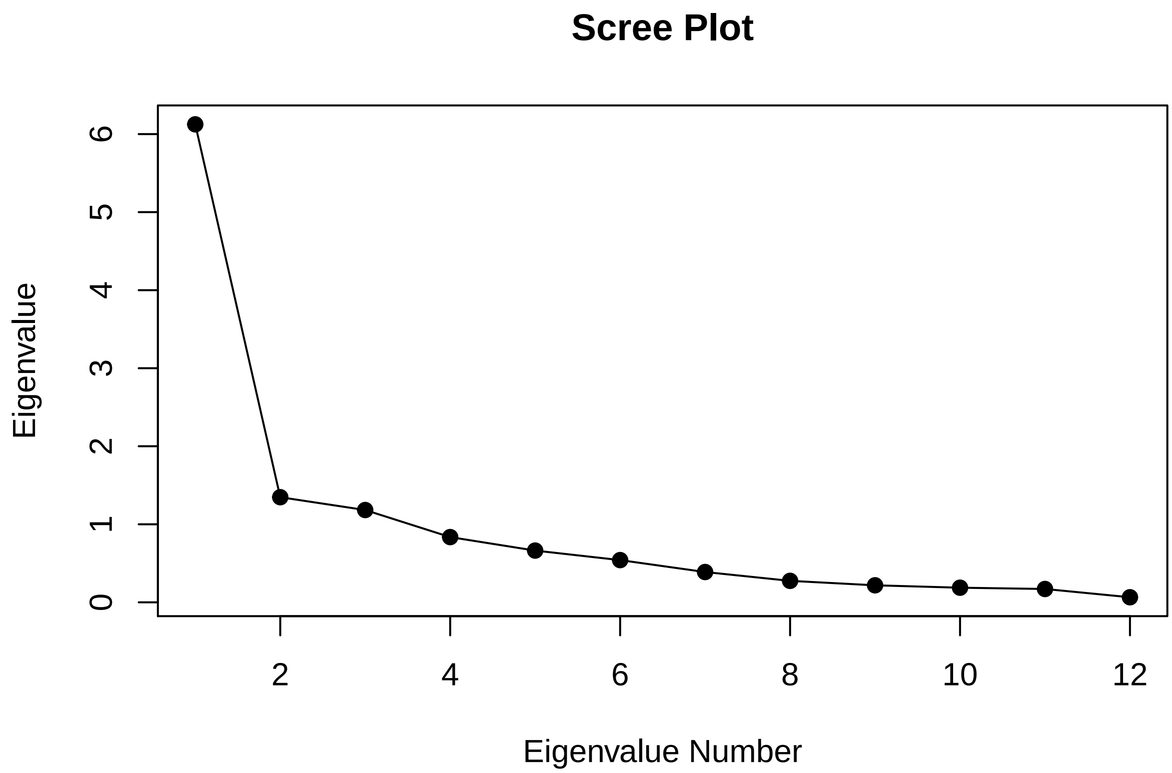

```
###The 1st Pcs seperated the observations better than the 2nd PCs.
###The 1st Pcs seperated 3 groups(at least).

##E)
plot(house1$MED~z[,1],main="scatter plot for 1st PCs",xlab="1st PCs", ylab="Median house price")
```




```
###The scatter plot for 1st PCs shows the observations are seperated well,
###it is believed the 1st PCs is interpretable.
plot(house1$MED~z[,2],main="scatter plot for 2nd PCs",xlab="2nd PCs", ylab="Median house price")
```




```
###The scatter plot for 2nd PCs showsit doesn't seperated the observations
###well, the data gathered together, it should not be used for interpretation.

#4.
h4<-house[,-4]
R<-cor(h4)
n<-nrow(h4)
p<-ncol(h4)
evec.4<-eigen(R)$vectors
eval.4<-eigen(R)$values
plot(1:12,eval.4, xlab="Eigenvalue Number", ylab = "Eigenvalue",
     main= "Scree Plot", pch=19); lines(1:12,eval.4)
```




```r
percentage1 <- rep(0,12)
for (i in 1:12){
  percentage1[i] <- sum(eval.4[1:i])/sum(eval.4)
}
percentage1
```

```
##  [1] 0.5104021 0.6226708 0.7212335 0.7909049 0.8461643 0.8912414 0.9235968
##  [8] 0.9465229 0.9647001 0.9803252 0.9945713 1.0000000
```

```r
length(eval.4[eval.4>mean(eval.4)])
```

```
## [1] 3
```

```r
##B)
FA.ML5 <- factanal(x=h4,factors=5,rotation = "varimax")
FA.ML4 <- factanal(x=h4,factors=4,rotation = "varimax")
FA.ML3 <- factanal(x=h4,factors=3,rotation = "varimax")
Psi.ml5 <- diag(diag(R-FA.ML5$loadings%*%t(FA.ML5$loadings)))
FA5 <- (FA.ML5$loadings%*%t(FA.ML5$loadings)+Psi.ml5)
ll5 <- -n/2*(log(det(FA5))+sum(diag(solve(FA5)%*%R)))
Psi.ml4 <- diag(diag(R-FA.ML4$loadings%*%t(FA.ML4$loadings)))
FA4 <- (FA.ML4$loadings%*%t(FA.ML4$loadings)+Psi.ml4)
ll4 <- -n/2*(log(det(FA4))+sum(diag(solve(FA4)%*%R)))
Psi.ml3 <- diag(diag(R-FA.ML3$loadings%*%t(FA.ML3$loadings)))
FA3 <- (FA.ML3$loadings%*%t(FA.ML3$loadings)+Psi.ml3)
ll3 <- -n/2*(log(det(FA3))+sum(diag(solve(FA3)%*%R)))
```

```
k5<- p*(5+1)-5*(5-1)
AIC5<- -2*ll5+2*k5
k4<- p*(4+1)-4*(4-1)
AIC4<- -2*ll4+2*k4
k3<- p*(3+1)-3*(3-1)
AIC3<- -2*ll3+2*k3
cbind(AIC5,AIC4,AIC3)
```

```
##          AIC5     AIC4     AIC3
## [1,] 1710.044 1839.185 1983.508
```

```
###Since AIC5=1710.044 is smallest, we choose m=5
sum(eigen(R)$values[1:5])/sum(eigen(R)$values)
```

```
## [1] 0.8461643
```

```
###It explains 84.62% of the total variance.
##C)
FA.ML5$loadings
```

```
##
## Loadings:
##       Factor1 Factor2 Factor3 Factor4 Factor5
## CRIM   0.178   0.601   0.179
## PLAND -0.736          -0.141  -0.384
## PBUS   0.595   0.416   0.304   0.162   0.456
## NOC    0.691   0.495   0.210           0.213
## ARM   -0.126          -0.625  -0.249  -0.115
## PAGE   0.729   0.309   0.278
## WDIS  -0.852  -0.309  -0.105
## INDEX  0.229   0.916           0.271
## FTAX   0.269   0.835   0.154   0.233   0.283
## PTR            0.275   0.237   0.707
## BK    -0.169  -0.432  -0.165
## LSP    0.343   0.384   0.850
##
##                Factor1 Factor2 Factor3 Factor4 Factor5
## SS loadings      2.959   2.925   1.500   0.885   0.369
## Proportion Var   0.247   0.244   0.125   0.074   0.031
## Cumulative Var   0.247   0.490   0.615   0.689   0.720
```

```
###And the 4th factor and 5th factor are trivial factors. Because they only explain one variable.
```