Q. Perform clustering (Both hierarchical and K means clustering) for the airlines data to obtain optimum number of clusters.

Draw the inferences from the clusters obtained.

#Setting up Enviromnent for Clustering

>

> list.of.packages <- c("datasets", "ggplot2","cluster")

> new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()[,"Package"])]

> if(length(new.packages)) install.packages(new.packages, repos="http://cran.rstudio.com/")

> library(cluster)

> library(ggplot2)

> library(caret)

> #Importing Dataset

> getwd()

> library(openxlsx)

> Airlines <- read.xlsx(file.choose(),2)

> View(Airlines)

> a <-Airlines[,-1]

> View(a)

#\*\*\*\*\*\*\*\*\*\*\*\*K-means clustering\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

> scale\_a <- scale(a)

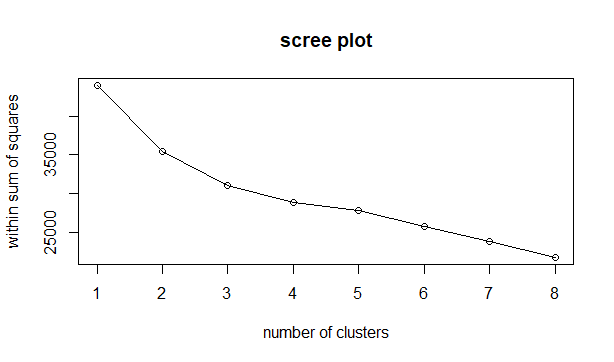
> View(scale\_a)

> wss=(nrow(scale\_a)-1)\*sum(apply(scale\_a,2,var))

for(i in 2:8)wss[i]=sum(kmeans(scale\_a,centers = i)$withinss)

> plot(1:8,wss,type = "o",xlab = "number of clusters",ylab = "within sum of squares")

> title("scree plot")



> twss <- NULL

> for(i in 2:8){

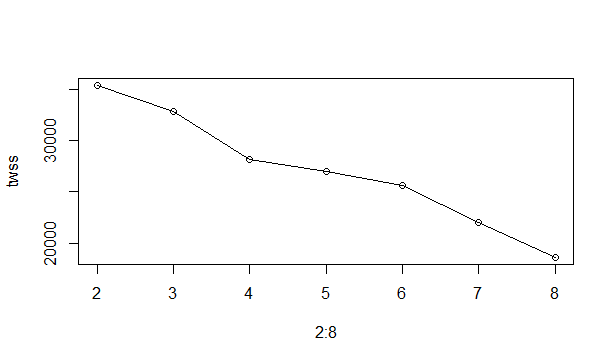
+ twss <- c(twss,kmeans(scale\_a,i)$tot.withinss)

+ }

> twss

[1] 35400.65 32881.99 28186.53 26973.47 25656.32 22026.72 18614.76

plot(2:8,twss,type = "o")



> #from elbow curve it is clear that k = 3 or 5

> km1 <- kmeans(scale\_a,3)

> str(km1) #twss=31001 ,bss=12977

List of 9

$ cluster : Named int [1:3999] 3 3 3 3 2 3 2 1 2 2 ...

..- attr(\*, "names")= chr [1:3999] "1" "2" "3" "4" ...

$ centers : num [1:3, 1:11] -0.1371 0.848 -0.2759 0.1477 0.0851 ...

..- attr(\*, "dimnames")=List of 2

.. ..$ : chr [1:3] "1" "2" "3"

.. ..$ : chr [1:11] "Balance" "Qual\_miles" "cc1\_miles" "cc2\_miles" ...

$ totss : num 43978

$ withinss : num [1:3] 6261 17589 9032

$ tot.withinss: num 32882

$ betweenss : num 11096

$ size : int [1:3] 929 867 2203

$ iter : int 3

$ ifault : int 0

- attr(\*, "class")= chr "kmeans"

km2 <- kmeans(scale\_a,5)

> str(km2) #twss=26308,bss=17670

List of 9

$ cluster : Named int [1:3999] 1 1 1 1 2 1 2 3 4 2 ...

..- attr(\*, "names")= chr [1:3999] "1" "2" "3" "4" ...

$ centers : num [1:5, 1:11] -0.139 0.645 -0.152 1.218 -0.387 ...

..- attr(\*, "dimnames")=List of 2

.. ..$ : chr [1:5] "1" "2" "3" "4" ...

.. ..$ : chr [1:11] "Balance" "Qual\_miles" "cc1\_miles" "cc2\_miles" ...

$ totss : num 43978

$ withinss : num [1:5] 2654 10029 6686 4675 2937

$ tot.withinss: num 26981

$ betweenss : num 16997

$ size : int [1:5] 991 845 846 144 1173

$ iter : int 4

$ ifault : int 0

- attr(\*, "class")= chr "kmeans"

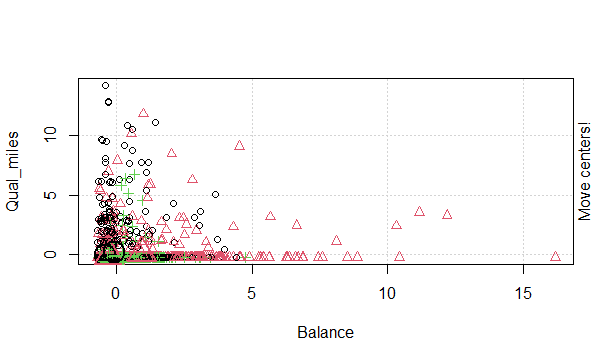
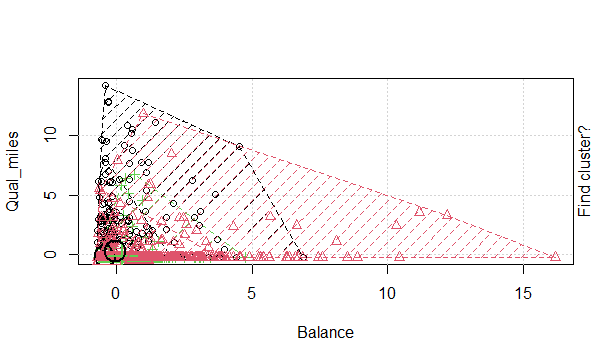
>> #the best cluster have less totwss and high betweenss

> #hence choosing number of clusters as 5

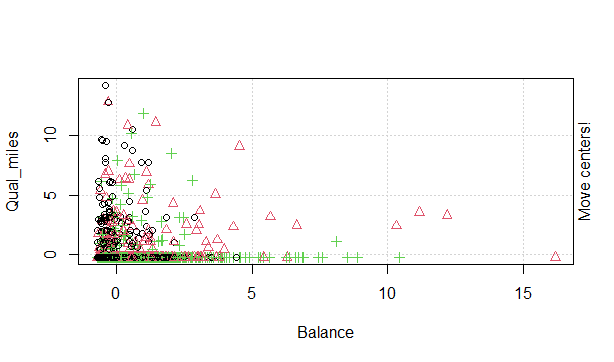
> library(animation)

km <- kmeans(scale\_a,5)

kmanimation <- kmeans.ani(scale\_a,5)



kmfinal <- data.frame(km$cluster,A)



> kmfinal <- data.frame(km$cluster,Airlines)

> View(kmfinal)

> aggregate(kmfinal[,-c(1,2)],by=list(km$cluster),FUN = mean)

Group.1 Balance Qual\_miles cc1\_miles cc2\_miles cc3\_miles Bonus\_miles

1 1 43540.09 72.62764 1.314779 1.000000 1.000480 4810.98

2 2 194887.68 777.66197 2.281690 1.000000 1.000000 34012.88

3 3 68876.58 23.25581 1.139535 2.348837 1.000000 14689.84

4 4 58350.55 237.80848 1.700606 1.000000 1.000000 10724.64

5 5 137921.79 129.65525 4.110497 1.000000 1.053039 48869.44

Bonus\_trans Flight\_miles\_12mo Flight\_trans\_12 Days\_since\_enroll Award.

1 7.161228 152.2610 0.4577735 3633.252 0.0000000

2 28.098592 5764.8169 16.7676056 4679.049 0.7957746

3 17.534884 582.6279 2.2093023 3968.930 0.3953488

4 10.609697 417.0752 1.2593939 4252.360 0.9951515

5 19.861878 369.8420 1.1314917 5033.299 0.5856354

>

> #===================Creating Hierarchical Clustering=====================================

setDT(Airlines)

> head(Airlines)

ID# Balance Qual\_miles cc1\_miles cc2\_miles cc3\_miles Bonus\_miles Bonus\_trans

1: 1 28143 0 1 1 1 174 1

2: 2 19244 0 1 1 1 215 2

3: 3 41354 0 1 1 1 4123 4

4: 4 14776 0 1 1 1 500 1

5: 5 97752 0 4 1 1 43300 26

6: 6 16420 0 1 1 1 0 0

Flight\_miles\_12mo Flight\_trans\_12 Days\_since\_enroll Award?

1: 0 0 7000 0

2: 0 0 6968 0

3: 0 0 7034 0

4: 0 0 6952 0

5: 2077 4 6935 1

6: 0 0 6942 0

> #lets check missing values

> colSums(is.na(Airlines))

ID# Balance Qual\_miles cc1\_miles

0 0 0 0

cc2\_miles cc3\_miles Bonus\_miles Bonus\_trans

0 0 0 0

Flight\_miles\_12mo Flight\_trans\_12 Days\_since\_enroll Award?

0 0 0 0

> #some of the variables have missing values. Let’s impute the missing values with median.

>

> #impute missing values with median

> for(i in colnames(Airlines)[!(colnames(Airlines) %in% c("V1"))])

+ set(x = Airlines,i = which(is.na(Airlines[[i]])), j = i, value = median(Airlines[[i]], na.rm = T))

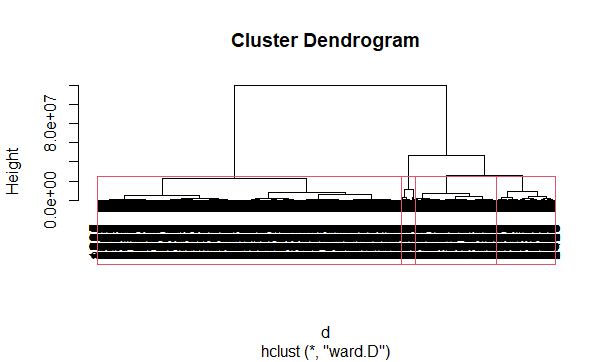
> #scale the variables

>

> d <- dist(Airlines,method = "euclidean") #distance matrix

> h\_clust <- hclust(d, method = "ward.D") #clustering

> plot(h\_clust,labels = Airlines$V1) #dendrogram



> # how can you estimate the number of clusters? Going by the logic of horizontal cut, four clusters are evident. Let’s see!

>

> rect.hclust(h\_clust,k=4)

> #To look at which observation went into which cluster, you can write:

> #extract clusters

> groups <- cutree(h\_clust,k=4)

> cluster\_no <- as.matrix(groups)

> groups

[1] 1 1 1 1 2 1 2 1 3 2 1 2 1 1 1 1 1 1 2 1 4 4 1 1 4 1 1 1 1 1 4 1 4 1 1 4 1 1 1

[40] 1 1 1 1 3 2 4 2 1 2 1 2 1 4 2 1 1 2 1 1 2 2 1 1 3 2 1 1 2 4 1 2 4 3 1 2 2 2 2

[79] 1 1 1 2 1 1 1 1 1 3 1 1 1 1 2 4 4 1 1 2 1 1 1 1 1 1 1 1 2 1 2 1 4 4 4 1 4 1 4

[118] 3 2 1 2 4 2 1 2 4 3 4 1 4 1 1 4 2 2 4 2 2 4 1 1 1 1 4 1 2 1 2 2 1 2 3 3 2 1 4

[157] 4 1 4 1 1 4 4 1 2 1 2 3 3 1 2 1 3 4 4 3 2 1 1 1 1 2 1 1 1 2 3 4 3 1 2 4 3 1 1

[196] 4 1 2 1 4 1 1 1 4 1 1 2 1 1 1 4 1 4 1 1 2 1 1 4 4 3 4 1 1 4 2 2 4 1 4 2 2 1 1

[235] 1 4 1 4 1 1 4 1 4 4 3 1 4 1 2 2 1 1 2 1 2 4 3 1 4 3 4 1 4 1 1 2 1 1 2 2 4 4 2

[274] 1 1 3 1 2 1 1 4 1 4 4 2 2 1 1 4 1 3 4 3 1 1 4 1 4 4 1 1 1 4 3 2 2 2 3 2 1 4 1

[313] 1 3 4 2 1 1 3 1 2 4 1 4 1 3 1 3 4 1 4 1 2 1 4 4 2 1 1 1 4 1 1 2 1 2 2 4 4 2 1

[352] 2 3 1 1 1 4 4 1 2 1 1 4 1 1 2 1 1 1 4 4 2 2 4 1 1 1 4 1 2 3 4 2 3 1 1 1 1 4 2

[391] 2 1 1 1 2 1 2 1 1 1 4 1 2 1 3 4 1 4 1 1 3 2 2 4 1 2 1 1 2 4 3 1 4 1 4 2 1 3 2

[430] 2 4 1 1 4 1 1 2 3 1 1 1 2 1 1 4 2 4 1 3 1 4 1 4 4 1 1 1 4 1 1 1 2 1 1 4 3 3 3

[469] 2 2 1 3 1 1 1 4 4 4 4 2 1 3 1 2 1 1 2 2 3 2 1 1 1 1 4 1 1 1 1 4 4 2 1 4 1 4 4

[508] 1 2 1 3 1 4 1 2 3 1 2 4 1 1 4 2 1 2 1 1 1 1 1 3 2 2 1 3 1 1 1 1 1 1 1 1 1 1 4

[547] 4 1 1 4 2 2 1 1 4 1 1 4 1 1 1 3 4 2 2 4 1 1 1 1 1 1 2 1 2 1 1 1 1 1 4 1 1 1 1

[586] 1 2 2 1 4 4 1 1 1 1 1 1 1 1 3 4 1 4 2 2 2 1 1 3 1 1 4 1 1 1 2 2 4 3 1 1 2 1 4

[625] 3 2 1 1 4 3 3 1 1 1 1 1 1 1 2 2 1 1 1 1 1 1 1 1 1 3 4 4 4 1 1 1 1 2 3 1 1 1 2

[664] 2 1 2 1 1 1 4 1 1 2 1 4 1 1 1 4 4 1 2 4 3 1 4 1 1 2 2 1 2 4 3 1 1 1 3 1 2 1 3

[703] 3 1 4 4 1 1 4 1 1 1 1 1 4 1 4 1 2 1 1 1 1 3 1 4 1 1 1 1 4 1 1 1 1 2 2 1 1 4 2

[742] 1 1 3 1 4 2 2 1 1 4 2 2 1 4 1 4 2 1 1 2 1 1 2 1 4 1 1 1 4 1 4 2 1 1 1 1 4 1 1

[781] 4 3 1 1 1 4 1 4 4 2 1 2 4 2 1 1 2 1 4 1 2 1 1 1 1 1 2 4 1 1 1 1 2 2 4 1 1 1 1

[820] 1 3 4 4 3 1 1 2 1 2 1 1 2 3 2 1 1 4 4 1 2 4 2 4 1 2 1 1 2 1 1 1 2 1 2 1 2 2 1

[859] 2 2 4 4 1 2 2 1 1 1 4 3 2 1 1 1 1 1 1 1 1 1 3 4 1 4 4 1 1 2 3 2 2 4 2 1 1 1 4

[898] 1 1 2 4 4 4 2 1 3 1 1 1 2 1 3 1 1 4 1 1 4 4 1 1 1 4 4 1 1 1 1 4 1 4 1 3 1 2 4

[937] 2 1 1 2 1 1 1 1 1 4 2 4 1 1 1 4 1 2 1 1 1 1 1 2 1 1 1 4 1 2 1 1 2 1 1 1 4 1 1

[976] 1 1 4 1 2 2 4 1 2 4 1 1 2 2 1 1 1 2 2 1 2 1 4 4 3

[ reached getOption("max.print") -- omitted 2999 entries ]

> finaldata <- data.frame(cluster\_no,Airlines)

> View(finaldata)

>

> aggregate(finaldata,by=list(cluster\_no),FUN = mean)

Group.1 cluster\_no ID. Balance Qual\_miles cc1\_miles cc2\_miles cc3\_miles

1 1 1 2202.262 28704.93 97.27139 1.579721 1.016962 1.004146

2 2 2 1826.419 94008.61 186.36158 2.858757 1.008475 1.025424

3 3 3 1056.782 480228.79 440.90323 3.403226 1.024194 1.024194

4 4 4 1537.967 179126.56 256.10311 3.110895 1.007782 1.033074

Bonus\_miles Bonus\_trans Flight\_miles\_12mo Flight\_trans\_12 Days\_since\_enroll

1 8806.132 8.921598 277.0192 0.8413117 3769.907

2 28511.305 15.764124 607.6808 1.8785311 4467.969

3 55704.790 20.604839 1520.4355 4.9274194 5948.968

4 35226.018 17.531128 945.6401 2.5680934 4995.257

Award.

1 0.3124764

2 0.4350282

3 0.7580645

4 0.4863813

>

>

> #o implement PCA, we’ll use princomp base function. For our convenience, we’ll take only the first two components.

>

> #pca

> pcmp <- princomp(Airlines)

> pred\_pc <- predict(pcmp, newdata=Airlines)[,1:2]

> #Now, we’ll create a data frame having pc values and their corresponding clusters. Then, using ggplot2 we’ll create the plot.

>

> comp\_dt <- cbind(as.data.table(pred\_pc),cluster = as.factor(groups), Labels = Airlines$V1)

> ggplot(comp\_dt,aes(Comp.1,Comp.2))+

+ geom\_point(aes(color = cluster),size=3)

>