Week9 Assignment

Mohit Mehndiratta

30/09/2021

# Clustering

## Question 1

### Use the iris dataset built in R.

library(ISLR)  
irisDS <- iris  
head(irisDS)

## Sepal.Length Sepal.Width Petal.Length Petal.Width Species  
## 1 5.1 3.5 1.4 0.2 setosa  
## 2 4.9 3.0 1.4 0.2 setosa  
## 3 4.7 3.2 1.3 0.2 setosa  
## 4 4.6 3.1 1.5 0.2 setosa  
## 5 5.0 3.6 1.4 0.2 setosa  
## 6 5.4 3.9 1.7 0.4 setosa

attach(irisDS)  
dim(irisDS)

## [1] 150 5

str(irisDS)

## 'data.frame': 150 obs. of 5 variables:  
## $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...  
## $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...  
## $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...  
## $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...  
## $ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

The iris data set has 150 Observations of 5 variables. All variables are numeric except Species.

## Part (a): k-means clustering

In clustering we try to divide the data into groups as per their similarities. Here we ignore the categorical variable and only use predictor variables.

### 1. Cluster the dataset into 2 groups using k-means clustering.

km.iris <- kmeans(irisDS[, 1:4], centers = 2, nstart = 20)  
km.iris

## K-means clustering with 2 clusters of sizes 53, 97  
##   
## Cluster means:  
## Sepal.Length Sepal.Width Petal.Length Petal.Width  
## 1 5.005660 3.369811 1.560377 0.290566  
## 2 6.301031 2.886598 4.958763 1.695876  
##   
## Clustering vector:  
## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2  
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [149] 2 2  
##   
## Within cluster sum of squares by cluster:  
## [1] 28.55208 123.79588  
## (between\_SS / total\_SS = 77.6 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

The above output displays the data set converted into 2 clusters using k-means clustering.

The “cluster means” in the output shows the centroid information of 2 clusters based on the predictor variables. The “clustering vector” depicts which observation represents what clusters. Here we can see that most of the observations belong to second cluster.

The total withinss of the clusters depicts the total sum of squares obtained from every cluster.

km.iris$tot.withinss

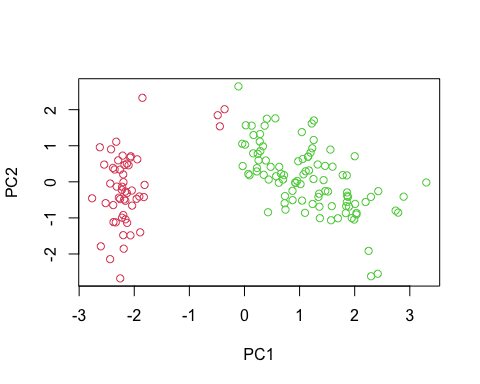
## [1] 152.348

The output shows the total sum of squares obtained from clusters. The total sum of squares is “152.348”

### 2. Draw a plot to visualize the clusters.

We can use PCA for visualisation.

pca.iris <- prcomp(irisDS[,1:4], scale. = TRUE)  
plot(pca.iris$x[,1:2], col = km.iris$cluster + 1)



### 3. Cluster the dataset into 3 groups using k-means clustering.

km.iris3 <- kmeans(irisDS[, 1:4], centers = 3, nstart = 20)  
km.iris3

## K-means clustering with 3 clusters of sizes 62, 38, 50  
##   
## Cluster means:  
## Sepal.Length Sepal.Width Petal.Length Petal.Width  
## 1 5.901613 2.748387 4.393548 1.433871  
## 2 6.850000 3.073684 5.742105 2.071053  
## 3 5.006000 3.428000 1.462000 0.246000  
##   
## Clustering vector:  
## [1] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3  
## [38] 3 3 3 3 3 3 3 3 3 3 3 3 3 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [75] 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 2 2 2 2 1 2 2 2 2  
## [112] 2 2 1 1 2 2 2 2 1 2 1 2 1 2 2 1 1 2 2 2 2 2 1 2 2 2 2 1 2 2 2 1 2 2 2 1 2  
## [149] 2 1  
##   
## Within cluster sum of squares by cluster:  
## [1] 39.82097 23.87947 15.15100  
## (between\_SS / total\_SS = 88.4 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

The above output displays the data set converted into 3 clusters using k-means clustering.

The “cluster means” in the output shows the centroid value of 2 clusters based on the predictor variables. The “clustering vector” depicts which observation represents what clusters.

The total withinss of the clusters depicts the total sum of squares obtained from every cluster.

km.iris3$tot.withinss

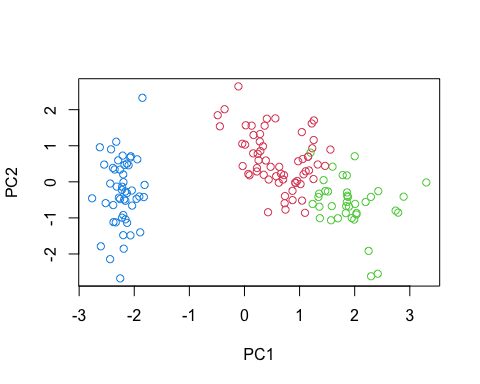
## [1] 78.85144

The output shows the total sum of squares obtained from clusters. The total sum of squares is “78.85144”.

Here, we can see that total sum of squares of 3 clusters is less than that of 2 clusters. This means that the observations are much more closer for 3 clusters. Thus due to small value of sum of squares, cluster 3 is better than cluster 2.

### 4. Draw a plot to visualize the clusters.

plot(pca.iris$x[,1:2], col = km.iris3$cluster + 1)



## Part (b): Hierarchical Clustering

### 1. Cluster the observations using complete linkage.

hh.complete <- hclust(dist(iris[,1:4]), method = "complete")  
hh.complete

##   
## Call:  
## hclust(d = dist(iris[, 1:4]), method = "complete")  
##   
## Cluster method : complete   
## Distance : euclidean   
## Number of objects: 150

### 2. Cluster the observations using average and single linkage.

Hierarchical clustering using average linkage

hh.average <- hclust(dist(iris[,1:4]), method = "average")  
hh.average

##   
## Call:  
## hclust(d = dist(iris[, 1:4]), method = "average")  
##   
## Cluster method : average   
## Distance : euclidean   
## Number of objects: 150

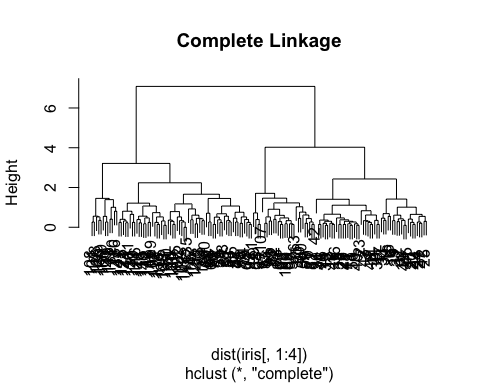
Hierarchical clustering using single linkage

hh.single <- hclust(dist(iris[,1:4]), method = "single")  
hh.single

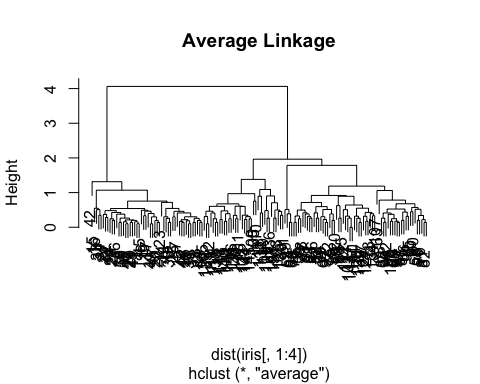
##   
## Call:  
## hclust(d = dist(iris[, 1:4]), method = "single")  
##   
## Cluster method : single   
## Distance : euclidean   
## Number of objects: 150

### 3. Plot the dendrogram for the above clustering methods.

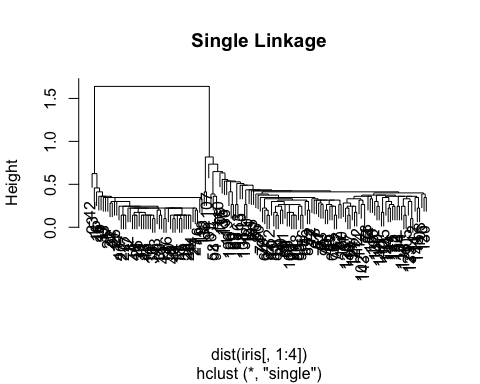
## Dendogram for complete linkage  
plot(hh.complete, main = "Complete Linkage")



## Dendogram for Average linkage  
plot(hh.average, main = "Average Linkage")



## Dendogram for Single linkage  
plot(hh.single, main = "Single Linkage")



## Question 2

### Generate the data set using the following codes

set.seed(2)  
x = matrix(rnorm(50\*2), ncol = 2)   
x[1:25, 1]= x[1:25, 1] + 3   
x[1:25, 2] = x[1:25, 2] - 4  
class(x)

## [1] "matrix" "array"

x <- as.data.frame(x)  
head(x)

## V1 V2  
## 1 2.103085 -4.838287  
## 2 3.184849 -1.933699  
## 3 4.587845 -4.562247  
## 4 1.869624 -2.724284  
## 5 2.919748 -5.047573  
## 6 3.132420 -5.965878

dim(x)

## [1] 50 2

Here we can see that we have 50 Observations of 2 variables. Both the variables are numeric in nature.

## Part (a): k-means clustering

### 1. Cluster the data set into 2 groups using k-means clustering.

km.mat <- kmeans(x, centers = 2, nstart = 20)  
km.mat

## K-means clustering with 2 clusters of sizes 25, 25  
##   
## Cluster means:  
## V1 V2  
## 1 3.3339737 -4.0761910  
## 2 -0.1956978 -0.1848774  
##   
## Clustering vector:  
## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [39] 2 2 2 2 2 2 2 2 2 2 2 2  
##   
## Within cluster sum of squares by cluster:  
## [1] 63.20595 65.40068  
## (between\_SS / total\_SS = 72.8 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

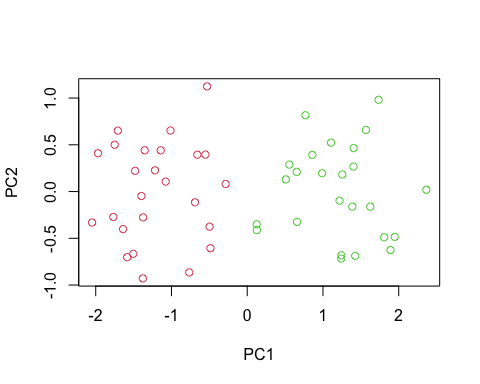
km.mat$tot.withinss

## [1] 128.6066

The output shows the total sum of squares obtained from clusters. The total sum of squares is “128.6066”

### 2. Draw a plot to visualize the clusters.

pca.mat <- prcomp(x, scale. = TRUE)  
plot(pca.mat$x[,1:2], col = km.mat$cluster + 1)



### 3. Cluster the data set into 3 groups using k-means clustering.

km.mat1 <- kmeans(x, centers = 3, nstart = 20)  
km.mat1

## K-means clustering with 3 clusters of sizes 23, 10, 17  
##   
## Cluster means:  
## V1 V2  
## 1 -0.3820397 -0.08740753  
## 2 2.3001545 -2.69622023  
## 3 3.7789567 -4.56200798  
##   
## Clustering vector:  
## [1] 3 2 3 2 3 3 3 2 3 2 3 2 3 2 3 2 3 3 3 3 3 2 3 3 3 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [39] 1 1 1 1 1 2 1 2 1 1 1 1  
##   
## Within cluster sum of squares by cluster:  
## [1] 52.67700 19.56137 25.74089  
## (between\_SS / total\_SS = 79.3 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

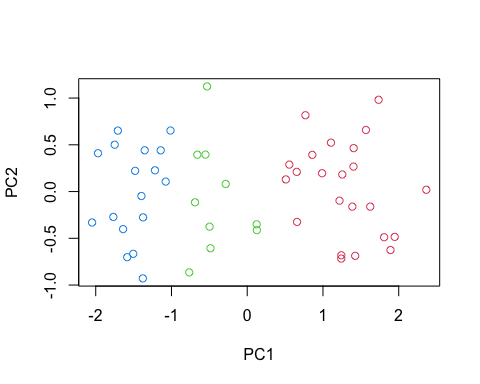
km.mat1$tot.withinss

## [1] 97.97927

The output shows the total sum of squares obtained from clusters. The total sum of squares is “97.97927”.

### 4. Draw a plot to visualize the clusters.

plot(pca.mat$x[,1:2], col = km.mat1$cluster + 1)



### 5. Compare the total within-cluster sum of squares when nstart = 1 and 20.

Comparing total within-cluster sum of squares when nstart = 1 and 20 for 2 group clusters

km.mat\_1 <- kmeans(x, centers = 2, nstart = 1)  
cat("Total within cluster sum of squares when nstart = 1 is", km.mat\_1$tot.withinss)

## Total within cluster sum of squares when nstart = 1 is 128.6066

cat("Total within cluster sum of squares when nstart = 20 is", km.mat$tot.withinss)

## Total within cluster sum of squares when nstart = 20 is 128.6066

Comparing total within-cluster sum of squares when nstart = 1 and 20 for 3 group clusters

km.mat1\_1 <- kmeans(x, centers = 3, nstart = 1)  
  
cat("Total within cluster sum of squares when nstart = 1 is", km.mat1\_1$tot.withinss)

## Total within cluster sum of squares when nstart = 1 is 98.16736

cat("Total within cluster sum of squares when nstart = 20 is", km.mat1$tot.withinss)

## Total within cluster sum of squares when nstart = 20 is 97.97927

## Part (b): Hierarchical Clustering

### 1. Cluster the observations using complete linkage.

Hierarchical clustering using complete linkage

hh.complete\_mat <- hclust(dist(x), method = "complete")  
hh.complete\_mat

##   
## Call:  
## hclust(d = dist(x), method = "complete")  
##   
## Cluster method : complete   
## Distance : euclidean   
## Number of objects: 50

### 2. Cluster the observations using average and single linkage.

Hierarchical clustering using average linkage

hh.average\_mat <- hclust(dist(x), method = "average")  
hh.average\_mat

##   
## Call:  
## hclust(d = dist(x), method = "average")  
##   
## Cluster method : average   
## Distance : euclidean   
## Number of objects: 50

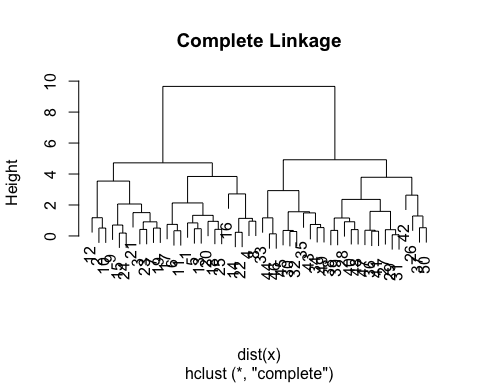
Hierarchical clustering using single linkage

hh.single\_mat <- hclust(dist(x), method = "single")  
hh.single\_mat

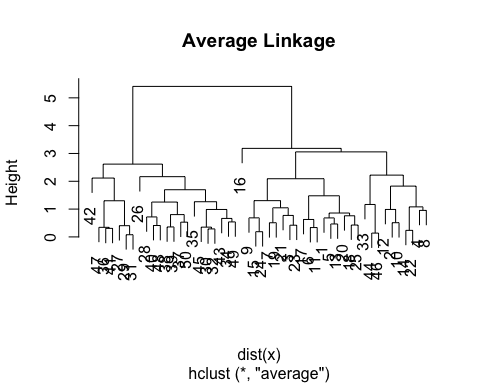
##   
## Call:  
## hclust(d = dist(x), method = "single")  
##   
## Cluster method : single   
## Distance : euclidean   
## Number of objects: 50

### 3. Plot the dendrogram for the above clustering methods.

## Dendogram for complete linkage  
plot(hh.complete\_mat, main = "Complete Linkage")



## Dendogram for Average linkage  
plot(hh.average\_mat, main = "Average Linkage")



## Dendogram for Single linkage  
plot(hh.single\_mat, main = "Single Linkage")

