Week 9 Lab Solutions

K-Means Clustering

Dataset 1 for K-means Clustering

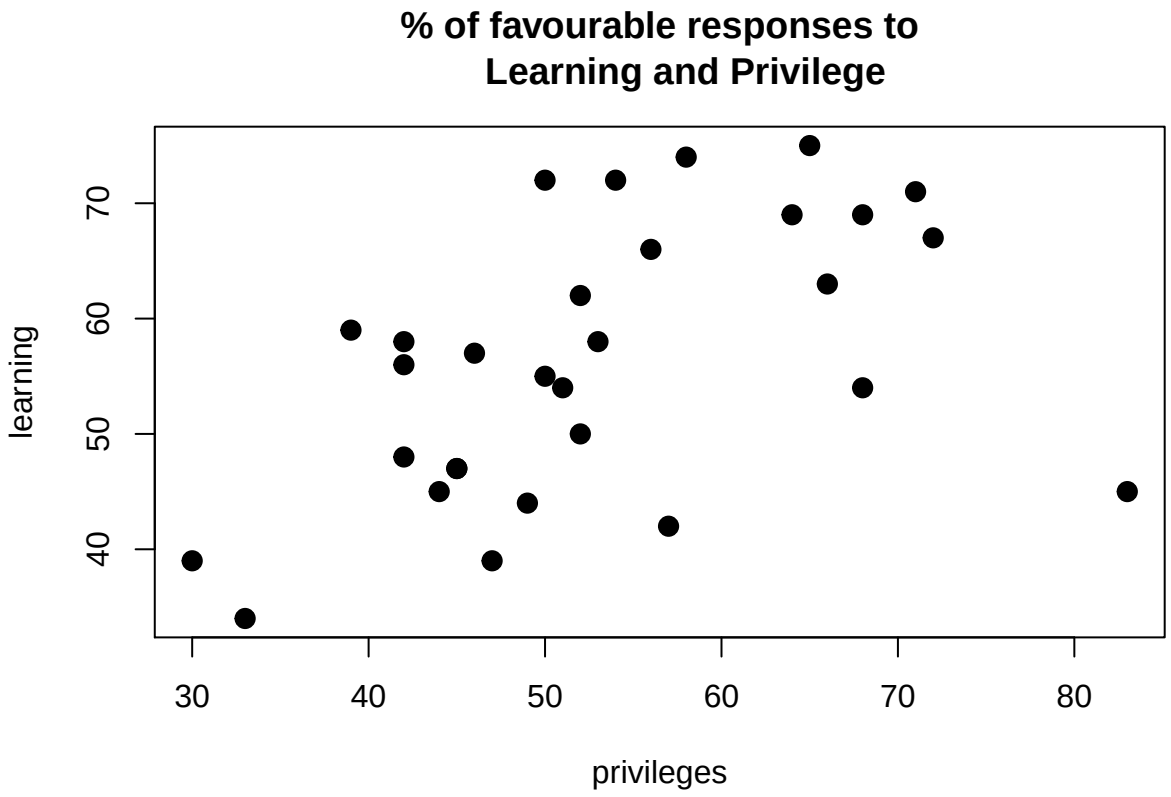
Chatterjee-Price Attitude Data attitude from the datasets package. The dataset is a survey of clerical employees of a large financial organization. The data are aggregated from questionnaires of approximately 35 employees for each of 30 (randomly selected) departments. The numbers give the percent proportion of favourable responses to seven questions in each department. For more details, see ?attitude.

In this exercise, we'll take a subset of the attitude dataset and consider only two variables privileges and learning, that is we would like to cluster the attitude dataset with the responses from all 30 departments when it comes to privileges and learning. The subset is defined as follows:

```
library(datasets)
dat <- attitude[,c(3,4)]
```

1) Plot the dataset dat.

```
plot(dat, main = "% of favourable responses to
     Learning and Privilege", pch =20, cex =2)
```



2) Let k = 2 and nstart = 1. Set a seed and then perform the k-means clustering based on the two parameters.

```
#set.seed(7)
km.out.1 <- kmeans(dat,2,nstart=1)
```

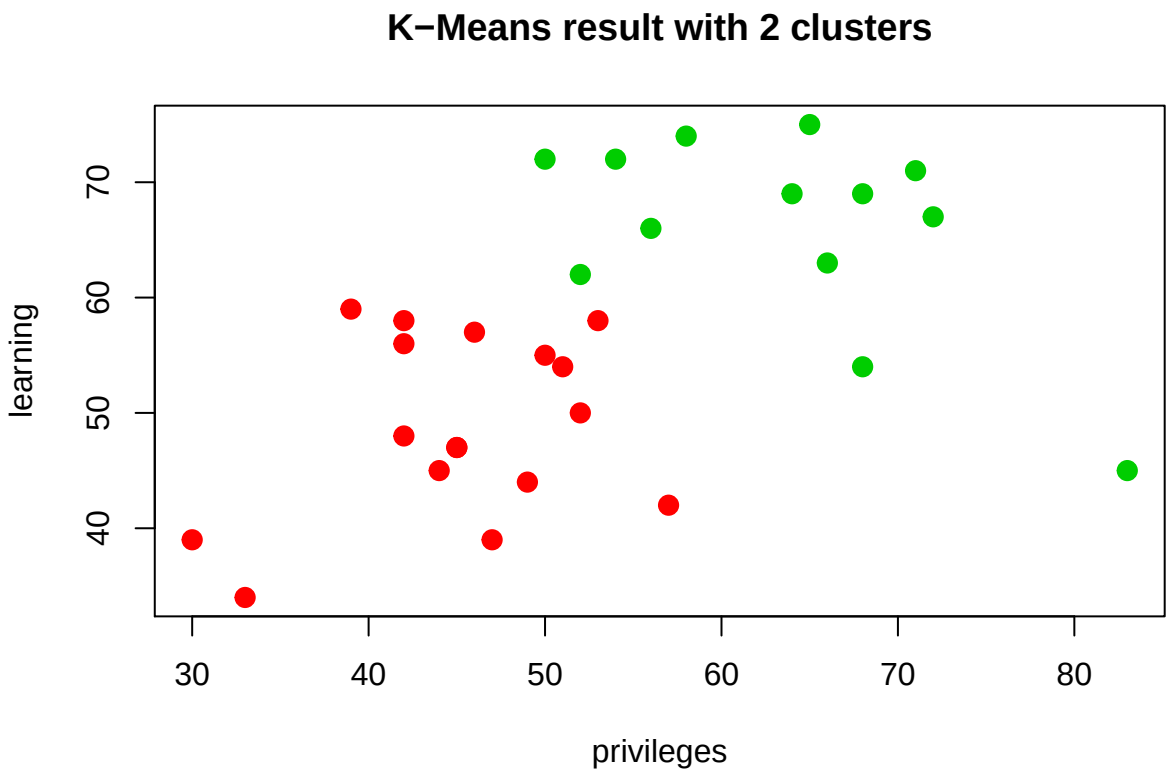
3) Report the tot.withinss.

```
km.out.1$tot.withinss
```

```
## [1] 3652.706
```

4) Plot the two clusters with two different colours.

```
plot(dat, col =(km.out.1$cluster +1), main="K-Means result with 2 clusters", pch=20, cex=2)
```

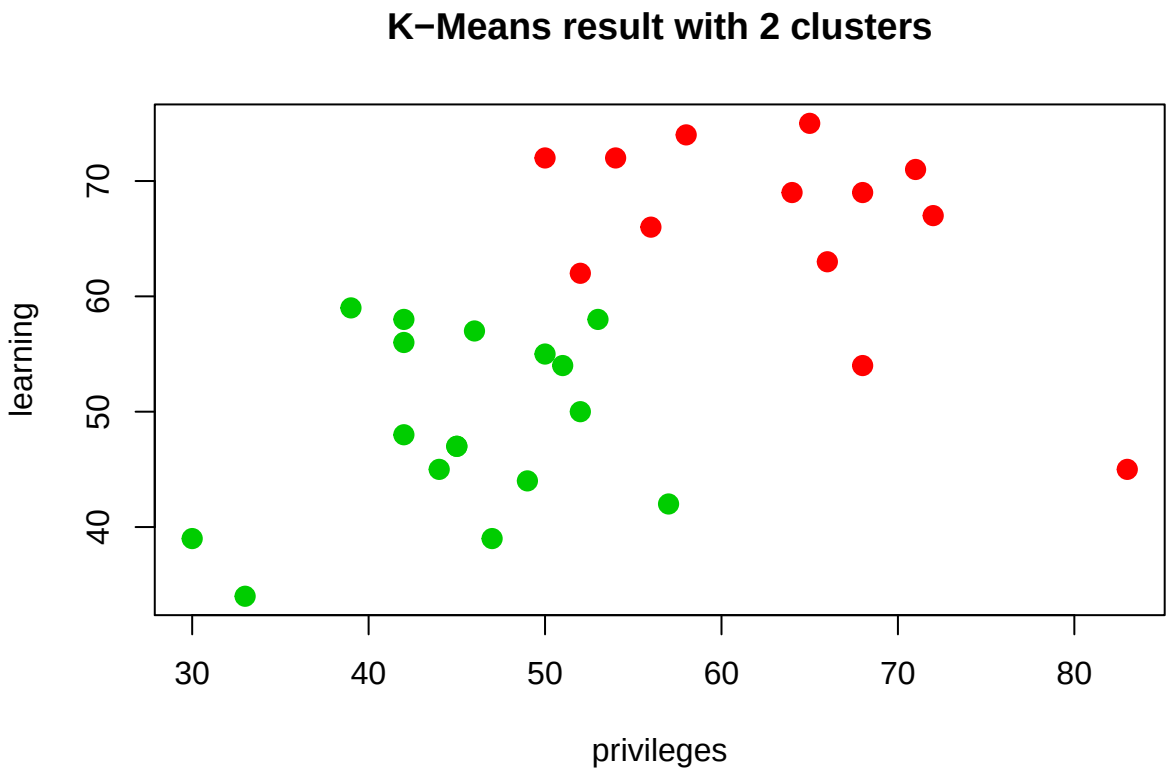


5) Let nstart = 100 and repeat 2)-4). Compare the two tot.withinss.

```
#set.seed(7)
km.out.100 <- kmeans(dat,2,nstart=100)
km.out.100$tot.withinss
```

```
## [1] 3652.706
```

```
plot(dat, col = (km.out.100$cluster +1),
     main="K-Means result with 2 clusters", pch=20, cex=2)
```



The two models have the same tot.withinss.

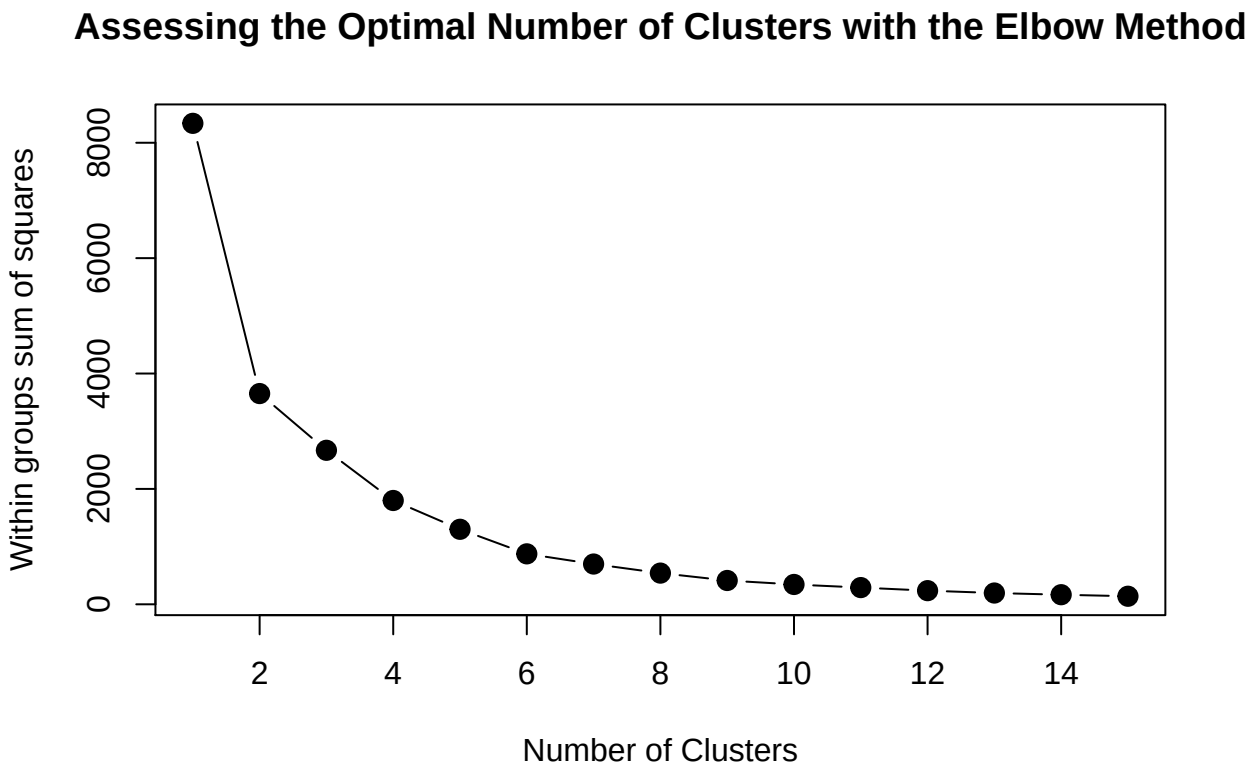
6) Write a for-loop to record the tot.withinss when k is 1 to 15. Plot the result.

```
totWithinSS <- rep(0,15)
for(i in 1:15){
  set.seed(70)
  totWithinSS[i] <- kmeans(dat,i,nstart=100)$tot.withinss
}
```

7) Use Elbow method to identify the best k.

With the elbow method, the solution criterion value (within groups sum of squares) will tend to decrease substantially with each successive increase in the number of clusters. From the plot, the decrease from k=6 started to slow down. Hence, the optimal number is k=6. However, it is quite a subjective decision, and should take the question/dataset into consideration too.

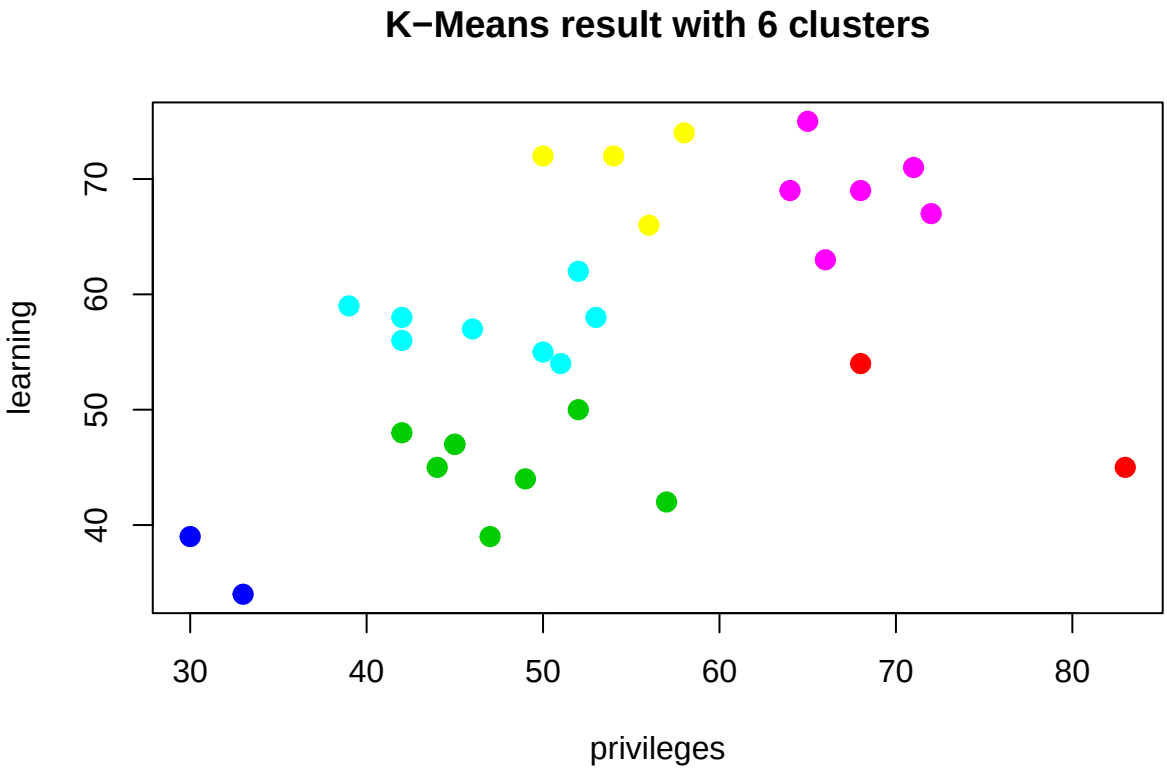
```
plot(1:15, totWithinSS, type="b", xlab="Number of Clusters",
     ylab="Within groups sum of squares",
     main="Assessing the Optimal Number of Clusters with the Elbow Method",
     pch=20, cex=2)
```



8) Plot the k clusters with the best k you get in 7).

k=6

```
set.seed(70)
km.out.k6 <- kmeans(dat,6,nstart=100)
plot(dat, col = (km.out.k6$cluster +1),
     main="K-Means result with 6 clusters", pch=20, cex=2)
```



Hierarchical Clustering

Dataset 2 for Hierarchical Clustering

On the book website, http://www-bcf.usc.edu/~gareth/ISL/data.html, there is a gene expression data set (Ch10Ex11.csv) that consists of 40 tissue samples (40 columns) with measurements on 1,000 genes (1000 rows). The first 20 samples are from healthy patients, while the second 20 are from a diseased group.

Please download the dataset from Moodle and load in the Ch10Ex11.csv file.

```
DF <- read.csv(".../Ch10Ex11.csv", header=FALSE)
```

9) Read the description of the dataset again. Do you think the current layout of the dataset is a natural way to present the relationship between tissue samples (as columns) and genes (as rows)? Note each tissue may contain hundreds of genes. If not, transform the dataset in a more natural way.

```
DF <- t(DF)
#Transpose the matrix such that the rows are for tissue samples (from different people) and the columns
```

10) Calculate the dissimilarity metric.

Hint: We will take as our dissimilarity metric between the ith and jth samples to be 1 - rij, where rij is the correlation between the two samples. Notice that this function will have its smallest value (of zero) if rij = 1 i.e. the two samples are perfectly correlated. This function will have its largest value (of two) if rij = -1 i.e. the two samples are perfectly anti-correlated.

```r
D <-  as.dist( 1 - cor(t(DF)) )
# cor computes the correlation of *columns* so we need to take the transpose of DF
```

11) Apply hierarchical clustering to the samples using correlation based distance for

a. Complete linkage

```r
hclust.cor.comp <- hclust(D, method="complete")
```

b. Average linkage

```r
hclust.cor.ave <- hclust(D, method="average")
```

c. Single linkage

```r
hclust.cor.sing <- hclust(D, method="single")
```

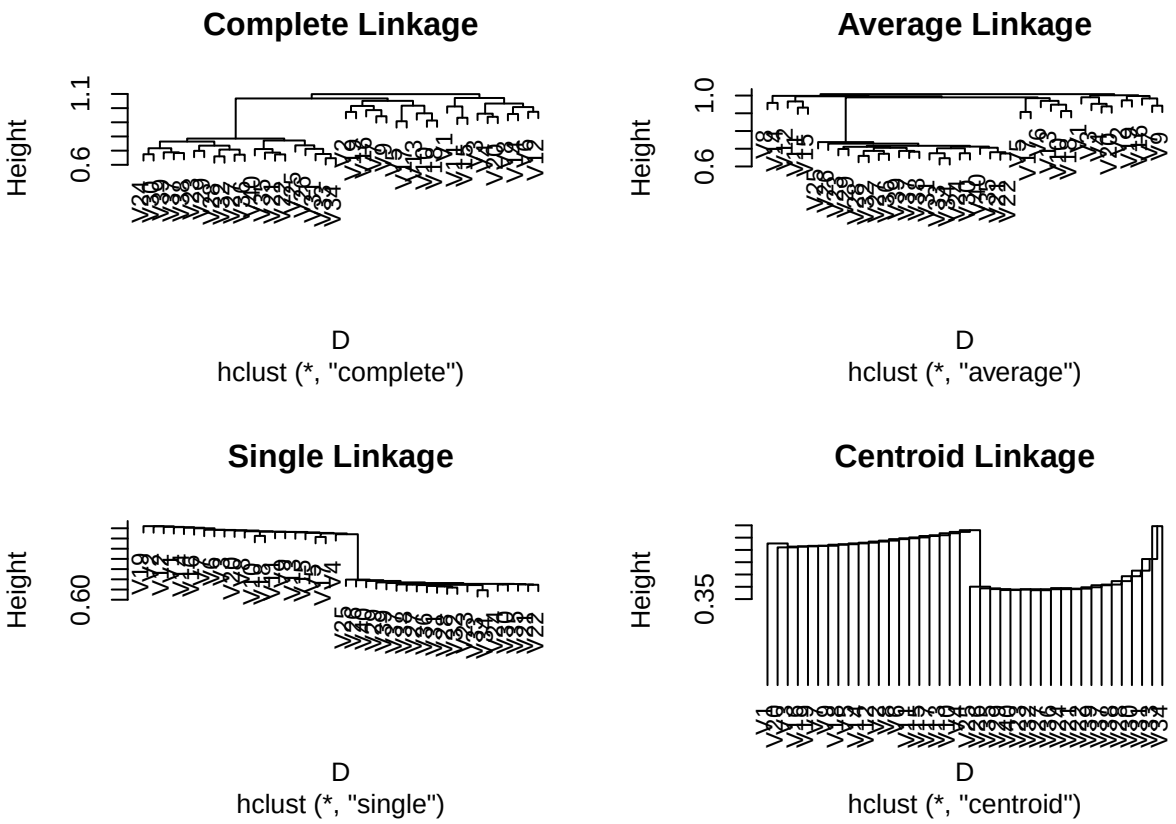
d. Centroid linkage

```r
hclust.cor.cent <- hclust(D, method="centroid")
```

12) Plot the four dendrograms in the same plot by using `par(mfrow=c(i,j))`, where i is the number of rows and j is the number of columns in the plot.

```r
par(mfrow = c(2,2))

plot(hclust.cor.comp, main="Complete Linkage", cex=0.9)
plot(hclust.cor.ave, main="Average Linkage", cex=0.9)
plot(hclust.cor.sing, main="Single Linkage", cex=0.9)
plot(hclust.cor.cent, main="Centroid Linkage", cex=0.9)
```



13) Do the genes separate the samples into the two groups? To answer this question, we need to generate a confusion matrix on the predicted and true number of healthy/diseased patients.

```
# How well does our clustering predict health vs. diseased:
# the first 20 are healthy (=0), last 20 are diseased (=1)

###complete linkage
print(table(predicted = cutree(hclust.cor.comp, k = 2), truth=c(rep(0,20), rep(1,20))))
```

```
##           truth
## predicted  0  1
##         1 10  0
##         2 10 20
```

```
#repeat it for average and single linkage
print(table(predicted = cutree(hclust.cor.ave, k = 2), truth=c(rep(0,20), rep(1,20))))
```

```
##           truth
## predicted  0  1
##         1  9  0
##         2 11 20
```

```
print(table(predicted = cutree(hclust.cor.sing, k = 2), truth=c(rep(0,20), rep(1,20))))
```

```
##           truth
## predicted  0  1
##         1 19 20
##         2  1  0
```

```
print(table(predicted = cutree(hclust.cor.cent, k = 2), truth=c(rep(0,20), rep(1,20))))
```

```
##           truth
## predicted  0  1
##         1  1  0
##         2 19 20
```

If we compare the predicted cluster label to the known truth label where we take 0 to be a healthy patient and 1 to be a diseased patient we get

```
##           truth
## predicted  0  1
##         1 10  0
##         2 10 20
```

or

```
##           truth
## predicted  0  1
##         1  9  0
##         2 11 20
```

It looks like our predicted class label of `1` corresponds to 10 healthy patients, while the predicted class of `2` corresponds to 10 healthy patients and 20 diseased patients.

14) Do your results depend on the type of linkage used?

These results do depend on the linkage used as is typically the case.