STAT 4051 HW2

Ruwiada Al Harrasi

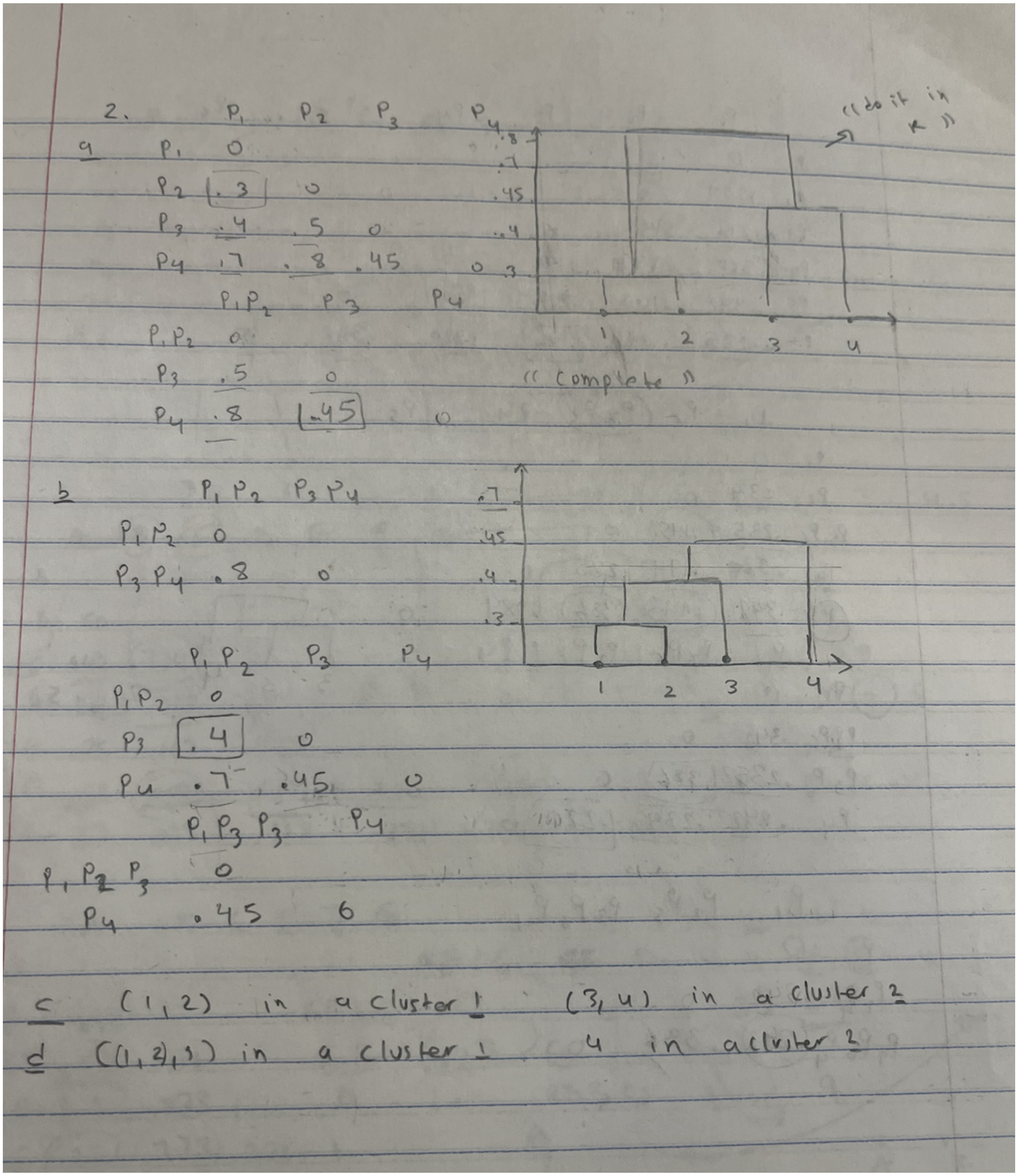
2023-10-06

```
is_already_installed <- require(imager)
```

```
## Loading required package: imager
```

```
## Warning in library(package, lib.loc = lib.loc, character.only = TRUE,
## logical.return = TRUE, : there is no package called 'imager'
```

```
knitr::include_graphics("~/Desktop/hw2.png")
```


.8
.7
.45
.4
.3
1
2
3
4
.7
.45
.4
.3
1
2
3
4

2.

a

	P1	P2	P3	P4
P1	0			
P2	$\boxed{.3}$	0		
P3	.4	.5	0	
P4	.7	.8	.45	0

	P1P2	P3	P4
P1P2	0		
P3	.5	0	
P4	.8	$\boxed{.45}$	0

« do it in R »

« complete »

b

	P1 P2	P3 P4
P1P2	0	
P3 P4	.8	0

	P1 P2	P3	P4
P1P2	0		
P3	$\boxed{.4}$	0	
Pu	.7	.45	0

	P1 P3 P3	P4
P1 P2 P3	0	
P4	.45	6

c (1, 2) in a cluster 1 (3, 4) in a cluster 2

d ((1, 2), 3) in a cluster 1 4 in a cluster 2

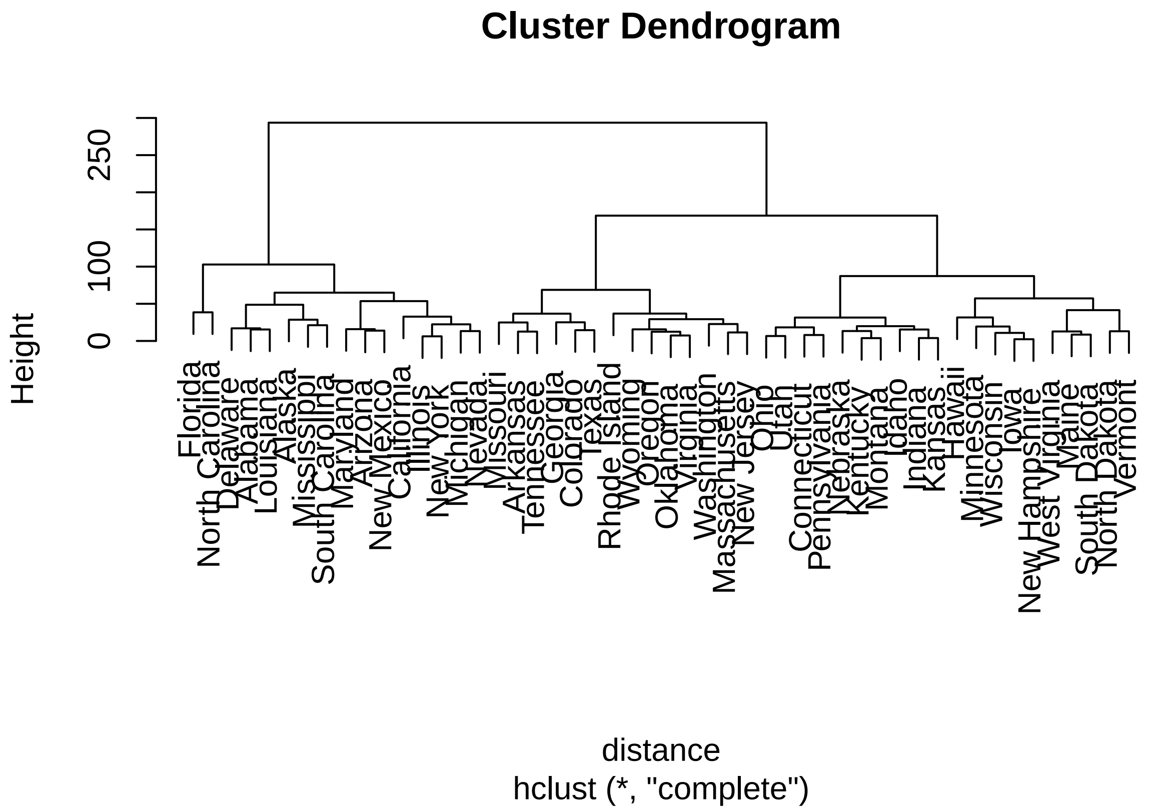
2. (10 pts) Consider the USArrests data, directly available in R. You can directly call summary(USArrests) in R. It is a data set containing the arrests per 100,000 residents for assault, murder, and rape in each of the 50 US states in 1973. We will now perform hierarchical clustering on the states.

(a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states.

```
summary(USArrests)
```

```
##      Murder          Assault         UrbanPop          Rape      
##  Min.   : 0.800   Min.   : 45.0   Min.   :32.00   Min.   : 7.30  
##  1st Qu.: 4.075   1st Qu.:109.0   1st Qu.:54.50   1st Qu.:15.07  
##  Median : 7.250   Median :159.0   Median :66.00   Median :20.10  
##  Mean   : 7.788   Mean   :170.8   Mean   :65.54   Mean   :21.23  
##  3rd Qu.:11.250   3rd Qu.:249.0   3rd Qu.:77.75   3rd Qu.:26.18  
##  Max.   :17.400   Max.   :337.0   Max.   :91.00   Max.   :46.00  
```

```
## dist() calculates the distance of every pair in the dataset and return it as matrix
distance <- dist(USArrests, method='euclidean')
## hclust() doesn't require the dataset since all the information
## of similarity is included in the distance matrix
hc_complete <- hclust(distance, method='complete') ## complete linkage
plot(hc_complete)
```



(b) Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters?

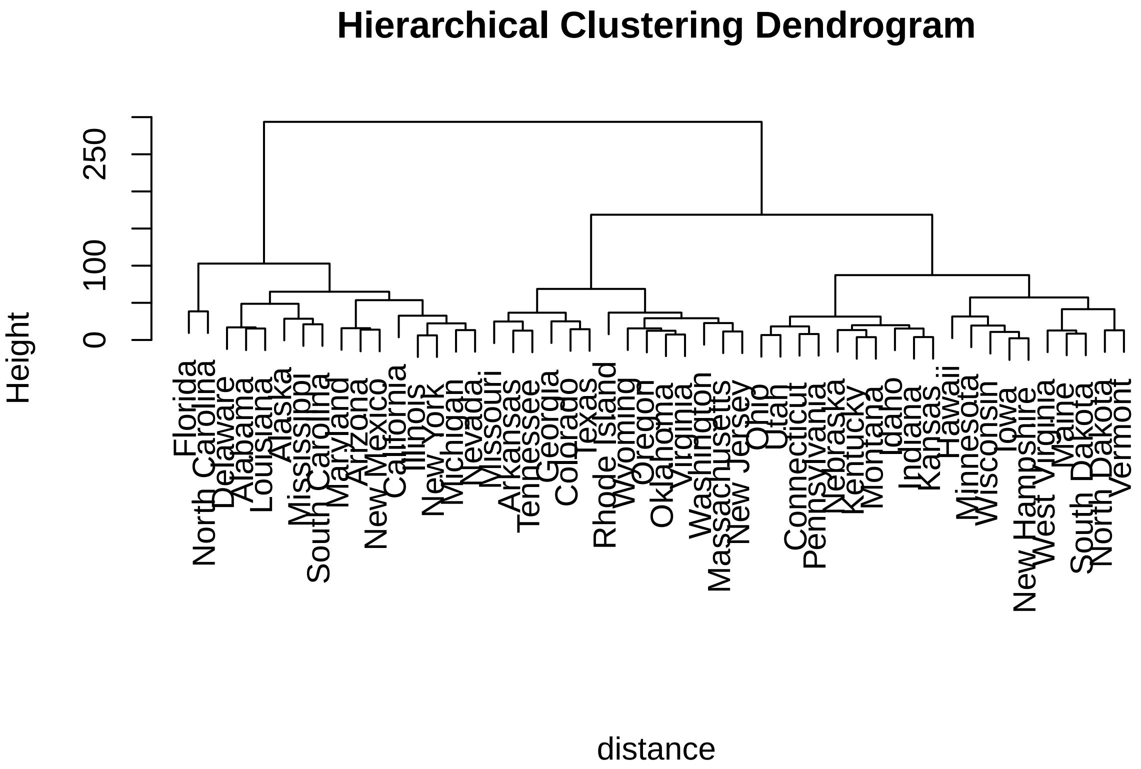
```
clsuters <- cutree(tree=hc_complete, k=3)
table(clsuters)
```

```
## clsuters
##  1  2  3 
## 16 14 20
```

```
clsuters
```

```
##        Alabama         Alaska        Arizona       Arkansas     California 
##              1              1              1              2              1 
##       Colorado    Connecticut       Delaware        Florida        Georgia 
##              2              3              1              1              2 
##         Hawaii          Idaho       Illinois        Indiana           Iowa 
##              3              3              1              3              3 
##         Kansas       Kentucky      Louisiana          Maine       Maryland 
##              3              3              1              3              1 
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri 
##              2              1              3              1              2 
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey 
##              3              3              1              3              2 
##     New Mexico       New York North Carolina   North Dakota           Ohio 
##              1              1              1              3              3 
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina 
##              2              2              3              2              1 
##   South Dakota      Tennessee          Texas           Utah        Vermont 
##              3              2              2              3              3 
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming 
##              2              2              3              3              2
```

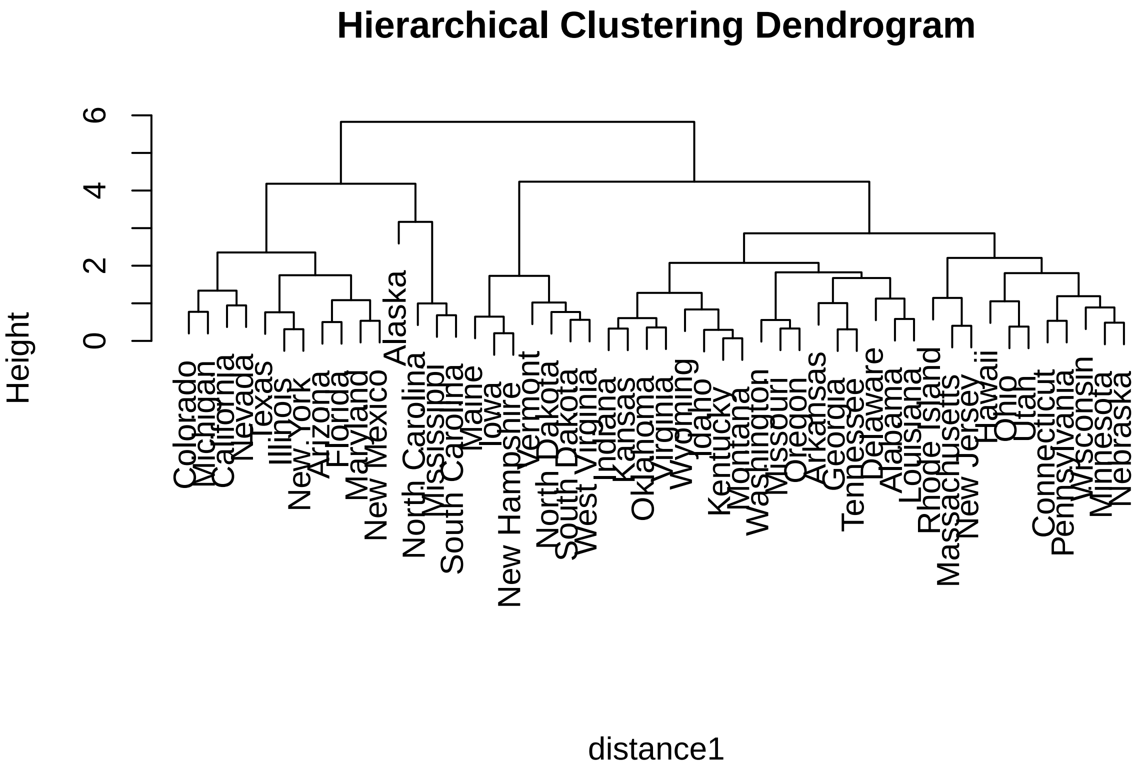
```
# Plot the hierarchical clustering dendrogram
plot(hc_complete, main = "Hierarchical Clustering Dendrogram", sub = "")
```



Based on the head of result which print the state with the cluster that it is assigned to i can see that Alabama, Alaska, Arizona,and California are in cluster 1. However, Arkansas and Colorado are in cluster 2.

(c) Hierarchically cluster the states using complete linkage and Euclidean distance, after scaling the variables to have standard deviation one. Now which states belong to which clusters?

```
scaled=scale(USArrests[, -1])
## dist() calculates the distance of every pair in the dataset and return it as matrix
distance1 <- dist(scaled, method='euclidean')
## hclust() doesn't require the dataset since all the information
## of similarity is included in the distance matrix
hc_complete1 <- hclust(distance1, method='complete') ## complete linkage
clsuter1 <- cutree(tree=hc_complete1, k=3)
# Plot the hierarchical clustering dendrogram
plot(hc_complete1, main = "Hierarchical Clustering Dendrogram", sub = "")
```



```
# Display which states belong to each cluster
# Create a data frame with Cluster and State columns
result1 <- data.frame(clsuter1, State = rownames(USArrests))
result1
```

```
##                clsuter1          State
## Alabama               1        Alabama
## Alaska                2         Alaska
## Arizona               2        Arizona
## Arkansas              1       Arkansas
## California            2     California
## Colorado              2       Colorado
## Connecticut           1    Connecticut
## Delaware              1       Delaware
## Florida               2        Florida
## Georgia               1        Georgia
## Hawaii                1         Hawaii
## Idaho                 1          Idaho
## Illinois              2       Illinois
## Indiana               1        Indiana
## Iowa                  3           Iowa
## Kansas                1         Kansas
## Kentucky              1       Kentucky
## Louisiana             1      Louisiana
## Maine                 3          Maine
## Maryland              2       Maryland
## Massachusetts         1  Massachusetts
## Michigan              2       Michigan
## Minnesota             1      Minnesota
## Mississippi           2    Mississippi
## Missouri              1       Missouri
## Montana               1        Montana
## Nebraska              1       Nebraska
## Nevada                2         Nevada
## New Hampshire         3  New Hampshire
## New Jersey            1     New Jersey
## New Mexico            2     New Mexico
## New York              2       New York
## North Carolina        2 North Carolina
## North Dakota          3   North Dakota
## Ohio                  1           Ohio
## Oklahoma              1       Oklahoma
## Oregon                1         Oregon
## Pennsylvania          1   Pennsylvania
## Rhode Island          1   Rhode Island
## South Carolina        2 South Carolina
## South Dakota          3   South Dakota
## Tennessee             1      Tennessee
## Texas                 2          Texas
## Utah                  1           Utah
## Vermont               3        Vermont
## Virginia              1       Virginia
## Washington            1     Washington
## West Virginia         3  West Virginia
## Wisconsin             1      Wisconsin
## Wyoming               1        Wyoming
```

```
# Display the result with Cluster column first
print(head(result1))
```

```
##            clsuter1      State
## Alabama           1    Alabama
## Alaska            2     Alaska
## Arizona           2    Arizona
## Arkansas          1   Arkansas
## California        2 California
## Colorado          2   Colorado
```

After scaling the variables to have sd one I noticed that some states have changed their cluster labels for example Alaska, Arizona,and California were in cluster 1 before the scaling now they are in cluster 2. And Arkansas moved from cluster 2 to 1. However, Alabama and Colorado are still in the same cluster before the scaling.

(d) What effect does scaling the variables have on the hierarchical clustering obtained? In your opinion, should the variables be scaled before the inter-observation dis- similarities are computed? Provide a justification for your answer.

In these data, the variables are measured in different units; Murder, Rape, and Assault are reported as the number of occurrences per 100, 000 people, and UrbanPop is the percentage of the state's population that lives in an urban area. These four variables have variances of 18.97, 87.73, 6945.16, and 209.5, respec- tively. Consequently, if we perform PCA on the unscaled variables, then the first principal component loading vector will have a very large loading for Assault. Scaling variables before computing inter-observation dissimilarities (such as Euclidean distances) in hierarchical clustering is generally a good practice, especially when dealing with variables of different units or scales. Scaling helps ensure that the clustering process is not biased by the magnitude or units of the variables.

3

```r
library(kernlab)
```

```r
X<- read.csv("~/Downloads/CircleClusters.csv", header=T)
```

K-means

```r
# Perform K-means clustering for 2 clusters
kmeans <- kmeans(X, centers = 2, nstart=20)
```

complete

```r
distance <- dist(X, method='euclidean')
hc_complete <- hclust(distance, method='complete') ## complete linkage
hc_complete <- cutree(tree=hc_complete, k=2)
```

single

```r
hc_single <- hclust(distance, method='single')
hc_single <- cutree(tree=hc_single , k=2)
```

average

```r
hc_average <- hclust(distance, method='average')
hc_average <- cutree(tree=hc_average , k=2)
```

spectral clustering

```r
cluster_specc <- specc(x=as.matrix(X), centers=2, kernel='rbfdot')
```

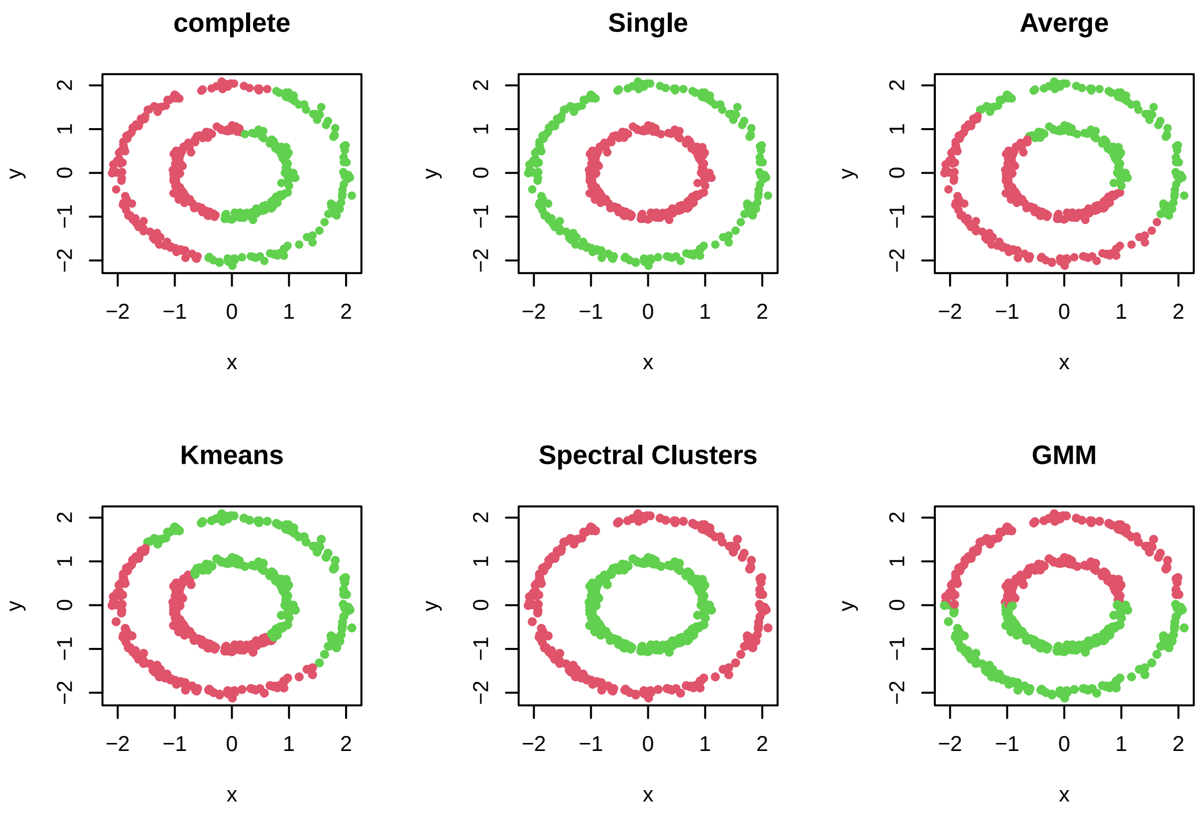
GMM

```r
library(mclust)
```

```
## Package 'mclust' version 6.0.0
## Type 'citation("mclust")' for citing this R package in publications.
```

```r
gmm= Mclust(X, G = 2)
gmm_c= gmm$classification
```

```
par(mfrow=c(2,3))
plot(X, col=hc_complete+1, pch=20, main= "complete")
plot(X, col=hc_single+1, pch=20,main= "Single")
plot(X, col=hc_average+1, pch=20, main="Averge")
## plot the Kmeans clusters
plot(X, col=kmeans$cluster+1, cex=1.1, pch=20, main="Kmeans")
## plot the spectral clusters
plot(X, col=cluster_specc+1, cex=1.1, pch=20, main="Spectral Clusters")
## GMM
plot(X, col=gmm_c+1, cex=1.1, pch=20, main="GMM")
```


complete

Single

Averge

Kmeans

Spectral Clusters

GMM

From my observation complete linkage, average linkage, kmeans, GMM split the two clusters in half each side is a cluster. However single linkage and spectral clustering it represnted the two clusters as circles.this makes sense because points in each spiral have closer distance while points in different spirals have greater distance single linkage and spectral cluster are best fit for this data as they had better clustring .

4.Apply hierarchical clustering (selecting your preferred distance and linkage) and spectral clustering to the data, clustering into 2, 4, and 6 groups, and the GMM with automatic BIC model selection. How do the results compare to Kmeans from HW1?

```
data<- read.csv("~/Downloads/CancerData-Small.csv", header=T)
X1mean <- mean(data$X1, na.rm = TRUE)
data$X1[is.na(data$X1)]= X1mean
# Extract the three medical measures (X1, X2, X3)
X1 <- data[, c("X1", "X2", "X3")]
X1
```

```
##           X1    X2     X3
## 1  1.5790000 0.014 39.477
## 2  0.7160000 0.000  0.000
## 3  0.5590476 0.000  3.939
## 4  0.9710000 0.013 16.569
## 5  1.4930000 0.075 19.133
## 6  1.9550000 0.040  5.865
## 7  0.1930000 0.006  1.485
## 8  0.0700000 0.003  3.387
## 9  1.8980000 0.011 32.821
## 10 0.1000000 0.005 23.624
## 11 0.0000000 0.000  0.000
## 12 0.1980000 0.007  0.000
## 13 0.0410000 0.008  5.401
## 14 0.0000000 0.000  7.336
## 15 0.0000000 0.000 10.365
## 16 0.0000000 0.000 24.345
## 17 0.0140000 0.000 80.374
## 18 1.5380000 0.020 32.710
## 19 0.0000000 0.035 76.320
## 20 0.0270000 0.000 11.858
## 21 0.9300000 0.000 87.012
## 22 0.0170000 0.023 58.547
```

hierarchical complete linkage with Euclidean distance 2 ,4 ,6

```
X1= as.matrix(X1)
cancer_labels=data$Cancer
distance1 <- dist(X1, method='euclidean')
hc_complete1 <- hclust(distance1, method='complete')
hc_complete2 <- cutree(tree=hc_complete1, k=2)
hc_complete4 <- cutree(tree=hc_complete1, k=4)
hc_complete6 <- cutree(tree=hc_complete1, k=6)
cross_table2 <- table(hc_complete2, cancer_labels)
cross_table2
```

```
##             cancer_labels
## hc_complete2  1  2
##            1  7 11
##            2  3  1
```

```
cross_table4 <- table(hc_complete4, cancer_labels)
cross_table4
```

```
##             cancer_labels
## hc_complete4  1  2
##            1  6  1
##            2  1 10
##            3  3  0
##            4  0  1
```

```
cross_table6 <- table(hc_complete6, cancer_labels)
cross_table6
```

```
##             cancer_labels
## hc_complete6 1 2
##            1 3 0
##            2 1 8
##            3 3 1
##            4 0 2
##            5 3 0
##            6 0 1
```

When applying complete linkage hierarchical clustering with 4 clusters, most data points are closely associated with their assigned clusters, demonstrating clear separation. However, with 6 or 2 clusters, the patterns become less distinct, and data points appear less consistently grouped. so it is similler to kmeans.

spectral clustering to the data, clustering into 2, 4, and 6

When applying spectral clustering with 4 clusters, most data points closely align with their assigned clusters, showing clear separation. However, with 6 or 2 clusters, the patterns become less distinct, and data points appear less consistently grouped, resembling the behavior of K-means clustering. In the spectral clustering analysis, varying the number of clusters (2, 4, and 6) results in different assignments of data points from two cancer labels (1 and 2) to these clusters.When using 2 clusters, the majority of data points are assigned to just two clusters, with one cluster having more data points from cancer label 2, and the other having a mix of data points from both labels.When using 4 clusters, the data points are spread across four clusters, with some clusters predominantly containing data points from one label, and others having a mix of both labels. When using 6 clusters, the data points are divided into six clusters, again with varying proportions of data points from the two cancer labels in each cluster.

GMM to the data, clustering into 2, 4, and 6

```
library(mclust)
gmm1= Mclust(X1)
gmm_c1= gmm1$classification

cross_table11 <- table(gmm_c1, cancer_labels)
cross_table11
```

```
##       cancer_labels
## gmm_c1 1 2
##      1 3 0
##      2 1 2
##      3 0 2
##      4 3 7
##      5 2 0
##      6 1 1
```

If we automatically assigned the Bayesian Information Criterion (BIC) to the model, it becomes evident that only one cluster predominantly contains data points from cancer labe. However, in the remaining clusters, there is a mixture of data points from both cancer labels, and no clear discernible pattern emerges. These results suggest that the GMM clustering algorithm, guided by BIC, has separated the data into clusters, each exhibiting different compositions of cancer labels. This indicates potential variations in characteristics or patterns associated with cancer labels across the clusters. Think it is similler to kmeans.