DPA Assignment 5

Girish Rajani

2023-04-30

#Recitation Problems

#Chapter 12

#1a Prove 12.18

$$\frac{1}{|C_k|}\sum_{i,i'\in C_k}\sum_{j=1}^{p}\left(x_{ij}-x_{i'j}\right)^2 = 2\sum_{i\in C_k}\sum_{j=1}^{p}\left(x_{ij}-\bar{x}_{kj}\right)^2$$

$$= \frac{1}{|C_k|}\sum_{i,i'\in C_k}\sum_{j=1}^{p}\left(\left(x_{ij}-\bar{x}_{kj}\right)-\left(x_{i'j}-\bar{x}_{kj}\right)\right)^2$$

$$= \frac{1}{|C_k|}\sum_{i,i'\in C_k}\sum_{j=1}^{p}\left(\left(x_{ij}-\bar{x}_{kj}\right)^2-2\left(x_{ij}-\bar{x}_{kj}\right)\left(x_{i'j}-\bar{x}_{kj}\right)+\left(x_{i'j}-\bar{x}_{kj}\right)^2\right)$$

$$= \frac{|C_k|}{|C_k|}\sum_{i\in C_k}\sum_{j=1}^{p}\left(x_{ij}-\bar{x}_{kj}\right)^2+\frac{|C_k|}{|C_k|}\sum_{i'\in C_k}\sum_{j=1}^{p}\left(x_{i'j}-\bar{x}_{kj}\right)^2-\frac{2}{|C_k|}\sum_{i,i'\in C_k}\sum_{j=1}^{p}\left(x_{ij}-\bar{x}_{kj}\right)\left(x_{i'j}-\bar{x}_{kj}\right)$$

$$= 2\sum_{i\in C_k}\sum_{j=1}^{p}\left(x_{ij}-\bar{x}_{kj}\right)^2+0$$

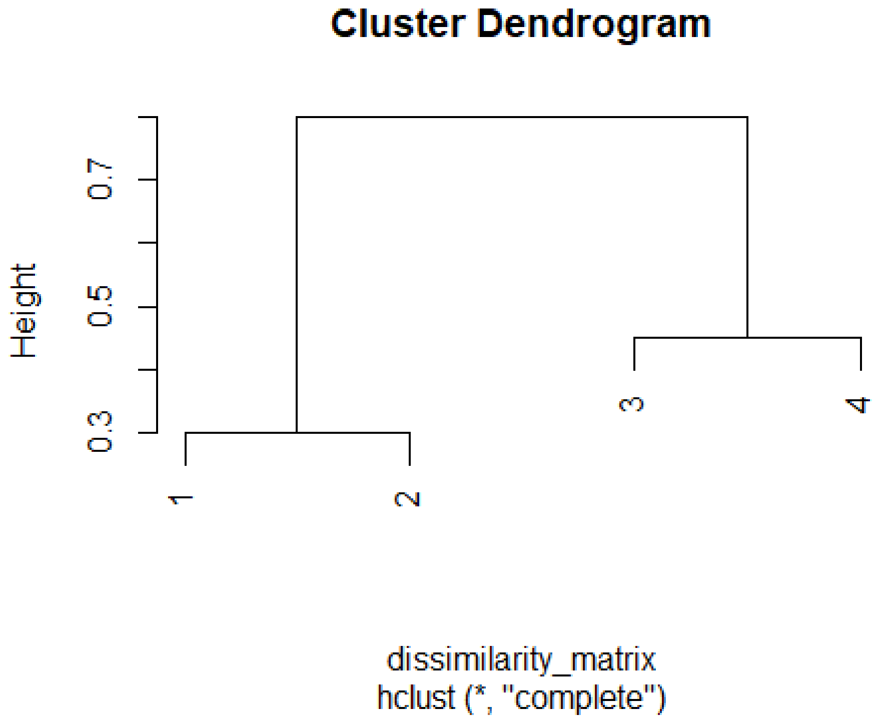
#1b

At each iteration, since each observation is assigned to the closest centroid based on the euclidean distance dissimilarity measure. This minimizes the sum of squared Euclidean distance as proven above. This is equivalent to minimizing the within-cluster variation for each cluster. This is guaranteed to decrease the value of the objective 12.17.

#2a

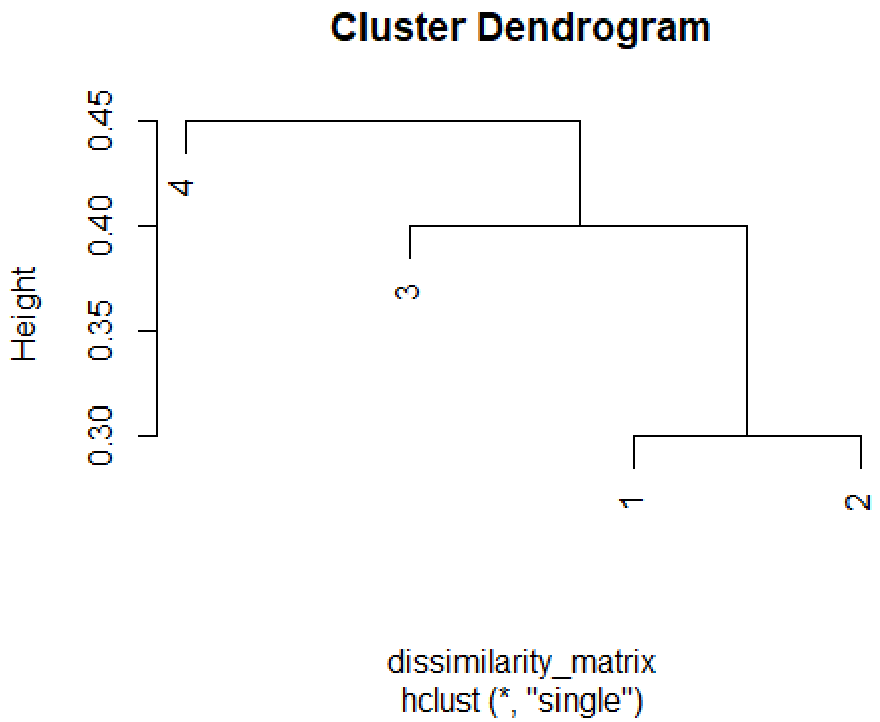
```r
#create the dissimilarity matrix
dissimilarity_matrix <- as.dist(matrix(c(
                     0, 0.3, 0.4, 0.7,
                     0.3, 0, 0.5, 0.8,
                     0.4, 0.5, 0.0, 0.45,
                     0.7, 0.8, 0.45, 0.0), nrow = 4))

#hierarchically clustering using complete linkage
complete_linkage_clustering <- hclust(dissimilarity_matrix, method =
"complete")

#Heights at which each fusion occurs
cat("Heights at which each fusion occurs for complete
linkage:",complete_linkage_clustering$height)
```

```
## Heights at which each fusion occurs for complete linkage: 0.3 0.45 0.8

#sketch the dendrogram
plot(complete_linkage_clustering)
```



#2b

```
#hierarchically clustering using single linkage
single_linkage_clustering <- hclust(dissimilarity_matrix, method = "single")

#Heights at which each fusion occurs
cat("Heights at which each fusion occurs for single
linkage:",single_linkage_clustering$height)

## Heights at which each fusion occurs for single linkage: 0.3 0.4 0.45

#sketch the dendrogram
plot(single_linkage_clustering)
```



#2c

```
#Cut the tree that resulted from hclust into several groups by specifying 2 clusters (k=2)
complete_linkage_cut <- cutree(complete_linkage_clustering, k = 2)

#returns a vector with group memberships
complete_linkage_cut

## [1] 1 1 2 2
```

We can see from above that based on the returned group memberships from the vector, the observations in each cluster from (a) are: (1,2), (3,4)

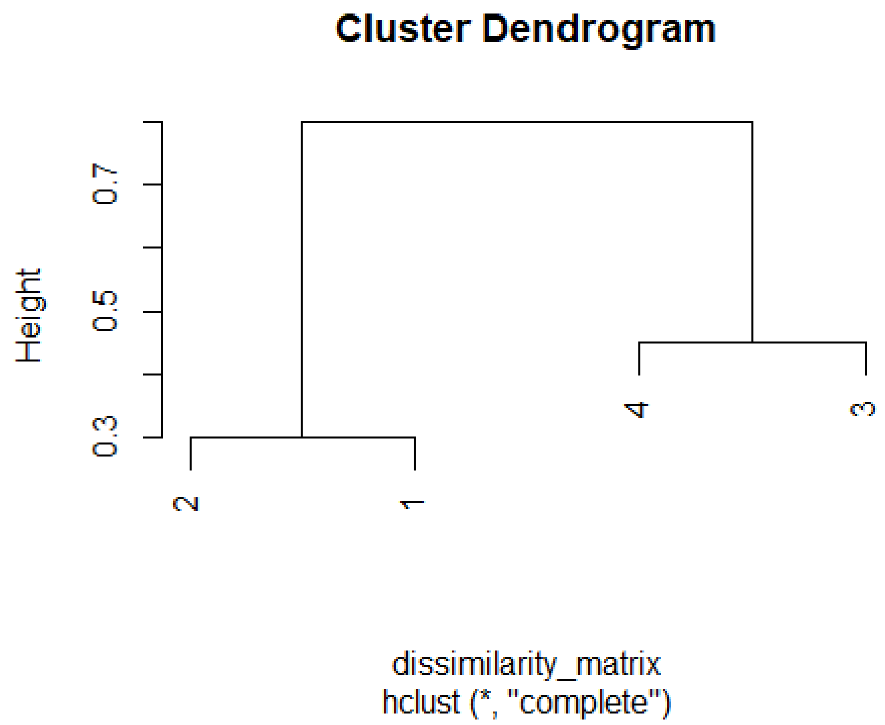
#2d

```
#Cut the tree that resulted from hclust into several groups by specifying 2 clusters (k=2)
single_linkage_cut <- cutree(single_linkage_clustering, k = 2)

#returns a vector with group memberships
single_linkage_cut

## [1] 1 1 1 2
```

We can see from above that based on the returned group memberships from the vector, the observations in each cluster from (b) are: ((1,2),3), (4)

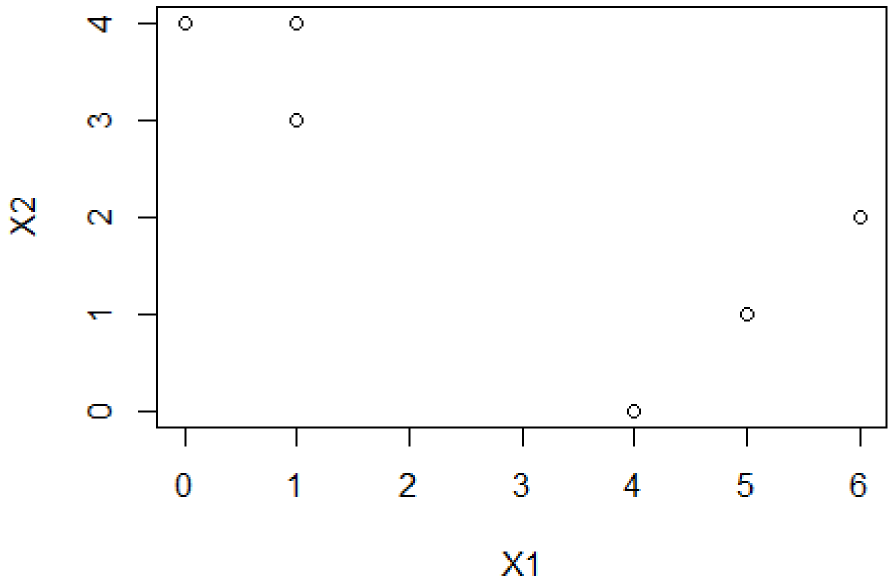
#2e

```
#Plotting a dendrogram that is equivalent to the dendrogram in (a)
plot(hclust(dissimilarity_matrix, method="complete"), labels=c(2,1,4,3))
```



#3a

```
#Creating a data frame of the observations given
observations = data.frame(X1 = c(1, 1, 0, 5, 6, 4), X2 = c(4, 3, 4, 1, 2, 0))
observations

##   X1 X2
## 1  1  4
## 2  1  3
## 3  0  4
## 4  5  1
## 5  6  2
## 6  4  0

#Plotting the observations
plot(observations)
```



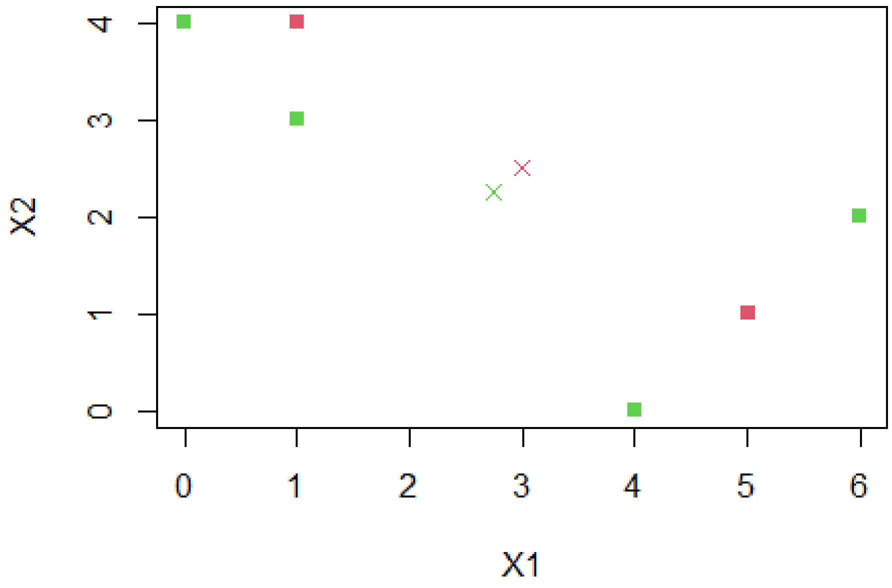
#3b

```
#Randomly assign a cluster label to each observation using the sample() command
set.seed(3)
cluster_label = sample(2, nrow(observations), replace=T)
cbind(observations,cluster_label)

##   X1 X2 cluster_label
## 1  1  4             1
## 2  1  3             2
## 3  0  4             2
## 4  5  1             1
## 5  6  2             2
## 6  4  0             2
```

#3c

```
#Computing the centroid for each cluster
centroid_label_1 = c(mean(observations[cluster_label==1, 1]), mean(observations[cluster_label==1, 2]))
centroid_label_2 = c(mean(observations[cluster_label==2, 1]), mean(observations[cluster_label==2, 2]))

cat("centroid for label 1:",centroid_label_1)

## centroid for label 1: 3 2.5
```

```
cat("\ncentroid for label 2:",centroid_label_2)

##
## centroid for label 2: 2.75 2.25

#Plotting both centroid points on the plot along with the observations color
coded based on their cluster label
plot(observations, pch = 15, col = (cluster_label+1))
points(centroid_label_1[1], centroid_label_1[2], col=2, pch=4)
points(centroid_label_2[1], centroid_label_2[2], col=3, pch=4)
```



#3d

```
#Function to compute euclidean distance
euclidean_distance = function(a, b) {
  (sqrt((a[1] - b[1])^2 + (a[2]-b[2])^2))
}

#Go through each observation and assigns them to the centroid to which it is
closest, in terms of the euclidean distance
cluster_labels = function(observations, centroid_label_1, centroid_label_2) {
  cluster_label = rep(NA, nrow(observations))

  for (i in 1:nrow(observations)) {
    if (euclidean_distance(observations[i,], centroid_label_1) <
euclidean_distance(observations[i,], centroid_label_2)) {
      cluster_label[i] = 1
    } else {
      cluster_label[i] = 2
    }
  }
  return(cluster_label)
}

cluster_label = cluster_labels(observations, centroid_label_1,
centroid_label_2)
cat("cluster labels for each observation after assigning in terms of the
euclidean distance:",cluster_label)

## cluster labels for each observation after assigning in terms of the
euclidean distance: 2 2 2 1 1 2
```

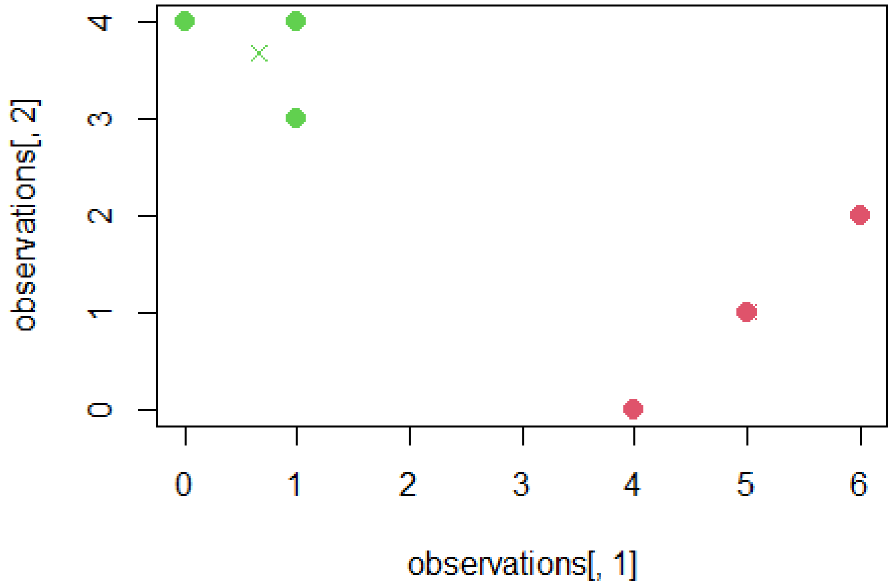
#3e

```
#Repeat (c) and (d) until the answers obtained stop changing
last_labels = rep(-1, 6)

while (!all(last_labels == cluster_label)) {
 last_labels = cluster_label

 #Compute centroid for each cluster (c)
 centroid_label_1 = c(mean(observations[cluster_label==1, 1]),
mean(observations[cluster_label==1, 2]))
 centroid_label_2 = c(mean(observations[cluster_label==2, 1]),
mean(observations[cluster_label==2, 2]))

 print(centroid_label_1)
 print(centroid_label_2)

 #Assign each observation to the centroid using the function from (d)
 cluster_label = cluster_labels(observations, centroid_label_1,
centroid_label_2)
}

## [1] 5.5 1.5
## [1] 1.50 2.75
## [1] 5 1
## [1] 0.6666667 3.6666667

print(cluster_label)

## [1] 2 2 2 1 1 1
```

#3f

```
#Coloring the observations based on the cluster labels obtained
plot(observations[,1], observations[,2], col=(cluster_label+1), pch=20,
cex=2)
```

```
points(centroid_label_1[1], centroid_label_1[2], col=2, pch=4)
points(centroid_label_2[1], centroid_label_2[2], col=3, pch=4)
```



#4a

There is not enough information to tell which fusion will occur higher on the tree or whether they will fuse at the same height. To determine this, we will need more information such as the dissimilarity matrix. This is because the dissimilarity between two clusters determines the height at which fusion takes place. If the dissimilarity for both the single and complete linkage are the same then fusion will occur at the same height. Otherwise, single linkage would typically fuse at a lower height on the tree than complete linkage.

#4b

The different types of linkage will impact the height at which 'clusters' fuse but in this case, since we are fusing leaf nodes, the type of linkage will not affect. Therefore, They will fuse at the same height.

#Practicum Problems

```
#install.packages("collections")
library(factoextra)

## Warning: package 'factoextra' was built under R version 4.2.3

## Loading required package: ggplot2
## Welcome! Want to learn more? See two factoextra-related books at
https://goo.gl/ve3WBa
```

```
library(collections)
```

```
## Warning: package 'collections' was built under R version 4.2.3
```

```
##
## Attaching package: 'collections'
```

```
## The following object is masked from 'package:utils':
##
##     stack
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

#Question 1

```
#load the dataset and label the columns
wine_dataframe <- read.csv(url("https://archive.ics.uci.edu/ml/machine-
learning-databases/wine/wine.data"), sep=",", header=F)

col_names <- c('Alcohol','Malic acid','Ash','Alcalinity of
ash','Magnesium','Total
phenols','Flavanoids','Nonflavanoid','phenols','Proanthocyanins','Color
intensity','Hue','OD280/OD315 of diluted wines','Proline')

colnames(wine_dataframe) <- col_names

summary(wine_dataframe)
```

```
##     Alcohol         Malic acid          Ash         Alcalinity of ash
##  Min.   :1.000   Min.   :11.03   Min.   :0.740   Min.   :1.360
##  1st Qu.:1.000   1st Qu.:12.36   1st Qu.:1.603   1st Qu.:2.210
##  Median :2.000   Median :13.05   Median :1.865   Median :2.360
##  Mean   :1.938   Mean   :13.00   Mean   :2.336   Mean   :2.367
##  3rd Qu.:3.000   3rd Qu.:13.68   3rd Qu.:3.083   3rd Qu.:2.558
##  Max.   :3.000   Max.   :14.83   Max.   :5.800   Max.   :3.230
##    Magnesium      Total phenols      Flavanoids     Nonflavanoid
##  Min.   :10.60   Min.   : 70.00   Min.   :0.980   Min.   :0.340
##  1st Qu.:17.20   1st Qu.: 88.00   1st Qu.:1.742   1st Qu.:1.205
##  Median :19.50   Median : 98.00   Median :2.355   Median :2.135  
##  Mean   :19.49   Mean   : 99.74   Mean   :2.295   Mean   :2.029  
##  3rd Qu.:21.50   3rd Qu.:107.00   3rd Qu.:2.800   3rd Qu.:2.875  
##  Max.   :30.00   Max.   :162.00   Max.   :3.880   Max.   :5.080  
##     phenols       Proanthocyanins Color intensity       Hue        
##  Min.   :0.1300   Min.   :0.410   Min.   : 1.280   Min.   :0.4800  
##  1st Qu.:0.2700   1st Qu.:1.250   1st Qu.: 3.220   1st Qu.:0.7825  
##  Median :0.3400   Median :1.555   Median : 4.690   Median :0.9650  
##  Mean   :0.3619   Mean   :1.591   Mean   : 5.058   Mean   :0.9574  
##  3rd Qu.:0.4375   3rd Qu.:1.950   3rd Qu.: 6.200   3rd Qu.:1.1200  
##  Max.   :0.6600   Max.   :3.580   Max.   :13.000   Max.   :1.7100  
##  OD280/OD315 of diluted wines    Proline      
##  Min.   :1.270                Min.   : 278.0  
##  1st Qu.:1.938                1st Qu.: 500.5  
##  Median :2.780                Median : 673.5  
##  Mean   :2.612                Mean   : 746.9  
##  3rd Qu.:3.170                3rd Qu.: 985.0  
##  Max.   :4.000                Max.   :1680.0
```

```
head(wine_dataframe)
```

```
##   Alcohol Malic acid  Ash Alcalinity of ash Magnesium Total phenols
Flavanoids
## 1       1      14.23 1.71              2.43      15.6           127
2.80
## 2       1      13.20 1.78              2.14      11.2           100
2.65
## 3       1      13.16 2.36              2.67      18.6           101
2.80
## 4       1      14.37 1.95              2.50      16.8           113
3.85
## 5       1      13.24 2.59              2.87      21.0           118
2.80
## 6       1      14.20 1.76              2.45      15.2           112
3.27
##   Nonflavanoid phenols Proanthocyanins Color intensity  Hue
## 1         3.06    0.28            2.29            5.64 1.04
## 2         2.76    0.26            1.28            4.38 1.05
## 3         3.24    0.30            2.81            5.68 1.03
## 4         3.49    0.24            2.18            7.80 0.86
## 5         2.69    0.39            1.82            4.32 1.04
## 6         3.39    0.34            1.97            6.75 1.05
##   OD280/OD315 of diluted wines Proline
## 1                         3.92    1065
## 2                         3.40    1050
## 3                         3.17    1185
## 4                         3.45    1480
## 5                         2.93     735
## 6                         2.85    1450
```

```
#Observe the variance among the features to decide whether to scale or not

print(apply(wine_dataframe,2,var))
```

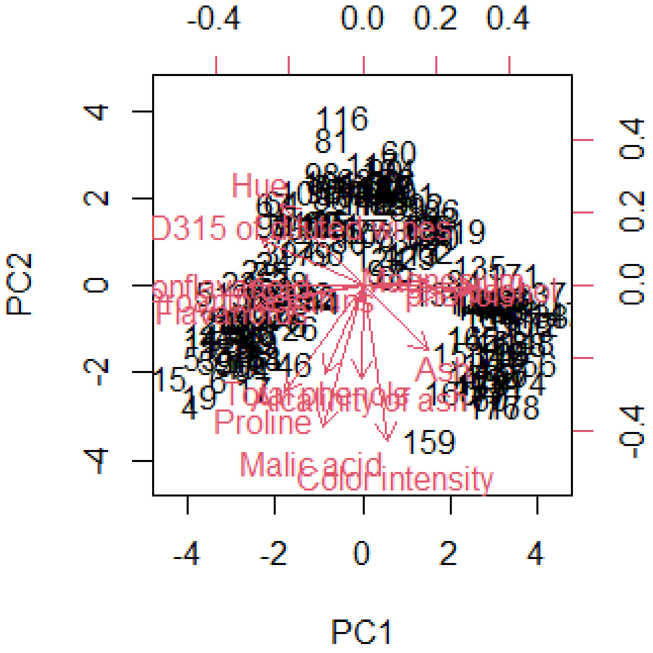
```
##                      Alcohol                   Malic acid
##                 6.006792e-01                 6.590623e-01
##                          Ash            Alcalinity of ash
##                 1.248015e+00                 7.526464e-02
##                    Magnesium                Total phenols
##                 1.115269e+01                 2.039893e+02
##                   Flavanoids                 Nonflavanoid
##                 3.916895e-01                 9.977187e-01
##                      phenols              Proanthocyanins
##                 1.548863e-02                 3.275947e-01
##              Color intensity                          Hue
##                 5.374449e+00                 5.224496e-02
## OD280/OD315 of diluted wines                      Proline
##                 5.040864e-01                 9.916672e+04
```

As shown from the variance of each feature above, there is a large difference in variance amongst some features and so to bring all features onto the same scale to avoid bias of the principal components, scaling will be performed.

```
#Use prcomp to perform a PCA on the wine data using scaling
pca_model_scaling <- prcomp(wine_dataframe , scale=TRUE)
summary(pca_model_scaling)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6
PC7
## Standard deviation     2.3529 1.5802 1.2025 0.96328 0.93675 0.82023
0.74418
## Proportion of Variance 0.3954 0.1784 0.1033 0.06628 0.06268 0.04806
0.03956
## Cumulative Proportion  0.3954 0.5738 0.6771 0.74336 0.80604 0.85409
0.89365
##                           PC8     PC9    PC10    PC11    PC12    PC13
PC14
## Standard deviation     0.5916 0.54272 0.51216 0.47524 0.41085 0.35995
0.24044
## Proportion of Variance 0.0250 0.02104 0.01874 0.01613 0.01206 0.00925
0.00413
## Cumulative Proportion  0.9186 0.93969 0.95843 0.97456 0.98662 0.99587
1.00000
```

```
#Plotting biplot of the results
biplot(pca_model_scaling,scale=0)
```



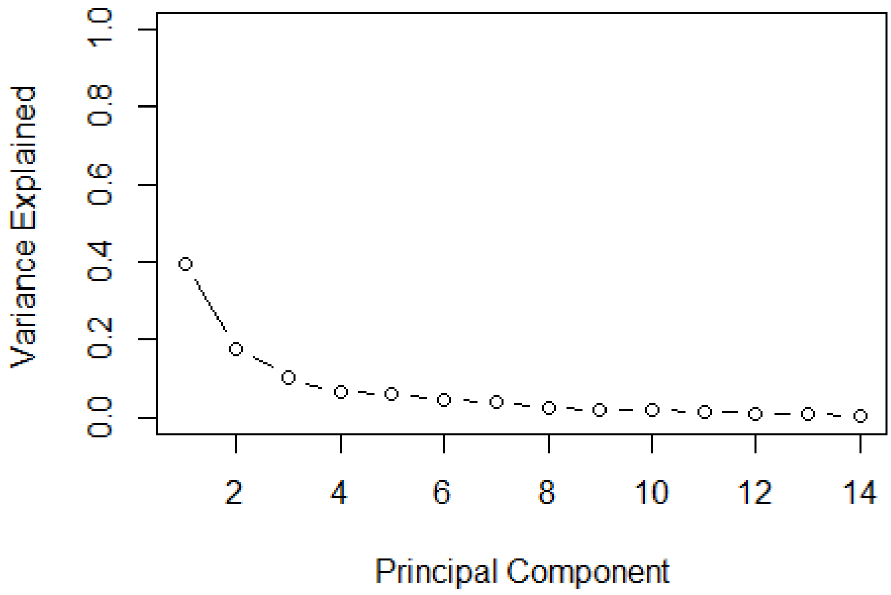
Based on the biplot above, Ash is a feature which is pointed in the opposite direction of Hue in the principal component/rotated feature space. Regarding the correlation of this feature to Hue, they are inversely correlated.

```
calculated_value <- cor.test(wine_dataframe$Ash, wine_dataframe$Hue,method =
"pearson")
calculated_value

##
##  Pearson's product-moment correlation
##
## data:  wine_dataframe$Ash and wine_dataframe$Hue
## t = -8.9975, df = 176, p-value = 3.648e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.6543579 -0.4514847
## sample estimates:
##        cor
## -0.5612957
```

Based on the correlation value of Hue and Ash as shown above (-0.5612957), we can also confirm that they are indeed negatively correlated.

```
#Compute the proportion of variance explained by each principal component
prop_variance <- pca_model_scaling$sdev^2 / sum(pca_model_scaling$sdev^2)
prop_variance
```

```
##  [1] 0.395424860 0.178362589 0.103291016 0.066279845 0.062678751
0.048055596
##  [7] 0.039557068 0.025002441 0.021038710 0.018736150 0.016132030
0.012056908
## [13] 0.009254584 0.004129451

cat("\nThe total variance explained by PC1 and PC2 is:", (prop_variance[1] +
prop_variance[2])*100,"%")

##
## The total variance explained by PC1 and PC2 is: 57.37874 %

#Plotting a screeplot of results
plot(prop_variance, xlab = "Principal Component", ylab = "Variance Explained
", ylim = c(0,1), type="b")
```

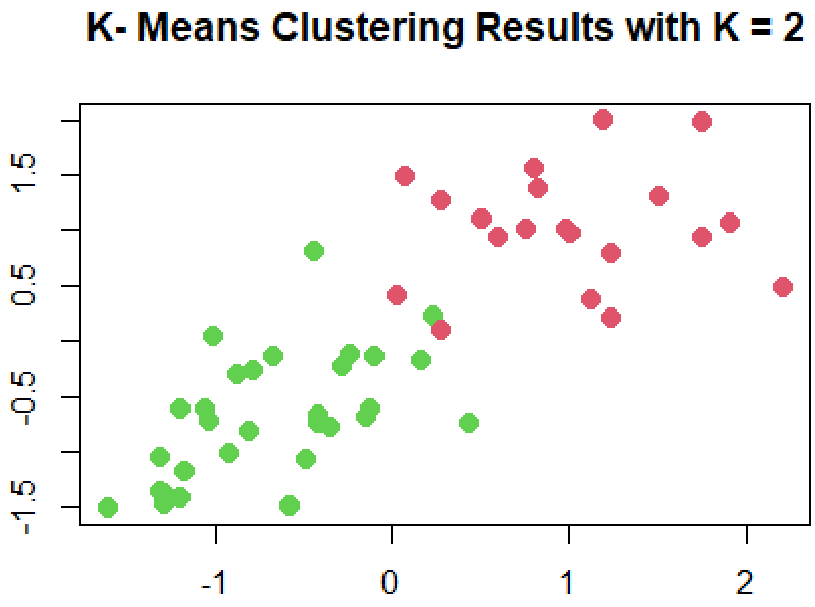


#Question 2

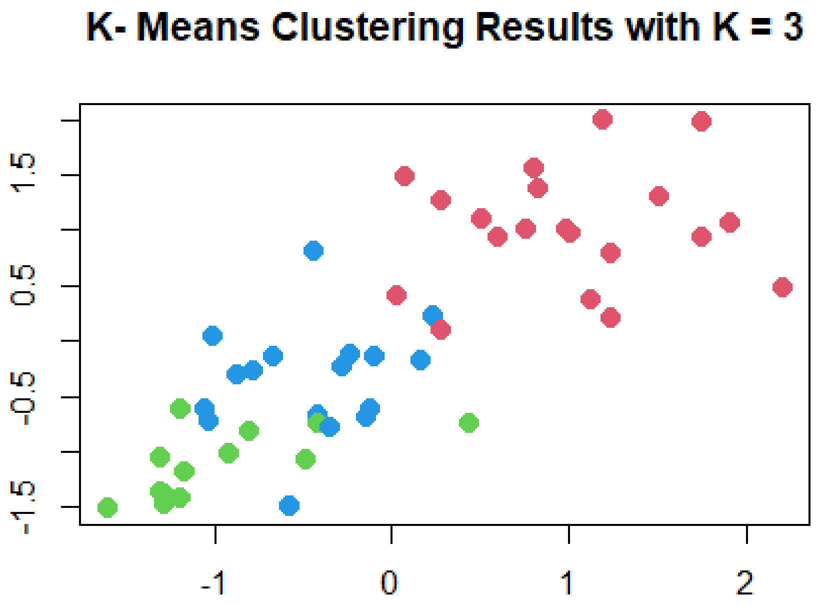
```
set.seed(30)
arrest_data <- data.frame(USArrests)

summary(arrest_data)

##      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

head(arrest_data)

##            Murder Assault UrbanPop Rape
## Alabama      13.2     236       58 21.2
## Alaska       10.0     263       48 44.5
## Arizona       8.1     294       80 31.0
## Arkansas      8.8     190       50 19.5
## California    9.0     276       91 40.6
## Colorado      7.9     204       78 38.7

#Observe the variance among the features to decide whether to scale or not

print(apply(arrest_data,2,var))

##     Murder    Assault   UrbanPop       Rape 
##   18.97047 6945.16571  209.51878   87.72916
```

As shown above, since there is a significant difference in the variance for each feature, by applying scaling, all variables will be on the same scale and have equal weight which will be be beneficial during k means clustering

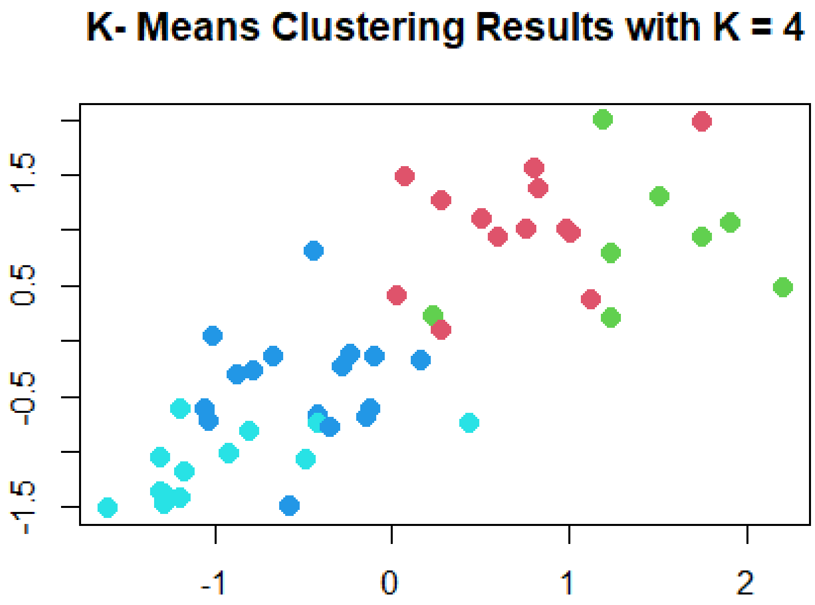
```
#Perform scaling
arrest_data_scaled <- scale(arrest_data,center = TRUE,scale=TRUE)

#Perform kmeans clustering on the scaled observations with increasing values
of k from 2 to 10
k2 <- kmeans(arrest_data_scaled, centers = 2, nstart = 25)
k3 <- kmeans(arrest_data_scaled, centers = 3, nstart = 25)
k4 <- kmeans(arrest_data_scaled, centers = 4, nstart = 25)
k5 <- kmeans(arrest_data_scaled, centers = 5, nstart = 25)
k6 <- kmeans(arrest_data_scaled, centers = 6, nstart = 25)
k7 <- kmeans(arrest_data_scaled, centers = 7, nstart = 25)
k8 <- kmeans(arrest_data_scaled, centers = 8, nstart = 25)
k9 <- kmeans(arrest_data_scaled, centers = 9, nstart = 25)
k10 <- kmeans(arrest_data_scaled, centers = 10, nstart = 25)

#Visualize the clustering for each value of k performed
plot2 <- fviz_cluster(k2, geom = "point", data = arrest_data_scaled) +
ggtitle("k = 2")
plot2_normal <- plot (arrest_data_scaled, col = (k2$cluster + 1),main = "K-
Means Clustering Results with K = 2",xlab = "", ylab = "", pch = 20, cex = 2)
```



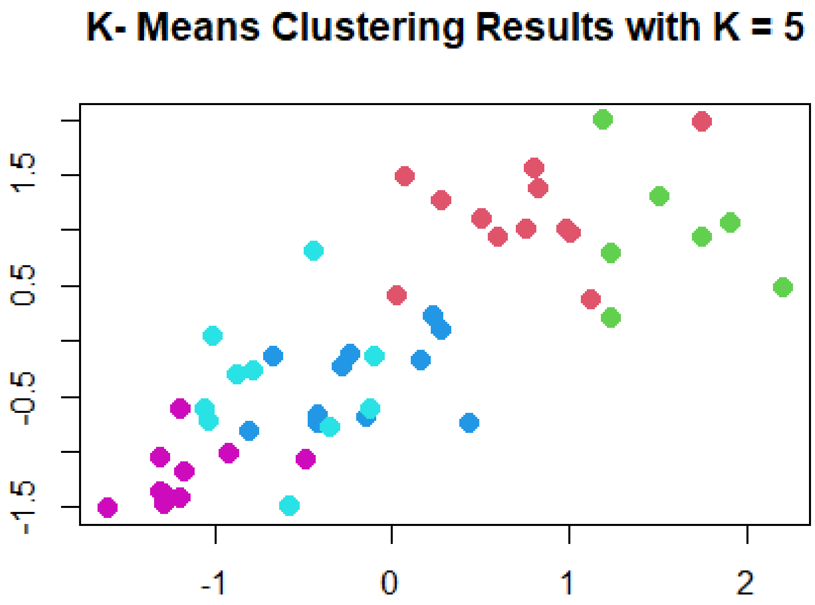
```
plot3 <- fviz_cluster(k3, geom = "point",  data = arrest_data_scaled) +
ggtitle("k = 3")
plot3_normal <- plot (arrest_data_scaled, col = (k3$cluster + 1),main = "K-
Means Clustering Results with K = 3",xlab = "", ylab = "", pch = 20, cex = 2)
```



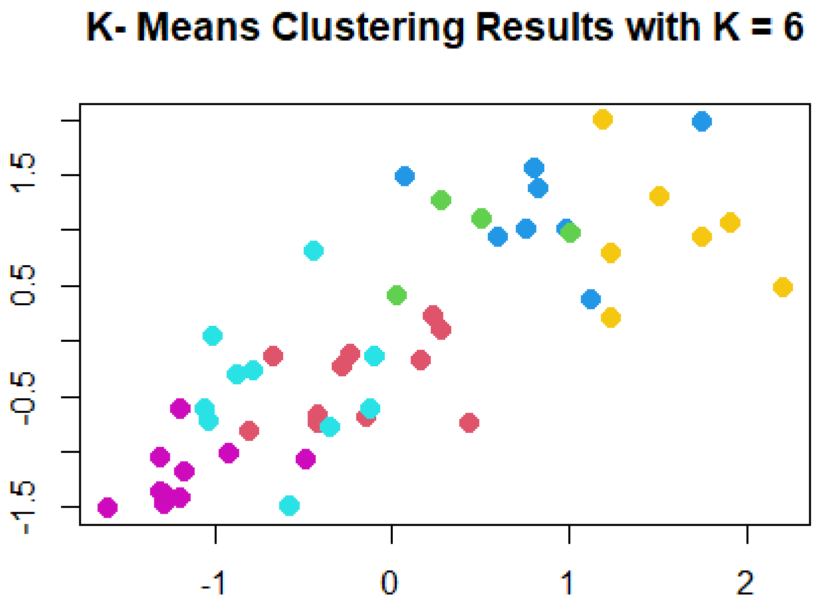
```
plot4 <- fviz_cluster(k4, geom = "point",  data = arrest_data_scaled) +
ggtitle("k = 4")
plot4_normal <- plot (arrest_data_scaled, col = (k4$cluster + 1),main = "K-
Means Clustering Results with K = 4",xlab = "", ylab = "", pch = 20, cex = 2)
```



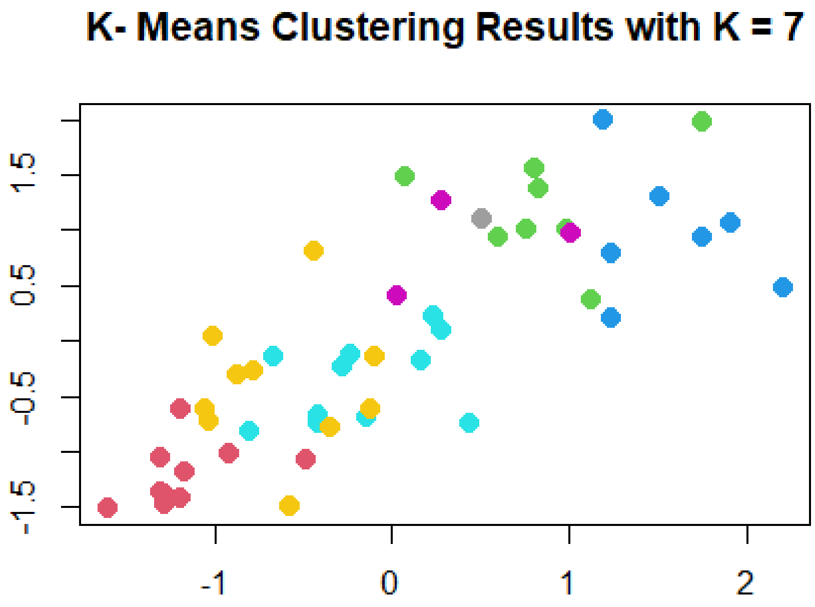
```
plot5 <- fviz_cluster(k5, geom = "point",  data = arrest_data_scaled) +
ggtitle("k = 5")
plot5_normal <- plot (arrest_data_scaled, col = (k5$cluster + 1),main = "K-
Means Clustering Results with K = 5",xlab = "", ylab = "", pch = 20, cex = 2)
```



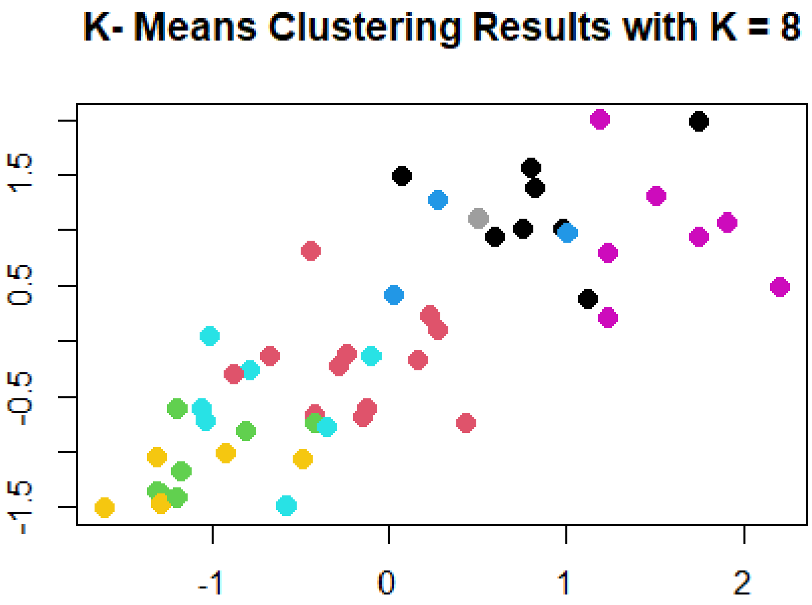
```
plot6 <- fviz_cluster(k6, geom = "point", data = arrest_data_scaled) +
ggtitle("k = 6")
plot6_normal <- plot (arrest_data_scaled, col = (k6$cluster + 1),main = "K-
Means Clustering Results with K = 6",xlab = "", ylab = "", pch = 20, cex = 2)
```



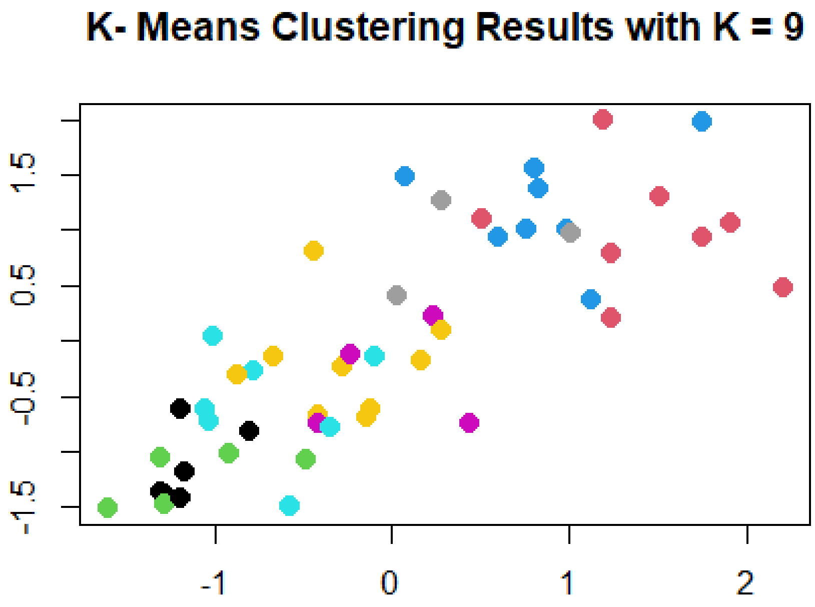
```
plot7 <- fviz_cluster(k7, geom = "point",  data = arrest_data_scaled) +
ggtitle("k = 7")
plot7_normal <- plot (arrest_data_scaled, col = (k7$cluster + 1),main = "K-
Means Clustering Results with K = 7",xlab = "", ylab = "", pch = 20, cex = 2)
```



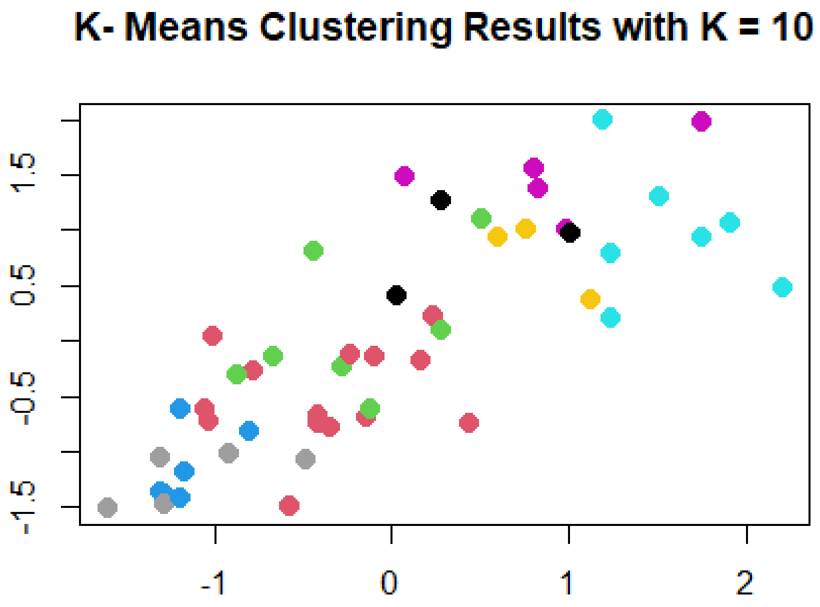
```
plot8 <- fviz_cluster(k8, geom = "point",  data = arrest_data_scaled) +
ggtitle("k = 8")
plot8_normal <- plot (arrest_data_scaled, col = (k8$cluster + 1),main = "K-
Means Clustering Results with K = 8",xlab = "", ylab = "", pch = 20, cex = 2)
```



```
plot9 <- fviz_cluster(k9, geom = "point",  data = arrest_data_scaled) +
ggtitle("k = 9")
plot9_normal <- plot (arrest_data_scaled, col = (k9$cluster + 1),main = "K-
Means Clustering Results with K = 9",xlab = "", ylab = "", pch = 20, cex = 2)
```



```
plot10 <- fviz_cluster(k10, geom = "point", data = arrest_data_scaled) +
ggtitle("k = 10")
plot10_normal <- plot (arrest_data_scaled, col = (k10$cluster + 1),main = "K-
Means Clustering Results with K = 10",xlab = "", ylab = "", pch = 20, cex =
2)
```



K- Means Clustering Results with K = 10

```
k2

## K-means clustering with 2 clusters of sizes 20, 30
## 
## Cluster means:
##      Murder    Assault   UrbanPop       Rape
## 1  1.004934  1.0138274  0.1975853  0.8469650
## 2 -0.669956 -0.6758849 -0.1317235 -0.5646433
## 
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California 
##              1              1              1              2              1 
##       Colorado    Connecticut       Delaware        Florida        Georgia 
##              1              2              2              1              1 
##         Hawaii          Idaho       Illinois        Indiana           Iowa 
##              2              2              1              2              2 
##         Kansas       Kentucky      Louisiana          Maine       Maryland 
##              2              2              1              2              1 
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri 
##              2              1              2              1              1 
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey 
##              2              2              1              2              2 
##     New Mexico       New York North Carolina   North Dakota           Ohio 
##              1              1              1              2              2 
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina 
##              2              2              2              2              1 
##   South Dakota      Tennessee          Texas           Utah        Vermont 
##              2              1              1              2              2
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              2              2              2              2              2
##
## Within cluster sum of squares by cluster:
## [1] 46.74796 56.11445
##  (between_SS / total_SS =  47.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
k3
```

```
## K-means clustering with 3 clusters of sizes 20, 13, 17
##
## Cluster means:
##       Murder    Assault   UrbanPop       Rape
## 1  1.0049340  1.0138274  0.1975853  0.8469650
## 2 -0.9615407 -1.1066010 -0.9301069 -0.9667633
## 3 -0.4469795 -0.3465138  0.4788049 -0.2571398
##
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California
##              1              1              1              3              1
##       Colorado    Connecticut       Delaware        Florida        Georgia
##              1              3              3              1              1
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              3              2              1              3              2
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##              3              2              1              2              1
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              3              1              2              1              1
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##              2              2              1              2              3
##     New Mexico       New York North Carolina   North Dakota           Ohio
##              1              1              1              2              3
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##              3              3              3              3              1
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              2              1              1              3              2
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              3              3              2              2              3
##
## Within cluster sum of squares by cluster:
## [1] 46.74796 11.95246 19.62285
##  (between_SS / total_SS =  60.0 %)
##
```

```
## Available components:
## 
## [1] "cluster"      "centers"      "totss"        "withinss"     
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
k4
```

```
## K-means clustering with 4 clusters of sizes 13, 8, 16, 13
## 
## Cluster means:
##       Murder    Assault   UrbanPop        Rape
## 1  0.6950701  1.0394414  0.7226370  1.27693964
## 2  1.4118898  0.8743346 -0.8145211  0.01927104
## 3 -0.4894375 -0.3826001  0.5758298 -0.26165379
## 4 -0.9615407 -1.1066010 -0.9301069 -0.96676331
## 
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California 
##              2              1              1              2              1 
##       Colorado    Connecticut       Delaware        Florida        Georgia 
##              1              3              3              1              2 
##         Hawaii          Idaho       Illinois        Indiana           Iowa 
##              3              4              1              3              4 
##         Kansas       Kentucky      Louisiana          Maine       Maryland 
##              3              4              2              4              1 
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri 
##              3              1              4              2              1 
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey 
##              4              4              1              4              3 
##     New Mexico       New York North Carolina   North Dakota           Ohio 
##              1              1              2              4              3 
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina 
##              3              3              3              3              2 
##   South Dakota      Tennessee          Texas           Utah        Vermont 
##              4              2              1              3              4 
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming 
##              3              3              4              4              3 
## 
## Within cluster sum of squares by cluster:
## [1] 19.922437  8.316061 16.212213 11.952463
##  (between_SS / total_SS =  71.2 %)
## 
## Available components:
## 
## [1] "cluster"      "centers"      "totss"        "withinss"     
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
k5
```

```
## K-means clustering with 5 clusters of sizes 12, 7, 11, 10, 10
## 
## Cluster means:
##       Murder    Assault   UrbanPop        Rape
## 1  0.7298036  1.1188219  0.7571799  1.32135653
## 2  1.5803956  0.9662584 -0.7775109  0.04844071
## 3 -0.1642225 -0.3658283 -0.2822467 -0.11697538
## 4 -0.6286291 -0.4086988  0.9506200 -0.38883734
## 5 -1.1727674 -1.2078573 -1.0045069 -1.10202608
## 
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California 
##              2              1              1              3              1 
##       Colorado    Connecticut       Delaware        Florida        Georgia 
##              1              4              4              1              2 
##         Hawaii          Idaho       Illinois        Indiana           Iowa 
##              4              5              1              3              5 
##         Kansas       Kentucky      Louisiana          Maine       Maryland 
##              3              3              2              5              1 
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri 
##              4              1              5              2              3 
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey 
##              3              3              1              5              4 
##     New Mexico       New York North Carolina   North Dakota           Ohio 
##              1              1              2              5              4 
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina 
##              3              3              4              4              2 
##   South Dakota      Tennessee          Texas           Utah        Vermont 
##              5              2              1              4              5 
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming 
##              3              4              5              5              3 
## 
## Within cluster sum of squares by cluster:
## [1] 18.257332  6.128432  7.788275  9.326266  7.443899
##  (between_SS / total_SS =  75.0 %)
## 
## Available components:
## 
## [1] "cluster"      "centers"      "totss"        "withinss"     
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
k6
```

```
## K-means clustering with 6 clusters of sizes 11, 4, 8, 10, 10, 7
## 
## Cluster means:
##       Murder    Assault   UrbanPop        Rape
## 1 -0.1642225 -0.3658283 -0.2822467 -0.11697538
## 2  0.4562038  0.9358314  0.6190084  2.26533514
## 3  0.8666035  1.2103171  0.8262657  0.84936722
## 4 -0.6286291 -0.4086988  0.9506200 -0.38883734
## 5 -1.1727674 -1.2078573 -1.0045069 -1.10202608
## 6  1.5803956  0.9662584 -0.7775109  0.04844071
## 
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California 
##              6              2              3              1              2 
##       Colorado    Connecticut       Delaware        Florida        Georgia 
##              2              4              4              3              6 
##         Hawaii          Idaho       Illinois        Indiana           Iowa 
##              4              5              3              1              5 
##         Kansas       Kentucky      Louisiana          Maine       Maryland 
##              1              1              6              5              3 
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri 
##              4              3              5              6              1 
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey 
##              1              1              2              5              4 
##     New Mexico       New York North Carolina   North Dakota           Ohio 
##              3              3              6              5              4 
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina 
##              1              1              4              4              6 
##   South Dakota      Tennessee          Texas           Utah        Vermont 
##              5              6              3              4              5 
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming 
##              1              4              5              5              1 
## 
## Within cluster sum of squares by cluster:
## [1] 7.788275 6.257771 5.888384 9.326266 7.443899 6.128432
##  (between_SS / total_SS =  78.1 %)
## 
## Available components:
## 
## [1] "cluster"      "centers"      "totss"        "withinss"     
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
k7
```

```
## K-means clustering with 7 clusters of sizes 10, 8, 7, 11, 3, 10, 1
## 
## Cluster means:
##       Murder    Assault   UrbanPop        Rape
## 1 -1.1727674 -1.2078573 -1.0045069 -1.10202608
## 2  0.8666035  1.2103171  0.8262657  0.84936722
## 3  1.5803956  0.9662584 -0.7775109  0.04844071
## 4 -0.1642225 -0.3658283 -0.2822467 -0.11697538
## 5  0.4389842  0.8788344  1.2292659  2.19237920
## 6 -0.6286291 -0.4086988  0.9506200 -0.38883734
## 7  0.5078625  1.1068225 -1.2117642  2.48420294
```

```
##
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California
##              3              7              2              4              5
##       Colorado    Connecticut       Delaware        Florida        Georgia
##              5              6              6              2              3
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              6              1              2              4              1
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##              4              4              3              1              2
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              6              2              1              3              4
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##              4              4              5              1              6
##     New Mexico       New York North Carolina   North Dakota           Ohio
##              2              2              3              1              6
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##              4              4              6              6              3
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              1              3              2              6              1
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              4              6              1              1              4
##
## Within cluster sum of squares by cluster:
## [1] 7.443899 5.888384 6.128432 7.788275 1.682387 9.326266 0.000000
##  (between_SS / total_SS =  80.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
k8
```

```
## K-means clustering with 8 clusters of sizes 12, 7, 3, 7, 7, 5, 1, 8
##
## Cluster means:
##       Murder    Assault    UrbanPop        Rape
## 1 -0.1675273 -0.2141089 -0.03154916 -0.02476943
## 2 -1.0500985 -1.0736357 -0.44195146 -0.83923219
## 3  0.4389842  0.8788344  1.22926592  2.19237920
## 4 -0.6958674 -0.5679476  1.12728218 -0.55096728
## 5  1.5803956  0.9662584 -0.77751086  0.04844071
## 6 -1.1176648 -1.2258563 -1.61246159 -1.23334676
## 7  0.5078625  1.1068225 -1.21176419  2.48420294
## 8  0.8666035  1.2103171  0.82626566  0.84936722
##
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California
##              5              7              8              1              3
##       Colorado    Connecticut       Delaware        Florida        Georgia
##              3              4              1              8              5
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              4              2              8              1              2
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##              1              1              5              6              8
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              4              8              2              5              1
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##              2              2              3              2              4
##     New Mexico       New York North Carolina   North Dakota           Ohio
##              8              8              5              6              1
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##              1              1              4              4              5
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              6              5              8              4              6
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              1              1              6              2              1
## 
## Within cluster sum of squares by cluster:
## [1] 9.890427 2.746293 1.682387 5.244931 6.128432 2.196512 0.000000
5.888384
##  (between_SS / total_SS =  82.8 %)
## 
## Available components:
## 
## [1] "cluster"      "centers"      "totss"        "withinss"
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

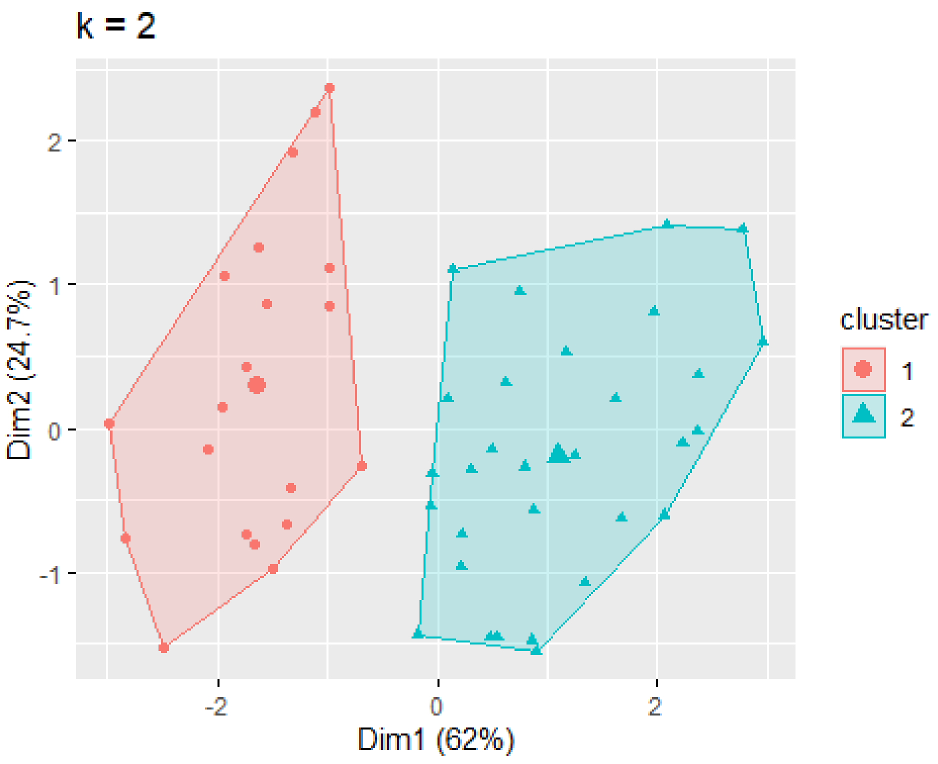
```
k9
```

```
## K-means clustering with 9 clusters of sizes 7, 5, 8, 7, 4, 9, 3, 6, 1
## 
## Cluster means:
##         Murder    Assault   UrbanPop        Rape
## 1  1.580395624  0.9662584 -0.7775109  0.04844071
## 2 -1.117664812 -1.2258563 -1.6124616 -1.23334676
## 3  0.866603499  1.2103171  0.8262657  0.84936722
## 4 -0.695867374 -0.5679476  1.1272822 -0.55096728
## 5  0.008494987 -0.3421022 -0.8145211 -0.45716680
## 6 -0.272757970 -0.2157755  0.2236843  0.11283851
## 7  0.438984207  0.8788344  1.2292659  2.19237920
## 8 -1.156695834 -1.1290614 -0.3712208 -0.89312299
## 9  0.507862482  1.1068225 -1.2117642  2.48420294
## 
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California
##              1              9              3              5              7
##       Colorado    Connecticut       Delaware        Florida        Georgia
##              7              4              6              3              1
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              4              8              3              6              8
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##              6              5              1              2              3
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              4              3              8              1              6
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##              5              8              7              8              4
##     New Mexico       New York North Carolina   North Dakota           Ohio
##              3              3              1              2              6
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##              6              6              4              4              1
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              2              1              3              4              2
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              6              6              2              8              5
## 
## Within cluster sum of squares by cluster:
## [1] 6.128432 2.196512 5.888384 5.244931 1.537684 5.381629 1.682387
1.807927
## [9] 0.000000
##  (between_SS / total_SS =  84.8 %)
## 
## Available components:
## 
## [1] "cluster"      "centers"      "totss"        "withinss"
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

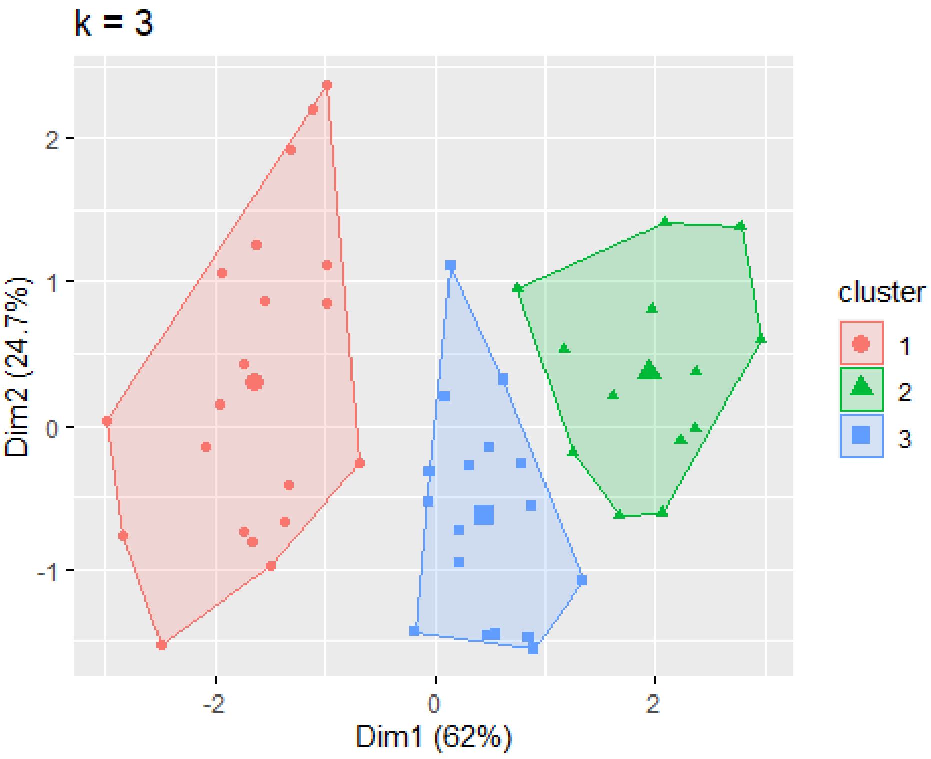
```
k10
```

```
## K-means clustering with 10 clusters of sizes 7, 1, 6, 7, 5, 3, 5, 3, 7, 6
## 
## Cluster means:
##         Murder     Assault   UrbanPop        Rape
## 1  -0.69586737 -0.56794765  1.1272822 -0.55096728
## 2   0.50786248  1.10682252 -1.2117642  2.48420294
## 3  -1.15669583 -1.12906137 -0.3712208 -0.89312299
## 4   1.58039562  0.96625839 -0.7775109  0.04844071
## 5   0.88898894  1.47640333  0.5982827  1.10480196
## 6   0.82929443  0.76684018  1.2062373  0.42364265
## 7  -1.11766481 -1.22585634 -1.6124616 -1.23334676
## 8   0.43898421  0.87883436  1.2292659  2.19237920
## 9  -0.04972355 -0.41538414 -0.4912984 -0.32218561
## 10 -0.34546282 -0.06711651  0.3656939  0.24036311
## 
## Clustering vector:
##        Alabama         Alaska        Arizona       Arkansas     California
##              4              2              5              9              8
##       Colorado    Connecticut       Delaware        Florida        Georgia
##              8              1             10              5              4
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              1              3              6              9              3
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##              9              9              4              7              5
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              1              5              3              4             10
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##              9              3              8              3              1
##     New Mexico       New York North Carolina   North Dakota           Ohio
##              5              6              4              7             10
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##             10             10              1              1              4
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              7              4              6              1              7
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              9             10              7              3              9
## 
## Within cluster sum of squares by cluster:
##  [1] 5.2449313 0.0000000 1.8079271 6.1284315 2.8689766 0.5057261 2.1965118
##  [8] 1.6823873 3.1835153 3.8140217
##  (between_SS / total_SS =  86.0 %)
## 
## Available components:
## 
## [1] "cluster"      "centers"      "totss"        "withinss"    
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

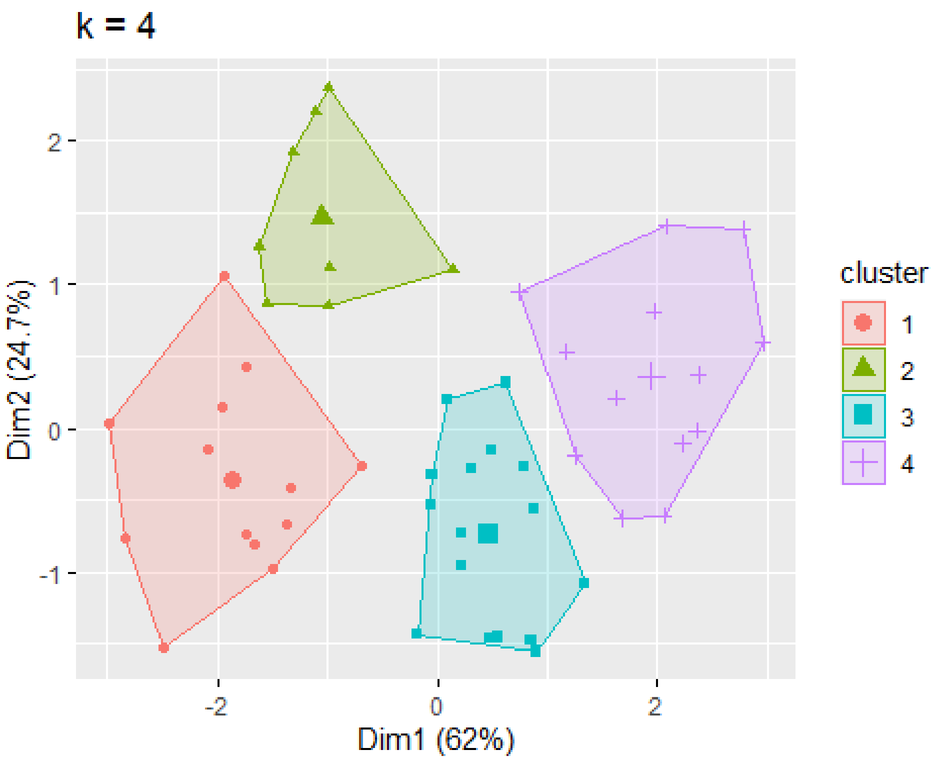
```
plot2
```



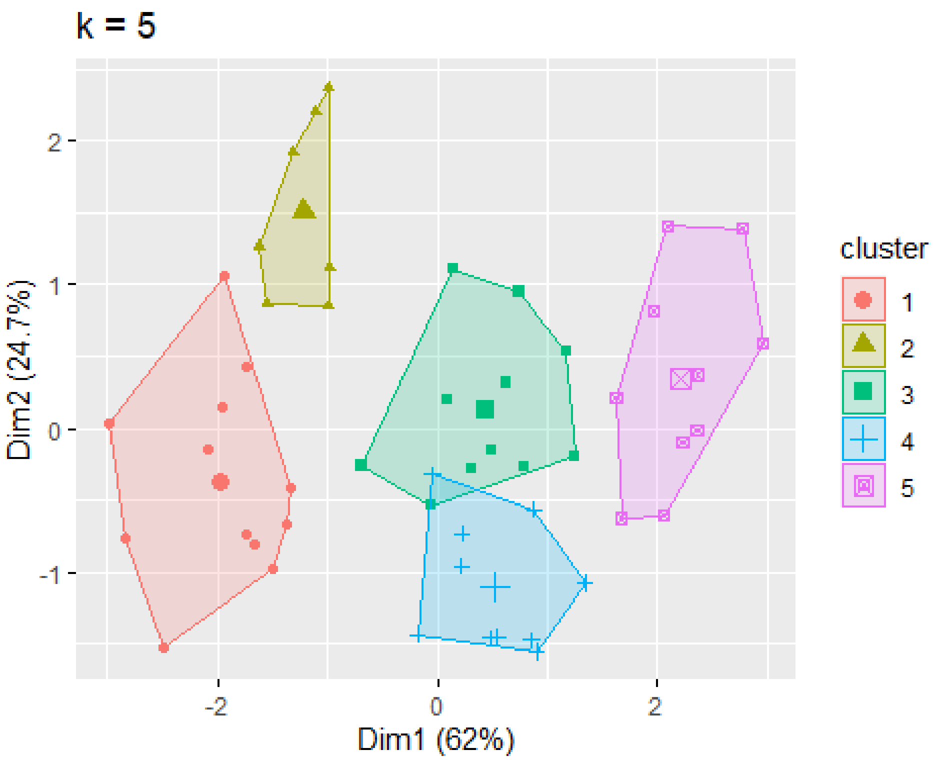
```
plot2_normal

## NULL

plot3
```



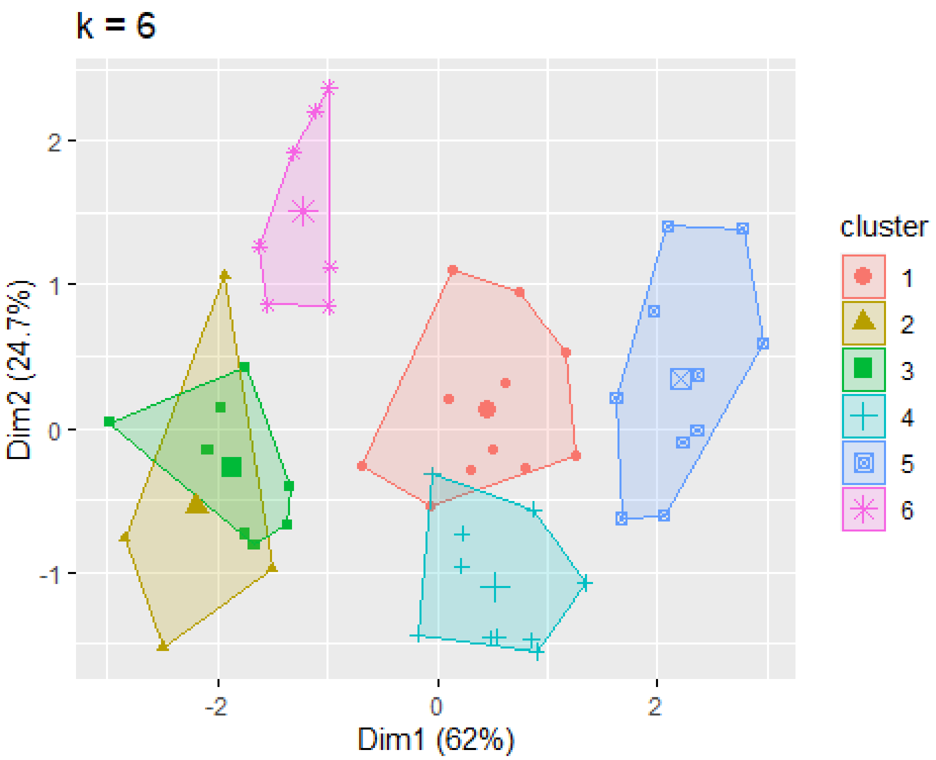
```
plot3_normal

## NULL

plot4
```



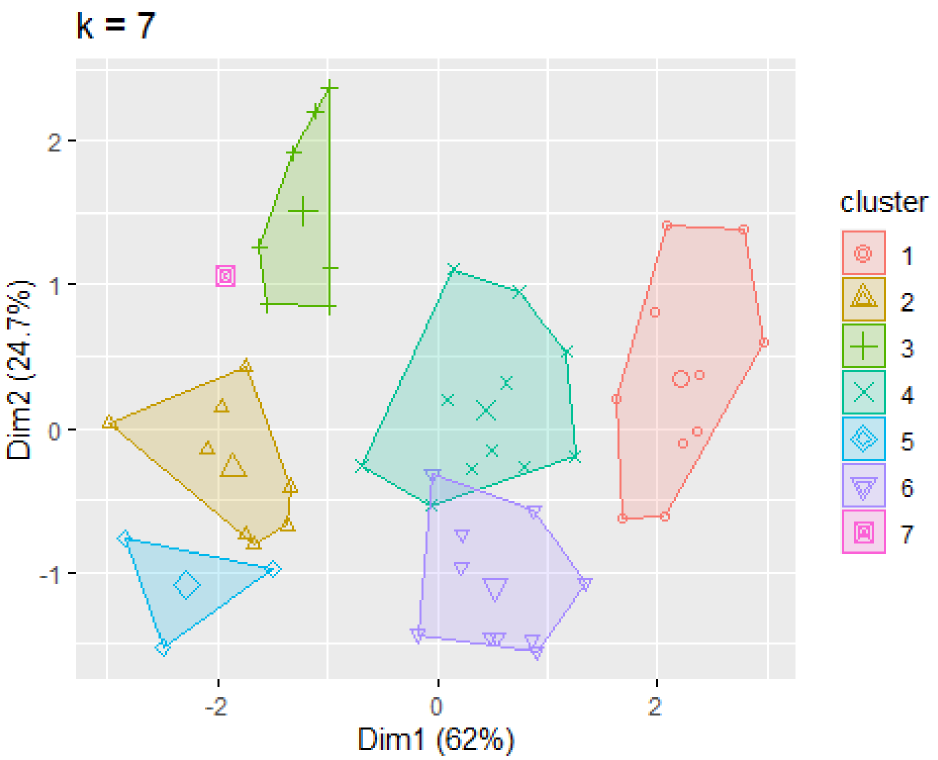
```
plot4_normal

## NULL

plot5
```



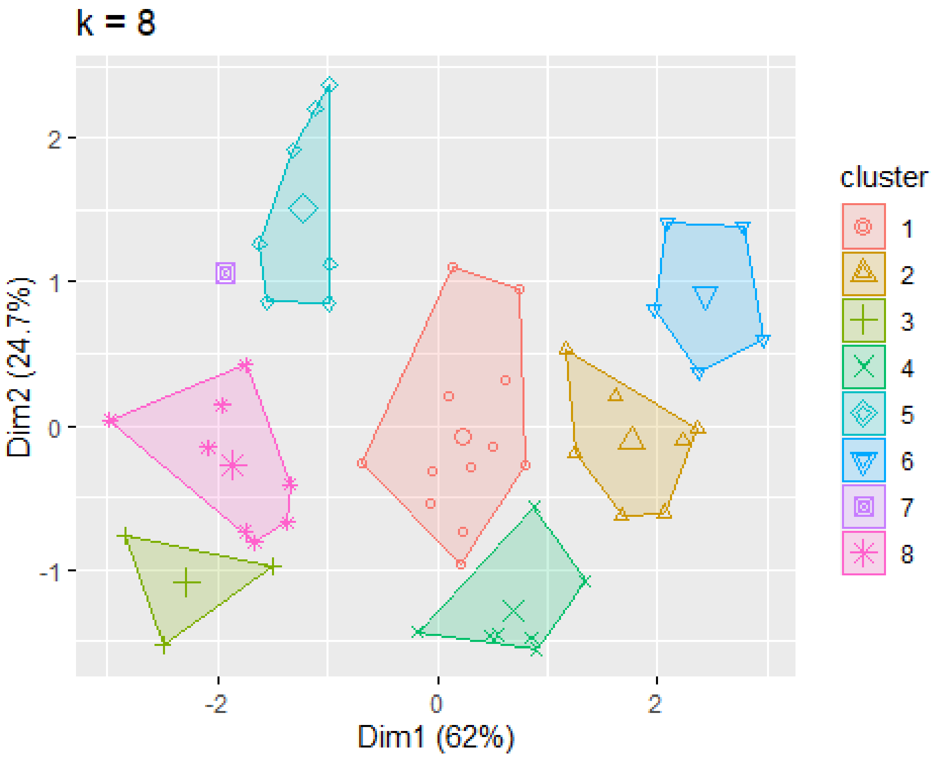
```
plot5_normal

## NULL

plot6
```



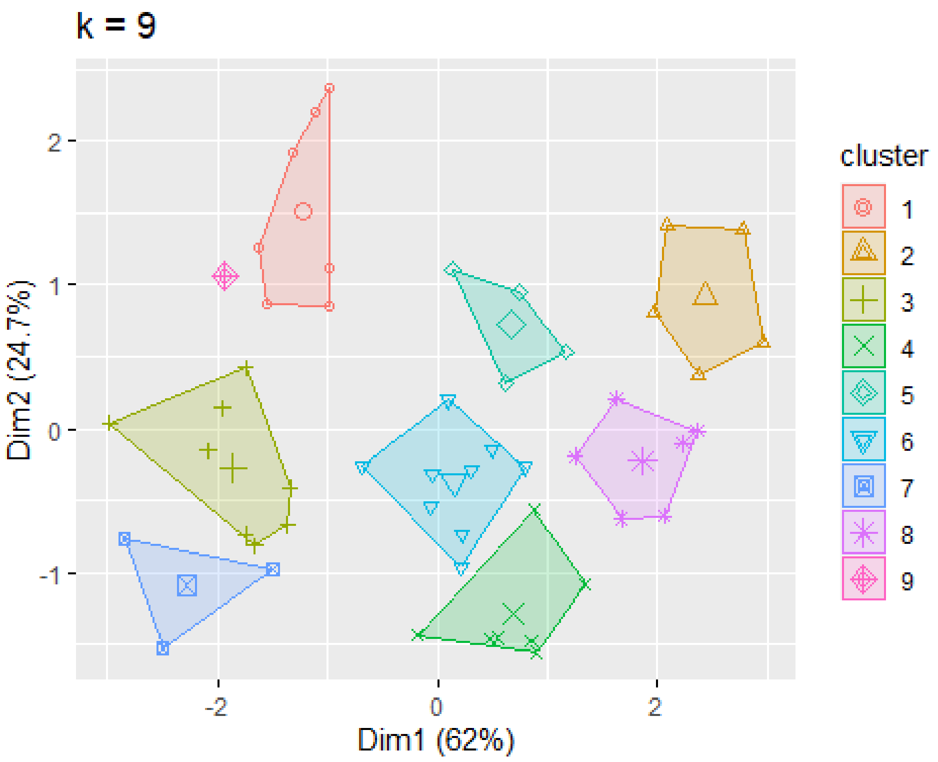
```
plot6_normal

## NULL

plot7
```



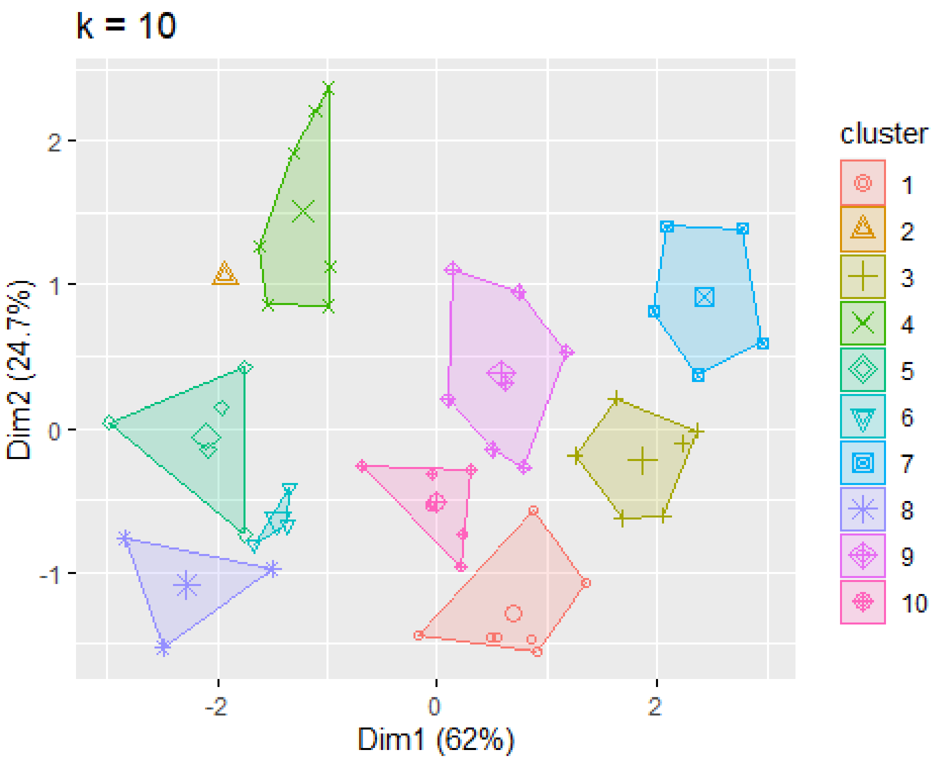
```
plot7_normal

## NULL

plot8
```



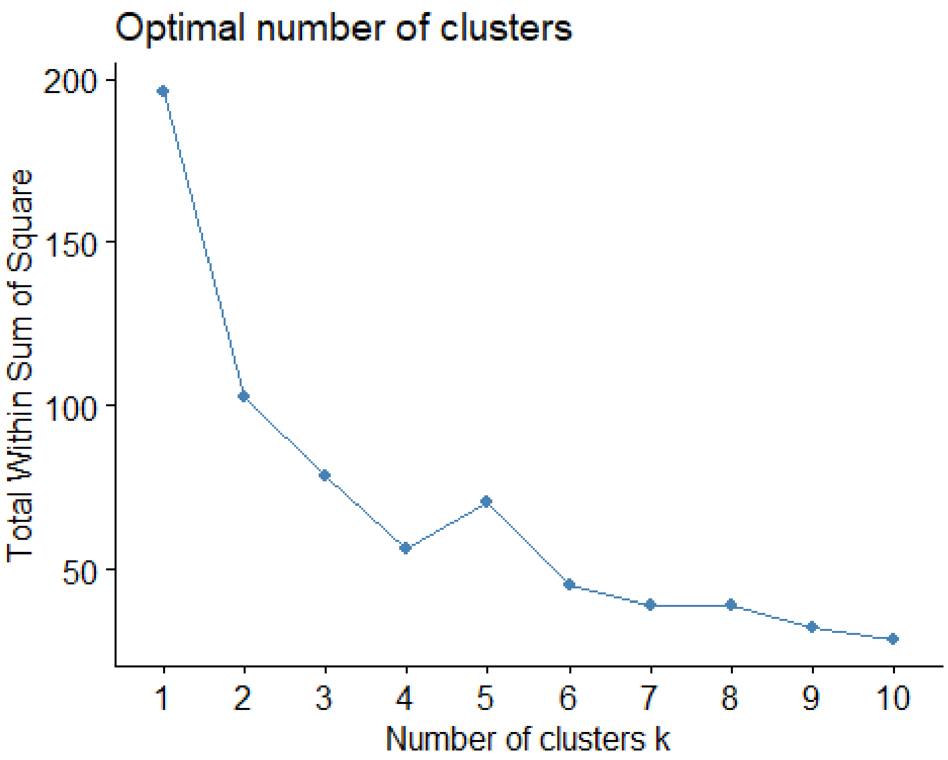
```
plot8_normal

## NULL

plot9
```



```
plot9_normal

## NULL

plot10
```



```
plot10_normal

## NULL

#Plot the within-cluster sum of squares for each value of k
fviz_nbclust(arrest_data_scaled, kmeans, method = "wss")
```



Based on the within-cluster sum of squares plotting from above, we can see that the optimal number of clusters is 4. This is because there is an elbow in the plot after the fourth cluster

#Question 3

```
white_wine_data <- read.csv('https://archive.ics.uci.edu/ml/machine-learning-
databases/wine-quality/winequality-white.csv', header = TRUE, sep = ";")
summary(white_wine_data)
```

```
##  fixed.acidity    volatile.acidity  citric.acid     residual.sugar
##  Min.   : 3.800   Min.   :0.0800   Min.   :0.0000   Min.   : 0.600
##  1st Qu.: 6.300   1st Qu.:0.2100   1st Qu.:0.2700   1st Qu.: 1.700
##  Median : 6.800   Median :0.2600   Median :0.3200   Median : 5.200
##  Mean   : 6.855   Mean   :0.2782   Mean   :0.3342   Mean   : 6.391
##  3rd Qu.: 7.300   3rd Qu.:0.3200   3rd Qu.:0.3900   3rd Qu.: 9.900
##  Max.   :14.200   Max.   :1.1000   Max.   :1.6600   Max.   :65.800
##    chlorides       free.sulfur.dioxide total.sulfur.dioxide    density
##  Min.   :0.00900   Min.   :  2.00      Min.   :  9.0        Min.   :0.9871
##  1st Qu.:0.03600   1st Qu.: 23.00      1st Qu.:108.0        1st Qu.:0.9917
##  Median :0.04300   Median : 34.00      Median :134.0        Median :0.9937
##  Mean   :0.04577   Mean   : 35.31      Mean   :138.4        Mean   :0.9940
##  3rd Qu.:0.05000   3rd Qu.: 46.00      3rd Qu.:167.0        3rd Qu.:0.9961
##  Max.   :0.34600   Max.   :289.00      Max.   :440.0        Max.   :1.0390
##        pH          sulphates         alcohol          quality
##  Min.   :2.720   Min.   :0.2200   Min.   : 8.00   Min.   :3.000
##  1st Qu.:3.090   1st Qu.:0.4100   1st Qu.: 9.50   1st Qu.:5.000
##  Median :3.180   Median :0.4700   Median :10.40   Median :6.000  
##  Mean   :3.188   Mean   :0.4898   Mean   :10.51   Mean   :5.878  
##  3rd Qu.:3.280   3rd Qu.:0.5500   3rd Qu.:11.40   3rd Qu.:6.000  
##  Max.   :3.820   Max.   :1.0800   Max.   :14.20   Max.   :9.000
```

```
head(white_wine_data)
```

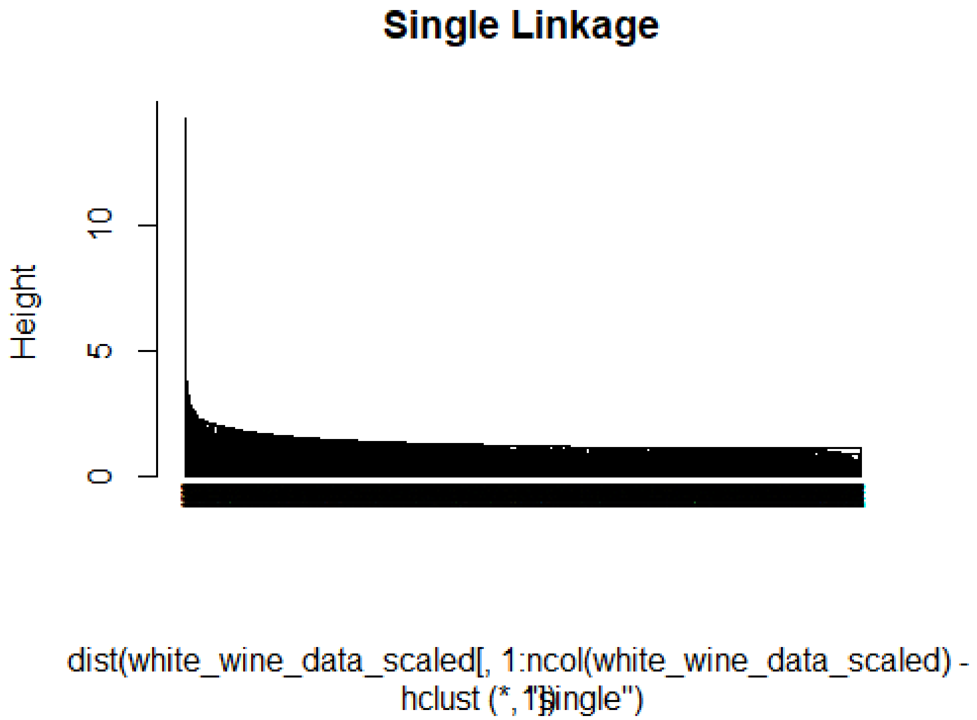
```
##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1           7.0             0.27        0.36           20.7     0.045
## 2           6.3             0.30        0.34            1.6     0.049
## 3           8.1             0.28        0.40            6.9     0.050
## 4           7.2             0.23        0.32            8.5     0.058
## 5           7.2             0.23        0.32            8.5     0.058
## 6           8.1             0.28        0.40            6.9     0.050
##   free.sulfur.dioxide total.sulfur.dioxide density   pH sulphates alcohol
## 1                  45                  170  1.0010 3.00      0.45     8.8
## 2                  14                  132  0.9940 3.30      0.49     9.5
## 3                  30                   97  0.9951 3.26      0.44    10.1
## 4                  47                  186  0.9956 3.19      0.40     9.9
## 5                  47                  186  0.9956 3.19      0.40     9.9
## 6                  30                   97  0.9951 3.26      0.44    10.1
##   quality
## 1       6
## 2       6
## 3       6
## 4       6
## 5       6
## 6       6
```

```
#Observe the variance among the features to decide whether to scale or not

print(apply(white_wine_data,2,var))
```

```
##        fixed.acidity     volatile.acidity          citric.acid 
##         7.121136e-01         1.015954e-02         1.464579e-02 
##       residual.sugar            chlorides  free.sulfur.dioxide 
##         2.572577e+01         4.773337e-04         2.892427e+02 
## total.sulfur.dioxide              density                   pH 
##         1.806085e+03         8.945524e-06         2.280118e-02 
##            sulphates              alcohol              quality 
##         1.302471e-02         1.514427e+00         7.843557e-01
```

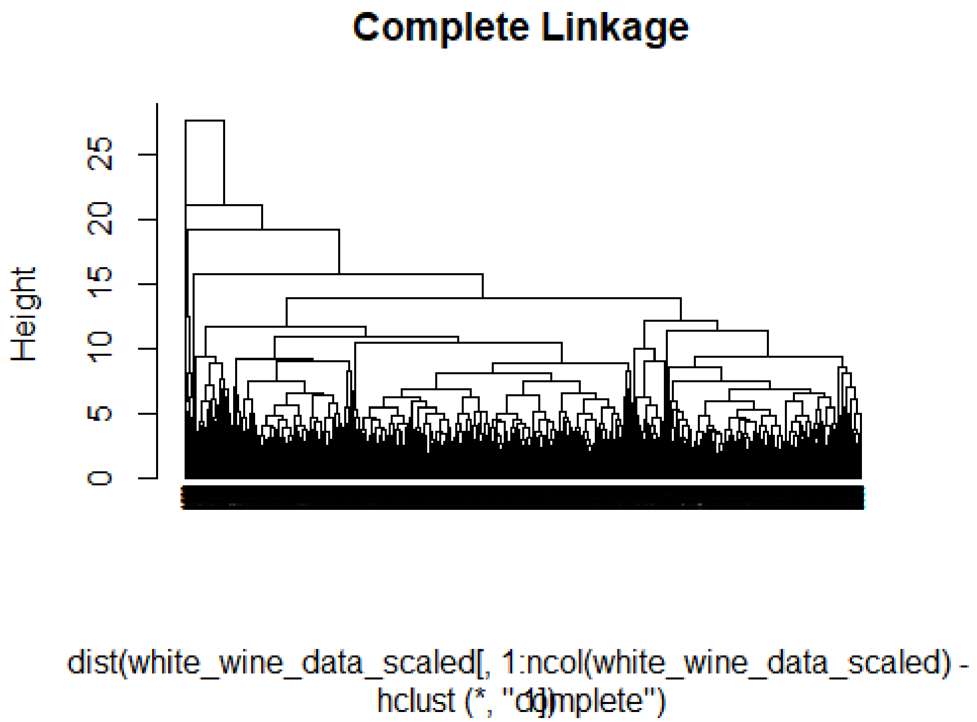
When observing the variance of each feature, it is observed that free.sulfur.dioxide and total.sulfur.dioxide have high variance when compared to the other features and so we perform scaling to bring them all to the same scale.

```
#Perform scaling
white_wine_data_scaled <- scale(white_wine_data, center = TRUE, scale=TRUE)

#Performing hierarchical clustering using single linkage
single_linkage_clust <- hclust(dist(white_wine_data_scaled[ ,
1:ncol(white_wine_data_scaled)-1]), method = "single")
plot(single_linkage_clust, cex = 0.3, hang = -1, main = "Single Linkage")
```



The clusters for single linkage merged around distance: 9

```
#Performing hierarchical clustering using complete linkage
complete_linkage_clust <- hclust(dist(white_wine_data_scaled[ ,
1:ncol(white_wine_data_scaled)-1]),method = "complete")
plot(complete_linkage_clust, cex = 0.3, hang = -1, main = "Complete Linkage")
```



The clusters for complete linkage merged around distance: 21

From above, we can see that complete linkage produces a more balanced clustering

```
single_linkage_cut = cutree(single_linkage_clust,k=2)
complete_linkage_cut = cutree(complete_linkage_clust,k=2)

#Summary statistics for single linkage

#Get summary statistics and check how many observations are in each cluster
print(table(single_linkage_cut))

## single_linkage_cut
##    1    2 
## 4897    1 

print(table(single_linkage_cut,white_wine_data$quality))

##                   
## single_linkage_cut    3    4    5    6    7    8    9
##                  1   20  163 1457 2197  880  175    5
##                  2    0    0    0    1    0    0    0

summary(white_wine_data_scaled[ , 1:ncol(white_wine_data_scaled)-
1],by=single_linkage_cut)

##  fixed.acidity      volatile.acidity   citric.acid       residual.sugar   
##  Min.   :-3.61998   Min.   :-1.9668   Min.   :-2.7615   Min.   :-1.1418
##  1st Qu.:-0.65743   1st Qu.:-0.6770   1st Qu.:-0.5304   1st Qu.:-0.9250
##  Median :-0.06492   Median :-0.1810   Median :-0.1173   Median :-0.2349
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.52758   3rd Qu.: 0.4143   3rd Qu.: 0.4612   3rd Qu.: 0.6917
##  Max.   : 8.70422   Max.   : 8.1528   Max.   :10.9553   Max.   :11.7129
##    chlorides       free.sulfur.dioxide total.sulfur.dioxide    density
##  Min.   :-1.6831   Min.   :-1.95848    Min.   :-3.0439      Min.   :-
2.31280
##  1st Qu.:-0.4473   1st Qu.:-0.72370    1st Qu.:-0.7144      1st Qu.:-
0.77063
##  Median :-0.1269   Median :-0.07691    Median :-0.1026      Median :-
0.09608
##  Mean   : 0.0000   Mean   : 0.00000    Mean   : 0.0000      Mean   :
0.00000
##  3rd Qu.: 0.1935   3rd Qu.: 0.62867    3rd Qu.: 0.6739      3rd Qu.:
0.69298
##  Max.   :13.7417   Max.   :14.91679    Max.   : 7.0977      Max.
:15.02976
##        pH              sulphates          alcohol
##  Min.   :-3.10109   Min.   :-2.3645   Min.   :-2.04309
##  1st Qu.:-0.65077   1st Qu.:-0.6996   1st Qu.:-0.82419
##  Median :-0.05475   Median :-0.1739   Median :-0.09285
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.00000
##  3rd Qu.: 0.60750   3rd Qu.: 0.5271   3rd Qu.: 0.71974
##  Max.   : 4.18365   Max.   : 5.1711   Max.   : 2.99502
```

```
single_linkage_clust
```

```
##
## Call:
## hclust(d = dist(white_wine_data_scaled[, 1:ncol(white_wine_data_scaled) -
1]), method = "single")
##
## Cluster method   : single
## Distance         : euclidean
## Number of objects: 4898
```

```
#Summary statistics for complete linkage

#Get summary statistics and check how many observations are in each cluster
print(table(complete_linkage_cut))
```

```
## complete_linkage_cut
##    1    2
## 4897    1
```

```
print(table(complete_linkage_cut,white_wine_data$quality))
```

```
##
## complete_linkage_cut    3    4    5    6    7    8    9
##                     1   20  163 1457 2197  880  175    5
##                     2    0    0    0    1    0    0    0

summary(white_wine_data_scaled[ , 1:ncol(white_wine_data_scaled)-
1],by=complete_linkage_cut)

##  fixed.acidity      volatile.acidity   citric.acid      residual.sugar
##  Min.   :-3.61998   Min.   :-1.9668   Min.   :-2.7615   Min.   :-1.1418
##  1st Qu.:-0.65743   1st Qu.:-0.6770   1st Qu.:-0.5304   1st Qu.:-0.9250
##  Median :-0.06492   Median :-0.1810   Median :-0.1173   Median :-0.2349
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.52758   3rd Qu.: 0.4143   3rd Qu.: 0.4612   3rd Qu.: 0.6917
##  Max.   : 8.70422   Max.   : 8.1528   Max.   :10.9553   Max.   :11.7129
##    chlorides       free.sulfur.dioxide total.sulfur.dioxide    density
##  Min.   :-1.6831   Min.   :-1.95848    Min.   :-3.0439      Min.   :-
2.31280
##  1st Qu.:-0.4473   1st Qu.:-0.72370    1st Qu.:-0.7144      1st Qu.:-
0.77063
##  Median :-0.1269   Median :-0.07691    Median :-0.1026      Median :-
0.09608
##  Mean   : 0.0000   Mean   : 0.00000    Mean   : 0.0000      Mean   :
0.00000
##  3rd Qu.: 0.1935   3rd Qu.: 0.62867    3rd Qu.: 0.6739      3rd Qu.:
0.69298
##  Max.   :13.7417   Max.   :14.91679    Max.   : 7.0977      Max.
:15.02976
##        pH             sulphates          alcohol
##  Min.   :-3.10109   Min.   :-2.3645   Min.   :-2.04309
##  1st Qu.:-0.65077   1st Qu.:-0.6996   1st Qu.:-0.82419
##  Median :-0.05475   Median :-0.1739   Median :-0.09285
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.00000
##  3rd Qu.: 0.60750   3rd Qu.: 0.5271   3rd Qu.: 0.71974
##  Max.   : 4.18365   Max.   : 5.1711   Max.   : 2.99502

complete_linkage_clust

##
## Call:
## hclust(d = dist(white_wine_data_scaled[, 1:ncol(white_wine_data_scaled) -
1]), method = "complete")
##
## Cluster method   : complete
## Distance         : euclidean
## Number of objects: 4898
```