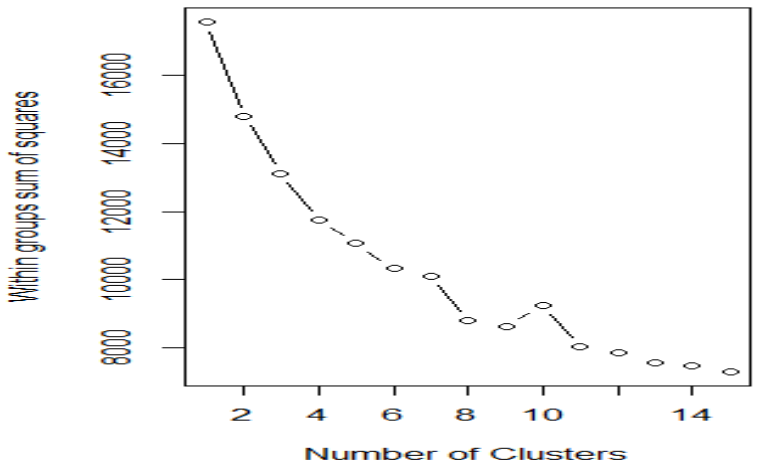
K-Means Clustering

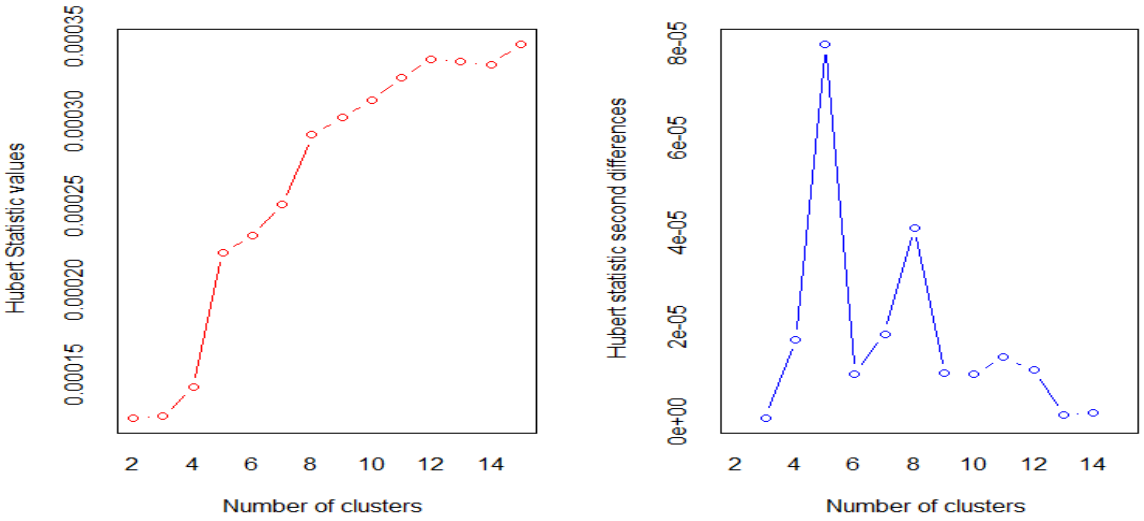
```
library(NbClust)
df <- read.csv(file.choose() ,header=TRUE)
# Plots within groups sum of squares
wssplot <- function(data, nc=15, seed=1234){
  wss <- (nrow(data)-1)*sum(apply(data,2,var))
  for (i in 2:nc){
    set.seed(seed)
    wss[i] <- sum(kmeans(data, centers=i)$withinss)}
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares")}
# Center and scale data frame columns
df_scaled <- scale(df[-1])
head(df_scaled)
```

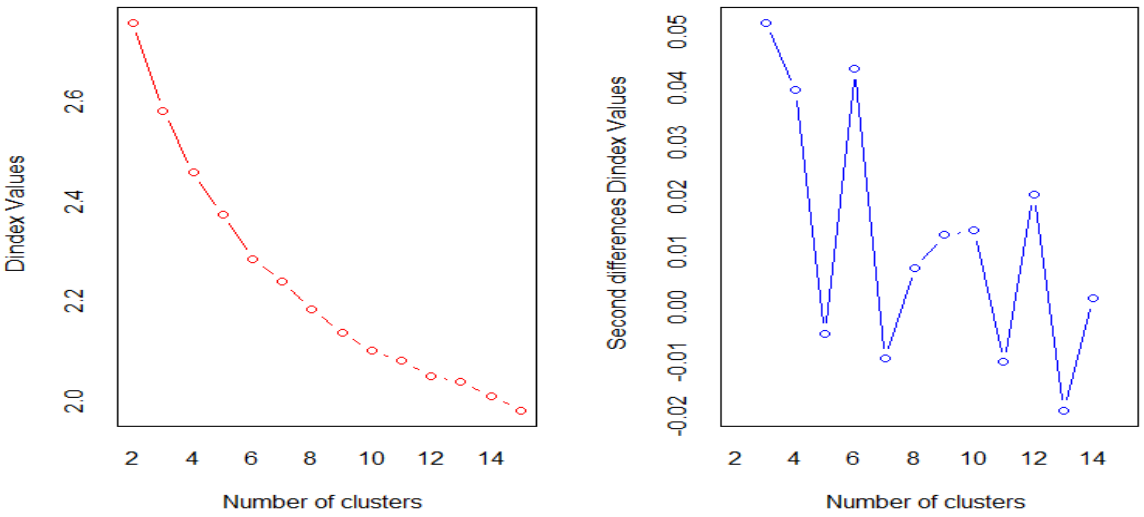
```
> head(df_scaled)
     volatile.acidity citric.acid residual.sugar   chlorides
[1,]        0.9615758   -1.391037     -0.45307667 -0.24363047
[2,]        1.9668271   -1.391037      0.04340257  0.22380518
[3,]        1.2966596   -1.185699     -0.16937425  0.09632273
[4,]       -1.3840105    1.483689     -0.45307667 -0.26487754
[5,]        0.9615758   -1.391037     -0.45307667 -0.24363047
[6,]        0.7381867   -1.391037     -0.52400227 -0.26487754
     free.sulfur.dioxide total.sulfur.dioxide    density         pH   sulphates
[1,]          -0.46604672           -0.3790141 0.55809987  1.2882399 -0.57902538
[2,]           0.87236532            0.6241680 0.02825193 -0.7197081  0.12891007
[3,]          -0.08364328            0.2289750 0.13422152 -0.3310730 -0.04807379
[4,]           0.10755844            0.4113718 0.66406945 -0.9787982 -0.46103614
[5,]          -0.46604672           -0.3790141 0.55809987  1.2882399 -0.57902538
[6,]          -0.27484500           -0.1966174 0.55809987  1.2882399 -0.57902538
        alcohol    quality
[1,] -0.9599458 -0.7875763
[2,] -0.5845942 -0.7875763
[3,] -0.5845942 -0.7875763
[4,] -0.5845942  0.4507074
[5,] -0.9599458 -0.7875763
[6,] -0.9599458 -0.7875763
```

```
# Determine optimal number of columns

wssplot(df_scaled)
```







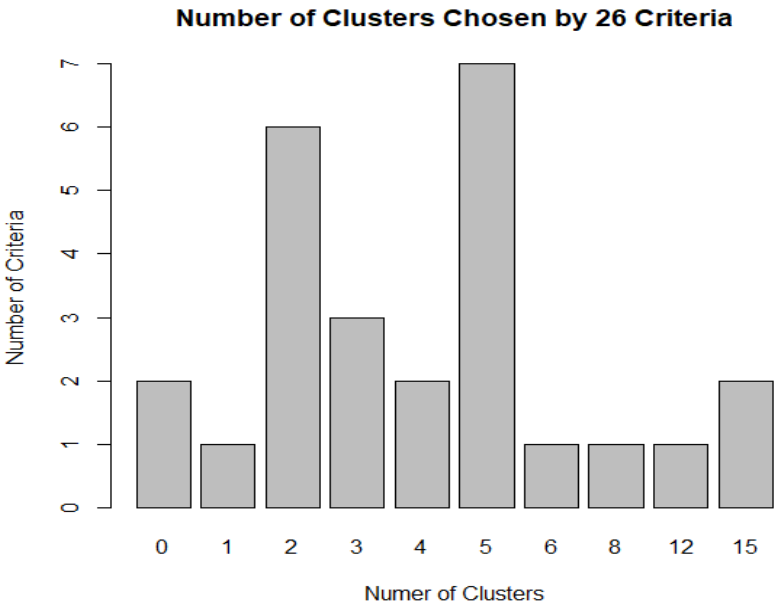

```
set.seed(1234)
nc <- NbClust(df_scaled, min.nc=2, max.nc=15, method="kmeans")
```

```
*******************************************************************
* Among all indices:
* 6 proposed 2 as the best number of clusters
* 3 proposed 3 as the best number of clusters
* 2 proposed 4 as the best number of clusters
* 7 proposed 5 as the best number of clusters
* 1 proposed 6 as the best number of clusters
* 1 proposed 8 as the best number of clusters
* 1 proposed 12 as the best number of clusters
* 2 proposed 15 as the best number of clusters

                   ***** Conclusion *****

* According to the majority rule, the best number of clusters is  5
```

```
# Optimal Number of clusters
barplot(table(nc$Best.n[1,]),
    xlab="Numer of Clusters", ylab="Number of Criteria",
    main="Number of Clusters Chosen by 26 Criteria")
```




```
# K-means cluster analysis
set.seed(1234)
fit.km <- kmeans(df, 3, nstart=25)
fit.km$size # Number of items in each cluster
```

```
> fit.km$size # Number of items in each cluster
[1] 519 839 241
```

```
> fit.km$centers # returns central value for each cluster
  fixed.acidity volatile.acidity citric.acid residual.sugar  chlorides free.sulfur.dioxide
1      8.258189        0.5210019   0.2688632       2.488054 0.09184586           21.939306
2      8.481764        0.5192431   0.2709416       2.370977 0.08448987            9.171633
3      7.887552        0.5723651   0.2756432       3.232365 0.08839834           26.151452
  total.sulfur.dioxide   density       pH sulphates  alcohol  quality
1             55.69750 0.9968552 3.322428 0.6767630 10.34399 5.603083
2             22.93802 0.9966256 3.310286 0.6483909 10.59078 5.750894
3            108.50622 0.9969347 3.289627 0.6520332 10.00892 5.307054
> aggregate(df[-1], by=list(cluster=fit.km$cluster),mean)
  cluster volatile.acidity citric.acid residual.sugar  chlorides free.sulfur.dioxide
1       1        0.5210019   0.2688632       2.488054 0.09184586           21.939306
2       2        0.5192431   0.2709416       2.370977 0.08448987            9.171633
3       3        0.5723651   0.2756432       3.232365 0.08839834           26.151452
  total.sulfur.dioxide   density       pH sulphates  alcohol  quality
1             55.69750 0.9968552 3.322428 0.6767630 10.34399 5.603083
2             22.93802 0.9966256 3.310286 0.6483909 10.59078 5.750894
3            108.50622 0.9969347 3.289627 0.6520332 10.00892 5.307054
>
```

```
# Cluster plot (using scaled data for plotting)
fviz_cluster(fit.km, data = df_scaled,
             geom = "point",
             ellipse.type = "convex",
             palette = "jco",
             ggtheme = theme_minimal(),
             main = "K-means Clustering")
```