K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Batch: D2 **Roll No.: 16010122323**

Experiment / assignment / tutorial No. 8

Title: Implementation of clustering algorithm –K-means, Hierarchical Python.

AIM: To understand the Clustering algorithm.

Expected Outcome of Experiment:

CO:

Books/ Journals/ Websites referred:

1. https://uc-r.github.io/kmeans_clustering
2. https://en.wikipedia.org/wiki/K-means_clustering

Pre Lab/ Prior Concepts:

K-means Algorithm

K-Means Clustering is an Unsupervised Learning algorithm, which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.

It allows us to cluster the data into different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training.

It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The k-means clustering algorithm mainly performs two tasks:

Determines the best value for K center points or centroids by an iterative process.

Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

Hence each cluster has datapoints with some commonalities, and it is away from other clusters.

Agglomerative Clustering approach

Also known as bottom-up approach or hierarchical agglomerative clustering (HAC). A structure that is more informative than the unstructured set of clusters returned by flat clustering. This clustering algorithm does not require us to prespecify the number of clusters. Bottom-up algorithms treat each data as a singleton cluster at the outset and then successively agglomerates pairs of clusters until all clusters have been merged into a single cluster that contains all data.

Description of the dataset used in implementation:

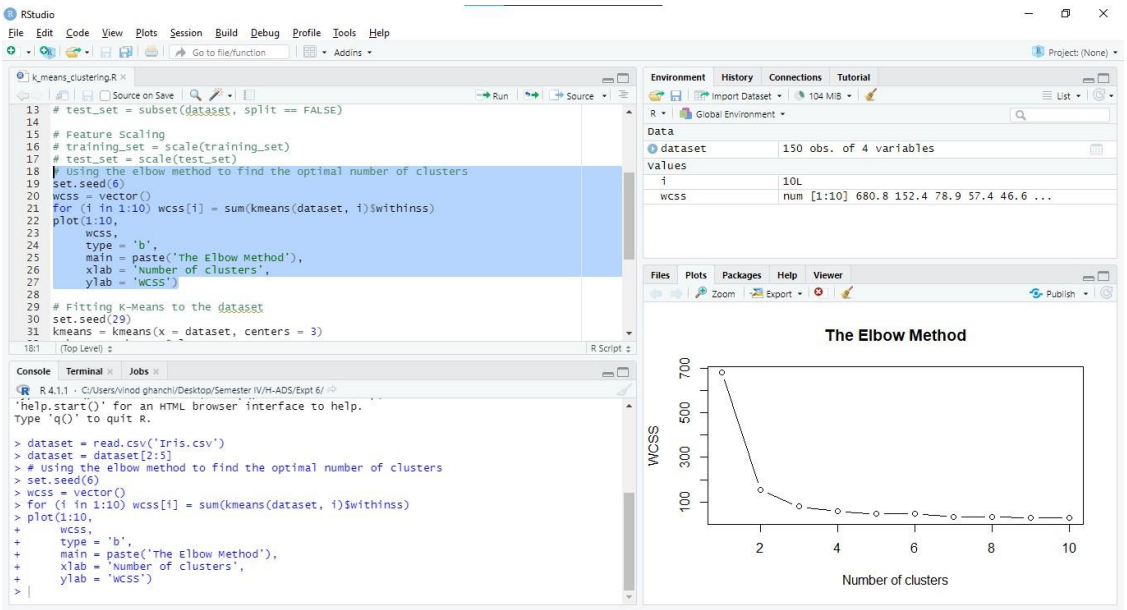
Iris dataset

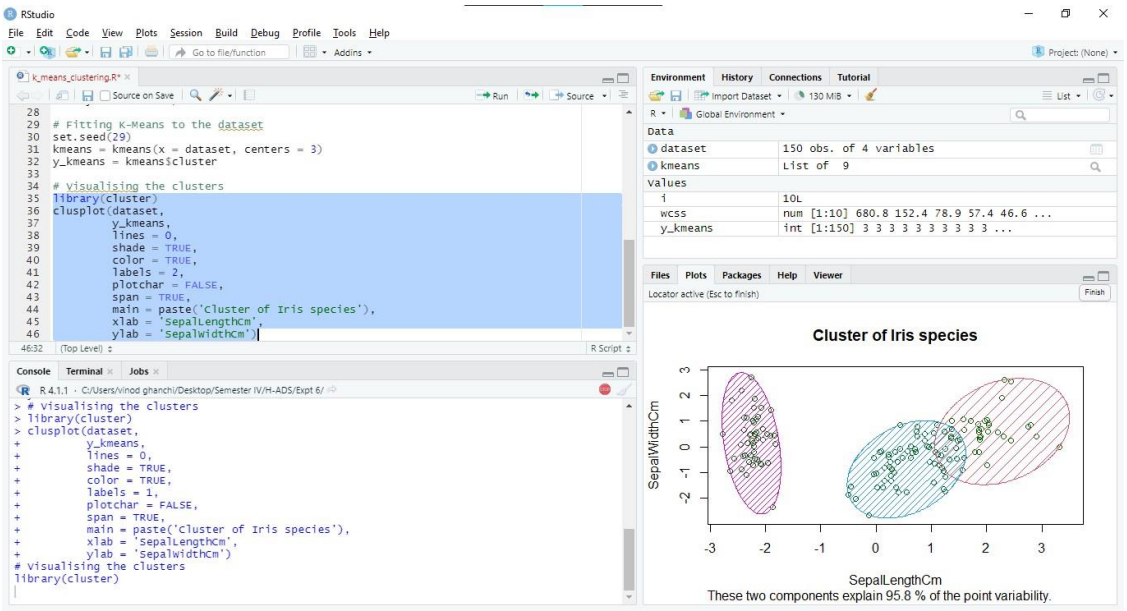
Code (R code):

```
# K-Means Clustering

# Importing the dataset
dataset = read.csv('Iris.csv')
dataset = dataset[2:5]

# Splitting the dataset into the Training set and Test set
# install.packages('caTools')
# library(caTools)
# set.seed(123)
# split = sample.split(dataset$DependentVariable, SplitRatio = 0.8)
# training_set = subset(dataset, split == TRUE)
# test_set = subset(dataset, split == FALSE)

# Feature Scaling
# training_set = scale(training_set)
# test_set = scale(test_set)
# Using the elbow method to find the optimal number of clusters
set.seed(6)
wcss = vector()
for (i in 1:10) wcss[i] = sum(kmeans(dataset, i)$withinss)
```

```
plot(1:10,
     wcss,
     type = 'b',
     main = paste('The Elbow Method'),
     xlab = 'Number of clusters',
     ylab = 'WCSS')

# Fitting K-Means to the dataset
set.seed(29)
kmeans = kmeans(x = dataset, centers = 3)
y_kmeans = kmeans$cluster

# Visualising the clusters
library(cluster)
clusplot(dataset,
         y_kmeans,
         lines =  0,
         shade = TRUE,
         color = TRUE,
         labels = 2,
         plotchar = FALSE,
         span = TRUE,
         main = paste('Cluster of Iris species'),
         xlab = 'SepalLengthCm',
         ylab = 'SepalWidthCm')
```

Output:





Conclusion:

In the above experiment we have implemented k-means clustering. To find the optimal number of clusters (k value) we have used the Elbow method.

In the Elbow method we varied the value of k from 1-10 and calculated the wcss(within cluster sum of square). On plotting the wcss with the k value the plot looks like and elbow. As the number of clusters increases the wcss value starts decreasing.

At a point in the wcss vs k value plot the graph changes rapidly creating an elbow shape. From this point the graph starts to move almost parallel to the x-axis. The k value corresponding to this point = 3 is the optimal k value which represents the optimal number of clusters.

Based on the optimal value of k we trained the model on the dataset with a number of 3 clusters.

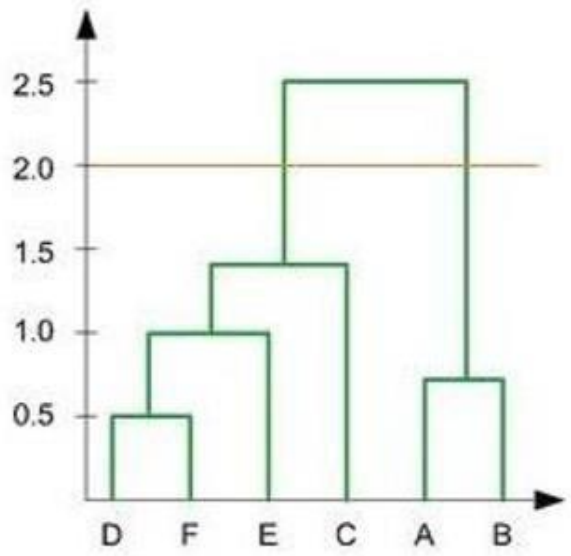
Post Lab Questions

1. What is true about K-mean clustering
   - I. K-means is extremely sensitive to cluster center initializations
   - II. Bad Initialization can lead to poor convergence speed
   - III. Bad initialization can lead to bad overall clustering

   a) I and II
   b) I and III
   c) all of the above
   d) II and III

Ans. c) all of the above

2. **In the figure below, if you draw a horizontal line on y-axis for y=2. What will be the number of clusters formed?**



a. 1
b. 2
c. 3
d. 4

Ans. b.2

3. Which of the following is not true about the DBSCAN algorithm?
   1. It is a density based clustering algorithm
   2. It requires two parameters MinPts and epsilon
   3. The number of clusters need to be specified in advance
   4. It can produce non-convex shaped clusters

Ans. 4. It can produce non-convex shaped clusters

4. Distance between two clusters in complete linkage clustering is defined as:

   a) Distance between the closest pair of points between the clusters
   b) Distance between the furthest pair of points between the clusters
   c) Distance between the most centrally located pair of points in the clusters
   d) None of the above

Ans. b) Distance between the furthest pair of points between the clusters

5. Consider a set of five 2-dimensional points p1=(0, 0), p2=(0, 1), p3=(5, 8), p4=(5, 7), and p5=(0, 0.5). Euclidean distance is the distance function. Single linkage clustering is used to cluster the points into two clusters. The clusters are

a) {p1, p2, p3} {p4, p5}
b) {p1, p4, p5} {p2, p3}
c) {p1, p2, p5} {p3, p4}
d) {p1, p2, p4} {p3, p5}

Ans. b). {p1, p2, p5} {p3, p4}