

## Implement clustering techniques – Hierarchical and K-Means

### AIM:

To Implement clustering techniques Hierarchical and K-Means using R programming in R Studio.

### a) HIERARCHIAL CLUSTERING

```
# Load the iris dataset data(iris)

# Use only the numeric columns for clustering (exclude the Species column) iris_data <-
iris[, -5]
# Standardize the data iris_scaled
<- scale(iris_data)

# Compute the distance matrix distance_matrix <- dist(iris_scaled, method
= "euclidean")

# Perform hierarchical clustering using the "complete" linkage method hc_complete <-
hclust(distance_matrix, method = "complete")

# Plot the dendrogram plot(hc_complete, main = "Hierarchical Clustering Dendrogram", xlab
= "", sub = "", cex =
0.6)

# Cut the tree to form 3 clusters clusters
<- cutree(hc_complete, k = 3)

# Print the cluster memberships print(clusters)

# Add the clusters to the original dataset iris$Cluster
<- as.factor(clusters)

# Display the first few rows of the updated dataset head(iris)
```

**OUTPUT:**

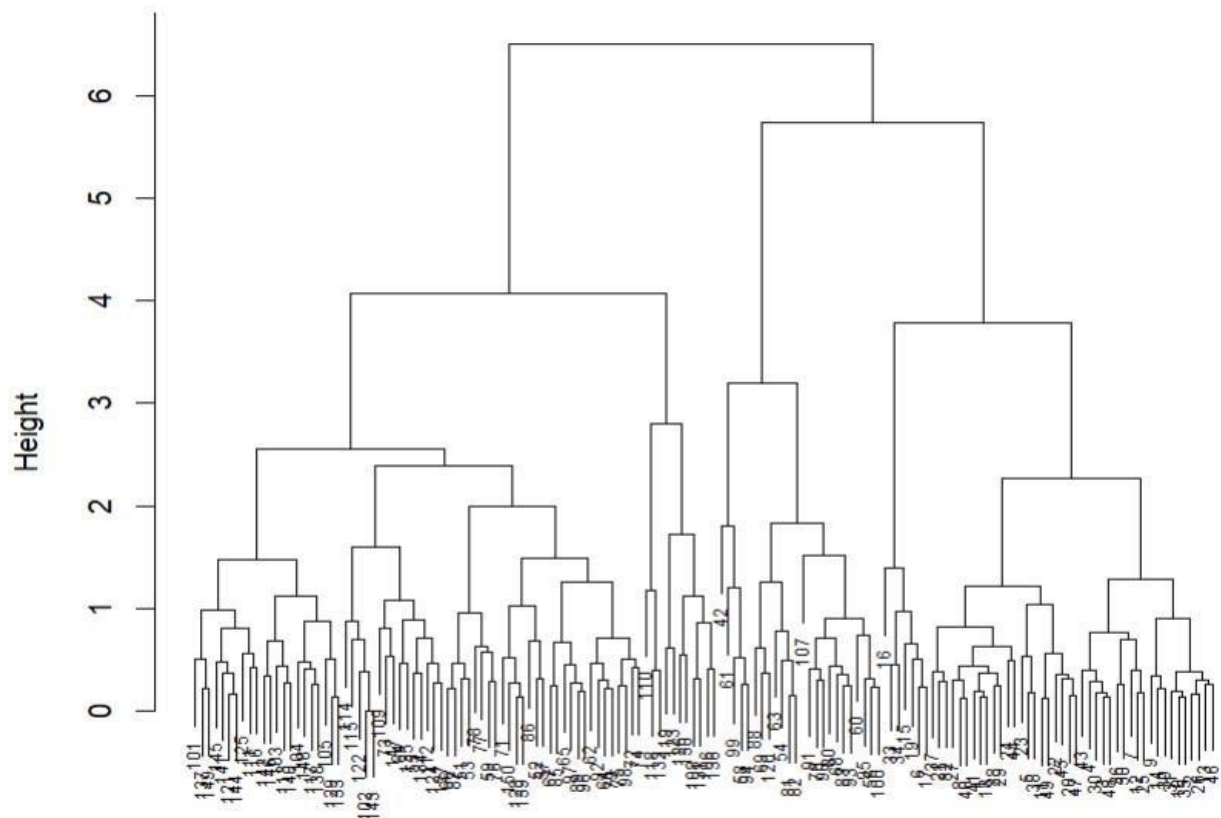
```

1 # Load the iris dataset
2 data(iris)
3 # Use only the numeric columns for clustering (exclude the Species column)
4 iris_data <- iris[, -5]
5 # Standardize the data
6 iris_scaled <- scale(iris_data)
7 # Compute the distance matrix
8 distance_matrix <- dist(iris_scaled, method = "euclidean")
9 # Perform hierarchical clustering using the "complete" linkage method
10 hc_complete <- hclust(distance_matrix, method = "complete")
11 # Plot the dendrogram
12 plot(hc_complete, main = "Hierarchical Clustering Dendrogram", xlab = "", sub = "", cex =
13      0.6)
14 # Cut the tree to form 3 clusters
15 clusters <- cutree(hc_complete, k = 3)
16 # Print the cluster memberships
17 print(clusters)
18 # Add the clusters to the original dataset
19 iris$Cluster <- as.factor(clusters)
20 # Display the first few rows of the updated dataset
21 head(iris)

```

Files Plots Packages Help Viewer Presentation

Zoom Export Publish

**Hierarchical Clustering Dendrogram**

**b) K-MEANS CLUSTERING**

```
# Load the iris dataset data(iris)

# Use only the numeric columns for clustering (exclude the Species column) iris_data <-
iris[, -5]
# Standardize the data  iris_scaled
<- scale(iris_data)

# Set the number of clusters set.seed(123)
# For reproducibility k <- 3
# Number of clusters

# Perform K-Means clustering
kmeans_result <- kmeans(iris_scaled, centers = k, nstart = 25)

# Print the K-Means result
print(kmeans_result)

# Print the cluster centers
print(kmeans_result$centers)

# Add the cluster assignments to the original dataset iris$Cluster <-
as.factor(kmeans_result$cluster)

# Display the first few rows of the updated dataset head(iris)

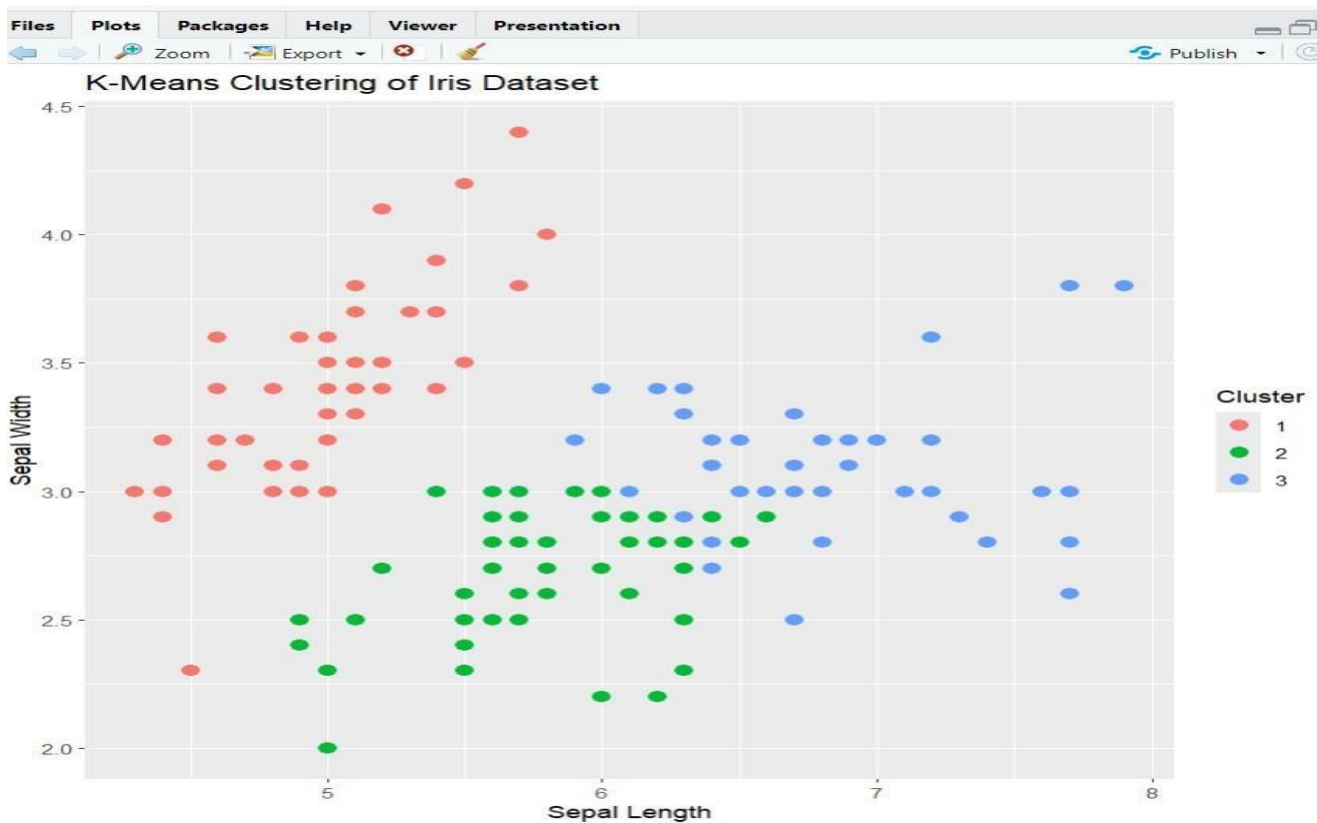
# Plot the clusters library(ggplot2) ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color =
Cluster)) +  geom_point(size = 3) +  labs(title = "K-Means Clustering of Iris Dataset", x = "Sepal
Length", y = "Sepal Width")
```

**OUTPUT:**

```

1 # Load the iris dataset
2 data(iris)
3 # Load the iris dataset
4 data(iris)
5 # Use only the numeric columns for clustering (exclude the Species column)
6 iris_data <- iris[, -5]
7 # Standardize the data
8 iris_scaled <- scale(iris_data)
9 # Set the number of clusters
10 set.seed(123) # For reproducibility
11 k <- 3 # Number of clusters
12 # Perform K-Means clustering
13 kmeans_result <- kmeans(iris_scaled, centers = k, nstart = 25)
14 # Print the K-Means result
15 print(kmeans_result)
16 # Print the cluster centers
17 print(kmeans_result$centers)
18 # Add the cluster assignments to the original dataset
19 iris$Cluster <- as.factor(kmeans_result$cluster)
20 # Display the first few rows of the updated dataset
21 head(iris)
22 # Plot the clusters
23 library(ggplot2)
24 ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color = Cluster)) +
25   geom_point(size = 3) +
26   labs(title = "K-Means Clustering of Iris Dataset", x = "Sepal Length", y = "Sepal width")
27

```

**RESULT:**

Thus, the Implement clustering techniques Hierarchical and K-Means using R programming in R Studio have been successfully executed.