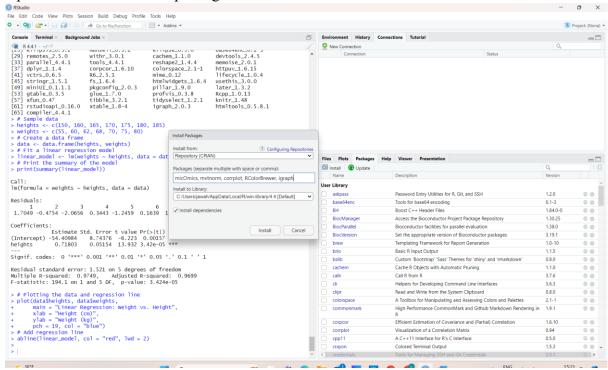
Ex 9 Implement clustering techniques – Hierarchical and K-Means

Aim:

To implement SVM/ Decision Tree classification technique in R Programming

PROCEDURE:

- 1. Install R for windows.
- 2. Install R Studio.
- 3. Open R Studio and install packages



Thus R studio is set up successfully.

a) HIERARCHIAL CLUSTERING

Program:

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)

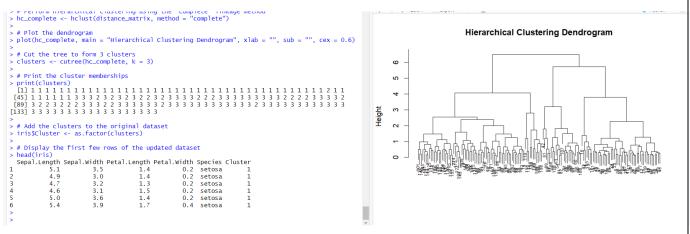
# Add the clusters to the original dataset

iris$Cluster <- as.factor(clusters)

# Display the first few rows of the updated dataset

head(iris)
```

Output:



b) K-MEANS CLUSTERING

Program

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)</pre>
# 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") +
 theme minimal() # Optional: makes the plot look cleaner
Output:
```

```
K-means clustering with 3 clusters of sizes 50, 53, 47
Cluster means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
  -1.01119138 0.85041372
                             -1.3006301 -1.2507035
0.3465767 0.2805873
   -0.05005221 -0.88042696
1.13217737 0.08812645
                                          0.2805873
                              0.9928284
                                          1.0141287
Clustering vector:
  [133] 3 2 2 3 3 3 2 3 3 3 2 3 3 3 2 3 3 2
Within cluster sum of squares by cluster: [1] 47.35062 44.08754 47.45019 (between_SS / total_SS = 76.7 %)
Available components:
[1] "cluster"
[7] "size"
                   "centers"
                                  "totss"
                                                 "withinss"
                                                                "tot.withinss" "betweenss"
                                  "ifault"
> # Print the cluster centers
> print(kmeans_result$centers)
  Sepal.Length Sepal.Width Petal.Length Petal.Width
                            -1.3006301 -1.2507035
0.3465767 0.2805873
  -1.01119138 0.85041372
  -0.05005221 -0.88042696
1.13217737 0.08812645
                                          0.2805873
                              0.9928284
                                          1.0141287
  # Add the cluster assignments to the original dataset
  iris$Cluster <- as.factor(kmeans_result$cluster)</pre>
  # Display the first few rows of the updated dataset
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species Cluster
           5.1
4.9
                       3.5
3.0
                                                0.2 setosa
0.2 setosa
                                    1.4
                                    1.4
                                                                   1
                                                0.2
3
           4.7
                       3.2
                                    1.3
                                                    setosa
                                                                   1
           4.6
                       3.1
                                    1.5
                                                0.2
4
                                                    setosa
                                                                   1
                                    1.4
                                                0.2
           5.0
                       3.6
                                                     setosa
6
                                                0.4 setosa
      Plots Packages Help Viewer Presentation
                                                                                       🔑 Zoom 🛮 🔁 Export 🕶 🚨 🥑

◆ Publish 
▼ | ②
     K-Means Clustering of Iris Dataset
  4.5
  4.0
  3.5
Sepal Width
                                                                                     Cluster
                                                                                      1
                                                                                        2
                                                                                        3
```

Result:

2.5

2.0

Thus the k-means clustering and hierarchical clustering is implemented successfully using R Programming

Sepal Length

8