

**Ex no: 9**

## **Implement clustering techniques – Hierarchical and K-Means**

**Aim:**

To implement clustering techniques – Hierarchical and K-Means.

**Procedure:**

**a) Hierarchical Clustering:**

1. Load the iris dataset into the environment.
2. Exclude the non-numeric Species column and use only the numeric columns for clustering.
3. Standardize the numeric data using the `'scale()'` function to ensure all variables have the same scale.
4. Compute the distance matrix using Euclidean distance with the `'dist()'` function.
5. Perform hierarchical clustering on the distance matrix using the complete linkage method via `'hclust()'`.
6. Plot the dendrogram of the hierarchical clustering to visualize the clustering structure.
7. Cut the dendrogram tree into 3 clusters using `'cutree()'` based on the number of desired clusters.
8. Print the cluster memberships of each data point to see which cluster they belong to.
9. Add the cluster assignments as a new column in the original iris dataset.
10. Display the first few rows of the updated dataset to verify the clusters have been added.

**b) K-Means Clustering:**

1. Load the iris dataset into the environment.
2. Exclude the non-numeric Species column, using only the numeric columns for clustering.
3. Standardize the numeric data using the `'scale()'` function to ensure all features are on the same scale.
4. Set the number of clusters and a seed for reproducibility.
5. Perform K-Means clustering using the `'kmeans()'` function with the predefined number of clusters and multiple random starts.

6. Print the K-Means clustering results, including cluster assignments and within-cluster sum of squares.
7. Print the cluster centers to examine the centroids of each cluster.
8. Add the cluster assignments to the original iris dataset as a new column.
9. Display the first few rows of the updated dataset to verify cluster assignments.
10. Visualize the clusters using ggplot2, plotting Sepal Length against Sepal Width with points colored by cluster membership.

**Program:**

**a) Hierarchical 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)
```

**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:**

**a) Hierarchical Clustering:**

[illegible]



### b) K-Means Clustering:

[illegible]

```

Sepal.Length Sepal.Width Petal.Length Petal.Width Species Cluster
1           5.1           3.5           1.4           0.2   setosa       1
2           4.9           3.0           1.4           0.2   setosa       1
3           4.7           3.2           1.3           0.2   setosa       1
4           4.6           3.1           1.5           0.2   setosa       1
5           5.0           3.6           1.4           0.2   setosa       1
6           5.4           3.9           1.7           0.4   setosa       1

> # 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 Data ..." ... [TRUNCATED]
> 
```

Environment	History	Connections	Tutorial
	Import Dataset ▾	224 MiB ▾	
R ▾	Global Environment ▾		
Data			
▶ data	7 obs. of 2 variables		
▶ hc_complete	List of 7		
▶ iris	150 obs. of 6 variables		
▶ iris_data	150 obs. of 4 variables		
iris_scaled	num [1:150, 1:4] -0.898 -1.139 -1.381 -1.50...		
▶ kmeans_result	List of 9		
▶ linear_model	List of 12		
▶ logistic_model	List of 30		
▶ mtcars	32 obs. of 11 variables		
▶ svm_model	List of 31		
▶ test_data	45 obs. of 5 variables		
▶ train_data	105 obs. of 5 variables		
▶ tree_model	List of 14		
Values			
accuracy	0.9777777777777778		
clusters	int [1:150] 1 1 1 1 1 1 1 1 1 1 ...		
confusion_matrix	'table' int [1:3, 1:3] 14 0 0 0 18 0 0 1 12		
distance_matrix	'dist' num [1:11175] 1.172 0.843 1.1 0.259 1...		
heights	num [1:7] 150 160 165 170 175 180 185		
k	3		
predicted_probs	Named num [1:32] 0.461 0.461 0.598 0.492 0.2...		
predictions	Factor w/ 3 levels "setosa","versicolor",...:		
sample_indices	int [1:105] 14 50 118 43 150 148 90 91 143 9...		
weights	num [1:7] 55 60 62 68 70 75 80		



**Result:**

Thus the implementation of clustering techniques – Hierarchical and K-Means using R programming has been executed successfully.