

Exp.No: 9**IMPLEMENT CLUSTERING TECHNIQUES – HIERARCHICAL AND KMEANS****AIM:**

To write an R code to implement hierarchical and k-means clustering techniques.

PROCEDURE:

1. Load the iris dataset and use only the numeric columns for clustering by excluding the Species column.
2. Standardize the data to ensure all variables have equal weight in the clustering process.
3. Compute the distance matrix using the Euclidean method and perform hierarchical clustering using the "complete" linkage method, plot the dendrogram, and cut the tree to form 3 clusters.
4. Perform K-means clustering by setting the number of clusters, run the clustering algorithm, and add cluster assignments to the original dataset.
5. Display the first few rows of the updated dataset and plot the clusters using ggplot2 for visualization.

PROGRAM CODE:**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")

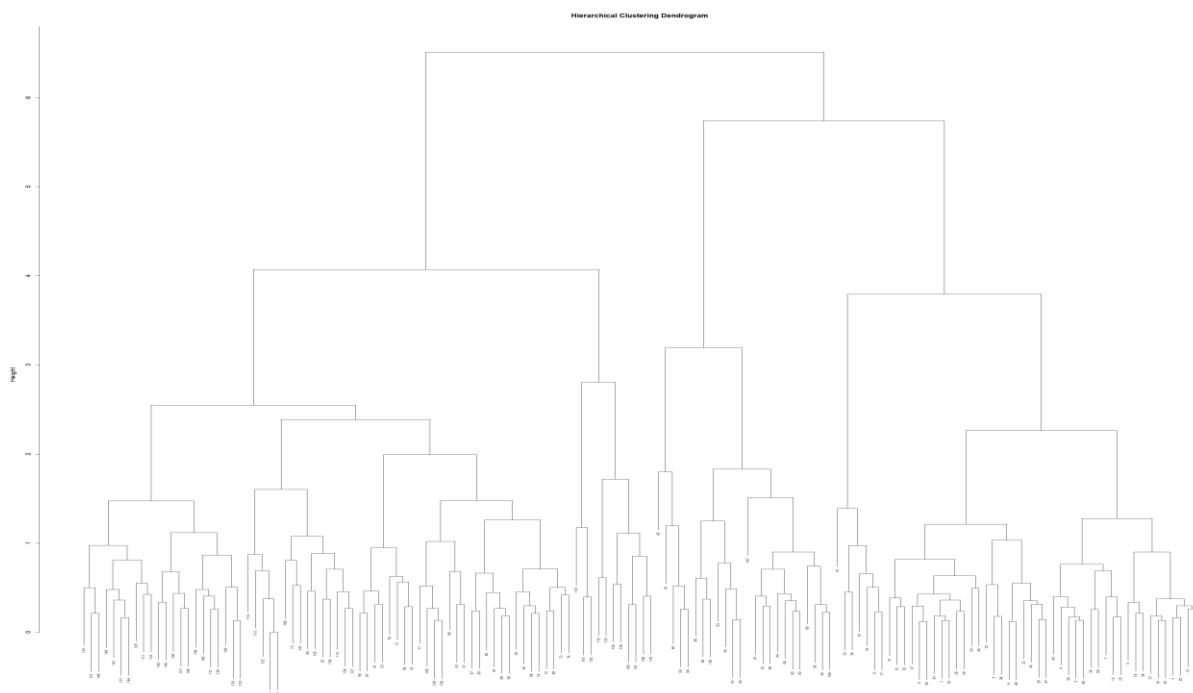
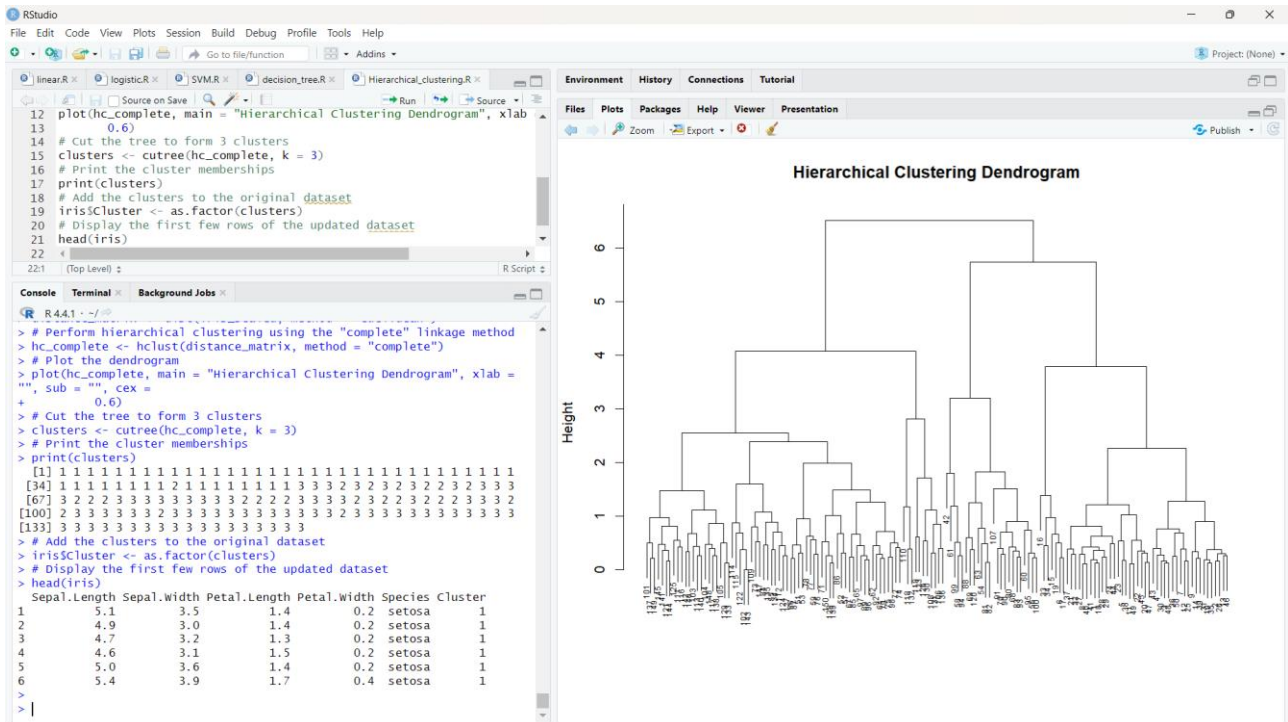
# Perform hierarchical clustering using the "complete" linkage method
hc_complete <- hclust(distance_matrix, method = "complete")

# 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:



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")
```

Reg.No: 210701266

OUTPUT:

The screenshot shows the RStudio interface with the following content:

Script Editor:

```

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 # Set the number of clusters
8 k <- 3
9 set.seed(123) # For reproducibility
10 kmeans_result <- kmeans(iris_scaled, k)
11
12 # Print the cluster centers
13 print(kmeans_result$centers)
14
15 # Add the cluster assignments to the original dataset
16 iris$Cluster <- as.factor(kmeans_result$cluster)
17 # Display the first few rows of the updated dataset
18 head(iris)
19
20 # Plot the clusters
21 library(ggplot2)
22 ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color = Cluster)) +
23   geom_point(size = 3) +
24   labs(title = "K-Means Clustering of Iris Dataset", x = "Sepal Length",
25        y = "Sepal Width")
26 
```

Console Output:

```

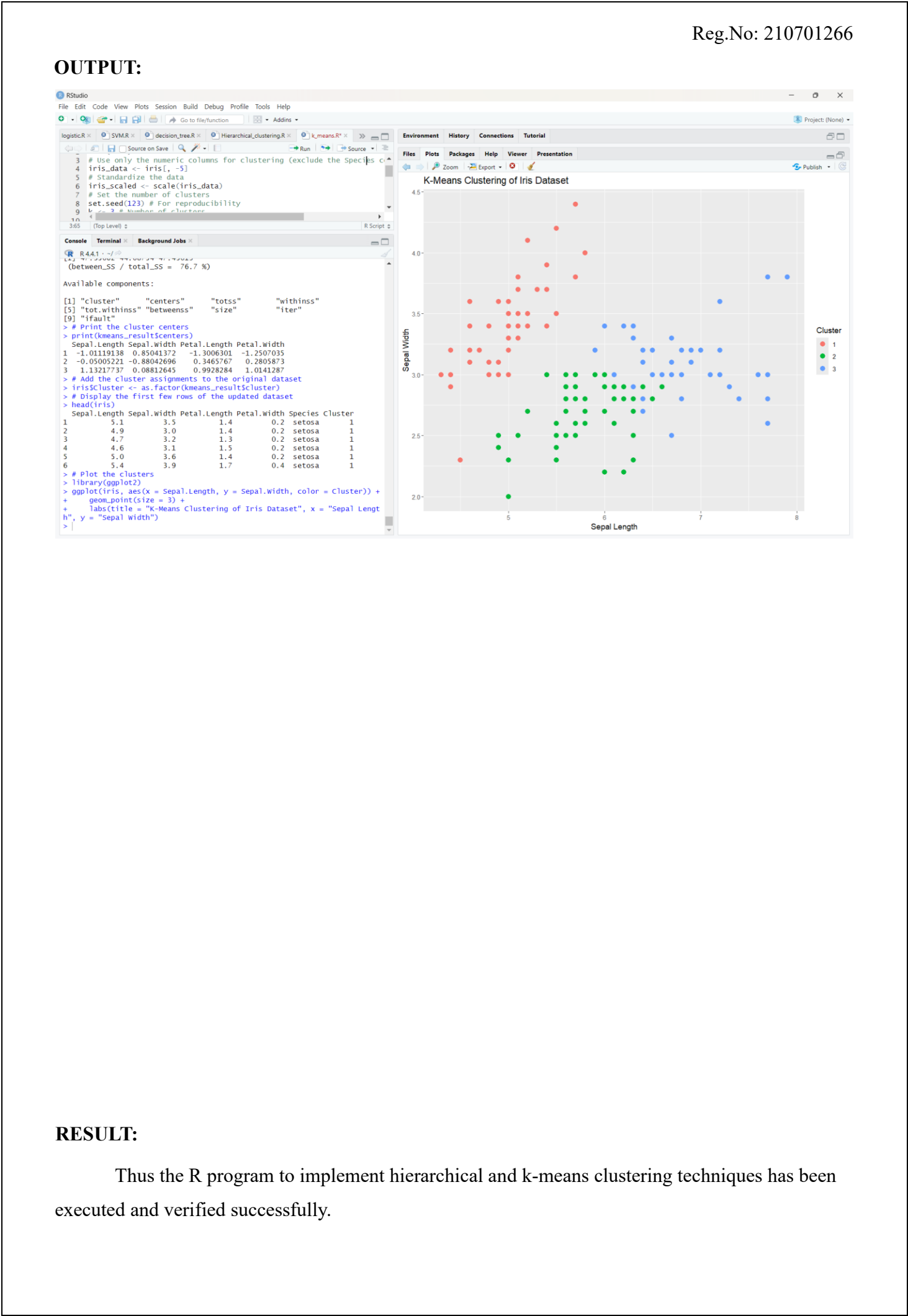
R441 ~ /
> kmeans_result$centers
(between_SS / total_SS = 76.7 %)

Available components:
[1] "cluster" "centers" "totss" "withinss"
[5] "tot.withinss" "betweenss" "size" "iter"
[9] "ifault"
> # Print the cluster centers
> print(kmeans_result$centers)
Sepal.Length Sepal.Width Petal.Length Petal.Width
1 -1.01119138 0.85041372 -1.3006301 -1.2507035
2 -0.05005221 -0.88042696 0.3465767 0.2805873
3 1.13217737 0.08812645 0.9928284 1.0141287
> # Add the cluster assignments to the original dataset
> iris$Cluster <- as.factor(kmeans_result$cluster)
> # Display the first few rows of the updated dataset
> head(iris)
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
27 
```

Plot: A scatter plot titled "K-Means Clustering of Iris Dataset" showing Sepal.Length on the x-axis (ranging from 4 to 8) and Sepal.Width on the y-axis (ranging from 2.0 to 4.5). The data points are colored by cluster: red for Cluster 1, green for Cluster 2, and blue for Cluster 3. The plot shows three distinct clusters of points.

RESULT:

Thus the R program to implement hierarchical and k-means clustering techniques has been executed and verified successfully.



Reg.No: 210701266

OUTPUT:

The screenshot shows the RStudio interface with the following content:

Script Editor:

```

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 # Set the number of clusters
8 k <- 3
9 set.seed(123) # For reproducibility
10 kmeans_result <- kmeans(iris_scaled, k)
11
12 # Print the cluster centers
13 print(kmeans_result$centers)
14
15 # Add the cluster assignments to the original dataset
16 iris$Cluster <- as.factor(kmeans_result$cluster)
17 # Display the first few rows of the updated dataset
18 head(iris)
19
20 # Plot the clusters
21 library(ggplot2)
22 ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color = Cluster)) +
23   geom_point(size = 3) +
24   labs(title = "K-Means Clustering of Iris Dataset", x = "Sepal Length",
25        y = "Sepal Width")
26

```

Console Output:

```

R44.1 ~ /
> kmeans_result$centers
(between_SS / total_SS = 76.7 %)

Available components:
[1] "cluster" "centers" "totss" "withinss"
[5] "tot.withinss" "betweenss" "size" "iter"
[9] "ifault"
> # Print the cluster centers
> print(kmeans_result$centers)
Sepal.Length Sepal.Width Petal.Length Petal.Width
1 -1.01119138 0.85041372 -1.3006301 -1.2507035
2 -0.05005221 -0.88042696 0.3465767 0.2805873
3 1.13217737 0.08812645 0.9928284 1.0141287
> # Add the cluster assignments to the original dataset
> iris$Cluster <- as.factor(kmeans_result$cluster)
> # Display the first few rows of the updated dataset
> head(iris)
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 Dataset", x = "Sepal Length",
+        y = "Sepal Width")
>

```

Plot: A scatter plot titled "K-Means Clustering of Iris Dataset". The x-axis is labeled "Sepal Length" and ranges from 4 to 8. The y-axis is labeled "Sepal Width" and ranges from 2.0 to 4.5. The plot shows three distinct clusters of data points, colored red (Cluster 1), green (Cluster 2), and blue (Cluster 3). The red cluster is located in the upper-left region, the green cluster is in the lower-left region, and the blue cluster is in the upper-right region.

RESULT:

Thus the R program to implement hierarchical and k-means clustering techniques has been executed and verified successfully.

Reg.No: 210701266

OUTPUT:

The screenshot shows the RStudio interface with the following content:

Script Editor:

```

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 # Set the number of clusters
8 k <- 3
9 set.seed(123) # For reproducibility
10 kmeans_result <- kmeans(iris_scaled, k)
11
12 # Print the cluster centers
13 print(kmeans_result$centers)
14
15 # Add the cluster assignments to the original dataset
16 iris$Cluster <- as.factor(kmeans_result$cluster)
17 # Display the first few rows of the updated dataset
18 head(iris)
19
20 # Plot the clusters
21 library(ggplot2)
22 ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color = Cluster)) +
23   geom_point(size = 3) +
24   labs(title = "K-Means Clustering of Iris Dataset", x = "Sepal Length",
25        y = "Sepal Width")
26

```

Console Output:

```

R44.1 ~ /
> kmeans_result$centers
(between_SS / total_SS = 76.7 %)

Available components:
[1] "cluster" "centers" "totss" "withinss"
[5] "tot.withinss" "betweenss" "size" "iter"
[9] "ifault"
> # Print the cluster centers
> print(kmeans_result$centers)
Sepal.Length Sepal.Width Petal.Length Petal.Width
1 -1.01119138 0.85041372 -1.3006301 -1.2507035
2 -0.05005221 -0.88042696 0.3465767 0.2805873
3 1.13217737 0.08812645 0.9928284 1.0141287
> # Add the cluster assignments to the original dataset
> iris$Cluster <- as.factor(kmeans_result$cluster)
> # Display the first few rows of the updated dataset
> head(iris)
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 Dataset", x = "Sepal Length",
+        y = "Sepal Width")
>

```

Plot: A scatter plot titled "K-Means Clustering of Iris Dataset". The x-axis is labeled "Sepal Length" and ranges from 4 to 8. The y-axis is labeled "Sepal Width" and ranges from 2.0 to 4.5. The plot shows three distinct clusters of data points, colored red (Cluster 1), green (Cluster 2), and blue (Cluster 3). The red cluster is located in the upper-left region, the green cluster is in the lower-left region, and the blue cluster is in the upper-right region.

RESULT:

Thus the R program to implement hierarchical and k-means clustering techniques has been executed and verified successfully.