Name: **Varun Sudhir**

Reg No: **21BDS0040**

We will use the IRIS dataset and perform K-means on it

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
1	5.1	3.5	1.4	0.2
2	4.9	3.0	1.4	0.2
3	4.7	3.2	1.3	0.2
4	4.6	3.1	1.5	0.2
5	5.0	3.6	1.4	0.2
6	5.4	3.9	1.7	0.4
7	4.6	3.4	1.4	0.3
8	5.0	3.4	1.5	0.2
9	4.4	2.9	1.4	0.2
10	4.9	3.1	1.5	0.1
11	5.4	3.7	1.5	0.2
12	4.8	3.4	1.6	0.2
13	4.8	3.0	1.4	0.1
14	4.3	3.0	1.1	0.1
15	5.8	4.0	1.2	0.2
16	5.7	4.4	1.5	0.4

Code:

```
# Varun Sudhir 21BDS0040

euclidean_distance <- function(point1, point2) {
  sqrt(sum((point1 - point2)^2))
}

assign_clusters <- function(data, centroids) {
  clusters <- sapply(1:nrow(data), function(i) {
    distances <- sapply(1:nrow(centroids), function(j)
euclidean_distance(data[i,], centroids[j,]))
    return(which.min(distances))
  })
  return(clusters)
}

update_centroids <- function(data, clusters, k) {
```

```r
  new_centroids <- matrix(NA, nrow = k, ncol = ncol(data))
  for (i in 1:k) {
    cluster_points <- data[clusters == i, ]
    if (nrow(cluster_points) > 0) {
      new_centroids[i, ] <- colMeans(cluster_points)
    } else {
      new_centroids[i, ] <- data[sample(1:nrow(data), 1), ]
    }
  }
  return(new_centroids)
}

# K-means function
k_means <- function(data, k, max_iter = 100) {
  centroids <- data[sample(1:nrow(data), k), ]

  for (i in 1:max_iter) {
    clusters <- assign_clusters(data, centroids)
    new_centroids <- update_centroids(data, clusters, k)
    if (all(centroids == new_centroids)) {
      cat("Convergence reached at iteration", i, "\n")
      break
    }
    centroids <- new_centroids
  }
  list(clusters = clusters, centroids = centroids)
}

display_clusters <- function(data, clusters, centroids) {
  k <- nrow(centroids)

  for (i in 1:k) {
    cat("\nCluster", i, "\n")
    cluster_points <- data[clusters == i, ]
    cat("Centroid:", centroids[i, ], "\n")
    cat("Number of points:", nrow(cluster_points), "\n")
    cat("Variance of features:\n")
    print(apply(cluster_points, 2, var))
    cat("Minimum values of features:\n")
    print(apply(cluster_points, 2, min))

    cat("Maximum values of features:\n")
    print(apply(cluster_points, 2, max))
    distances <- apply(cluster_points, 1, function(point)
euclidean_distance(point, centroids[i, ]))
    cat("Average distance from centroid:", mean(distances), "\n")
  }
}
```

```r
# Function to plot clusters and centroids
plot_clusters <- function(data, clusters, centroids) {
  data_df <- as.data.frame(data)
  data_df$cluster <- as.factor(clusters)
  colnames(data_df)[1:2] <- c("Feature1", "Feature2")
  centroid_df <- as.data.frame(centroids[, 1:2])
  centroid_df$Cluster <- as.factor(1:nrow(centroid_df))
  colnames(centroid_df)[1:2] <- c("Feature1", "Feature2")

  ggplot(data_df, aes(x = Feature1, y = Feature2, color = cluster)) +
    geom_point(size = 2) +
    geom_point(data = centroid_df, aes(x = Feature1, y = Feature2), color =
"black", shape = 4, size = 5) +
    labs(title = "K-Means Clustering", x = "Feature 1", y = "Feature 2") +
    theme_minimal()
}

run_k_means <- function() {
  # iris dataset (4 features: Sepal.Length, Sepal.Width, Petal.Length,
Petal.Width)
  data <- as.matrix(iris[, 1:4])
  data <- scale(data)

  cat("Enter the number of clusters (k): ")
  k <- as.integer(readline())
  result <- k_means(data, k)
  display_clusters(data, result$clusters, result$centroids)
  plot_clusters(data[, 1:2], result$clusters, result$centroids[, 1:2])
}
run_k_means()
cat("Varun Sudhir 21BDS0040")
```

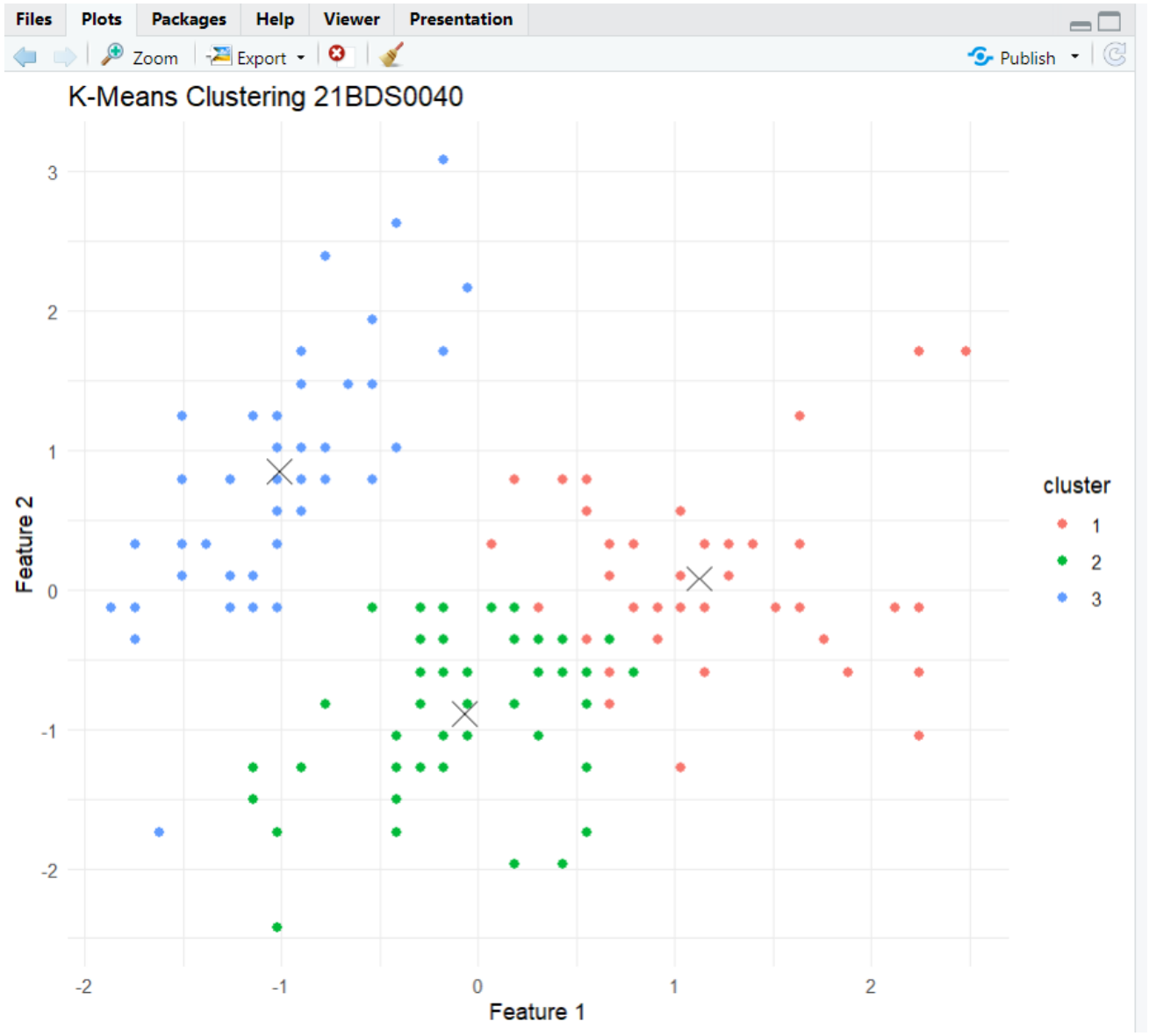
Output:

```
> run_k_means()
Enter the number of clusters (k):
3
Convergence reached at iteration 4

Cluster 1
Centroid: 1.127627 0.07877034 0.9820815 0.9957524
Number of points: 48
Variance of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
   0.3445740    0.3580082    0.1333879    0.2005569
Minimum values of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
  0.06843254  -1.27867961   0.36367793   0.13206729
Maximum values of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
    2.483699     1.703886     1.779869     1.706379
Average distance from centroid: 0.9139393

Cluster 2
Centroid: -0.06858736 -0.8904166 0.3440691 0.2834435
Number of points: 52
Variance of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
   0.2259220    0.3417057    0.1024204    0.1696749
Minimum values of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
  -1.1392005   -2.4258204   -0.4293892   -0.2615107
Maximum values of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
   0.7930124   -0.1315388    1.0434497    1.5751867
Average distance from centroid: 0.8507722


Cluster 3
Centroid: -1.011191 0.8504137 -1.30063 -1.250704
Number of points: 50
Variance of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
 0.181201918  0.756344014  0.009677951  0.019115323
Minimum values of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
   -1.863780    -1.737536    -1.562342    -1.442245
Maximum values of features:
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
 -0.05233076   3.08045544  -1.05251337  -0.78628144
Average distance from centroid: 0.8109485
>
```


Files
Plots
Packages
Help
Viewer
Presentation
Zoom
Export
Publish
K-Means Clustering 21BDS0040
Feature 2
Feature 1
cluster
1
2
3
3
2
1
0
-1
-2
-2
-1
0
1
2