Experiment 2.3

1. Aim:
To perform the cluster analysis by k-means method using R.

2. Objective:

- To identify natural groupings or clusters within a dataset using the k-means clustering algorithm in R

- To apply the k-means clustering algorithm in R to a dataset with a known number of clusters, and to evaluate the effectiveness of the clustering method

3. Script:

K Means Clustering in R Programming is an Unsupervised Non-linear algorithm that cluster data based on similarity or similar groups. It seeks to partition the observations into a pre-specified number of clusters. Segmentation of data takes place to assign each training example to a segment called a cluster. In the unsupervised algorithm, high reliance on raw data is given with large expenditure on manual review for review of relevance is given. It is used in a variety of fields like Banking, healthcare, retail, Media, etc.

K-Means clustering groups the data on similar groups. The algorithm is as follows:

- Choose the number **K** clusters.
- Select at random K points, the centroids (Not necessarily from the given data).
- Assign each data point to closest centroid that forms K clusters.
- Compute and place the new centroid of each centroid.
- After final reassignment, name the cluster as Final cluster.

4. Code:

```
# Installing required packages
# ClusterR is an R package for cluster analysis and provides functions for k-means clustering, hierarchical clustering, and more.

install.packages("ClusterR")

#The cluster package is an R package for cluster analysis, including functions for k-means clustering, hierarchical clustering, and other algorithms.

install.packages("cluster")

# Loading packages
library(ClusterR) # Load ClusterR library
library(cluster) # Load cluster library

# Loading Seatbelts dataset
data(Seatbelts)
```

```
# Removing rows with missing values
Seatbelts_1 <- na.omit(Seatbelts[, -1]) # Remove rows with missing values in Seatbelts dataset

# Fitting K-Means clustering Model to training dataset
set.seed(240) # Setting seed for reproducibility
kmeans.re <- kmeans(Seatbelts, centers = 3, nstart = 20) # Fit k-means clustering model to Seatbelts dataset with 3 clusters and 20 starts

# Cluster identification for each observation
kmeans.re$cluster  # Display the cluster identification for each observation

# Creating a confusion matrix
cm <- table(Seatbelts$front, kmeans.re$cluster) # Create a confusion matrix of Seatbelts dataset and k-means clustering result
cm      # Display the confusion matrix# Model Evaluation and visualization

# Plot drivers vs front for Seatbelts dataset
plot(Seatbelts[, c("drivers", "front")], ylim = c(0, max(Seatbelts_1$front)))

# Plot drivers vs front for Seatbelts dataset with cluster colors
plot(Seatbelts[, c("drivers", "front")], col = kmeans.re$cluster, ylim = c(0, max(Seatbelts$front)))

# Plot drivers vs front for Seatbelts dataset with cluster colors and main title
plot(Seatbelts[, c("drivers", "front")], col = kmeans.re$cluster, main = "K-means with 3 clusters", ylim = c(0, max(Seatbelts_1$front)))


# Plotting cluster centers
# Display the cluster centers
kmeans.re$centers

# Display the cluster centers for drivers and front features
kmeans.re$centers[, c("drivers", "front")]

# Plot the cluster centers with different colors, shapes and size
points(kmeans.re$centers[, c("drivers", "front")], col = 1:3, pch = 8, cex = 3)

# Visualizing clusters
y_kmeans <- kmeans.re$cluster     # Assign the cluster identification to y_kmeans variable
clusplot(Seatbelts[, c("drivers", "front")],     # Plot a cluster plot of drivers vs front for Seatbelts dataset

         y_kmeans,
         lines  =  0,
         shade = TRUE,
         color = TRUE,
         labels = 2,
         plotchar = FALSE,
         span = TRUE,
         main = "Cluster Seatbelt",
         xlab = 'drivers',ylab = 'front')
```

5. Output:

1. Seatbelts dataset

```
> Seatbelts
         DriversKilled drivers front rear   kms PetrolPrice VanKilled law
Jan 1969           107    1687   867  269  9059  0.10297181        12   0
Feb 1969            97    1508   825  265  7685  0.10236300         6   0
Mar 1969           102    1507   806  319  9963  0.10206249        12   0
Apr 1969            87    1385   814  407 10955  0.10087330         8   0
May 1969           119    1632   991  454 11823  0.10101967        10   0
Jun 1969           106    1511   945  427 12391  0.10058119        13   0
Jul 1969           110    1559  1004  522 13460  0.10377398        11   0
Aug 1969           106    1630  1091  536 14055  0.10407640         6   0
Sep 1969           107    1579   958  405 12106  0.10377398        10   0
Oct 1969           134    1653   850  437 11372  0.10302640        16   0
Nov 1969           147    2152  1109  434  9834  0.10273011        13   0
Dec 1969           180    2148  1113  437  9267  0.10199719        14   0
Jan 1970           125    1752   925  316  9130  0.10127456        14   0
Feb 1970           134    1765   903  311  8933  0.10070398         6   0
Mar 1970           110    1717  1006  351 11000  0.10013961         8   0
Apr 1970           102    1558   892  362 10733  0.09862110        11   0
May 1970           103    1575   990  486 12912  0.09834929         7   0
Jun 1970           111    1520   866  429 12926  0.09808018        13   0
Jul 1970           120    1805  1095  551 13990  0.09727921        13   0
Aug 1970           129    1800  1204  646 14926  0.09741062        11   0
Sep 1970           122    1719  1029  456 12900  0.09742524        11   0
Oct 1970           183    2008  1147  475 12034  0.09638063        14   0
```

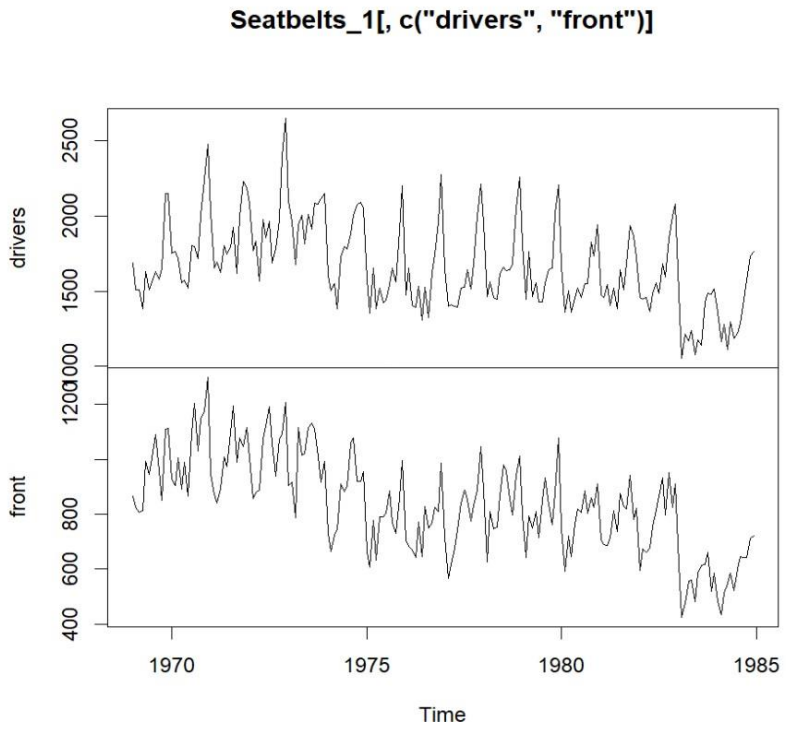
Clustering Identification:

```
> # Loading Seatbelts dataset
> data(Seatbelts)
> # Removing rows with missing values
> Seatbelts_1 <- na.omit(Seatbelts[, -1])
> # Fitting K-Means clustering Model to training dataset
> set.seed(240) # Setting seed for reproducibility
> kmeans.re <- kmeans(Seatbelts_1, centers = 3, nstart = 20)
> # Cluster identification for each observation
> kmeans.re$cluster
  [1] 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 1 2 1 1 1 2 2 2 2 2 2
 [40] 1 1 1 3 1 1 1 2 2 2 2 1 1 1 1 3 3 1 1 2 2 2 2 2 1 1 1 1 3 1 1 2 2 2 2 2 2 1 1
 [79] 1 3 1 1 2 2 2 2 1 1 1 1 3 3 1 1 2 2 2 2 2 1 1 1 3 3 3 1 1 1 1 2 2 1 1 3 3 3 3 3
[118] 1 1 1 2 2 1 1 3 3 3 3 3 3 1 1 1 1 1 3 3 3 3 3 3 3 1 1 1 1 3 3 3 3 3 3 3 3 1 1
[157] 1 1 3 3 3 3 3 3 3 3 3 3 1 1 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3
```

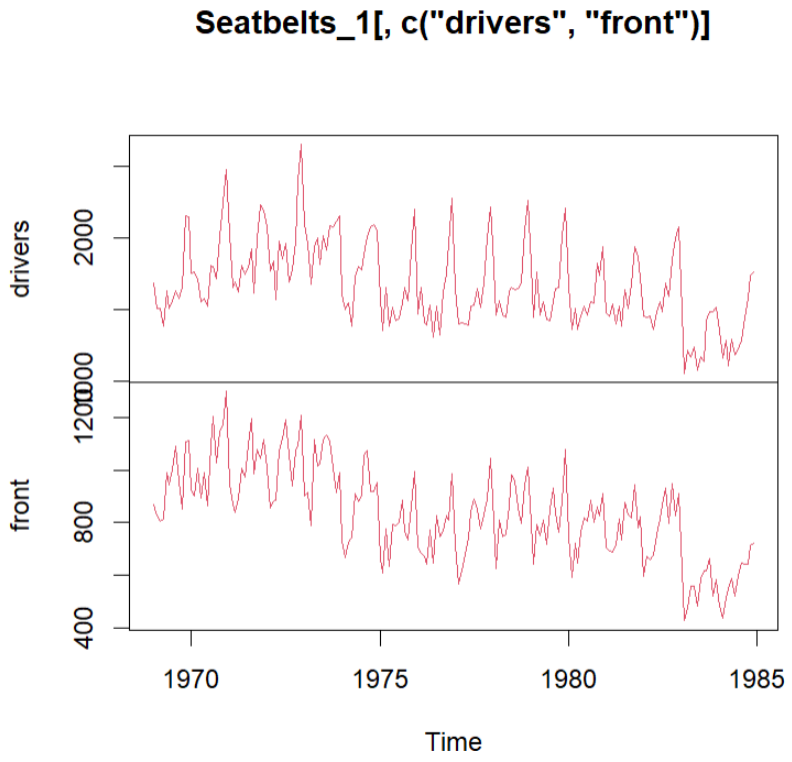
Confusion Matrix

```
> cm
function (x)
2.54 * x
<bytecode: 0x0000017e18d2d878>
<environment: namespace:grDevices>
```

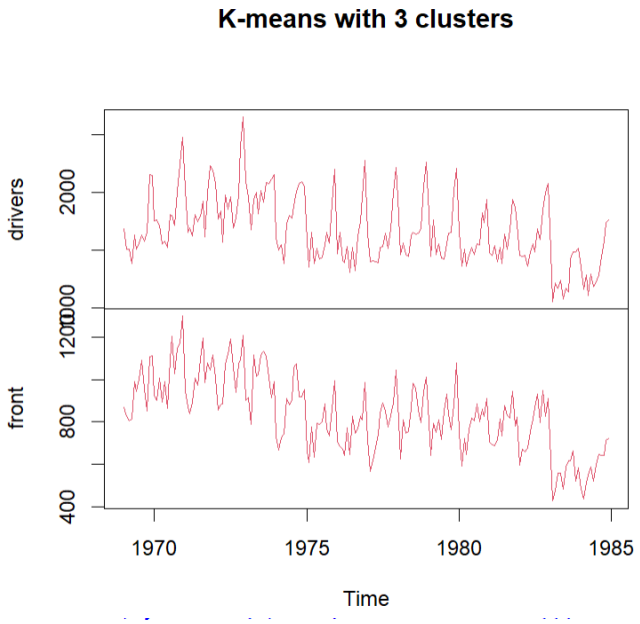
 # Plot drivers vs front for Seatbelts dataset



 # Plot drivers vs front for Seatbelts dataset with cluster colors



 # Plot drivers vs front for Seatbelts dataset with cluster colors and main title




```
> # Plotting cluster centers
> kmeans.re$centers
   drivers    front     rear      kms PetrolPrice VanKilled        law
1 1692.463 858.2388 401.2388 14798.22   0.1001847  9.059701 0.02985075
2 1815.172 896.1724 367.9483 11512.90   0.1003265 11.586207 0.00000000
3 1522.746 765.1642 429.9701 18202.13   0.1099178  6.865672 0.31343284
> kmeans.re$centers[, c("drivers", "front")]
   drivers    front
1 1692.463 858.2388
2 1815.172 896.1724
```

 # Plot a cluster plot of drivers vs front for Seatbelts dataset