

Experiment 2.3

Student Name: Yash Gupta
Branch: BE-CSE
Semester: 6
Subject Name: DM LAB

UID: 20BCS5009
Section/Group: 20BCS_DM-716 B
Date of Performance: 25/04/23
Subject Code: 20CSP_376

AIM :-

To perform the cluster analysis by K-mean method using R.

Theory :-

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.

Output :-

```
# Installing Packages
install.packages("ClusterR")
install.packages("cluster")

# Loading package
library(ClusterR)
library(cluster)

# Removing initial label of
# Species from original dataset
iris_1 <- iris[, -5]
```

```

# Fitting K-Means clustering Model
# to training dataset
set.seed(240) # Setting seed
kmeans.re <- kmeans(iris_1, centers = 3, nstart = 20)
kmeans.re
Cluster identification for
# each observation
kmeans.re$cluster
# Confusion Matrix
cm <- table(iris$Species, kmeans.re$cluster)
cm
# Model Evaluation and visualization
plot(iris_1[c("Sepal.Length", "Sepal.Width")])
plot(iris_1[c("Sepal.Length", "Sepal.Width")],
     col = kmeans.re$cluster)
plot(iris_1[c("Sepal.Length", "Sepal.Width")],
     col = kmeans.re$cluster,
     main = "K-means with 3 clusters")
## Plotting cluster centers
kmeans.re$centers
kmeans.re$centers[, c("Sepal.Length", "Sepal.Width")]
# cex is font size, pch is symbol
points(kmeans.re$centers[, c("Sepal.Length", "Sepal.Width")],
       col = 1:3, pch = 8, cex = 3)
## Visualizing clusters
y_kmeans <- kmeans.re$cluster
clusplot(iris_1[, c("Sepal.Length", "Sepal.Width")],
         y_kmeans,
         lines = 0,
         shade = TRUE,
         color = TRUE,
         labels = 2,
         plotchar = FALSE,
         span = TRUE,

```

```
main = paste("Cluster iris"),
xlab = 'Sepal.Length', ylab = 'Sepal.Width')
```

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Open file... Open URL... Open DB... Generate... Undo Edit... Save...

Filter: Choose **None** Apply Stop

Current relation: supermarket
Instances: 4627
Attributes: 217
Sum of weights: 4627

Attributes: All None Invert Pattern

No.	Name
1	department1
2	department2
3	department3
4	department4
5	department5
6	department6
7	department7
8	department8
9	department9
10	grocery misc
11	department11
12	baby needs
13	bread and cake
14	baking needs
15	coupons
16	juice-sat-cord-ms
17	tea
18	biscuits
19	canned fish-meat
20	canned fruit
21	canned vegetables
22	breakfast food
23	cigs-tobacco pkts
24	cigarette cartons

Remove

Status: OK

Selected attribute: Name: department1
Missing: 3580 (77%)
Distinct: 1
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	1	1047	1047

Class total (Nom): Visualize All

Log x0

Weka Explorer

Preprocess Classify **Cluster** Associate Select attributes

Clusterer

- weka
 - clusterers
 - Canopy
 - Cobweb
 - EM
 - FarthestFirst
 - FilteredClusterer
 - HierarchicalClusterer
 - MakeDensityBasedClusterer
 - SimpleKMeans**

weka.gui.GenericObjectEditor

weka.clusterers.SimpleKMeans

About: Cluster data using the k means algorithm. More Capabilities

canopyMaxNumCanopiesToHoldInMemory: 100

canopyMinimumCanopyDensity: 2.0

canopyPeriodicPruningRate: 10000

canopyT1: -1.25

canopyT2: -1.0

debug: False

displayStdDevs: False

distanceFunction: Choose **EuclideanDistance** -R first-l

doNotCheckCapabilities: False

dontReplaceMissingValues: False

fastDistanceCalc: False

initializationMethod: Random

maxIterations: 500

numClusters: 2

numExecutionSlots: 1

preserveInstancesOrder: False

reduceNumberOfDistanceCalcsViaCanopies: False

seed: 10

Open... Save... OK Cancel

Clusterer output

```
=== Run information ===
```

```

Scheme:      weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance"
Relation:    supermarket
Instances:    4627
Attributes:   217
              [list of attributes omitted]
Test mode:    split 66% train, remainder test

```

```
=== Clustering model (full training set) ===
```

```
kMeans
=====
```

```
Number of iterations: 2
Within cluster sum of squared errors: 0.0
```

Initial starting points (random):

[illegible]

Missing values globally replaced with mean/mode

Final cluster centroids:

	Cluster#		
Attribute	Full Data	0	1
	(4627.0)	(1679.0)	(2948.0)

department1	t	t	t
department2	t	t	t
department3	t	t	t
department4	t	t	t
department5	t	t	t
department6	t	t	t
department7	t	t	t

department210	t	t	t
department211	t	t	t
department212	t	t	t
department213	t	t	t
department214	t	t	t
department215	t	t	t
department216	t	t	t
total	low	low	high

Time taken to build model (percentage split) : 0.12 seconds

Clustered Instances

0	987 (63%)
1	587 (37%)

The screenshot shows the Weka Explorer window with the 'Visualize' tab selected. The 'Plot Matrix' is displayed, showing a table of data. The columns are labeled 'department1' through 'department9', 'grocery misc', 'department11', 'baby needs', and 'bread and cal'. The rows are labeled 'total' and 'total'. The matrix shows a grid of data points, with some cells containing a small red circle, indicating a specific data point or cluster.

	department1	department2	department3	department4	department5	department6	department7	department8	department9	grocery misc	department11	baby needs	bread and cal
total	○	○	○	○	○		○		○	○		○	○
total	○	○	○	○	○		○		○	○		○	○