Practical no 8

Aim: Demonstration of Clustering

Theory:

K-means Clustering Algo:

K-means is an unsupervised learning method for clustering data points. The algorithm iteratively divides data

points into K clusters by minimizing the variance in each cluster.

How does it work?

First, each data point is randomly assigned to one of the K clusters. Then, we compute the centroid

(functionally the center) of each cluster, and reassign each data point to the cluster with the closest centroid.

We repeat this process until the cluster assignments for each data point are no longer changing.

K-means clustering requires us to select K, the number of clusters we want to group the data into. The elbow

method lets us graph the inertia (a distance-based metric) and visualize the point at which it starts decreasing

linearly.

Steps:

- Open Excel create a data
- Save it as .CSV(MS-DOS)
- Keep the dataset and R code in a same folder.

Dataset:

	AGE	SPEND
2	18	10
3	20	25
4	22	30
5	24	10
6	26	25
7	28	30
8	30	80
9	32	14
10	34	45
11	36	78
12	38	45
13	40	56
14	42	5
15	44	56
16	46	56
17	48	0
18	50	55
19	52	89
20	54	55
21	56	56
22		

Code:

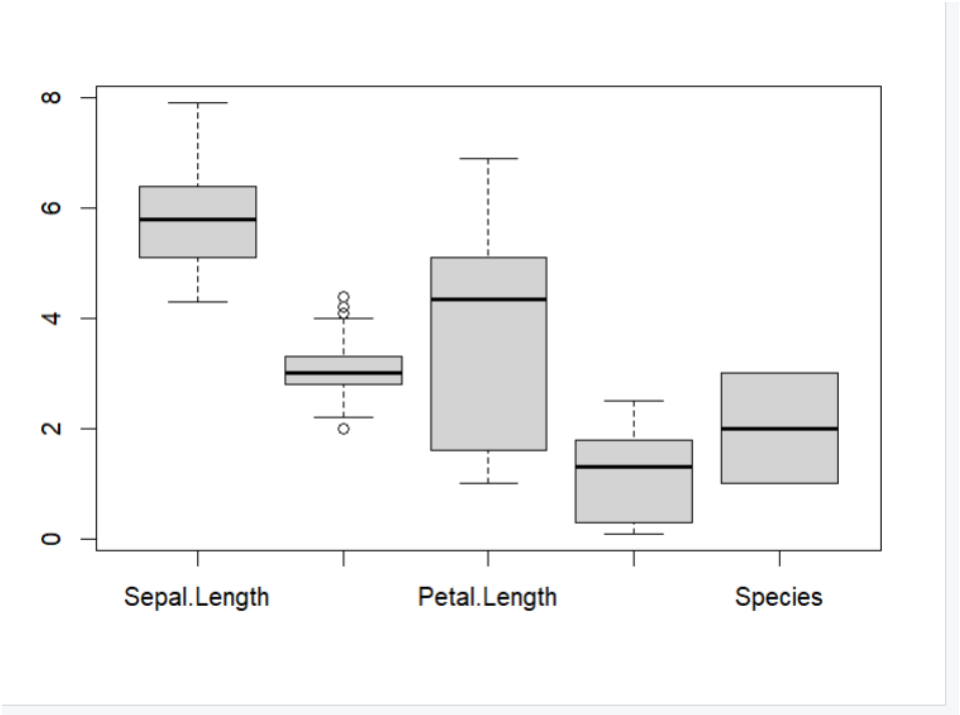
```
df=read.csv("C:/Users/admin/Downloads/Desktop/Materials/COMPUTER SCIENCE/Sem 6/Data Science/All Pracs/AGE.csv")
df
plot(df)
boxplot(df)
set.seed(20)
c1=kmeans(df[,1:2],3)
c1
iris
View(iris)
head(iris)
summary(iris)
plot(iris)
plot(iris[,3:4])
kmeansc1=kmeans(iris[,3:4],3)
kmeansc1
table(kmeansc1$cluster,iris$Species)
boxplot(iris)

```

Output:

```
> df
   AGE SPEND
1   18    10
2   20    25
3   22    30
4   24    10
5   26    25
6   28    30
7   30    80
8   32    14
9   34    45
10  36    78
11  38    45
12  40    56
13  42     5
14  44    56
15  46    56
16  48     0
17  50    55
18  52    89
19  54    55
20  56    56
> plot(df)
> boxplot(df)
> set.seed(20)
> c1=kmeans(df[,1:2],3)
> c1
K-means clustering with 3 clusters of sizes 2, 11, 7

Cluster means:
       AGE    SPEND
1 45.00000  2.50000
2 43.63636 61.00000
3 24.28571 20.57143

Clustering vector:
 [1] 3 3 3 3 3 3 2 3 2 2 2 2 1 2 2 1 2 2 2 2

Within cluster sum of squares by cluster:
[1]   30.5000 2876.5455  623.1429
 (between_SS / total_SS =  77.3 %)
```

```
2 43.63636 61.00000
3 24.28571 20.57143

Clustering vector:
 [1] 3 3 3 3 3 3 3 2 3 2 2 2 2 2 1 2 2 1 2 2 2 2

Within cluster sum of squares by cluster:
[1]   30.5000 2876.5455  623.1429
 (between_SS / total_SS =  77.3 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss" "betweenss"
[7] "size"         "iter"         "ifault"
> iris
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
1            5.1         3.5          1.4         0.2     setosa
2            4.9         3.0          1.4         0.2     setosa
3            4.7         3.2          1.3         0.2     setosa
4            4.6         3.1          1.5         0.2     setosa
5            5.0         3.6          1.4         0.2     setosa
6            5.4         3.9          1.7         0.4     setosa
7            4.6         3.4          1.4         0.3     setosa
8            5.0         3.4          1.5         0.2     setosa
9            4.4         2.9          1.4         0.2     setosa
10           4.9         3.1          1.5         0.1     setosa
11           5.4         3.7          1.5         0.2     setosa
12           4.8         3.4          1.6         0.2     setosa
13           4.8         3.0          1.4         0.1     setosa
14           4.3         3.0          1.1         0.1     setosa
15           5.8         4.0          1.2         0.2     setosa
16           5.7         4.4          1.5         0.4     setosa
17           5.4         3.9          1.3         0.4     setosa
18           5.1         3.5          1.4         0.3     setosa
19           5.7         3.8          1.7         0.3     setosa
20           5.1         3.8          1.5         0.3     setosa
21           5.4         3.4          1.7         0.2     setosa
22           5.1         3.7          1.5         0.4     setosa
23           4.6         3.6          1.0         0.2     setosa
24           5.1         3.3          1.7         0.5     setosa
25           4.8         3.4          1.9         0.2     setosa
26           5.0         3.0          1.6         0.2     setosa
27           5.0         3.4          1.6         0.4     setosa
28           5.2         3.5          1.5         0.2     setosa
```

```
6          5.4         3.9          1.7         0.4  setosa
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
> plot(iris)
> plot(iris[,3:4])
> kmeansc1=kmeans(iris[,3:4],3)
> kmeansc1
K-means clustering with 3 clusters of sizes 48, 50, 52

Cluster means:
  Petal.Length Petal.Width
1     5.595833    2.037500
2     1.462000    0.246000
3     4.269231    1.342308

Clustering vector:
  [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [50] 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3
 [99] 3 3 3 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 1
[148] 1 1 1

Within cluster sum of squares by cluster:
[1] 16.29167  2.02200 13.05769
 (between_SS / total_SS =  94.3 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss" "betweenss"
[7] "size"         "iter"         "ifault"
> table(kmeansc1$cluster,iris$Species)

    setosa versicolor virginica
  1      0          2        46
  2     50          0         0
  3      0         48         4
> boxplot(iris)
>
```



Conclusion: Hence we have successfully learnt and performed Clustering