

## **Data Science Final Project Report**

**Project Title** : Supermarket customer data segmentation

**Submitted By** : Rubaba Binte Rahman

**ID** : 20-42182-1 , **Section** - D

### **Project Description:**

I took a dataset of 200 individuals from a local supermarket who have paid bills through membership cards. In the dataset there are 5 attributes such as – customer ID, gender, age, annual income and spending score. Spending score is something that assigned by the customers based on their defined parameters like customer behavior and purchasing data. To understand the customer's need like who can be easily converge [Target Customers]. We applied K-means clustering algorithm to extract information and gather insights for finding target customer .The result of the analysis can be given to marketing team and plan the strategy accordingly.

Dataset From Kaggle : <https://www.kaggle.com/datasets/vjchoudhary7>

#### **1. Installing necessary packages**

```
#Installing necessary packages
```

```
install.packages("stats")
```

```
install.packages("dplyr")
```

```
install.packages("factoextra")
```

```
install.packages("ggfortify")
```

```
install.packages("NbClust")
```

```
library(stats)
```

```
library(dplyr)
```

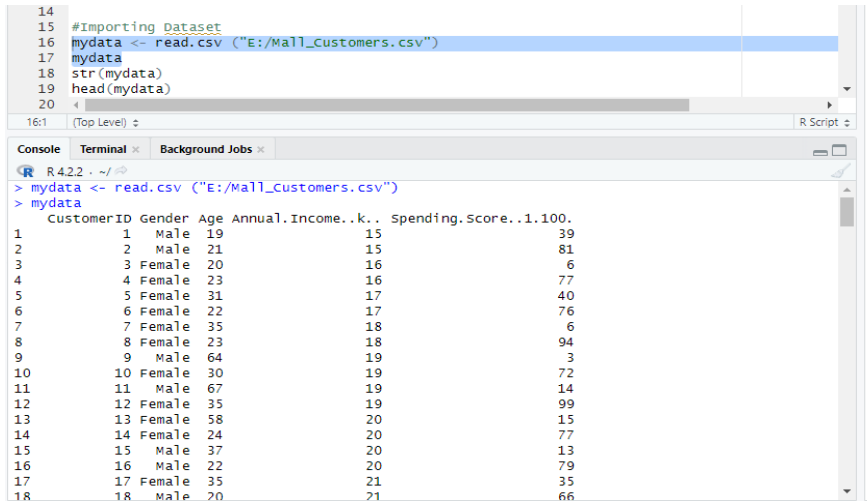
```
library(factoextra)
```

```
library(ggfortify)
```

```
library(NbClust)
```

## 2. Imported Dataset and Summarized

```
mydata <- read.csv("E:/Mall_Customers.csv")  
mydata
```



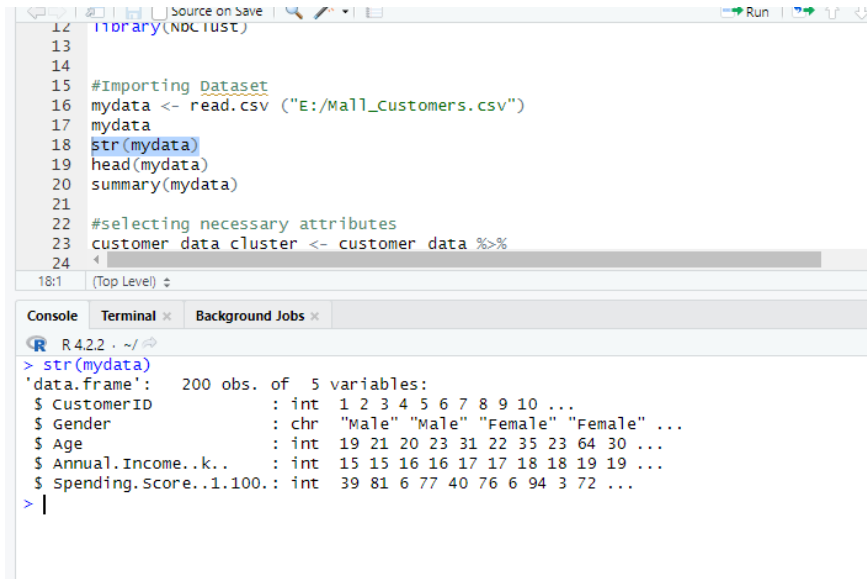
The screenshot shows the RStudio interface. The script editor contains the following code:

```
14  
15 #Importing Dataset  
16 mydata <- read.csv("E:/Mall_Customers.csv")  
17 mydata  
18 str(mydata)  
19 head(mydata)  
20
```

The console shows the output of the code:

```
R 4.2.2 . ~/ > mydata <- read.csv("E:/Mall_Customers.csv")  
> mydata  
  CustomerID Gender Age Annual.Income..k.. Spending.Score..1.100.  
1          1   Male  19              15              39  
2          2   Male  21              15              81  
3          3 Female  20              16               6  
4          4 Female  23              16              77  
5          5 Female  31              17              40  
6          6 Female  22              17              76  
7          7 Female  35              18               6  
8          8 Female  23              18              94  
9          9   Male  64              19               3  
10         10 Female  30              19              72  
11         11   Male  67              19              14  
12         12 Female  35              19              99  
13         13 Female  58              20              15  
14         14 Female  24              20              77  
15         15   Male  37              20              13  
16         16   Male  22              20              79  
17         17 Female  35              21              35  
18         18   Male  20              21              66
```

```
#List of objects  
str(mydata)
```



The screenshot shows the RStudio interface. The script editor contains the following code:

```
12 library(NBCTUST)  
13  
14  
15 #Importing Dataset  
16 mydata <- read.csv("E:/Mall_Customers.csv")  
17 mydata  
18 str(mydata)  
19 head(mydata)  
20 summary(mydata)  
21  
22 #selecting necessary attributes  
23 customer data cluster <- customer data %>%  
24
```

The console shows the output of the `str(mydata)` command:

```
R 4.2.2 . ~/ > str(mydata)  
'data.frame': 200 obs. of 5 variables:  
 $ CustomerID : int 1 2 3 4 5 6 7 8 9 10 ...  
 $ Gender : chr "Male" "Male" "Female" "Female" ...  
 $ Age : int 19 21 20 23 31 22 35 23 64 30 ...  
 $ Annual.Income..k.. : int 15 15 16 16 17 17 18 18 19 19 ...  
 $ Spending.Score..1.100.: int 39 81 6 77 40 76 6 94 3 72 ...  
> |
```

#list of objects in a dataframe

head(mydata)

```
17 mydata
18 str(mydata)
19 head(mydata)
20 summary(mydata)
21
22 #selecting necessary attributes
23 customer_data_cluster <- customer_data %>%
24 <
```

19:1 (Top Level) ⌵

Console Terminal × Background Jobs ×

R 4.2.2 . ~/

```
> head(mydata)
  CustomerID Gender Age Annual.Income..k.. Spending.Score..1.100.
1          1   Male  19             15              39
2          2   Male  21             15              81
3          3 Female  20             16               6
4          4 Female  23             16             77
5          5 Female  31             17             40
6          6 Female  22             17             76
> |
```

#Summary of attributes

summary(mydata)

```
18 str(mydata)
19 head(mydata)
20 summary(mydata)
21
22 #selecting necessary attributes
23 customer_data_cluster <- customer_data %>%
24 <
```

20:1 (Top Level) ⌵

Console Terminal × Background Jobs ×

R 4.2.2 . ~/

```
> summary(mydata)
  CustomerID      Gender      Age      Annual.Income..k..  Spending.Score..1.100.
Min.   : 1.00   Length:200   Min.   :18.00   Min.   : 15.00   Min.   : 1.00
1st Qu.: 50.75   Class :character   1st Qu.:28.75   1st Qu.: 41.50   1st Qu.:34.75
Median :100.50   Mode  :character   Median :36.00   Median : 61.50   Median :50.00
Mean   :100.50                      Mean   :38.85   Mean   : 60.56   Mean   :50.20
3rd Qu.:150.25                      3rd Qu.:49.00   3rd Qu.: 78.00   3rd Qu.:73.00
Max.   :200.00                      Max.   :70.00   Max.   :137.00   Max.   :99.00
> |
```

### 3. Selecting necessary attributes

```
customer_data_cluster <- customer_data %>%
```

```
select(Age, Annual.Income..k., Spending.Score..1.100.)
```

```
customer_data_cluster
```

```

20 summary(mydata)
21
22 #selecting necessary attributes
23 customer_data_cluster <- customer_data %>%
24   select(Age, Annual.Income..k.., Spending.Score..1.100.)
25
26 customer_data_cluster
27
28
29 # Scaling the data
30 customer_data_scale <- scale(customer_data_cluster)
31
23:1 (Top Level)

```

```

R 4.2.2 . ~/
> customer_data_cluster <- customer_data %>%
+   select(Age, Annual.Income..k.., Spending.Score..1.100.)
>
> customer_data_cluster
  Age Annual.Income..k.. Spending.Score..1.100.
1   19                15                   39
2   21                15                   81
3   20                16                    6
4   23                16                   77
5   31                17                   40
6   22                17                   76
7   35                18                    6
8   23                18                   94
9   64                19                    3
10  30                19                   72
11  67                19                   14
12  35                19                   99
13  58                20                    15
14  24                20                   77
15  37                20                    13
16  22                20                    79

```

## 4. Scaling the data

```
customer_data_scale <- scale(customer_data_cluster)
customer_data_scale
```

```

28 # Scaling the data
29 customer_data_scale <- scale(customer_data_cluster)
30 customer_data_scale
31 #Distance
32 customer_dist_data <- dist(customer_data_scale)
33 customer_dist_data
34
35
36
37 #pie chart of gender
38 install.packages("plotrix")
39
29:1 (Top Level)

```

```

R 4.2.2 . ~/
> # Scaling the data
> customer_data_scale <- scale(customer_data_cluster)
> customer_data_scale
  Age Annual.Income..k.. Spending.Score..1.100.
[1,] -1.42100291      -1.73464625      -0.433713114
[2,] -1.27782881      -1.73464625      1.192711064
[3,] -1.34941586      -1.69657236     -1.711617825
[4,] -1.13465471      -1.69657236      1.037813523
[5,] -0.56195833      -1.65849848     -0.394988729
[6,] -1.20624176      -1.65849848      0.999089138
[7,] -0.27561014      -1.62042459     -1.711617825
[8,] -1.13465471      -1.62042459      1.696128071
[9,] 1.80041426       -1.58235070     -1.827790981
[10,] -0.63354538     -1.58235070      0.844191597
[11,] 2.01317540       -1.58235070     -1.401822744
[12,] -0.27561014     -1.58235070      1.889749997
[13,] 1.37089197       -1.54427682     -1.363098359
[14,] -1.06306766     -1.54427682     1.037813523
[15,] -0.13243604     -1.54427682     -1.440547129
[16,] -1.20624176     -1.54427682      1.115262293

```

## 5. Distance Measure

```
customer_dist_data <- dist(customer_data_scale)
customer_dist_data
```

```

29 # Scaling the data
30 customer_data_scale <- scale(customer_data_cluster)
31 customer_data_scale
32 #Distance
33 customer_dist_data <- dist(customer_data_scale)
34 customer_dist_data
35
36
37 #pie chart of gender
38 install.packages("plotrix")
39
29:1 (Top Level)

```

```

R 4.2.2 . ~/
[188,] -0.77671947      1.53970795      0.689294056
[189,] 0.15391215      1.61585572     -1.285649588
[190,] -0.20402309     1.61585572      1.347608605
[191,] -0.34719718     1.61585572     -1.053303277
[192,] -0.49037128     1.61585572      0.728018442
[193,] -0.41878423     1.99659458     -1.634169055
[194,] -0.06084899     1.99659458      1.579954916
[195,] 0.58343444      2.26311179     -1.324373973
[196,] -0.27561014     2.26311179      1.115262293
[197,] 0.44026035      2.49155510     -0.859681351
[198,] -0.49037128     2.49155510      0.921640367
[199,] -0.49037128     2.91036785     -1.246925203
[200,] -0.63354538     2.91036785      1.270159834
attr(,"scaled:center")
  Age Annual.Income..k.. Spending.Score..1.100.
 38.85                60.56                50.20
attr(,"scaled:scale")
  Age Annual.Income..k.. Spending.Score..1.100.
13.96901                26.26472                25.82352
>

```

```

28
29 # Scaling the data
30 customer_data_scale <- scale(customer_data_cluster)
31 customer_data_scale
32 #Distance
33 customer_dist_data <- dist(customer_data_scale)
34 customer_dist_data
35
36
37 #pie chart of gender
38 install.packages("plotrix")
39
33:1 (Top Level)

```

```

R 4.2.2 . ~/
> customer_dist_data <- dist(customer_data_scale)
> customer_dist_data

```

	1	2	3	4	5	6	7	8
2	1.63271382							
3	1.28047443	2.90546048						
4	1.49961180	0.21434013	2.75780621					
5	0.86328190	1.74328990	1.53461759	1.54348731				
6	1.45080775	0.22002872	2.71475242	0.08985491	1.53575852			
	9	10	11	12	13	14	15	16
2								
3								
4								
5								
6								
	17	18	19	20	21	22	23	24
2								
3								
4								
5								
6								
	25	26	27	28	29	30	31	32

## 6. Pie chart of gender

```
install.packages("plotrix")
```

```
a= table(mydata$Gender)
```

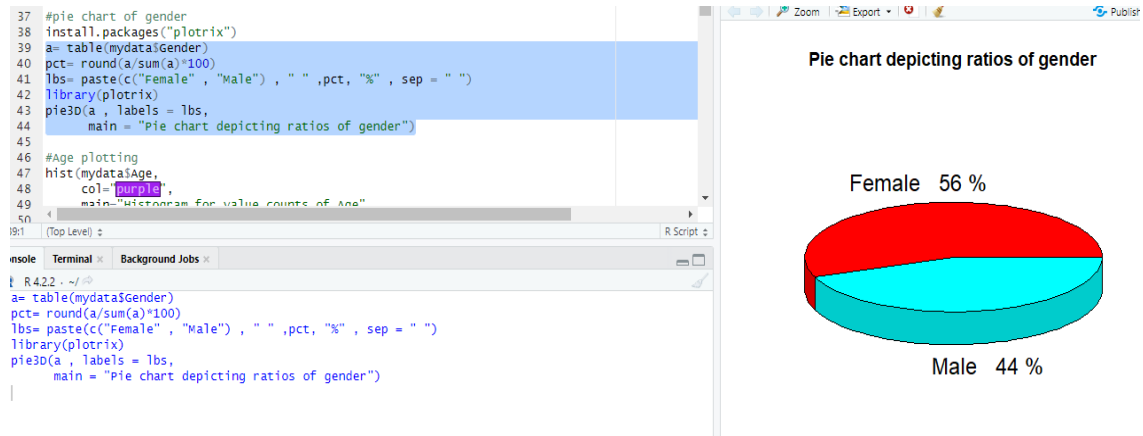
```
pct= round(a/sum(a)*100)
```

```
lbs= paste(c("Female" , "Male") , " " ,pct, "%", sep = " ")
```

```
library(plotrix)
```

```
pie3D(a , labels = lbs,
```

```
main = "Pie chart depicting ratios of gender")
```



## 7. Age plotting

```
hist(mydata$Age,
```

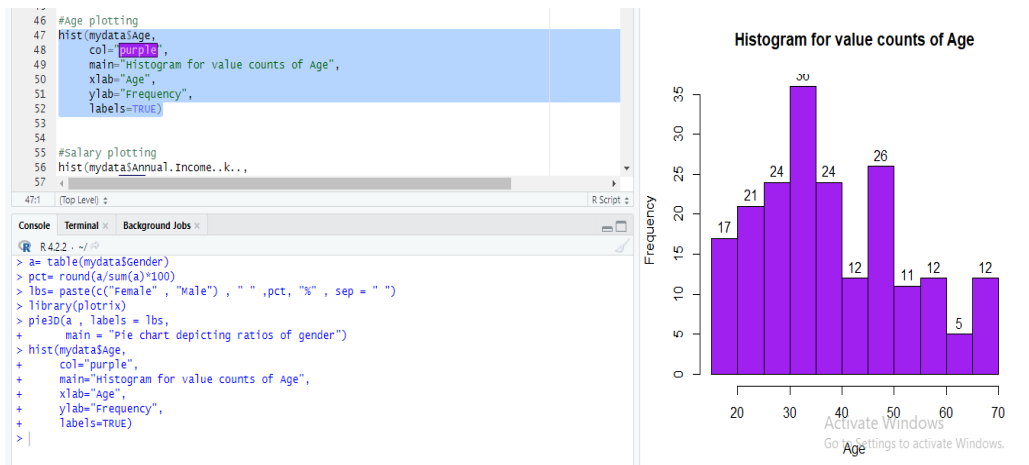
```
col="purple",
```

```
main="Histogram for value counts of Age",
```

```

xlab="Age",
ylab="Frequency",
labels=TRUE)

```

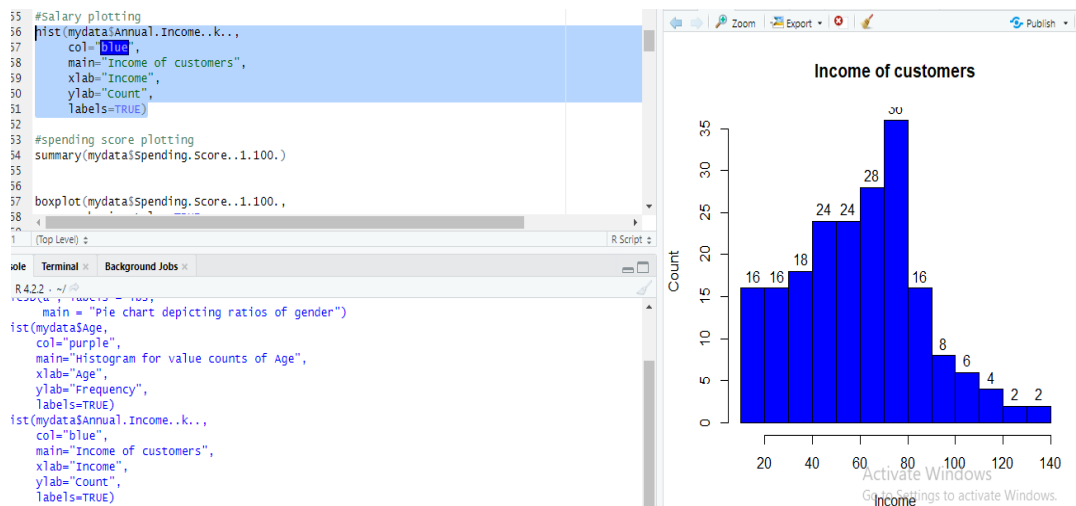


## 8. Salary plotting

```

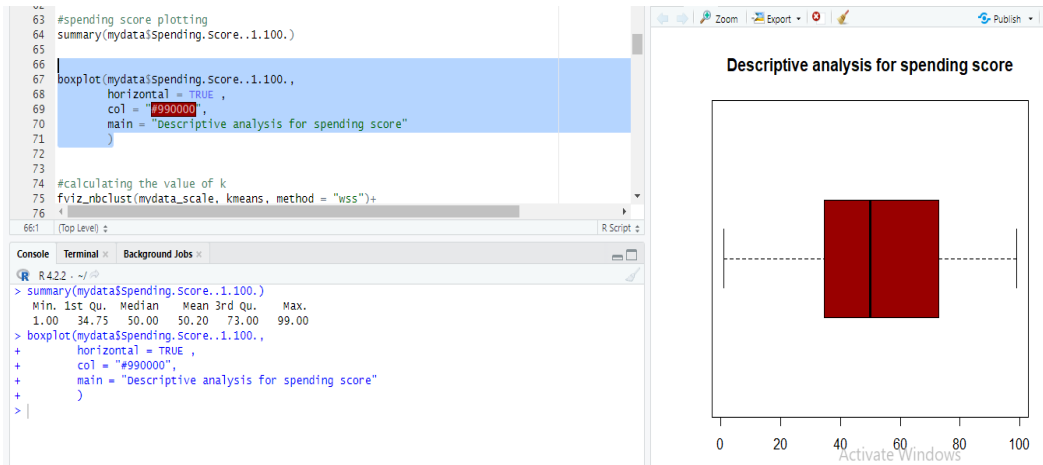
hist(mydata$Annual.Income..k.,
     col="blue",
     main="Income of customers",
     xlab="Income",
     ylab="Count",
     labels=TRUE)

```



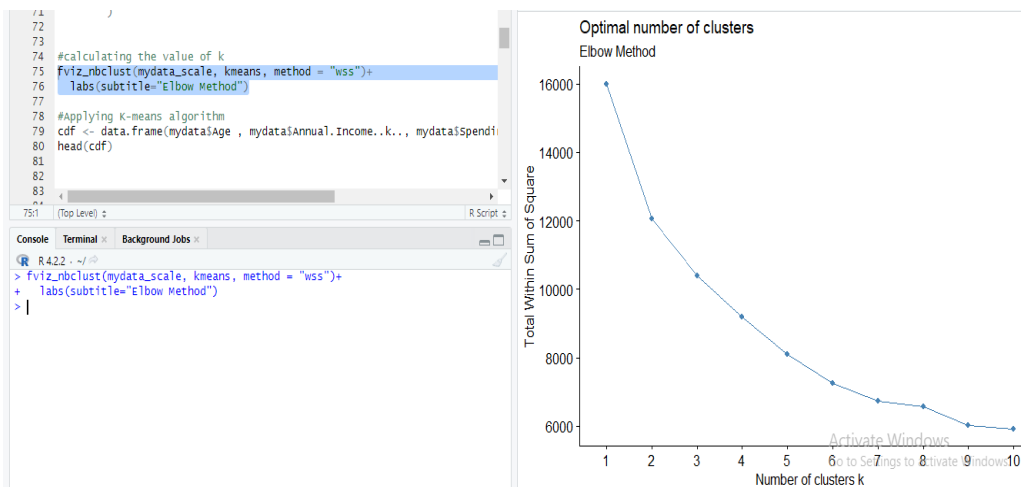
## 9. Spending score plotting

```
summary(mydata$Spending.Score..1.100.)  
boxplot(mydata$Spending.Score..1.100.,  
        horizontal = TRUE ,  
        col = "#990000",  
        main = "Descriptive analysis for spending score"  
)
```



## 10. Calculating the value of k

```
fviz_nbclust(mydata_scale, kmeans, method = "wss")+  
  labs(subtitle="Elbow Method")
```



## 11. Applying K-means algorithm

```
cdf<-data.frame(mydata$Age,mydata$Annual.Income..k.,  
mydata$Spending.Score..1.100.)  
head(cdf)
```

```
77  
78 #Applying K-means algorithm  
79 cdf <- data.frame(mydata$Age , mydata$Annual.Income..k. , mydata$Spending.Score..1.100.)  
80 head(cdf)  
81  
82  
83  
84 #using the gap statistics method.  
85 library(cluster)  
86 set.seed(125)  
87 stat_gap <- clusGap(cdf, FUN = kmeans, nstart = 25,  
88                    K.max = 10, B = 50)  
89  
90  
91  
79:1 (Top Level) R Script
```

```
R 4.2.2 . ~/>  
> fviz_nbclust(mydata_scale, kmeans, method = "wss")+  
+   labs(subtitle="Elbow Method")  
> cdf <- data.frame(mydata$Age , mydata$Annual.Income..k. , mydata$Spending.Score..1.100.)  
> head(cdf)  
  mydata.Age mydata.Annual.Income..k. mydata.Spending.Score..1.100.  
1         19                15             39  
2         21                15             81  
3         20                16              6  
4         23                16             77  
5         31                17             40  
6         22                17             76  
> |
```

## 12. Using the gap statistics method.

```
library(cluster)  
set.seed(125)  
stat_gap <- clusGap(cdf, FUN = kmeans, nstart = 25,  
                    K.max = 10, B = 50)
```

```
80 head(cdf)  
81  
82  
83  
84 #using the gap statistics method.  
85 library(cluster)  
86 set.seed(125)  
87 stat_gap <- clusGap(cdf, FUN = kmeans, nstart = 25,  
88                    K.max = 10, B = 50)  
89  
90  
91 #Dividing into 6 clusters  
92 k6<-kmeans(cdf,6,iter.max=100,nstart=50,algorithm="Lloyd")  
93 k6  
94  
95  
85:1 (Top Level) R Script
```

```
R 4.2.2 . ~/>  
> library(cluster)  
> set.seed(125)  
> stat_gap <- clusGap(cdf, FUN = kmeans, nstart = 25,  
+                    K.max = 10, B = 50)  
Clustering k = 1,2,..., K.max (= 10): .. done  
Bootstrapping, b = 1,2,..., B (= 50) [one "." per sample]:  
..... 50  
> |
```



### 13.Dividing into 6 clusters

```
k6<-kmeans(cdf,6,iter.max=100,nstart=50,algorithm="Lloyd")
```

k6

[illegible]

## 14.principal component analysis

```
pcclust=prcomp(cdf,scale=FALSE)
```

```
summary(pcclust)
```

```
pcclust$rotation[,1:2]
```

```

95 #principal component analysis
96 pcclust=prcomp(cdf,scale=FALSE)
97 summary(pcclust)
98 pcclust$rotation[,1:2]
99
100 #cluster plotting
101 options(repr.plot.width = 12, repr.plot.height = 10)
102 clusplot(cdf, k6$cluster, color=TRUE, shade=TRUE, labels=0,lines=0)
103
104

```

Console

```

R 4.2.2 ~ /
> pcclust=prcomp(cdf,scale=FALSE)
> summary(pcclust)
Importance of components:
              PC1      PC2      PC3
Standard deviation 26.4625 26.1597 12.9317
Proportion of Variance 0.4512 0.4410 0.1078
Cumulative Proportion 0.4512 0.8922 1.0000
> pcclust$rotation[,1:2]
              PC1      PC2
mydata.Age      0.1889742 -0.1309652
mydata.Anual.Income.k.. -0.5886410 -0.8083757
mydata.Spending.Score..1.100. -0.7859965 0.5739136
>

```

## 15.Cluster plotting

options(repr.plot.width = 12, repr.plot.height = 10)

clusplot(cdf, k6\$cluster, color=TRUE, shade=TRUE, labels=0,lines=0)

```

99
100 #cluster plotting
101 options(repr.plot.width = 12, repr.plot.height = 10)
102 clusplot(cdf, k6$cluster, color=TRUE, shade=TRUE, labels=0,lines=0)
103
104
105
106
107 #Scatter plotting of clusters
108
109

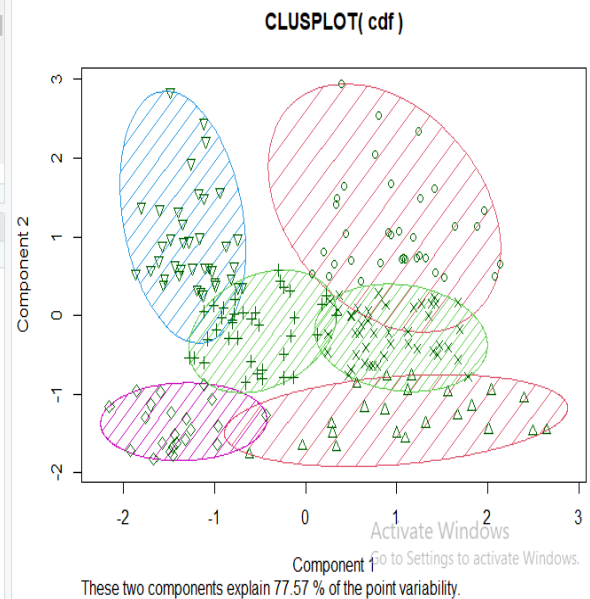
```

Console

```

R 4.2.2 ~ /
> pcclust=prcomp(cdf,scale=FALSE)
> summary(pcclust)
Importance of components:
              PC1      PC2      PC3
Standard deviation 26.4625 26.1597 12.9317
Proportion of Variance 0.4512 0.4410 0.1078
Cumulative Proportion 0.4512 0.8922 1.0000
> pcclust$rotation[,1:2]
              PC1      PC2
mydata.Age      0.1889742 -0.1309652
mydata.Anual.Income.k.. -0.5886410 -0.8083757
mydata.Spending.Score..1.100. -0.7859965 0.5739136
> options(repr.plot.width = 12, repr.plot.height = 10)
> clusplot(cdf, k6$cluster, color=TRUE, shade=TRUE, labels=0,lines=0)
>

```



## 16.Scatter plotting of clusters

```
library("ggplot2")
set.seed(1)
options(repr.plot.width = 12, repr.plot.height = 8)
ggplot(cdf, aes(x = mydata.Age, y = mydata$Spending.Score..1.100.)) +
  geom_point(stat = "identity", aes(color = as.factor(k6$cluster))) +
  scale_color_discrete(name=" ",
    breaks=c("1", "2", "3", "4", "5", "6"),
    labels=c("Cluster 1", "Cluster 2", "Cluster 3", "Cluster 4", "Cluster 5", "Cluster
6")) +
  ggtitle("Segments of Customers", subtitle = "Using K-means Clustering")
```



## 17.Final visualization

```
kcols = function(vec){ cols = rainbow(length(unique(vec)))  
  return(cols[as.numeric(as.factor(vec))])  
}  
digCluster<- k6$cluster; dignm <- as.character(digCluster); #k-means cluster  
  
plot(pcclust$x[,1:2], col = kcols(digCluster) , pch = 19 , xlab = "K-Means" , ylab = "Classes")  
  
legend("bottomleft" , unique(dignm) , fill = unique(kcols(digCluster)))
```

