Assignment No. 5

CODE AND OUTPUT:

```
# Install the required libraries if not already installed

install.packages("ggplot2")

install.packages("factoextra")

install.packages("dplyr")


install.packages("colorspace")

# Load the required libraries

library(ggplot2)

library(factoextra)

library(dplyr)


#Load the Mall Customers Dataset

mall_customers <- read.csv('Mall_Customers.csv')
```

```
mall_customers    200 obs. of 5 variables
    $ CustomerID   : int  1 2 3 4 5 6 7 8 9 10 ...
    $ Genre        : chr  "Male" "Male" "Female" "Female" ...
    $ Age          : int  19 21 20 23 31 22 35 23 64 30 ...
    $ AnnualIncome : int  15 15 16 16 17 17 18 18 19 19 ...
    $ SpendingScore: int  39 81 6 77 40 76 6 94 3 72 ...
```

```
head(mall_customers)
```

```
> head(mall_customers)
  CustomerID  Genre Age AnnualIncome SpendingScore
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
```

names(mall_customers)

```
> names(mall_customers)
[1] "CustomerID"    "Genre"         "Age"           "AnnualIncome"  "SpendingScore"
```

print(colnames(mall_customers))

```
> print(colnames(mall_customers))
[1] "CustomerID"    "Genre"         "Age"           "AnnualIncome"  "SpendingScore"
```

data <- mall_customers[c("AnnualIncome","SpendingScore")]

```
data                200 obs. of 2 variables
    $ AnnualIncome : int  15 15 16 16 17 17 18 18 19 19 ...
    $ SpendingScore: int  39 81 6 77 40 76 6 94 3 72 ...
```

head(data)

```
> head(data)
  AnnualIncome SpendingScore
1           15            39
2           15            81
3           16             6
4           16            77
5           17            40
6           17            76
```

Scale the data (optional, but recommended for K-Means)

data_scaled <- scale(data)

```
  data_scaled       num [1:200, 1:2] -1.73 -1.73 -1.7 -1.7 -1.66 ...
```

print("scaled data")

```
> print("scaled data")
[1] "scaled data"
```

head(data_scaled)

```
> head(data_scaled)
     AnnualIncome SpendingScore
[1,]    -1.734646    -0.4337131
[2,]    -1.734646     1.1927111
[3,]    -1.696572    -1.7116178
[4,]    -1.696572     1.0378135
[5,]    -1.658498    -0.3949887
[6,]    -1.658498     0.9990891
```

calculated WSS for each clustes from 1 to 15

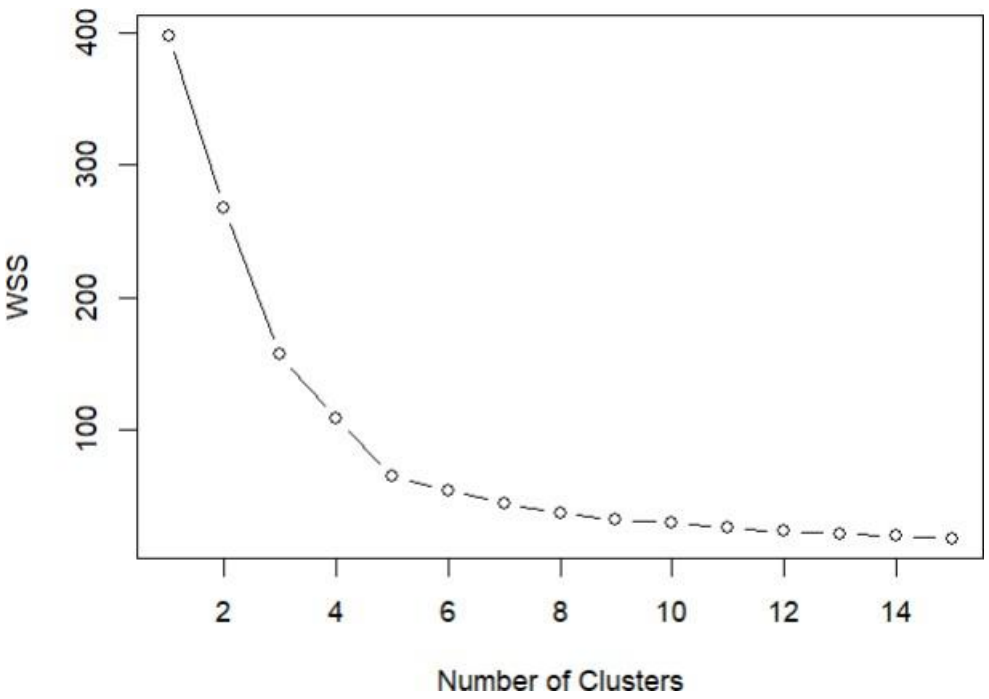
wss <- numeric(15)

for (k in 1:15) wss[k] <- sum(kmeans(data_scaled, centers=k,nstart=25)$withinss)

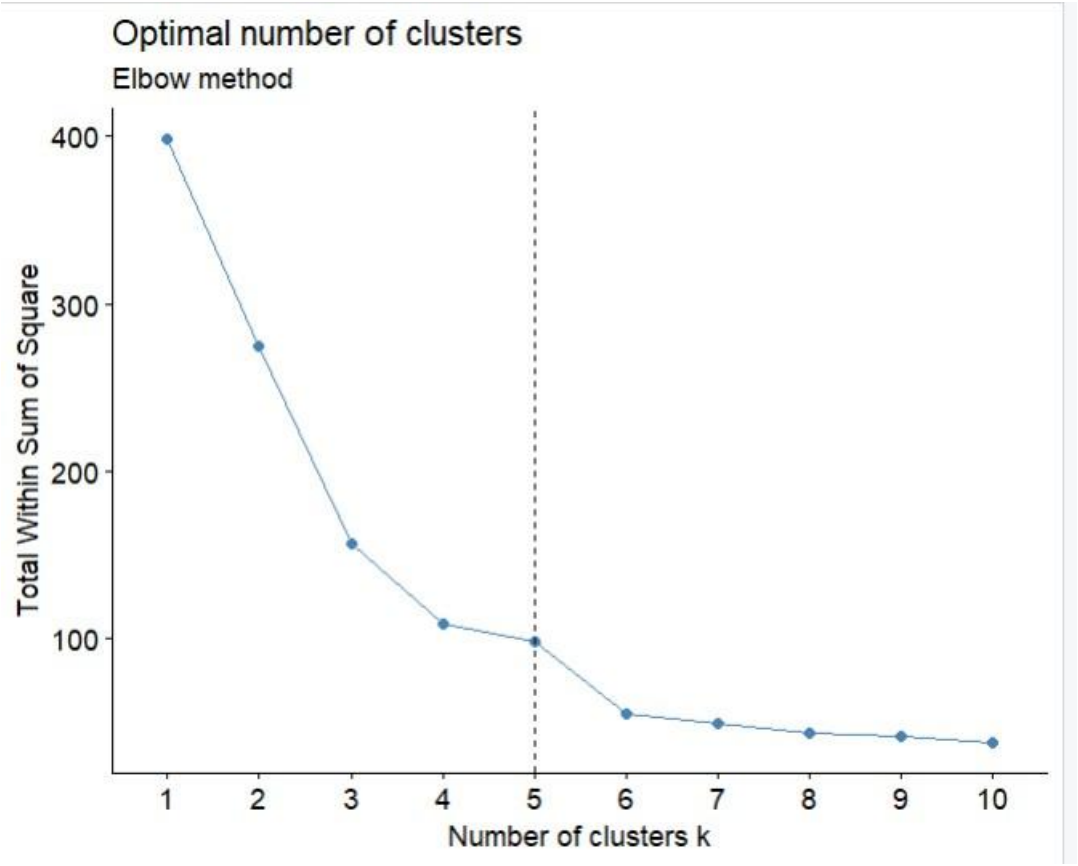
Values	
k	15L
wss	num [1:15] 398 267.7 156.9 108.4 65.2 ...

plot the graph of number of clusters vs WSS

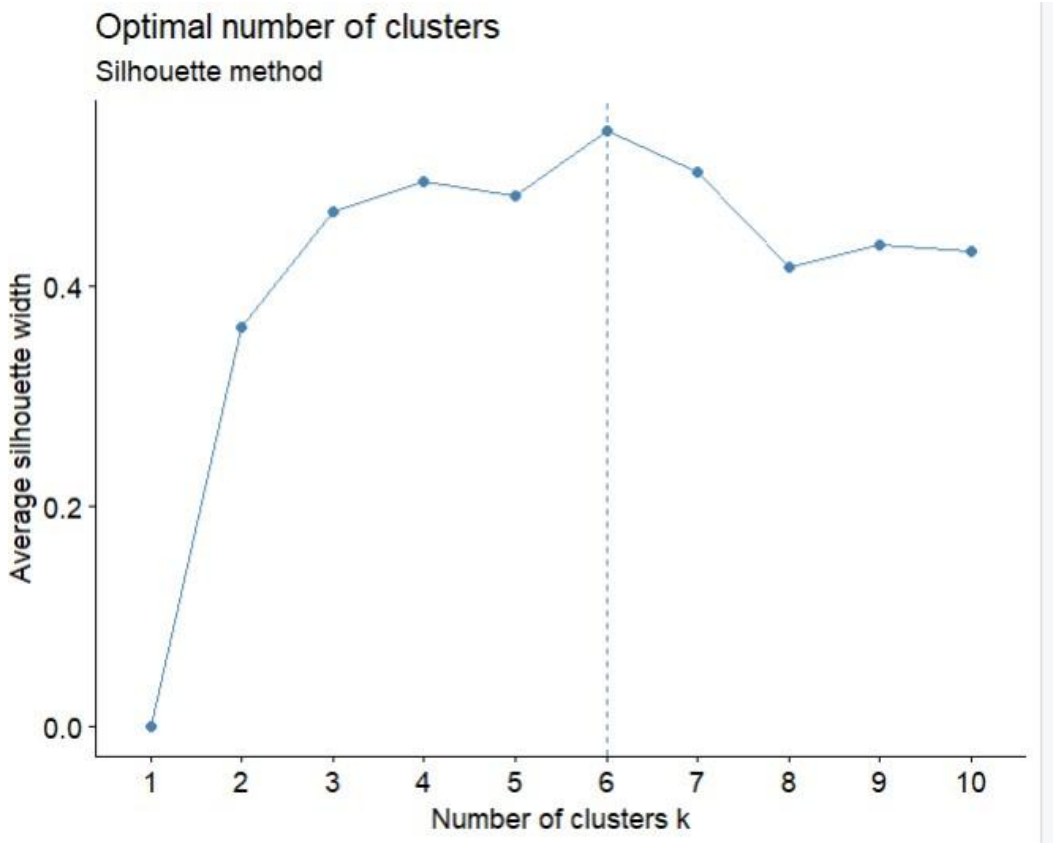
plot(1:15, wss, type="b", xlab="Number of Clusters", ylab="WSS")



```
# Elbow method to find the optimal number of clusters
fviz_nbclust(data_scaled, kmeans, method = "wss") +
geom_vline(xintercept = 5, linetype = 2) +
 labs(subtitle = "Elbow method")
```



```
# Silhouette method
fviz_nbclust(data_scaled, kmeans, method = "silhouette") +
 labs(subtitle = "Silhouette method")
```



Apply K-Means clustering with 3 clusters

set.seed(123) # Set seed for reproducibility in case python- random_state

kmeans_result <- kmeans(data_scaled, centers = 5, nstart = 25)

```
kmeans_result          List of  9
    $ cluster     : int [1:200] 2 3 2 3 2 3 2 3 2 3 ...
    $ centers     : num [1:5, 1:2] -0.2 -1.304 -1.326 1.052 0.989 ...
     ..- attr(*, "dimnames")=List of 2
     .. ..$ : chr [1:5] "1" "2" "3" "4" ...
     .. ..$ : chr [1:2] "AnnualIncome" "SpendingScore"
    $ totss       : num 398
    $ withinss    : num [1:5] 14.49 7.58 5.22 18.3 19.66
    $ tot.withinss: num 65.2
    $ betweenss   : num 333
    $ size        : int [1:5] 81 23 22 35 39
    $ iter        : int 3
    $ ifault      : int 0
    - attr(*, "class")= chr "kmeans"
```

Add the cluster assignments to the original dataset

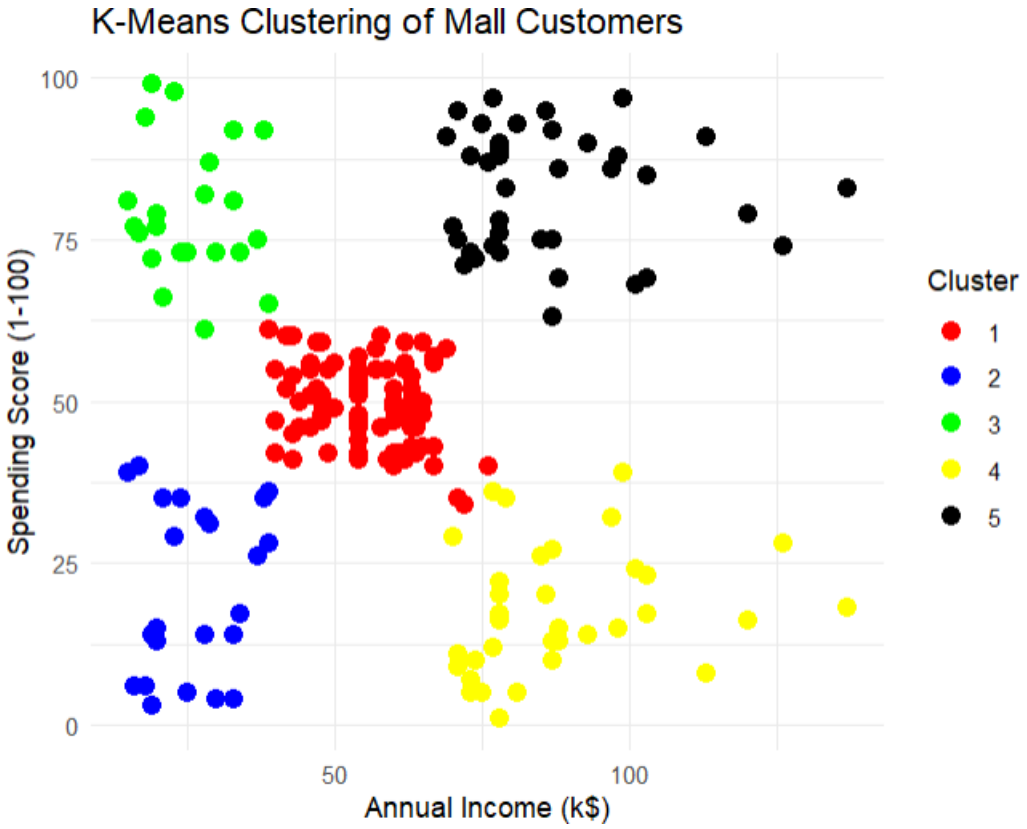
mall_customers$Cluster <- as.factor(kmeans_result$cluster)

```
mall_customers    200 obs. of 6 variables
    $ CustomerID    : int  1 2 3 4 5 6 7 8 9 10 ...
    $ Genre         : chr  "Male" "Male" "Female" "Female" ...
    $ Age           : int  19 21 20 23 31 22 35 23 64 30 ...
    $ AnnualIncome  : int  15 15 16 16 17 17 18 18 19 19 ...
    $ SpendingScore: int  39 81 6 77 40 76 6 94 3 72 ...
    $ Cluster       : Factor w/ 5 levels "1","2","3","4",..: 2
```

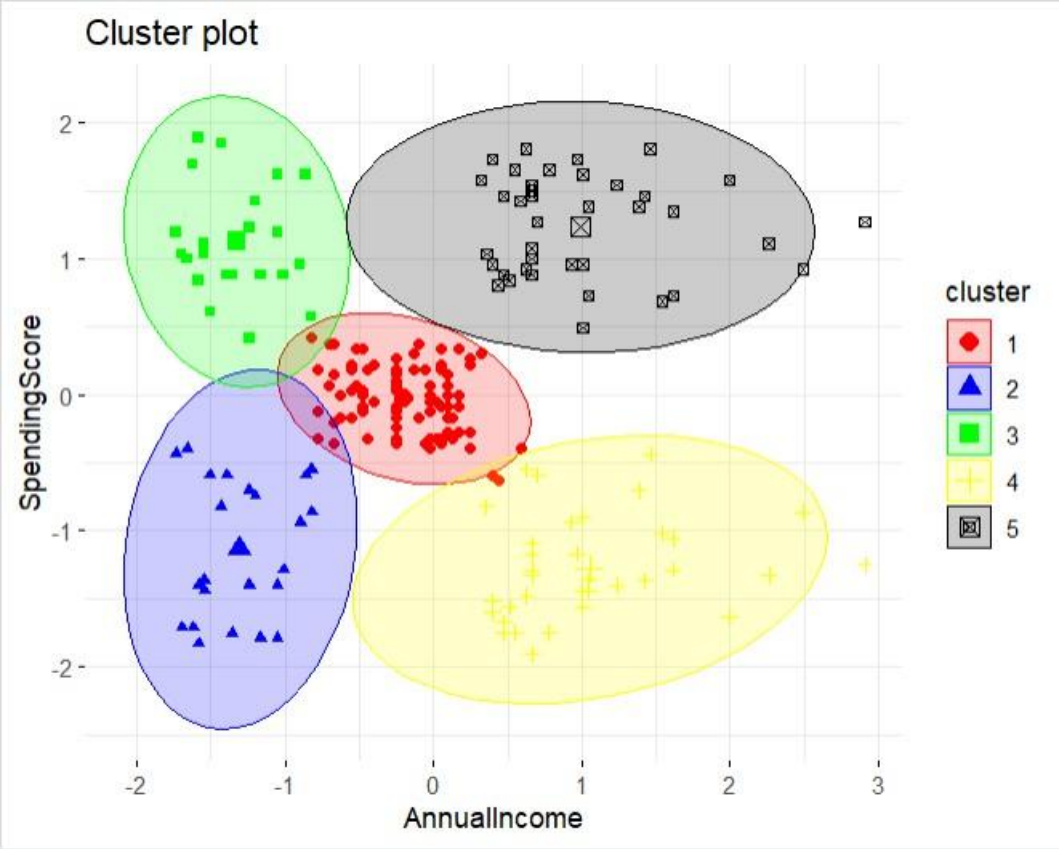
tail(mall_customers)

```
> tail(mall_customers)
    CustomerID  Genre Age AnnualIncome SpendingScore Cluster
195        195 Female  47          120            16       4
196        196 Female  35          120            79       5
197        197 Female  45          126            28       4
198        198   Male  32          126            74       5
199        199   Male  32          137            18       4
200        200   Male  30          137            83       5
```

```
# Scatter plot of the clusters
ggplot(mall_customers, aes(x = AnnualIncome, y = SpendingScore, color = Cluster)) +
 geom_point(size = 3) +
 scale_color_manual(values = c("red", "blue", "green","yellow","black")) +
 labs(title = "K-Means Clustering of Mall Customers",
    x = "Annual Income (k$)",
    y = "Spending Score (1-100)") +
 theme_minimal()
```



```
# Scatter plot with cluster centers
fviz_cluster(kmeans_result, data = data_scaled,
        geom = "point",
        ellipse.type = "norm",
        ggtheme = theme_minimal(),
        palette = c("red", "blue", "green","yellow","black"))
kmeans_result
```




```
K-means clustering with 5 clusters of sizes 81, 23, 22, 35, 39

Cluster means:
  AnnualIncome SpendingScore
1   -0.2004097   -0.02638995
2   -1.3042458   -1.13411939
3   -1.3262173    1.12934389
4    1.0523622   -1.28122394
5    0.9891010    1.23640011

Clustering vector:
  [1] 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 1 1 1 1 1 1 1 1
 [56] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[111] 1 1 1 1 1 1 1 1 1 1 1 1 1 5 4 5 1 5 4 5 4 5 1 5 4 5 4 5 4 5 4 5 1 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5
[166] 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5

Within cluster sum of squares by cluster:
[1] 14.485632  7.577407  5.217630 18.304646 19.655252
 (between_SS / total_SS =  83.6 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss" "betweenss"    "size"
[8] "iter"         "ifault"
```