**Customer Segmentation Using K – Means**

**Clustering Method**

**By**

**Rohit Vikaas P**

**2005039**

**Introduction :**

Customer segmentation is the process of dividing customers into groups based on common characteristics or behaviors. It helps businesses to target different segments effectively and appropriately. One of the methods to perform customer segmentation is using machine learning techniques such as K-Means clustering. K-Means clustering is an unsupervised algorithm that assigns data points to clusters based on their similarity. In R, there are various packages and functions that can be used to implement K-Means clustering and visualize the results. Some of the steps involved in customer segmentation using R are:

* Importing and exploring the data
* Data visualization and analysis
* Determining the optimal number of clusters using methods such as elbow method or silhouette method
* Applying K-Means clustering and assigning cluster labels to the data
* Interpreting and profiling the clusters based on their features
* Optionally, using principal component analysis (PCA) to reduce the dimensionality of the data and plot the clusters

Some of the benefits of customer segmentation based on web search results are:

* It allows you to fine-tune your message and communicate with your customers in the most relevant and effective way
* It improves customer engagement, experience, loyalty and retention by meeting their needs and preferences
* It gives you an insight into your customers and their behaviors, such as spending habits, interests, budgets, etc.
* It helps you to gain a competitive advantage and differentiate your products or services from others
* It helps you to focus on the right people and prioritize the most profitable or potential segments
* It improves your pricing and product development strategies by aligning them with the customer segments’ needs and willingness to pay
* It improves your assessment of campaigns and marketing performance by measuring the results for each segment

**Code :**

#Customer Segmentation in R implementing K-means algorithm

getwd()

setwd("C:/Users/parip/OneDrive/Desktop/Customer Segmentation")

#Loading the data

customer\_data <- read.csv("Mall\_Customers.csv")

customer\_data

#Quick summary of data

head(customer\_data)

summary(customer\_data$Age)

#Some statistical value of features

sd(customer\_data$Age)

summary(customer\_data$Annual.Income..k..)

sd(customer\_data$Annual.Income..k..)

summary(customer\_data$Age)

sd(customer\_data$Spending.Score..1.100.)

#Plot gender visualization

a=table(customer\_data$Gender)

barplot(a,main="Using Barplot to display Gender Comparision",

ylab="Count",

xlab="Gender",

col=rainbow(2),

legend=rownames(a))

#Piechart

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

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

library(plotrix)

pie3D(a,labels=lbs,main="Pie Chart depicting ratio of male and female")

#Age Graph Histogram

summary(customer\_data$Age)

hist(customer\_data$Age,

col="green",

main="Histogram of show count of age class",

xlab="Age Class",

ylab="Frequency",

labels = TRUE)

#Boxplot for age

boxplot(customer\_data$Age,

horizontal = TRUE,

col="green",

main="Boxplot for analysis of age")

#Annual income

summary(customer\_data$Annual.Income..k..)

hist(customer\_data$Annual.Income..k..,

col="#660033",

main="Histogram for annual income",

xlab="Annual income class",

ylab = "Frequency",

labels = TRUE)

plot(density(customer\_data$Annual.Income..k..),

col="yellow",

main="Density plot for annual income",

xlab="Annual income class",

ylab="Density")

polygon(density(customer\_data$Annual.Income..k..),

col="#ccff66")

#Spending Score

summary(customer\_data$Spending.Score..1.100.)

boxplot(customer\_data$Spending.Score..1.100.,

horizontal = TRUE,

col="#990000",

main="Boxploy for spending score")

hist(customer\_data$Spending.Score..1.100.,

main="Histogram for spending score",

xlab = "Spending Score Class",

ylab = "Frequency",

col="#6600cc",

labels = TRUE)

#K-means Clustering

#Elbow method : intra-cluster variation stays minimum

library(purrr)

set.seed(123)

#Function to calculate total intra-cluster sum of square

iss <- function(k)

{

kmeans(customer\_data[,3:5],k,iter.max = 100,nstart=100,algorithm = "Lloyd")$tot.withinss

}

k.values <- 1:10

iss\_values <- map\_dbl(k.values,iss)

plot(k.values, iss\_values,

type="b",pch=19,frame=FALSE,

xlab="Number of clusters K",

ylab = "Total intra-clusters sum of squares")

#Clusters

#Average silhoouette method

library(cluster)

library(gridExtra)

library(grid)

k2<-kmeans(customer\_data[,3:5],2,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s2<-plot(silhouette(k2$cluster,dist(customer\_data[,3:5],"euclidean")))

k3<-kmeans(customer\_data[,3:5],3,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s3<-plot(silhouette(k3$cluster,dist(customer\_data[,3:5],"euclidean")))

k4<-kmeans(customer\_data[,3:5],4,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s4<-plot(silhouette(k4$cluster,dist(customer\_data[,3:5],"euclidean")))

k5<-kmeans(customer\_data[,3:5],5,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s5<-plot(silhouette(k5$cluster,dist(customer\_data[,3:5],"euclidean")))

k6<-kmeans(customer\_data[,3:5],6,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s6<-plot(silhouette(k6$cluster,dist(customer\_data[,3:5],"euclidean")))

k7<-kmeans(customer\_data[,3:5],7,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s7<-plot(silhouette(k7$cluster,dist(customer\_data[,3:5],"euclidean")))

k8<-kmeans(customer\_data[,3:5],8,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s8<-plot(silhouette(k8$cluster,dist(customer\_data[,3:5],"euclidean")))

k9<-kmeans(customer\_data[,3:5],9,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s9<-plot(silhouette(k9$cluster,dist(customer\_data[,3:5],"euclidean")))

k10<-kmeans(customer\_data[,3:5],10,iter.max = 100,nstart = 50,algorithm = "Lloyd")

s10<-plot(silhouette(k10$cluster,dist(customer\_data[,3:5],"euclidean")))

#Determine and visualize the optimal number of clusters

library(NbClust)

library(factoextra)

fviz\_nbclust(customer\_data[,3:5],kmeans,method="silhouette")

#Gap static method

set.seed(125)

stat\_gap <- clusGap(customer\_data[,3:5], FUN = kmeans, nstart = 25,

K.max = 10 , B = 50)

fviz\_gap\_stat(stat\_gap)

k6<-kmeans(customer\_data[,3:5],6,iter.max = 100,nstart = 50,algorithm = "Lloyd")

k6

#Visualizing the clustring results using the first two principle components

pcclust=prcomp(customer\_data[,3:5],scale=FALSE) #Principal Component Analysis

summary(pcclust)

pcclust$rotation[,1:2]

#Visualize

set.seed(1)

ggplot(customer\_data, aes(x=Annual.Income..k..,y=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 mall customers",subtitle = "Using K-means Clustering")

#K-means

ggplot(customer\_data, aes(x=Spending.Score..1.100.,y=Age)) +

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 mall customers",subtitle = "Using K-means Clustering")

#Finally visulaize K-means values

kcols = function(vec)

{

cols=rainbow(length(unique(vec)))

return (cols[as.numeric(as.factor(vec))])

}

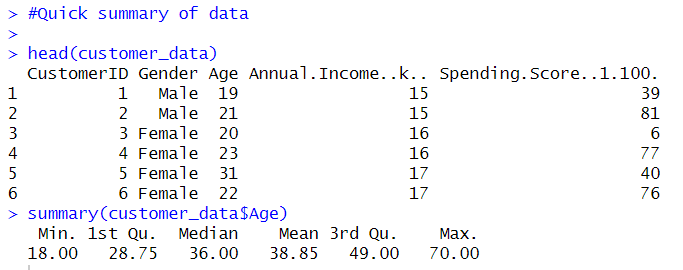
digCluster <- k6$cluster; dignm<-as.character(digCluster);

plot(pcclust$x[,1:2], col=kcols(digCluster),pch=19,xlab = "K-means",ylab = "Classes")

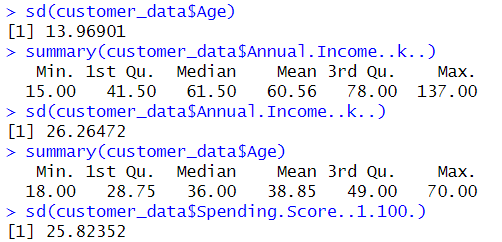
legend("bottomleft",unique(dignm),fill=unique(kcols(digCluster)))

**Output :**

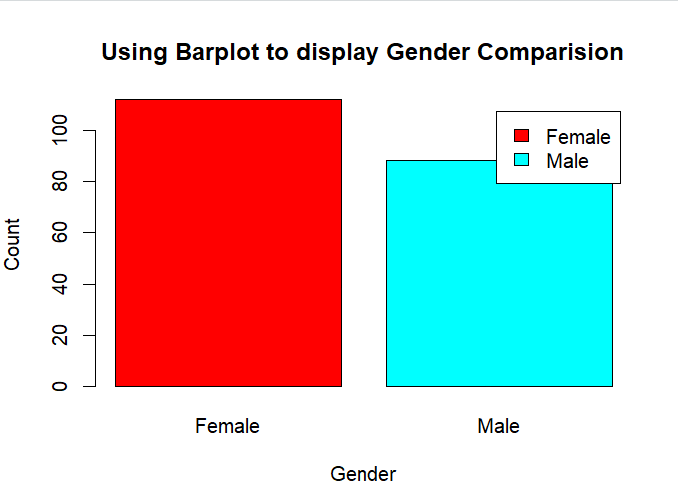
Summary of the dataset and customer age :



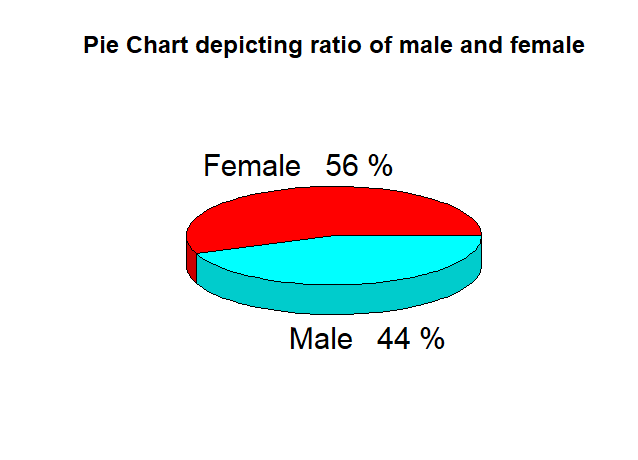
Some statistical value of features :



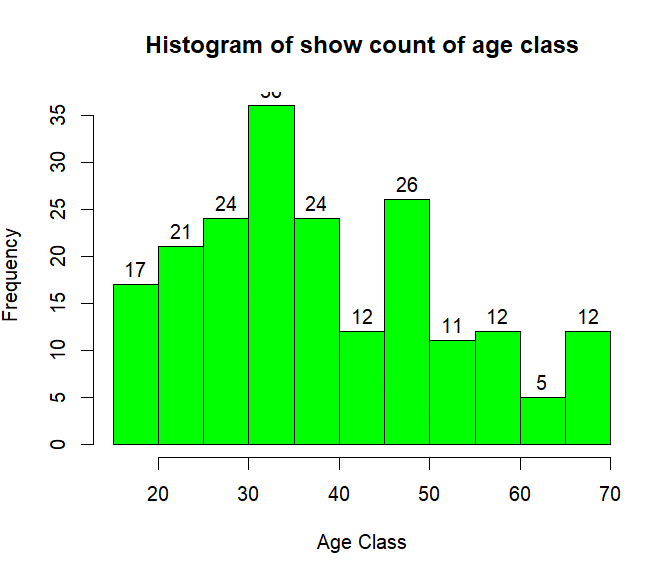
Plot gender visualization :



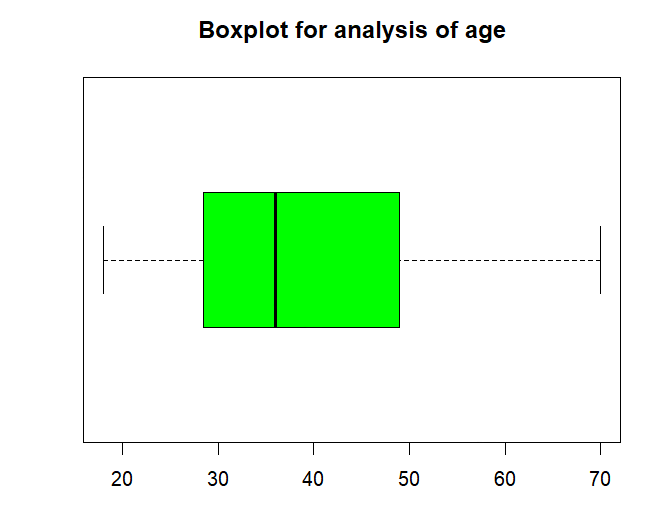
Piechart for the given data :



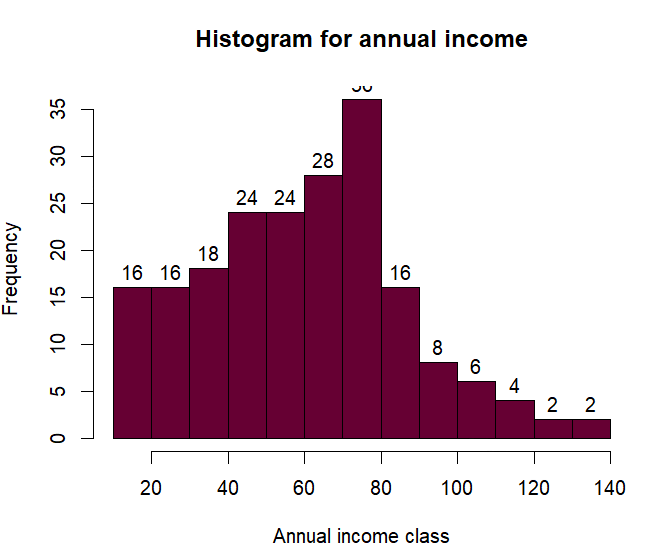
Age Graph Histogram :

****

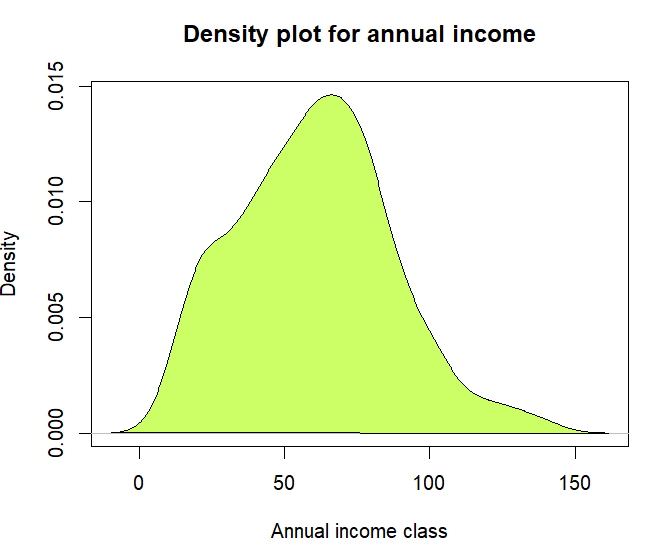
Boxplot for age :



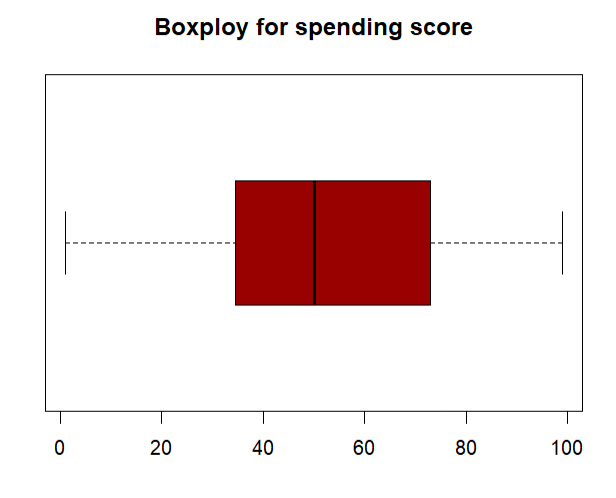
Annual income – Histogram :



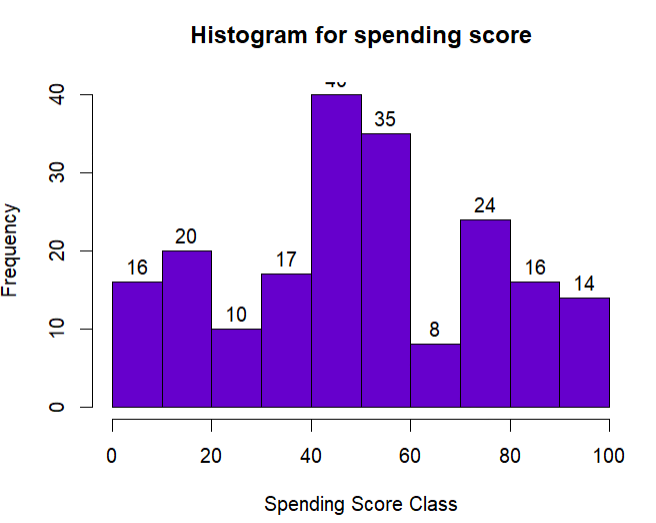
Density plot for annual income:



Spending Score – Boxplot :

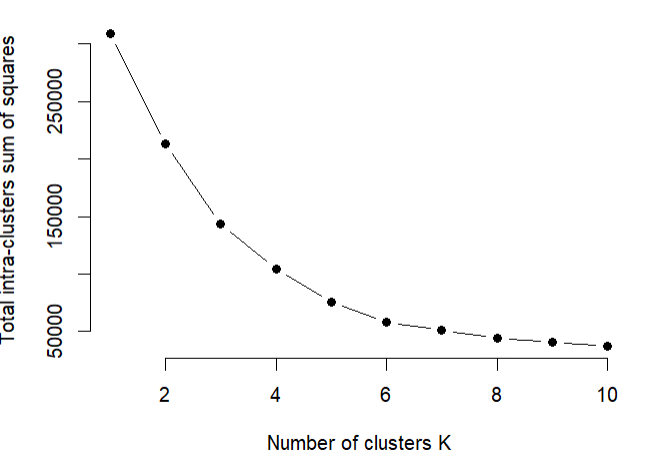


Spending Score – Histogram:

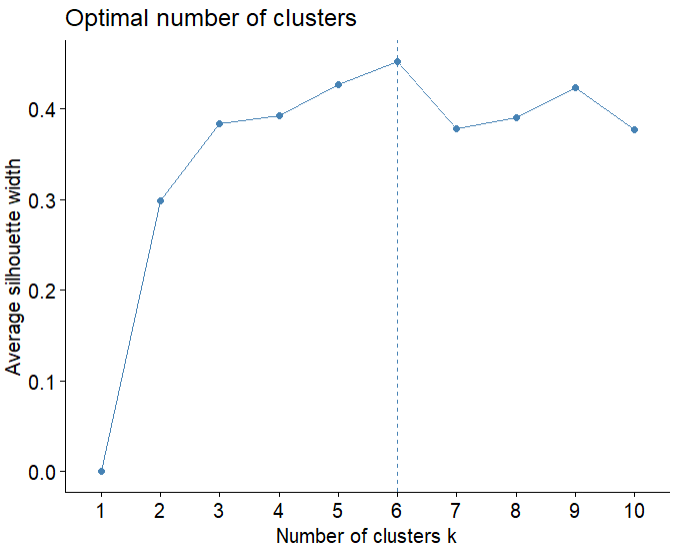


K-means Clustering :

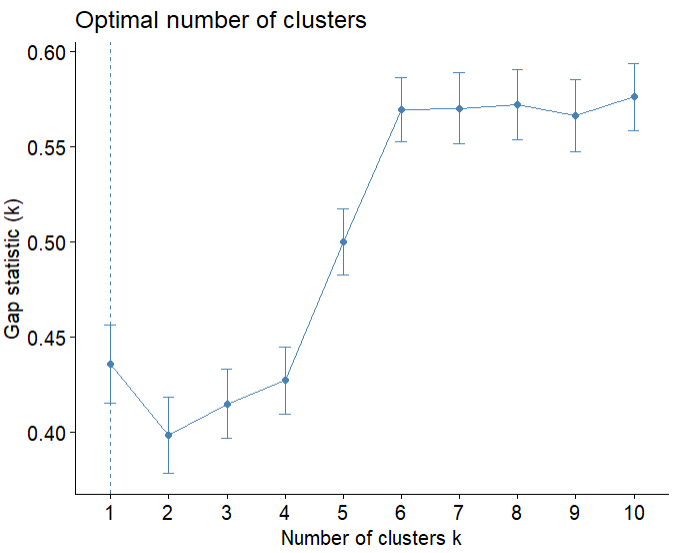
Elbow method : Intra-cluster variation stays minimum :



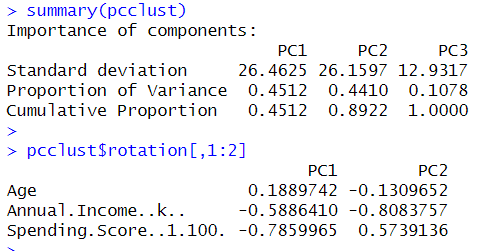
Determine and visualize the optimal number of clusters :



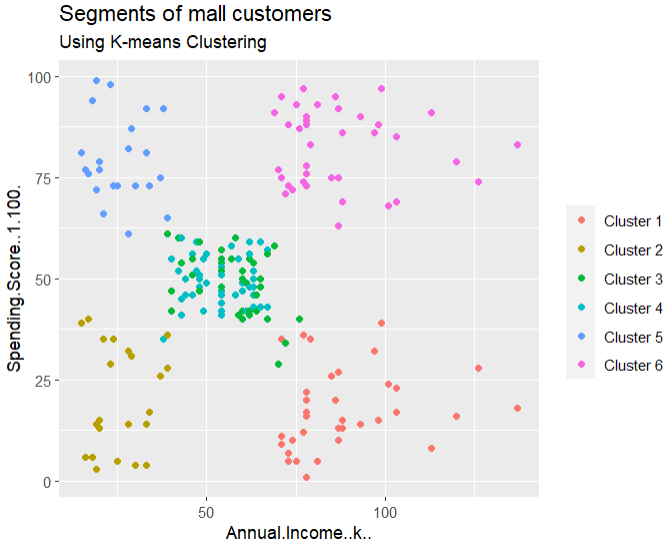
Gap static method :



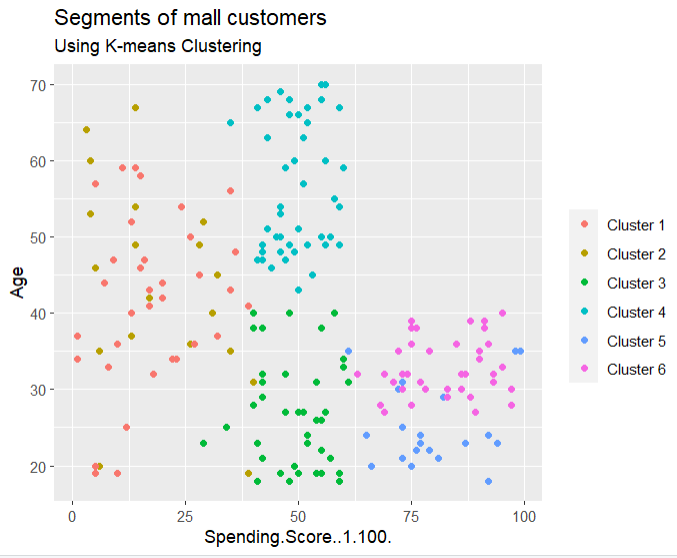
Visualizing the clustring results using the first two principle components :



Visualize :



K-means :



Finally visulaize K-means values :

