PROJECT

Clustering of Mall Customers Through K-Means,Hierarchical and Gaussian Model

Prepared by

Abhilash Singh

Indian Institute of Technology
Kanpur -208016

ACKNOWLEDGEMENT

I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project work.

It gives me immense pleasure to express my gratefulness to my mentor Dr Amit Mitra**,** Department of Mathematic & Statistics, Indian Institute of Technology, Kanpur, for his constant encouragement and worthy supervision, without which it would not have been possible for me to complete this project. The unmatched support and guidance that I received from him, has contributed immensely towards the successful completion of this project.

I would also like to express my heartfelt thanks to my friends, who were there to help and support me whenever I felt the need, thereby helping me complete this project.

Certificate

This is to certify that the content of this project entitled, "**Clustering of Mall Customers Through K-Means, Hierarchical and Gaussian Model" by** Abhilash Singh is a bona fide work of him submitted to IIT Kanpur, Department of Mathematics & Statistics for consideration in partial fulfilment for completion of the course MTH552A, 2nd semester.

The original project work was carried out by him under my supervision in the academic year 2019-2020.On the basis of declaration made by him I recommend this project for evaluation.

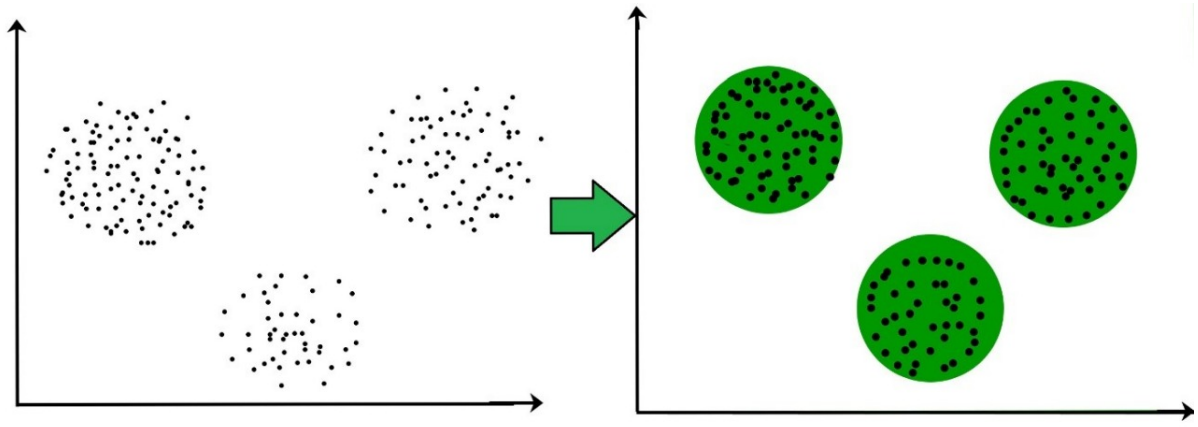
Dr Amit Mitra

Department of Mathematics & Statistics

……

Introduction

Clustering is a type of unsupervised learning method, which draws references from datasets where the data objects are not labelled. It is generally used to look for meaningful structures and generative features within data.
Clustering is defined as a process of grouping a set of data points/objects into a number of groups called clusters, such that the objects in the same cluster are similar, but are dissimilar to those present in other clusters. It is primarily used in exploratory data mining, statistical data analysis, and in many fields including machine learning, pattern recognition, bioinformatics, among others. For example, in the figure below, we have a scattered plot of data points, where it can be clearly seen that the data points can be grouped separately into 3 different clusters, as shown in the figure, where each green circle represents a cluste



Types of Clustering Algorithm:
Clustering is a process that can be achieved through various algorithms and each of them differs in their methods and rules to perform the task. The major clustering methods can be divided into the following categories-
Partition-based Methods:
These methods are mostly distance-based and an initial partitioning of the data objects into k distinct clusters needs to be done and after this, an iterative relocation process is used to improve the partitioning by moving objects from one cluster to another based upon their closeness (related) to other objects in a cluster. The data objects in different clusters are very different and are far apart. K-means clustering algorithm falls under this method of

clustering.

Hierarchical Methods:

In this method, the data points form clusters in the form of a tree-type structure depending on the hierarchy i.e. it creates a hierarchical decomposition of the given data points. An example of this method is the Hierarchical clustering algorithm. This method has two classification – Agglomerative or bottom-up approach and Divisive or top-down approach.

Density-based Methods:

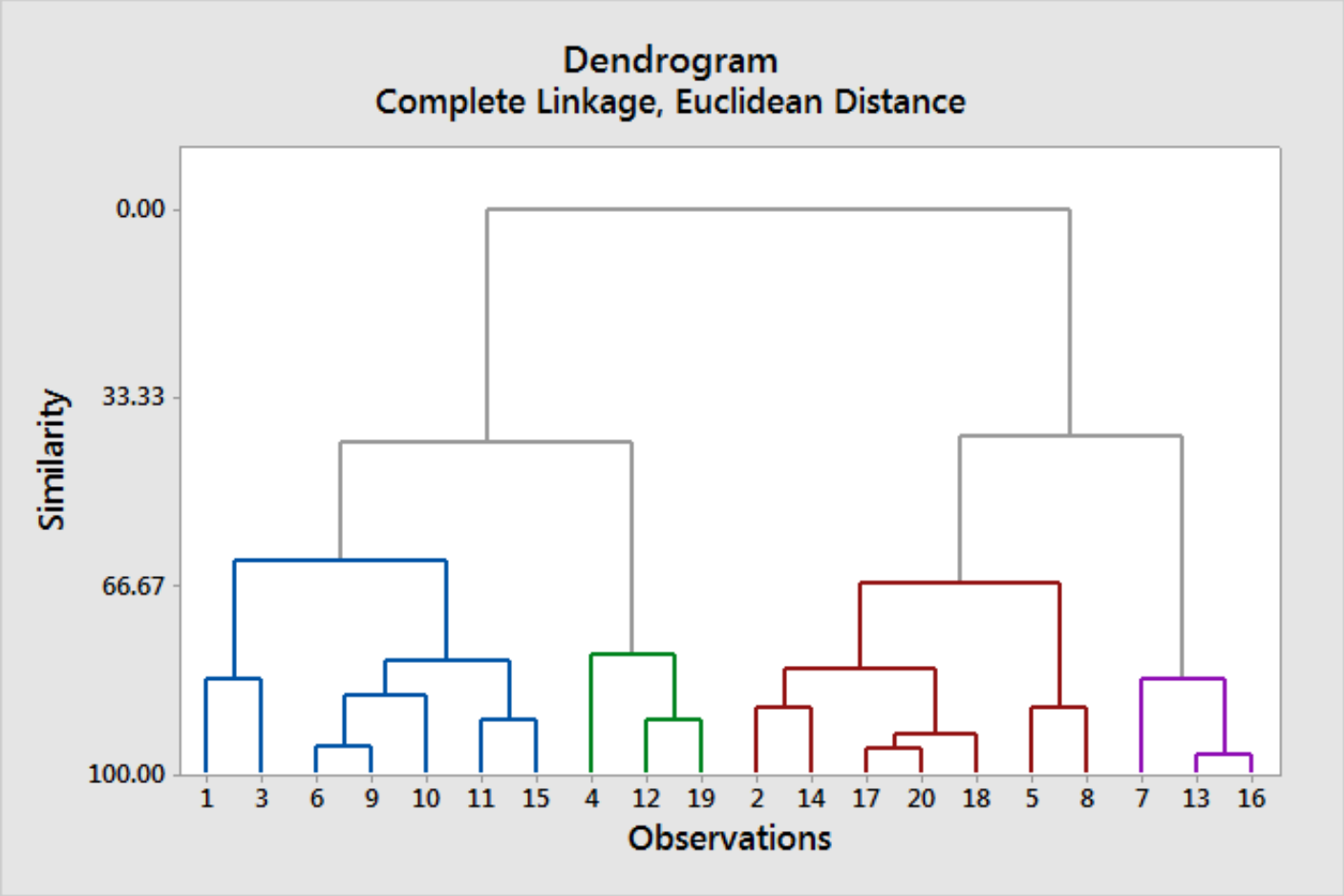
In this, data points are added to a cluster until the density (number of objects) in a particular neighbourhood exceeds a set threshold. This, in turn, helps to form clusters of arbitrary shapes and easily filter out noise or outliers in a set of objects. For example, DBSCAN and OPTICS are density-based approaches that define clusters as connected dense regions in the data space.

Grid-based Methods:

This method includes the division of data space into a finite number of cells that form a grid-like structure. The method is independent of the number of data objects and is dependent only on the number of cells in each dimension in the data space, thereby making the clustering process fast. Examples include STING (Statistical Information Grid), CLIQUE (Clustering In Quest) and others.

Hierarchical clustering:

Hierarchical clustering technique is the one of most popular and easy technique also known as HCA,and it is an unsupervised learning algorithm.In this algorithm we group the objects in such a way that the objects in same group are more similiar than the other group of objects,similar groups of objects are called clusters. Hierarchical clustering is visualized through a hierarchical tree called dendrogram ,which is as shown below



The above tree is called dendrogram of hierarchical clustering,and height of each dendrogram represents the dissimilarty between two objects or groups. we look at horizontal level and set a threshold also called dissimilarty threshold for finding the optimal number of cluster.In order to choose the optimal number of clusters we draw a horizantal line which cuts the dendrogram tree at a height where that line can travel the maximum vertical distance without intersecting any merge point.

Characteristics

i) Sectioning the tree at a perticular level partitions the data into "g" disjoint groups

ii) Sectioning the tree at 2 different levels and if we choose two groups from these two sections then the these two groups are either disjoint or one totally contain within the other

iii) There is a numerical value associated with every merger point of branches.this

neumerical value is a measure of the distance between two merged clusters.

There are two types of hierarchical clustering approach:

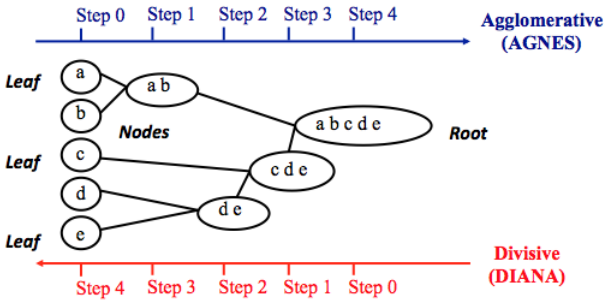
(1) Agglomerative HC algorithm
(2) Divisive HC algorithm

Agglomerative HC algorithm:

This algorithm is also called bottom-up algorithm.In this technique ,we assume each single objects point as indivisual cluster,and with each iteration we merged two most similiar groups of objects in one group, this process isn't stopped until we get single group.

Divissive HC algoritham:

Divisive algorithm is not generally used in real wold.Divisive algorithm is just opposite of agglomerative algorithm.In this algorithm, we assume all object as single group,and after each iteration we seprate objects into two diffrent groups.This process isn't stop until we get each single point as a group.



this is the example of agglomerative and divisive HC algoritham.

Here the one question which arises is that when will we assume that two groups are similiar.For this problem there are several method arises such as Sinle-linkage,Complete-linkage, Average-linkage etc.

Step for agglomerativeH.C algorithm:

Step1
Starts with n objects in n cluster and a nxn distance matrix(symmetric) D=$((d_{i,j}))$

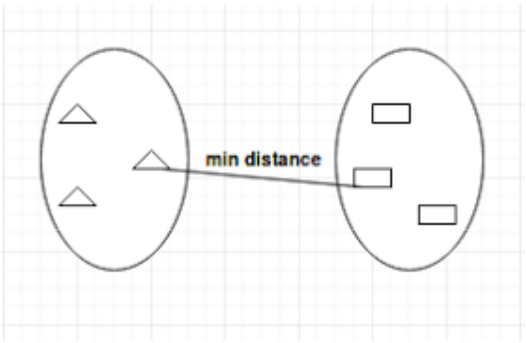
$$D = \begin{pmatrix} d11 & - & - & - \\ d21 & d22 & - & - \\ - & - & - & - \\ - & - & - & dnn \end{pmatrix}$$

here order of matrix is n by n. **Step2**
Search the distance matrix for the most Similiar pair of objects.Suppose U and V in such that $d_{u,v}=min\ d_{i,j}$
Step3
Merge u & v to from (u,v) cluster
and to update distance matrix -
(i) delete row & column corresponding to u and v
(ii) add a new row & column specifying the distance between(u,v) and remaining clusters
Step4
Repeat steps 2 and 3 (n-1) times so that we get a single cluster with n objects record the cluster merged at each and the merger levels.
Step5
Construct the dendogram tree with the information of merger.
Distance between the object
We generally calculate the distance between two point by Euclidean distance. It calculate straight line distance between two points in Euclidean space.
Suppose **a** and **b** are two point in Euclidean space,then the Euclidean distance between two point are given by

$$||a-b||_2 = \sqrt{\sum_{n=1}^{n}(a-b)}$$

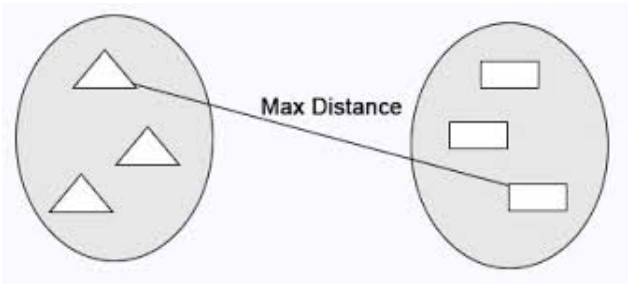
Distance measures between clusters:

1.Single linkage distance
Single linkage distance is defined in such way that it calculte the shortest distance between the two objects in each cluster.For example



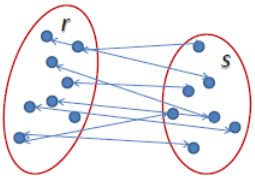
2.Complete linkage distance

Complete linkage distance is defined in such way that it calculate the maximum distance between the two objects in each cluster.For example



2.Average linkage distance

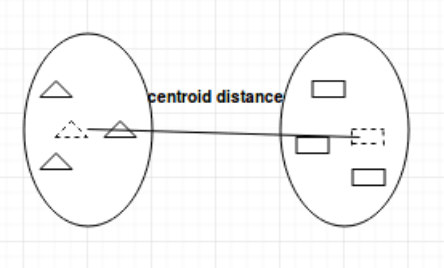
Average linkage distance is defined as the average distance of each objects in one cluster to every objects in other cluster. For example



$$L(r,s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} D(x_{ri}, x_{sj})$$

2.Centroid linkage distance

Centroid linkage distance is defined in such way that it calculate distance between the center of each cluster.For example



The dotted that symbol inside each cluster represents the centroid

Single linkage A.H.C:

In case of Single linkage A.H.C ., we only change the **Step 3** of **Agglomerative H.C algorithm**
changes are given below
In the distance matrix updation stage
(u,v) $\longrightarrow$ merged cluster
w $\longrightarrow$ cluster from the previous stage
$d_{(u,v),w} = min(d_{u,v}, d_{v,w})$

Complete linkage A.H.C:

In case of Complete linkage A.H.C. we only changed the **Step 3** of **Agglomerative H.C algorithm**
changes are given below
In the distance matrix updation stage
(u,v) $\longrightarrow$ merged cluster
w $\longrightarrow$ cluster from the previous stage
$d_{(u,v),w} = max(d_{u,v}, d_{v,w})$

Average linkage A.H.C:

In case of Average linkage A.H.C. we only changed the **Step 3** of **Agglomerative H.C algorithm**
changes are given below
In the distance matrix updation stage
(u,v) $\longrightarrow$ merged cluster
w $\longrightarrow$ cluster from the previous stage

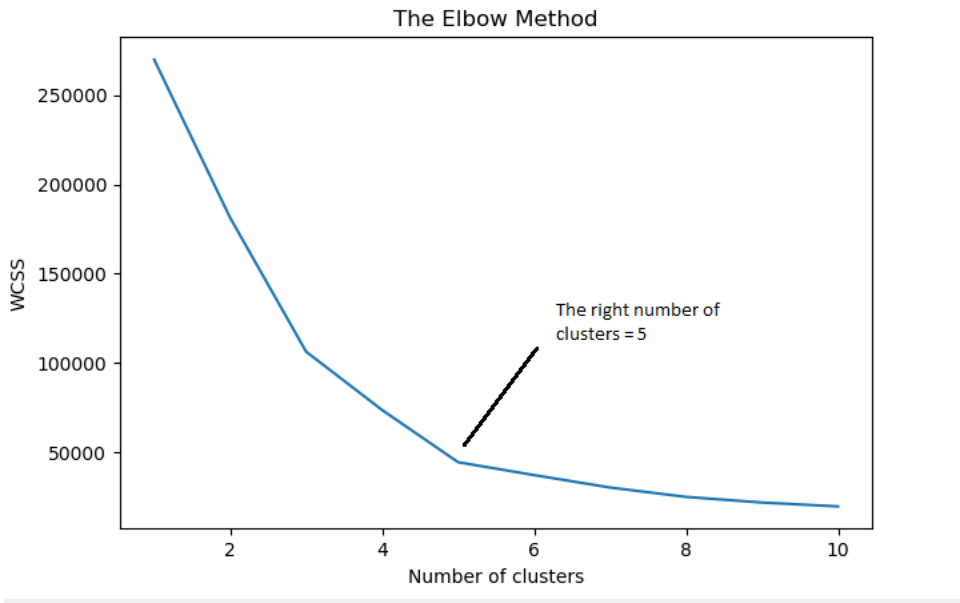
$$d_{(u,v),w} = \frac{\sum\sum d_{(i,j)_{i\in(u,v)\ \ j\in w}}}{N_{(u,v)}N_w}$$

Advantage And Disadvantages:

Advantage
i Here we do'nt need to specify the number number of cluster.
ii It is easy to implement and easy to understand.
Advantage
i It is difficult to determine the correct number of cluster by the dendogram when the data is very large
ii time complexity for the clustering can result very long time

K-Means Clustring:

K-Means algorithm is an unsupervised learning algorithm and it is very simple learning algorithm.In this technique, our aim is partition the objects into k non-overlapping distinct cluster and each object does not belongs to more than one cluster.In this technique we try to minimize within cluster sum of square and maximize between cluster sum of square.
Here the one question which arises is that how to choose k such that we get a good result. For this problem we use **elbow** method which is as fallow
Elbow method
In the elbow method ,we calculate total cluster sum of square for different value of k (here k is number of clusters) and plot the graph between WCSS and k ,location of knee in the graph is generally consider as the indicator of the appropiate number of clusters.
Mathematical expression of WCSS:
Let x_1,x_2,x_3,x_4,x_5.....x_n be set of data point and m_1, m_2, m_3..... m_k are the cluster mean of c_1 , c_2, c_3..... c_k clusters respectively .
$WCSS = \sum_{i=1}^{k} \sum_{x \in c_i} ||x_i - m_i||^2$
For example



We observe in the above graph,k=5 is our optimal number of clusters becouse if we increase the number of cluster above 5 ,we don't get appropiate changes in WCSS

The method starts either with a random partition of the data point into k cluster or a set of randomly choosen set of k need points to act as nucleus of the k cluster
Let x_1,x_2,x_3,x_4,x_5.....x_n be set of data point and m_1, m_2, m_3..... m_k are the cluster mean c_1 , c_2, c_3..... c_k respectively .

K-Means Method:

1.Random Partition K-means

step.1 Random partition of the n cases into k clusters.
step.2 Reassign the object if the objects is closer(Euclidean sense) to cluster center of another cluster then to its randomly assigned cluster.
step.3 Recalculate cluster means of clusters losing an object and also for the clusters gaining the object.
step.4 Continue step.2 and step.3 till no further reassignment is possible.

2.Random Seed point:

step.1 Generate K-random initial seed point
step.2 Walk throught the full data set and assign the n objects into k cluster,corresponding to those seed points
step.3 Reassign the object if the objects is closer(Euclidean sense) to cluster center of another cluster than to its randomly assigned cluster.
step.4 Recalculate cluster means of clusters losing an object and also for the clusters gaining the object.
step.5 Continue step.3 and step.4 till no further reassignment is possible.
Some Mathematical formula for computation:

1. Total point Scatter within the cluster is written

$$W(c) = \sum_{i=1}^{k} N_k \sum_{C(i)=k} ||x_i - \bar{x}_k||^2 \text{.....(1)}$$

where $\bar{x}_k$ is the mean vector associated with the kth cluster and N_k denotes the number of object in k^{th} cluster .We assign the no of observation in such a way that in each cluster the average dissimiliarity(distance) of the observation from the cluster mean as default by point in that cluster is minimized.

$$C^* = \min_{C} \sum_{k=1}^{K} N_k \sum_{C(i)=k} ||x_i - m||^2 \quad \text{.....(a)}$$

for any set of observation S

$$\bar{x}_S = \underset{m}{argmin} \sum_{i \in S} ||x_i - m||^2 \quad \text{.....(2)}$$

we can obtainC^* by solving the enlarged optimization problem

$$\min_{C,\{m_1,m_2 \ldots m_k\}} \sum_{k=1}^{K} N_k \sum_{C(i)=k} ||x_i - m_k||^2 \quad \text{....(3)}$$

K-Means Clustering Algorithm:

Step.1 For a given configuartion $\{C_1, C_2, ...C_k\}$ equation (3) is minimized with respect to $\{m_1, m_2, ...m_k\}$yeilding tthe means of the currently assigned cluster (2)
Step.2 Given current set of means $\{m_1, m_2, ...m_k\}$,equation(3) is minimized by assigning each observation to nearest centroid i.e
$C(i) = \underset{1 \le k \le K}{argmin} ||x_i - m_k||^2$

Step.3 Repeat Step.1 and Step.2 untill no movement is possible

Advantage

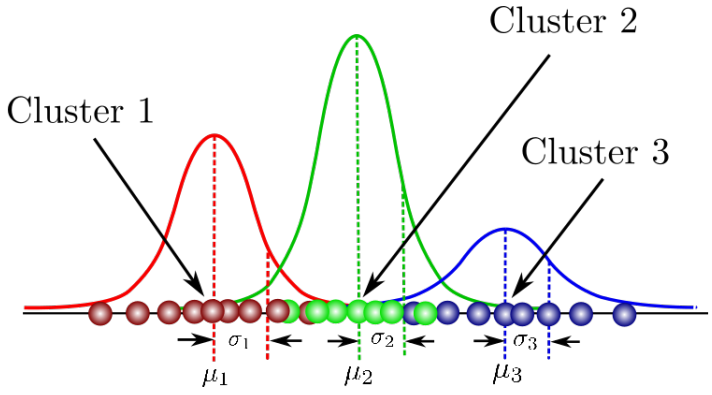
i Fast and easy to understand.
ii Relatively efficient
iii Gives the best result when data point of dataset are distinct

Disadvantage

i Requires apriori specification of the number of clusters.
ii Applicable only when mean is defined and it fails for categorical data
iii It gives only local optima
iv Euclidean distance measures can unequally weight underlying factor
v Hard assignment of the object to clusters

Gaussian Mixture density:

Gaussian mixture model is a probabilistic model,and make a assumption that the distribution of data in whatever dimension may be 1,2.....n is a gaussian distribution with unknown parameter.Gaussian mixture is a function that is the mixture of finite number of gaussian distribution ,and each distribution represents the cluster of the data(objects) through means(centriod of the clusters),covariance and width(weight) of the cluster.For example



We can see that there are three cluster and each cluster is defined by different gaussian distribution.

Univariate Gaussian Distribution

Probability dansity function of gaussian distribution are given below

$G(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$

where σ denotes variance

μ denotes mean of distribution

Multivariate Gaussian Distribution

$N(x|\mu,\sigma) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$

Gaussian Mixture model

$f(x) = \sum_{k=1}^{K} \pi_k N(x|\mu_k, \Sigma_k)$

where K denotes the number of gaussian distribution(clusters).

π_k denotes the prior probability of k^{th} distribution(cluster),and also satisfy the condition-

$0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{K} \pi_k = 1$

Here we try to find the estimate of parameter using **log likelihood function**

$\log(f(X)) = \sum_{n=1}^{N} log\left(f(x_n)\right)$

$= \sum_{n=1}^{N} log\left\{\sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k)\right\}$.....(a)

From equation (a) we can not get the estimate of parameter directly,hence we will use Expectation Maximization Technique

Latent variable:Posterior probability

For a given value of x we can evalute the corresponding posterior probability called responsibilities

Bayes Rule

$g_K(x)$=P(k/X)=$\frac{\pi_k N(x/\mu_k,\Sigma_k)}{\sum_{j=1}^{K} \pi_j N(x/\mu,\Sigma_j)}$

where $g_k(x)$ is called latent variable ,and $\pi_k = \frac{N_k}{N}$

N denotes the total number of sample,andN_k denotes the total number of

sample associated with k^{th} distribution(cluster). For Estimating the parameter (μ_k, π_k, Σ_k) we will use **Expectation Maximization Technique**.

Expectation Maximization Technique
E-M algorithm is an itirative optimization method
E-Step
For a given parameter values we can compute expected values of the latent variable

M-Step
Updates the parameters of the model based on the latent variable calculated using maximum likelihood method

E-M Algorithm for gaussian mixture model
In given gaussian mixture model our aim is maximize the likelihood function with respect to means,covariance and the mixing coefficient.
1 Initialize the mean μ_j,covariance matrix Σ_j and mixing coefficients $\pi_j$and evaluate the initial value of the log likelihood.
E-Step Evalaute the responsibilities using the current parameter values

$$g_k(x) = \frac{\pi_k N(x/\mu_k,\Sigma_k)}{\sum_{j=1}^{K} \pi_j N(x/\mu_j,\Sigma_j)}$$

M-Step Re-estimate the parameter using the current responsibilities

$$\mu_j = \frac{\sum_{n=1}^{N} g_k(x_n) x_n}{\sum_{n=1}^{N} g_j(x_n)}$$

$$\Sigma_j = \frac{\sum_{n=1}^{N} g_j(X_n)(x_n-\mu_j)(x_n-\mu_j)^T}{\sum_{n=1}^{N} g_j(x_n)}$$

$$\pi_j = \frac{\sum_{n=1}^{N} g_j(x_n)}{N}$$

Evalute log likelihood function

$$log\left(f(x/\mu,\Sigma,\pi)\right) = \sum_{n=1}^{N} log\left\{\sum_{k=1}^{K} \pi_k N(x_n/\mu_k,\Sigma_k)\right\}$$

After calculating the value of log likelihood function we go to E-Step for updating the latent variable then go to M-Step for estimating the parameter after estimating the parameter we again calculate the value of log likelihood function.If the value of log likelihood function converges to a specifice value then we will stop this process otherwise repeat this process until function

value converges.

Applying Above Three Algorithms On Real Dataset:

we apply the above three algoritham on mall dataset

```
Console C:/Users/abhil/OneDrive/Desktop/AI/
> # k-means clustering
> # first to check working directory
> getwd()
[1] "C:/Users/abhil/OneDrive/Desktop/AI"
> # set working directory
> setwd("C:/Users/abhil/OneDrive/Desktop/AI")
>
> #Again check working changed or not
> getwd()
[1] "C:/Users/abhil/OneDrive/Desktop/AI"
> #Now my working directory changed
>
> #loading the data set
> data<-read.csv("abhi.csv")
>
```

In the above we used getwd function for checking the our working directory,and setwd() is used for changing our working directory
we used read.csv fuction to loading the mall datadet on our console,and top rows of our data are given below

```
> #loading the data set
> data<-read.csv("abhi.csv")
>
> #checking data table
>  head(data,10)
   CustomerID  Genre 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
>
```

In our mall dataset we have five variables such as CustomerID, Genre,Age,Income and Spending score,here the income are given in annual and spending score

is a score that the mall computed for each of their clients(customer) based on several criteria including. For example their income and how much amount they spend etc. A spending score tending to 1 indictes that the customer spend less money while the spending score tending to 100 means tthe money spent by customer is very high
we use summary() fuction to get summary of the data

```
>  #we are intrested to analysing Annual Income and Spending Score of our customer
>
> #checking summary of data
> data$CustomerID<-as.factor(data$CustomerID)
> summary(data)
   CustomerID     Genre          Age        Annual.Income..k..  Spending.Score..1.100.
 1      :  1   Female:112   Min.   :18.00   Min.   : 15.00     Min.   : 1.00
 2      :  1   Male  : 88   1st Qu.:28.75   1st Qu.: 41.50     1st Qu.:34.75
 3      :  1                Median :36.00   Median : 61.50     Median :50.00
 4      :  1                Mean   :38.85   Mean   : 60.56     Mean   :50.20
 5      :  1                3rd Qu.:49.00   3rd Qu.: 78.00     3rd Qu.:73.00
 6      :  1                Max.   :70.00   Max.   :137.00     Max.   :99.00
 (Other):194
>
> X<-data[,4:5]
>
> head(X)
  Annual.Income..k.. Spending.Score..1.100.
1                 15                      39
2                 15                      81
3                 16                       6
4                 16                      77
5                 17                      40
6                 17                      76
>
```

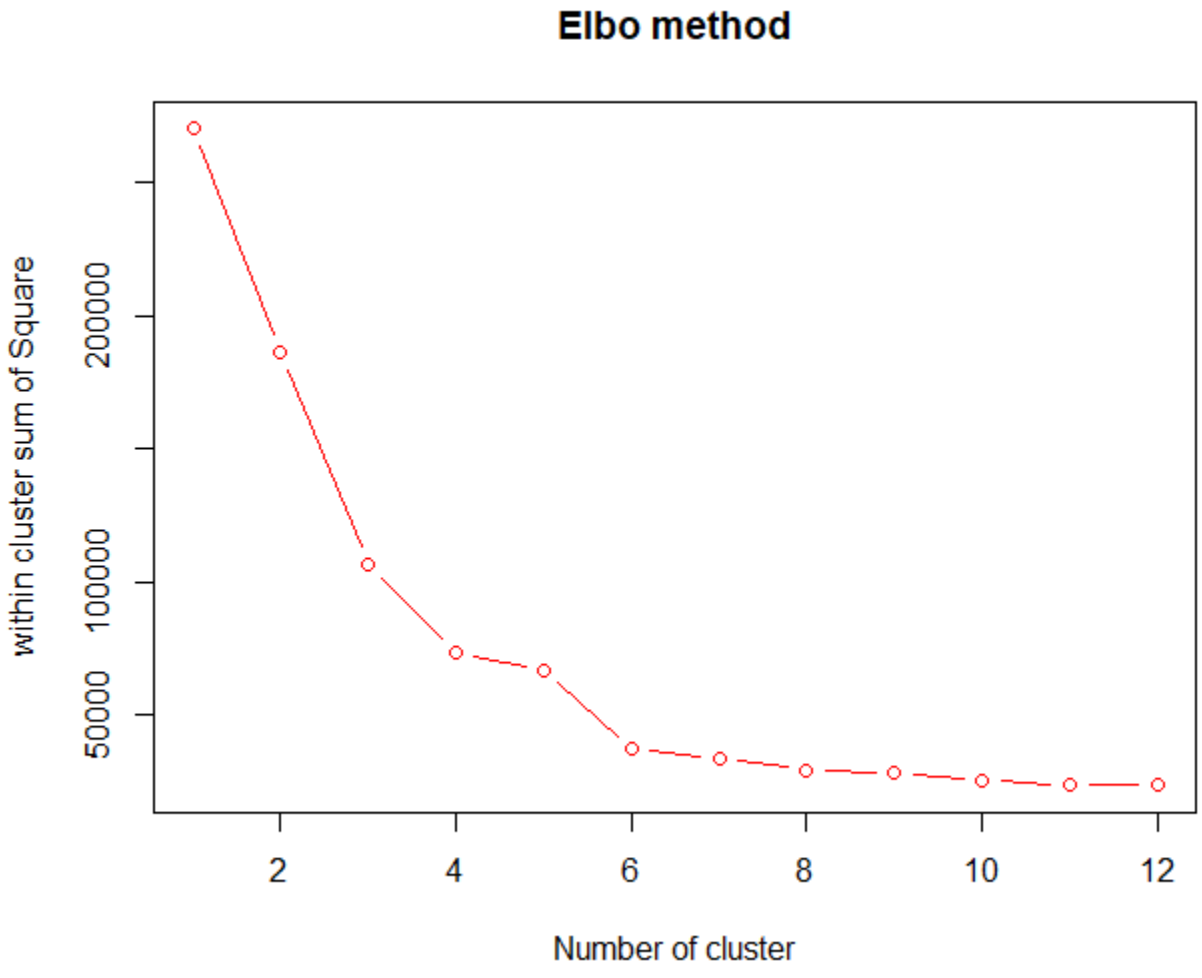
The above graph we observed that there are 112 female clients and 88 are male clients ,and minimum age of client is 18 and maximum age is 70 and average age is 39
minimum anual income of the clients is 15 thousand and maximum 137thousand and average income is 60.56
minimum spending score is 1 and maximum score is 99 and average score is 50

k-means :

In k-means for finding the optimal number of cluster we use albow method.We calculate the total cluster sum of square(wcss) for 12 cluster plot with respect to number of cluster.

```
> #Now our training set is ready for visualising
> # for finding optimal number of cluster we use albo result
> set.seed(123)
> # here we use set.seed  to get same result
> wcss<-vector()
> # where wcss is the within clustor sum of square
> for(i in 1:12){
+   wcss[i]<-sum(kmeans(X,i)$withinss)
+ }
>
> plot(1:12,wcss,main="Elbo method",xlab="Number of cluster",ylab="within cluster sum of Square",type="b",col=
>
> #by observing the our graph  k=5
>
```

After running above code we get-

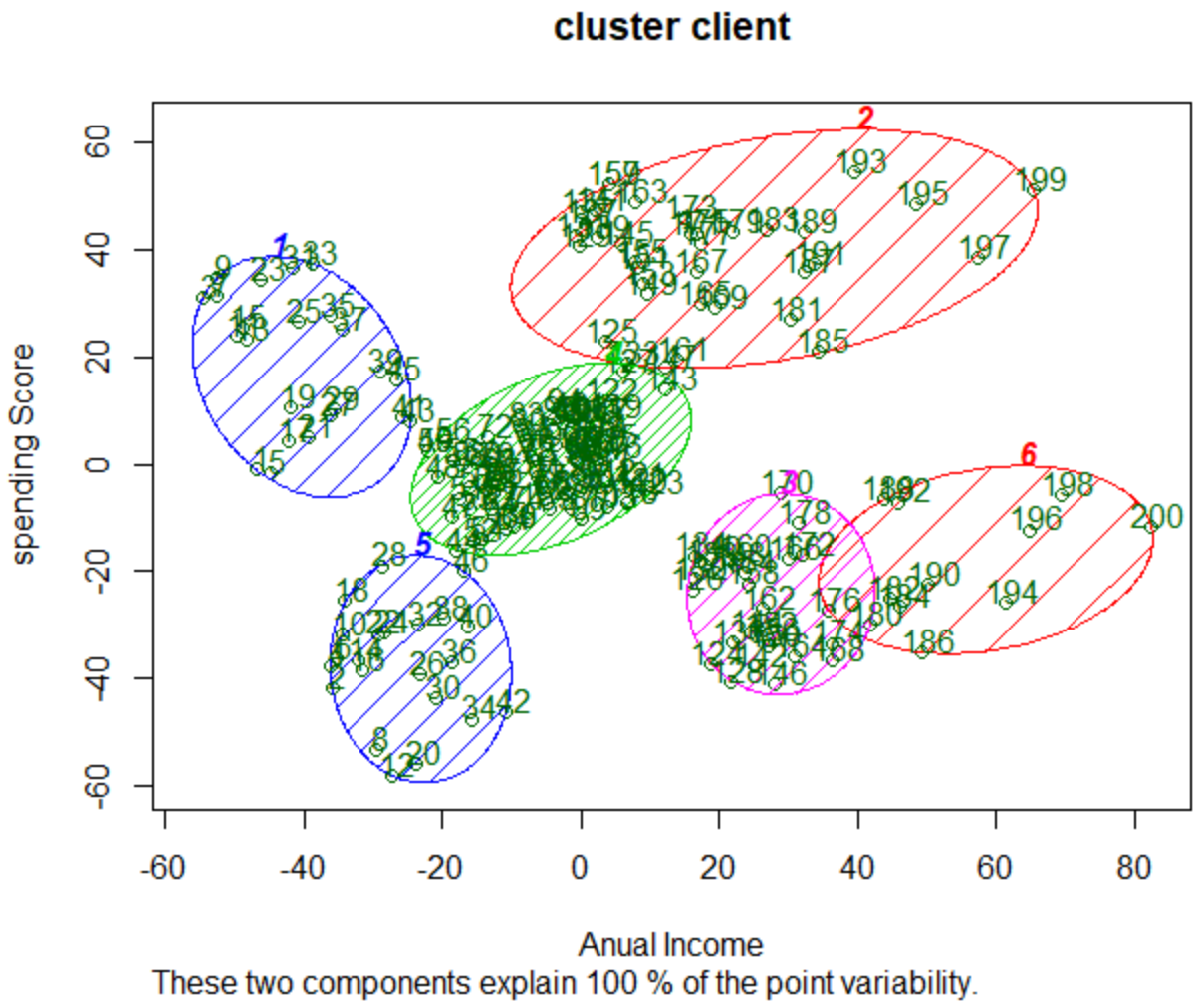


From above graph we choose k=7
Now we apply k-means algorithm on mall dataset by using**kmeans** fuction

which is available in R

```
> #by observing the our graph  k=6
>
>
> # Applying K- means algoritham on data set
> #to get same result we use seed fuction
>
> set.seed(124)
> k_means<-kmeans(X,6,iter.max=350,nstart = 12)
> |
```

we take maximum etiration 350
For visualising k-means clustering we use **clusplot** function which is available in **cluster** package in R
After applying clusplot function on k_means we get fallowing plot-

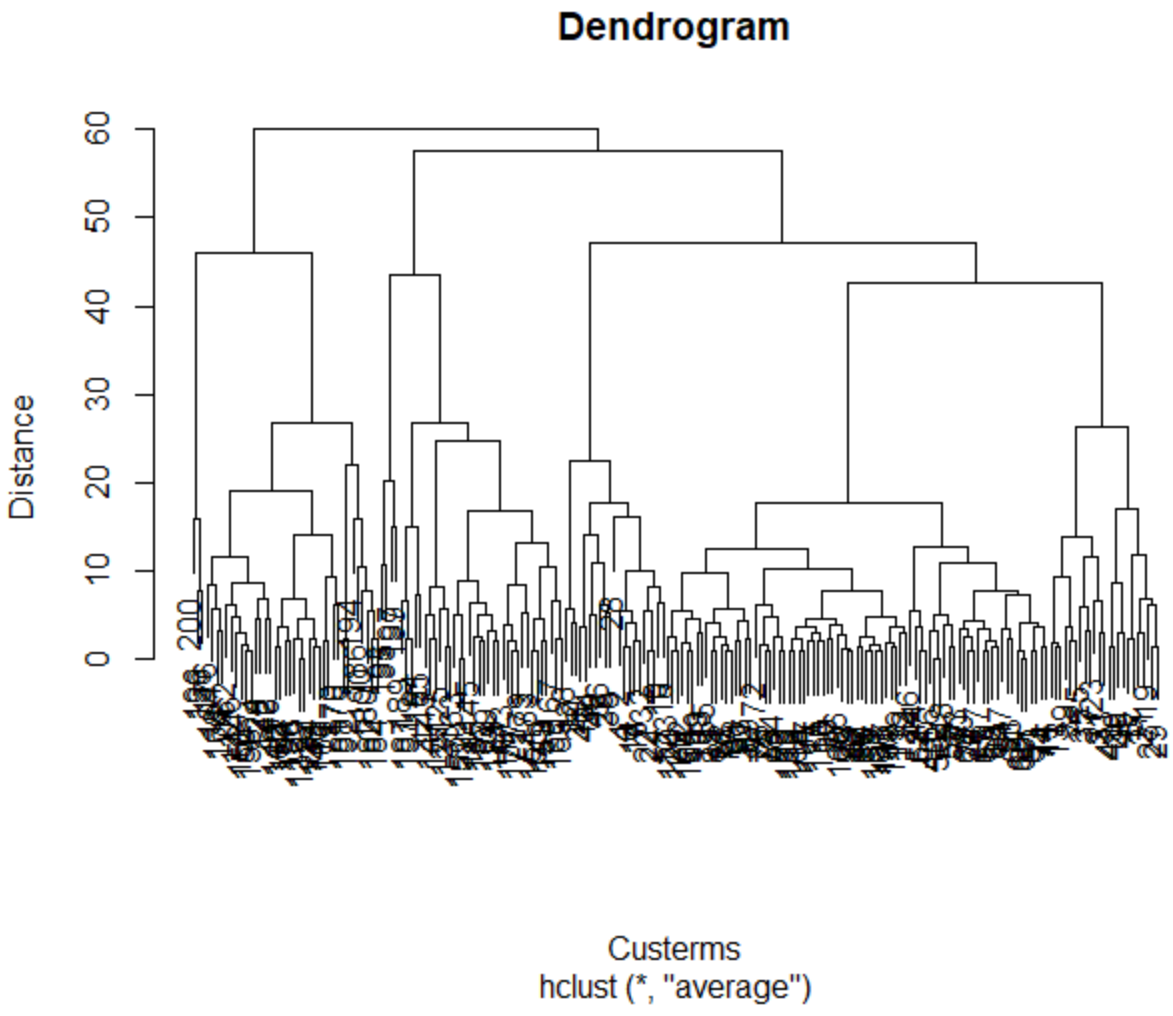


Hierarchical Clustering :

For obtaining the number of optimal cluster we used dendrogram tree
We used hclust function for finding the dendrogram plot.We used average method for finding the groups. Code for dendrogram are given below

```
> #hierarchical clustering
>  #for obtaining  the optimal number of cluster we use dendrogram
> dendo<-hclust(dist(X,method="euclidean"),method="average")
> plot(dendo,main="Dendrogram",xlab="Customers",ylab="Distance")
>
```

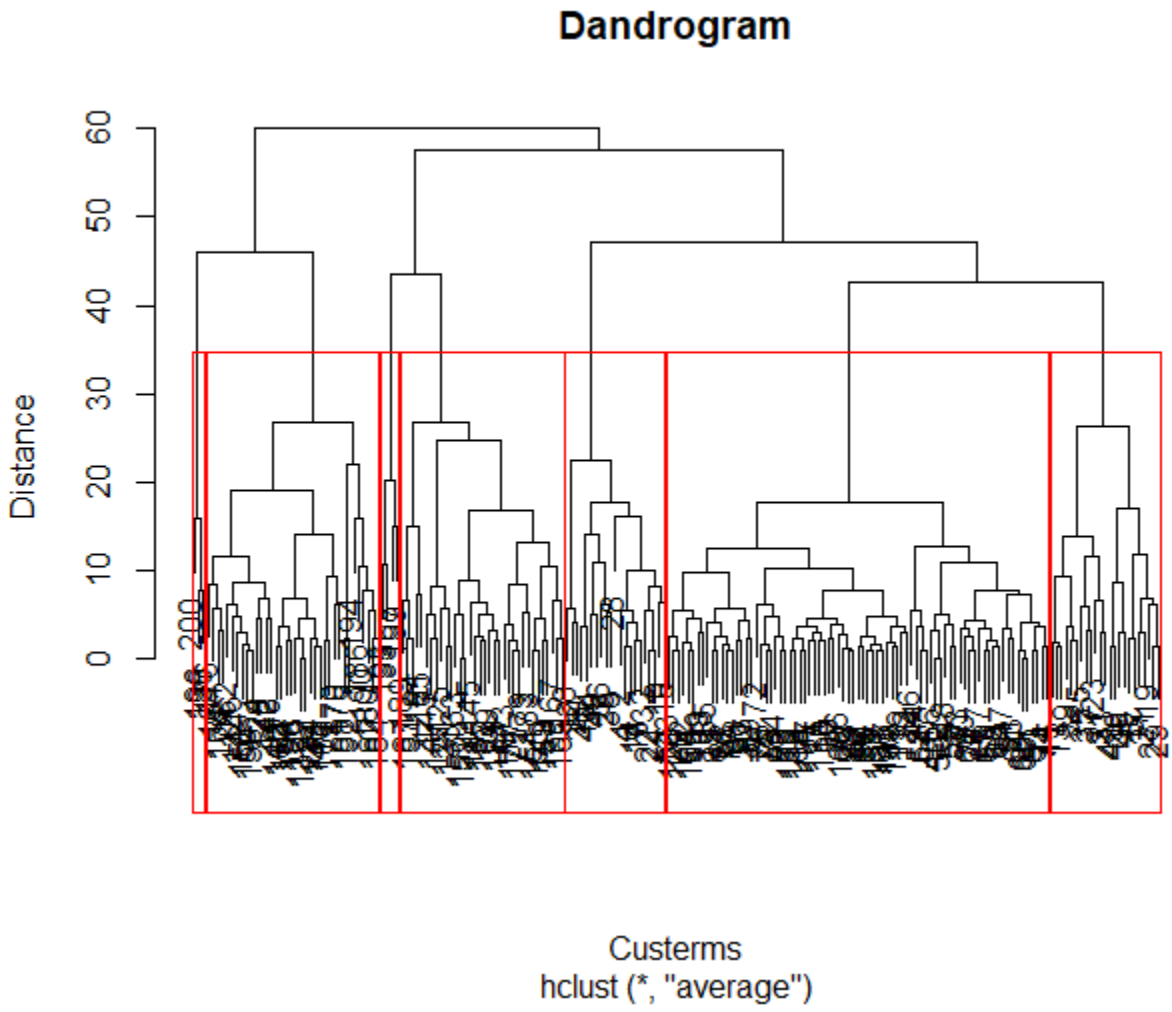
After applying this code we get



From the graph ,we take k=7 as optimal number of cluster

```
> rect.hclust(dendo,k=7)
>
```

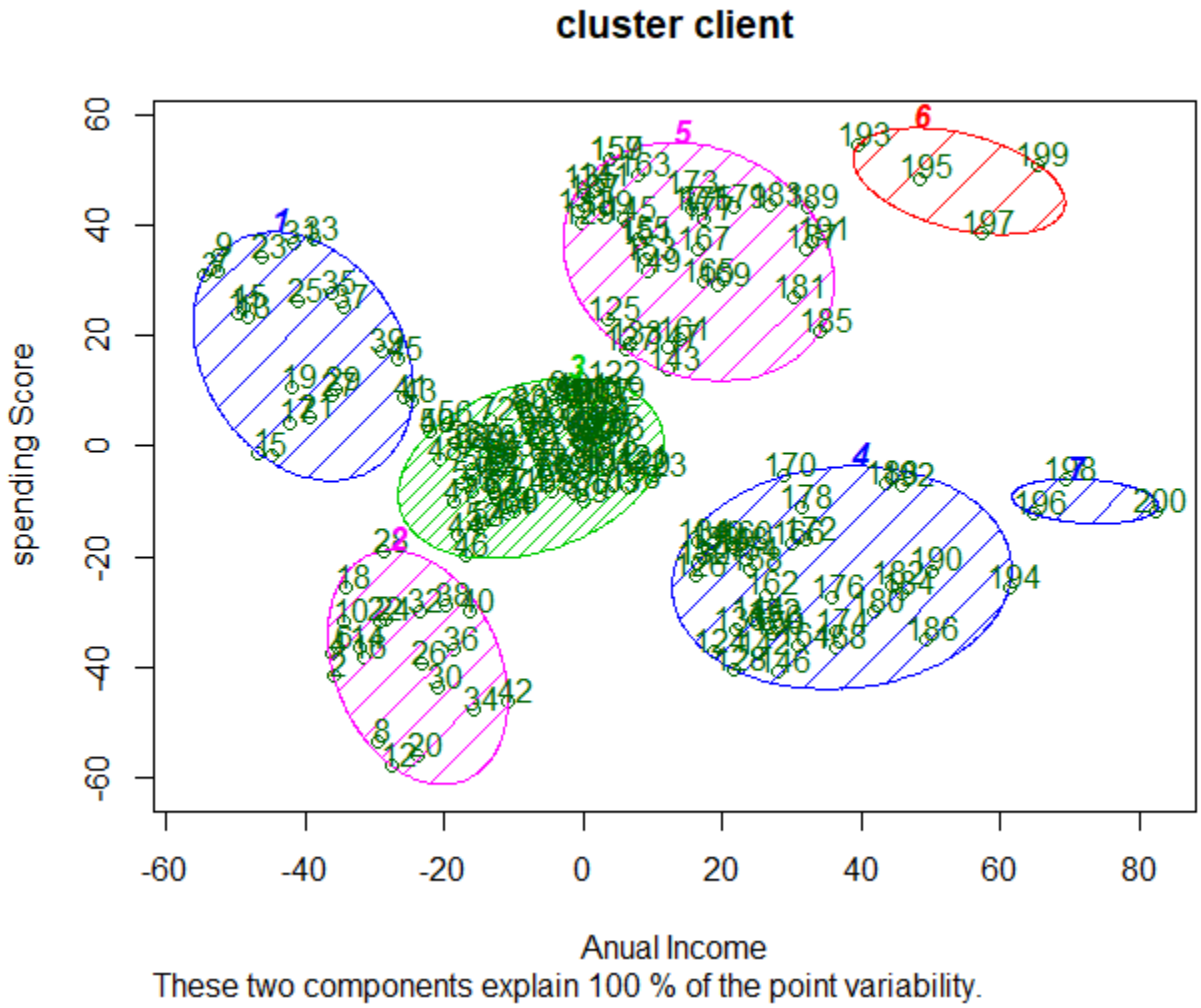
After applying rect.hclust function we get-



For visualization of cluster we will use clusplot function

```
> Y_hc<-cutree(dendo,k=7)
> #library("cluster")
>
> clusplot(X,
+          Y_hc,lines=0,shade=T,
+          labels=2,color=T,plotchar=F,span=T,
+          main="cluster client",
+          xlab="Anual Income",ylab="spending Score"
+ )
>
```

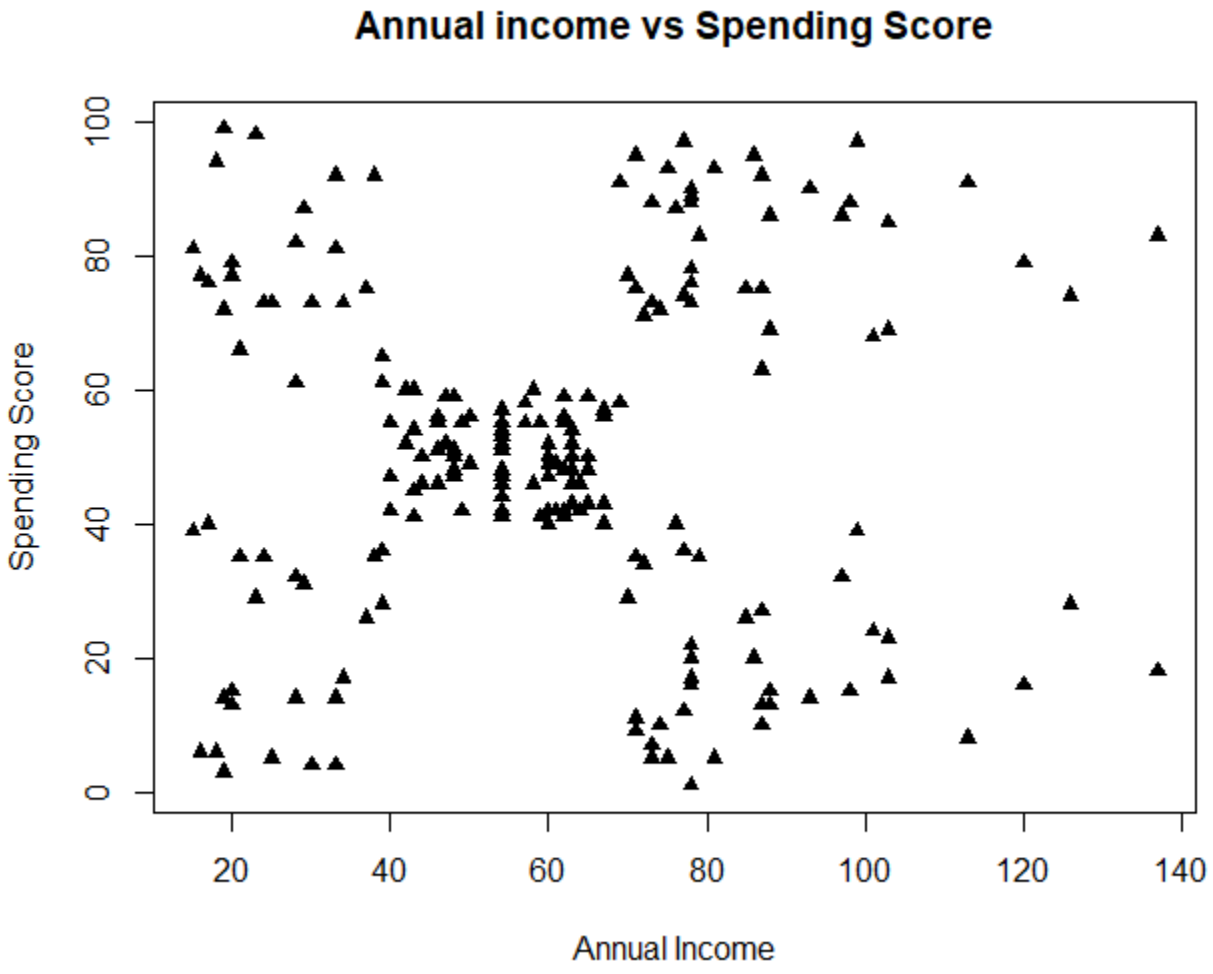
After applying the fallowing code we get-



Gaussian Mixture Model :

In case of gaussian mixture model we plot real data matrix first
Code are given below

```
> plot(X[,1],X[,2],main="Annual income vs Spending Score",xlab="Annual Income",ylab="Spending Score",pch=17)
>
```

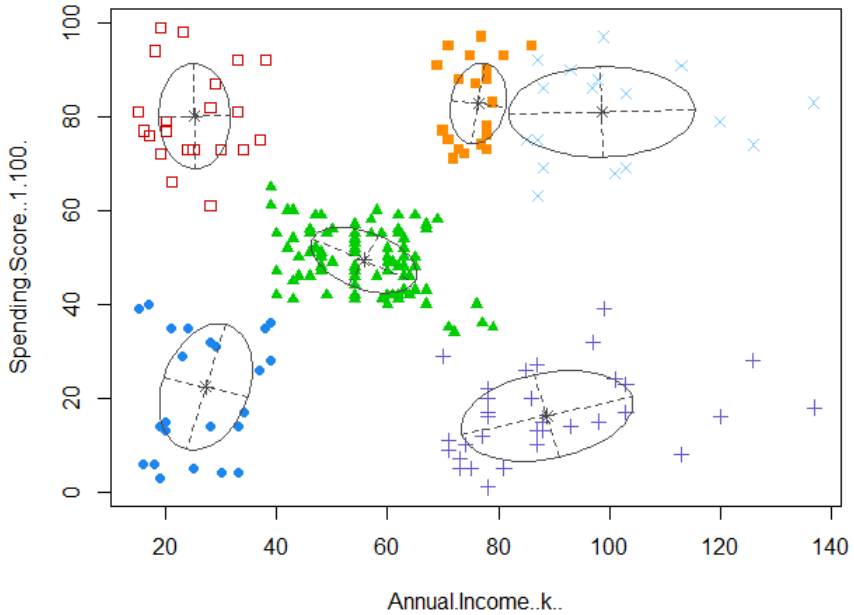


From above graph, we can assume that the given data are the mixture of six density function
For gaussian distribution we will use Mclust function which is available in mclust package in R
Code for gaussian distribution are given below

```
> #Gaussian Mixture dansity
> #install.packages("mclust")
> library("mclust")
    __  ___________    __  _____________
   /  |/  / ____/ /   / / / / ___/_  __/
  / /|_/ / /   / /   / / / /\__ \ / /
 / /  / / /___/ /___/ /_/ /___/ // /
/_/  /_/\____/_____/\____//____//_/    version 5.4.6
Type 'citation("mclust")' for citing this R package in publications.
Warning message:
package 'mclust' was built under R version 3.6.3
> fit<-Mclust(X,G=6,model="VEV")
fitting ...
  |=======================================================================================
> summary(fit)
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust VEV (ellipsoidal, equal shape) model with 6 components:

 log-likelihood   n df       BIC      ICL
      -1746.647 200 30 -3652.244 -3670.36

Clustering table:
 1  2  3  4  5  6
23 21 84 33 22 17
> plot(fit,what="classification")
>
```

After running this code we get fallowing result-

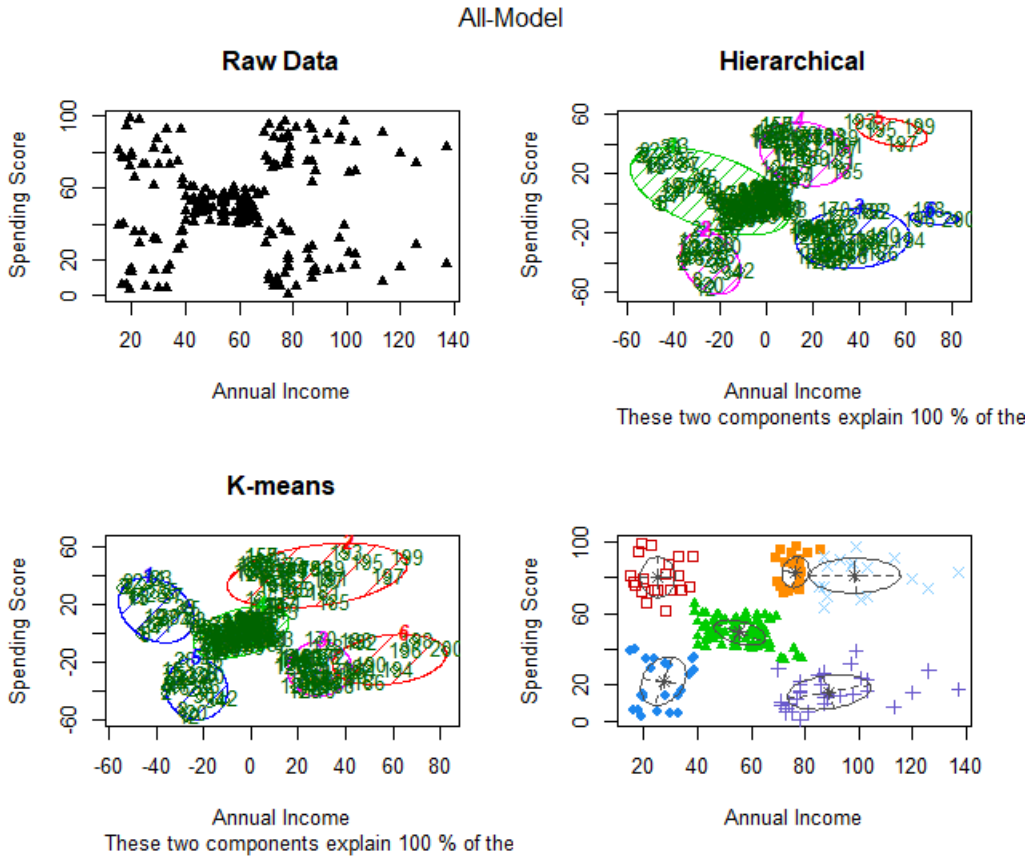
Gaussian Model



Now we will compare all the plot which comes from different algorithm Code are given below

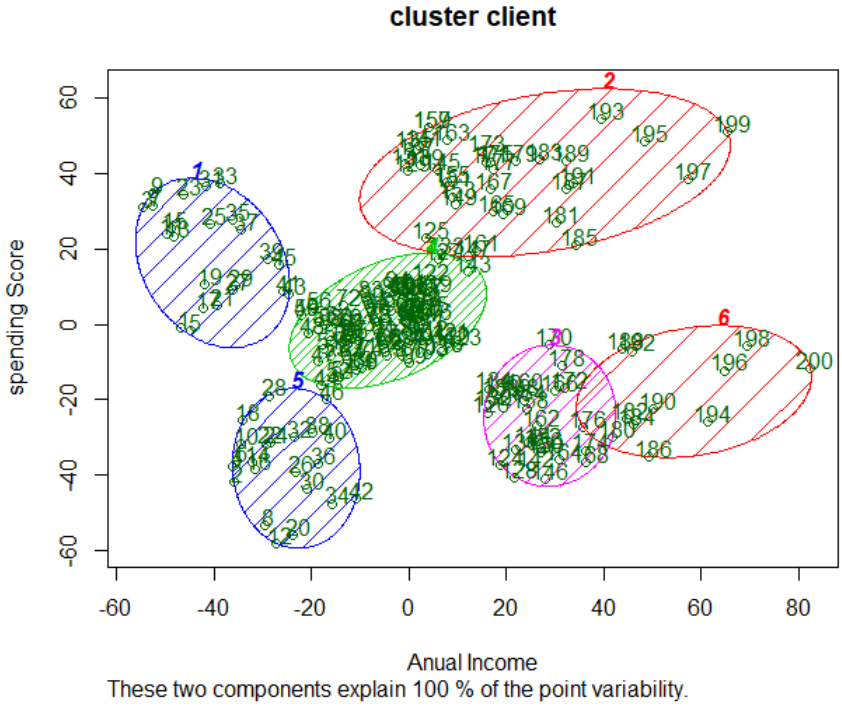
```
> par(mfrow=c(2,2))
>
> plot(X[,1],X[,2],main="Raw Data",xlab="Annual Income",ylab="Spending Score",pch=17)
> Y_hc<-cutree(dendo,k=6)
> clusplot(X,
+          Y_hc,lines=0,shade=T,
+          labels=2,color=T,plotchar=F,span=T,
+          main="Hierarchical",
+          xlab="Annual Income",ylab="Spending Score"
+ )
>
> clusplot(X,
+          k_means$cluster,lines=0,shade=T,
+          labels=2,color=T,plotchar=F,span=T,
+          xlab="Annual Income",ylab="Spending Score",
+          main="K-means"
+ )
> plot(fit,what="classification",xlab="Annual Income",ylab="Spending Score",main="Mixture-model")
> mtext("All-Model", side = 3, line = -1, outer = TRUE)
>
```

After running this code,we get fallowing results-



From above four plot we can say that the gaussian mixture model is better than the k-means and hierarchical clustring technique

Now we draw a conclusion about these cluster



1.Client in cluster 3 and 6 have high income and low spending score.In this cluster client earn high income but don't spend much money

2. Cluster 4 have average income and average spending score this cluster called standard clusters

3. Cluster 2 have high income and high spending score,So that is the cluster of clients that would be the main potential target of the mall marketing companies

4. Cluster 1 have low income and high spending score, called as careless cluster of client

5. Cluster 5 have low income and low spending score are called sensible cluster of client

Applications

Cluster analysis serve as a powerful tool for the unsupervised datasets. From customer segmentation to outlier detection, it has a broad range of uses and different techniques that fit into different cases. The most common algorithms in machine learning employed for the unsupervised data are Heirarchical Clustering and K-Means Clustering. Both of them are extremely powerful and applied in different scenarios to gain new insights from the data.

Moreover,In real life, many datasets can be modeled by Gaussian Distribution (Univariate or Multivariate). So it is quite natural and intuitive to assume that the clusters come from different Gaussian Distributions. Or in other words, it is tried to model the dataset as a mixture of several Gaussian Distributions. This is the core idea of GMM.

Some of the well known applications of the clustering are:

1. Cluster analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
2. Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
3. In the field of biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionalities and gain insight into structures inherent to populations.
4. Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location.
5. Clustering is also used in outlier detection applications such as detection of credit card frauds a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster.
6. As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster. 7. Clustering in identification of fake news, spam filter and so on.

8. Document analysis is another important use of Heirarchical clustering. Imagine you are limited in time and need to organize information held in documents quickly. To complete this ask you need to understand the theme of the text, compare it with other documents and classify it.
9. Clustering can also be used in fantasy Football and IPL matches. Who should you have in your team? Which players are going to perform best for your team and allow you to beat the competition? The challenge at the start of the season is that there is very little if any data available to help you identify the winning players.

GMMs have been used recently for feature extraction from speech data for use in speech recognition systems. They have also been used extensively in object tracking of multiple objects, where the number of mixture components and their means predict object locations at each frame in a video sequence. The EM algorithm is used to update the component means over time as the video frames update, allowing object tracking

As we can see, clustering plays a pivotal role in the Data Science and Analytics circle to gain insights from the data and covers a wide range of practical real life applications.

#R-Code

```
# k-means clustering
# first to check working directory
getwd()
# set working directory
setwd("C:/Users/abhil/OneDrive/Desktop/AI")
#Again check working changed or not
getwd()
#Now my working directory changed
#loading the data set
data<-read.csv("abhi.csv")
#checking data table
 head(data,10)
 #we are interested to analyzing Annual Income and Spending Score of our customer
#checking summary of data
data$CustomerID<-as.factor(data$CustomerID)
summary(data)
X<-data[,4:5]
head(X)
plot(X[,1],X[,2],main="Annual income vs Spending Score",xlab="Annual
Income",ylab="Spending Score",pch=17)
# For above picture we can say there 6 cluster roughly
#hierarchical clustering
 #for obtaining  the optimal number of cluster we use dendrogram
dendo<-hclust(dist(X,method="euclidean"),method="average")
plot(dendo,main="Dendrogram",xlab="Customers",ylab="Distance")
rect.hclust(dendo,k=7)
Y_hc<-cutree(dendo,k=7)
#library("cluster")
```

```
clusplot(X,
        Y_hc,lines=0,shade=T,
        labels=2,color=T,plotchar=F,span=T,
        main="cluster client",
        xlab="Anual Income",ylab="spending Score"
)
#by observing the our graph  k=6
# Applying K- means algoritham on data set
#to get same result we use seed fuction
#Gaussian Mixture dansity
#install.packages("mclust")
library("mclust")
fit<-Mclust(X,G=6,model="VEV")
summary(fit)
plot(fit,what="classification")
par(mfrow=c(2,2))
plot(X[,1],X[,2],main="Raw Data",xlab="Annual Income",ylab="Spending Score",pch=17)
Y_hc<-cutree(dendo,k=6)
clusplot(X,
        Y_hc,lines=0,shade=T,
        labels=2,color=T,plotchar=F,span=T,
        main="Hierarchical",
        xlab="Annual Income",ylab="Spending Score"
)
clusplot(X,
        k_means$cluster,lines=0,shade=T,
        labels=2,color=T,plotchar=F,span=T,
        xlab="Annual Income",ylab="Spending Score",
        main="K-means"
```

)

plot(fit,what="classification",xlab="Annual Income",ylab="Spending Score",main="Mixture-model")

mtext("All-Model", side = 3, line = -1, outer = TRUE)

#References

1. https://www.kdnuggets.com/2019/09/hierarchical-clustering.html
2. https://en.wikipedia.org/wiki/Hierarchical_clustering
3. https://www.displayr.com/what-is-hierarchical-clustering/
4. https://towardsdatascience.com/understanding-the-concept-of-hierarchical-clustering-technique-c6e8243758ec
5. https://sites.google.com/site/dataclusteringalgorithms/k-means-clustering-algorithm
6. https://home.deib.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html
7. https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/
8. https://www.geeksforgeeks.org/k-means-clustering-introduction/
9. https://towardsdatascience.com/introduction-to-image-segmentation-with-k-means-clustering-83fd0a9e2fc3
10. https://en.wikipedia.org/wiki/K-means_clustering
11. https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a
12. https://www.edureka.co/blog/k-means-clustering/

13 The Elements of Statistical Learning