Experiment No : 5
K-Means Clustering Technique

Aim: Write a program to illustrate the concept of K-Means Clustering

Algorithm.

Introduction:

K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity. The results of the K-means clustering algorithm are:

1. The centroids of the K clusters, which can be used to label new data

2. Labels for the training data (each data point is assigned to a single cluster)

Algorithm:

The algorithm clusters observations into k groups, where k is provided as an input parameter. It then assigns each observation to clusters based upon the observation's proximity to the mean of the cluster. The cluster's mean is then recomputed and the process begins again. Here's how the

algorithm works:

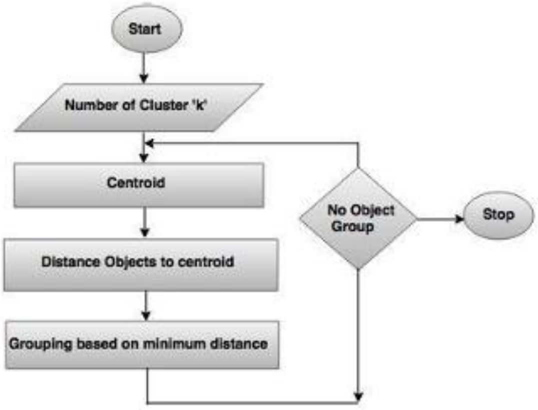
1. The algorithm arbitrarily selects k points as the initial cluster centers (the means).

2. Each point in the dataset is assigned to the closed cluster, based upon the Euclidean distance between each point and each cluster center.

3. Each cluster center is recomputed as the average of the points in that cluster.

4. Steps 2 and 3 repeat until the clusters converge. Convergence may be defined differently depending upon the implementation, but it normally means that either no observations change clusters when steps 2 and 3 are repeated, or that the changes do not make a material difference in the definition of the clusters .

Flowchart:


Start
Number of Cluster 'k'
Centroid
Distance Objects to centroid
Grouping based on minimum distance
No Object Group
Stop

K- mean Algorithm

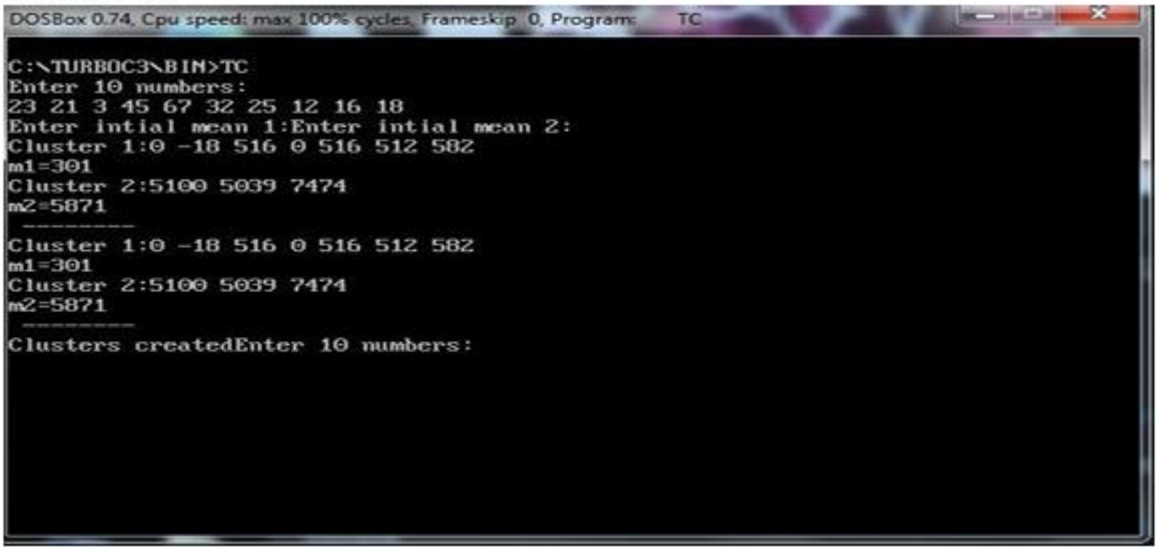
:

Program:

```
#include<stdio.h>
void main()
{
int i1,i2,i3,t1,t2,m1,m2,om1,om2;
int k0[10],k1[10],k2[10];
printf("Enter 10 numbers:");
for(i1=0;i1<10;i1++)
{
scanf("%d",&k0[i1]);
}
printf("Enter intial mean 1:");
scanf("%d",&m1);
printf("Enter intial mean 2:");
scanf("%d",&m2);
do
{
om1=m1;
om2=m2;
i1= i2=i3=0;
for(i1=0;i1<10;i1++)
{
t1=k0[i1]-m1;
if(t1<0)
{
t1=-t1;
}
t2=k0[i1]-m2;
if(t2<0)
{
t2=-t2;
}
if(t1<t2)
{
k1[i2]=k0[i1];
i2++;
}
else
{
k2[i3]=k0[i1];
i3++;
}
  }
t2=0;
for(t1=0;t1<i2;t1++)
{
t2=t2+k1[t1];
}
m1=t2/i2;
t2=0;
for(t1=0;t1<i3;t1++)
{
t2=t2+k2[t1];
}
m2=t2/i3;
printf("\nCluster 1:");
for(t1=0;t1<i2;t1++)
{
printf("%d ",k1[t1]);
}
printf("\nm1=%d",m1);
printf("\nCluster 2:");
for(t1=0;t1<i3;t1++)
{
printf("%d ",k2[t1]);
}
printf("\nm2=%d",m2);
printf("\n --------- ");
}
while(m1!=om1 && m2!=om2);
printf("\nClusters created");
}
```

OUTPUT:

OUTPUT:



DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC

```
C:\TURBOC3\BIN>TC
Enter 10 numbers:
23 21 3 45 67 32 25 12 16 18
Enter intial mean 1:Enter intial mean 2:
Cluster 1:0 -18 516 0 516 512 582
m1=301
Cluster 2:5100 5039 7474
m2=5871
 --------
Cluster 1:0 -18 516 0 516 512 582
m1=301
Cluster 2:5100 5039 7474
m2=5871
 --------
Clusters createdEnter 10 numbers:
```