WHAT IS CLUSTERING?

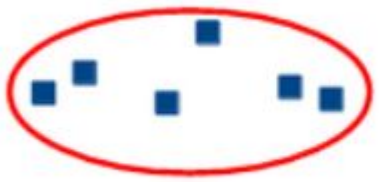
Clustering is the process of dividing the entire data into groups (also known as clusters) based on the patterns in the data.

Unsupervised learning

PROPERTIES OF CLUSTER:

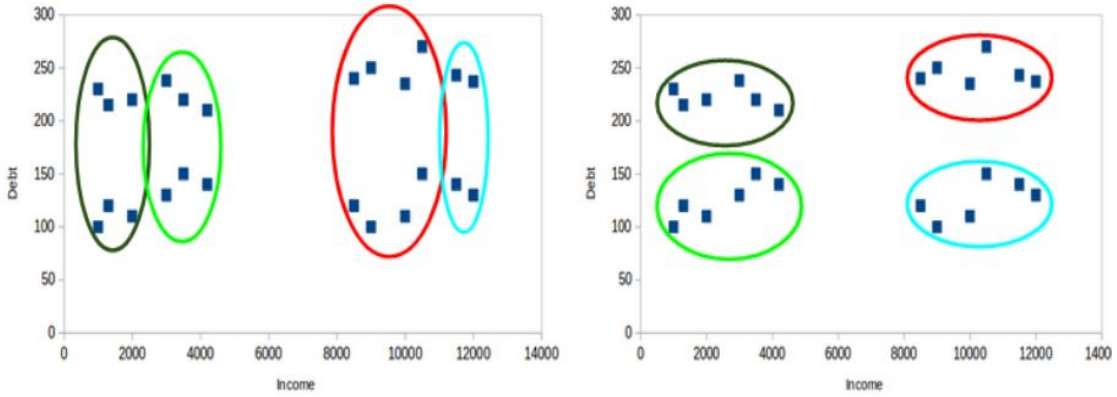
Property 1: All the data points in a cluster should be similar to each other.

This can be illustrated like this:



Property 2: The data points from different clusters should be as different as possible.

This can be illustrated like this:



APPLICATION OF CLUSTERING:

Customer Segmentation

Document Clustering

Image Segmentation

Recommendation Engines

Gene Expression

Disease classification

K-MEAN CLUSTERING:

It is an iterative algorithm that divides the unlabeled dataset into k different clusters in such a way that each dataset belongs only one group that has similar properties.

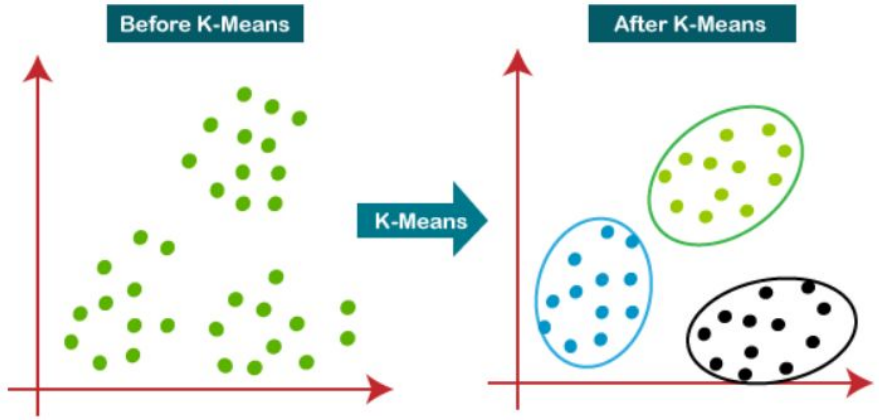
It is a centroid-based algorithm.

The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

K-MEAN CLUSTERING:

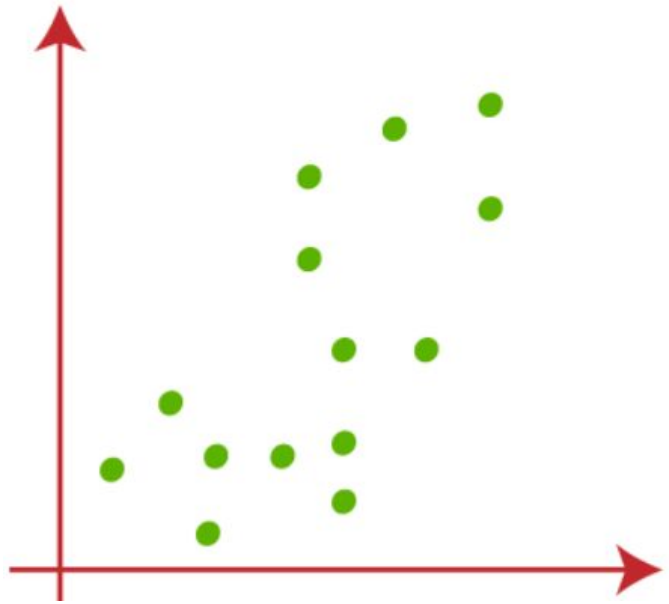
The k-means clustering algorithm mainly performs two tasks:

- Determines the best value for K center points or centroids by an iterative process.
- Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.



WORKING OF K-MEAN ALGORITHM:

Suppose we have two variables M1 and M2. The x-y axis scatter plot of these two variables is given below



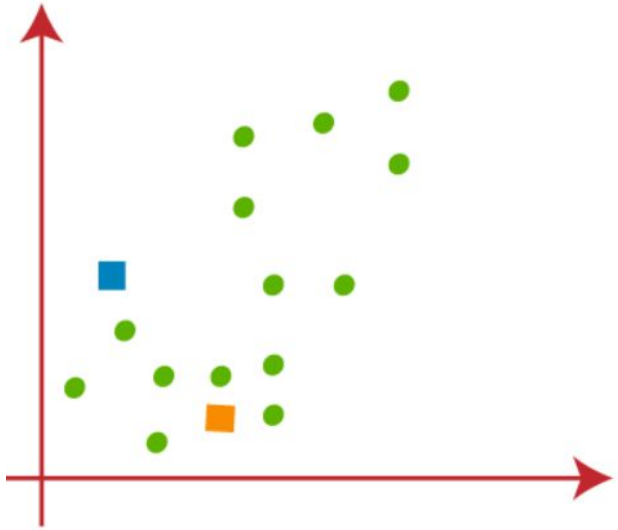
VOCABULARY OF 7 WORDS IN TWO DOCUMENTS

Documents (Data Points)	W1 (x-axis)	W2 (y-axis)
D1	2	0
D2	1	3
D3	3	5
D4	2	2
D5	4	6

WORKING OF K-MEAN ALGORITHM:

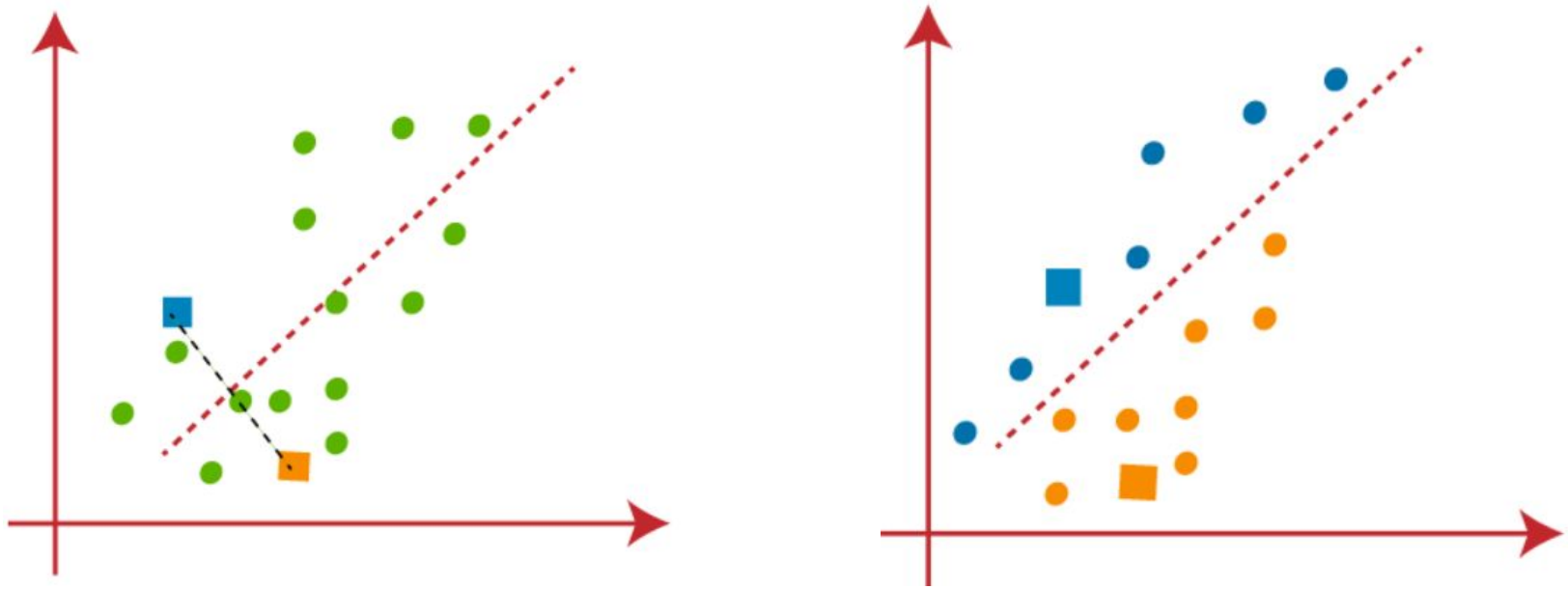
Let's take number k of clusters, i.e., K=2, to identify the dataset and to put them into different clusters.

We need to choose some random k points or centroid to form the cluster.



WORKING OF K-MEAN ALGORITHM:

Now we will assign each data point of the scatter plot to its closest K-point or centroid. calculate the distance between two points. we will draw a median between both the centroids.

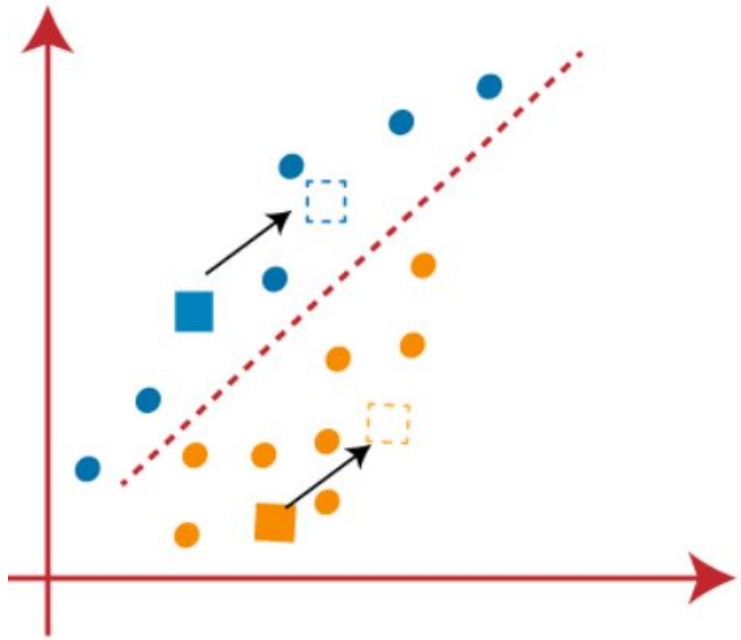


LET THE TWO CENTROIDS BE D2 AND D4

Distance between D1 and D2	Distance between D1 and D4
$\sqrt{(2-1)^2+(0-3)^2}$	$\sqrt{(2-2)^2+(0-2)^2}$
$= \sqrt{(1)^2+(3)^2}$	$= \sqrt{(0)^2+(-2)^2}$
$= \sqrt{1+9}$	$= \sqrt{0+4}$
$= \sqrt{10} = 3.17$	$= \sqrt{4} = 2$

WORKING OF K-MEAN ALGORITHM:

As we need to find the closest cluster, so we will repeat the process by choosing **a new centroid**. To choose the new centroids, we will compute the center of gravity of these centroids, and will find new centroids.



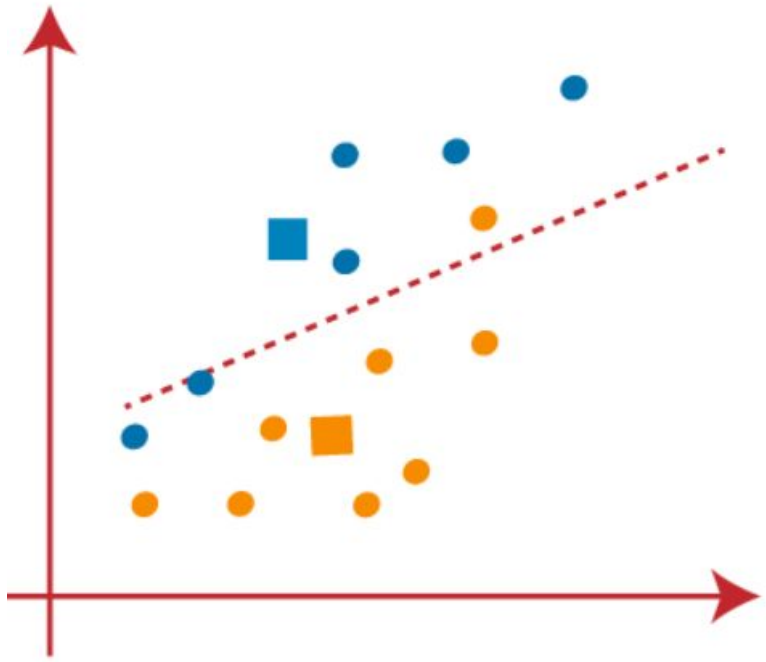
CENTRE OF GRAVITY

Find mean value along x1, x2……axes

CoG = $\{\overline{x_i}\}$

WORKING OF K-MEAN ALGORITHM:

Next, we will reassign each datapoint to the new centroid. For this, we will repeat the same process of finding a median line. The median will be like below image:

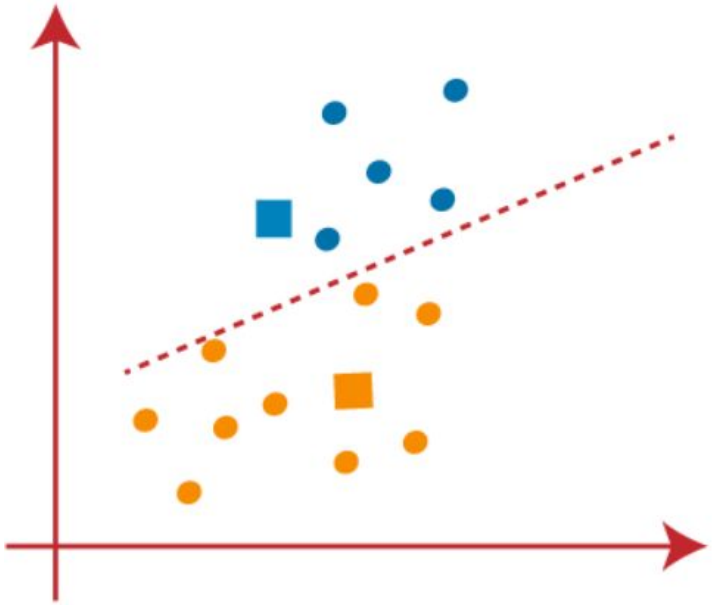


ALLOCATING DATA POINTS TO CLUSTERS

Documents (Data Points)	Distance between D2 and other data points	Distance between D4 and other data points
D1	3.17	2.0
D3	2.83	3.17
D5	4.25	4.48

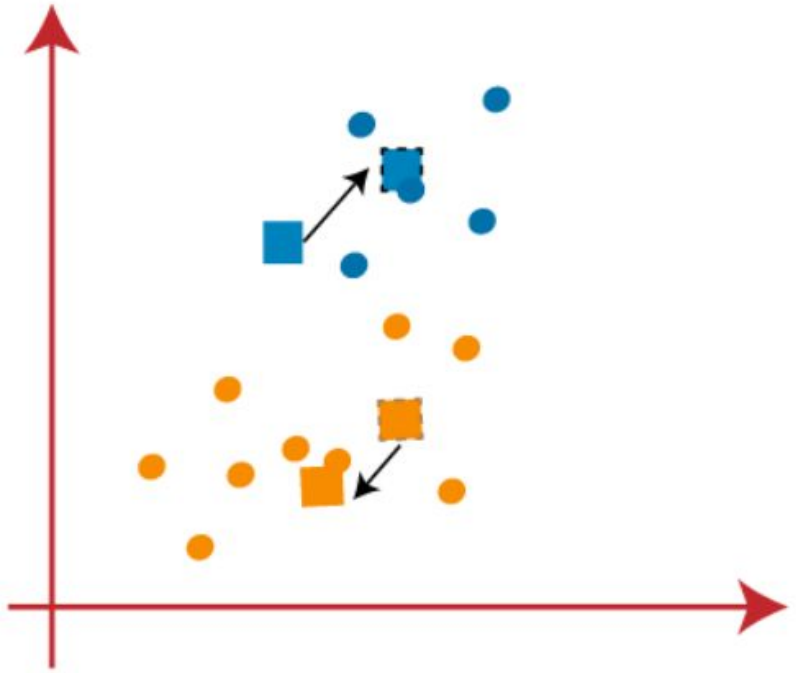
WORKING OF K-MEAN ALGORITHM:

From the previous image, we can see, one yellow point is on the left side of the line, and two blue points are right to the line. So, these three points will be assigned to new centroids.



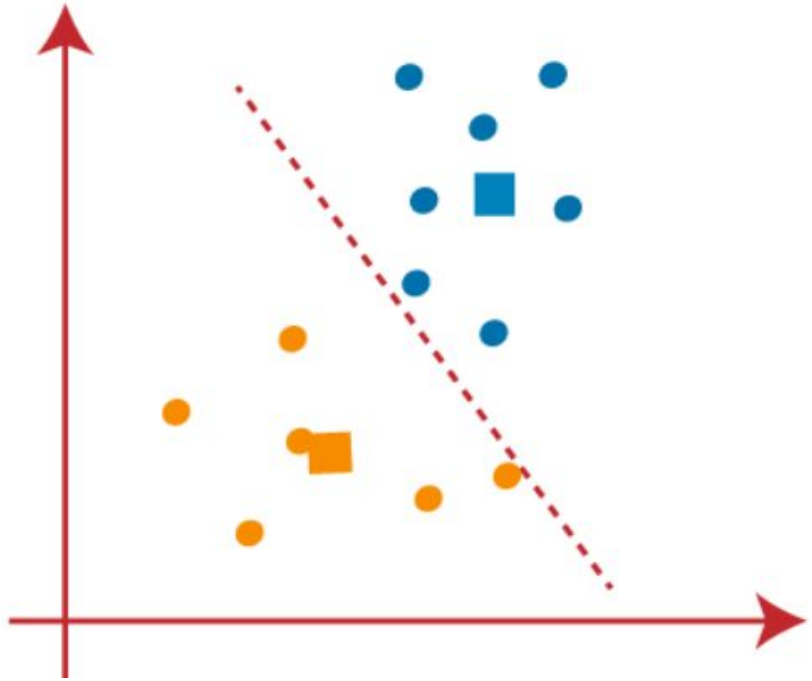
WORKING OF K-MEAN ALGORITHM:

We will repeat the process by finding the center of gravity of centroids, so the new centroids will be as shown in the below image:



WORKING OF K-MEAN ALGORITHM:

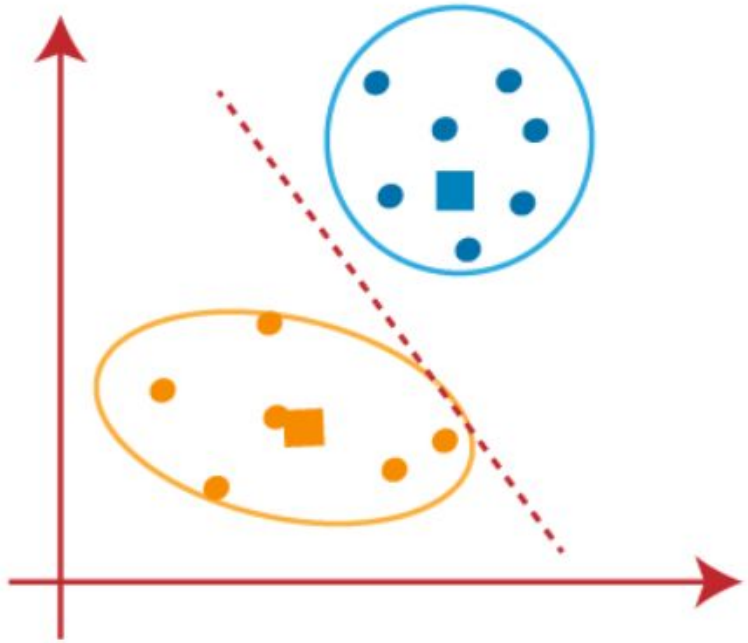
As we got the new centroids so again will draw the median line and reassign the data points.



Clusters	Mean value of data points along x-axis ……	Mean value of data points along y-axis ……
D1, D4	2.0	1.0
D2, D3, D5	2.67	4.67

WORKING OF K-MEAN ALGORITHM:

There are no dissimilar data points on either side of the line, which means our model is formed.



CHOOSING THE RIGHT NUMBER OF CLUSTERS:

Elbow Method:

Most popular ways to find the optimal number of clusters.

This method uses the concept of WCSS value.

WCSS stands for **Within Cluster Sum of Squares**, which defines the total variations within a cluster.

$$WCSS = \sum_{P_i \text{ in Cluster1}} distance(P_i\, C_1)^2 + \sum_{P_i \text{ in Cluster2}} distance(P_i\, C_2)^2 + \sum_{P_i \text{ in CLuster3}} distance(P_i\, C_3)^2$$

CHOOSING THE RIGHT NUMBER OF CLUSTERS:

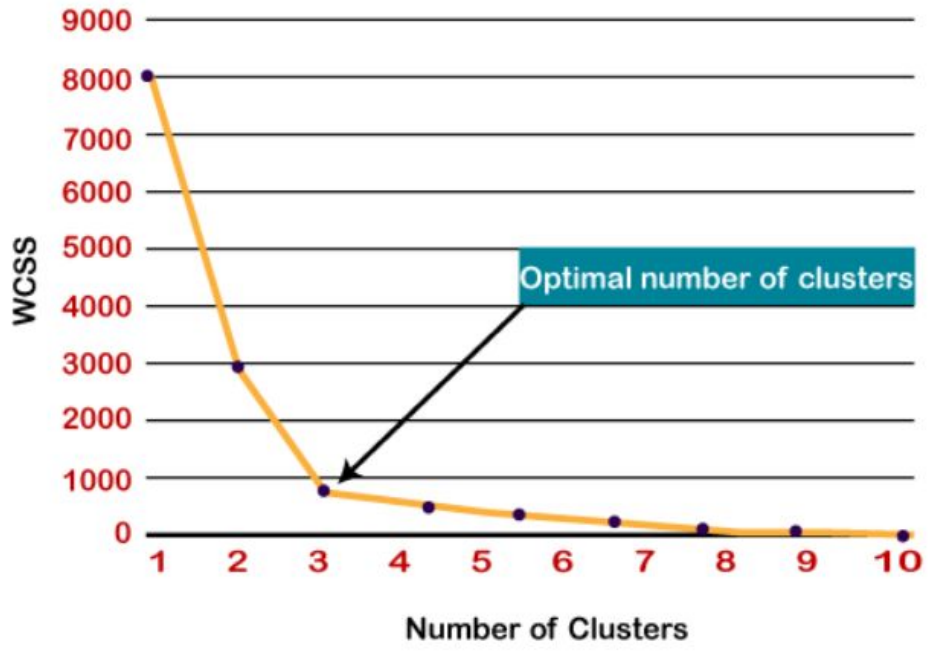
To measure the distance between data points and centroid, we can use any method such as Euclidean distance or Manhattan distance.

To find the optimal value of clusters, the elbow method follows the below steps:

- It executes the K-means clustering on a given dataset for different K values (ranges from 1-10).
- For each value of K, calculates the WCSS value.
- Plots a curve between calculated WCSS values and the number of clusters K.

CHOOSING THE RIGHT NUMBER OF CLUSTERS:

The sharp point of bend or a point of the plot looks like an arm, then that point is considered as the best value of K.



STOPPING CRITERIA FOR K-MEANS CLUSTERING:

1. Centroids of newly formed clusters do not change
2. Points remain in the same cluster
3. Maximum number of iterations are reached

IMPLEMENTATION:

About Dataset:

We have a dataset of **Mall_Customers**, which is the data of customers who visit the mall and spend there. In the given dataset, we have **Customer_Id, Gender, Age, Annual Income ($), and Spending Score** (which is the calculated value of how much a customer has spent in the mall, the more the value, the more he has spent). From this dataset, we need to calculate some patterns, as it is an unsupervised method.

IMPLEMENTATION:

```
# importing libraries
import numpy as nm
import matplotlib.pyplot as mtp
import pandas as pd
```

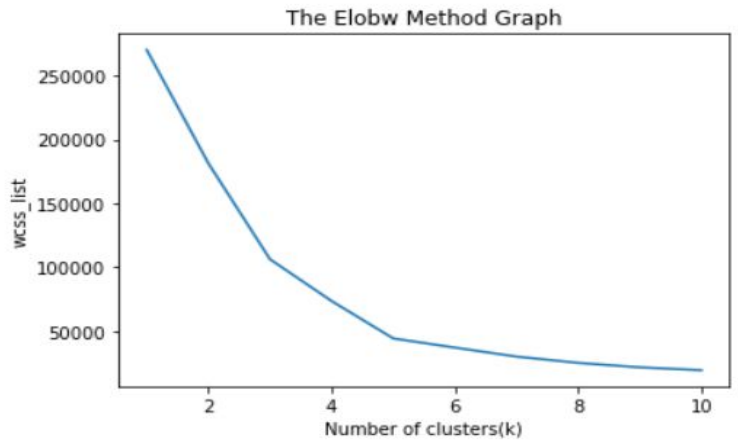
```
# Importing the dataset
dataset = pd.read_csv('Mall_Customers.csv')
```

```
x = dataset.iloc[:, [3, 4]].values
x
```

```
array([[ 15,  39],
       [ 15,  81],
       [ 16,   6],
       [ 16,  77],
       [ 17,  40],
       [ 17,  76],
       [ 18,   6],
       [ 18,  94],
       [ 19,   3],
       [ 19,  72],
       [ 19,  14],
       [ 19,  99],
       [ 20,  15],
       [ 20,  77],
       [ 20,  13],
       [ 20,  79],
       [ 21,  35],
       [ 21,  66],
       [ 23,  29],
       [ 23,  98],
```

IMPLEMENTATION:

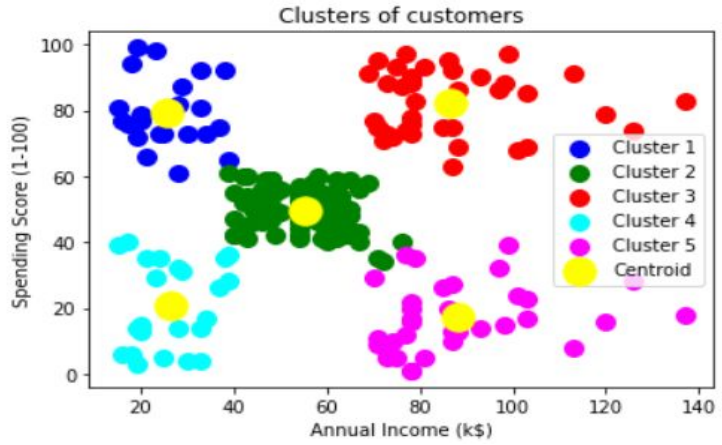
```
#finding optimal number of clusters using the elbow method
from sklearn.cluster import KMeans
wcss_list= []  #Initializing the list for the values of WCSS

#Using for loop for iterations from 1 to 10.
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', random_state= 42)
    kmeans.fit(x)
    wcss_list.append(kmeans.inertia_)
mtp.plot(range(1, 11), wcss_list)
mtp.title('The Elobw Method Graph')
mtp.xlabel('Number of clusters(k)')
mtp.ylabel('wcss_list')
mtp.show()
```



IMPLEMENTATION:

```
#training the K-means model on a dataset
kmeans = KMeans(n_clusters=5, init='k-means++', random_state= 42)
y_predict= kmeans.fit_predict(x)
```

```
#visulaizing the clusters
mtp.scatter(x[y_predict == 0, 0], x[y_predict == 0, 1], s = 100, c = 'blue', label = 'Cluster 1') #for first cluster
mtp.scatter(x[y_predict == 1, 0], x[y_predict == 1, 1], s = 100, c = 'green', label = 'Cluster 2') #for second cluster
mtp.scatter(x[y_predict== 2, 0], x[y_predict == 2, 1], s = 100, c = 'red', label = 'Cluster 3') #for third cluster
mtp.scatter(x[y_predict == 3, 0], x[y_predict == 3, 1], s = 100, c = 'cyan', label = 'Cluster 4') #for fourth cluster
mtp.scatter(x[y_predict == 4, 0], x[y_predict == 4, 1], s = 100, c = 'magenta', label = 'Cluster 5') #for fifth cluster
mtp.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 300, c = 'yellow', label = 'Centroid')
mtp.title('Clusters of customers')
mtp.xlabel('Annual Income (k$)')
mtp.ylabel('Spending Score (1-100)')
mtp.legend()
mtp.show()
```



RESULT:

The output image is clearly showing the five different clusters with different colors. The clusters are formed between two parameters of the dataset; Annual income of customer and Spending.

Cluster1 shows the customers with average salary and average spending.

- Cluster2 shows the customer has a high income but low spending, so we can categorize them as careful.
- Cluster3 shows the low income and also low spending so they can be categorized as sensible.
- Cluster4 shows the customers with low income with very high spending so they can be categorized as careless.
- Cluster5 shows the customers with high income and high spending so they can be categorized as target, and these customers can be the most profitable customers for the mall owner.