Date:05-01-2022

**Program - 9**

**Aim:**

Program to implement k-means clustering technique using any standard dataset available in the public domain

**Program:**

import numpy as nm

import matplotlib.pyplot as mtp

import pandas as pd

dataset = pd.read\_csv('Mall\_Customers.csv')

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

print(x)

from sklearn.cluster import KMeans

wcss\_list=[]

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 Elbow Method Graph')

mtp.xlabel('Number of clusters(k)')

mtp.ylabel('wcss\_list')

mtp.show()

kmeans=KMeans(n\_clusters=5,init='k-means++',random\_state=42)

y\_predict=kmeans.fit\_predict(x)

print(y\_predict)

mtp.scatter(x[y\_predict ==0,0],x[y\_predict ==0,1],s=100,c='blue',label='cluster 1')

mtp.scatter(x[y\_predict ==1,0],x[y\_predict ==1,1],s=100,c='green',label='cluster 2')

mtp.scatter(x[y\_predict ==2,0],x[y\_predict ==2,1],s=100,c='red',label='cluster 3')

mtp.scatter(x[y\_predict ==3,0],x[y\_predict ==3,1],s=100,c='cyan',label='cluster 4')

mtp.scatter(x[y\_predict ==4,0],x[y\_predict ==4,1],s=100,c='magenta',label='cluster 5')

mtp.scatter(kmeans.cluster\_centers\_[:,0],kmeans.cluster\_centers\_[:,1],s=300,c='black',label='cluster')

mtp.title('Clusters of customers')

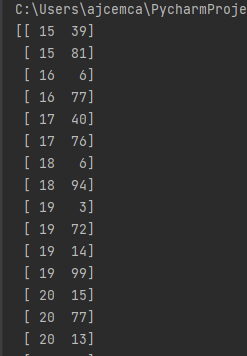
mtp.xlabel('Annual Income (K$)')

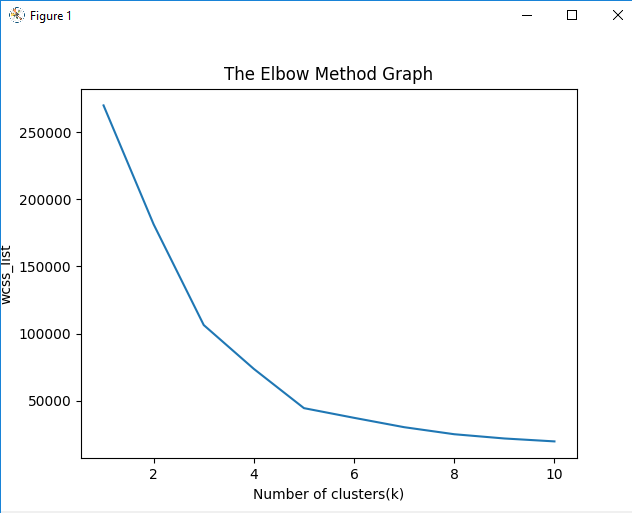
mtp.ylabel('Spending Score(1-100)')

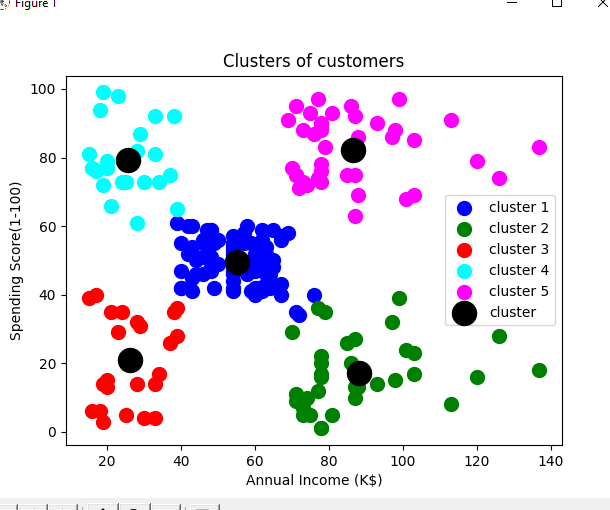
mtp.legend()

mtp.show()

**OUTPUT**

****

****

****

**Program** :

import numpy as nm

import matplotlib.pyplot as mtp

import pandas as pd

dataset = pd.read\_csv('world\_country\_and\_usa\_states\_latitude\_and\_longitude\_values.csv')

x=dataset.iloc[:,[1,2]].values

print(x)

from sklearn.cluster import KMeans

wcss\_list=[]

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 Elbow Method Graph')

mtp.xlabel('Number of clusters(k)')

mtp.ylabel('wcss\_list')

mtp.show()

kmeans=KMeans(n\_clusters=3,init='k-means++',random\_state=42)

y\_predict=kmeans.fit\_predict(x)

print(y\_predict)

mtp.scatter(x[y\_predict ==0,0],x[y\_predict ==0,1],s=100,c='blue',label='cluster 1')

mtp.scatter(x[y\_predict ==1,0],x[y\_predict ==1,1],s=100,c='green',label='cluster 2')

mtp.scatter(x[y\_predict ==2,0],x[y\_predict ==2,1],s=100,c='red',label='cluster 3')

mtp.scatter(kmeans.cluster\_centers\_[:,0],kmeans.cluster\_centers\_[:,1],s=300,c='black',label='cluster')

mtp.title('Clusters of customers')

mtp.xlabel('Annual Income (K$)')

mtp.ylabel('Spending Score(1-100)')

mtp.legend()

mtp.show()

**OUTPUT**

