Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

**08**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 01 | Write a python program which can read any dataset and implement KMEAN for that set  plot your result |
| 02 | Write a python program to implement kmean for 3 features |
|  |  |

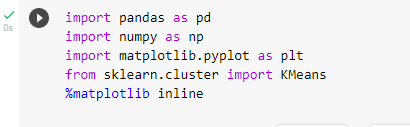
Submitted On:

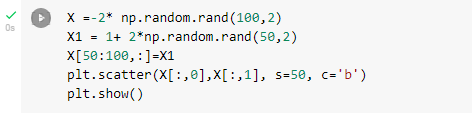
**Date: 10 JUNE 2022**

**EXAMPLE:**

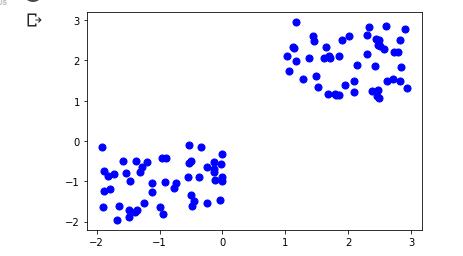
*“IMPLEMENTATION OF KMEANS”*

*Solution:*

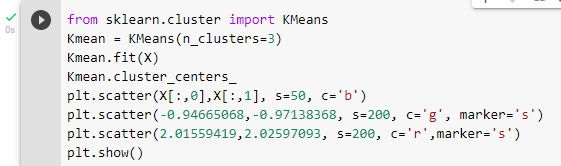




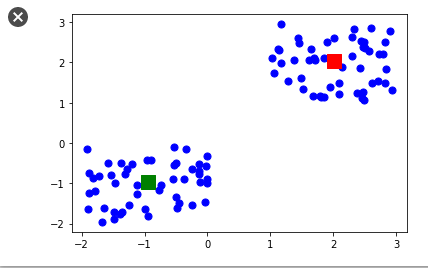
*Output:*



*Solution:*



*Output:*



**Task No. 1:**

Write a python program which can read any dataset and implement KMEAN for that set

plot your result

*Solution:*

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

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

X=dataset.iloc[:, [3,4]].values

from sklearn.cluster import KMeans

wcss=[]

for i in range(1,11):

     kmeans = KMeans(n\_clusters=i, init ='k-means++', max\_iter=300,  n\_init=10,random\_state=0 )

kmeans.fit(X)



wcss.append(kmeans.inertia\_)

plt.plot(range(1,11),wcss)

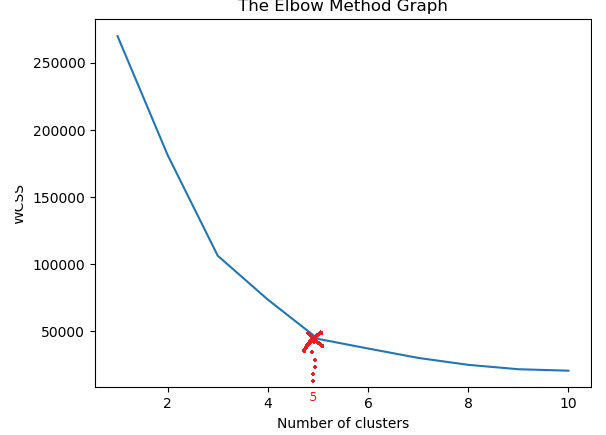
plt.title('The Elbow Method Graph')

plt.xlabel('Number of clusters')

plt.ylabel('WCSS')

plt.show()

*Output:*





*Solution:*

# According to the Elbow graph we deterrmine the clusters number as 5. Applying k-means algorithm to the X dataset.

kmeans = KMeans(n\_clusters=5, init ='kmeans++', max\_iter=300, n\_init=10,random\_state=0 )

y\_kmeans = kmeans.fit\_predict(X)

# Visualising the clusters

plt.scatter(X[y\_kmeans==0, 0], X[y\_kmeans==0, 1], s=100, c='red', label ='Cluster 1')

plt.scatter(X[y\_kmeans==1, 0], X[y\_kmeans==1, 1], s=100, c='blue', label ='Cluster 2')

plt.scatter(X[y\_kmeans==2, 0], X[y\_kmeans==2, 1], s=100, c='green', label ='Cluster 3')

plt.scatter(X[y\_kmeans==3, 0], X[y\_kmeans==3, 1], s=100, c='cyan', label ='Cluster 4')

plt.scatter(X[y\_kmeans==4, 0], X[y\_kmeans==4, 1], s=100, c='magenta', label ='Cluster 5')

#Plot the centroid. This time we're going to use the cluster centres  #attribute that returns here the coordinates of the centroid.

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

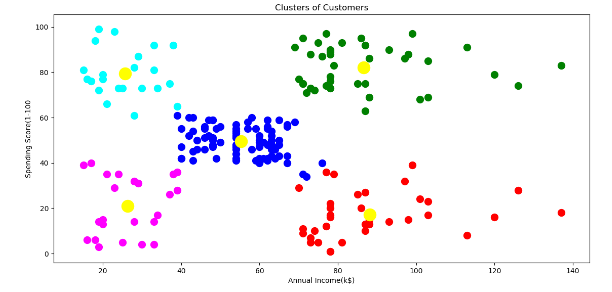
plt.title('Clusters of Customers')

plt.xlabel('Annual Income(k$)')

plt.ylabel('Spending Score(1-100')

plt.show()

*Output:*



**Task No. 2:**

Write a python program to implement kmean for 3 features

*Solution:*

from sklearn.datasets import load\_digits

from sklearn.decomposition import PCA

from sklearn.cluster import KMeans

import numpy as np

import matplotlib.pyplot as plt

#Loading Data

data = load\_digits().data

pca = PCA(3)

#Transforming the data

df = pca.fit\_transform(data)

from yellowbrick.cluster import KElbowVisualizer

model = KMeans()

visualizer = KElbowVisualizer(model, k=(1,12)).fit(df)

visualizer.show()

*Output:*

Chart, line chart

Description automatically generated

*Solution:*

from sklearn.cluster import KMeans

#Initialize the class object

kmeans = KMeans(n\_clusters= 2)

#predict the labels of clusters.

label = kmeans.fit\_predict(df)

#Getting unique labels

u\_labels = np.unique(label)

#plotting the results:

for i in u\_labels:

    plt.scatter(df[label == i , 0] , df[label == i , 1], df[label == i , 2]  , label = i)

plt.legend()

plt.show()

*Output:*

Chart, scatter chart

Description automatically generated