

Experiment–8

Aim:

Program to implement K-Means clustering techniques.

Code:

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from scipy.cluster.hierarchy import dendrogram, linkage
4 from sklearn.datasets import make_blobs
5 from sklearn.cluster import AgglomerativeClustering
6 from sklearn.cluster import DBSCAN
7 from sklearn.cluster import KMeans
8 from sklearn.datasets import make_moons
9
10
11 x,y=make_blobs(n_samples=150,n_features=2,centers=3,cluster_std=0.5,
12
13 shuffle=True,random_state=0)
14
15 plt.scatter(x[:,0],x[:,1],c='white',edgecolor='black',marker='o',s=50)
16 plt.show()
17
18 km=KMeans(n_clusters=3,init='random',max_iter=100,n_init=10)
19 y_km=km.fit_predict(x)
20 print('Distortion=',km.inertia_)
21 print(km.cluster_centers_)
22
23 plt.figure()
24 plt.scatter(
25     x[y_km==0,0], x[y_km==0,1], s=50,
26     c='lightgreen', marker='d', edgecolor='black',
27     label='cluster-1'
28 )
29 plt.scatter(
30     x[y_km==1,0], x[y_km==1,1], s=50,
31     c='orange', marker='o', edgecolor='black',
32     label='cluster-2'
33 )
34 plt.scatter(
35     x[y_km==2,0], x[y_km==2,1], s=50,
36     c='blue', marker='>', edgecolor='black',
37     label='cluster-3'
38 )
39 plt.scatter(
40     km.cluster_centers_[0,0], km.cluster_centers_[0,1], s=100,
41     c='red', marker='*', edgecolor='black',
```