Experiment 10: Implement K-means Clustering algorithm and evaluate its performance

Theory:

1. K-means Clustering Algorithm

Cluster analysis is the assignment of a set of observations into subsets (called clusters) so that observations within the same cluster are similar according to some pre designated criterion or criteria, while observations drawn from different clusters are dissimilar. Different clustering techniques make different assumptions on the structure of the data, often defined by some similarity metric and evaluated for example by internal compactness (similarity between members of the same cluster) and separation between different clusters. Other methods are based on estimated density and graph connectivity. Clustering is a method of unsupervised learning, and a common technique for statistical data analysis.

K-Means Algorithm: Steps

1. Load data set
2. Clusters the data into k groups where k is predefined.
3. Select k points at random as cluster centers.
4. Assign objects to their closest cluster center according to the Euclidean distance function.
5. Calculate the centroid or mean of all objects in each cluster.
6. Repeat steps 3, 4 and 5 until the same points are assigned to each cluster in consecutive rounds.

Implementation:

1. sklearn.cluster.KMeans

Syntax:

class sklearn.cluster.**KMeans**(*n_clusters=8*, ***, *init='k-means++'*, *n_init='warn'*, *max_iter=300*, *tol=0.0001*, *verbose=0*, *random_state=None*, *copy_x=True*, *algorithm='lloyd'*) ¶

Implements K-Means clustering.

2. **kmeans.labels_** - Labels of each point
3. **kmeans.cluster_centers_** - Coordinates of cluster centers
4. **kmeans.inertia_** - Sum of squared distances of samples to their closest cluster center, weighted by the sample weights if provided.

About Dataset:

(Describe your dataset)

Conclusion: In this way, we implemented the K-Means clustering algorithm and visualized the clusters.

****