Experiment 8

Perform K-Means clustering for customer segmentation.

Aim: Perform K-Means clustering for customer segmentation.

Theory

What is K-Means Clustering?

- **K-Means clustering** is an unsupervised machine learning algorithm used to group unlabeled data points into distinct clusters based on similarity.
- Each customer is assigned to one of the k clusters, where k is predefined, so that each group consists of customers with similar attributes or behaviors.

How K-Means Works:

1. **Initialization**: Randomly select k initial centroids (cluster centers).
2. **Assignment**: Assign each data point (customer) to the nearest centroid using a distance metric (typically Euclidean distance).
3. **Update**: Recalculate the centroids as the mean of all points assigned to each cluster.
4. **Iterate**: Repeat the assignment and update steps until the centroids no longer move significantly or a maximum number of iterations is reached.

   The result is a set of clusters where each member of a cluster is more similar to others in the same group than to members of other groups.

Choosing the Right Number of Clusters

- The **Elbow Method** is typically used to determine the optimal number of clusters (k). It involves plotting the sum of squared distances within clusters (inertia) and selecting the point where the decrease slows down (the "elbow") as the ideal k value.

Customer Segmentation Application

- By applying K-Means to customer datasets (e.g., age, income, spending score), businesses can:
    - Identify distinct customer segments.
    - Develop targeted marketing strategies.

- Improve product recommendations and customer service.

Conclusion

- **K-Means clustering** effectively segments customers into groups based on similar features such as age, income, and spending habits.
- This technique allows businesses to better understand their customer base, enabling more personalized marketing and strategic decision-making.
- The simple yet powerful implementation provided by sklearn streamlines the clustering process, from feature scaling and optimal k selection to model fitting and interpretation.

Viva Questions

- Why is K-Means considered an unsupervised algorithm?
- Describe the main steps of the K-Means algorithm.
- How can you determine the optimal number of clusters (k) for customer segmentation?
- Why is feature scaling important in K-Means?
- What are some real-life applications of customer segmentation using K-Means?

Program:

```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt

# Load dataset (assuming mall_customers.csv is available)
data = pd.read_csv('Mall_Customers.csv')

# Select features for clustering
features = data[['Age', 'Annual Income (k$)', 'Spending Score (1-100)']]

# Feature scaling
scaler = StandardScaler()
scaled_features = scaler.fit_transform(features)

# Using Elbow Method to choose k
inertia = []
for k in range(1, 11):
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(scaled_features)
    inertia.append(kmeans.inertia_)

plt.plot(range(1, 11), inertia, marker='o')
plt.xlabel('Number of clusters (k)')
plt.ylabel('Inertia')
```

```
plt.title('Elbow Method For Optimal k')
plt.show()

# Fit KMeans for chosen k (e.g., k=5)
kmeans = KMeans(n_clusters=5, random_state=42)
data['Cluster'] = kmeans.fit_predict(scaled_features)

print(data.groupby('Cluster').mean())
```