EX.NO: 7
16/9/25

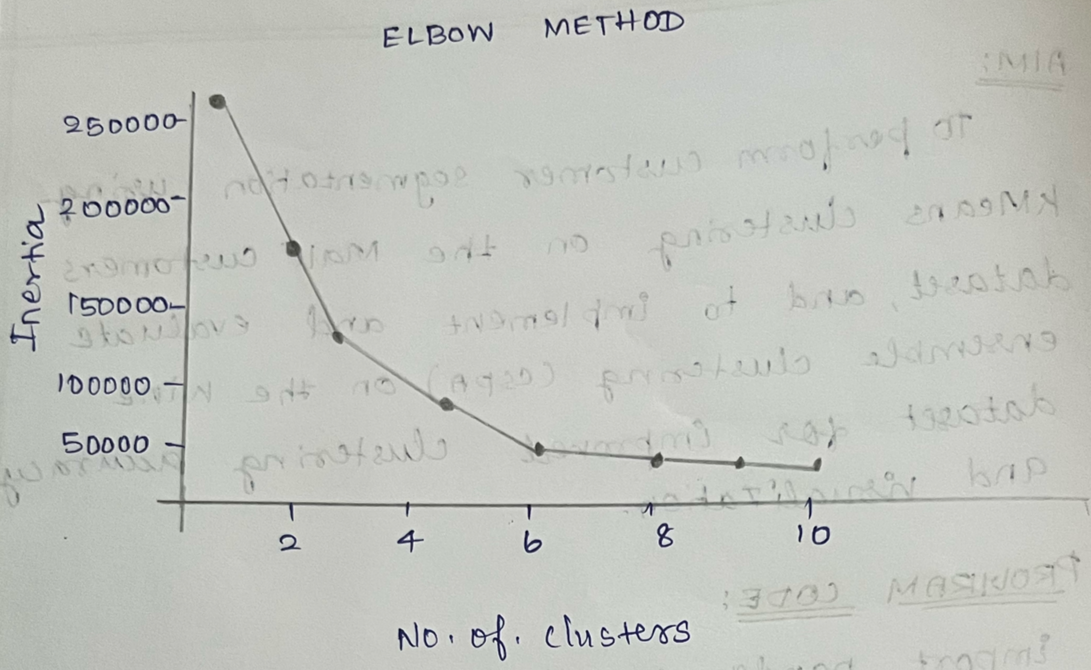
CLUSTERING

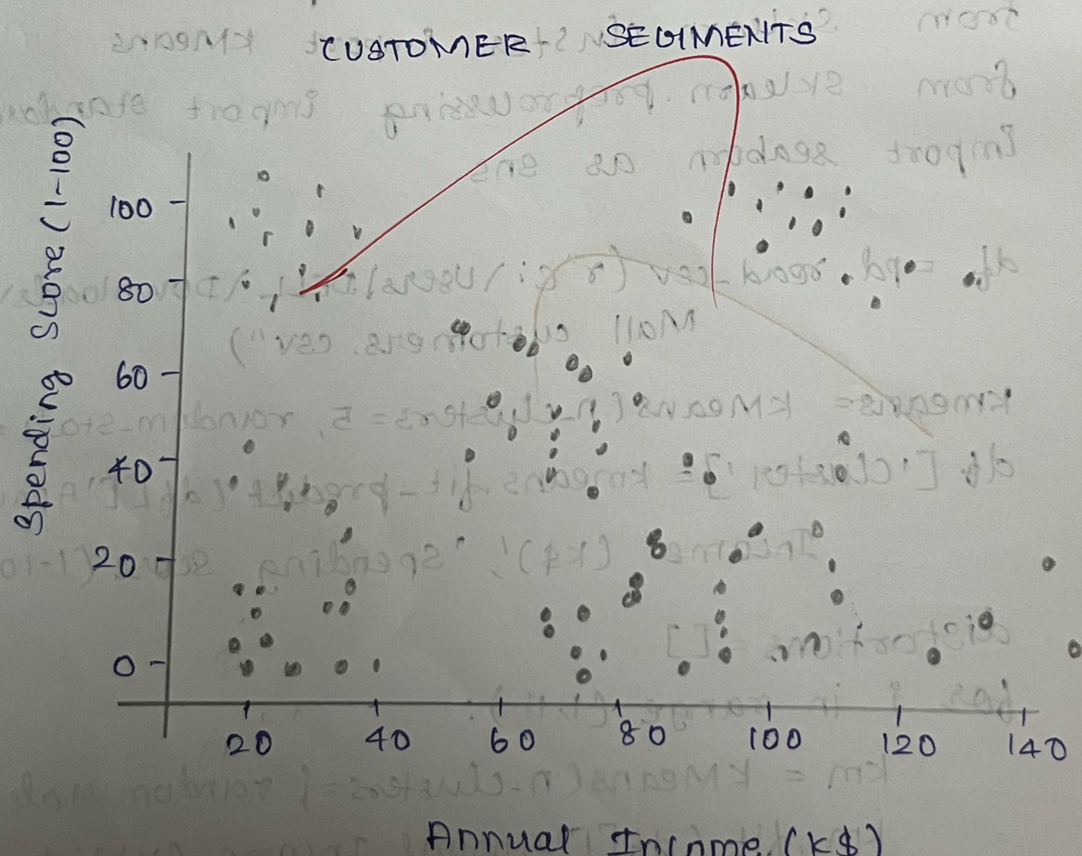
AIM:

To perform customer segmentation using KMeans clustering on the Mall customers dataset, and to implement and evaluate ensemble clustering (CSPA) on the wine dataset for improved clustering accuracy and visualization.

PROGRAM CODE:

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

df = pd.read_csv(r"C:\Users\DELL\Downloads\
                 Mall customers.csv")

kmeans= KMeans(n_clusters=5, random_state=42)
df['Cluster']= kmeans.fit_predict(df[['Annual
        Income (K$)', 'Spending Score (1-100)']])

distortions = []
for i in range(1,11):
    km = KMeans(n-clusters=i, random_state=42)
    km.fit(df[['Annual Income (K$)','
                  Spending Score (1-100)']])
    distortions.append(km.inertia_)

plt.plot(range(1,11), distortions, marker='o')
plt.title('Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('Inertia')
plt.show()
```


ELBOW METHOD
Inertia
250000
200000
150000
100000
50000
2
4
6
8
10
No. of. clusters


CUSTOMER SEGMENTS
Spending Score (1-100)
100
80
60
40
20
0
20
40
60
80
100
120
140
Annual Income (k$)

```
sns.scatterplot(data=df, x='Annual Income (k$)',
                y='Spending Score (1-100)',
                hue='Cluster',
                Palette='set2')
plt.title('Customer Segments')
plt.show()

from sklearn.datasets import load_wine
from sklearn.decomposition import PCA
from sklearn.metrics import silhouette_score
from skleaen.cluster import SpectralClustering
import numpy as np.

wine = load_wine()
x = pd.DataFrame(wine.data, columns=wine.feature_
                                      names)
x_scaled = StandardScaler().fit_transform(x)
base_clusterings = []
for k in [3,4,5]:
    km = KMeans(n_clusters=k, random_state=42)
    base_clusterings.append(km.fit_predict(
                                 x_scaled))

def cspa_ensemble(clusterings):
    n_samples = len(clusterings[0])
    Similarity_matrix = np.zeros((n_samples, n_
                                     samples))
    for clustering in clusterings;
        for i in range(n_samples):
            for j in range(n_samples):
                if clustering[i]==clustering[j]:
                    Similarity_matrix[i][j] +=1
    Similarity_matrix = Similarity_matrix / len(clust
                                              erings)
```

```
ensemble_labels = SpectralClustering(
        n_clusters=3,
        affinity='precomputed',
        random_state=42).fit_predict(similarity_
                                     matrix)
    return ensemble_labels

ensemble_labels = cspa_ensemble(base_clusterings)
print("Silhouette Score:", silhouette_score(X_scaled,
                          ensemble_labels))
pca = PCA(n_components=2)
X_pca = pca.fit_tranform(X_scaled)

plt.figure(figsize=(10,6))
plt.scatter(X_pca[:, 0], X_pca[:, 1],
            c=ensemble_labels, cmap='viridis',
            s=50, edgecolor='k')
plt.title("CSPA Ensemble Clustering on Wine
           Dataset (PCA-reduced)")
plt.xlabel("PCA Component 1")
plt.ylabel("PCA Component 2")
plt.colorbar(label='Cluster label')
plt.grid(True)
plt.show()
```

RESULT:

The kMeans algorithm successfully segmented mall customers into distinct groups based on income & spending scores, revealing clear customer clusters.

18/9