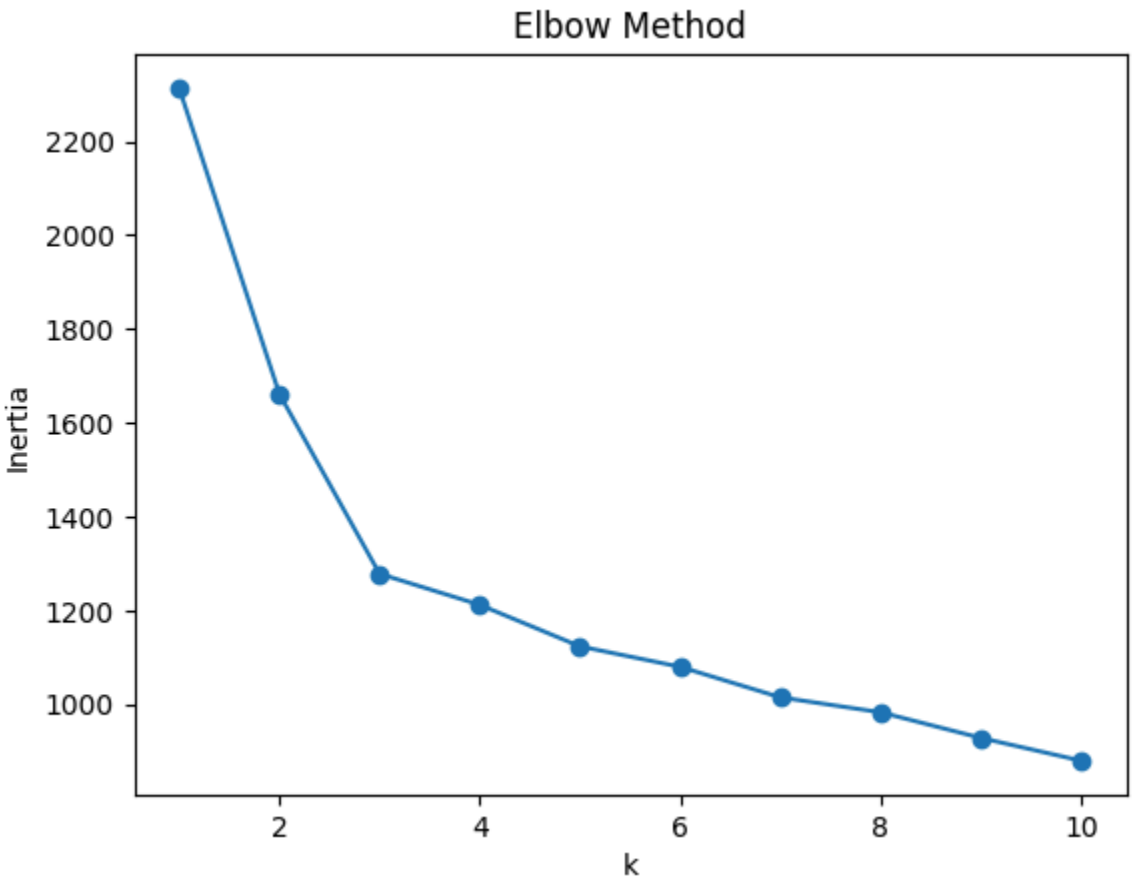
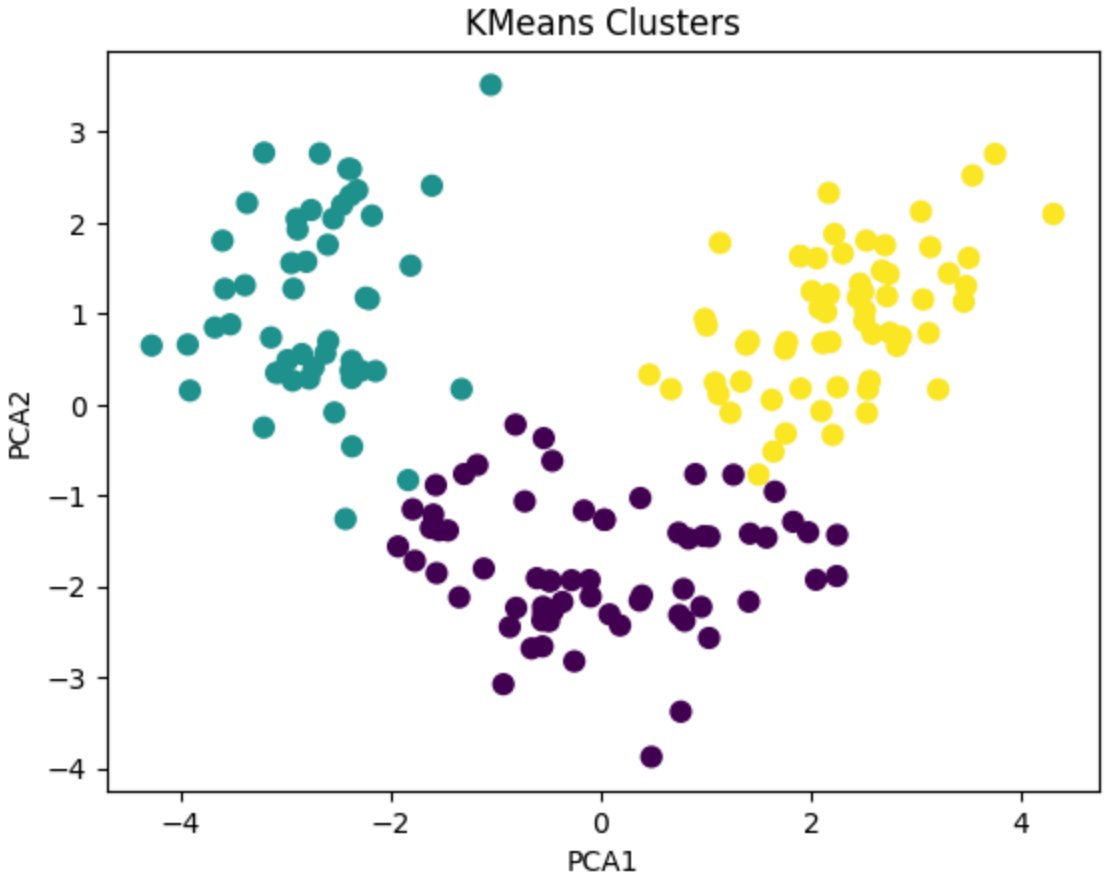
In [1]:

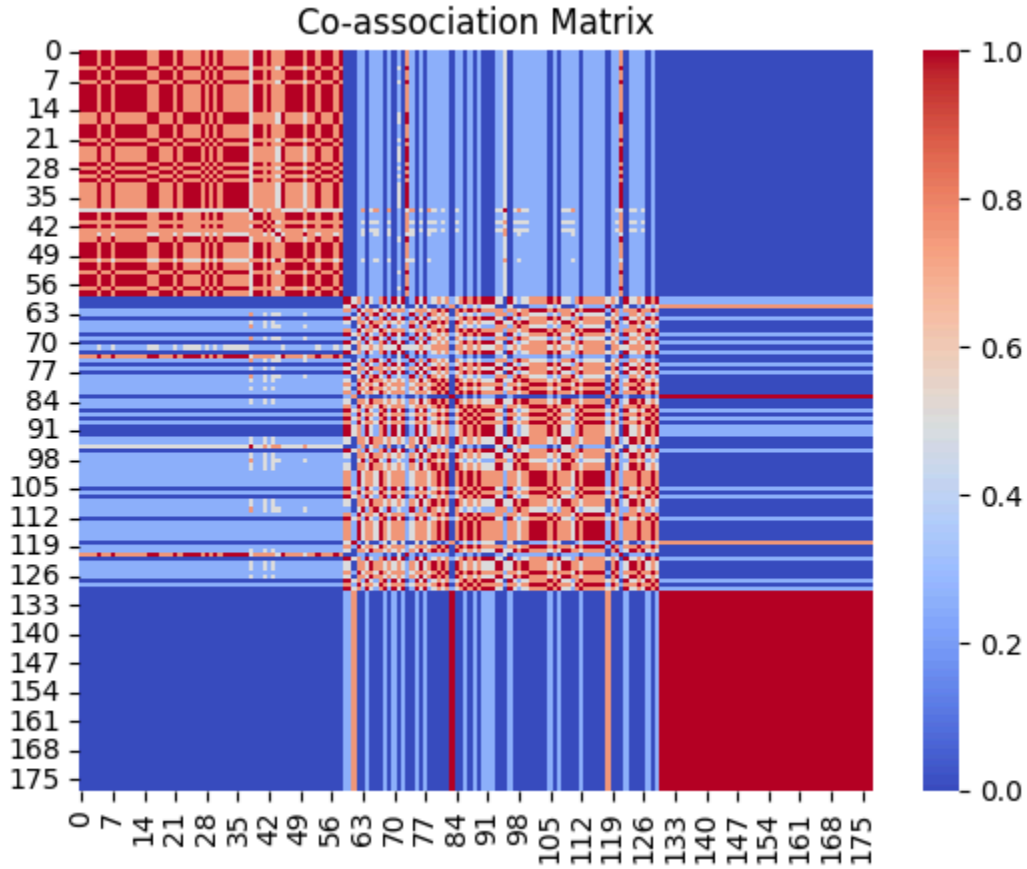
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans, SpectralClustering
from sklearn.metrics import silhouette_score

url = "https://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.dat
col_names = ["Class","Alcohol","Malic_acid","Ash","Alcalinity_of_ash","Magnesi
df = pd.read_csv(url, header=None, names=col_names)

X = df.drop("Class", axis=1).copy()
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

inertias = []
K_range = range(1, 11)
for k in K_range:
    km = KMeans(n_clusters=k, random_state=42)
    km.fit(X_scaled)
    inertias.append(km.inertia_)
plt.plot(K_range, inertias, 'o-')
plt.xlabel("k")
plt.ylabel("Inertia")
plt.title("Elbow Method")
plt.show()

k_opt = 3
kmeans = KMeans(n_clusters=k_opt, random_state=42)
labels_kmeans = kmeans.fit_predict(X_scaled)
sil_kmeans = silhouette_score(X_scaled, labels_kmeans)
print("KMeans silhouette:", sil_kmeans)

pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)
plt.scatter(X_pca[:,0], X_pca[:,1], c=labels_kmeans, cmap='viridis', s=50)
plt.title("KMeans Clusters")
plt.xlabel("PCA1")
plt.ylabel("PCA2")
plt.show()

base_clusterings = []
k_values = [2, 3, 4, 5]
for k in k_values:
    km2 = KMeans(n_clusters=k, random_state=42)
    labels2 = km2.fit_predict(X_scaled)
    base_clusterings.append(labels2)
base_clusterings = np.array(base_clusterings)

n_samples = X_scaled.shape[0]
co_assoc = np.zeros((n_samples, n_samples))
```

```
n_runs = base_clusterings.shape[0]
for run in range(n_runs):
    labels = base_clusterings[run]
    for i in range(n_samples):
        for j in range(n_samples):
            if labels[i] == labels[j]:
                co_assoc[i, j] += 1
co_assoc = co_assoc / n_runs

sns.heatmap(co_assoc, cmap="coolwarm")
plt.title("Co-association Matrix")
plt.show()

spectral = SpectralClustering(n_clusters=k_opt, affinity='precomputed', random
labels_ensemble = spectral.fit_predict(co_assoc)
sil_ens = silhouette_score(X_scaled, labels_ensemble)
print("Ensemble silhouette:", sil_ens)

plt.scatter(X_pca[:,0], X_pca[:,1], c=labels_ensemble, cmap='plasma', s=50)
plt.title("Ensemble Clusters")
plt.xlabel("PCA1")
plt.ylabel("PCA2")
plt.show()

print("Comparison:\nKMeans:", sil_kmeans, "\nEnsemble:", sil_ens)
```

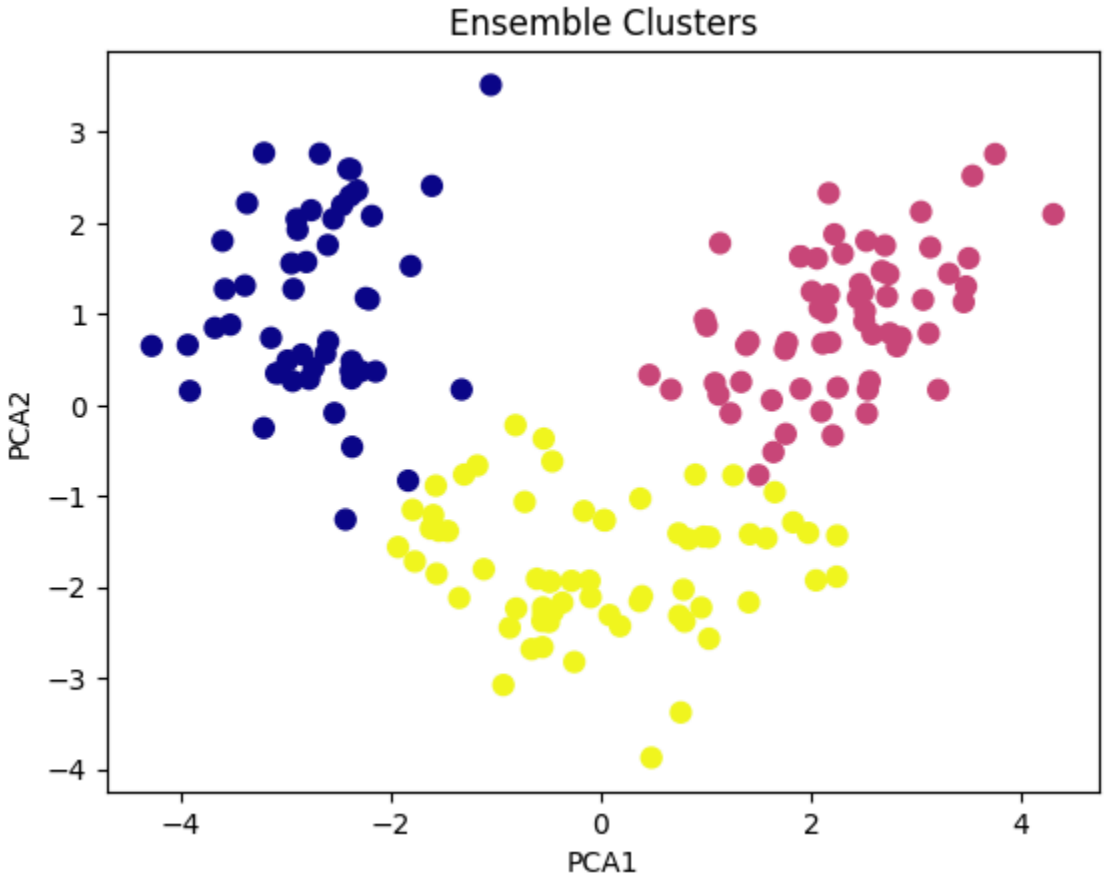



```
KMeans silhouette: 0.2848589191898987
```


KMeans Clusters
PCA2
PCA1
3
2
1
0
−1
−2
−3
−4
−4
−2
0
2
4


Co-association Matrix
0
7
14
21
28
35
42
49
56
63
70
77
84
91
98
105
112
119
126
133
140
147
154
161
168
175
1.0
0.8
0.6
0.4
0.2
0.0

```
Ensemble silhouette: 0.2848589191898987
```




```
Comparison:
KMeans: 0.2848589191898987
Ensemble: 0.2848589191898987
```