```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
from sklearn.metrics import silhouette_score
```

```
iris = load_iris()
X = iris.data
y_true = iris.target
feature_names = iris.feature_names
```

```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

```
silhouette_scores = []
K_range = range(2, 11)
for k in K_range:
    kmeans = KMeans(n_clusters=k, random_state=42, n_init=10)
    kmeans.fit(X_scaled)
    score = silhouette_score(X_scaled, kmeans.labels_)
    silhouette_scores.append(score)
    print(f"k = {k}, Silhouette Score = {score:.4f}")

best_k = K_range[np.argmax(silhouette_scores)]
print(f"\nBest number of clusters (k) = {best_k}")
```

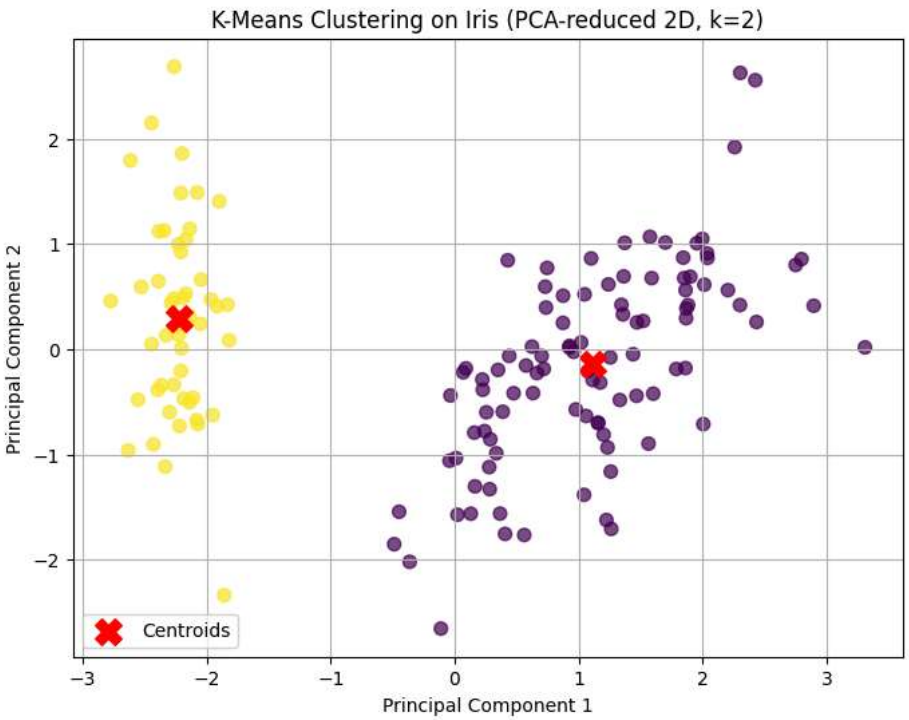
```
k = 2, Silhouette Score = 0.5818
k = 3, Silhouette Score = 0.4599
k = 4, Silhouette Score = 0.3869
k = 5, Silhouette Score = 0.3459
k = 6, Silhouette Score = 0.3171
k = 7, Silhouette Score = 0.3202
k = 8, Silhouette Score = 0.3387
k = 9, Silhouette Score = 0.3424
k = 10, Silhouette Score = 0.3518

Best number of clusters (k) = 2
```

```
kmeans_final = KMeans(n_clusters=best_k, random_state=42, n_init=10)
kmeans_final.fit(X_scaled)
labels = kmeans_final.labels_
centers = kmeans_final.cluster_centers_
```

```
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)
centers_pca = pca.transform(centers)
```

```
plt.figure(figsize=(8, 6))
scatter = plt.scatter(X_pca[:, 0], X_pca[:, 1], c=labels, cmap='viridis', s=50, alpha=0.7)
plt.scatter(centers_pca[:, 0], centers_pca[:, 1], c='red', marker='X', s=200, label='Centroids')
plt.title(f'K-Means Clustering on Iris (PCA-reduced 2D, k={best_k})')
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.legend()
plt.grid(True)
plt.show()
```


K-Means Clustering on Iris (PCA-reduced 2D, k=2)
Principal Component 2
Principal Component 1
Centroids
2
1
0
−1
−2
−3
−2
−1
0
1
2
3