Experiment 05

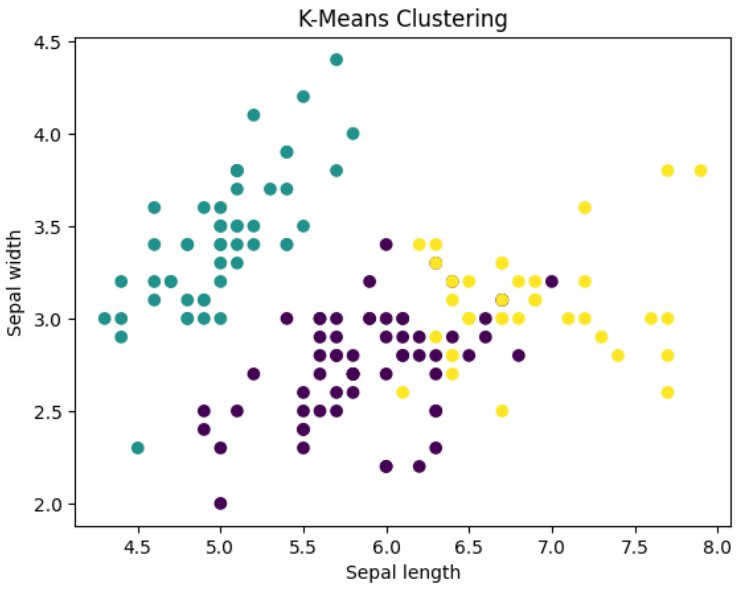
Aim: Implementation of Clustering Algorithm Using

1. k-means
2. Hierarchical (single/complete/average)

Code:

K means-clustering (iris dataset)

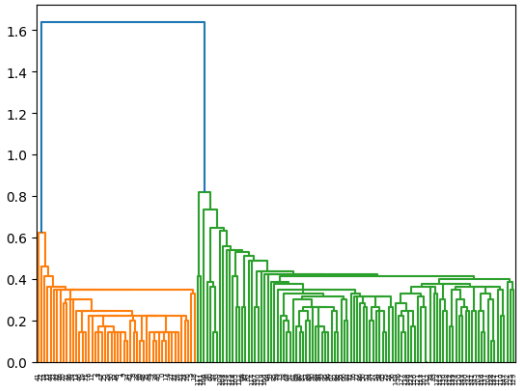
```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.datasets import load_iris

iris = load_iris()

n_clusters = 3
kmeans = KMeans(n_clusters=n_clusters, max_iter=100, random_state=0)
kmeans.fit(iris.data)
cluster_labels = kmeans.predict(iris.data)

import matplotlib.pyplot as plt

plt.scatter(iris.data[:, 0], iris.data[:, 1], c=cluster_labels)
plt.xlabel("Sepal length")
plt.ylabel("Sepal width")
plt.title("K-Means Clustering")
plt.show()
```

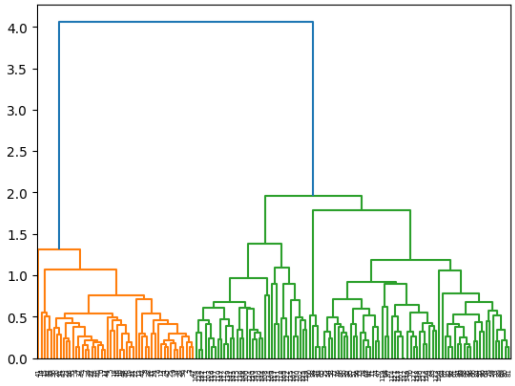


Subject: Data Mining and Warehousing **SAP ID: 60004220253 – Devansh Mehta**

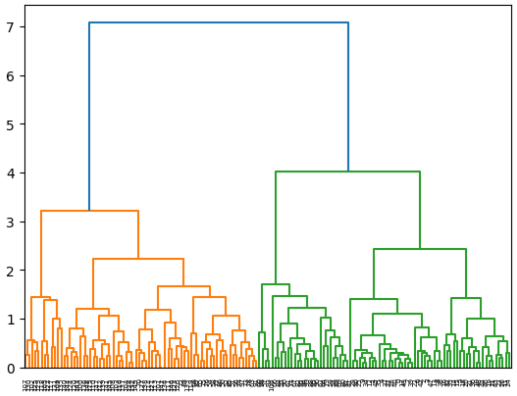
Hierarchical Clustering

```
from sklearn.cluster import AgglomerativeClustering
clustering = AgglomerativeClustering(linkage='ward', affinity='euclidean')
clustering.fit(iris.data)
cluster_labels = clustering.labels_
import scipy.cluster.hierarchy as sch

#single
dendrogram = sch.dendrogram(sch.linkage(iris.data, method='single'))
plt.show()

#average
dendrogram = sch.dendrogram(sch.linkage(iris.data, method='average'))
plt.show()

#complete
dendrogram = sch.dendrogram(sch.linkage(iris.data, method='complete'))
plt.show()
```



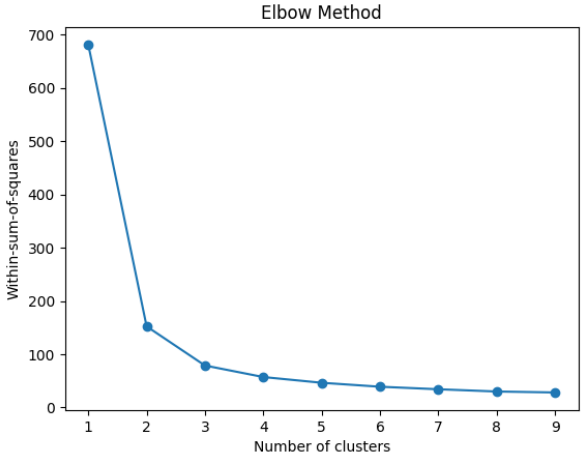


Subject: Data Mining and Warehousing **SAP ID: 60004220253 – Devansh Mehta**



Elbow optimization

```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.datasets import load_iris

iris = load_iris()
k_range = range(1, 10)
wss_scores = []
for k in k_range:
    kmeans = KMeans(n_clusters=k, random_state=0)
    kmeans.fit(iris.data)
    wss_scores.append(kmeans.inertia_)
import matplotlib.pyplot as plt
plt.plot(k_range, wss_scores, marker="o")
plt.xlabel("Number of clusters")
plt.ylabel("Within-sum-of-squares")
plt.title("Elbow Method")
plt.show()
```



Subject: Data Mining and Warehousing **SAP ID: 60004220253 – Devansh Mehta**

```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.datasets import load_iris

iris = load_iris()

n_clusters = 3
kmeans = KMeans(n_clusters=n_clusters, max_iter=100, random_state=0)
kmeans.fit(iris.data)
cluster_labels = kmeans.predict(iris.data)
import matplotlib.pyplot as plt

plt.scatter(iris.data[:, 0], iris.data[:, 1], c=cluster_labels)
plt.xlabel("Sepal length")
plt.ylabel("Sepal width")
plt.title("K-Means Clustering")
plt.show()
```