HYDERABAD INSTITUTE OF ARTS, SCIENCE, AND TECHNOLOGY

Artificial Intelligence Lab -5

Instructor: Ayesha Eman

Date: 24/10/2025

Lab Title:

Unsupervised Learning and Clustering using K-Means and Hierarchical Clustering

Objective:

The objective of this lab is to introduce unsupervised learning techniques and demonstrate how clustering algorithms group similar data points without predefined labels. Students will implement K-Means and Hierarchical Clustering using Python and analyze the resulting clusters through visualization and performance metrics.

1. Introduction to Unsupervised Learning

Unsupervised learning deals with unlabeled data, where the model identifies hidden patterns or groupings without predefined outcomes. Clustering is one of the most common unsupervised techniques used to discover natural groupings within data. Examples include customer segmentation, document grouping, and behavior analysis.

2. Clustering Algorithms Overview

2.1 K-Means Clustering

K-Means is a centroid-based algorithm that partitions the dataset into k clusters, minimizing the distance between points and their respective cluster centroids. It works by selecting initial centroids, assigning data points to the nearest centroid, and updating centroids until convergence. K-Means is efficient and suitable for large datasets but requires specifying the number of clusters in advance.

2.2 Hierarchical Clustering

Hierarchical clustering builds nested clusters by either agglomerating (bottom-up) or dividing (top-down) data points. It does not require specifying the number of clusters initially and produces a dendrogram, a tree-like diagram showing cluster relationships.

3. Implementation Using Python

```
# Import required libraries
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans, AgglomerativeClustering
from sklearn.metrics import silhouette_score
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.cluster.hierarchy import dendrogram, linkage

# Step 1: Load Dataset
data = pd.read_excel('students_dataset.xlsx')
data = data.drop(['Name'], axis=1, errors='ignore')
data = data.select_dtypes(include=[np.number]).dropna()

# Step 2: Feature Scaling
scaler = StandardScaler()
scaled_data = scaler.fit_transform(data)

# Step 3: K-Means Clustering
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans_labels = kmeans.fit_predict(scaled_data)

silhouette_kmeans = silhouette_score(scaled_data, kmeans_labels)
print(f"K-Means Silhouette Score: {silhouette_kmeans:.3f}")

plt.figure(figsize=(7,5))
sns.scatterplot(x=scaled_data[:,0], y=scaled_data[:,1], hue=kmeans_labels, palette='viridis')
plt.title("K-Means Clustering Results")
plt.xlabel("Feature 1")
plt.ylabel("Feature 2")
plt.show()

# Step 4: Hierarchical Clustering
hierarchical = AgglomerativeClustering(n_clusters=3)
hier_labels = hierarchical.fit_predict(scaled_data)
```

```
silhouette_hier = silhouette_score(scaled_data, hier_labels)
print(f"Hierarchical Clustering Silhouette Score: {silhouette_hier:.3f}")

plt.figure(figsize=(7,5))
sns.scatterplot(x=scaled_data[:,0], y=scaled_data[:,1], hue=hier_labels, palette='coolwarm')
plt.title("Hierarchical Clustering Results")
plt.xlabel("Feature 1")
plt.ylabel("Feature 2")
plt.show()

# Step 5: Dendrogram Visualization
plt.figure(figsize=(10,6))
linkage_matrix = linkage(scaled_data, method='ward')
dendrogram(linkage_matrix)
plt.title("Hierarchical Clustering Dendrogram")
plt.xlabel("Samples")
plt.ylabel("Distance")
plt.show()

# Step 6: Pair Plot Visualization
data['Cluster'] = kmeans_labels
sns.pairplot(data, hue='Cluster', palette='husl', diag_kind='kde')
plt.suptitle("Pair Plot of Clusters (K-Means)", y=1.02)
plt.show()
```

4. Evaluation Metrics

Metric	Description
Silhouette Score	Measures how well data points fit within their clusters. A higher score indicates better-defined clusters.
Inertia (K-Means)	Represents the sum of squared distances between data points and their assigned cluster centers. Lower values indicate tighter clusters.
Dendrogram	Provides a visual summary of hierarchical relationships among data points.

5. Visualization and Insights

The clustering visualizations demonstrate how data points are grouped based on similarity across features. K-Means forms compact, spherical clusters, while Hierarchical Clustering displays relationships through a dendrogram. The pair plot provides an intuitive view of how multiple features vary across clusters, showing clear separation patterns between high-performing, average, and low-performing groups.

6. Discussion Questions

1. What are the key differences between supervised and unsupervised learning?

2. How can you determine the optimal number of clusters in K-Means?

3. Why is feature scaling important before clustering?

4. Compare K-Means and Hierarchical Clustering in terms of performance and interpretability.

5. What insights can a dendrogram provide that K-Means cannot?

6. How does the pair plot help in interpreting clustering results?

7. Lab Tasks

1. Task 1: Use the same dataset to experiment with different numbers of clusters (k = 2, 3, 4, 5) and compare silhouette scores.
2. Task 2: Generate pair plots for each clustering method and analyze the separation of features in each cluster.