**Department of Electrical Engineering and   
Computer Science**

**Faculty Member:** Dr. Mohsin Kamal **Dated:** 14/12/2023

**Semester:** 7th **Section:** BEE 12C

**CS-477 Computer Vision**

Lab 13: Hierarchical Clustering

**Group Members**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **PLO4 - CLO4** | | **PLO5 -CLO5** | **PLO8 -CLO6** | **PLO9 -CLO7** |
| **Name** | **Reg. No** | **Viva / Quiz / Lab Performance** | **Analysis of Data in Lab Report** | **Modern Tool Usage** | **Ethics and Safety** | **Individual and Teamwork** |
|  |  | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** |
| Muhammad Ahmed Mohsin | 333060 |  |  |  |  |  |
| Muhammad Umer | 345834 |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**Table of Contents**

[2 Hierarchical Clustering 3](#_Toc154927877)

[2.1 Introduction 3](#_Toc154927878)

[2.2 Objectives 3](#_Toc154927879)

[2.3 Theory 3](#_Toc154927880)

[3 Lab Task 4](#_Toc154927881)

[4 Conclusion 7](#_Toc154927882)

# Hierarchical Clustering

## Introduction

Hierarchical clustering is a powerful unsupervised learning technique used to group similar data points into a nested hierarchy. This lab investigates various hierarchical clustering algorithms, their strengths and limitations, and their application in real-world scenarios.

## Objectives

The following are the main objectives of this lab:

* Understand the basics of hierarchical clustering, including its algorithm and steps.
* Apply clustering algorithms to a real-world dataset.
* Interpret the results of clustering analysis.

## Theory

Hierarchical clustering builds a hierarchy of clusters by iteratively merging or splitting clusters based on their similarity. Two main approaches exist:

* **Agglomerative:** Starts with individual data points as singletons and merges closest clusters in each step.
* **Divisive:** Starts with the entire dataset as a single cluster and recursively splits it into smaller, more homogeneous clusters.

# Lab Task

Download your own CSV dataset from the internet e.g. heatmap. Perform Hierarchical clustering of your dataset and showcase the plots.

### TASK CODE STARTS HERE ###

*# Importing the dataset*

df = pd.read\_csv(path\_data)

df.head()

le = preprocessing.LabelEncoder()

df["Gender"] = le.fit\_transform(df["Gender"])

df.head()

plt.figure(*figsize*=(10, 6))

plt.title("Heatmap of the dataset")

sns.heatmap(df)

plt.show()

plt.figure(*figsize*=(8, 6))

plt.title("Dendrograms")

dend = sch.dendrogram(sch.linkage(df, *method*="ward"))

*def* compute\_distance(*point1*, *point2*):

    return math.sqrt(sum([(p1 - p2) \*\* 2 for p1, p2 in zip(point1, point2)]))

*def* min\_link(*cluster1*, *cluster2*):

    return min([compute\_distance(v1, v2) for v1 in cluster1 for v2 in cluster2])

*def* max\_link(*cluster1*, *cluster2*):

    return max([compute\_distance(v1, v2) for v1 in cluster1 for v2 in cluster2])

*def* avg\_link(*cluster1*, *cluster2*):

    distances = [compute\_distance(v1, v2) for v1 in cluster1 for v2 in cluster2]

    return sum(distances) / len(distances)

*def* select\_distance\_metric(*metric*):

    metrics = {0: min\_link, 1: max\_link, 2: avg\_link}

    return metrics.get(metric, avg\_link)

*class* HierarchicalClustering:

*def* \_\_init\_\_(*self*, *dataset*, *num\_clusters*, *metric*):

*self*.dataset = dataset

*self*.num\_data\_points = len(dataset)

*self*.num\_clusters = num\_clusters

*self*.distance\_metric = select\_distance\_metric(metric)

*self*.clusters = *self*.initialize\_clusters()

*self*.cluster\_id\_counter = len(*self*.dataset)

*def* combine\_clusters(*self*, *cluster\_i\_id*, *cluster\_j\_id*):

        combined\_clusters = {

*self*.cluster\_id\_counter: *self*.clusters[cluster\_i\_id]

            + *self*.clusters[cluster\_j\_id]

        }

*self*.cluster\_id\_counter += 1

        for cluster\_id in *self*.clusters.keys():

            if (cluster\_id == cluster\_i\_id) or (cluster\_id == cluster\_j\_id):

                continue

            combined\_clusters[cluster\_id] = *self*.clusters[cluster\_id]

        return combined\_clusters

*def* display\_clusters(*self*):

        clusters\_str = ""

        for id, points in *self*.clusters.items():

            clusters\_str += *f*"Cluster: {id}\n"

            for point in points:

                clusters\_str += *f*"    {point}\n"

        return clusters\_str

*def* initialize\_clusters(*self*):

        return {

            data\_index: [data\_point]

            for data\_index, data\_point in enumerate(*self*.dataset)}

*def* identify\_nearest\_clusters(*self*):

        min\_distance = math.inf

        nearest\_clusters = None

        cluster\_ids = *list*(*self*.clusters.keys())

        for i, cluster\_i in enumerate(cluster\_ids[:-1]):

            for j, cluster\_j in enumerate(cluster\_ids[i + 1 :]):

                distance = *self*.distance\_metric(

*self*.clusters[cluster\_i], *self*.clusters[cluster\_j]

                )

                if distance < min\_distance:

                    min\_distance, nearest\_clusters = (distance,

(cluster\_i, cluster\_j))

        return nearest\_clusters

*def* execute\_clustering(*self*):

        while len(*self*.clusters.keys()) > *self*.num\_clusters:

            nearest\_clusters = *self*.identify\_nearest\_clusters()

*self*.clusters = *self*.combine\_clusters(\*nearest\_clusters)

N = len(df)

X = df.iloc[:, [3, 4]].values

hc = HierarchicalClustering(X, 5, 2)

hc.execute\_clustering()

y\_hc = hc.clusters

*# Visualising the clusters*

colors = ["red", "blue", "green", "purple", "orange"]

for i, cluster\_id in enumerate(hc.clusters.keys()):

    plt.scatter(X[np.array(hc.clusters[cluster\_id]), 0],

                X[np.array(hc.clusters[cluster\_id]), 1],

*c*=colors[i % len(colors)])

plt.title("Clusters of Customers (Hierarchical Clustering Model)")

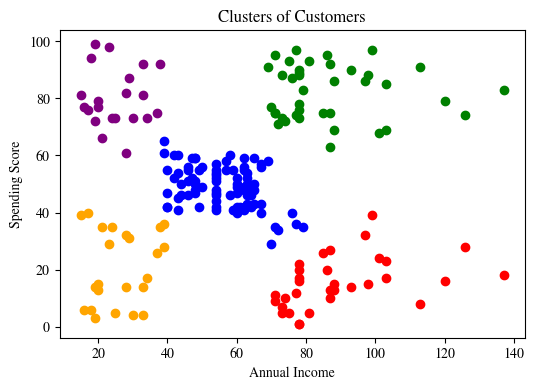
plt.xlabel("Annual Income")

plt.ylabel("Spending Score")

plt.show()

### TASK CODE ENDS HERE ###

### TASK OUTPUT SCREENSHOT STARTS HERE ###



### TASK OUTPUT SCREENSHOTS START HERE ###

# Conclusion

This lab explored the power of hierarchical clustering in uncovering hidden patterns and structures within data. We implemented and compared various algorithms, each with its own strengths and weaknesses. By analyzing the impact of distance metrics and linkage criteria, we gained a deeper understanding of how these choices influence cluster formation. Visualizing the results through dendrograms provided valuable insights into the hierarchical relationships between clusters, while silhouette coefficients helped assess their quality.