**University of Central Missouri**

**Department of Computer Science & Cybersecurity**

**CS5710 Machine Learning**

**Fall 2025**

# Home Assignment 4.

**Student name:**

**Manikanth Reddy Devarapalli**

**700765523**

**Submission Requirements:**

- Once finished your assignment push your source code to your repo (GitHub) and explain the work through the ReadMe file properly. Make sure you add your student info in the ReadMe file.
- Comment your code appropriately ***IMPORTANT.***
- Any submission after provided deadline is considered as a late submission.

**Part A: Calculation**

**Q1.** Find the cluster using the Average and MIN technique. Use Euclidean distance to build the complete distance matrix, updated the distance matrix to the final step and draw the dendrogram for each.

| | X | Y |
|---|---|---|
| P1 | 0.4 | 0.5 |
| P2 | 0.2 | 0.3 |
| P3 | 0.1 | 0.08 |
| P4 | 0.21 | 0.12 |
| P5 | 0.6 | 0.16 |
| P6 | 0.33 | 0.28 |
| P7 | 0.11 | 0.15 |

Answer:

Using Euclidean distance,

$$d(P_i, P_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

| | **P1** | **P2** | **P3** | **P4** | **P5** | **P6** | **P7** |
|---|---|---|---|---|---|---|---|
| **P1** | 0.000 | 0.283 | 0.516 | 0.425 | 0.394 | 0.231 | 0.455 |
| **P2** | 0.283 | 0.000 | 0.242 | 0.180 | 0.424 | 0.132 | 0.175 |
| **P3** | 0.516 | 0.242 | 0.000 | 0.117 | 0.506 | 0.305 | 0.071 |
| **P4** | 0.425 | 0.180 | 0.117 | 0.000 | 0.392 | 0.200 | 0.104 |
| **P5** | 0.394 | 0.424 | 0.506 | 0.392 | 0.000 | 0.295 | 0.490 |
| **P6** | 0.231 | 0.132 | 0.305 | 0.200 | 0.295 | 0.000 | 0.256 |
| **P7** | 0.455 | 0.175 | 0.071 | 0.104 | 0.490 | 0.256 | 0.000 |

Using MIN Technique

$D(A,B) = \min \{ d(x_i, x_j) \mid x_i \in A , y_j \in B\}$

**Steps:**

Step (1). Merge P3 and P7, smallest distance = 0.071

- Now cluster C1 = {P3, P7}

Step (2). Find minimum distance of C1 with other points:

- min(P3–P4, P7–P4) = 0.104
- so merge C1 + P4, distance = 0.104
- Now cluster C2 = {P3, P4, P7}

Step (3). Merge P2 and P6 → min distance = 0.132

- Now cluster C3 = {P2, P6}

Step (4). Minimum distance between clusters:

- C2 & C3 → min(P4–P2, P7–P2, P3–P2, P4–P6, P7–P6, P3–P6) = 0.175
- so merge C2 + C3, distance = 0.175
- Now cluster C4 = {P2, P3, P4, P6, P7}

Step (5). Nearest point to C4:

- P1 → min = 0.231
- so merge P1 + C4, distance = 0.231
- Now cluster C5 = {P1, P2, P3, P4, P6, P7}

Step (6). Remaining → C5 and P5

- min distance = 0.295

**(7). Final merge at 0.295**

| Step | Merge | Distance |
|---|---|---|
| 1 | P3 + P7 | 0.071 |
| 2 | (P3,P7) + P4 | 0.104 |
| 3 | P2 + P6 | 0.132 |
| 4 | (P3,P4,P7) + (P2,P6) | 0.175 |
| 5 | Add P1 | 0.231 |
| 6 | Add P5 | 0.295 |

**Step 3: Average Linkage (UPGMA) Technique**
**Rule:**

$$D(A,B) = \frac{1}{|A||B|} \sum_{x_i \in A, y_j \in B} (\ ,\ )$$

**Steps:**

Step (1). Merge P3 and P7, distance = 0.071

Cluster C1 = {P3, P7}

Step (2).Avg distance between C1 & P4:

$= (\ (\ 3,\ 4) + \ (\ 7,\ 4))/2 = (0.117 + 0.104)/2 = 0.1105$

Merge (P3,P7) + P4 → 0.111

Now C2 = {P3,P4,P7}

Step (3).Merge P2 + P6, distance = 0.132

Now C3 = {P2,P6}

Step (4).Avg distance between C2 and C3:

All 3×2 = 6 pairwise distances averaged:

$(0.2417 + 0.1749 + 0.1803 + 0.3048 + 0.2555 + 0.2000)/6 = 0.226$

Merge (P3,P4,P7) + (P2,P6) at 0.226

Now C4 = {P2,P3,P4,P6,P7}

Step (5).Avg distance P1 to C4:

$(0.516 + 0.425 + 0.455 + 0.283 + 0.231)/5 = 0.382$

Merge P1 + C4 → 0.382

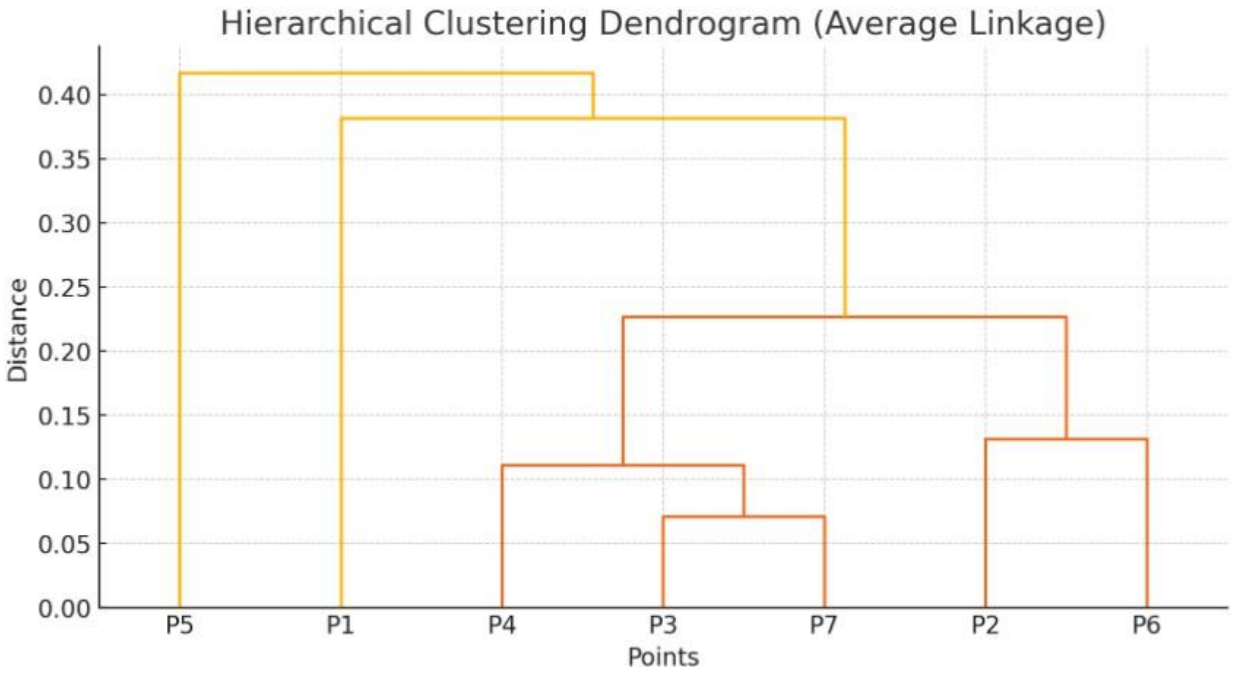
Step (6).Avg distance between P5 and the big cluster:

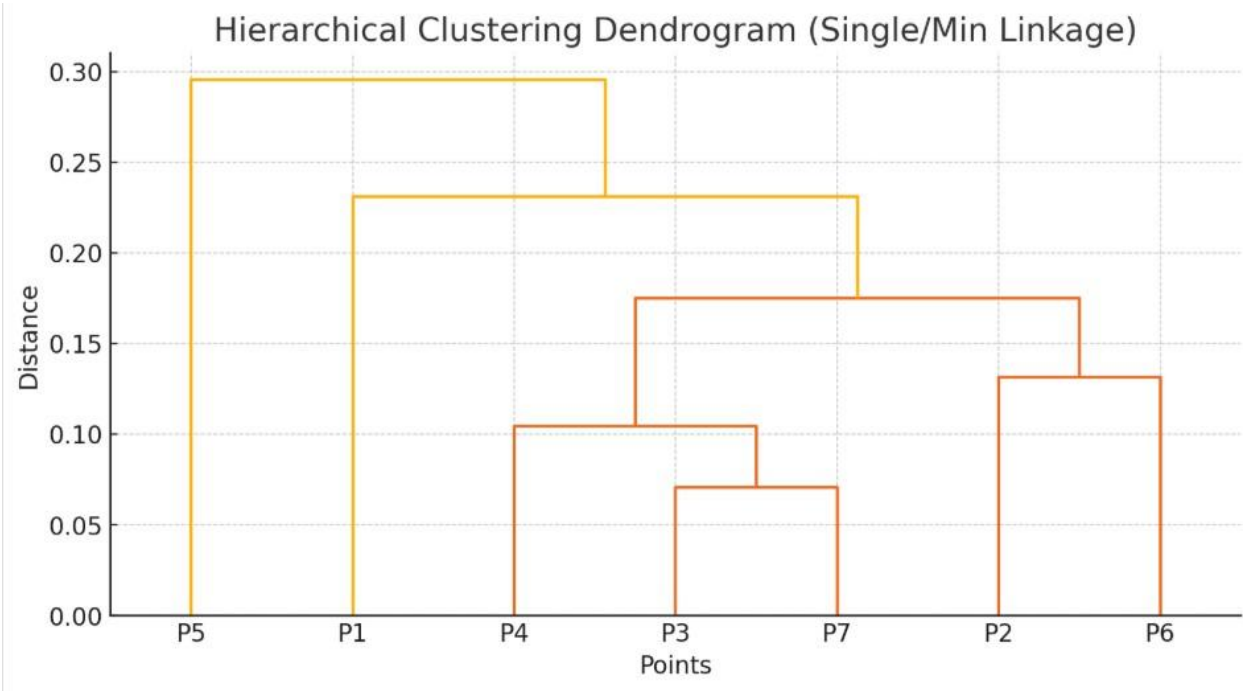
$(0.506 + 0.392 + 0.490 + 0.424 + 0.295 + 0.394)/6 = 0.417$

Merge P5 + cluster → 0.417

| Step | Merge | Distance |
|---|---|---|
| 1 | P3 + P7 | 0.071 |
| 2 | (P3,P7) + P4 | 0.111 |
| 3 | P2 + P6 | 0.132 |
| 4 | (P3,P4,P7) + (P2,P6) | 0.226 |
| 5 | Add P1 | 0.382 |
| 6 | Add P5 | 0.417 |

Final distance = 0.417




**Q2.** We have the following 2D data points:

Points: (2,1), (3,1), (3, 3), (4, 1), (5, 1), (6,7), (1,3), (2,5)
for K =3:

Centroid1: (2,1)
Centroid 2: (4, 1)
Centroid 3: (5, 1)

Euclidean Distance:

$$\text{Distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Answer :

Given points are

| Point | X | Y |
|---|---|---|
| P1 | 2 | 1 |
| P2 | 3 | 1 |
| P3 | 3 | 3 |
| P4 | 4 | 1 |
| P5 | 5 | 1 |
| P6 | 6 | 7 |
| P7 | 1 | 3 |
| P8 | 2 | 5 |

Given Centroid,

| Centroid | X | Y |
|---|---|---|
| C1 | 2 | 1 |
| C2 | 4 | 1 |
| C3 | 5 | 1 |

Using Euclidean distance,

$$d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

| Point | To C1 (2,1) | To C2 (4,1) | To C3 (5,1) | Assigned Cluster |
|---|---|---|---|---|
| P1 (2,1) | 0.00 | 2.00 | 3.00 | C1 |
| P2 (3,1) | 1.00 | 1.00 | 2.00 | C1 or C2 (tie → choose lower index → C1) |
| P3 (3,3) | 2.24 | 2.24 | 2.83 | C1 |
| P4 (4,1) | 2.00 | 0.00 | 1.00 | C2 |
| P5 (5,1) | 3.00 | 1.00 | 0.00 | C3 |
| P6 (6,7) | 7.21 | 6.32 | 6.08 | C3 |
| P7 (1,3) | 2.24 | 3.61 | 4.47 | C1 |
| P8 (2,5) | 4.00 | 4.47 | 5.00 | C1 |

Then Clusters points ,

| Cluster | Points |
|---|---|
| C1 (2,1) | P1, P2, P3, P7, P8 |
| C2 (4,1) | P4 |
| **C3 (5,1)** | P5, P6 |

Step 2: Compute New Centroids
Now,
For Cluster 1: (P1, P2, P3, P7, P8)

$$= \frac{2 + 3 + 3 + 1 + 2}{5} =$$

$$\frac{11}{5} = 2.2$$

$$= \frac{1 + 1 + 3 + 3 + 5}{5} =$$

$$13/5 = 2.6$$

New C1 = (2.2, 2.6)

For Cluster 2: (P4)
New C2 = (4, 1)

For Cluster 3: (P5, P6)

$$= \frac{5+6}{2} = 5.5, \qquad = (1+7)/2 = 4$$

New C3 = (5.5, 4)

Final Clusters,

| Cluster | X | Y |
|---|---|---|
| C1 | 2.2 | 2.6 |
| C2 | 4.0 | 1.0 |
| C3 | 5.5 | 4.0 |

**Part B — Short-Answer**

**Q1.**

a) Describe agglomerative hierarchical clustering.

It is a *bottom-up* approach where each data point starts as its own cluster, and pairs of clusters are merged step by step based on similarity until all points belong to a single cluster.

b) Describe divisive hierarchical clustering.

It is a *top-down* approach where all data points start in one large cluster, and splits are performed recursively until each point is in its own cluster.

c) Which one is more commonly used and why?

Agglomerative clustering is more common because it is computationally simpler, requires fewer assumptions, and is easier to implement than divisive clustering.

**Q2.**

a) To improve clustering quality, should inter-cluster distance be maximized or minimized?

It should be **maximized**, because clusters should be well separated from each other.

b) Same question for intra-cluster distance — explain the reasoning.

It should be **minimized**, because points within the same cluster should be close together, indicating high similarity.

**Q3.**

a) Define single link, complete link, and average link.

- **Single link:** Distance between two clusters = minimum distance between any pair of points (one from each cluster).
- **Complete link:** Distance between two clusters = maximum distance between any pair of points (one from each cluster).
- **Average link:** Distance between two clusters = average of all pairwise distances between points across clusters.

b) Explain one strength and one weakness of single-link clustering.

- **Strength (Single-link):** Can detect clusters of arbitrary shapes (good for non-spherical clusters).
- **Weakness:** Sensitive to noise and chaining effect — can link clusters through single outlier points.

**Q4.**

a) What is the role of tokenization and give one example.

It is the process of splitting text into smaller units (tokens) like words or subwords.
*Example:* "Data Science is fun" → ["Data", "Science", "is", "fun"].

b) Compare stemming vs. lemmatization in terms of speed and accuracy.

| Feature | Stemming | Lemmatization |
|---|---|---|
| **Definition** | Removes word suffixes to get the root form using simple rules. | Uses vocabulary and morphological analysis to find the base (lemma) form. |
| **Speed** | **Faster** — simple rule-based cutting of word endings. | **Slower** — requires dictionary lookup and linguistic processing. |
| **Accuracy** | **Lower** — may produce non-words (e.g., *studying* → *study* or *studies* → *studi*). | **Higher** — returns valid words with correct meaning (e.g., *studies* → *study*). |
| **Example** | "Better" → "bett" | "Better" → "good" |
| **Use Case** | When speed matters more than precision (e.g., search engines). | When linguistic accuracy and meaning are important (e.g., NLP analysis, chatbots). |

**Q5.**

a) Explain what word sense ambiguity is and provide an example.

Occurs when a word has multiple meanings.
*Example:* "Bank" can mean *river bank* or *financial institution*.

b) Explain why pronoun reference ambiguity can confuse a model.

Happens when a pronoun can refer to multiple nouns.
*Example:* "John told David that he won." → "He" could refer to John or David.
This confuses models because understanding depends on context

**Q6.**

a) Why can't NLP tasks like POS tagging be solved by predicting each token independently?

Because the part of speech of a word often depends on neighboring words (context).
For example, "book" can be a *noun* or *verb* depending on the sentence.

b) Give one example where decisions are mutually dependent in a sentence.

In "She will *book* a ticket," the word *will* helps determine that *book* is a verb, not a noun.
Thus, decisions about neighboring tokens influence each other