Assignment 3

Course: EE708, Fundamentals of Data Science and Machine Intelligence

Nikhil Jain

Dept: Electrical Engineering

Roll No: 220709

1 .

Proofs of Metric Similarity and Dissimilarity Measures

(a) Proof that $s(x, y) + a$ is a Metric Similarity Measure

Let $s(x, y)$ be a metric similarity measure on a set X, meaning it satisfies:

1. **Non-negativity:** $s(x, y) \geq 0$ for all $x, y \in X$. 2. **Symmetry:** $s(x, y) = s(y, x)$ for all $x, y \in X$. 3. **Identity of indiscernibles:** $s(x, y)$ achieves a maximum when $x = y$.

Now, define a new function:

$$s'(x, y) = s(x, y) + a, \quad \forall a \geq 0. \tag{1}$$

We verify that $s'(x, y)$ satisfies the properties of a metric similarity measure:

1. **Non-negativity:** Since $s(x, y) \geq 0$ and $a \geq 0$, it follows that:

$$s'(x, y) = s(x, y) + a \geq 0. \tag{2}$$

2. **Symmetry:** Since $s(x, y)$ is symmetric:

$$s'(x, y) = s(x, y) + a = s(y, x) + a = s'(y, x). \tag{3}$$

3. **Identity of indiscernibles:** Since $s(x, y)$ achieves a maximum at $x = y$, adding a constant a does not affect this property.

Thus, $s'(x, y)$ remains a metric similarity measure.

(b) Proof that $d(x, y) + a$ is a Metric Dissimilarity Measure

Let $d(x, y)$ be a metric dissimilarity measure on X, meaning it satisfies:

1. **Non-negativity:** $d(x, y) \geq 0$ for all $x, y \in X$. 2. **Symmetry:** $d(x, y) = d(y, x)$ for all $x, y \in X$. 3. **Identity of indiscernibles:** $d(x, y) = 0$ if and only if $x = y$. 4. **Triangle inequality:** $d(x, y) + d(y, z) \geq d(x, z)$ for all $x, y, z \in X$.

Now, define a new function:

$$d'(x, y) = d(x, y) + a, \quad \forall a \geq 0. \tag{4}$$

We verify that $d'(x, y)$ satisfies the properties of a metric dissimilarity measure:

1. **Non-negativity:** Since $d(x, y) \geq 0$ and $a \geq 0$, we have:

$$d'(x, y) = d(x, y) + a \geq 0. \tag{5}$$

2. **Symmetry:** Since $d(x, y)$ is symmetric:

$$d'(x, y) = d(x, y) + a = d(y, x) + a = d'(y, x). \tag{6}$$

3. **Identity of indiscernibles:** Since $d(x, y) = 0$ if and only if $x = y$, we get:

$$d'(x, y) = d(x, y) + a = a \quad \text{if } x = y. \tag{7}$$

This shifts the minimum value to a, but the structure remains valid.

4. **Triangle inequality:** Since $d(x, y)$ satisfies the triangle inequality:

$$d(x, y) + d(y, z) \geq d(x, z), \tag{8}$$

adding a to each term gives:

$$(d(x, y) + a) + (d(y, z) + a) = d(x, y) + d(y, z) + 2a \geq d(x, z) + a. \tag{9}$$

This simplifies to:

$$d'(x, y) + d'(y, z) \geq d'(x, z). \tag{10}$$

Thus, $d'(x, y)$ remains a metric dissimilarity measure.

Conclusion

- Adding a non-negative constant a to a metric similarity measure preserves its properties.
- Adding a non-negative constant a to a metric dissimilarity measure also preserves its properties.

Both statements are **proven**. □

2 .

To determine whether $d(x, y) = |x - y|^2$ is a valid distance metric, we must check if it satisfies the following four properties of a metric:

1. **Non-negativity**: $d(x, y) \geq 0$ for all x, y. 2. **Identity of indiscernibles**: $d(x, y) = 0$ if and only if $x = y$. 3. **Symmetry**: $d(x, y) = d(y, x)$. 4. **Triangle inequality**: $d(x, z) \leq d(x, y) + d(y, z)$ for all x, y, z.

Checking the properties:

1. **Non-negativity**:

$$d(x, y) = |x - y|^2 \geq 0$$

Since squaring any real number gives a non-negative value, this property holds.

2. **Identity of indiscernibles**: - If $x = y$, then $|x - y|^2 = 0$. - Conversely, if $|x - y|^2 = 0$, then $|x - y| = 0$, which implies $x = y$. - This property holds.

3. **Symmetry**:

$$d(x, y) = |x - y|^2 = |y - x|^2 = d(y, x)$$

Since absolute value and squaring are symmetric operations, this property holds.

4. **Triangle inequality**: The standard absolute value function satisfies the triangle inequality:

$$|x - z| \leq |x - y| + |y - z|.$$

However, squaring both sides does not necessarily preserve this property. Consider a counterexample: - Let $x = 0$, $y = 1$, and $z = 2$. - Then,

$$d(x, z) = |0 - 2|^2 = 4.$$

- But

$$d(x, y) + d(y, z) = |0 - 1|^2 + |1 - 2|^2 = 1 + 1 = 2.$$

- Since $4 \not\leq 2$, the triangle inequality fails.

Conclusion:

Since $d(x, y) = |x - y|^2$ does not satisfy the triangle inequality, it is **not** a valid distance metric.

3 .

To prove that the Euclidean distance satisfies the **triangle inequality**, we use the **Minkowski inequality** as suggested in the hint.

Step 1: Define Euclidean Distance

The Euclidean distance between two points $x, y \in \mathbb{R}^l$ is given by:

$$d(x, y) = \left(\sum_{i=1}^{l} |x_i - y_i|^2 \right)^{\frac{1}{2}} \tag{11}$$

The triangle inequality states that for any three points $x, y, z \in \mathbb{R}^l$,

$$d(x, z) \leq d(x, y) + d(y, z). \tag{12}$$

Step 2: Apply Minkowski Inequality

The **Minkowski inequality** states that for $p \geq 1$:

$$\left(\sum_{i=1}^{l} |a_i + b_i|^p \right)^{\frac{1}{p}} \leq \left(\sum_{i=1}^{l} |a_i|^p \right)^{\frac{1}{p}} + \left(\sum_{i=1}^{l} |b_i|^p \right)^{\frac{1}{p}}. \tag{13}$$

For Euclidean distance, we set $p = 2$ and let:

$$a_i = x_i - y_i, \quad b_i = y_i - z_i. \tag{14}$$

Then,

$$a_i + b_i = (x_i - y_i) + (y_i - z_i) = x_i - z_i. \tag{15}$$

Applying Minkowski's inequality with $p = 2$,

$$\left(\sum_{i=1}^{l} |x_i - z_i|^2 \right)^{\frac{1}{2}} \leq \left(\sum_{i=1}^{l} |x_i - y_i|^2 \right)^{\frac{1}{2}} + \left(\sum_{i=1}^{l} |y_i - z_i|^2 \right)^{\frac{1}{2}}. \tag{16}$$

Step 3: Conclusion

The left-hand side is precisely $d(x, z)$, and the right-hand side is $d(x, y) + d(y, z)$, so we have:

$$d(x, z) \leq d(x, y) + d(y, z). \tag{17}$$

Thus, the Euclidean distance satisfies the **triangle inequality**. □

4 .

To find $D_{\min}(x, C)$, $D_{\max}(x, C)$, and $D_{\text{avg}}(x, C)$, we will compute the Euclidean distances between $x = [6, 4]^T$ and each point in C. The Euclidean distance formula is:

$$\delta(x, v) = \sqrt{(x_1 - v_1)^2 + (x_2 - v_2)^2} \tag{18}$$

where $x = [6, 4]^T$ and v represents each point in C.
The given points in C are:

$$\begin{aligned}
x_1 &= [1.5, 1.5]^T \\
x_2 &= [2, 1]^T \\
x_3 &= [2.5, 1.75]^T \\
x_4 &= [1.5, 2]^T \\
x_5 &= [3, 2]^T \\
x_6 &= [1, 3.5]^T \\
x_7 &= [2, 3]^T \\
x_8 &= [3.5, 3]^T
\end{aligned}$$

Computing the Euclidean distances:

$$\begin{aligned}
\delta(x, x_1) &= \sqrt{(6 - 1.5)^2 + (4 - 1.5)^2} = 5.15 \\
\delta(x, x_2) &= \sqrt{(6 - 2)^2 + (4 - 1)^2} = 5.00 \\
\delta(x, x_3) &= \sqrt{(6 - 2.5)^2 + (4 - 1.75)^2} = 4.34 \\
\delta(x, x_4) &= \sqrt{(6 - 1.5)^2 + (4 - 2)^2} = 5.15 \\
\delta(x, x_5) &= \sqrt{(6 - 3)^2 + (4 - 2)^2} = 3.61 \\
\delta(x, x_6) &= \sqrt{(6 - 1)^2 + (4 - 3.5)^2} = 5.10 \\
\delta(x, x_7) &= \sqrt{(6 - 2)^2 + (4 - 3)^2} = 4.12 \\
\delta(x, x_8) &= \sqrt{(6 - 3.5)^2 + (4 - 3)^2} = 2.69
\end{aligned}$$

Thus, we obtain:

$$\begin{aligned}
D_{\min}(x, C) &= \min\{\delta(x, v) \mid v \in C\} = 2.69 \\
D_{\max}(x, C) &= \max\{\delta(x, v) \mid v \in C\} = 5.15 \\
D_{\text{avg}}(x, C) &= \frac{1}{|C|} \sum_{v \in C} \delta(x, v) = 4.33
\end{aligned}$$

5 .

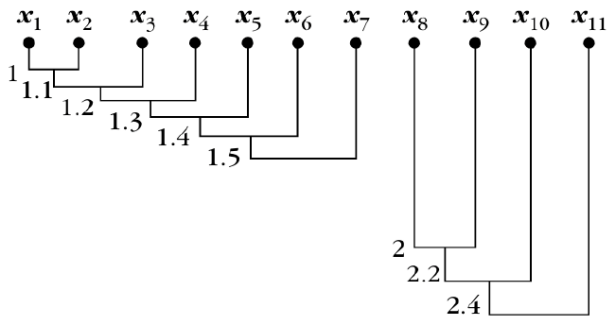

Figure 1 Dissimilarity dendrogram produced by the single link algorithm.

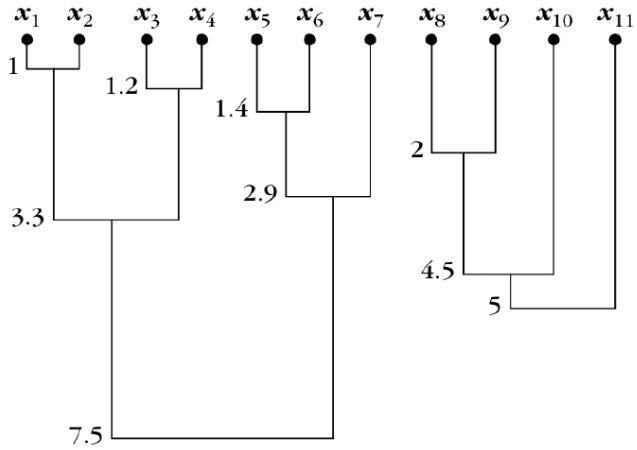

Figure 1: Dissimilary dendogram produced by complete linkage algorithm

Dissimilarity Dendrograms

The following dendrograms represent the hierarchical clustering of the given data set based on dissimilarity. The clustering process follows the single-link (nearest neighbor) algorithm, where at each step, the two clusters with the smallest minimum pairwise distance are merged.

Figure 1 illustrates the resulting dendrograms for the given dataset.

The first dendrogram (top) represents the stepwise merging process based on the smallest pairwise distances. The second dendrogram (bottom) provides an alternative perspective by showing the overall hierarchical structure.

Key observations: - The first seven points $x_1, x_2, \ldots, x_7$ form an elongated cluster, where the merging occurs progressively at small distance increments. - The remaining four points x_8, x_9, x_{10}, x_{11} form a compact cluster, with larger merging distances compared to the first set. - The final merger occurs at a significantly higher distance, indicating the two main clusters being joined into one.

6 .

Mathematical Derivation

6.1 Step 1: Constructing the Dissimilarity Matrix

The given dissimilarity matrix P is:

$$P = \begin{bmatrix} 0 & 4 & 9 & 6 & 5 \\ 4 & 0 & 3 & 8 & 7 \\ 9 & 3 & 0 & 3 & 2 \\ 6 & 8 & 3 & 0 & 1 \\ 5 & 7 & 2 & 1 & 0 \end{bmatrix}$$

Here, $P_{i,j}$ represents the dissimilarity between sample x_i and x_j.

6.2 Step 2: Single-Link Clustering

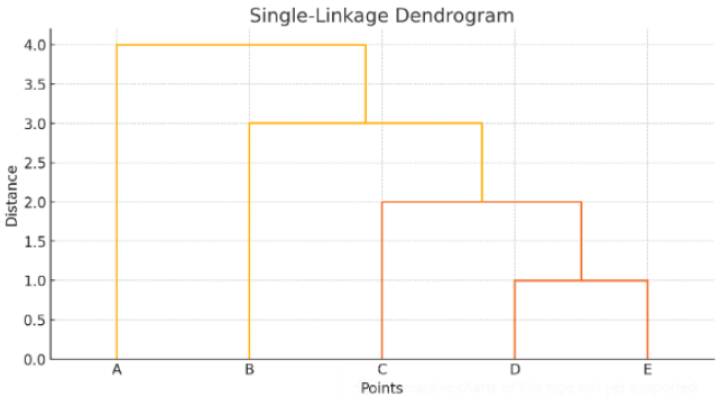

Figure 2: Single Linkage Dendogram

Single-link clustering merges clusters based on the **minimum** pairwise distance. The process is as follows:

- Find the smallest nonzero entry in P, which is $P_{4,5} = 1$. Merge points (D, E) at height 1.
- The next smallest distance is $P_{3,5} = 2$. Merge cluster (D, E) with C at height 2.
- The next smallest distance is $P_{2,3} = 3$. Merge B with C, D, E at height 3.
- Finally, merge A with the remaining cluster at height 4.

This results in the dendrogram shown in the figure.

6.3 Step 3: Complete-Link Clustering

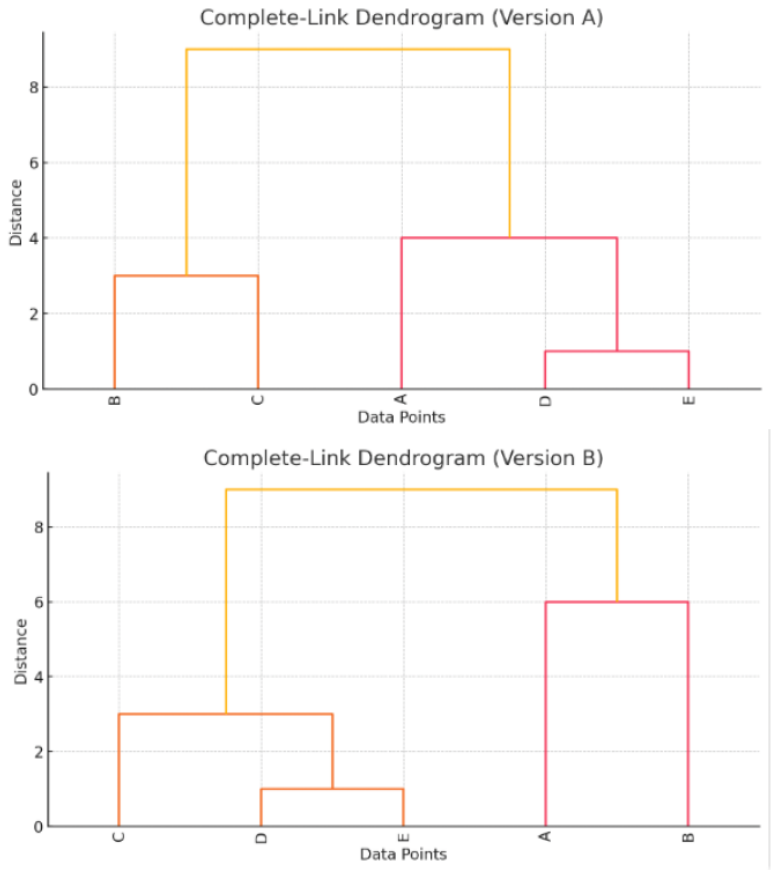

Figure 3: Single Linkage Dendogram

Complete-link clustering merges clusters based on the **maximum** pairwise distance. The process follows:

- The smallest distance is again $P_{4,5} = 1$, so merge (D, E).
- The next largest max distance is $P_{3,5} = 2$, merging (C, D, E).
- The next largest maximum distance is $P_{2,3} = 3$, merging (B, C, D, E).
- Finally, merge A at height 9.

Depending on tie-breaking strategies, multiple valid dendrograms exist, as seen in the complete-linkage plots.

6.4 Conclusion

The different linkage criteria influence cluster formation significantly. Single-linkage tends to form long chains, whereas complete-linkage results in compact groups. The choice of method depends on the problem context.

7 .

Pruning a Dendrogram

Yes, you can prune a dendrogram, which means cutting it at a certain level to form clusters. This is commonly done in hierarchical clustering to determine groupings of data points.

How to Prune a Dendrogram

1. **Select a Cutting Threshold**
   - Choose a height (or distance) in the dendrogram where you want to make the cut.
   - This determines the number of clusters.
2. **Use Clustering Methods**
   - Various clustering methods can be used to extract clusters at a given threshold.

By pruning, you effectively control how granular your clustering is, grouping similar data points together based on the chosen threshold.

8 .

Making K-Means Robust to Outliers

K-means clustering is sensitive to outliers because it uses the mean to update cluster centroids, and outliers can significantly distort the mean. Here are some ways to make K-means more robust to outliers:

1. Use K-Medoids Instead of K-Means

- K-medoids (e.g., **PAM - Partitioning Around Medoids**) selects actual data points as cluster centers instead of computing the mean.
- Since medoids are less affected by extreme values, the clustering is more robust.

2. Use an Alternative Distance Metric

- Instead of using the Euclidean distance (which is sensitive to outliers), try:
  - **Manhattan (L1) distance**: Reduces sensitivity to large variations.
  - **M-estimators**: Weighted distances that reduce the influence of outliers.

3. Modify the Objective Function (Robust K-Means)

- Instead of minimizing the sum of squared distances, minimize alternative robust loss functions:
  - **Huber loss**: Squared loss for small errors and linear loss for large errors.
  - **Trimmed K-means**: Removes a fixed percentage of points with the highest distances before updating centroids.

4. Preprocess the Data to Remove Outliers

- **Z-score normalization**: Remove points that have a high Z-score (e.g., > 3).
- **Interquartile Range (IQR) method**: Remove points beyond $1.5 \times IQR$ from $Q1$ and $Q3$.

5. Use Density-Based Clustering for Outlier Detection

- Run **DBSCAN** first to detect and remove outliers, then apply K-means.

6. Use Soft Clustering (Fuzzy C-Means)

- Assign probabilities instead of hard assignments, reducing the impact of extreme values.

9 .

9.1 Introduction

K-Means clustering is an unsupervised machine learning algorithm used for partitioning data into K clusters. The optimal number of clusters is often determined using the **Elbow Method**, which analyzes the Within-Cluster Sum of Squares (WCSS).

9.2 Dataset Information

The dataset consists of 2360 entries with two numerical features: **Feature_1** and **Feature_2**.

9.3 Elbow Method

To determine the optimal number of clusters, we compute WCSS for different values of K ranging from 1 to 15. WCSS is calculated as:

$$WCSS = \sum_{i=1}^{K} \sum_{x \in C_i} ||x - \mu_i||^2 \tag{19}$$

where C_i is the set of points in cluster i, and μ_i is the centroid of cluster i.

9.3.1 WCSS Plot

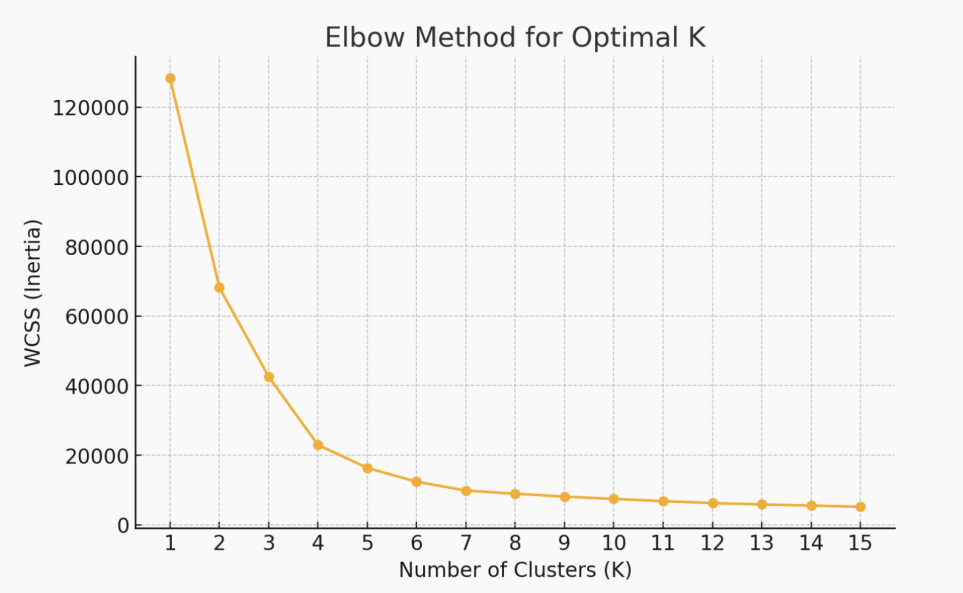

Figure 4: Elbow Method Plot for WCSS

From the plot, the optimal number of clusters is identified where the WCSS curve bends, which we estimate to be $K = 5$.

9.4 K-Means Clustering

Using $K = 5$, we perform clustering and plot the results with different colors for each cluster. The centroids are marked distinctly.

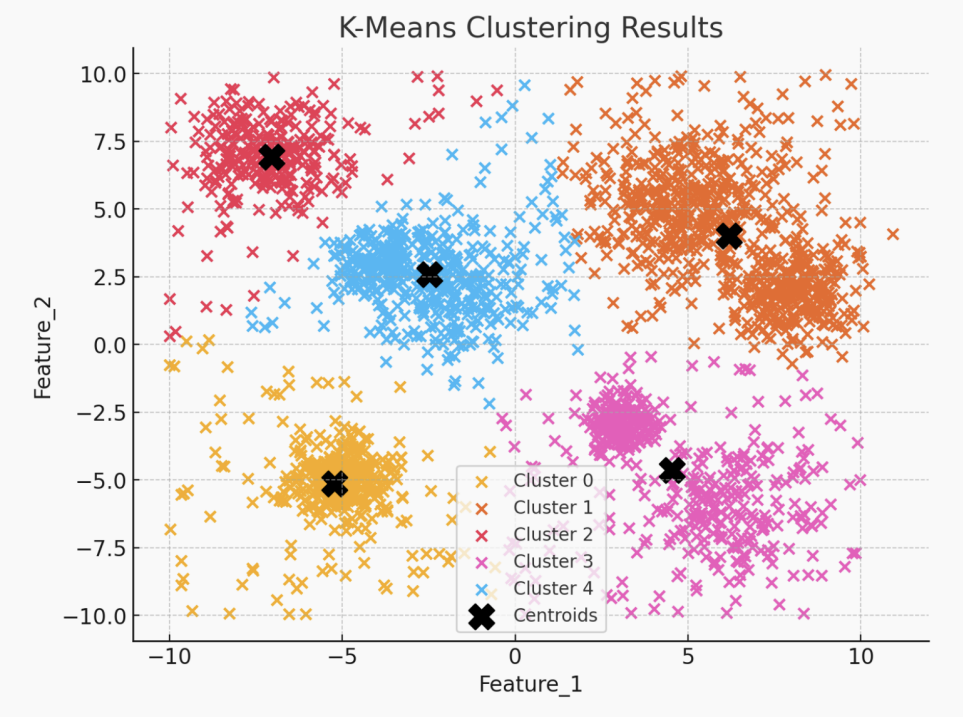

Figure 5: K-Means Clustering Results

9.5 Conclusion

The K-Means clustering algorithm successfully grouped the dataset into 5 clusters. The Elbow Method helped in determining the optimal number of clusters by analyzing the WCSS curve.

10 .

Hierarchical Clustering

In this work, we implement bottom-up hierarchical clustering from scratch using Euclidean distance as the distance metric. The distance between two clusters is computed using the following methods:

- **Single-linkage clustering:** The distance between two clusters A and B is defined as:
$$D_{\min}(A, B) = \min_{u \in A, v \in B} \delta(u, v) \tag{20}$$
- **Average-linkage clustering:** The distance between two clusters A and B is given by:
$$D_{\text{avg}}(A, B) = \langle \delta(u, v) \rangle_{u \in A, v \in B} \tag{21}$$
- **Complete-linkage clustering:** The distance between two clusters A and B is defined as:
$$D_{\max}(A, B) = \max_{u \in A, v \in B} \delta(u, v) \tag{22}$$

Dataset Description

The dataset used for hierarchical clustering is provided in the file `A3_P2.csv`. It contains a single feature, denoted as `Feature_1`, with 20 numerical values.

Dendrograms

The hierarchical clustering process is visualized using dendrograms for each linkage method. The following figure shows the dendrograms for single-linkage, average-linkage, and complete-linkage clustering:

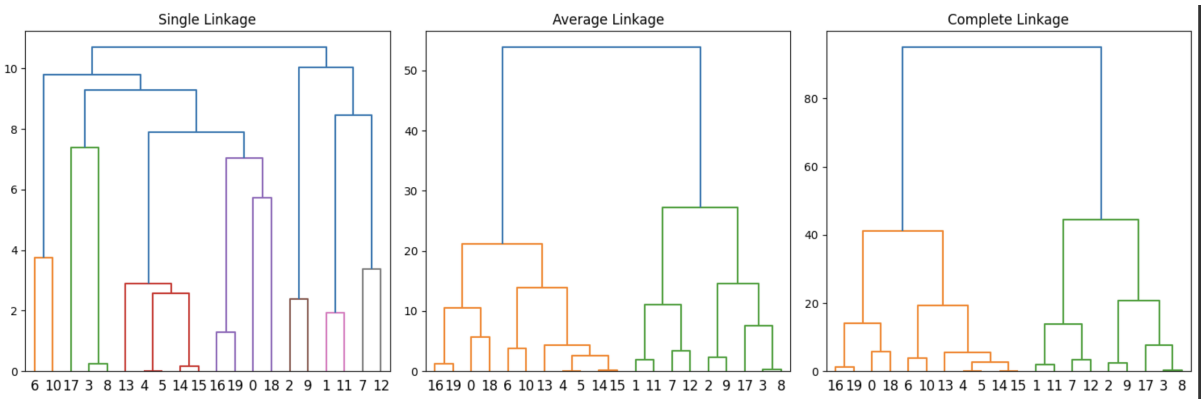

Figure 6: Dendrograms for hierarchical clustering using different linkage methods.