solution (1) To prove

$d(x, y) = (x-y)^2$ is valid matrix

should satisfy the cond^n

(I) Non-negative i.e $d(x, 0) \geq 0$

(II) $d(x, y) = 0$ if and only if $x = y$

(III) symmetry $\rightarrow$ $d(x, y) = d(y, x)$

(IV) Triangle inequality $\rightarrow$ $d(x, z) \leq d(x, y) + d(y, z)$

for $x = 0$, $y = 1$ & $z = 2$

$d(x, z) = 4$

$d(y, x) = 1$

$d(y, z) = 1$

Now we can say $d(x, z) \not\leq d(y, x) + d(y, z)$

so this does not satisfy the inequality

therefore $d(x, y) \rightarrow$ not valid matric

Solution (2) ∵ Euclidean distance b/w

$x = [x_1, x_2]$

$y = [y_1, y_2]$

and $d(x,y) = \sqrt{(x_1-y_1)^2 + (x_2-y_2)^2}$

$d(x, x_1) = 5.15$ $\quad$ $d(x, x_5) = 3.606$

$d(x, x_2) = 5$ $\quad$ $d(x, x_6) = 5.02$

$d(x, x_3) = 4.16$ $\quad$ $d(x, x_7) = 4.12$

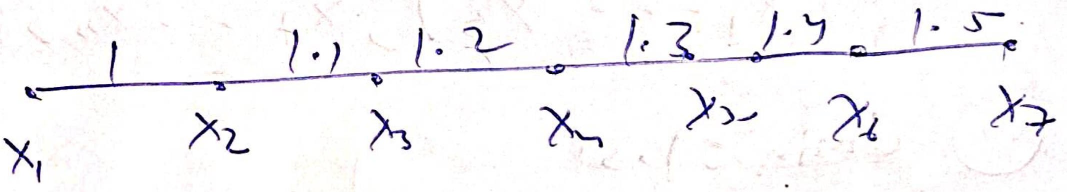
$d(x, x_4) = 4.42$ $\quad$ $d(x, x_8) = 2.69$

$D_{min}(x, C) = 2.69$

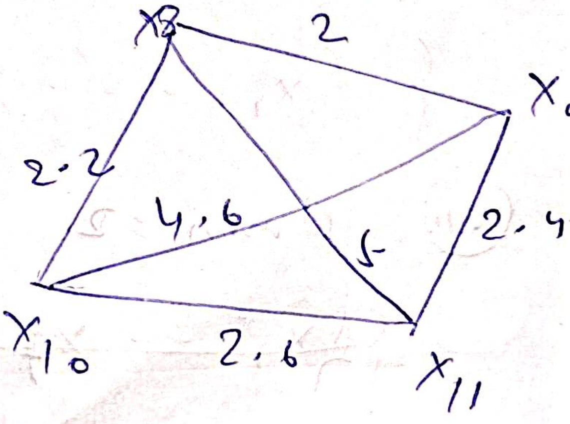
$D_{max}(x, C) = 5.15$

$D_{avg}(x, C) = \frac{1}{8} \sum_{i=1}^{8} d(x, x_i) = \underline{4.33}$

Solution (3)



For single linkage, we make dendrogram on min discrete cluster point



Initially $= \{x_1, x_2 \ldots x_{11}\}$

(i) $d_{min} = 1$ (x_1, x_2)

$\{(x_1, x_2), x_3, \dots, x_{11}\}$

(ii) $d_{min} = 1.1$, $(\{x_1, x_2\}, x_3)$

$\{(x_1, x_2, x_3), x_4, x_5 \dots x_{11}\}$

(iii) $d_{min} = 1.2$

$\{(x_1, x_2, x_3, x_4), x_5 \dots x_{11}\}$

(iv) $d_{min} = 1.3$

$\{(x_1 \dots x_5), x_6 \dots x_{11}\}$

(v) $d_{min} = 1.4$

$\{(x_1 \dots x_6), x_7 \dots x_{11}\}$

(vi) $d_{min} = 1.5$

$\{(x_1, \dots x_7\}, x_8 \dots x_{11}\}$

(vii) $d_{min} = 2$

$\cancel{\{\{x_1 \dots x_7\}}$

$\{(x_1, \dots x_7), (x_8, x_9), x_{10}, x_{11}\}$

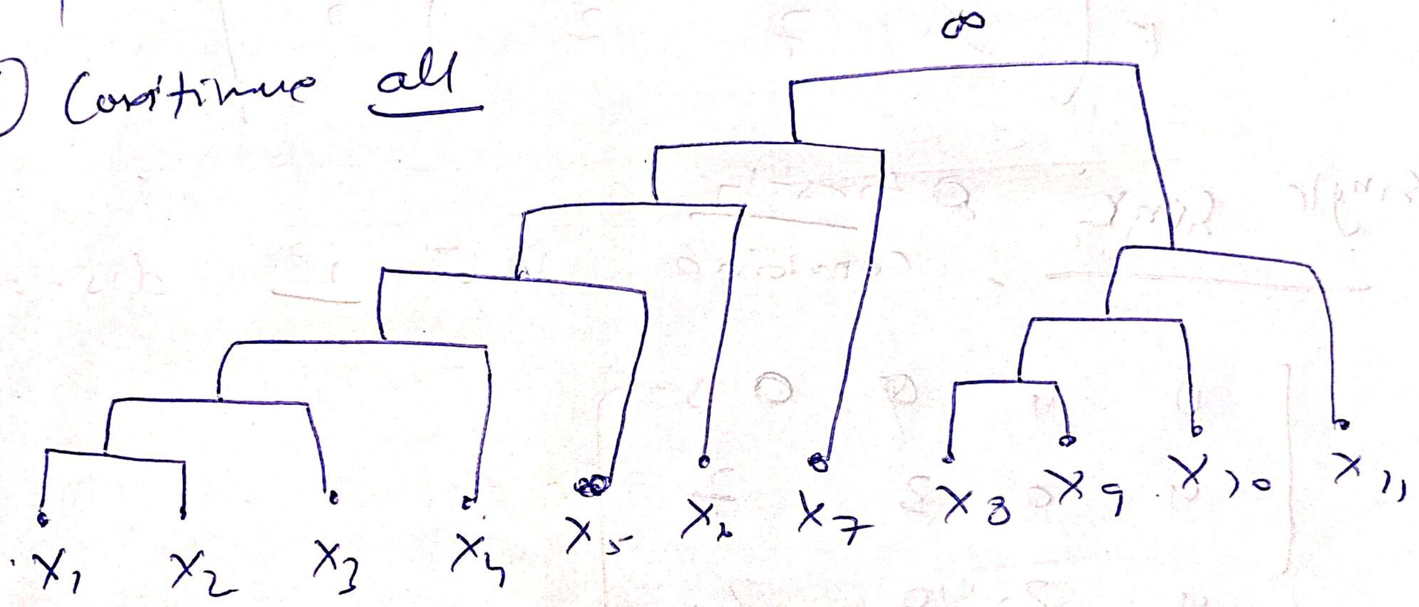
(viii) $d_{min} = 2.2$

$\{\{x_1 --- x_7\}, \{x_8, x_9, x_{10}\}, x_{11}\}$

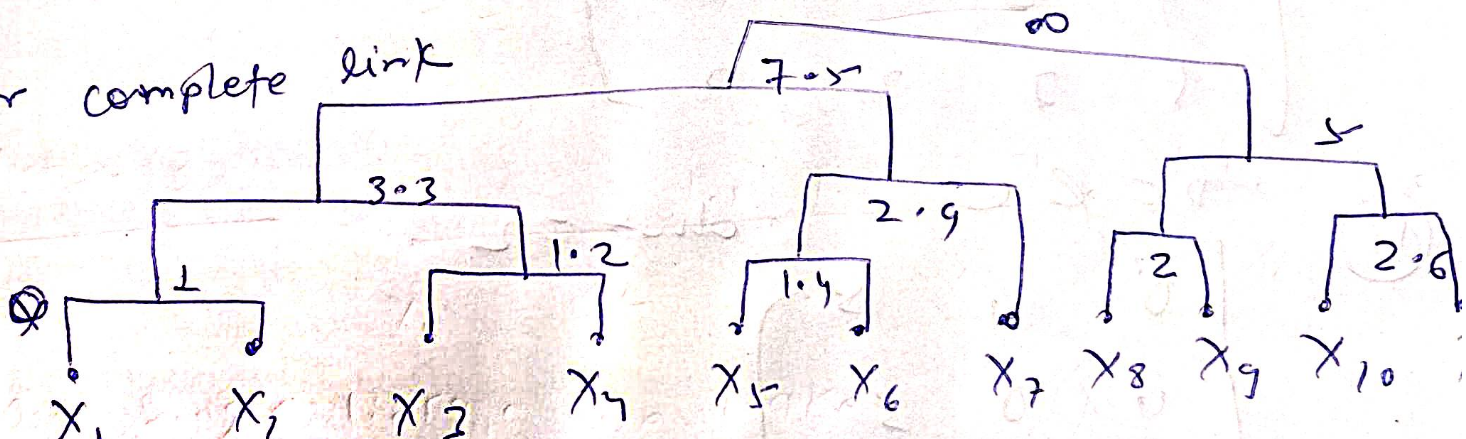
(IX) $d_{min} = 2.4$

$(\{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}, \{x_8, x_9, x_{10}, x_{11}\})$

(X) Continue all



for complete link



Solution (4)

$$P = \begin{array}{c|ccccc} & A & B & C & D & E \\ \hline A & 0 & 4 & 9 & 6 & 5 \\ B & 4 & 0 & 3 & 8 & 7 \\ C & 9 & 3 & 0 & 3 & 2 \\ D & 6 & 8 & 3 & 0 & 1 \\ E & 5 & 7 & 2 & 1 & 0 \end{array}$$

Single link, firstly combine DE ie dist = 1

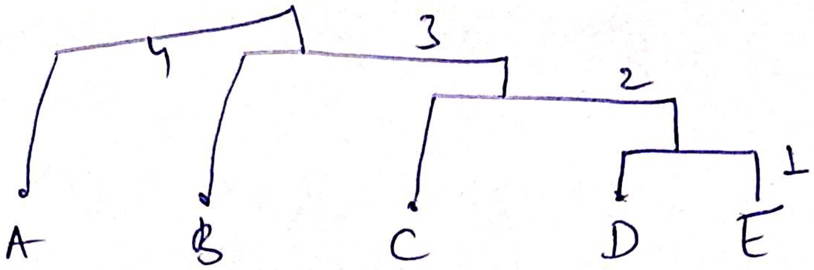
$$\begin{bmatrix} 0 & 4 & 9 & 5 \\ 4 & 0 & 3 & 7 \\ 9 & 3 & 0 & 2 \\ 5 & 7 & 2 & 0 \end{bmatrix}$$

(ii) DE & C dist = 2

$$\begin{bmatrix} 0 & 4 & 5 \\ 4 & 0 & 3 \\ 5 & 3 & 0 \end{bmatrix}$$

(iii) B & CDE dist = 3

$$\begin{bmatrix} 0 & 4 \\ 4 & 0 \end{bmatrix}$$



Complete link.

Step (i) Combine DE

$$\begin{array}{c} \begin{array}{cccc} A & B & C & (DE) \end{array} \\ \begin{bmatrix} 0 & 4 & 9 & 6 \\ 4 & 0 & 3 & 8 \\ 9 & 3 & 0 & 3 \\ 6 & 8 & 3 & 0 \end{bmatrix} \end{array}$$

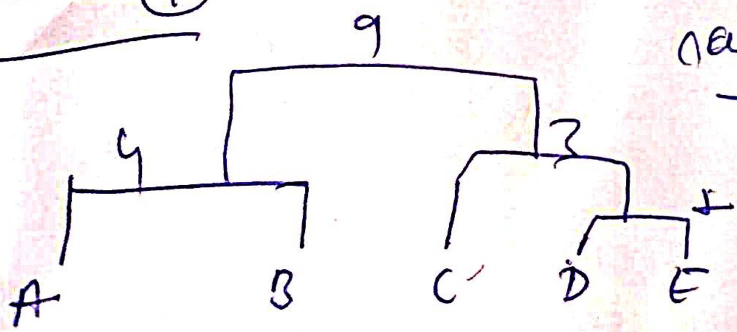
(ii) Combine (C, DE), & B, C

$$\begin{array}{c|c} \begin{array}{c} A \\ B \\ CDE \end{array} & \begin{bmatrix} 0 & 4 & 9 \\ 4 & 0 & 8 \\ 9 & 8 & 0 \end{bmatrix} \end{array} \qquad \begin{array}{c|c} \begin{array}{c} A \\ BC \\ DE \end{array} & \begin{bmatrix} 0 & 9 & 6 \\ 9 & 0 & 8 \\ 6 & 8 & 0 \end{bmatrix} \end{array}$$

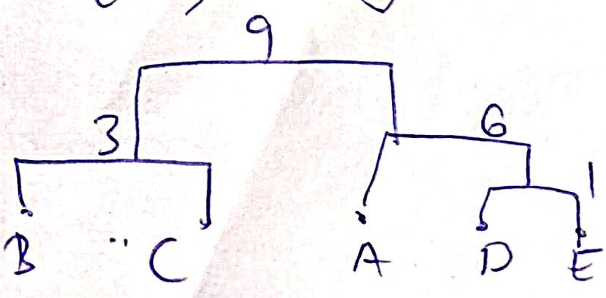
↓ Combine AB ↓ Combine (A, DE)

$$\begin{bmatrix} 0 & 9 \\ 9 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 9 \\ 9 & 0 \end{bmatrix}$$

Case (a)



Case (b)



Solution (5) 2-D feature space with N = 10 points

k = 3 cluster

→ distortion funcn = squared euclidean distance b/w original points and their assigned cluster centroids.

Initial cluster centroids

$C_1 = (2,3)$. $C_2 = (5,8)$ $C_3 = (9,4)$

Cluster	(1,2)	(3,4)	(6,7)	(8,3)	(5,5)
C_1	1.41	1.41	5.66	6	3.6
C_2	7.21	4.47	1.41	5.83	3
C_3	8.25	6	4.24	1.41	4.12

Cluster $C_1 = (1,2); (3,4)$

$C_2 = (6,7), (5,5)$

$C_3 = (8,3)$

New cluster ⇒ $\left(\frac{1+3}{2}, \frac{2+4}{2}\right) \equiv (2,3)$

$\left(\frac{6+5}{2}, \frac{7+5}{2}\right) \equiv (5.5, 6)$

$(8,3)$

Initial distortion →

distortion = squared Euclidean distance b/w original points & assigned cluster points

$$\Rightarrow \left((2-1)^2+(3-2)^2\right)+\left((2-3)^2+(3-4)^2\right)+$$

$$\left((5-6)^2+(8-7)^2\right)+\left((7-8)^2+(4-3)^2\right)+\left((5-5)^2+(8-5)^2\right)$$

$$= \underline{\underline{17}}$$

Distortion after one distortion

$$=\left((2-1)^2+(3-2)^2\right)+\left((2-3)^2+(3-4)^2\right)+$$

$$\left((5.5-6)^2+(6-7)^2\right)+\left((5.5-5)^2+(6-5)^2\right)$$

$$+\left((8-8)^2+(3-3)^2\right)$$

$$= \underline{\underline{6.5}}$$

distortion decreases $\Rightarrow$ $\boxed{6.5 < 17}$

problem (6)

$$E_Z[\ln p(X, Z \mid \mu, \Sigma, \pi)] = \sum_{n=1}^{N}\sum_{k=1}^{K} \gamma(z_{nk}) \left(\ln \pi_k + \ln N(x_n \mid \mu_k, \Sigma_k)\right)$$

So Gaussian density

$$\ln N(x_n \mid \mu_k, \Sigma_k) = \ln\left(\frac{1}{\sqrt{2\pi}\,|\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2}(x_n-\mu_k)^T \Sigma^{-1}(x_n-\mu_k)\right)\right)$$

$$= -\frac{D}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma_k| - \frac{1}{2}(x_n-\mu_k)^T \Sigma^{-1}(x_n-\mu_k)$$

So for maximizing the equation w.r.to Σ_k

$$-\frac{1}{2}\sum_{n=1}^{N} \gamma(z_{nk})\left(\Sigma_k^{-1} - \Sigma_k^{-1}(x_n-\mu_k)(x_n-\mu_k)^T \Sigma_k^{-1}\right) = 0$$

Now $\because$ $N_k = \sum_{n=1}^{N} \gamma(z_{nk})$

$$N_k - \sum_{n=1}^{N} \gamma(z_{nk})(x_n-\mu_k)(x_n-\mu_k)^T \Sigma_k^{-1} = 0$$

$$\boxed{\Sigma_k = \frac{1}{N_k}\sum_{n=1}^{N} \gamma(z_{nk})(x_n-\mu_k)(x_n-\mu_k)^T}$$

max. w.r to π_k $\quad \sum_{n=1}^{N}\sum_{k=1}^{K} \gamma(z_{nk}) \ln \pi_k$

$\because \sum_{k=1}^{K} \pi_k = 1$, use langrangian.

$$L = \sum_{n=1}^{N}\sum_{k=1}^{K} \gamma(z_{nk}) \ln \pi_k + \lambda\left(1 - \sum_{k=1}^{K} \pi_k\right)$$

diff w.r to π_k and equal to zero

$$\frac{\partial L}{\partial \pi_k} = \sum_{n=1}^{N} \frac{\gamma(z_{nk})}{\pi_k} - \lambda = 0 \Rightarrow$$

$$\lambda = \frac{1}{\pi_k}\left(\sum_{n=1}^{N} \gamma(z_{nk})\right)$$

$$\boxed{\lambda = \frac{N_k}{\pi_k}} \Rightarrow \boxed{\pi_k = \frac{N_k}{\lambda}}$$

use contraints $\rightarrow$ $\sum_{k=1}^{N} \pi_k = 1 = \sum_{k=1}^{K} \frac{N_k}{\lambda} = 1$ $\boxed{\lambda = N}$

so $\boxed{\pi_k = \frac{N_k}{N}}$

Solution (7) Density model

$$p(x) = \sum_{k=1}^{K} \pi_k \, P(x \mid k) \qquad , \qquad x = (x_a, x_b)$$

∴ Conditional probability.

$$p(x_a / x_b) = \frac{p(x_a, x_b)}{p(x_b)}$$

we know →

$$p(x_a, x_b) = P(x) = \sum_{k=1}^{K} \pi_k \cdot p(x \mid k) = \sum_{k=1}^{K} \pi_k \cdot p(x_a, x_b \mid k) \quad \text{—— (1)}$$

⇒ $P(x_b)$ as the marginal over the same matrix

$$p(x_b) = \sum_{k=1}^{K} \pi_k \cdot \int p(x_a, x_b \mid k) \, dx_a = \sum_{k=1}^{K} \pi_k \cdot P(x_b \mid k) \quad \text{—— (2)}$$

(1) & (2)

$$p(x_a \mid x_b, k) = \frac{P(x_a, x_b \mid k)}{P(x_b \mid k)}$$

$$P(x_a \mid x_b) = \frac{\sum_{k=1}^{K} \pi_k \cdot P(x_b \mid k) \cdot P(x_a \mid x_b, k)}{\sum_{j=1}^{K} \pi_j \cdot P(x_b \mid j)}$$

$$p(x_a \mid x_b) = \sum_{k=1}^{K} \frac{\pi_k \, P(x_b \mid k)}{\sum_j \pi_j \, P(x_b \mid j)} \cdot P(x_a \mid x_b, k)$$

Similarly

$$\boxed{P(x_b \mid x_a) = \sum_{k=1}^{K} \frac{\pi_k \, P(x_a \mid k)}{\sum \pi_j \, P(x_a \mid j)} \cdot P(x_b \mid x_a, k)}$$

component density

$$p(x_b \mid x_a, k) = \frac{p(x_b, x_a \mid k)}{p(x_b \mid k)}$$

mixing coefficient

$$\boxed{\pi'_k = \frac{\pi_k \cdot p(x_b \mid k)}{\sum_j \pi_j \cdot p(x_b \mid j)}}$$

Solution (8) As per question

(a)

$$\because p(X, Z \mid \theta) = \prod_{n=1}^{N} \prod_{k=1}^{K} \left(\pi_k \cdot N(x_n \mid \mu_k, \Sigma_k)\right)^{z_{nk}}$$

taking the log

$$\log p(x, z \mid \theta) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \left(\ln \pi_k + \ln N(x_n \mid \mu_k, \Sigma_k)\right)$$

$$\boxed{\ln p(x, z \mid \theta) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \left(\ln \pi_k - \frac{D}{2} \ln(2\pi) - \frac{1}{2} \ln |\Sigma_k| - \frac{1}{2} (x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k)\right]}$$

$\hookrightarrow$ log-likelihood

(b) MLE → for π_k, μ_k, Σ_k

update for π_k

$$\log p(X, Z|\theta) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \log \pi_k$$

Constraints. $\boxed{\sum_{k=1}^{K} \pi_k = 1}$

using Lagrangian

$$L = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \ln \pi_k + \lambda \left(1 - \sum_{k=1}^{K} \pi_k\right)$$

Take derivative w.r.to π_k

$$\frac{\partial L}{\partial \pi_k} = 0 \Rightarrow \boxed{\pi_k = \frac{\sum_{n=1}^{N} z_{nk}}{\lambda}}$$

using $\sum_{k=1}^{K} \pi_k = 1 \Rightarrow \frac{1}{\lambda} \sum_{k=1}^{K} \sum_{n=1}^{N} z_{nk} = 1 \Rightarrow \frac{N}{\lambda} = 1$

$\boxed{\lambda = N}$

So $$\pi_k = \frac{\sum_{n=1}^{N} z_{nk}}{N}$$

$$\boxed{\pi_k = \frac{N_k}{N}}$$ where → $\boxed{N_k = \sum_{n=1}^{N} z_{nk}}$

update for μ_k

$$\frac{\partial}{\partial \mu_k}\left[\sum_{n=1}^{N} z_{nk}\left(-\frac{1}{2}(x_n-\mu_k)^T \Sigma_k^{-1}(x_n-\mu_k)\right)\right]$$

$$\Rightarrow \sum_{n=1}^{N} z_{nk}\Sigma_k^{-1}(x_n-\mu_k)=0$$

multiply by Σ_k

$$\sum_{n=1}^{N} z_{nk}(x_n-\mu_k)=0$$

$$\mu_k = \frac{\sum_{n=1}^{N} z_{nk}x_n}{\sum_{n=1}^{N} z_{nk}} = \frac{\sum_{n=1}^{N} z_{nk}x_n}{N_k}$$

$$\boxed{\mu_k = \frac{\sum z_{nk}\cdot x_n}{N_k}}$$ where $N_k = \sum_{n=1}^{N} z_{nk}$

update for Σ_k

taking derivative of log-likelihood w.r.to Σ_k

$$\frac{\partial}{\partial \Sigma_k}\left(\sum_{n=1}^{N} z_{nk}\left(-\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x_n-\mu_k)^T\Sigma_k^{-1}(x_n-\mu_k)\right)\right)$$

$$\frac{\partial}{\partial \Sigma}\log|\Sigma| = \Sigma^{-1}$$

$$\frac{\partial}{\partial \Sigma}\left[(x_n-\mu_k)^T\Sigma^{-1}(x_n-\mu_k)\right] = -\Sigma^{-1}(x_n-\mu_k)(x_n-\mu_k)$$

apply these to above expression

$$-\frac{1}{2}\sum_{n=1}^{N} z_{nk}\left(\Sigma_k^{-1} - \Sigma_k^{-1}(x_n-\mu_k)(x_n-\mu_k)^T\Sigma_k^{-1}\right)=0$$

multiply both side Σ_k^{-1}

$$\sum_{n=1}^{N} z_{nk}\left(I - (x_n-\mu_k)(x_n-\mu_k)^T\Sigma_k^{-1}\right)=0$$

$$\boxed{\Sigma_k = \frac{\sum_{n=1}^{N} z_{nk}(x_n-\mu_k)(x_n-\mu_k)^T}{N_k}}$$

where $N_k = \sum_{n=1}^{N} z_{nk}$

for linkage (single)

- It has longer chain like structure which concludes that it can form elongated cluster
- It is more prone the chaing effect where one large cluster for by proficiency adding points

for complete linkage

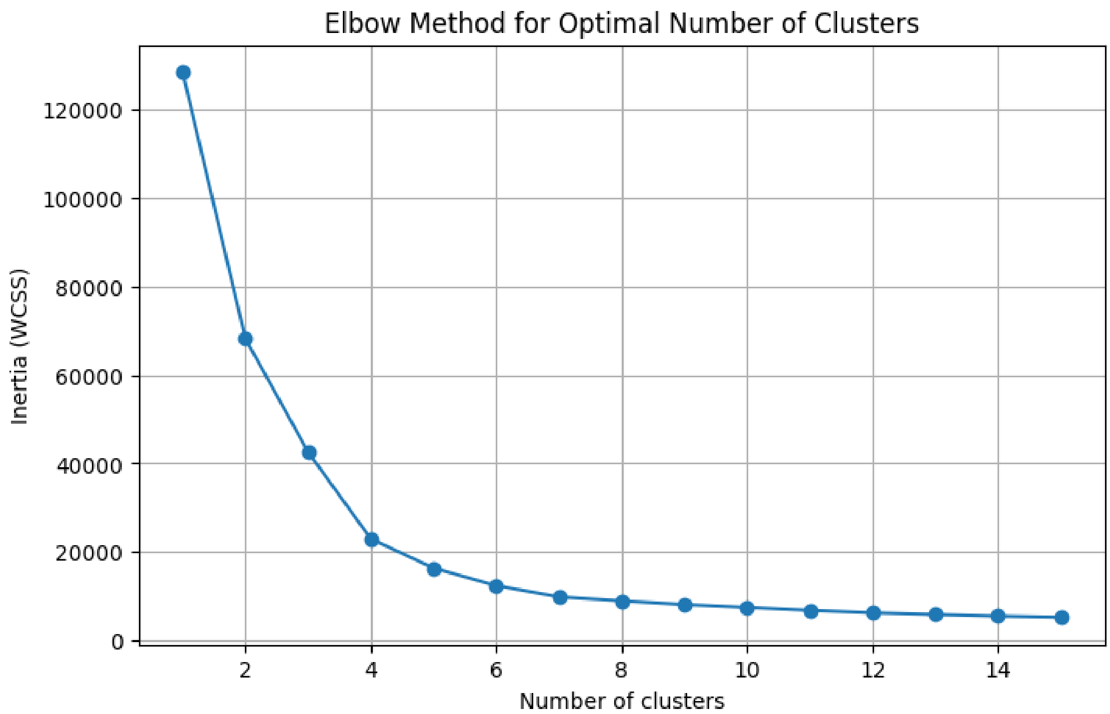
→ Its produces balanced cluster where links are merged

ee708-5

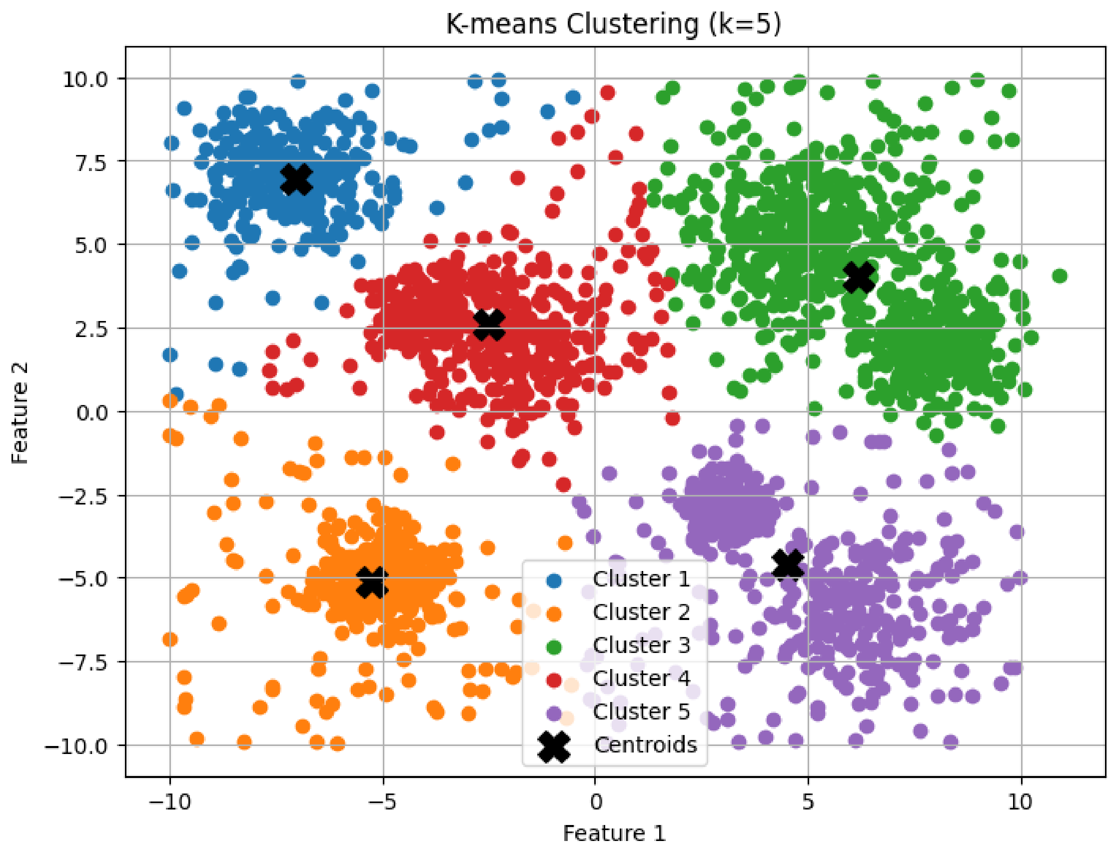
June 20, 2025

##Question : 9

[7]:
```python
# K-means clustering

import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans

# dataset
df_kmeans = pd.read_csv("A3_P1.csv")
X = df_kmeans.values

# Elbow method to find optimal number of clusters
inertias = []
for k in range(1, 16):
    kmeans = KMeans(n_clusters=k, random_state=42, n_init=10)
    kmeans.fit(X)
    inertias.append(kmeans.inertia_)

# Elbow graph
plt.figure(figsize=(8, 5))
plt.plot(range(1, 16), inertias, marker='o')
plt.title('Elbow Method for Optimal Number of Clusters')
plt.xlabel('Number of clusters')
plt.ylabel('Inertia (WCSS)')
plt.grid(True)
plt.show()
```



[8]:
```
#  K-means clustering using the optimal number of clusters
optimal_k = 5 #example
kmeans = KMeans(n_clusters=optimal_k, random_state=42, n_init=10)
labels = kmeans.fit_predict(X)

# Plot clusters
plt.figure(figsize=(8, 6))
for i in range(optimal_k):
    plt.scatter(X[labels == i, 0], X[labels == i, 1], label=f'Cluster {i+1}')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1],
            s=200, c='black', marker='X', label='Centroids')
plt.title(f'K-means Clustering (k={optimal_k})')
plt.xlabel('Feature 1')
plt.ylabel('Feature 2')
plt.legend()
plt.grid(True)
plt.show()
```

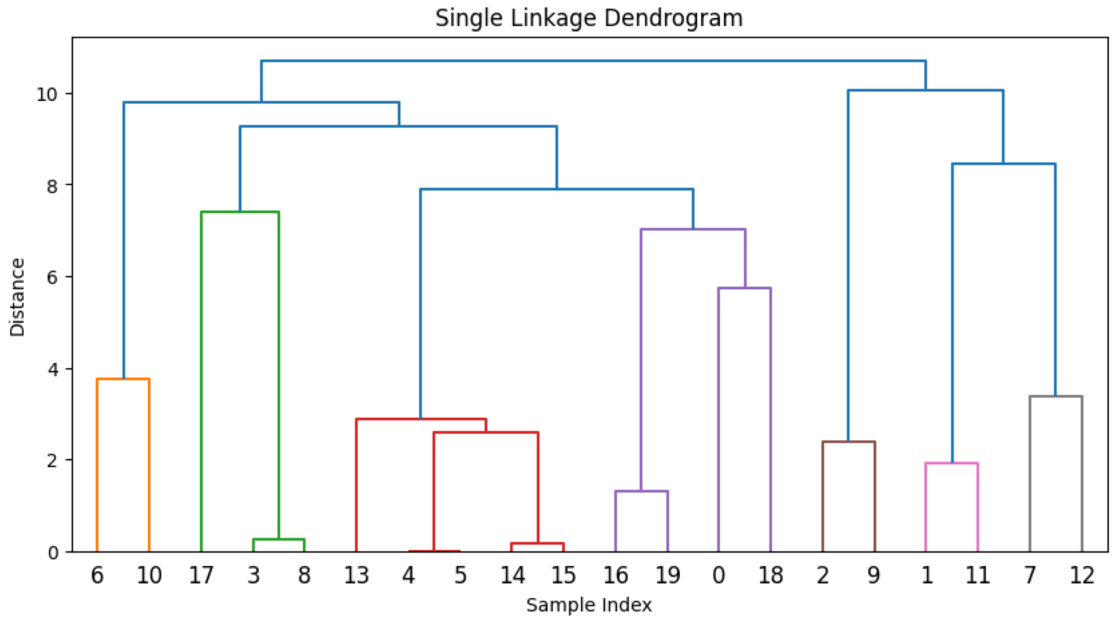


##Question: 10

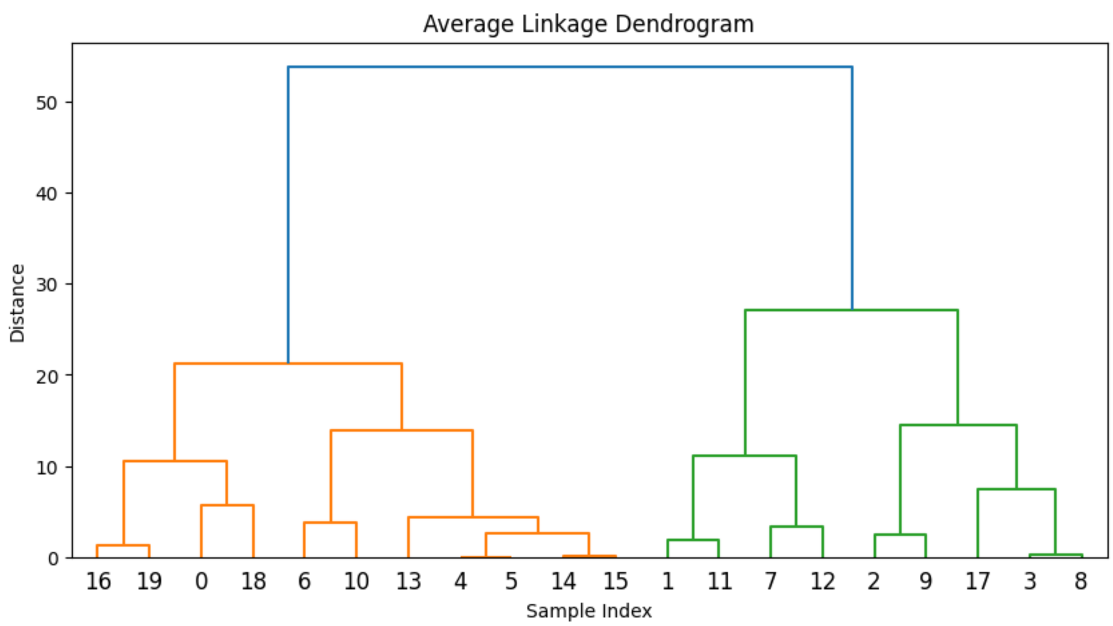
[9]:
```
# Hierarchical Clustering

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram

# data
df = pd.read_csv("A3_P2.csv")
data = df.values

# Distance function (Euclidean)
def euclidean_dist(a, b):
    return np.linalg.norm(a - b)

# Cluster linkage function
def compute_linkage_distance(cluster1, cluster2, method):
    dists = [euclidean_dist(u, v) for u in cluster1 for v in cluster2]
```

```python
    if method == 'single':
        return min(dists)
    elif method == 'average':
        return np.mean(dists)
    elif method == 'complete':
        return max(dists)

# Hierarchical clustering
def hierarchical_clustering(data, method='single'):
    clusters = [[x] for x in data]
    labels = list(range(len(data)))
    merge_log = []
    distances = []

    while len(clusters) > 1:
        min_dist = float('inf')
        best_pair = None
        for i in range(len(clusters)):
            for j in range(i + 1, len(clusters)):
                dist = compute_linkage_distance(clusters[i], clusters[j],␣
↪method)
                if dist < min_dist:
                    min_dist = dist
                    best_pair = (i, j)

        i, j = best_pair
        new_cluster = clusters[i] + clusters[j]
        merge_log.append([labels[i], labels[j], min_dist, len(new_cluster)])
        distances.append(min_dist)

        # Merge clusters and update labels
        clusters.append(new_cluster)
        labels.append(max(labels) + 1)
        for idx in sorted([i, j], reverse=True):
            del clusters[idx]
            del labels[idx]

    return np.array(merge_log)

# Generating and plot dendrograms
for method in ['single', 'average', 'complete']:
    linkage_matrix = hierarchical_clustering(data, method)
    plt.figure(figsize=(10, 5))
    dendrogram(linkage_matrix, labels=[str(i) for i in range(len(data))])
    plt.title(f'{method.capitalize()} Linkage Dendrogram')
    plt.xlabel('Sample Index')
    plt.ylabel('Distance')
```

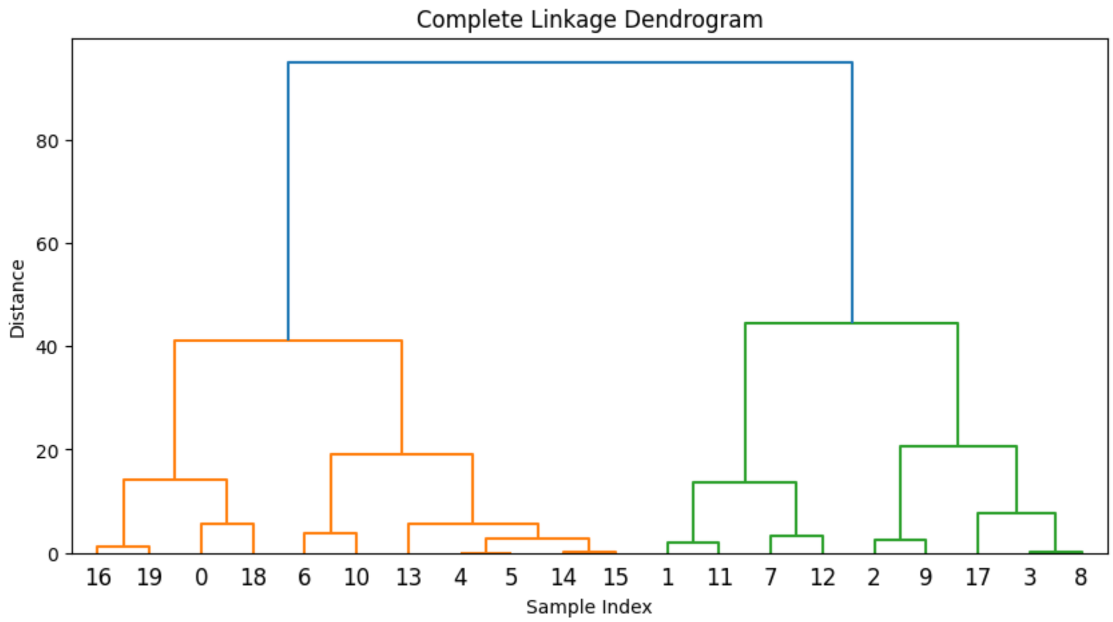
```
plt.show()
```



Single Linkage Dendrogram
Distance
0
2
4
6
8
10
6 10 17 3 8 13 4 5 14 15 16 19 0 18 2 9 1 11 7 12
Sample Index



Average Linkage Dendrogram
Distance
0
10
20
30
40
50
16 19 0 18 6 10 13 4 5 14 15 1 11 7 12 2 9 17 3 8
Sample Index


Complete Linkage Dendrogram
Distance
0
20
40
60
80
16 19 0 18 6 10 13 4 5 14 15 1 11 7 12 2 9 17 3 8
Sample Index