Chapter 5

Clustering Technique

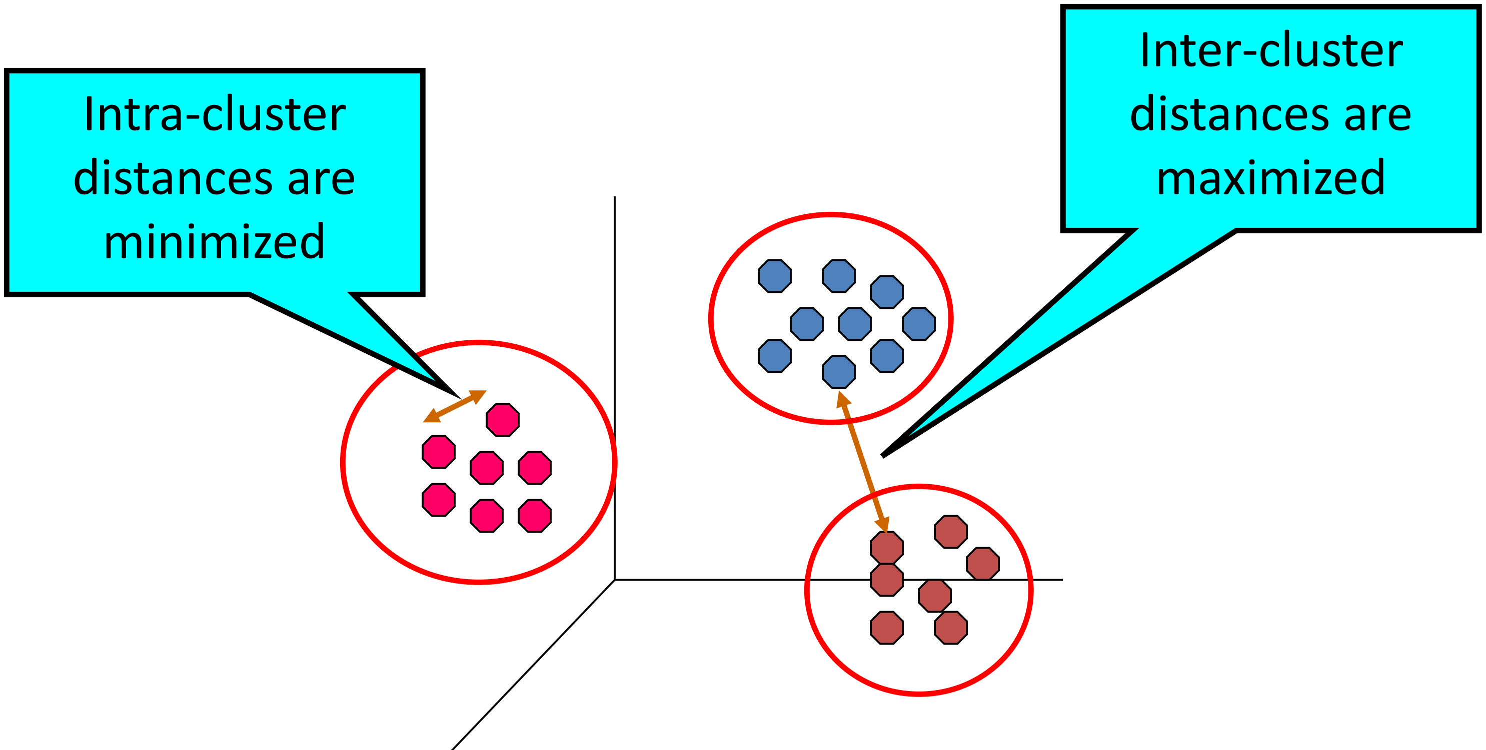
Instructor: Suresh Pokharel
ME in ICT (Asian Institute of Technology, Thailand)
BE in Computer (NCIT, Pokhara University)

What is Cluster Analysis?

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Grouping a set of data objects into clusters
- Clustering is unsupervised classification: no predefined classes
- Clustering is used:
  - As a stand-alone tool to get insight into data distribution
    - Visualization of clusters may unveil important information
  - As a preprocessing step for other algorithms
    - Efficient indexing or compression often relies on clustering

What is Cluster Analysis?

- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



General Applications of Clustering

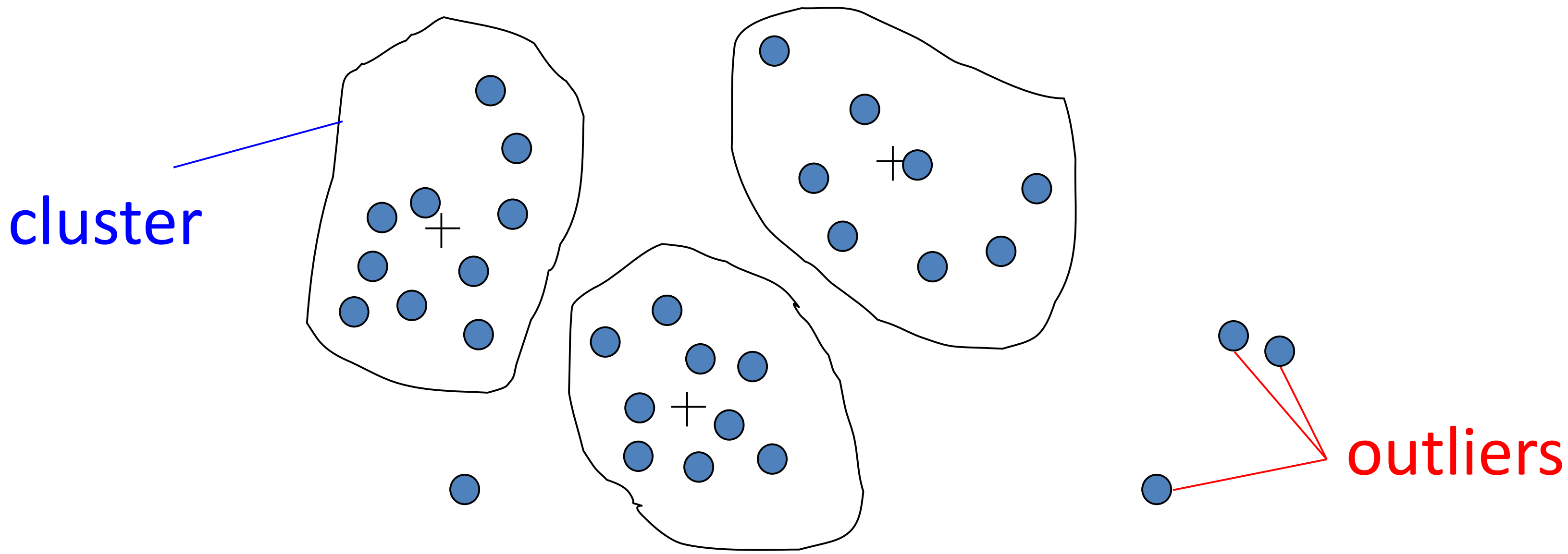
- Pattern Recognition
- Spatial Data Analysis
  - create thematic maps in GIS by clustering feature spaces
  - detect spatial clusters and explain them in spatial data mining
- Image Processing
  - cluster images based on their visual content
- Economic Science (especially market research)
- WWW and IR
  - document classification
  - cluster Weblog data to discover groups of similar access patterns

Examples of Clustering Applications

- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- Land use: Identification of areas of similar land use in an earth observation database
- Insurance: Identifying groups of motor insurance policy holders with a high average claim cost
- City-planning: Identifying groups of houses according to their house type, value, and geographical location

Outliers

- Outliers are objects that do not belong to any cluster or form clusters of very small cardinality



- In some applications we are interested in discovering outliers, not clusters (outlier analysis)

Data Structures

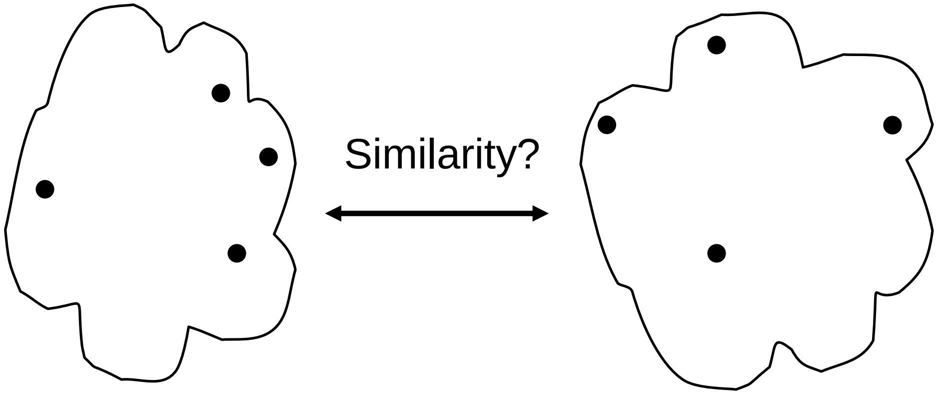
- *data* matrix
  - (two modes)

$$\begin{array}{cc} & \overbrace{\hphantom{x_{11} \quad \cdots \quad x_{1f} \quad \cdots \quad x_{1p}}}^{\text{attributes/dimensions}} \\ \text{tuples/objects} & \begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix} \end{array}$$

- *dissimilarity* or *distance* matrix

$$\begin{array}{cc} & \text{objects} \\ \text{objects} & \begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix} \end{array}$$

How to Define Inter-Cluster Similarity?



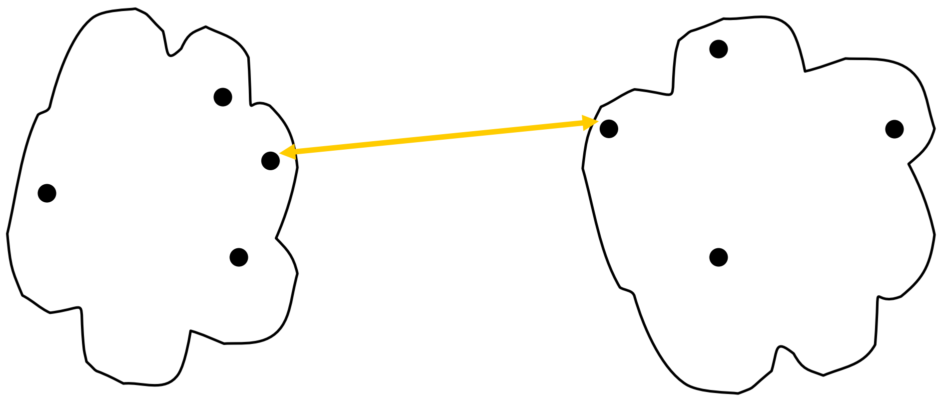
MIN

MAX

Group Average

Distance Between Centroids

How to Define Inter-Cluster Similarity?



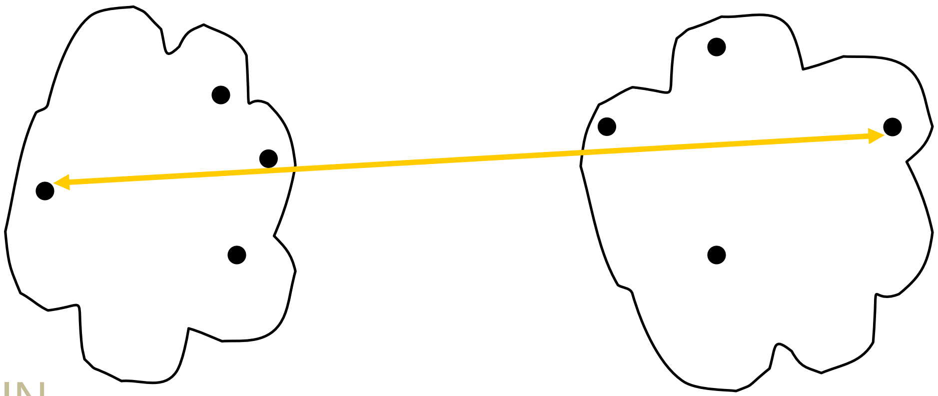
MIN

MAX

Group Average

Distance Between Centroids

How to Define Inter-Cluster Similarity?



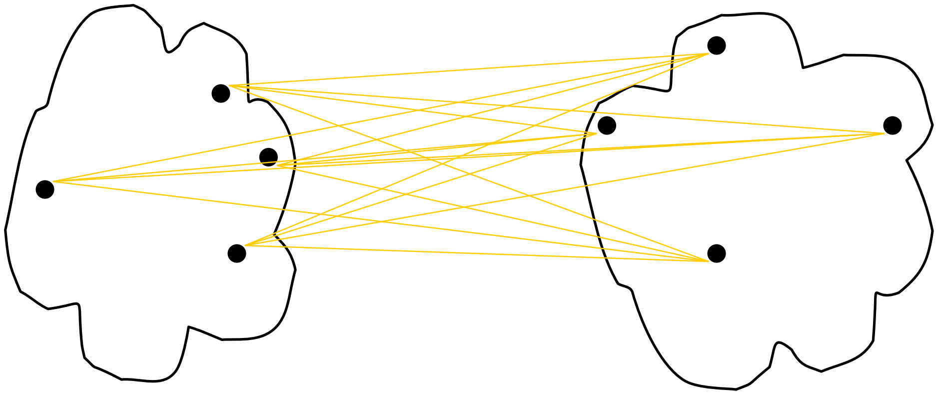
MIN

MAX

Group Average

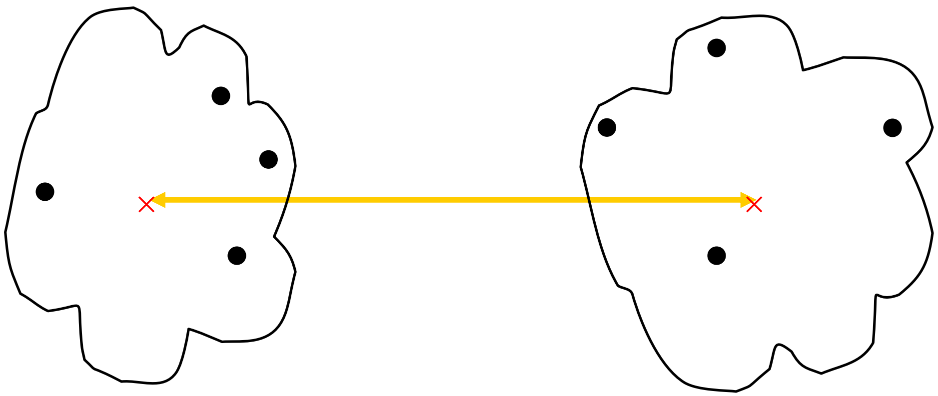
Distance Between Centroids

How to Define Inter-Cluster Similarity?



- MIN
- MAX
- Group Average
- Distance Between Centroids

How to Define Inter-Cluster Similarity?



MIN

MAX

Group Average

Distance Between Centroids

Major Clustering Approaches

- Partitioning algorithms: Construct random partitions and then iteratively refine them by some criterion
- Hierarchical algorithms: Create a hierarchical decomposition of the set of data (or objects) using some criterion
- Density-based: based on connectivity and density functions
- Grid-based: based on a multiple-level granularity structure
- Model-based: A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other

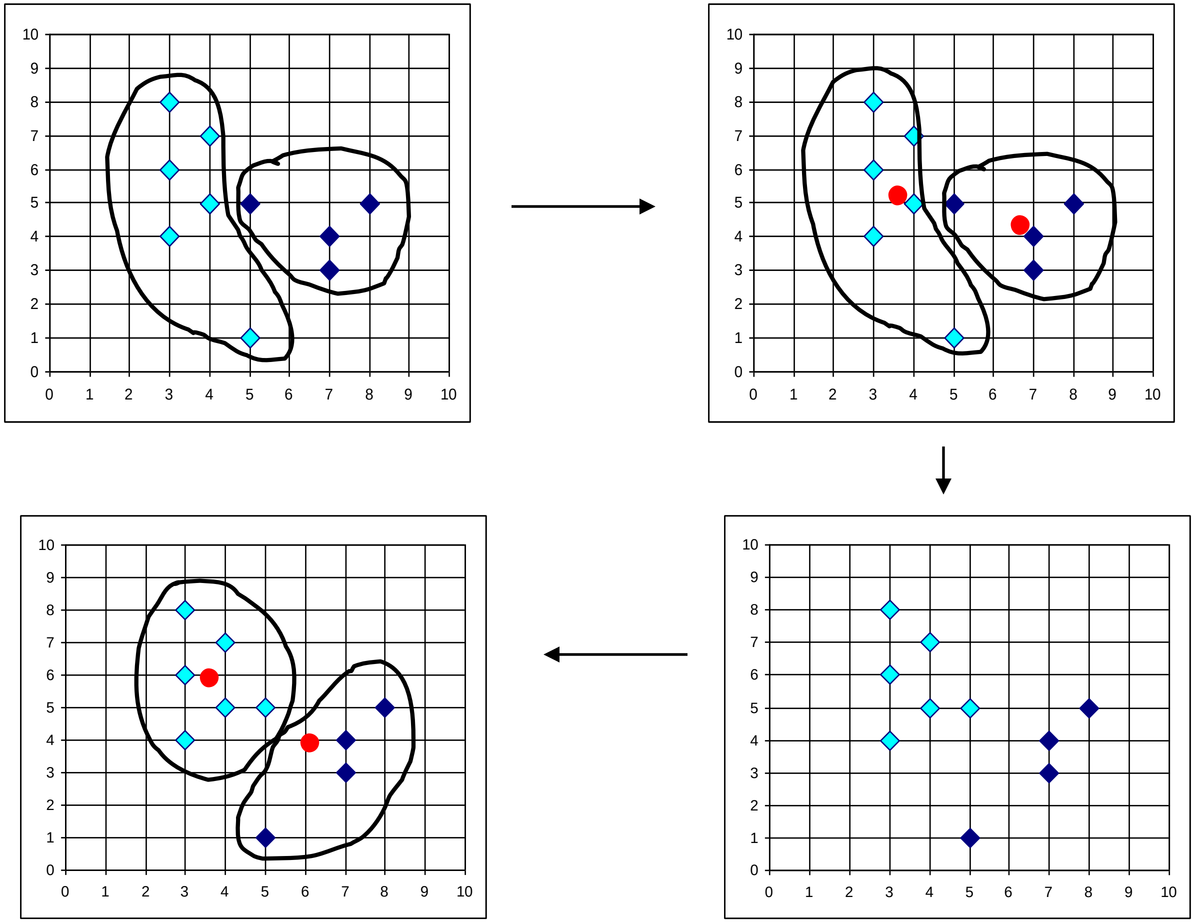
Partitioning Algorithms: Basic Concepts

- Partitioning method: Construct a partition of a database ***D*** of ***n*** objects into a set of ***k*** clusters
- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: *k-means* and *k-medoids* algorithms
  - *k-means* (MacQueen'67): Each cluster is represented by the center of the cluster
  - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

The k-means Clustering Method

- Given *k*, the *k-means* algorithm is implemented in 4 steps:
  1. Partition objects into *k* nonempty subsets
  2. Compute seed points as the centroids of the clusters of the current partition. The centroid is the center (mean point) of the cluster.
  3. Assign each object to the cluster with the nearest seed point.
  4. Go back to Step 2, stop when no more new assignment.

- Example



K-Means example

- 2, 3, 6, 8, 9, 12, 15, 18, 22 – break into 3 clusters
  - Cluster 1 - 2, 8, 15 – mean = 8.3
  - Cluster 2 - 3, 9, 18 – mean = 10
  - Cluster 3 - 6, 12, 22 – mean = 13.3
- Re-assign
  - Cluster 1 - 2, 3, 6, 8, 9 – mean = 5.6
  - Cluster 2 – mean = 0
  - Cluster 3 – 12, 15, 18, 22 – mean = 16.75
- Re-assign
  - Cluster 1 – 3, 6, 8, 9 – mean = 6.5
  - Cluster 2 – 2 – mean = 2
  - Cluster 3 = 12, 15, 18, 22 – mean = 16.75

K-Means example (continued)

- Re-assign
  - – Cluster 1 - 6, 8, 9 – mean = 7.6
  - – Cluster 2 – 2, 3 – mean = 2.5
  - – Cluster 3 – 12, 15, 18, 22 – mean = 16.75
- Re-assign
  - – Cluster 1 - 6, 8, 9 – mean = 7.6
  - – Cluster 2 – 2, 3 - mean = 2.5
  - – Cluster 3 – 12, 15, 18, 22 – mean = 16.75
- No change, so we're done

K-Means example – different starting order

- 2, 3, 6, 8, 9, 12, 15, 18, 22 – break into 3 clusters
  - Cluster 1 - 2, 12, 18 – mean = 10.6
  - Cluster 2 - 6, 9, 22 – mean = 12.3
  - Cluster 3 – 3, 8, 15 – mean = 8.6
- Re-assign
  - Cluster 1 - mean = 0
  - Cluster 2 – 12, 15, 18, 22 - mean = 16.75
  - Cluster 3 – 2, 3, 6, 8, 9 – mean = 5.6
- Re-assign
  - Cluster 1 – 2 – mean = 2
  - Cluster 2 – 12, 15, 18, 22 – mean = 16.75
  - Cluster 3 = 3, 6, 8, 9 – mean = 6.5

K-Means example (continued)

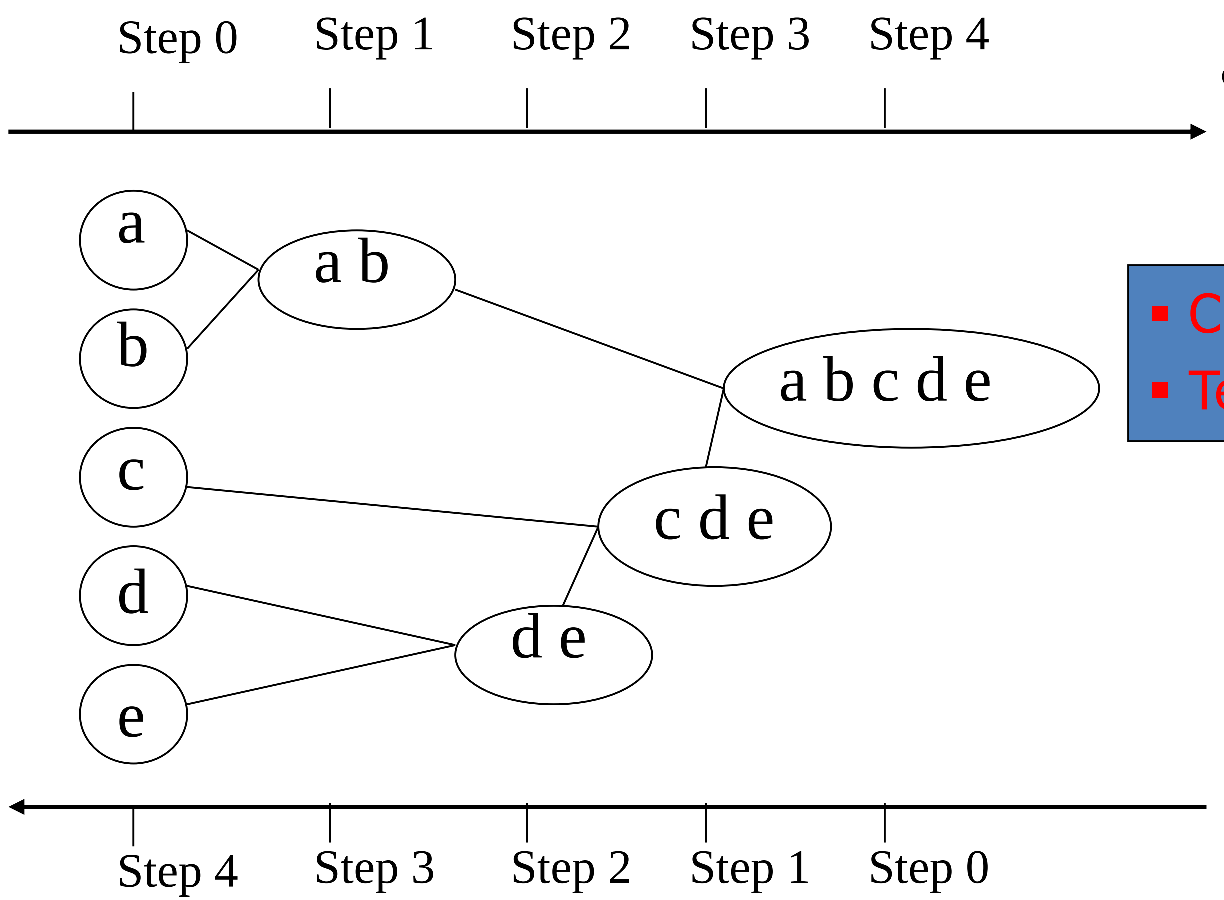
- Re-assign
  - – Cluster 1 – 2, 3 – mean = 2.5
  - – Cluster 2 – 12, 15, 18, 22 – mean = 16.75
  - – Cluster 3 – 6, 8, 9 – mean = 7.6
- Re-assign
  - – Cluster 1 – 2, 3 – mean = 2.5
  - – Cluster 2 – 12, 15, 18, 22 - mean = 16.75
  - – Cluster 3 – 6, 8, 9 – mean = 7.6
- No change, so we're done

Comments on the k-means Method

- Strength
  - *Relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, $k, t << n$.

- Weaknesses
  - Applicable only when *mean* is defined, then what about categorical data?
  - Need to specify *k,* the *number* of clusters, in advance
  - Unable to handle noisy data and *outliers*

Hierarchical Clustering



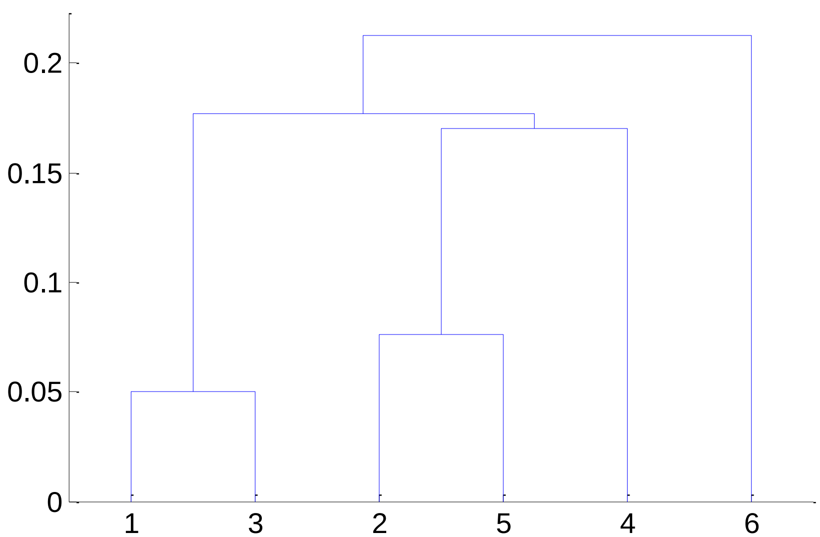
agglomerative
(AGNES)

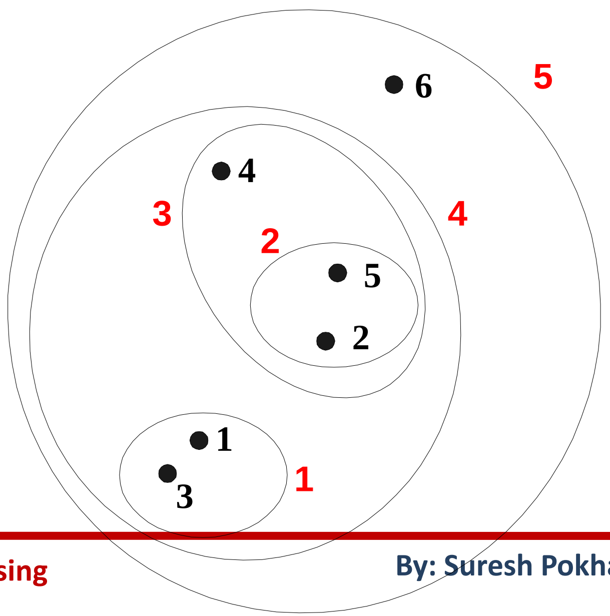
- Cluster distance
- Termination condition

divisive
(DIANA)

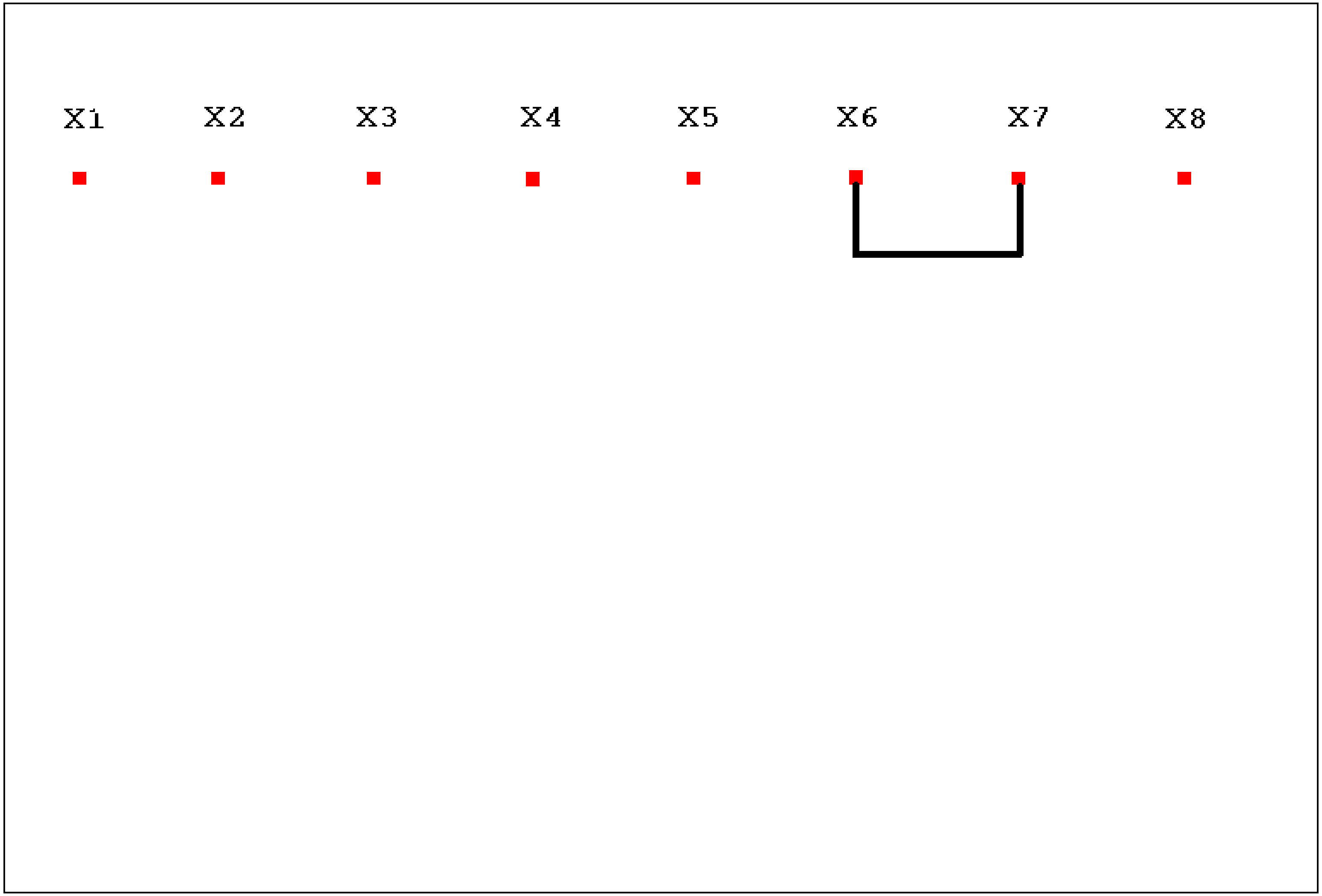
Hierarchical Clustering

- **Produces a set of nested clusters organized as a hierarchical tree**
- **Can be visualized as a “dendrogram”**
  - **A tree like diagram that records the sequences of merges or splits**



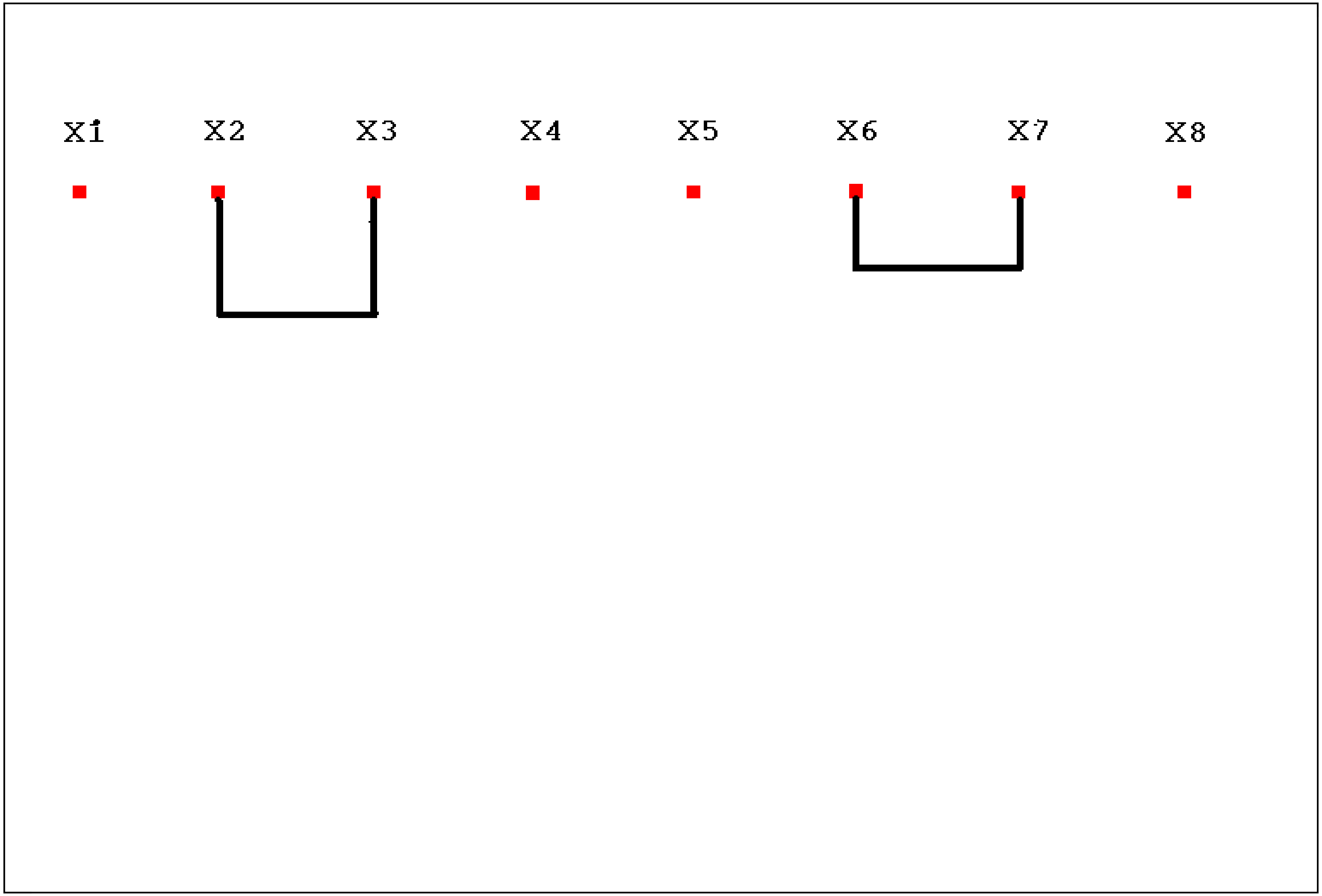


Nearest Neighbor, Level 2, k = 7 clusters.

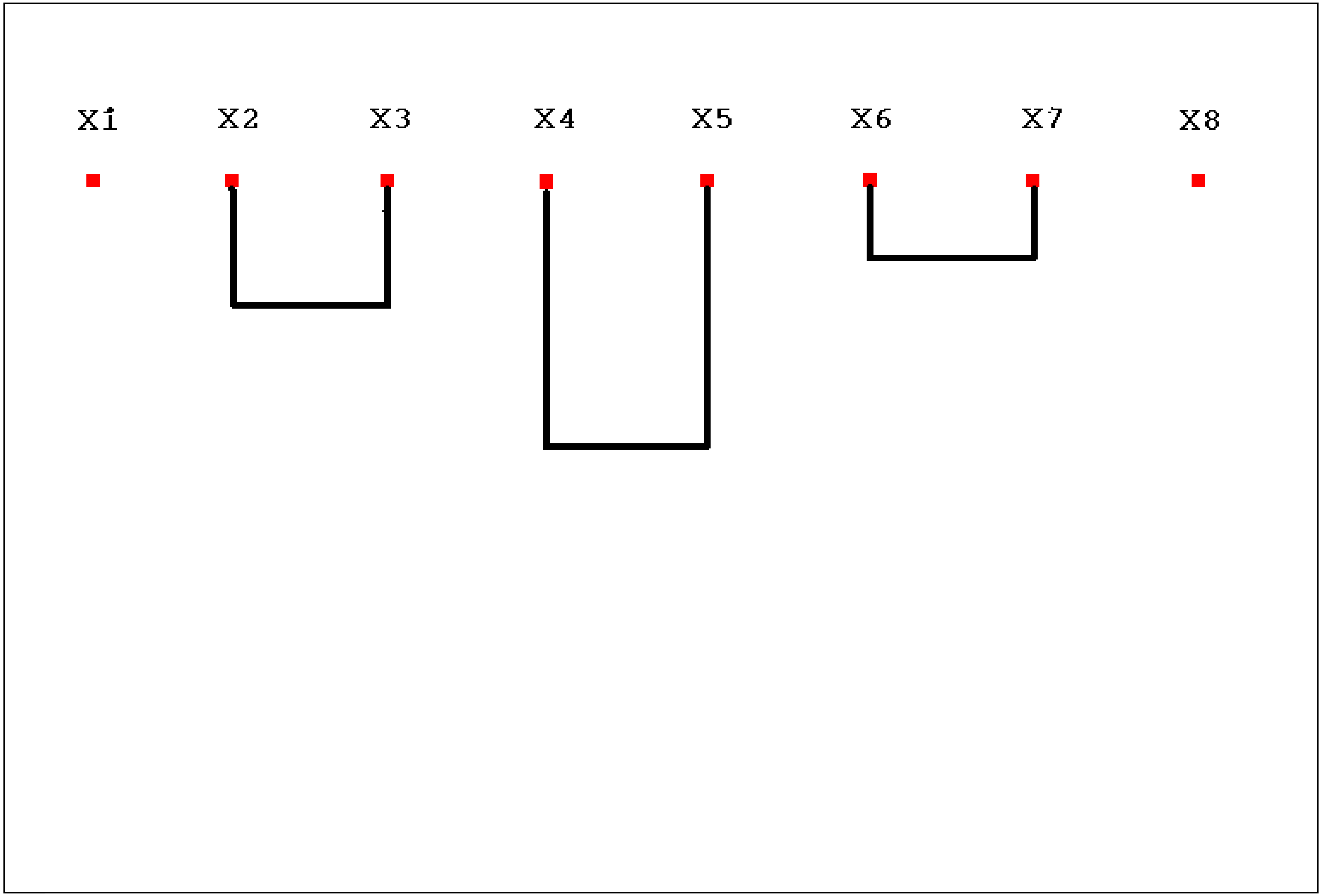


From http://www.stat.unc.edu/postscript/papers/marron/Stat321FDA/RimaIzempresentation.ppt

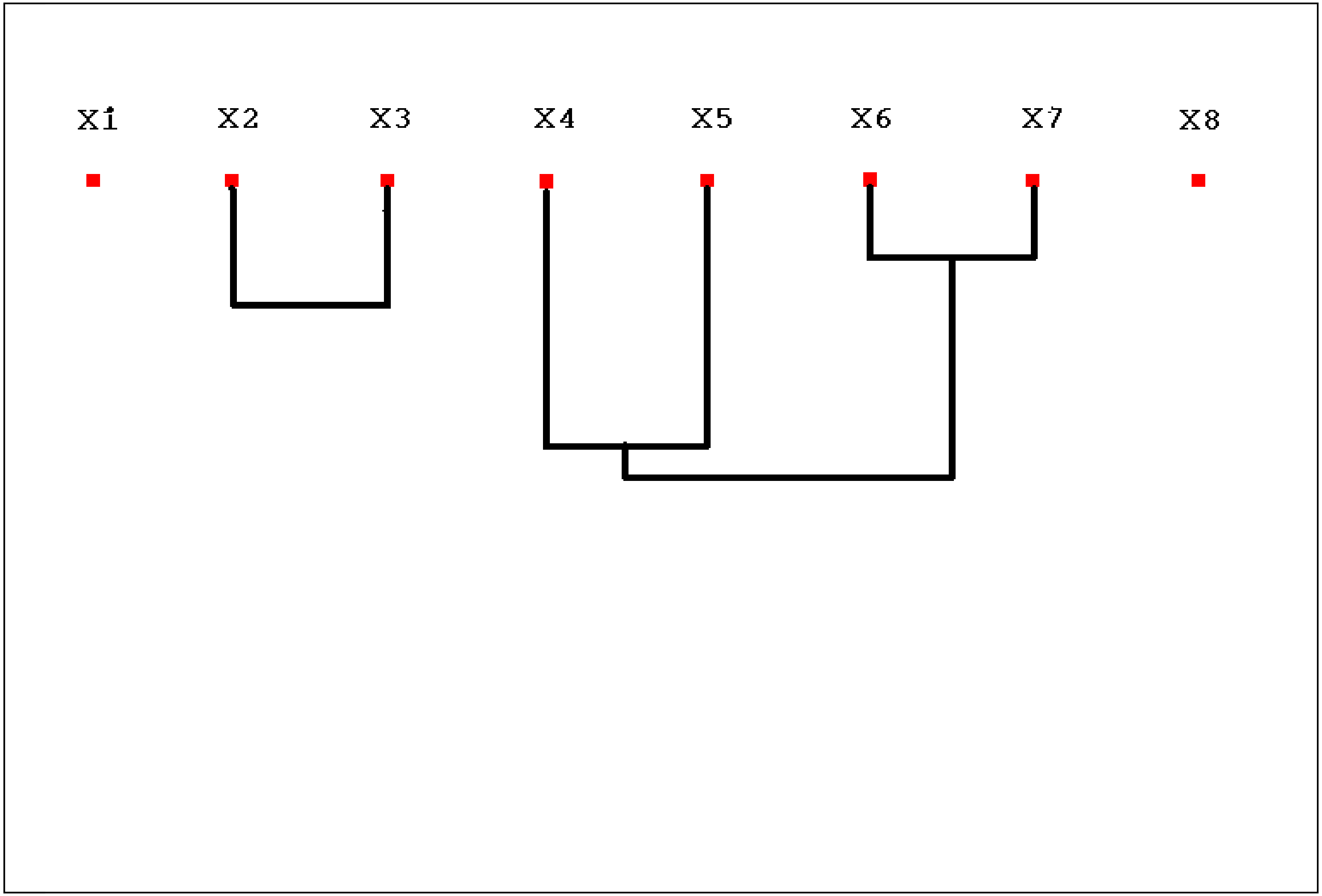
Nearest Neighbor, Level 3, k = 6 clusters.



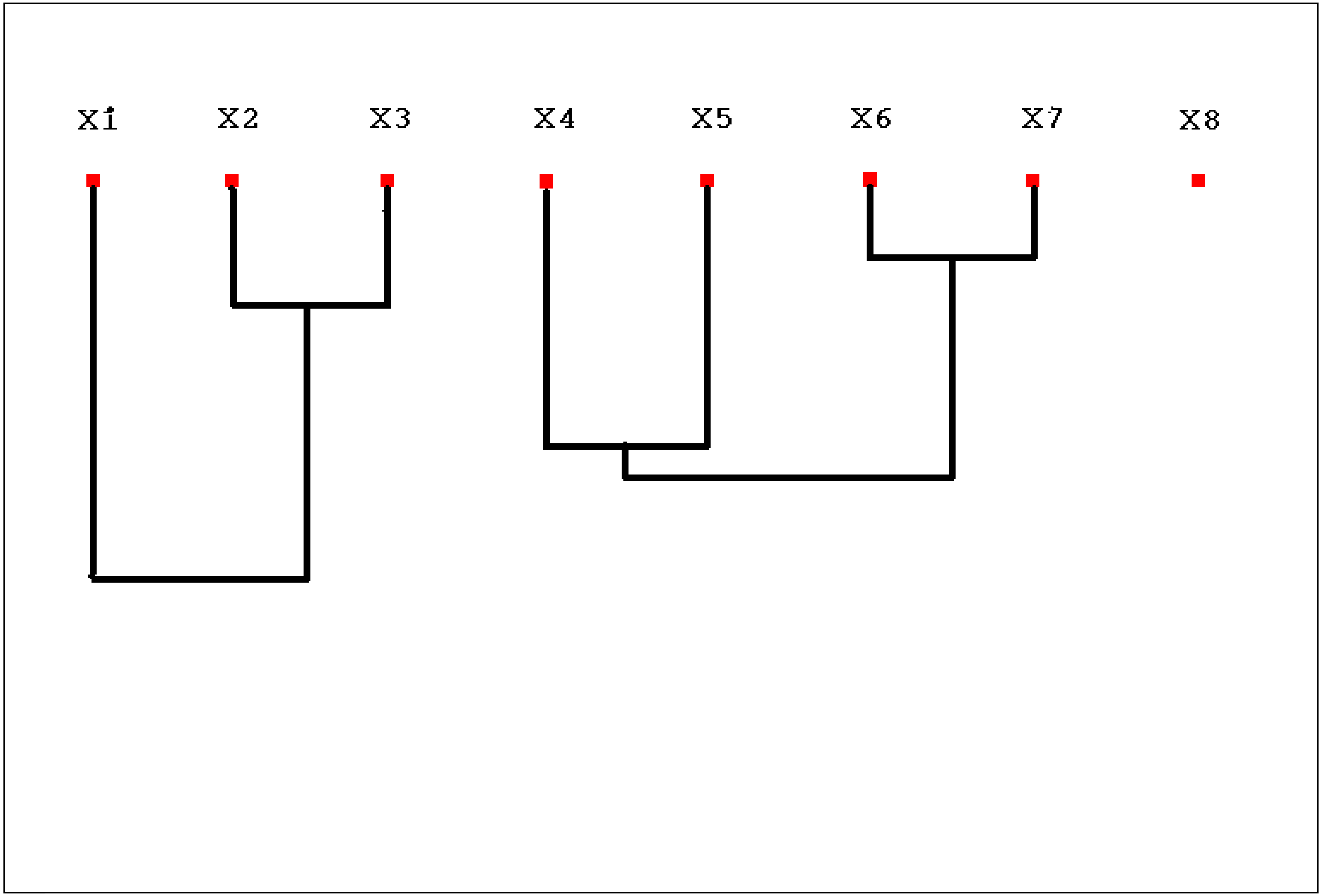
Nearest Neighbor, Level 4, k = 5 clusters.



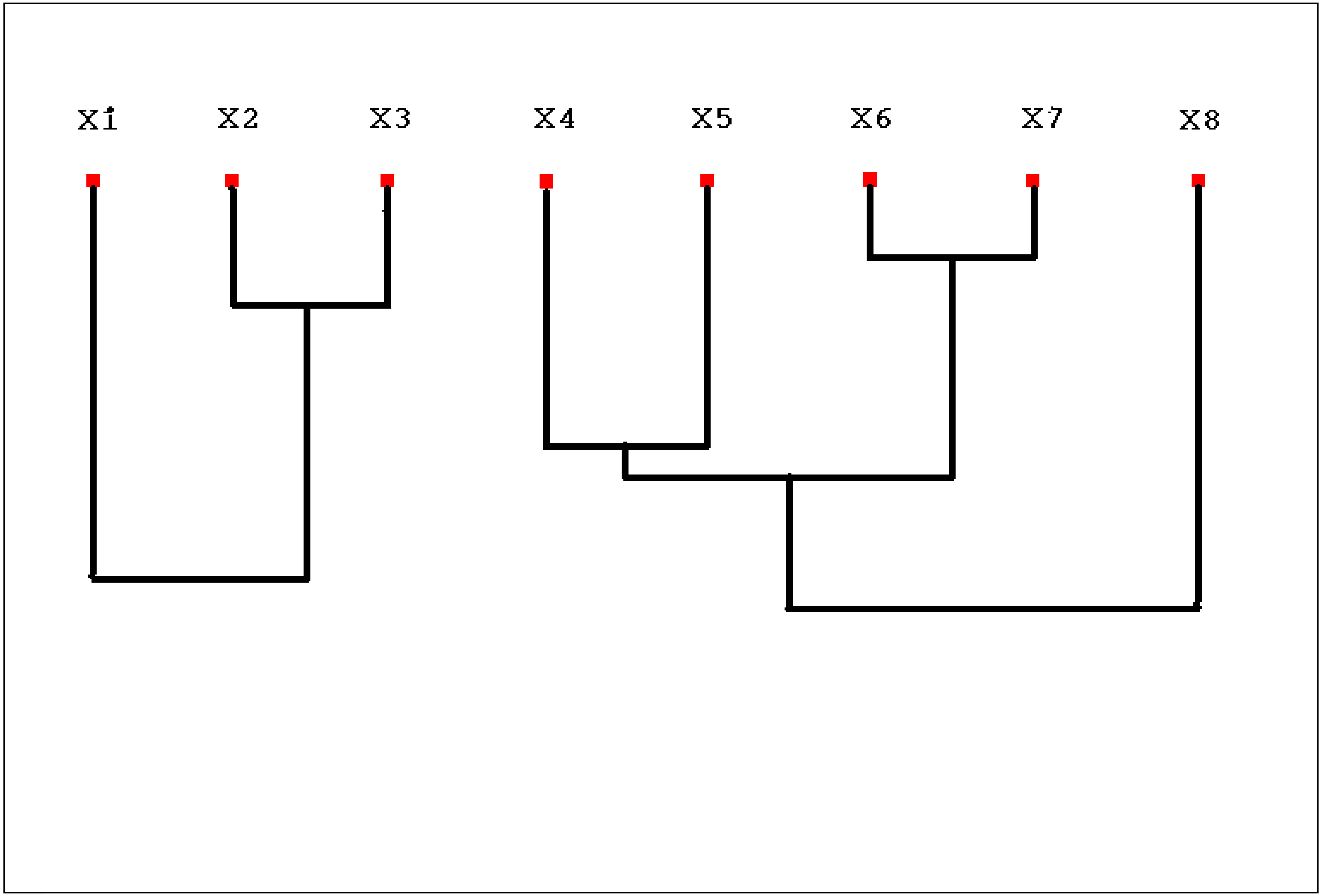
Nearest Neighbor, Level 5, k = 4 clusters.



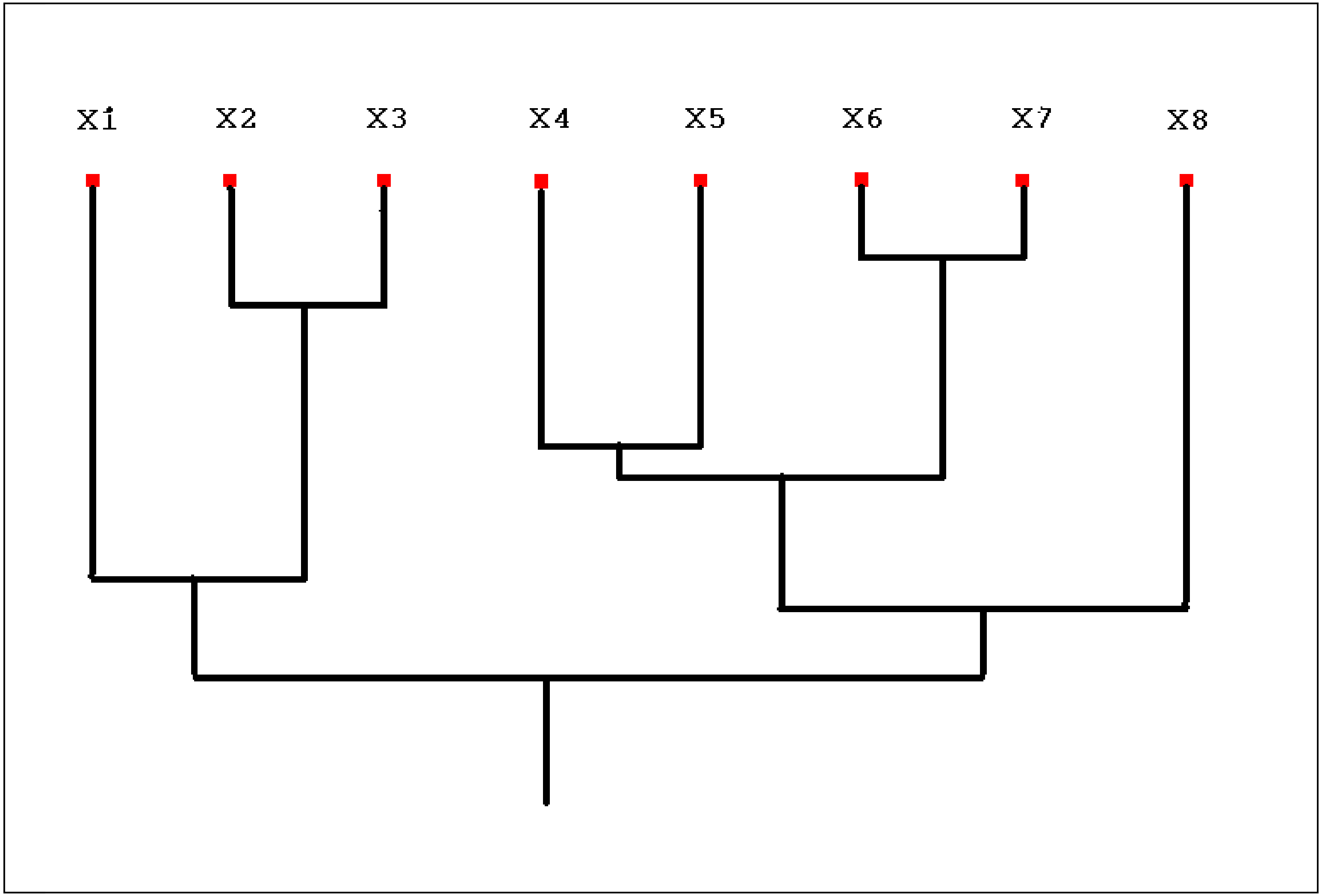
Nearest Neighbor, Level 6, k = 3 clusters.



Nearest Neighbor, Level 7, k = 2 clusters.

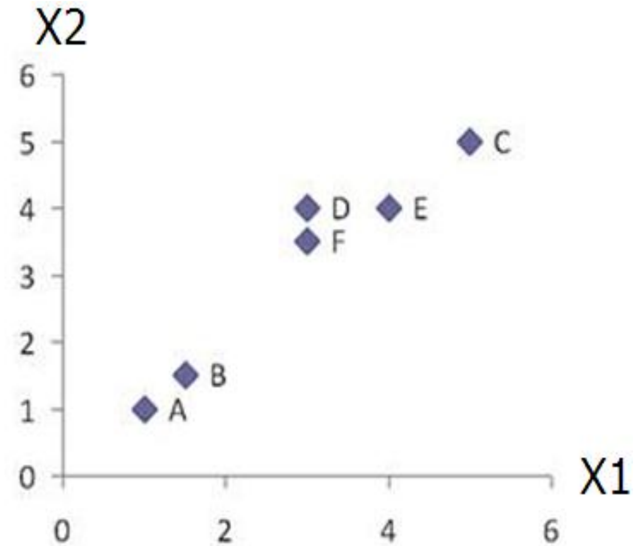


Nearest Neighbor, Level 8, k = 1 cluster.



Example and Demo

- Problem: clustering analysis with agglomerative al



	X1	X2
A	1	1
B	1.5	1.5
C	5	5
D	3	4
E	4	4
F	3	3.5

data matrix

$$d_{AB} = \left((1-1.5)^2 + (1-1.5)^2\right)^{\frac{1}{2}} = \sqrt{\tfrac{1}{2}} = 0.7071$$

$$d_{DF} = \left((3-3)^2 + (4-3.5)^2\right)^{\frac{1}{2}} = 0.5$$

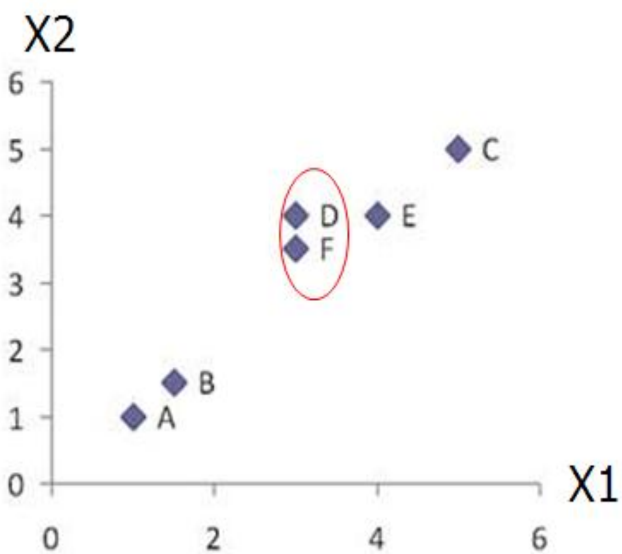
Euclidean distance

Dist	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

distance matrix

Example and Demo

- Merge two closest clusters



Dist	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	?	4.24
B	0.71	0.00	4.95	?	3.54
C	5.66	4.95	0.00	?	1.41
D, F	?	?	?	0.00	?
E	4.24	3.54	1.41	?	0.00

Example and Demo

- Update distance matrix

Dist	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

$$d_{(D,F)\mapsto A} = \min(d_{DA}, d_{FA}) = \min(3.61, 3.20) = 3.20$$

$$d_{(D,F)\mapsto B} = \min(d_{DB}, d_{FB}) = \min(2.92, 2.50) = 2.50$$

$$d_{(D,F)\mapsto C} = \min(d_{DC}, d_{FC}) = \min(2.24, 2.50) = 2.24$$

$$d_{E\to(D,F)} = \min(d_{ED}, d_{EF}) = \min(1.00, 1.12) = 1.00$$

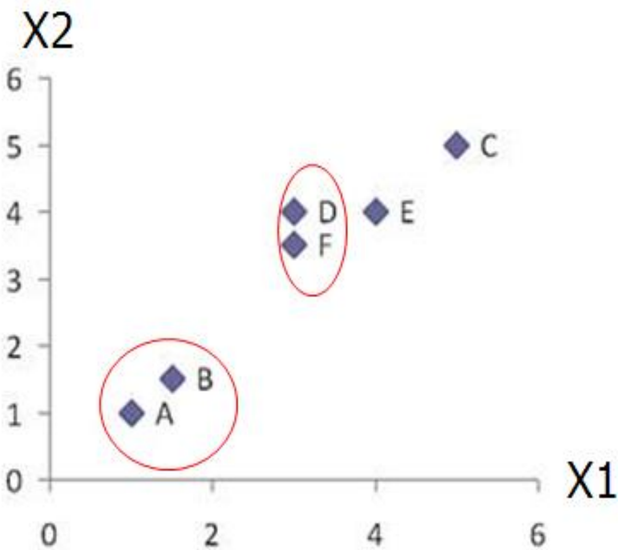
Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	?	4.24
B	0.71	0.00	4.95	?	3.54
C	5.66	4.95	0.00	?	1.41
D, F	?	?	?	0.00	?
E	4.24	3.54	1.41	?	0.00

Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	3.20	4.24
B	0.71	0.00	4.95	2.50	3.54
C	5.66	4.95	0.00	2.24	1.41
D, F	**3.20**	**2.50**	**2.24**	0.00	1.00
E	4.24	3.54	1.41	**1.00**	0.00

Example and Demo

- Merge two closest clusters



Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	3.20	4.24
B	0.71	0.00	4.95	2.50	3.54
C	5.66	4.95	0.00	2.24	1.41
D, F	**3.20**	**2.50**	**2.24**	0.00	1.00
E	4.24	3.54	1.41	**1.00**	0.00

Dist	A,B	C	(D, F)	E
A,B	0	?	?	?
C	?	0	2.24	1.41
(D, F)	?	2.24	0	1.00
E	?	1.41	1.00	0

Example and Demo

- Update distance matrix

Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	3.20	4.24
B	0.71	0.00	4.95	2.50	3.54
C	5.66	4.95	0.00	2.24	1.41
D, F	**3.20**	**2.50**	**2.24**	0.00	1.00
E	4.24	3.54	1.41	**1.00**	0.00

$$d_{C\to(A,B)} = \min\left(d_{CA}, d_{CB}\right) = \min\left(5.66, 4.95\right) = 4.95$$

$$d_{(D,F)\mapsto(A,B)} = \min\left(d_{DA}, d_{DB}, d_{FA}, d_{FB}\right) = \min\left(3.61, 2.92, 3.20, 2.50\right) = 2.50$$

$$d_{E\to(A,B)} = \min\left(d_{EA}, d_{EB}\right) = \min\left(4.24, 3.54\right) = 3.54$$

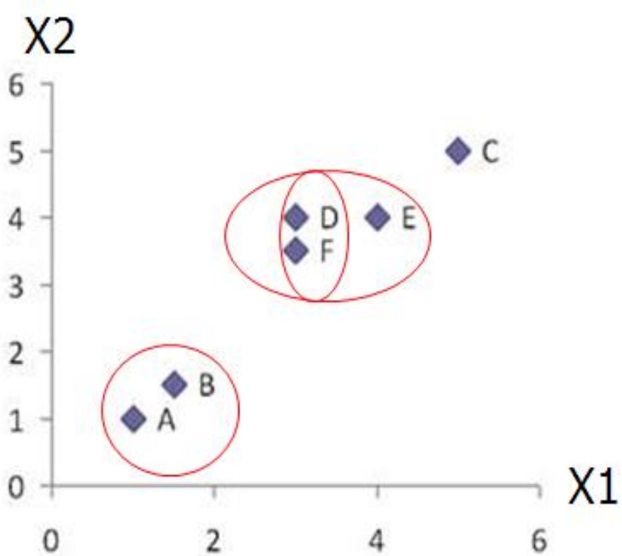
Dist	A,B	C	(D, F)	E
A,B	0	?	?	?
C	**?**	0	2.24	1.41
(D, F)	**?**	2.24	0	1.00
E	**?**	1.41	1.00	0

Min Distance (Single Linkage)

Dist	A,B	C	(D, F)	E
A,B	0	4.95	2.50	3.54
C	**4.95**	0	2.24	1.41
(D, F)	**2.50**	2.24	0	1.00
E	**3.54**	1.41	1.00	0

Example and Demo

- Merge two closest clusters/update distance matrix



Min Distance (Single Linkage)

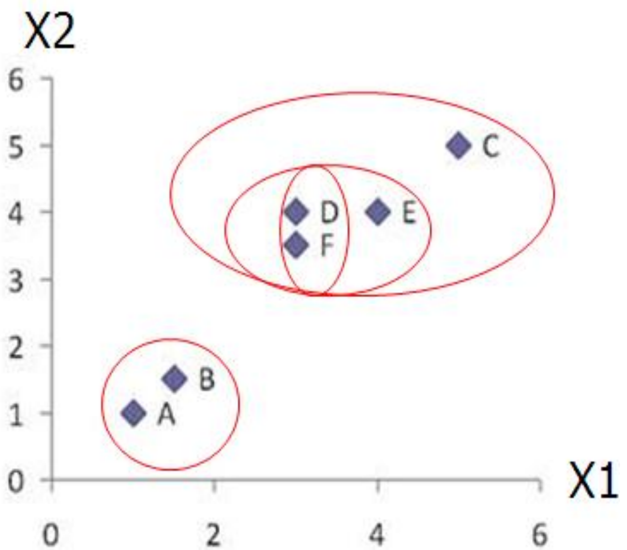
Dist	A,B	C	(D, F)	E
A,B	0	4.95	2.50	3.54
C	**4.95**	0	2.24	1.41
(D, F)	**2.50**	2.24	0	1.00
E	**3.54**	1.41	1.00	0

Min Distance (Single Linkage)

Dist	(A,B)	C	(D, F), E
(A,B)	0.00	4.95	2.50
C	4.95	0.00	1.41
(D, F), E	**2.50**	**1.41**	0.00

Example and Demo

- Merge two closest clusters/update distance matrix



Min Distance (Single Linkage)

Dist	(A,B)	C	(D, F), E
(A,B)	0.00	4.95	2.50
C	4.95	0.00	1.41
(D, F), E	2.50	1.41	0.00

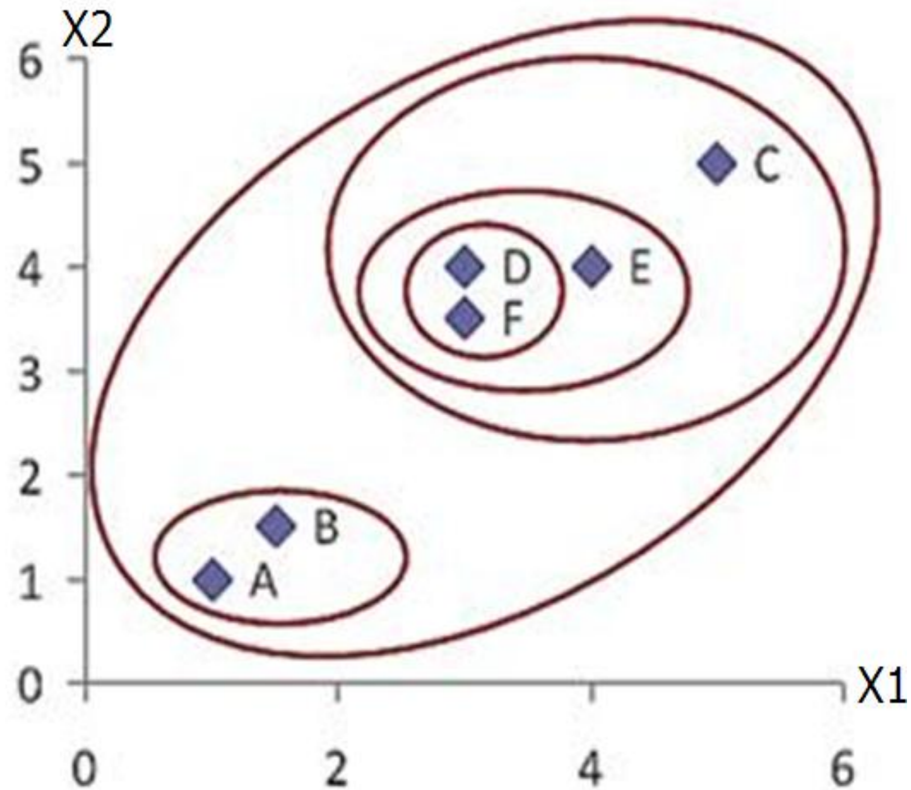
Min Distance (Single Linkage)

Dist	(A,B)	(D, F), E),C
(A,B)	0.00	2.50
((D, F), E),C	2.50	0.00

Example and Demo

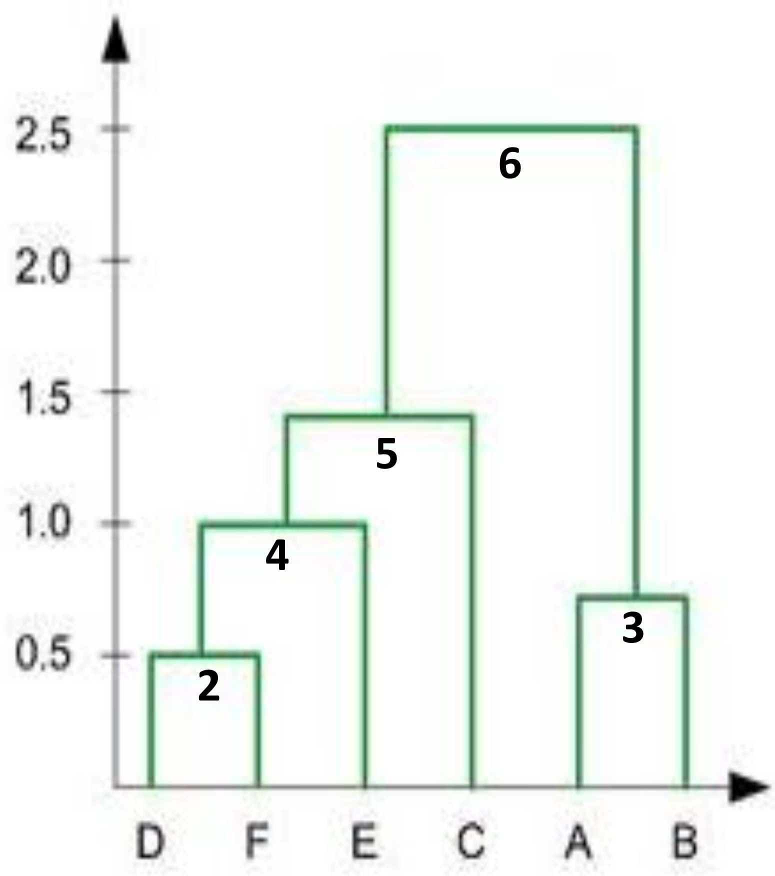
- Final result (meeting termination condition)

	X1	X2
A	1	1
B	1.5	1.5
C	5	5
D	3	4
E	4	4
F	3	3.5



Example and Demo

- Dendrogram tree representation



1. In the beginning we have 6 clusters: A, B, C, D, E and F
2. We merge cluster D and F into cluster (D, F) at distance 0.50
3. We merge cluster A and cluster B into (A, B) at distance 0.71
4. We merge cluster E and (D, F) into ((D, F), E) at distance 1.00
5. We merge cluster ((D, F), E) and C into (((D, F), E), C) at distance 1.41
6. We merge cluster (((D, F), E), C) and (A, B) into ((((D, F), E), C), (A, B)) at distance 2.50
7. The last cluster contain all the objects, thus conclude the computation


Q&A