Master Degree in Artificial Intelligence for Science and Technology

Cluster Analysis: Hierarchical Clustering

UNIVERSITA' DEGLI STUDI DI MILANO BICOCCA

Fabio Stella

Department of Informatics, Systems and Communication

University of Milan-Bicocca

fabio.stella@unimib.it

OUTLOOK

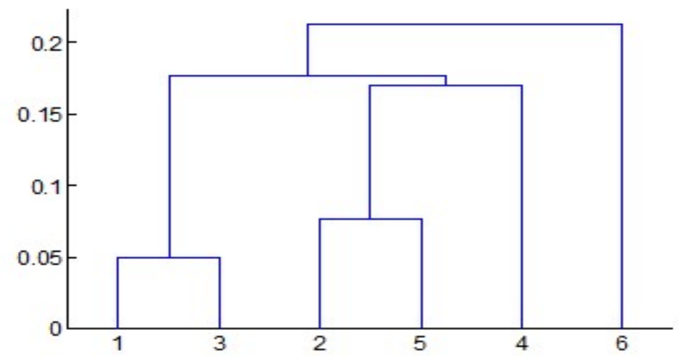
- Concept
- Strengths
- Types
  - ➢ Agglomerative
    - – single linkage
    - – complete linkage
    - – average linkage
    - – Ward's method
  - ➢ Divisive
- Complexity
- Limitations

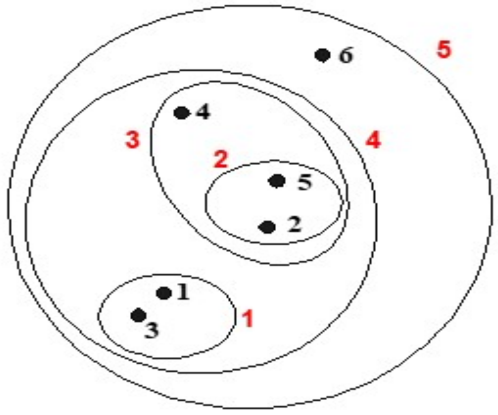
CONCEPT

— produces a set of **NESTED CLUSTERS** organized as a **HIERARCHICAL TREE**

— can be visualized as a **DENDROGRAM**

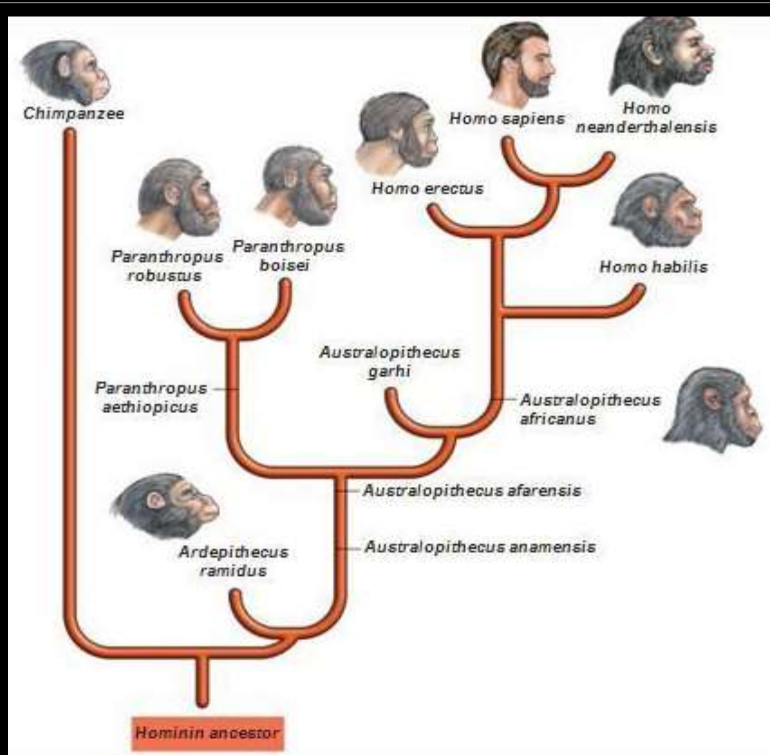
- a tree like diagram that records the sequences of merges or splits





STRENGTHS

- — do not have to assume any particular number of clusters
  - any desired number of clusters can be obtained by 'cutting' the dendrogram at the proper level
- — they may correspond to meaningful taxonomies
  - example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, …)



TYPES OF CLUSTERING

- **AGGLOMERATIVE**
  - start with the points as individual clusters
  - at each step, merge the closest pair of clusters until only one cluster (or k clusters) left

- **DIVISIVE**
  - start with one, all-inclusive cluster
  - at each step, split a cluster until each cluster contains an individual point (or there are k clusters)

- Traditional hierarchical algorithms use a similarity or distance matrix
  - merge or split one cluster at a time

AGGLOMERATIVE CLUSTERING

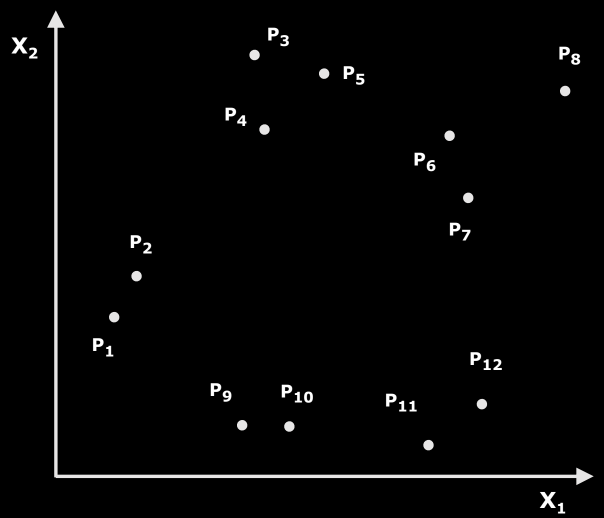
— **KEY IDEA**: successively merge closest clusters

BASIC ALGORITHM

1. Compute the proximity matrix
2. Let each data point be a cluster
3. **REPEAT**
4. Merge the two closest clusters
5. Update the proximity matrix
6. **UNTIL** only a single cluster remains

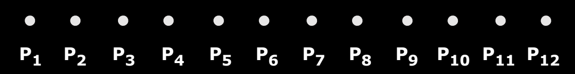
- Key operation is the computation of the proximity of two clusters
- Different approaches to defining the distance between clusters distinguish the different algorithms

AGGLOMERATIVE CLUSTERING



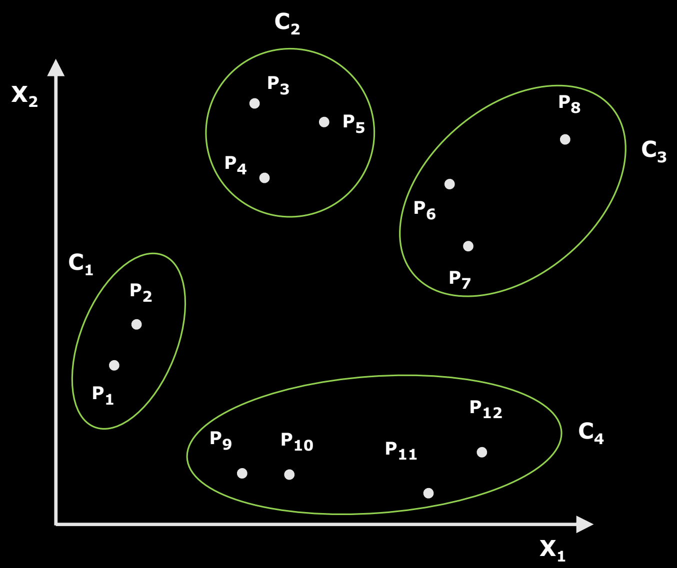
- **STEP 1:** compute the proximity matrix

	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12
p1												
p2												
p3												
p4												
p5												
p6												
p7												
p8												
p9												
p10												
p11												
p12												



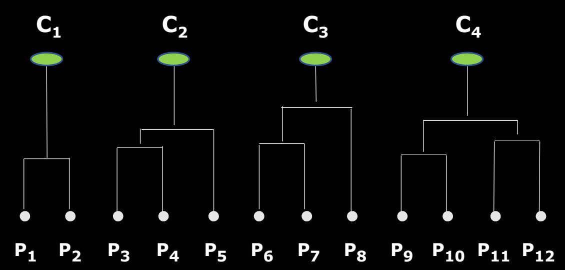
- **STEP 2:** each point is a cluster

AGGLOMERATIVE CLUSTERING



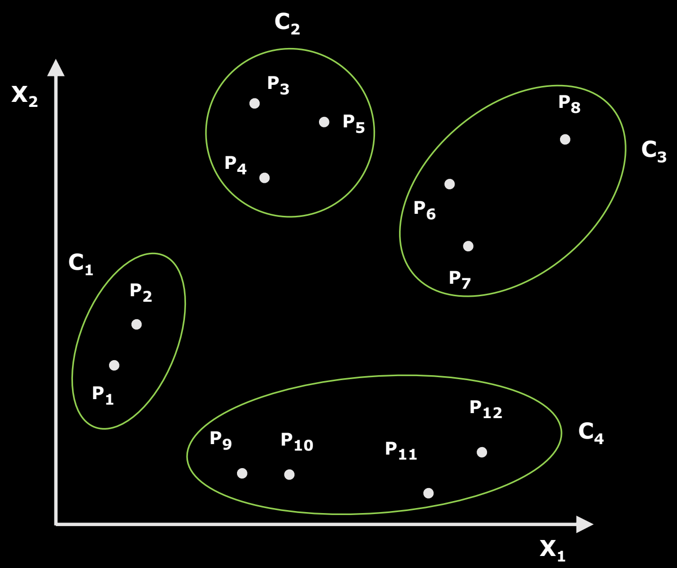
	C1	C2	C3	C4
C1				
C2				
C3				
C4				

PROXIMITY MATRIX



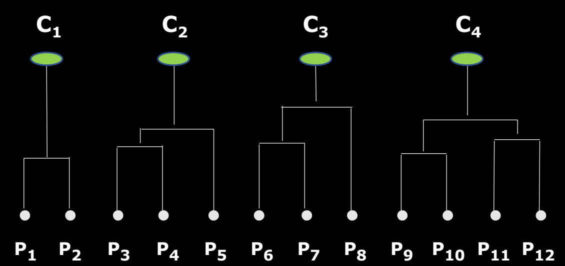
After some merging steps we have some clusters

AGGLOMERATIVE CLUSTERING



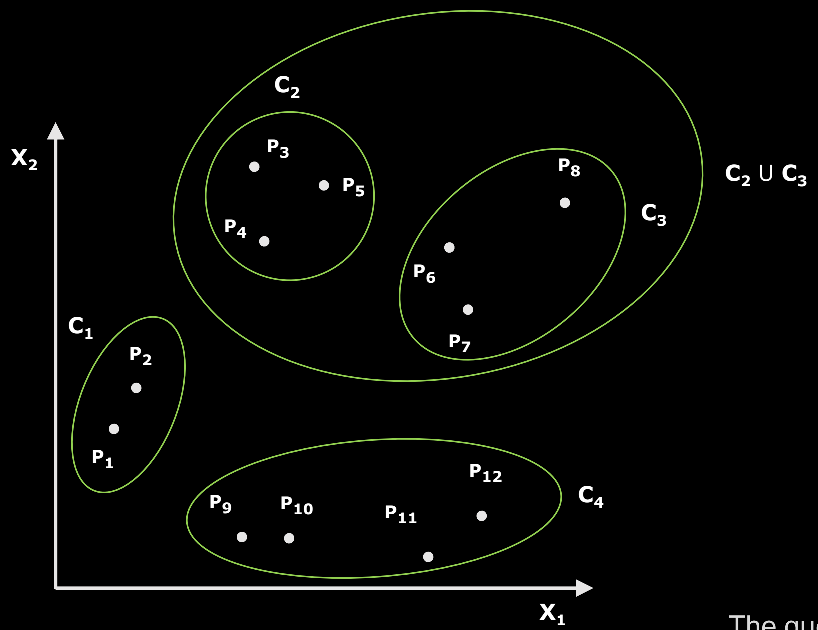
	C1	C2	C3	C4
C1				
C2				
C3				
C4				

PROXIMITY MATRIX



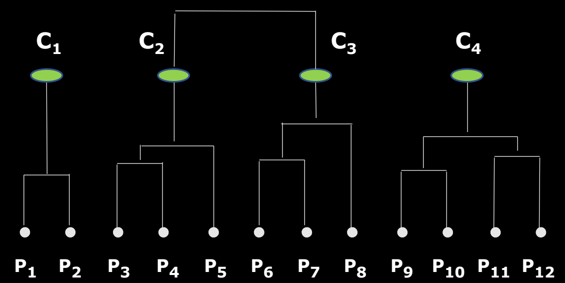
We want to merge the two closest clusters (C_2 and C_3) and update the proximity matrix.

AGGLOMERATIVE CLUSTERING



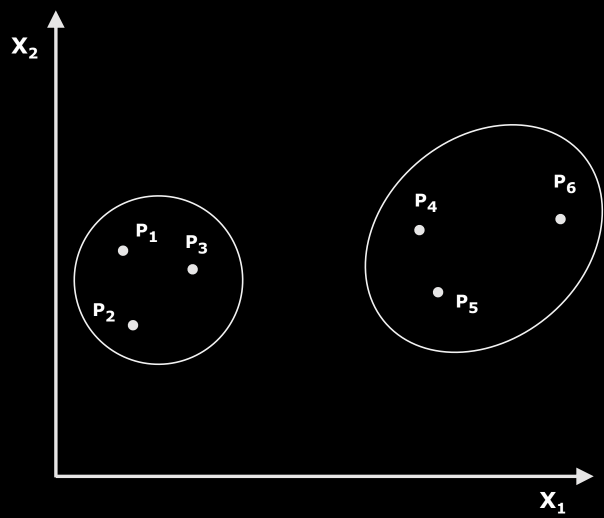
	C1	C2 C3	C4
C1		?	
C2 C3	?	?	?
C4		?	

PROXIMITY MATRIX



The question is "How do we update the proximity matrix?"

HOW TO DEFINE INTER-CLUSTER PROXIMITY?

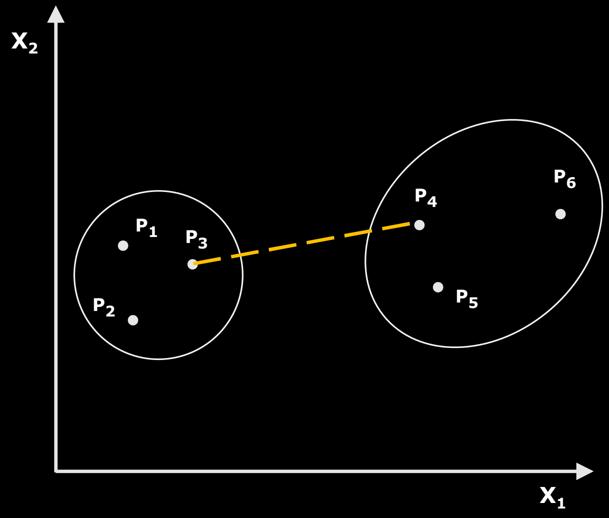


	C1	C2	C3	C4
C1				
C2				
C3				
C4				

PROXIMITY MATRIX

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  — Ward's method uses squared error

HOW TO DEFINE INTER-CLUSTER PROXIMITY?

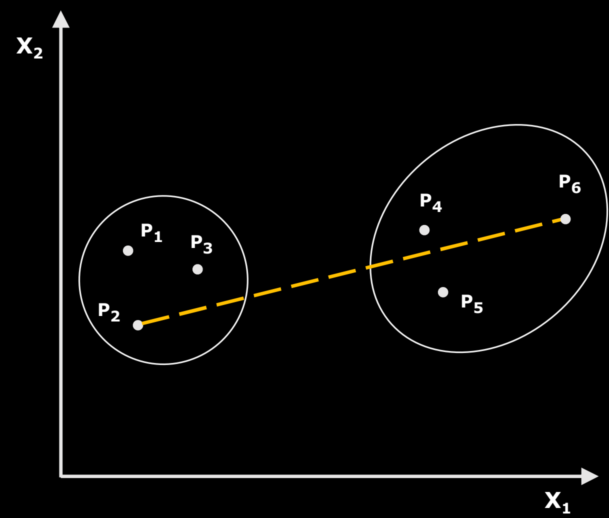


	C1	C2	C3	C4
C1				
C2				
C3				
C4				

PROXIMITY MATRIX

- **MIN**
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's method uses squared error

HOW TO DEFINE INTER-CLUSTER PROXIMITY?

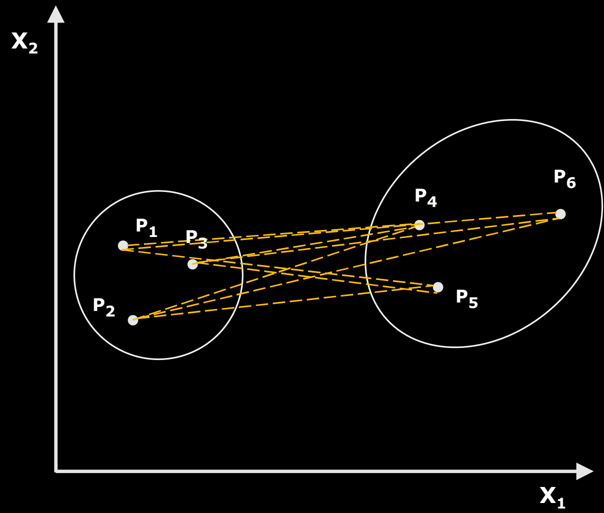


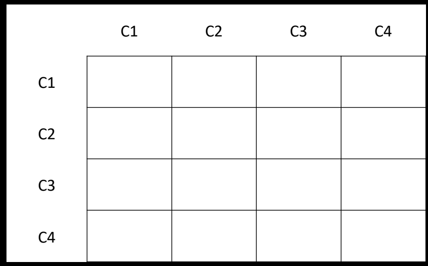
	C1	C2	C3	C4
C1				
C2				
C3				
C4				

PROXIMITY MATRIX

- MIN
- **MAX**
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's method uses squared error

HOW TO DEFINE INTER-CLUSTER PROXIMITY?

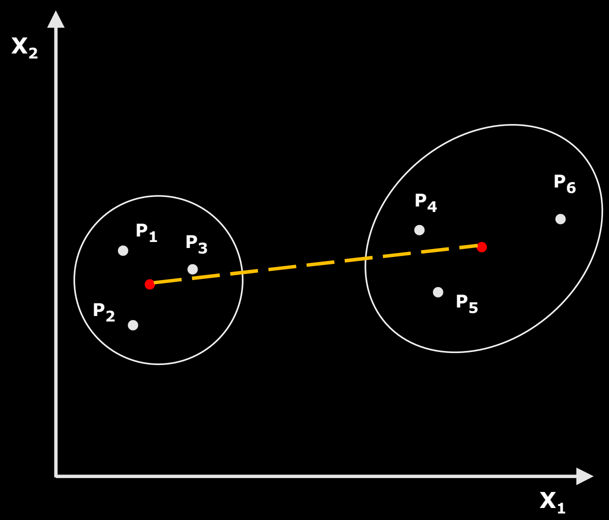




PROXIMITY MATRIX

- MIN
- MAX
- **Group Average**
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's method uses squared error

HOW TO DEFINE INTER-CLUSTER PROXIMITY?



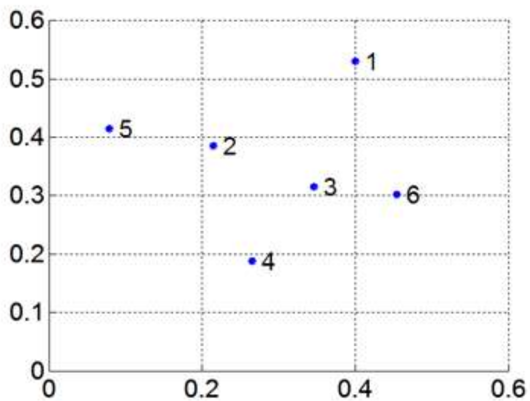
	C1	C2	C3	C4
C1				
C2				
C3				
C4				

PROXIMITY MATRIX

- MIN
- MAX
- Group Average
- **Distance Between Centroids**
- Other methods driven by an objective function
  - Ward's method uses squared error

MIN OR SINGLE LINKAGE

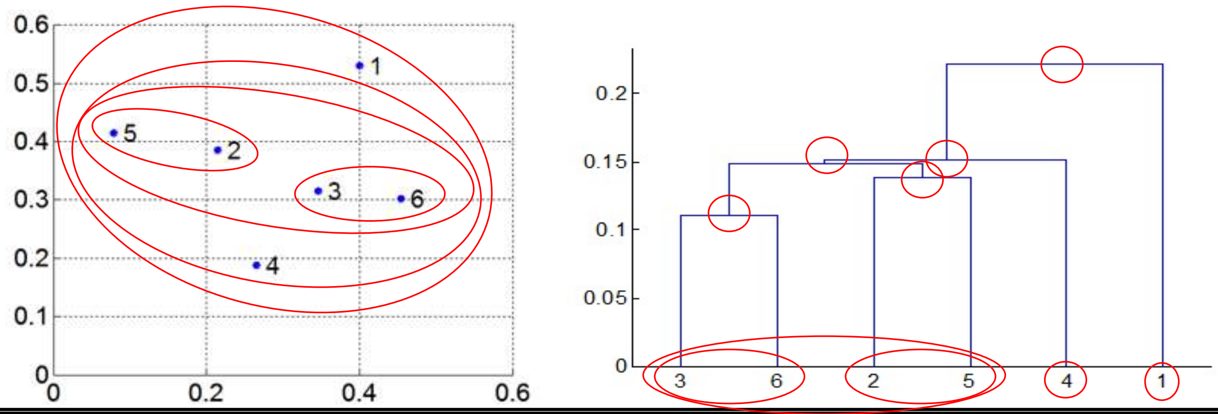
- Proximity of two clusters is based on the two closest points in the different clusters
  - determined by one pair of points, i.e., by one link in the proximity graph



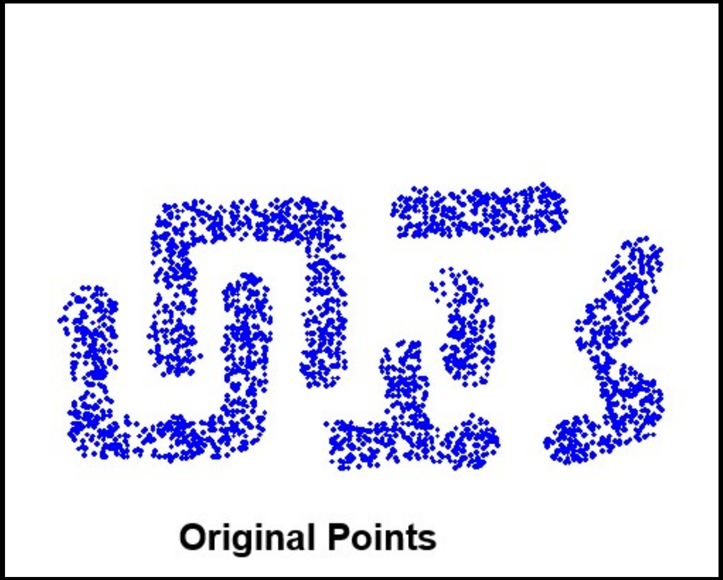
Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

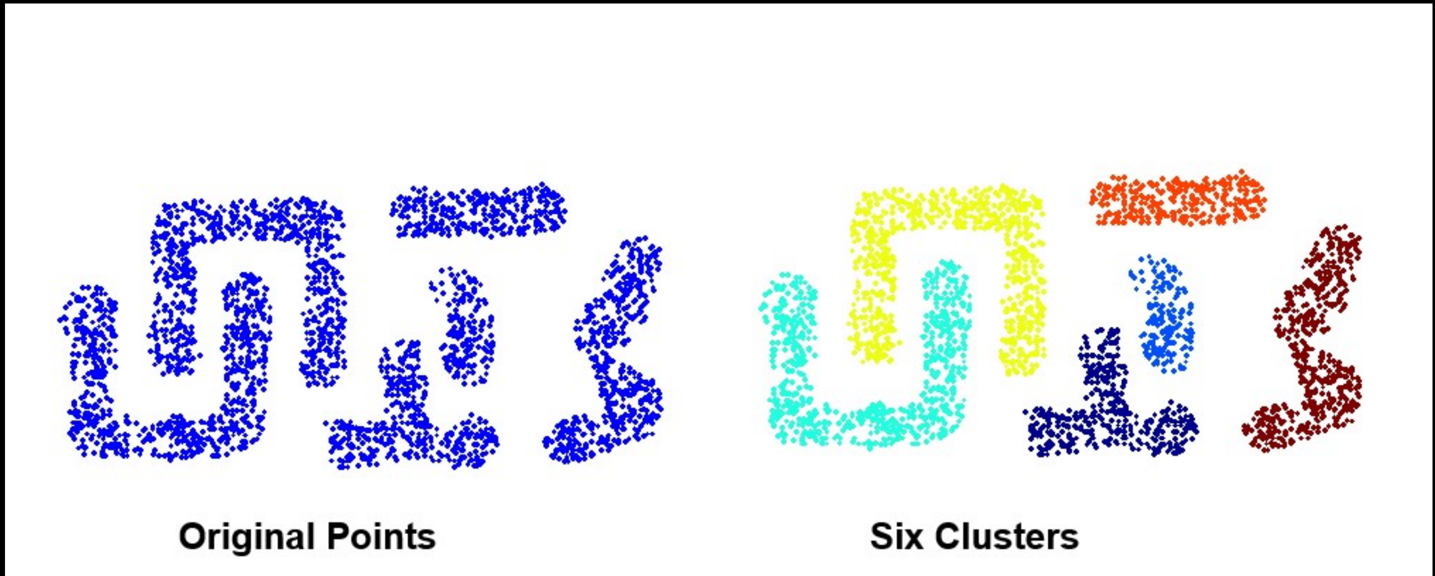
MIN OR SINGLE LINKAGE



MIN OR SINGLE LINKAGE (STRENGTHS)

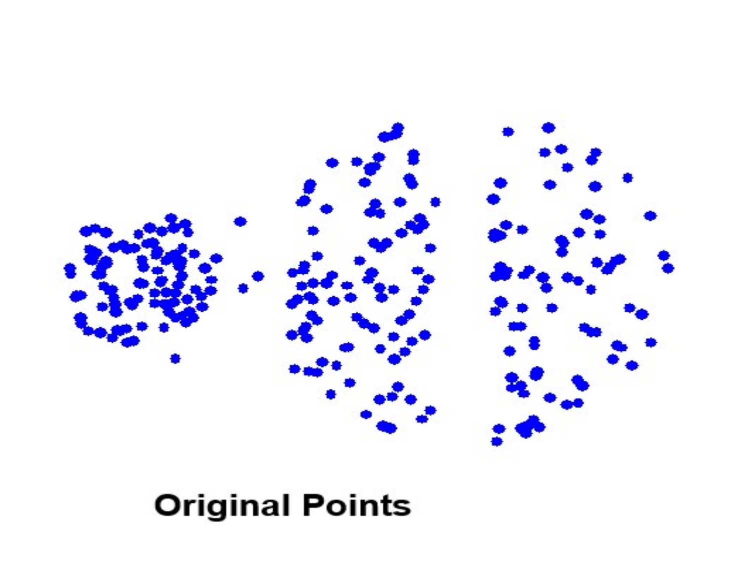


MIN OR SINGLE LINKAGE (STRENGTHS)

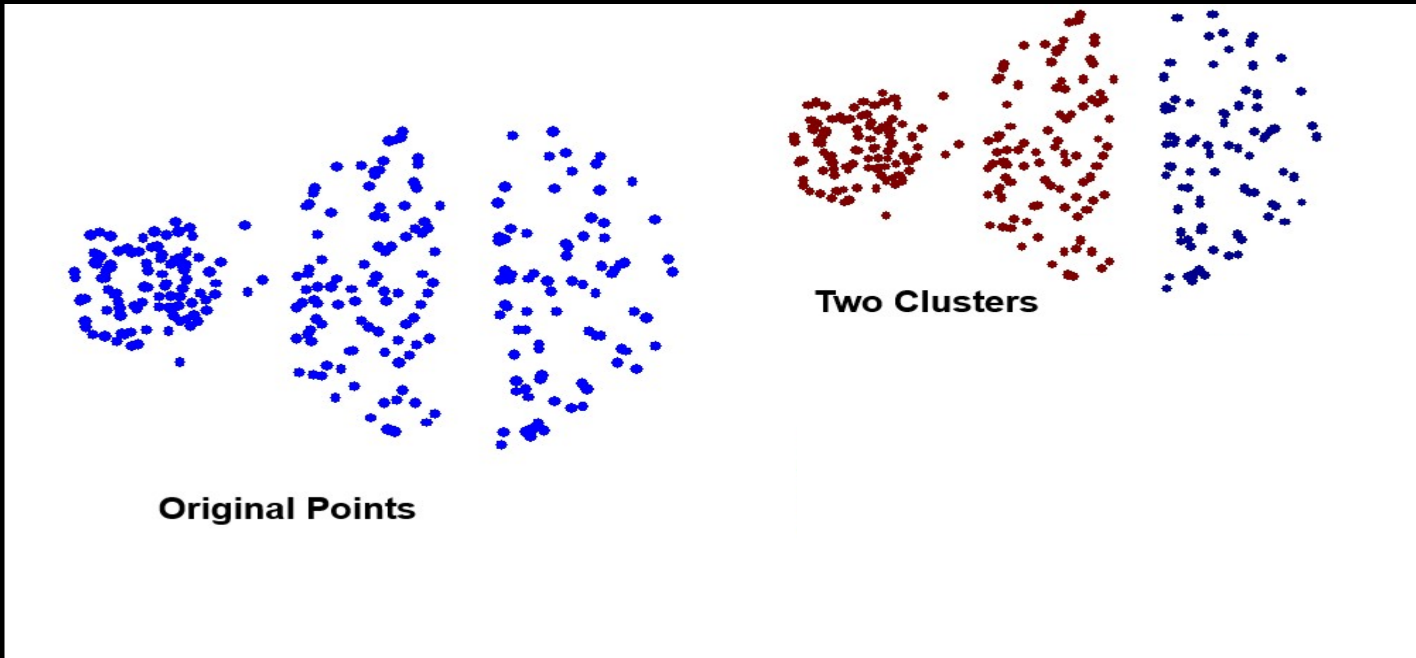


Can handle non-elliptical shapes

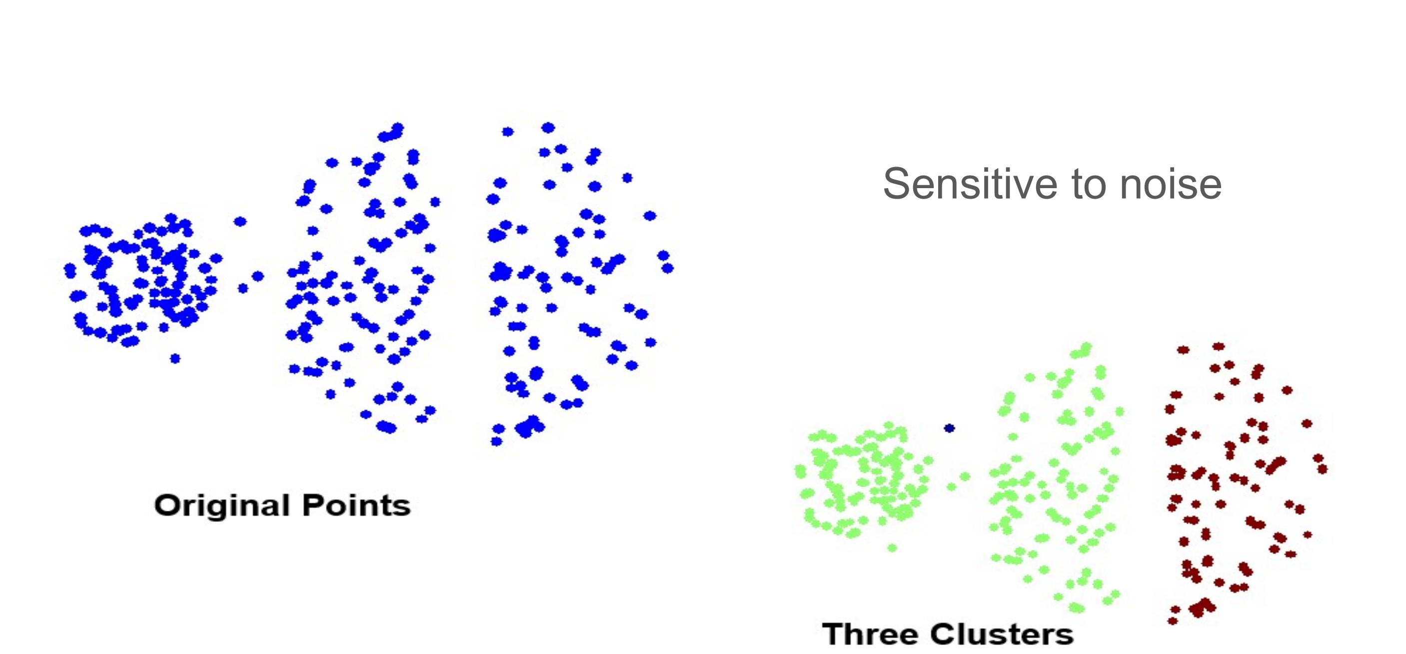
MIN OR SINGLE LINKAGE (LIMITATIONS)



MIN OR SINGLE LINKAGE (LIMITATIONS)

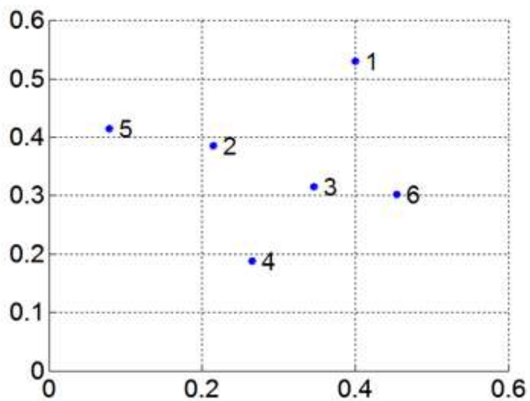


MIN OR SINGLE LINKAGE (LIMITATIONS)



MAX OR COMPLETE LINKAGE

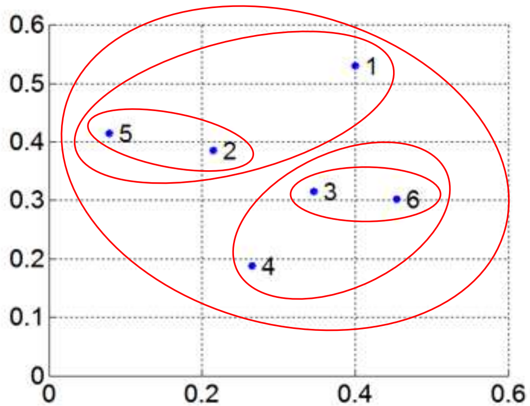
- Proximity of two clusters is based on the two most distant points in the different clusters
  - determined by all pairs of points in the two clusters

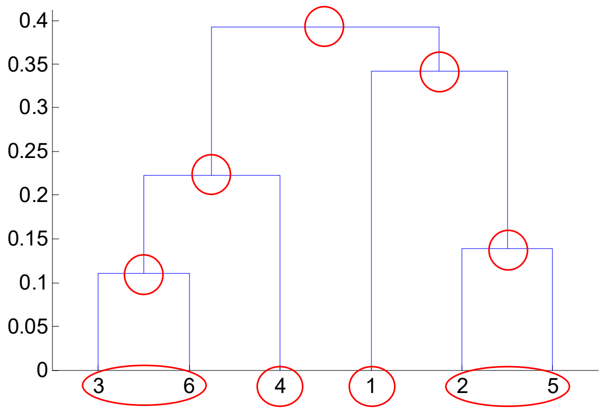


Distance Matrix:

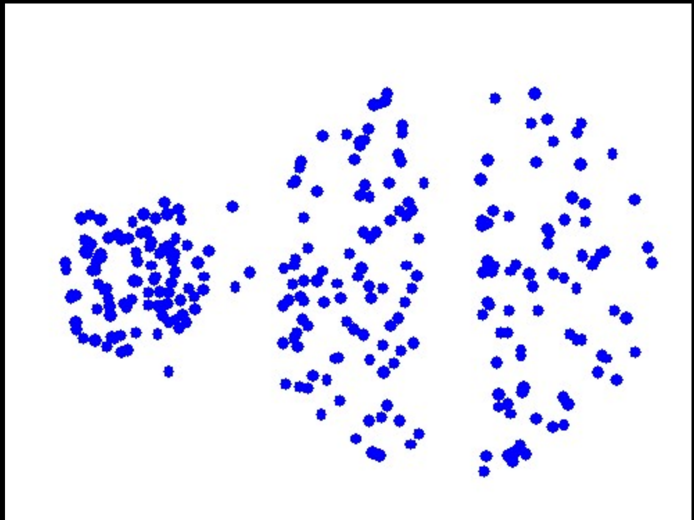
	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

MAX OR COMPLETE LINKAGE

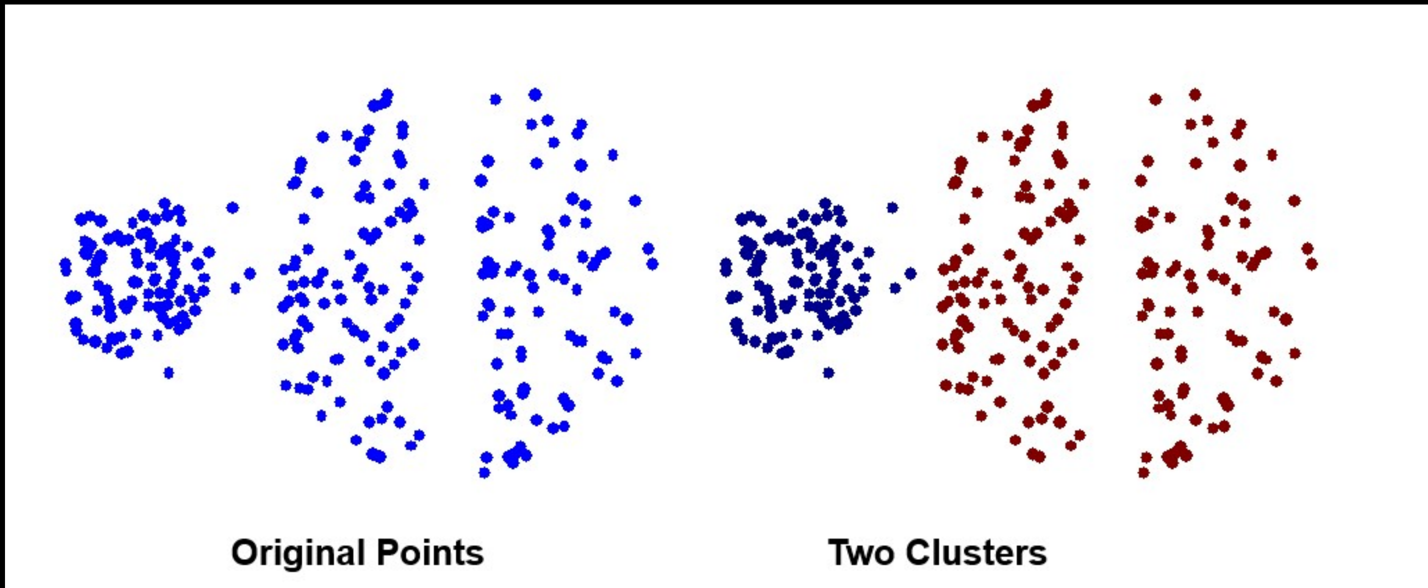




MAX OR COMPLETE LINKAGE (STRENGHTS)

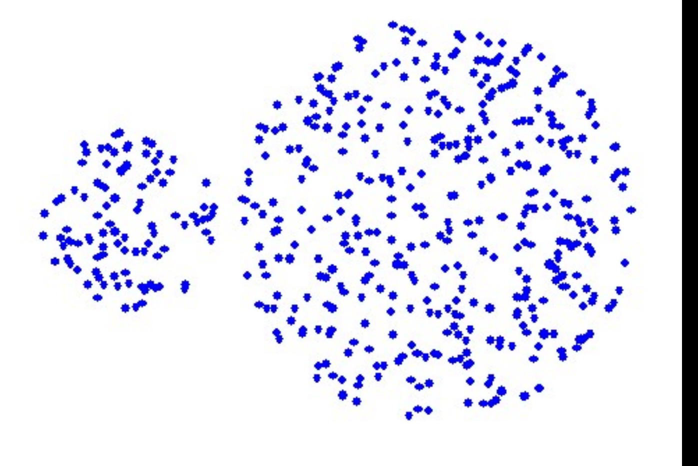

Original Points

MAX OR COMPLETE LINKAGE (STRENGHTS)



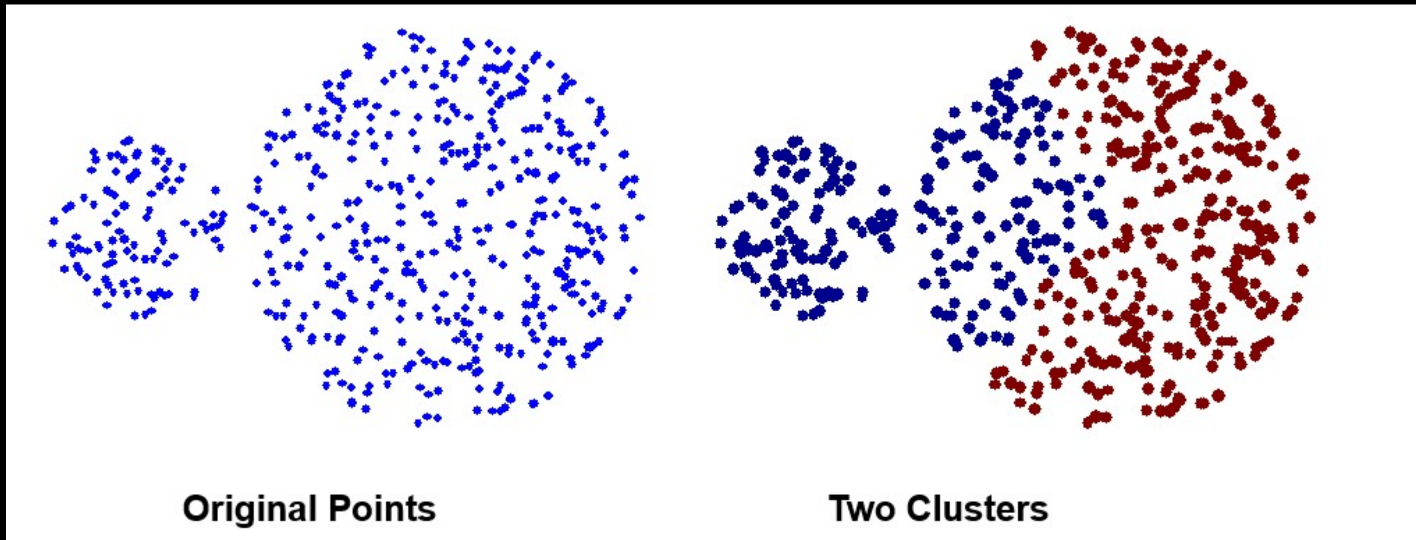
Less susceptible to noise

MAX OR COMPLETE LINKAGE (LIMITATIONS)



Original Points

MAX OR COMPLETE LINKAGE (LIMITATIONS)

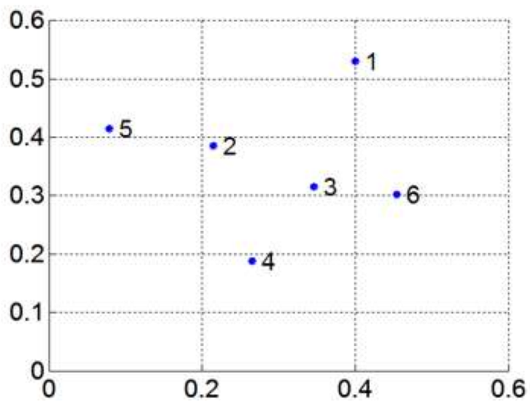


- Tends to break large clusters
- Biased towards globular clusters

GROUP AVERAGE

- Proximity of two clusters is the average of pairwise proximity between points in the two clusters.

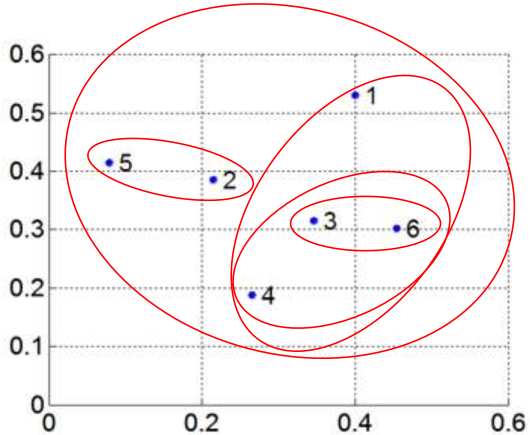
$$\text{proximity}(C_i, C_j) = \frac{\sum_{p_k \in C_i, p_m \in C_j} \text{proximity}(p_k, p_m)}{|C_i||C_j|}$$

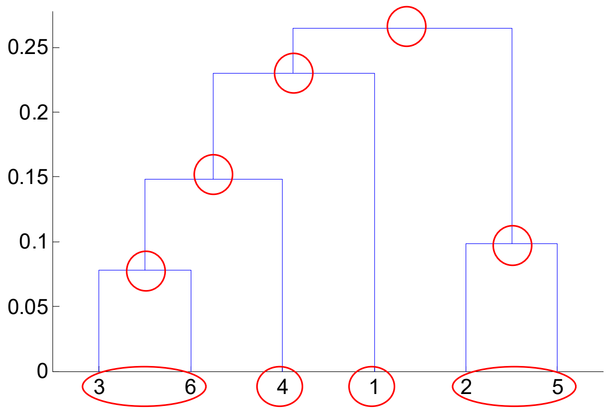


Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

GROUP AVERAGE





GROUP AVERAGE

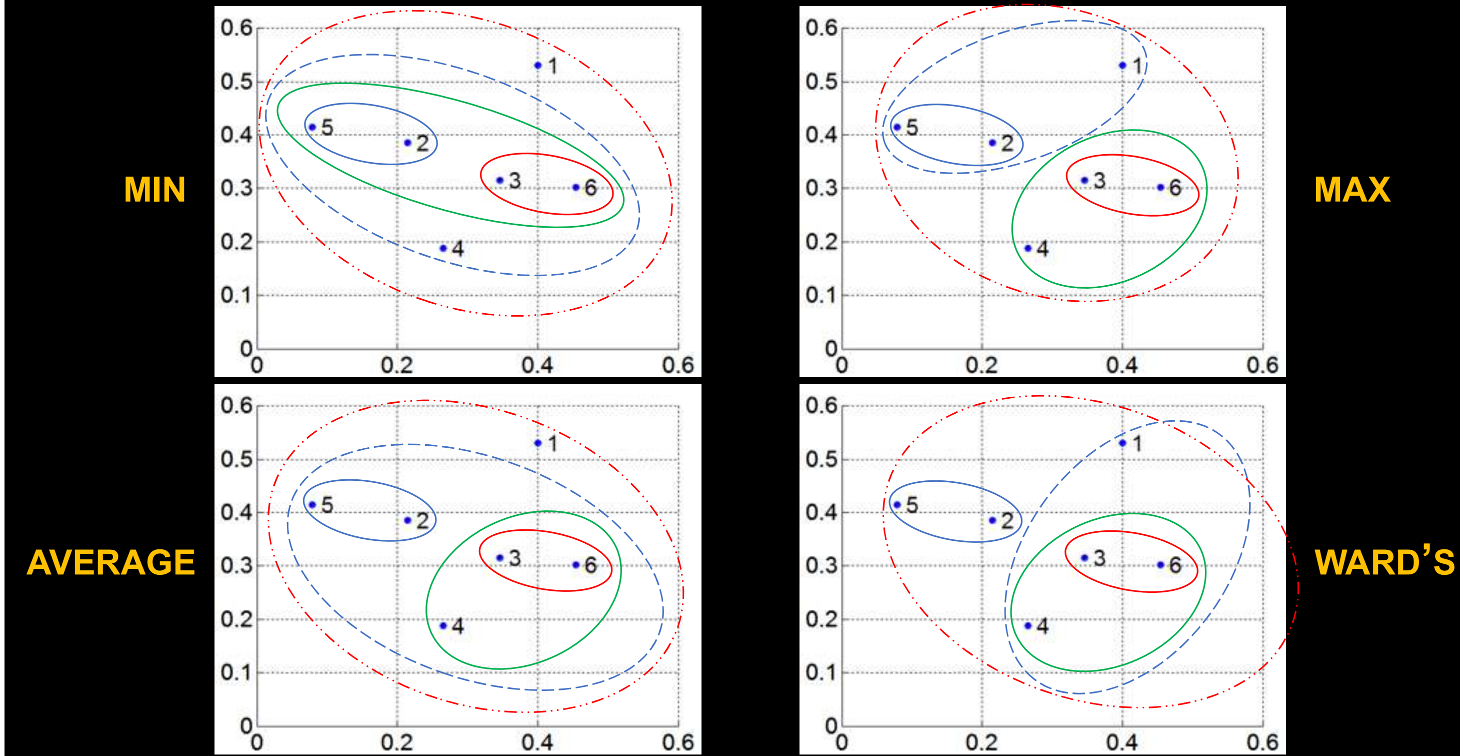
- Compromise between single and complete link

- Strengths
  - less susceptible to noise

- Limitations
  - biased towards globular clusters

WARD'S METHOD

- Similarity of two clusters is based on the increase in squared error when two clusters are merged
  - similar to group average if distance between points is distance squared
- Less susceptible to noise
- Biased towards globular clusters
- Hierarchical analogue of K-means
  - can be used to initialize K-means


MIN
MAX
AVERAGE
WARD'S
0.6
0.5
0.4
0.3
0.2
0.1
0
0.2
0.4
1
2
3
4
5
6

HIERARCHICAL CLUSTERING: TIME AND SPACE REQUIREMENTS

- $O(N^2)$ space since it uses the proximity matrix.
  - N is the number of points.
- $O(N^3)$ time in many cases
  - there are N steps and at each step the size, N^2 proximity matrix must be updated and searched
  - complexity can be reduced to $O(N^2 \log(N))$ time with some cleverness

HIERARCHICAL CLUSTERING: PROBLEMS AND LIMITATIONS

- Once a decision is made to combine two clusters, it cannot be undone
- No global objective function is directly minimized
- Different schemes have problems with one or more of the following:
  - sensitivity to noise
  - difficulty handling clusters of different sizes and non-globular shapes
  - breaking large clusters

RECAP

- Concept
- Strengths
- Types
  - Agglomerative
    - single linkage
    - complete linkage
    - average linkage
    - Ward’s method
  - Divisive
- Complexity
- Limitations