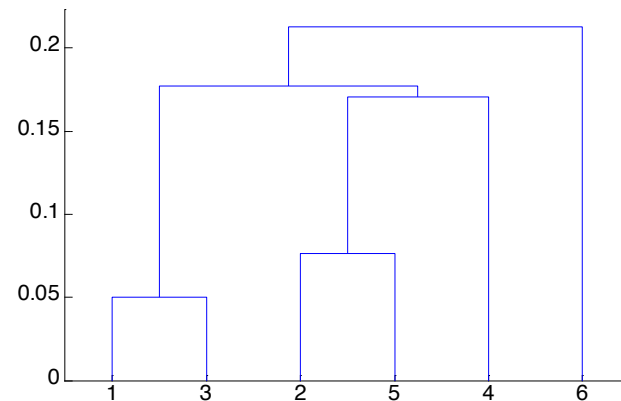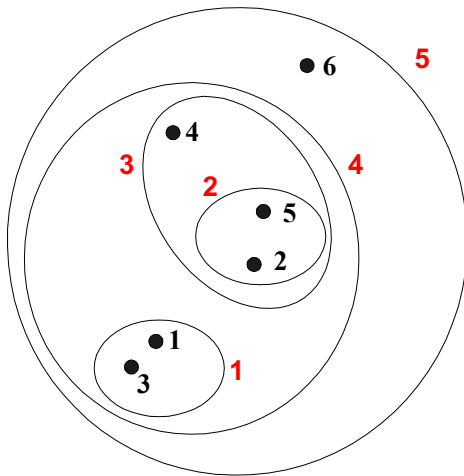# CS722/822: Machine Learning

Instructor: Jiangwen Sun

Computer Science Department

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

Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
  - Any desired number of clusters can be obtained by 'cutting' the dendogram at the proper level

- They may correspond to meaningful taxonomies
  - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, …)
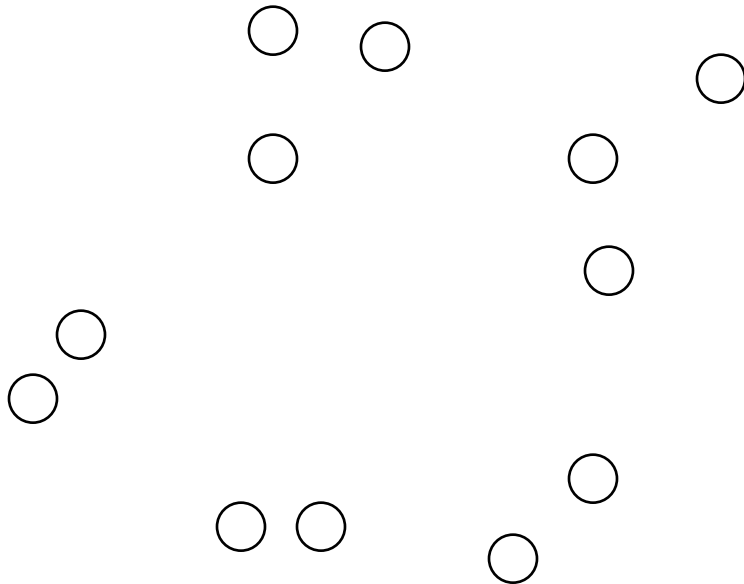
Hierarchical Clustering

- Two main types of hierarchical 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 a point (or there are k clusters)

- Traditional hierarchical algorithms use a similarity or distance matrix
    - Merge two clusters or split one cluster at a time

Agglomerative Clustering Algorithm

- More popular hierarchical clustering technique
- Basic algorithm is straightforward
  1. Let each data point be a cluster
  2. Compute the proximity matrix
  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

Starting Situation

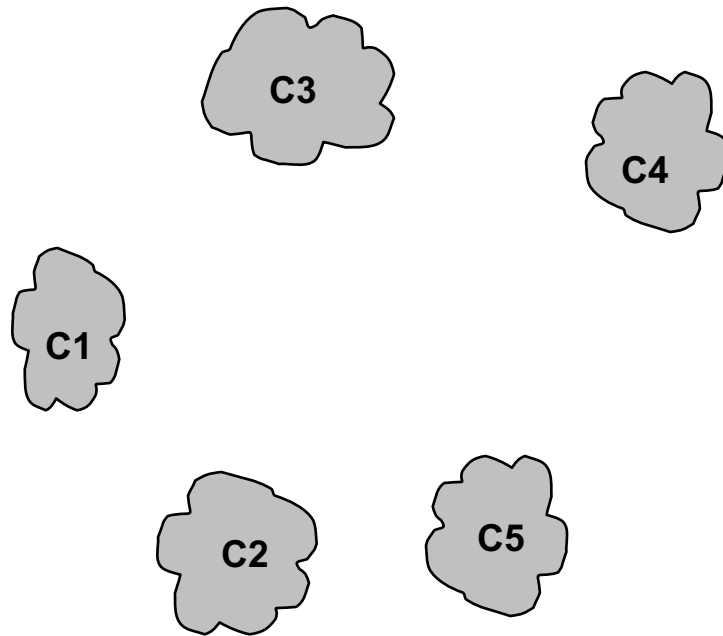- Start with clusters of individual points and a proximity matrix

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

p1   p2   p3   p4   . . .   p9   p10   p11   p12

Intermediate Situation

● After some merging steps, we have some clusters

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

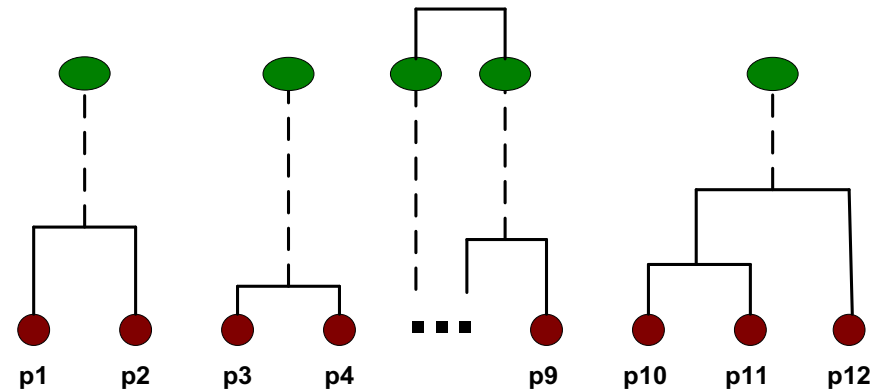**Proximity Matrix**

p1   p2   p3   p4   ...   p9   p10   p11   p12
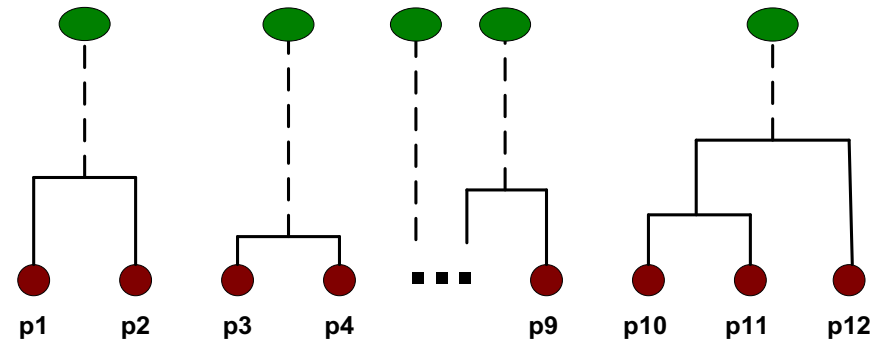
Intermediate Situation

- We want to merge the two closest clusters (C2 and C5) and update the proximity matrix.
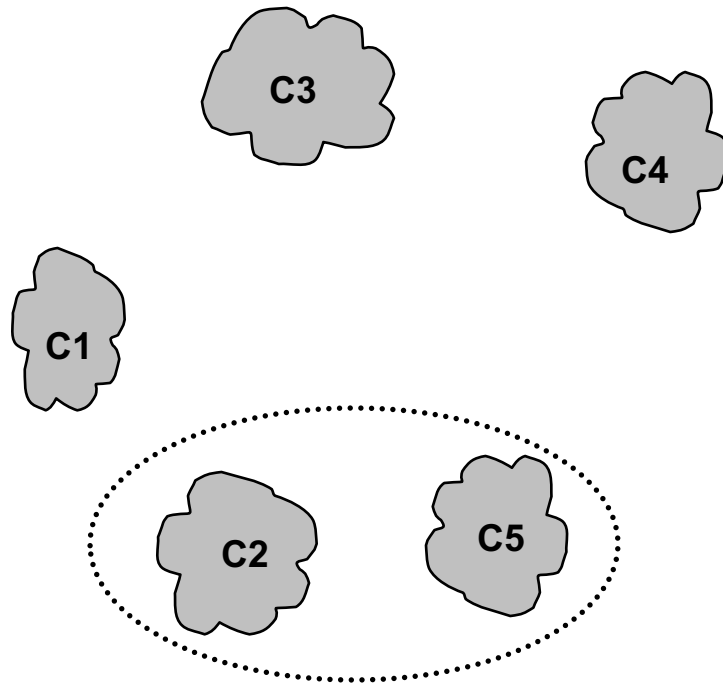
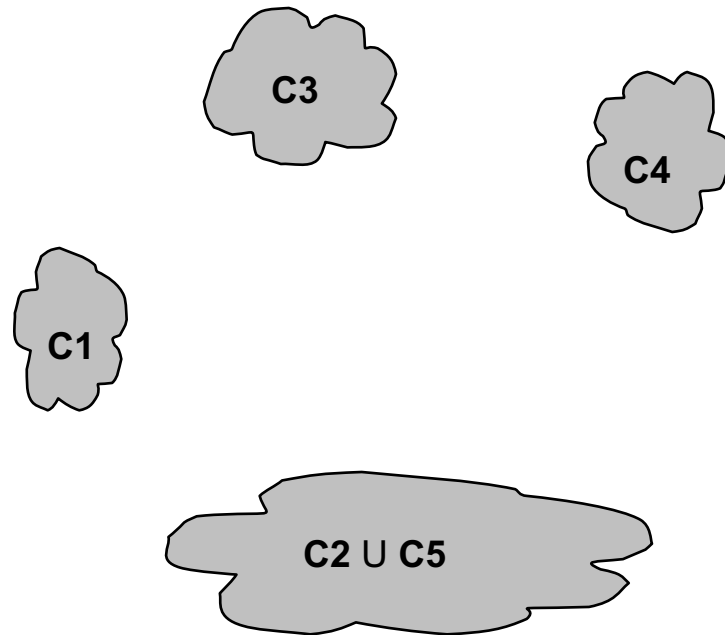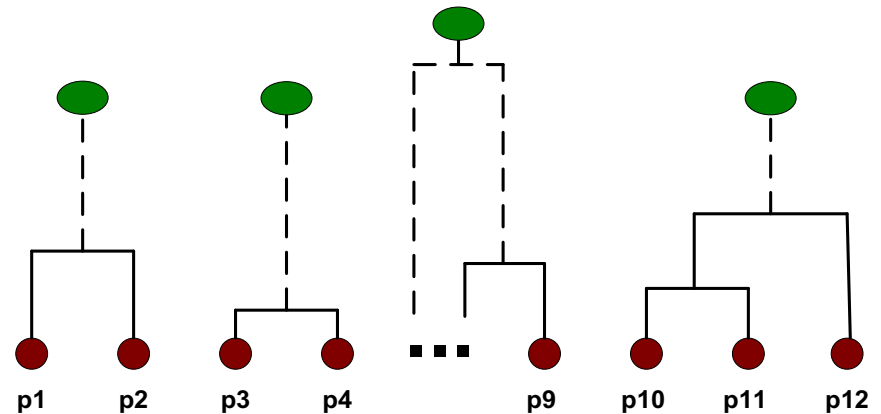|	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix

After Merging

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

	C1	C2 ∪ C5	C3	C4
C1		?		
C2 ∪ C5	?	?	?	?
C3		?		
C4		?		

Proximity Matrix

C3

C4

C1

C2 ∪ C5

p1   p2   p3   p4   ...   p9   p10   p11   p12

How to Define Inter-Cluster Similarity

Similarity?

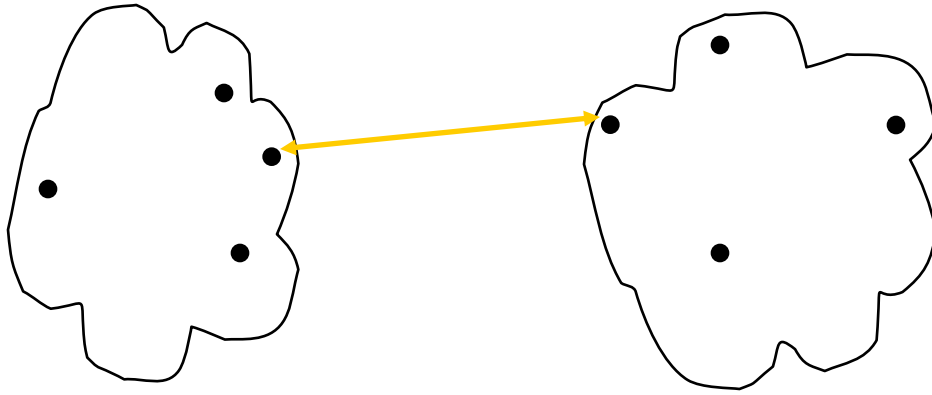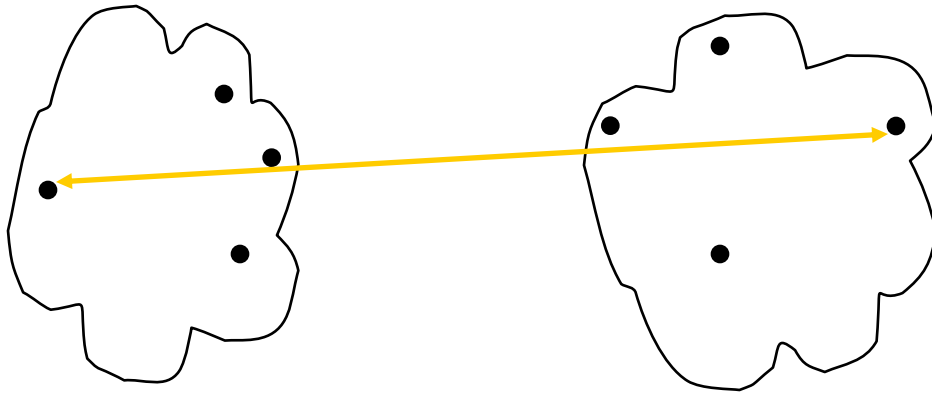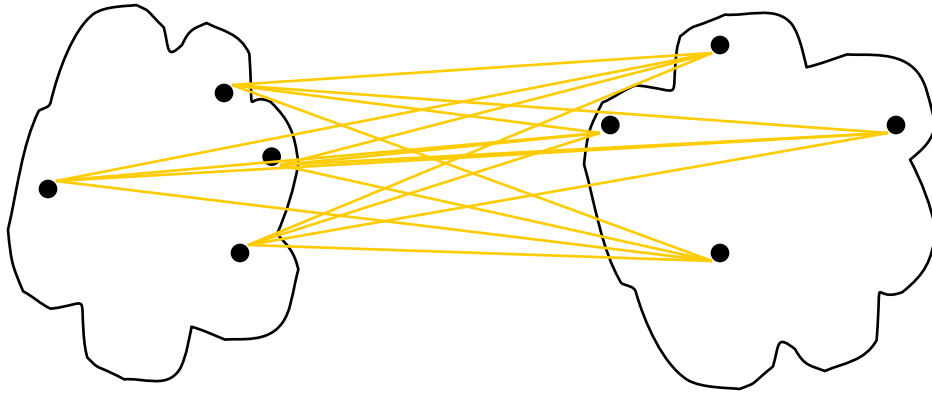|	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

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 Similarity

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

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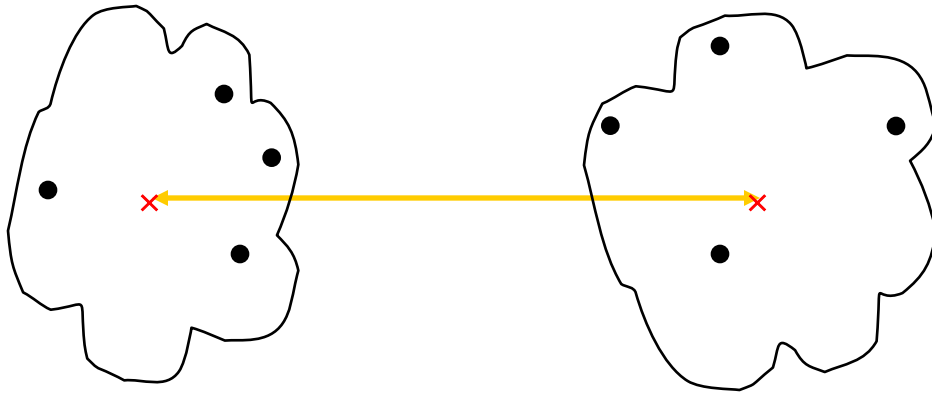- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

How to Define Inter-Cluster Similarity

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

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 Similarity

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

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

13

How to Define Inter-Cluster Similarity

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

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

Cluster Similarity: MIN or Single Link

● Similarity of two clusters is based on the two most similar (closest) points in the different clusters

   – Determined by one pair of points, i.e., by one link in the proximity graph.

	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00
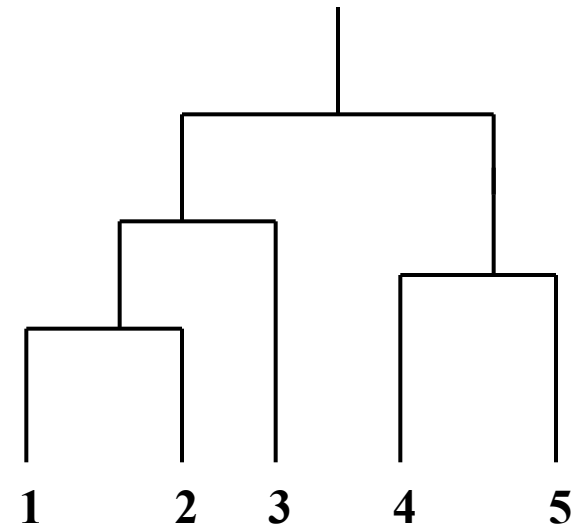
Hierarchical Clustering: MIN
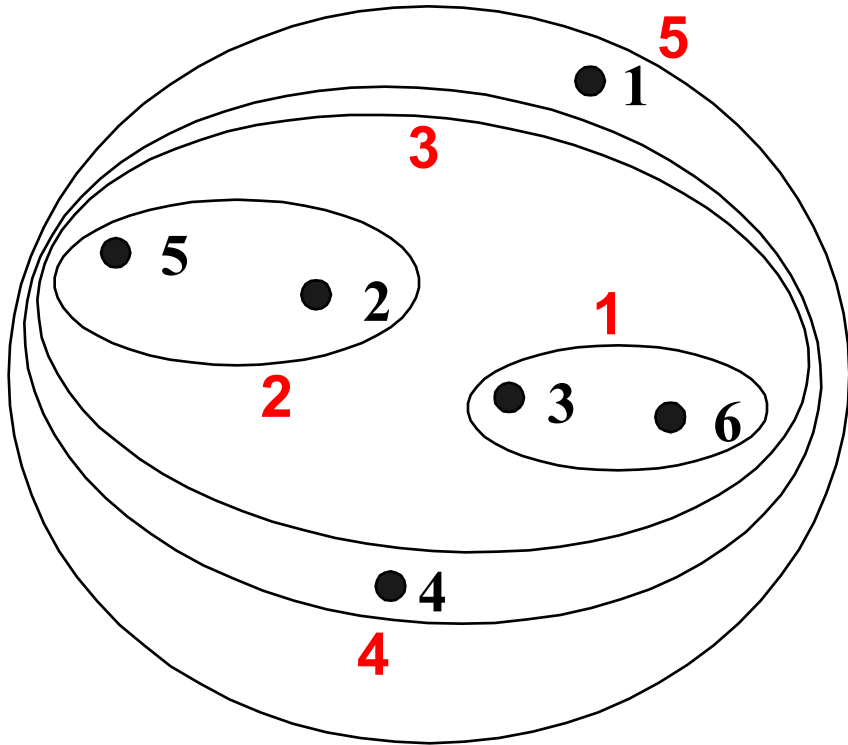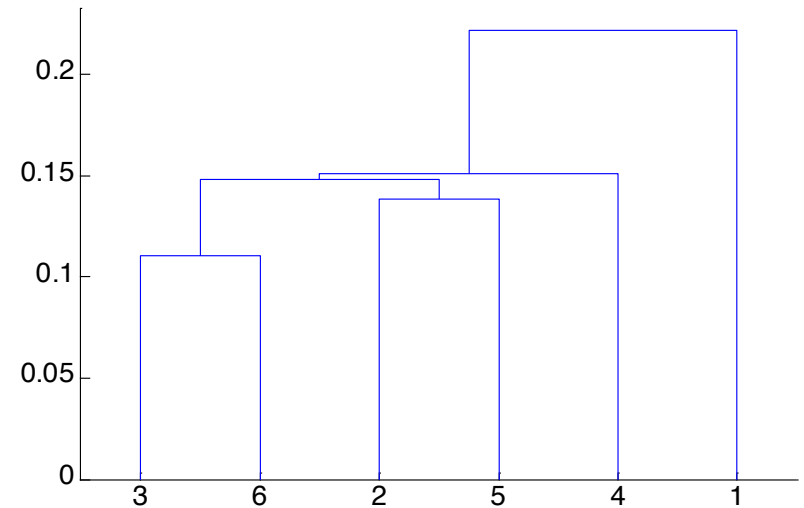
Nested Clusters   Dendrogram

Strength of MIN

Can handle non-elliptical shapes
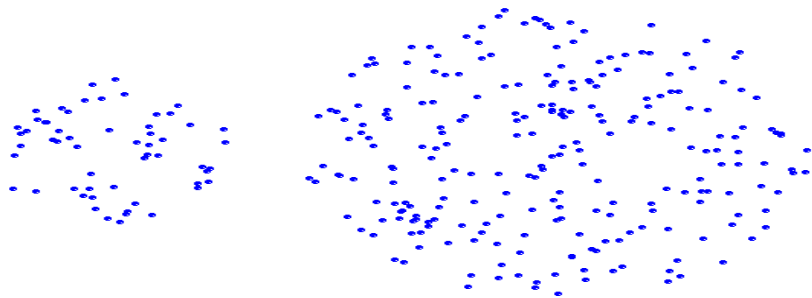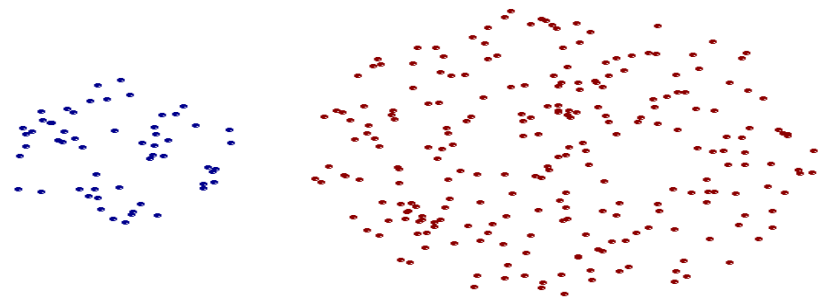
Original Points   Two Clusters

Limitations of MIN
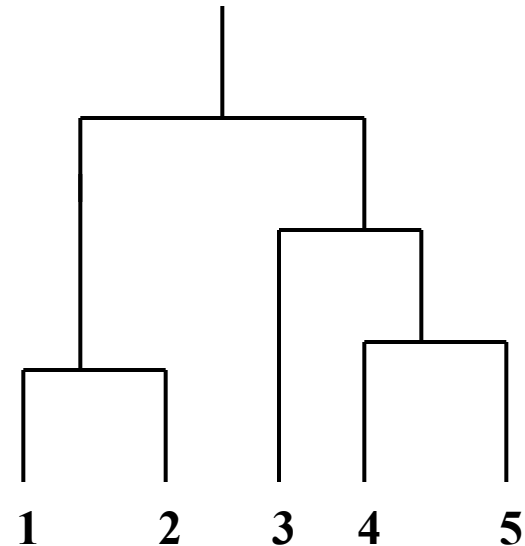
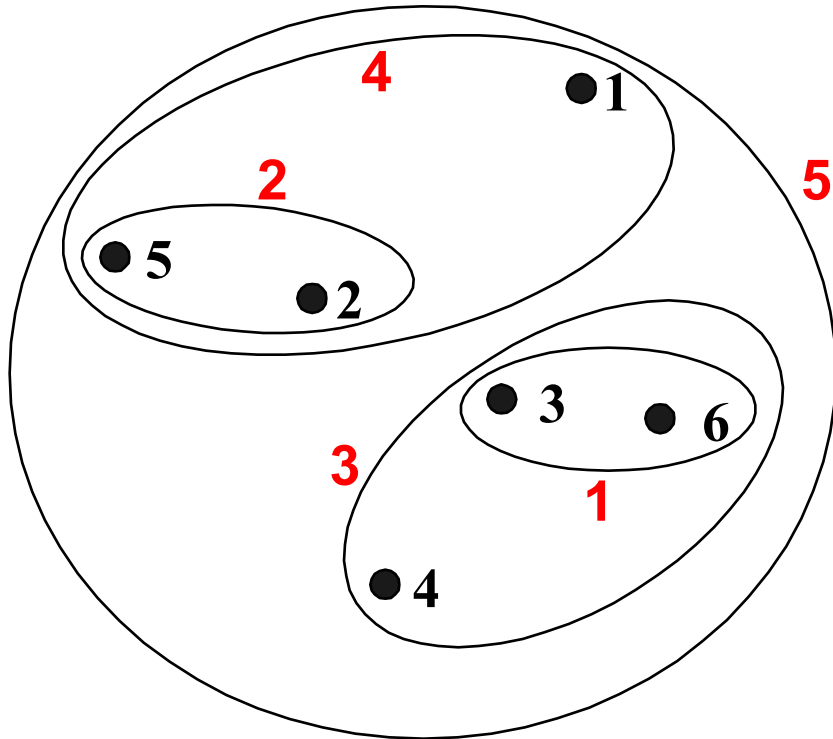Sensitive to noise and outliers
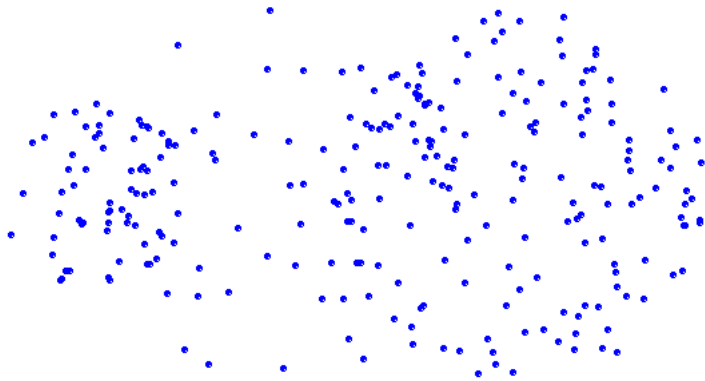
Original Points
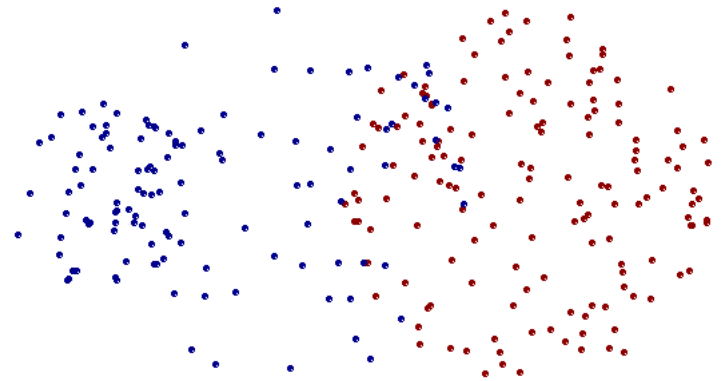
Two Clusters

Cluster Similarity: MAX or Complete Linkage

- ● Similarity of two clusters is based on the two least similar (most distant) points in the different clusters

  - – Determined by all pairs of points in the two clusters

	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00

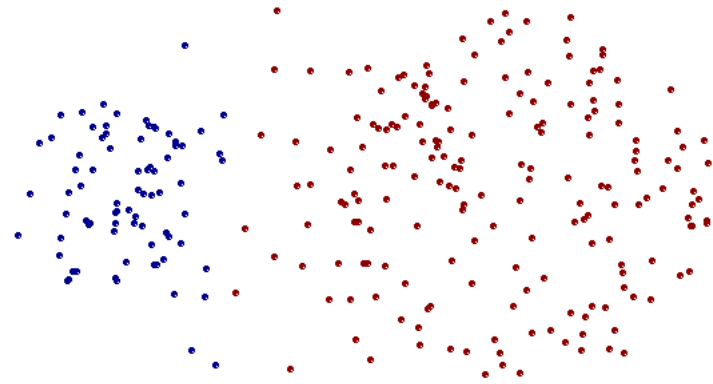# Hierarchical Clustering: MAX

Nested Clusters
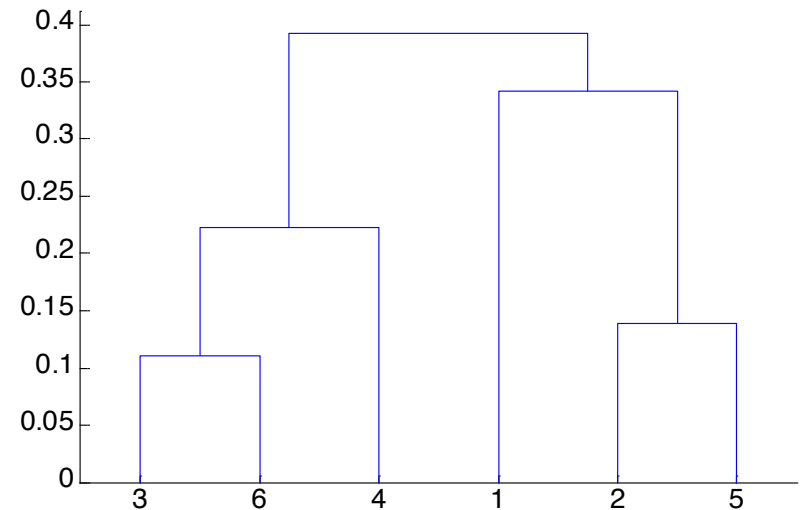
Dendrogram

Strength of MAX

Less susceptible to noise and outliers
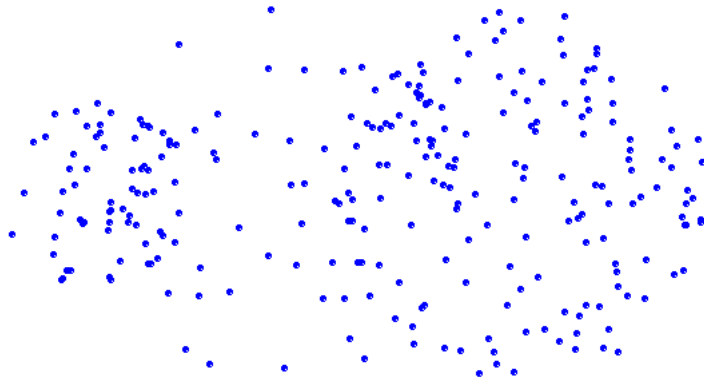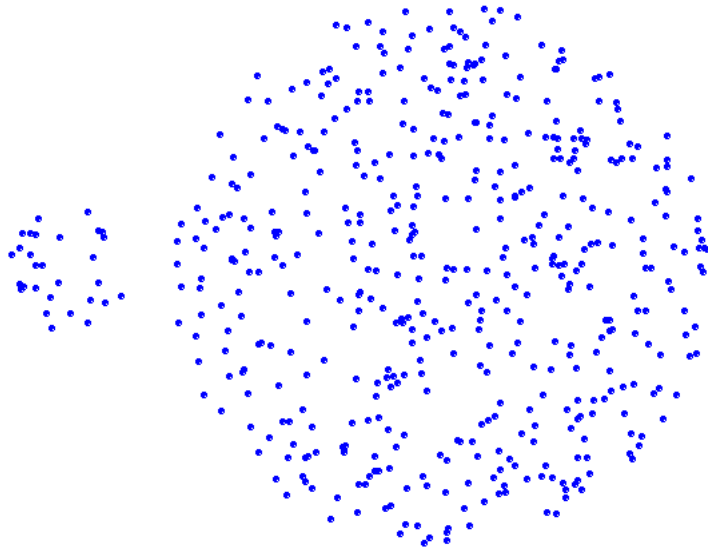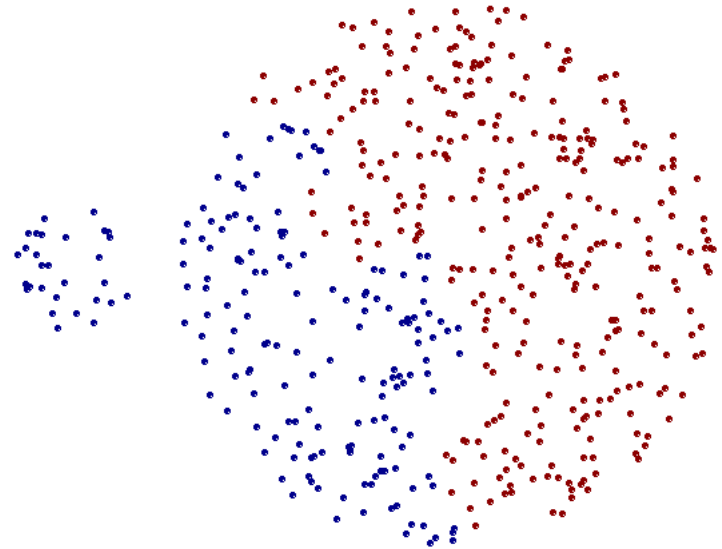
Original Points

Two Clusters

Limitations of MAX

- Tends to break large clusters

Original Points   Two Clusters

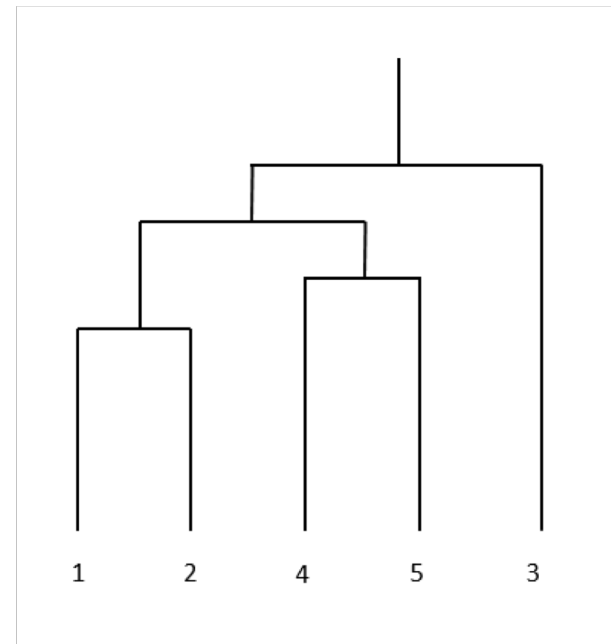- Biased towards globular clusters

Cluster Similarity: Group Average

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

$$\text{proximity}(\text{Cluster}_i, \text{Cluster}_j) = \frac{\displaystyle\sum_{\substack{p_i \in \text{Cluster}_i \\ p_j \in \text{Cluster}_j}} \text{proximity}(p_i, p_j)}{|\text{Cluster}_i| * |\text{Cluster}_j|}$$

- Need to use **average** connectivity for scalability since total proximity favors large clusters
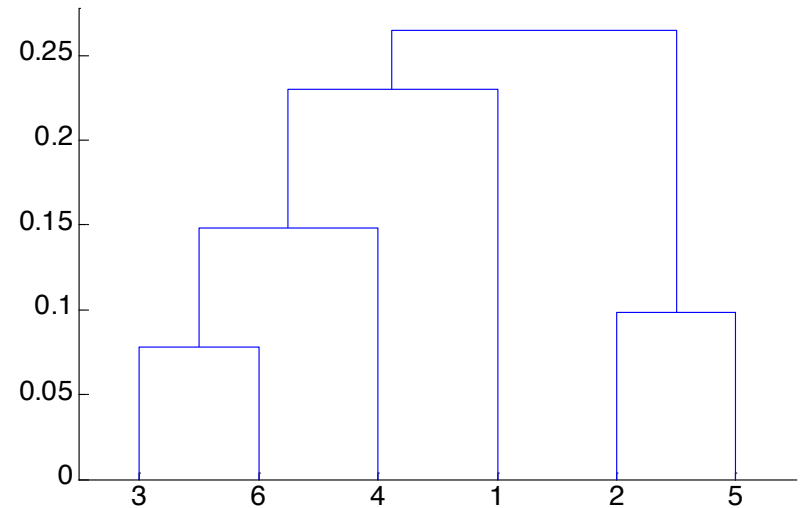
	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00

23

Hierarchical Clustering: Group Average
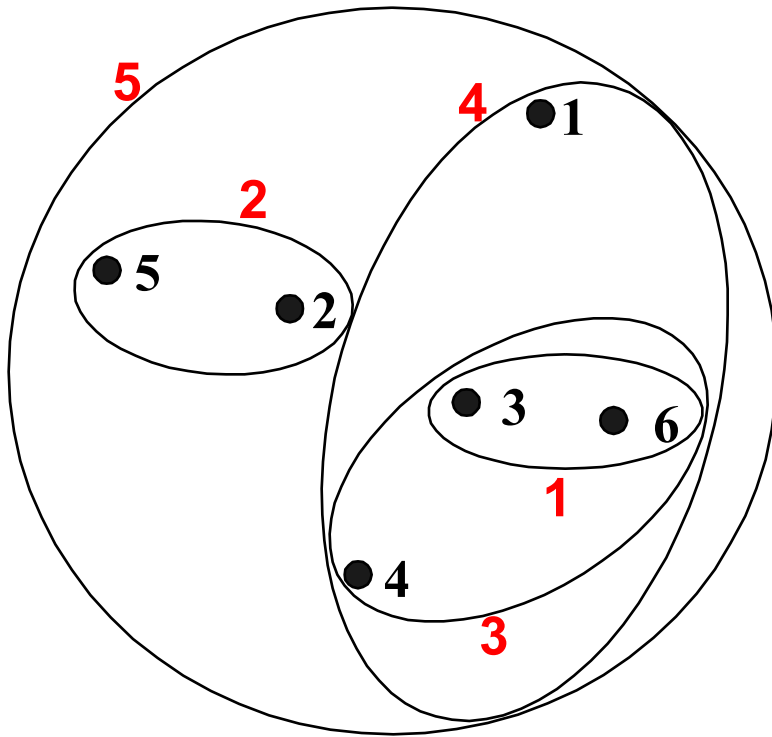

Nested Clusters

Dendrogram

Hierarchical Clustering: Group Average

- Compromise between Single and Complete Link

- Strengths
  - Less susceptible to noise and outliers

- Limitations
  - Biased towards globular clusters

Cluster Similarity: 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 and outliers

- Biased towards globular clusters

- Hierarchical analogue of K-means
  - Can be used to initialize K-means

Hierarchical Clustering: Comparison

MIN

MAX

Group average

Ward's Method
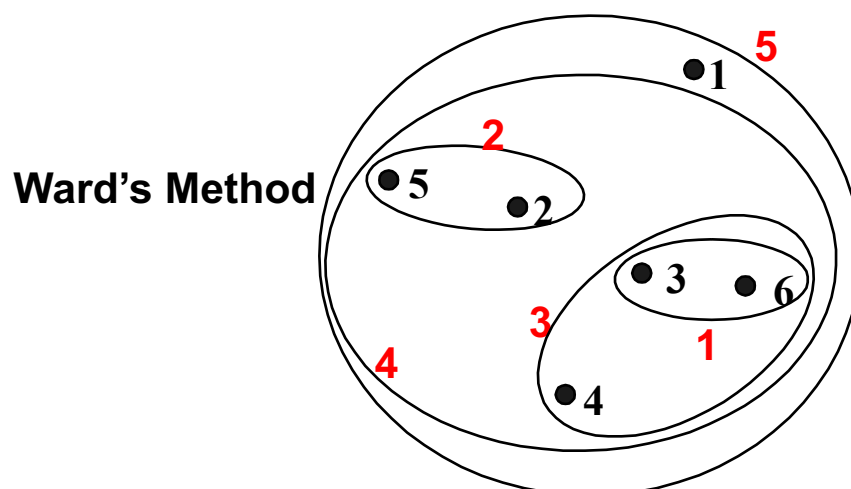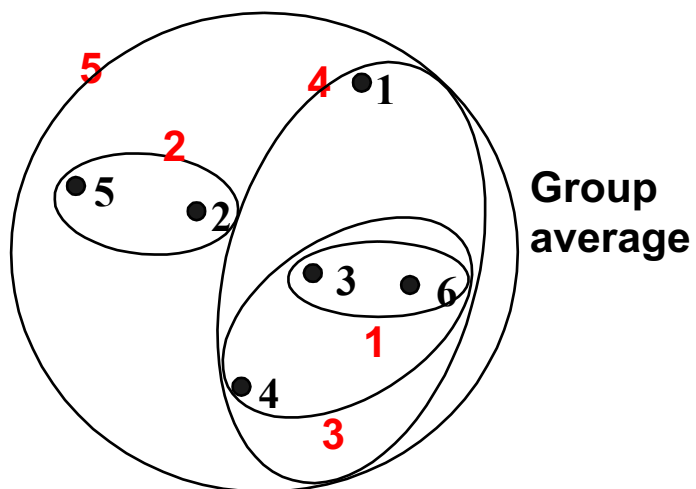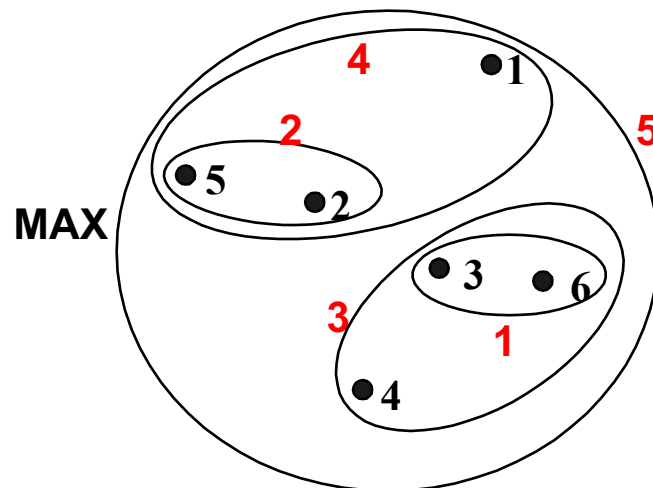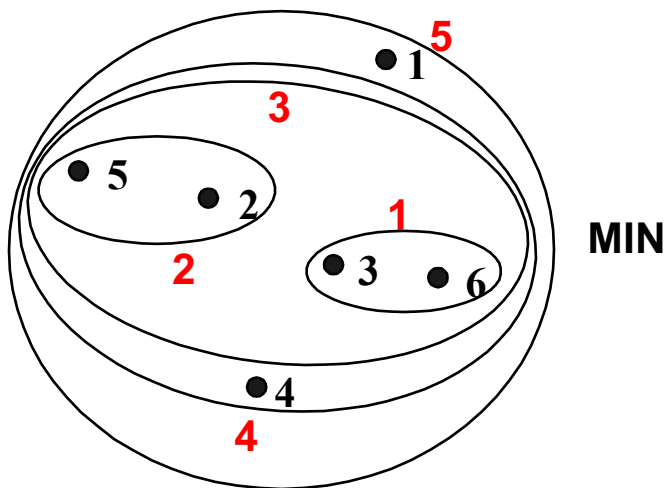
Hierarchical Clustering: Problems and Limitations

- Once a decision is made to combine two clusters, it cannot be undone

- No objective function is directly minimized

- Different schemes have problems with one or more of the following:
  - Sensitivity to noise and outliers
  - Difficulty handling different sized clusters
  - Biased towards globular clusters
  - Breaking large clusters

MST: Divisive Hierarchical Clustering

● Build MST (Minimum Spanning Tree)

- – Start with a tree that consists of any point

- – In successive steps, look for the closest pair of points (p, q) such that one point (p) is in the current tree but the other (q) is not
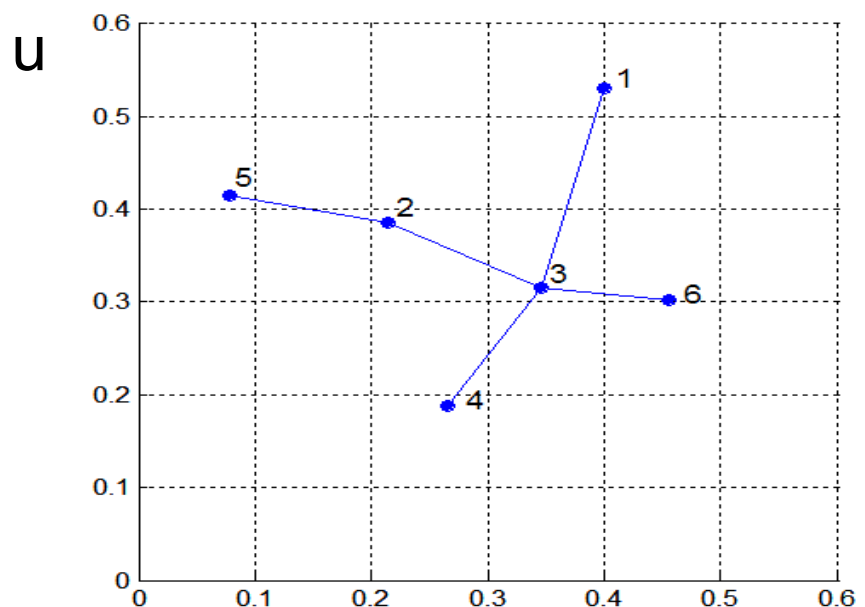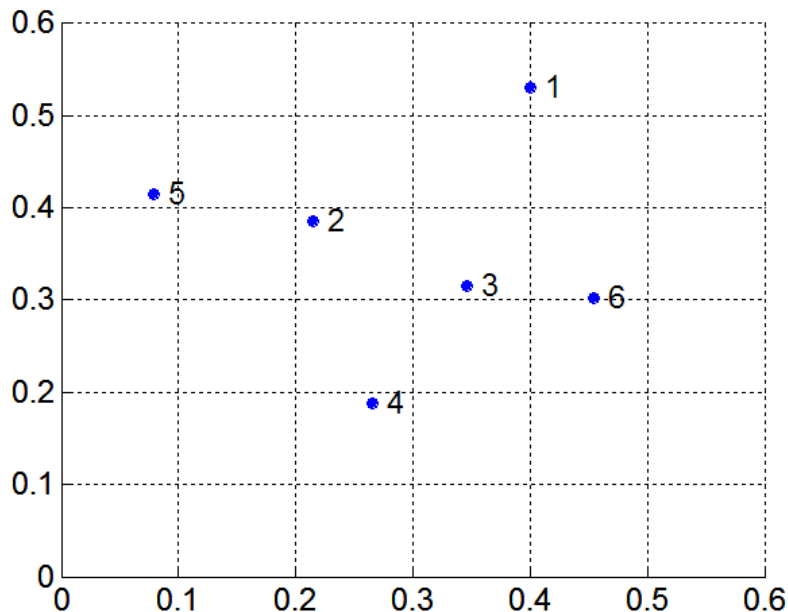
MST: Divisive Hierarchical Clustering

● Use MST for constructing hierarchy of clusters

Algorithm 7.5 MST Divisive Hierarchical Clustering Algorithm

1: Compute a minimum spanning tree for the proximity graph.

2: **repeat**

3:     Create a new cluster by breaking the link corresponding to the largest distance (smallest similarity).

4: **until** Only singleton clusters remain

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 for some approaches

Summary

- Discussed hierarchical clustering, particularly the bottom-up approaches

- Different ways to calculate intra-cluster distance/similarity for hierarchical clustering