

دانشگاه آزاد اسلامی واحد تبریز

نام درس: داده کاوی

بخش: خوشه بندی

نام استاد: دکتر مسعود کارگر

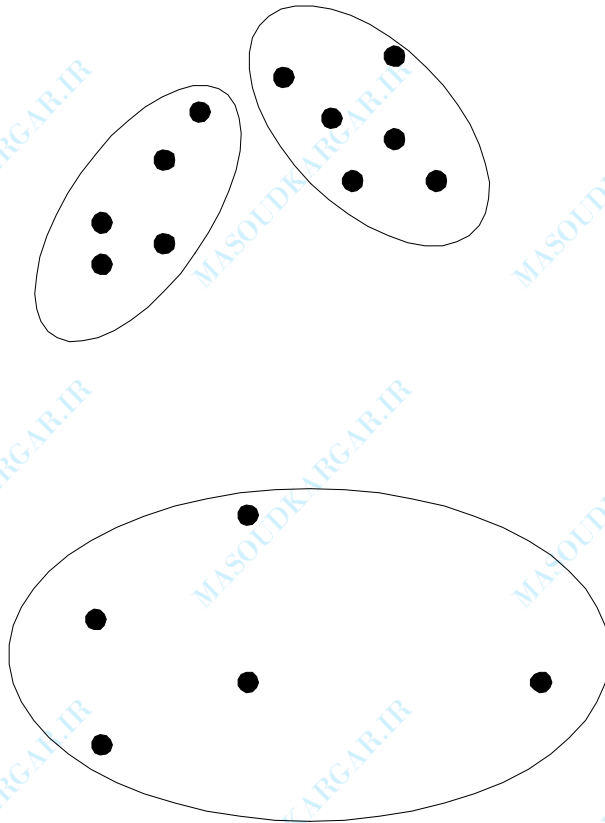


Roadmap

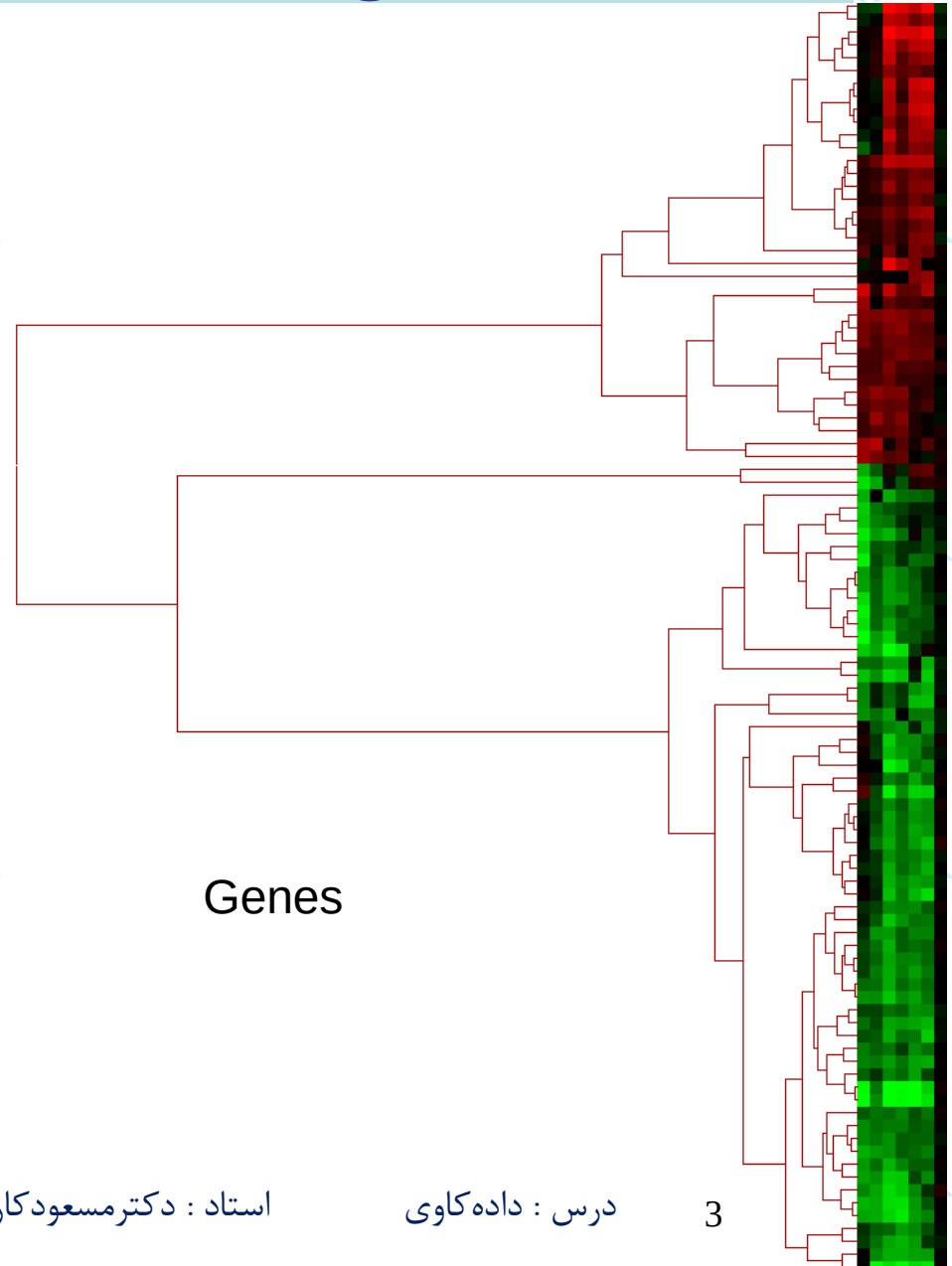
- Hierarchical Clustering vs. Partitional Clustering
- Agglomerative clustering algorithms
- Comparison of hierarchical clustering algorithms
- Divisive clustering algorithms
- Density-based clustering

Partitional Clustering v.s. Hierarchical Clustering

conditions

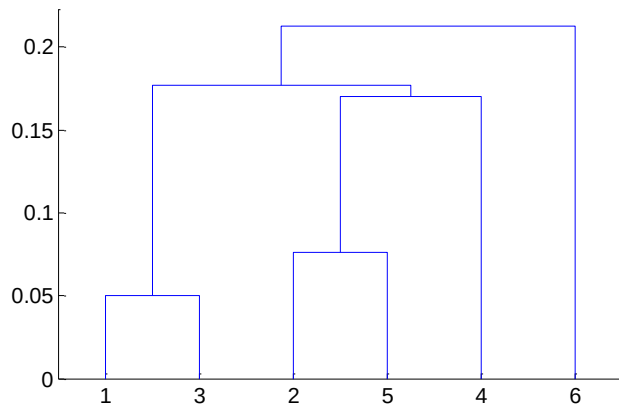


A Partitional Clustering



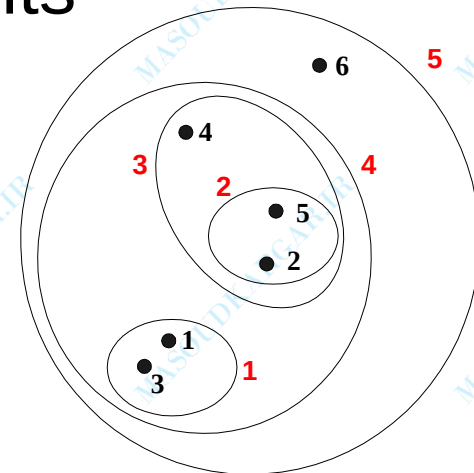
Hierarchical Clustering

- 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



دانشگاه آزاد اسلامی واحد تبریز

استاد : دکتر مسعود کارگر



درس : داده کاوی

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 dendrogram at the proper level
- They may correspond to meaningful taxonomies
 - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, ...)
- Can view data at many levels of granularity, all at the same time
- Simple heuristics for constructing hierarchical clusterings

Hierarchical Clustering

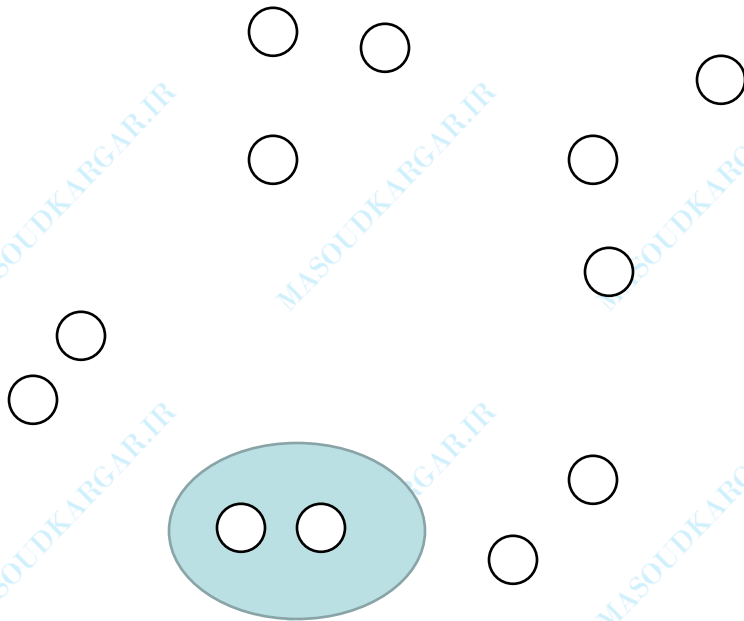
- Two main types of hierarchical clustering
 - Agglomerative:
 - Start with the points as individual clusters
 - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
 - Divisive:
 - Start with one, all-inclusive cluster
 - At each step, split a cluster until each cluster contains a point (or there are k clusters)
- Traditional hierarchical algorithms use a similarity or distance matrix
 - Merge or split one cluster at a time

Agglomerative Clustering Algorithm

- More popular hierarchical clustering technique
- Basic algorithm is straightforward
 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

Starting Situation

- Start with clusters of individual points and a proximity matrix



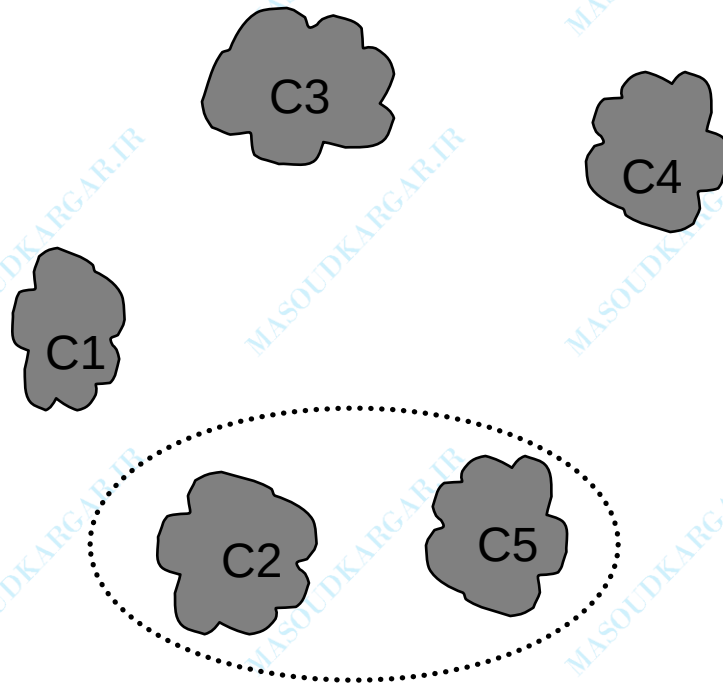
	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
...						

Proximity Matrix



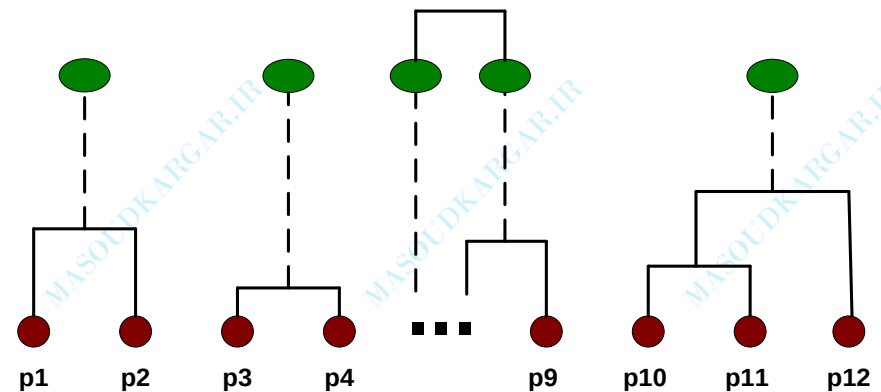
Intermediate Situation

- We want to merge the two closest clusters C_2 and C_5 and update the proximity matrix.



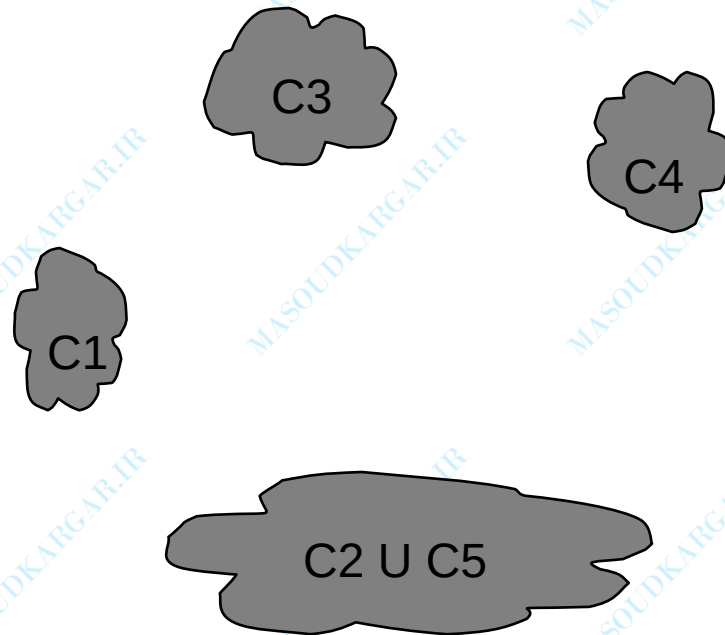
	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix



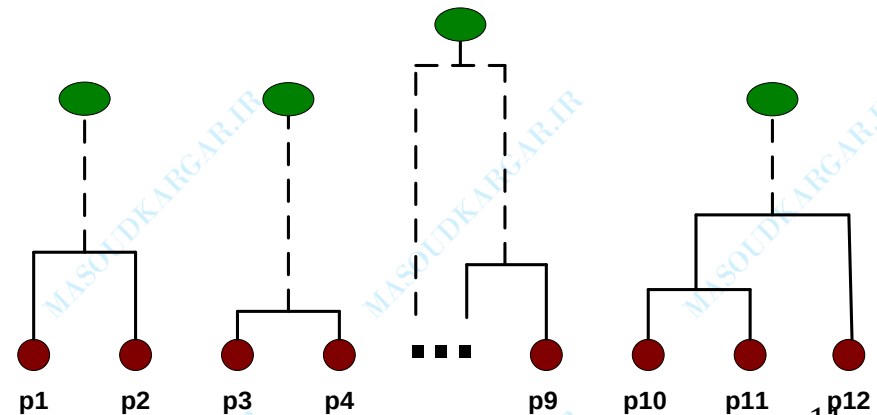
After Merging

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

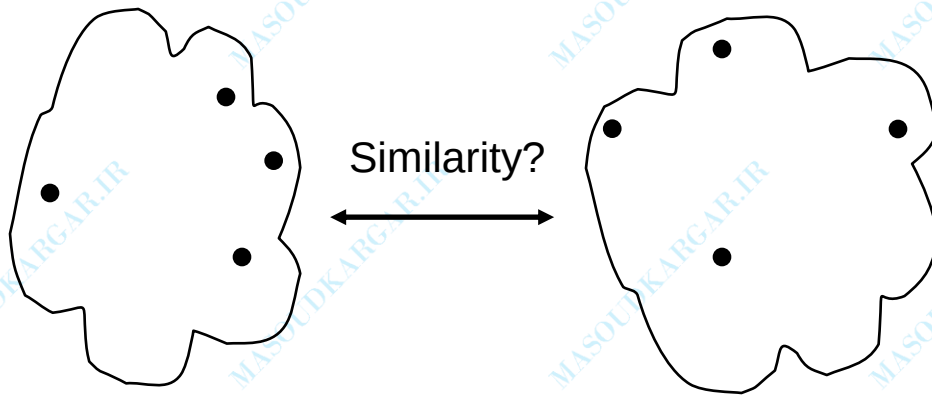


	C1	C2 U C5	C3	C4
C1		?		
C2 U C5	?	?	?	?
C3		?		
C4		?		

Proximity Matrix



How to Define Inter-Cluster Similarity

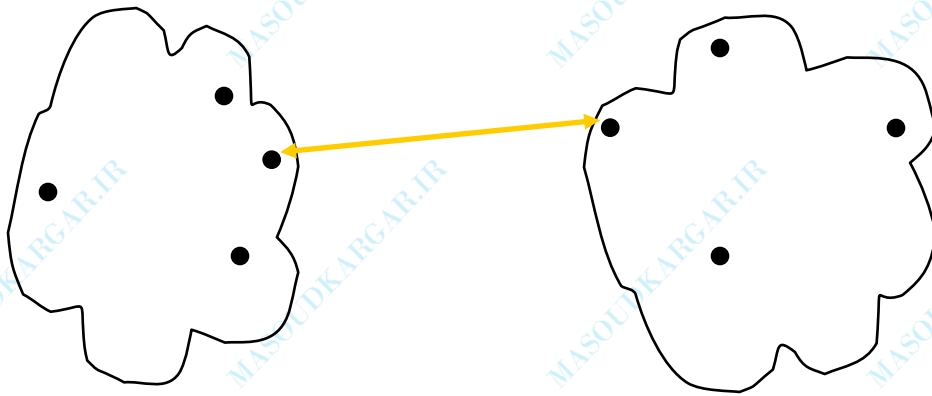


- ❑ MIN
- ❑ MAX
- ❑ Group Average
- ❑ Distance Between Centroids
- ❑ Other methods driven by an objective function
 - Ward's Method uses squared error

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

Proximity Matrix

How to Define Inter-Cluster Similarity

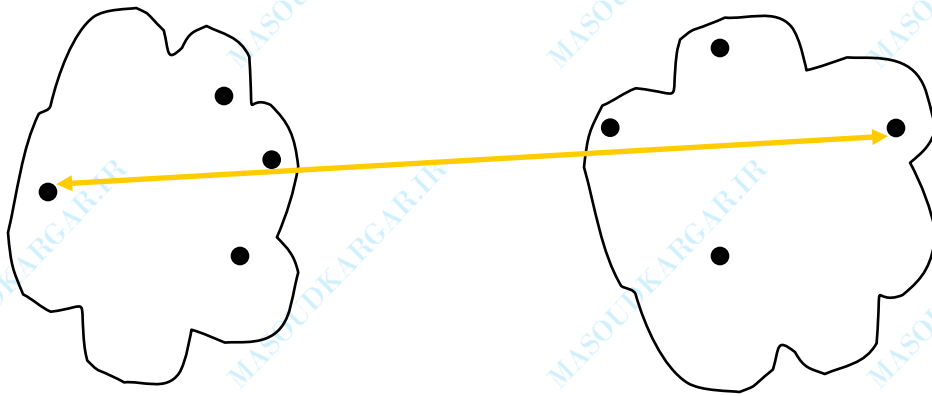


- ❑ MIN
- ❑ MAX
- ❑ Group Average
- ❑ Distance Between Centroids
- ❑ Other methods driven by an objective function
 - Ward's Method uses squared error

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

Proximity Matrix

How to Define Inter-Cluster Similarity

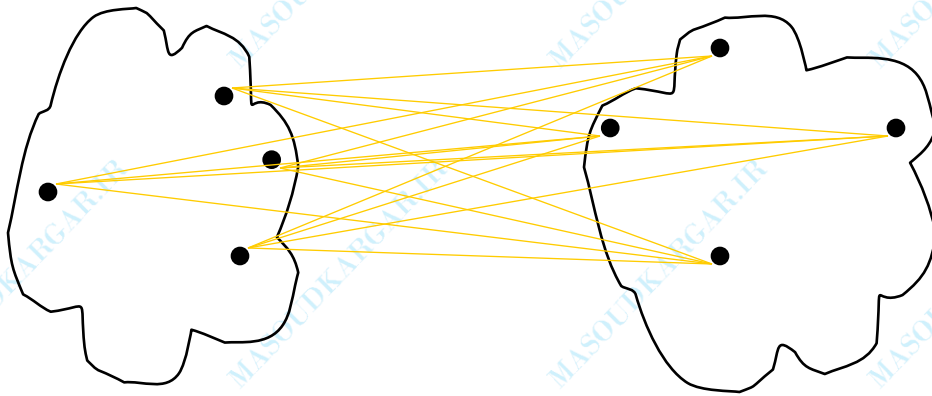


- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
 - Ward's Method uses squared error

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

Proximity Matrix

How to Define Inter-Cluster Similarity

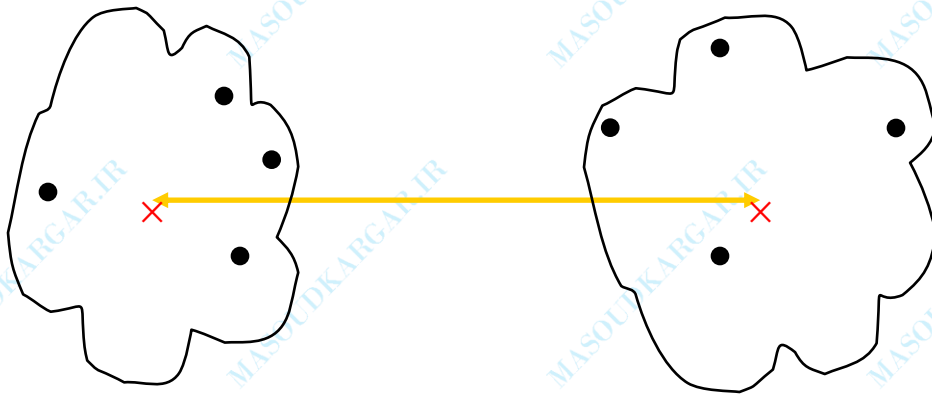


- ❑ MIN
- ❑ MAX
- ❑ **Group Average**
- ❑ Distance Between Centroids
- ❑ Other methods driven by an objective function
 - Ward's Method uses squared error

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

Proximity Matrix

How to Define Inter-Cluster Similarity



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

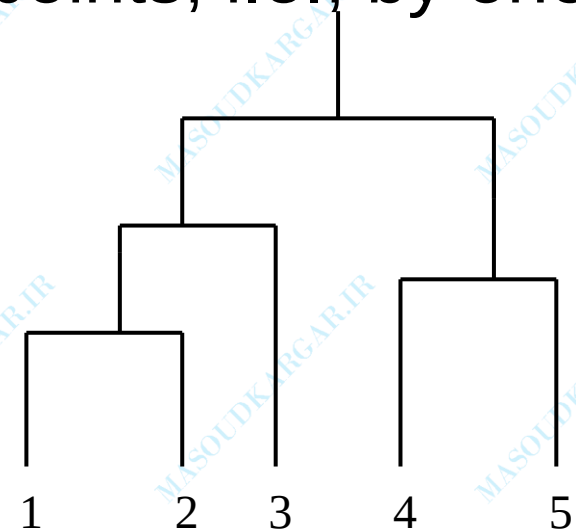
	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

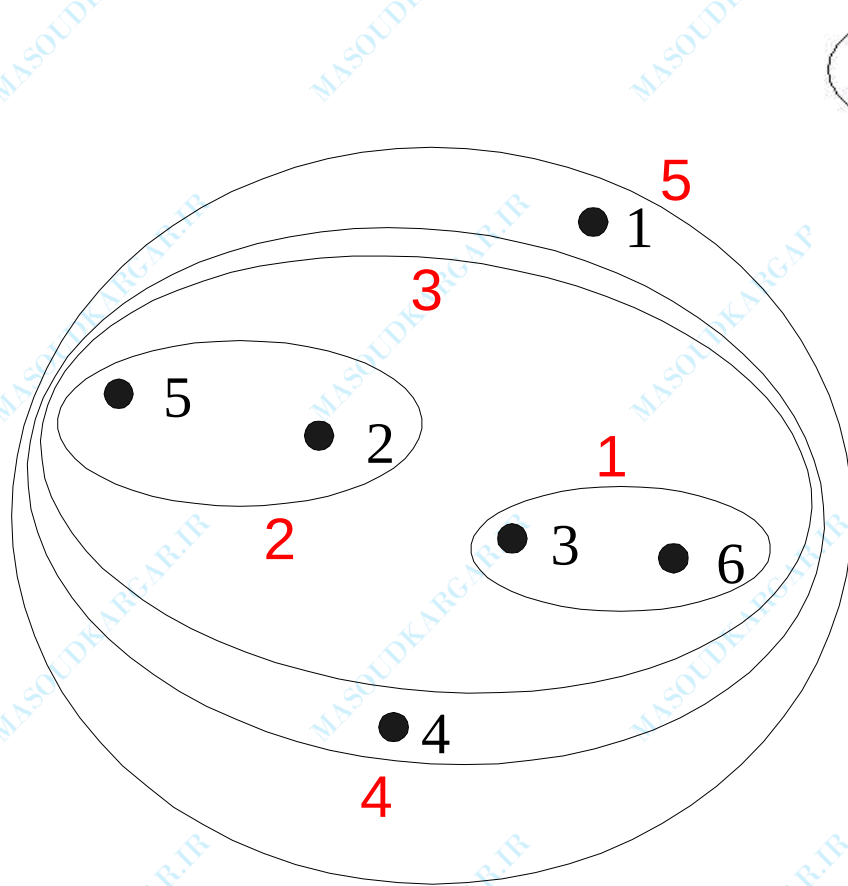
Cluster Similarity: MIN or Single Link

- 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.

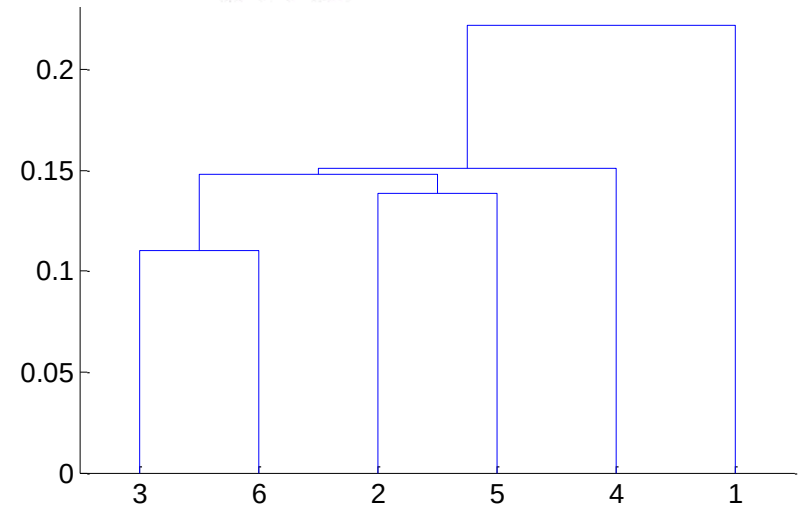
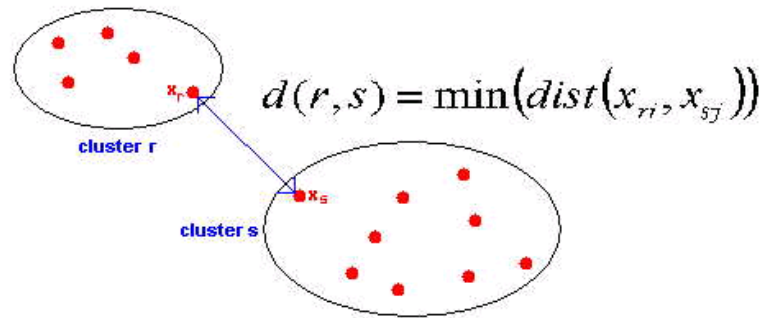
	l1	l2	l3	l4	l5
l1	1.00	0.90	0.10	0.65	0.20
l2	0.90	1.00	0.70	0.60	0.50
l3	0.10	0.70	1.00	0.40	0.30
l4	0.65	0.60	0.40	1.00	0.80
l5	0.20	0.50	0.30	0.80	1.00



Hierarchical Clustering: MIN

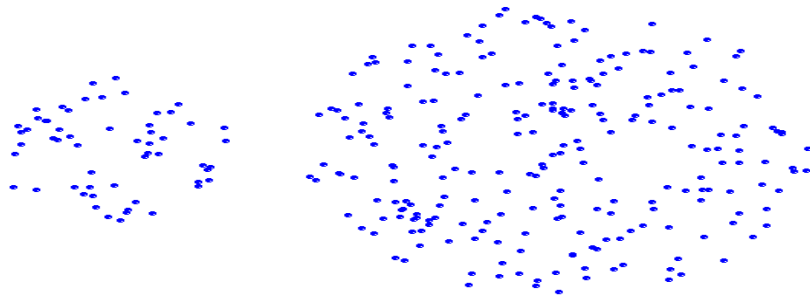


Nested Clusters

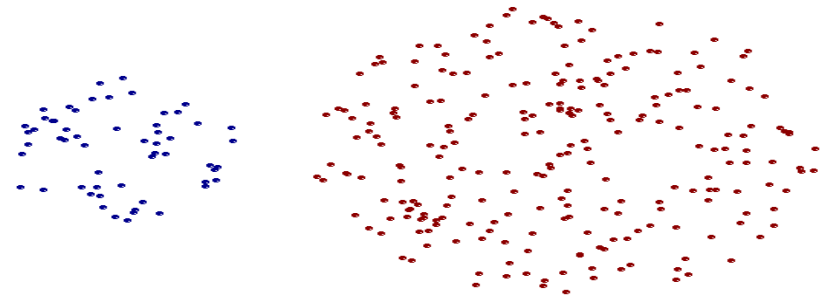


Dendrogram

Strength of MIN



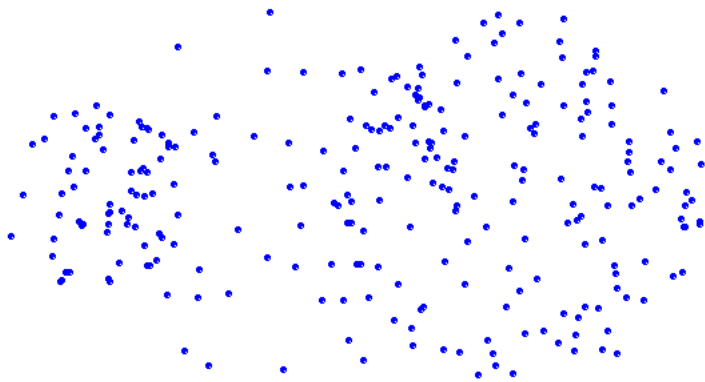
Original Points



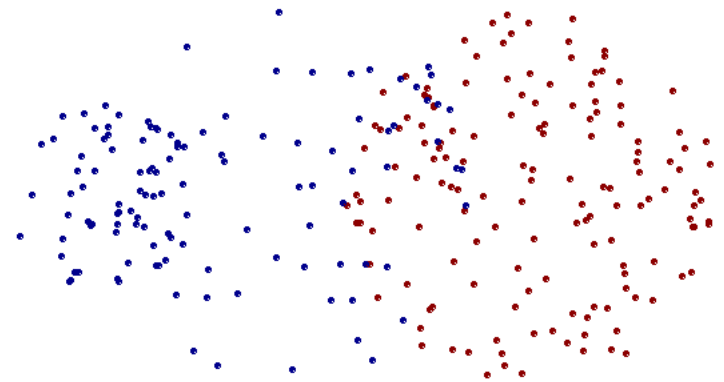
Two Clusters

- Can handle non-elliptical shapes

Limitations of MIN



Original Points



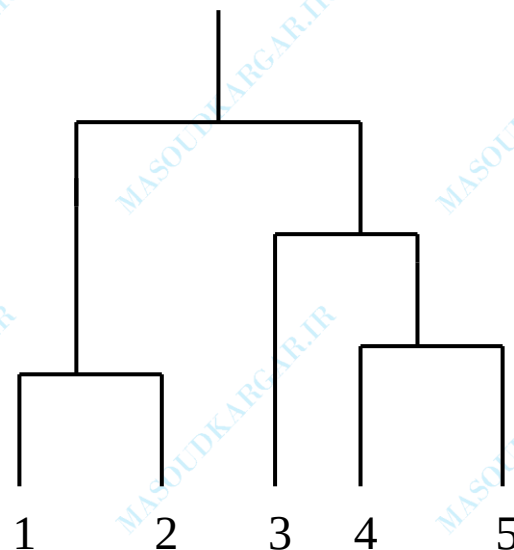
Two Clusters

- Sensitive to noise and outliers

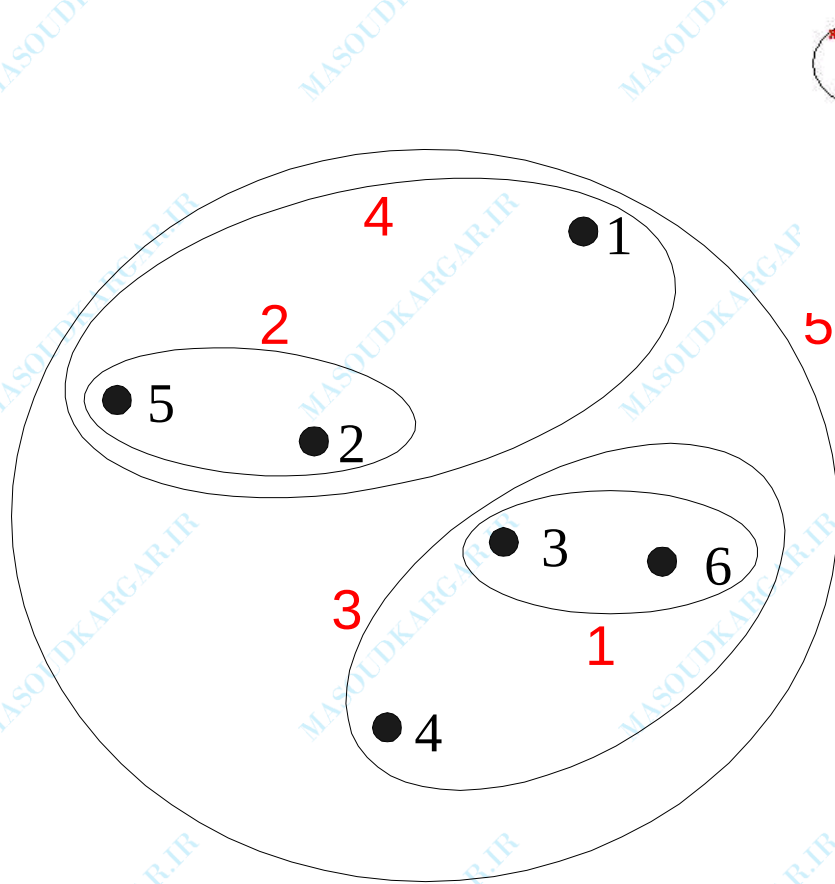
Cluster Similarity: MAX or Complete Linkage

- 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

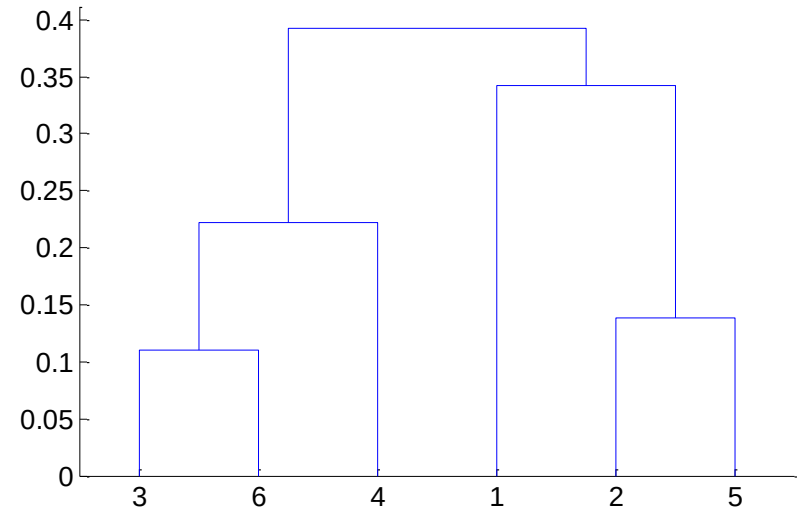
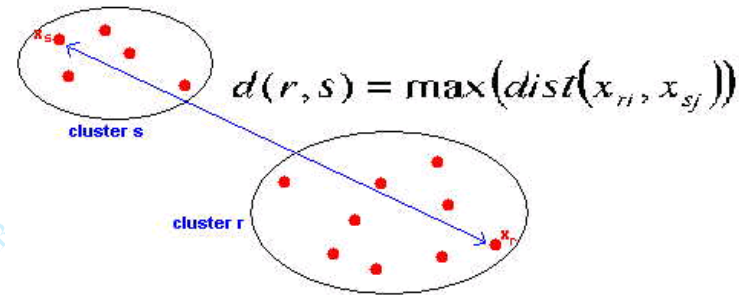
	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00



Hierarchical Clustering: MAX

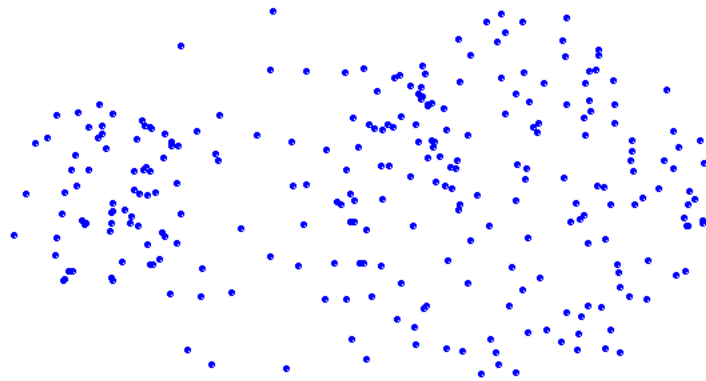


Nested Clusters

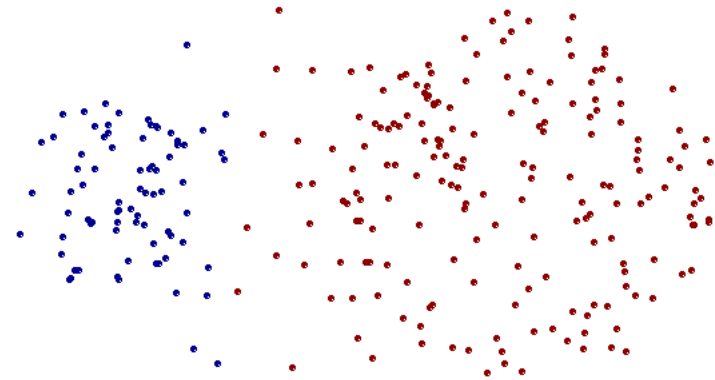


Dendrogram

Strength of MAX



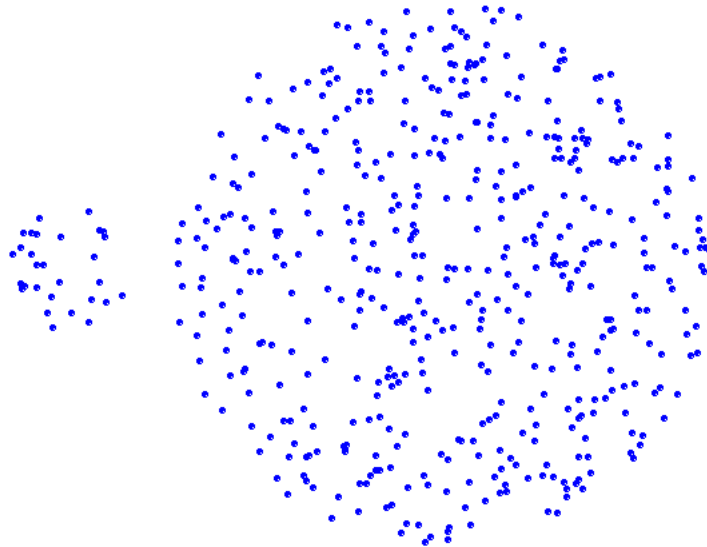
Original Points



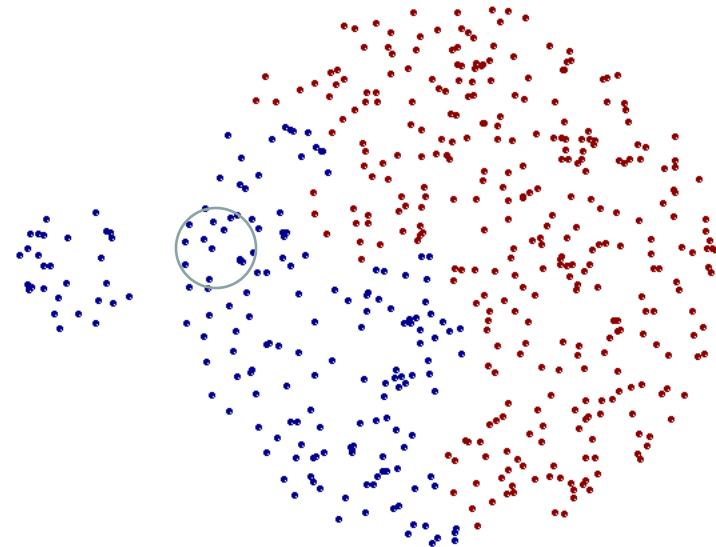
Two Clusters

- Less susceptible to noise and outliers

Limitations of MAX



Original Points



Two Clusters

- Tends to break large clusters
- Biased towards globular clusters

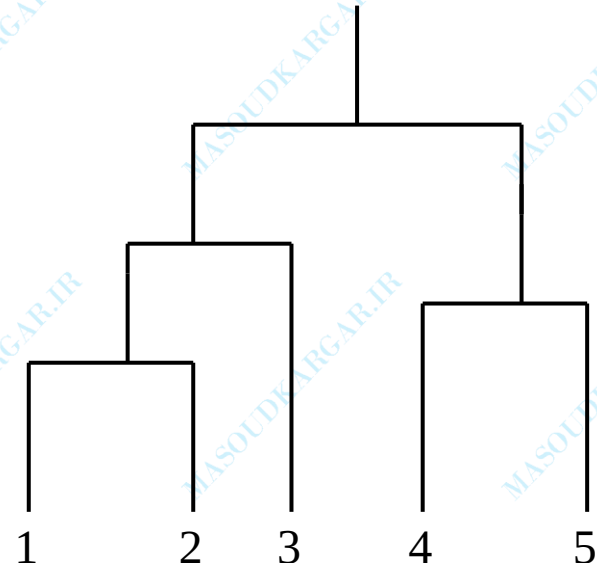
Cluster Similarity: Group Average

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

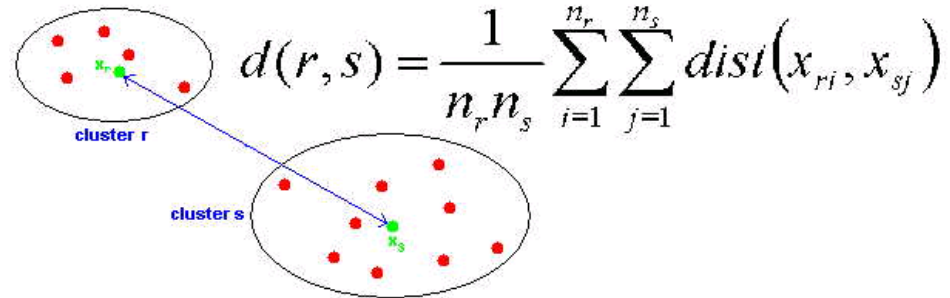
$$\text{proximity}(\text{Cluster}_i, \text{Cluster}_j) = \frac{\sum_{\substack{p_i \in \text{Cluster}_i \\ p_j \in \text{Cluster}_j}} \text{proximity}(p_i, p_j)}{|\text{Cluster}_i| * |\text{Cluster}_j|}$$

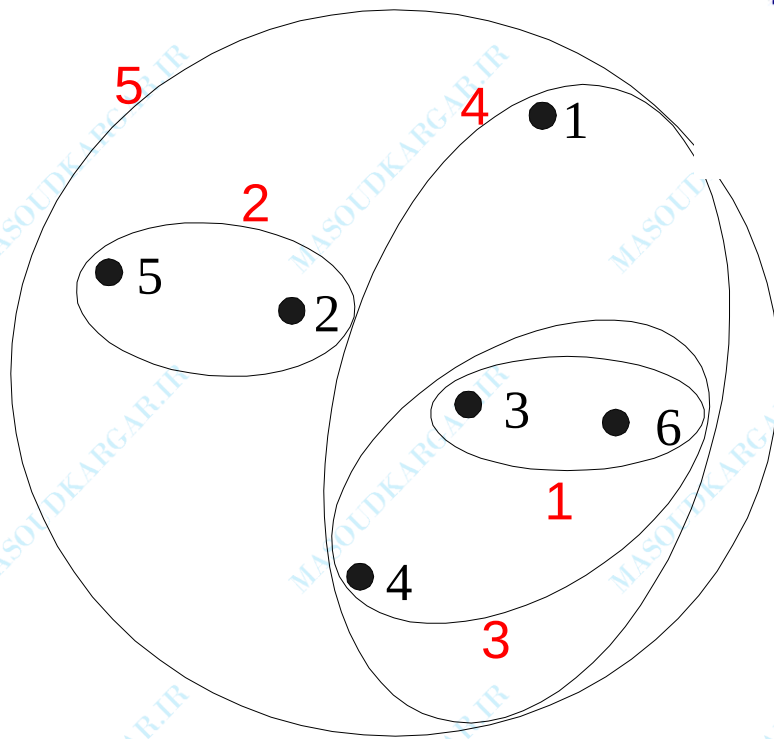
- Need to use average connectivity for scalability since total proximity favors large clusters

	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

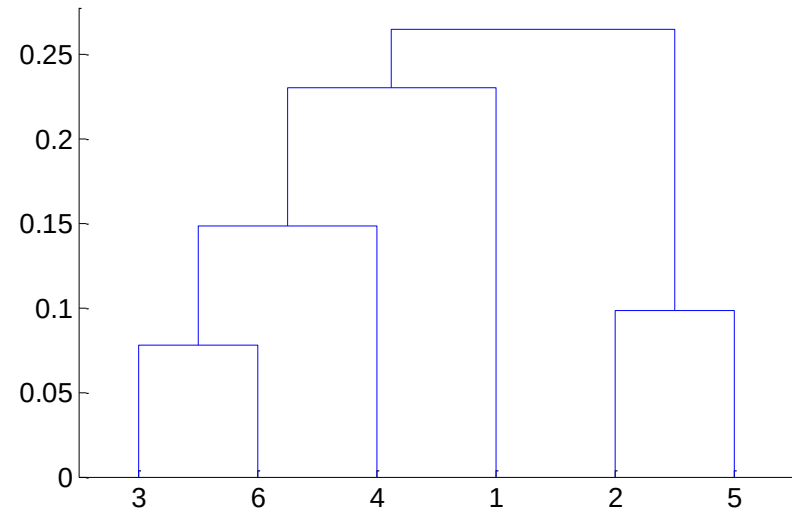


Hierarchical Clustering: Group Average

$$d(r, s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} \text{dist}(x_{ri}, x_{sj})$$




Nested Clusters



Dendrogram

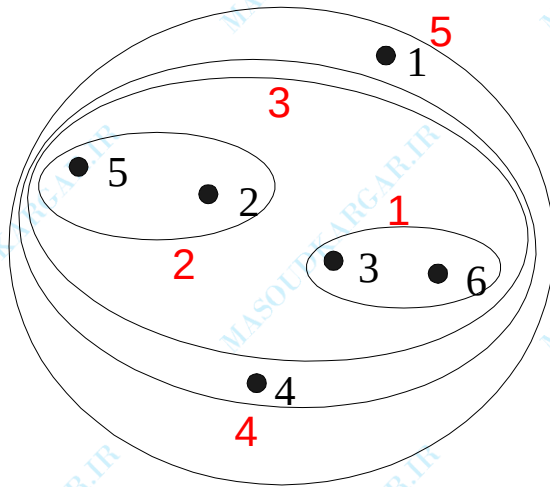
Hierarchical Clustering: Group Average

- Compromise between Single and Complete Link
- Strengths
 - Less susceptible to noise and outliers
- Limitations
 - Biased towards globular clusters

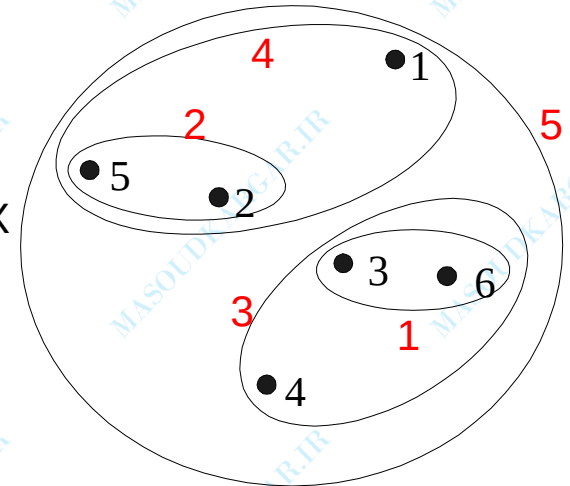
Cluster Similarity: Ward's Method

- 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

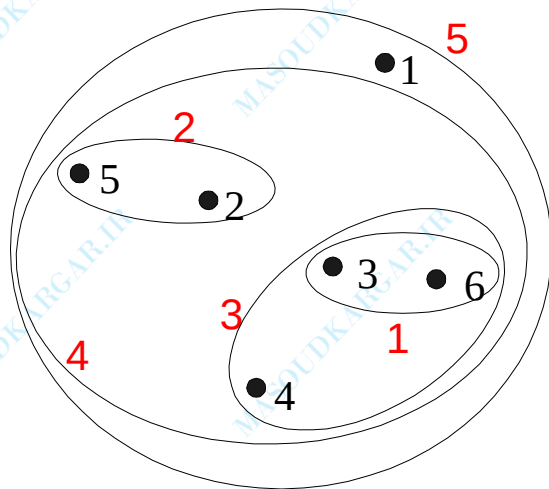
Hierarchical Clustering: Comparison



MIN

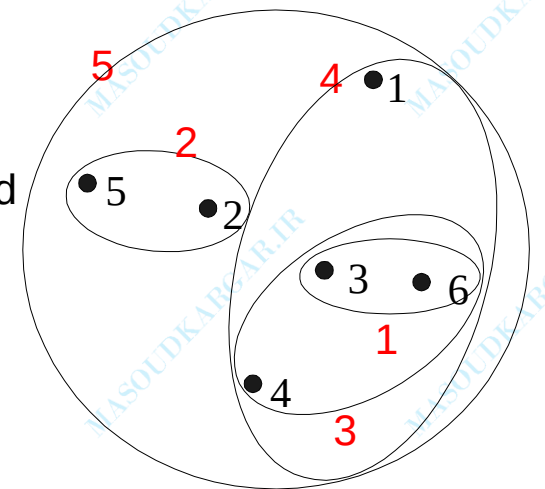


MAX



Group Average

Ward's Method



Hierarchical Clustering: Time and Space requirements

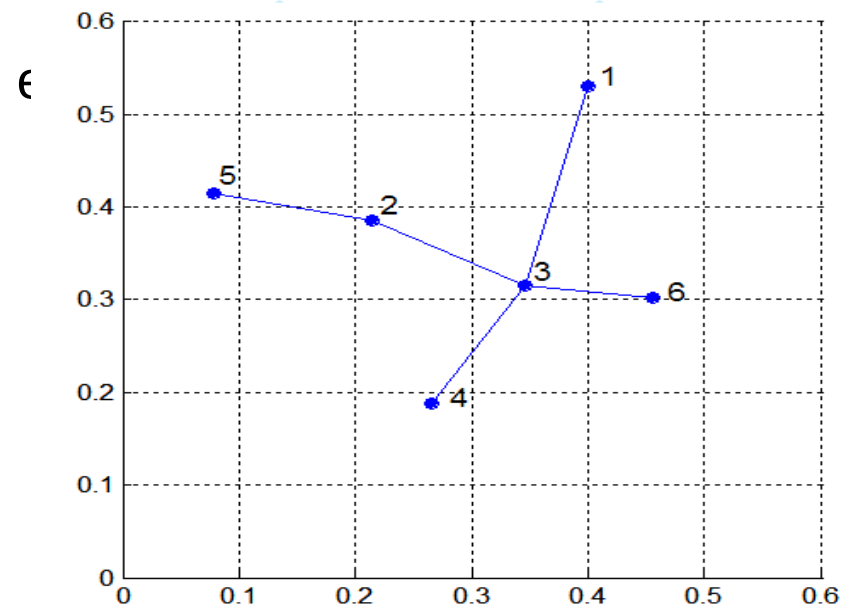
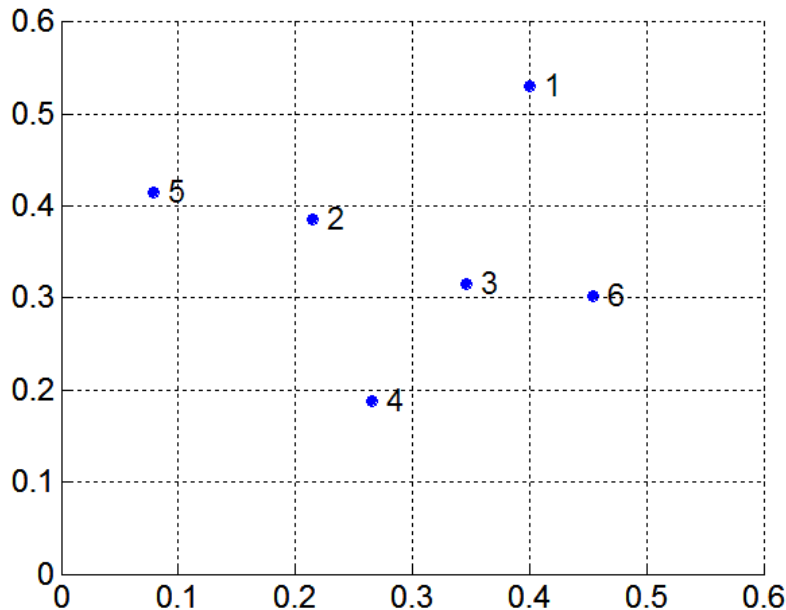
- $O(N^2)$ space since it uses the proximity matrix.
 - N is the number of points.
- $O(N^3)$ time in many cases
 - There are N steps and at each step the size, N^2 , proximity matrix must be updated and searched
 - Complexity can be reduced to $O(N^2 \log(N))$ time for some approaches

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 and convex shapes
 - Breaking large clusters

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



MST: Divisive Hierarchical Clustering

- Use MST for constructing hierarchy of

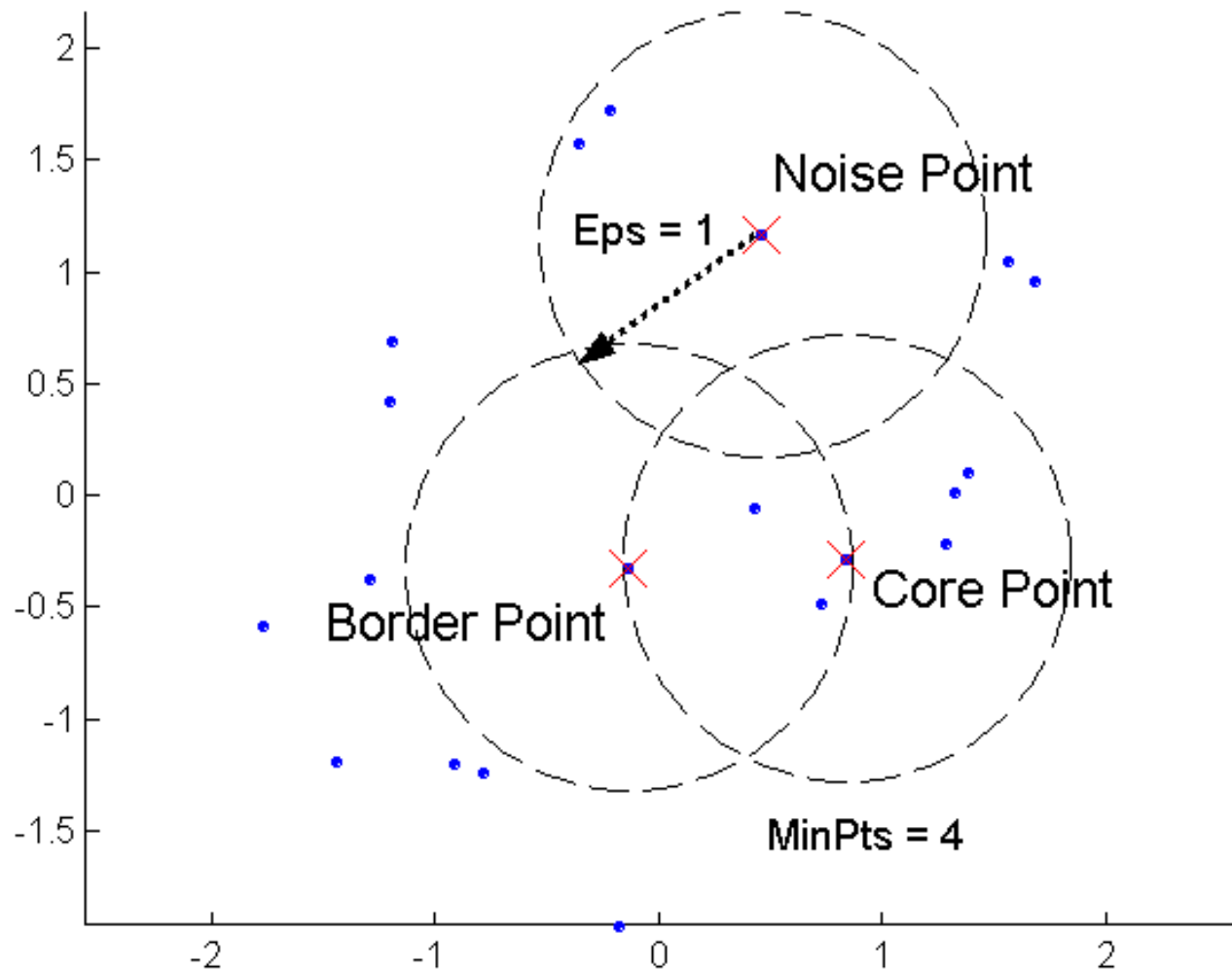
Algorithm 7.5 MST Divisive Hierarchical Clustering Algorithm

- 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
-

Density based Clustering: DBSCAN

- DBSCAN is a density-based algorithm.
 - Density = number of points within a specified radius (Eps)
 - A point is a **core point** if it has more than a specified number of points (MinPts) within Eps
 - These are points that are at the interior of a cluster
 - A **border point** has fewer than MinPts within Eps, but is in the neighborhood of a core point
 - A **noise point** is any point that is not a core point or a border point.

DBSCAN: Core, Border, and Noise Points



DBSCAN Algorithm

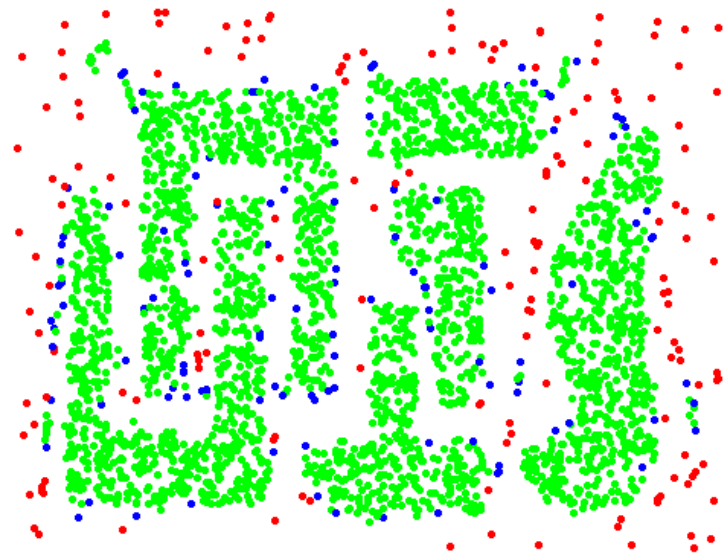
- Eliminate noise points

- $\text{current_cluster_label} \leftarrow 1$
 for all core points **do**
 if the core point has no cluster label **then**
 $\text{current_cluster_label} \leftarrow \text{current_cluster_label} + 1$
 Label the current core point with cluster label $\text{current_cluster_label}$
 end if
 for all points in the Eps -neighborhood, except i^{th} the point itself **do**
 if the point does not have a cluster label **then**
 Label the point with cluster label $\text{current_cluster_label}$
 end if
 end for
 end for

DBSCAN: Core, Border and Noise Points



Original Points



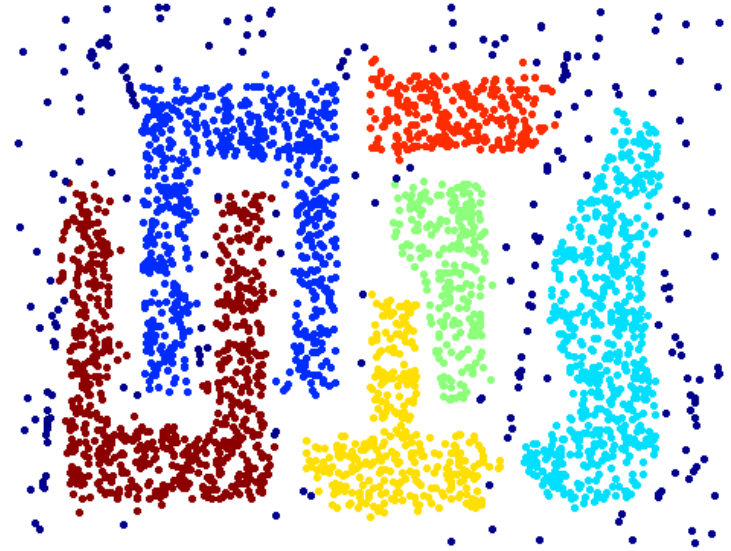
Point types: **core**,
border and **noise**

Eps = 10, MinPts = 4

When DBSCAN Works Well



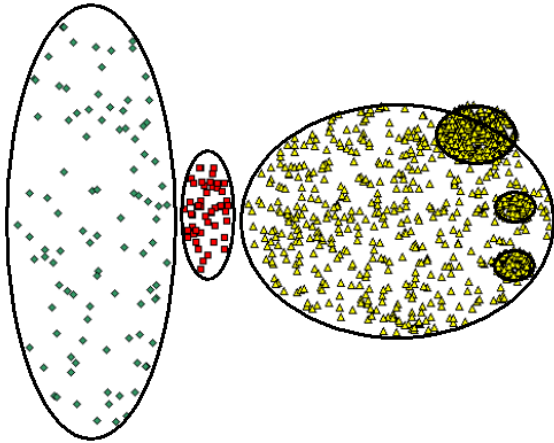
Original Points



Clusters

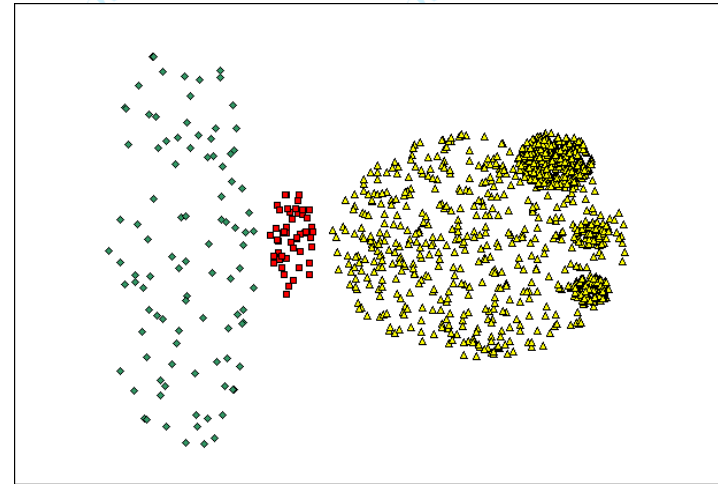
- Resistant to Noise
- Can handle clusters of different shapes and sizes

When DBSCAN Does NOT Work Well

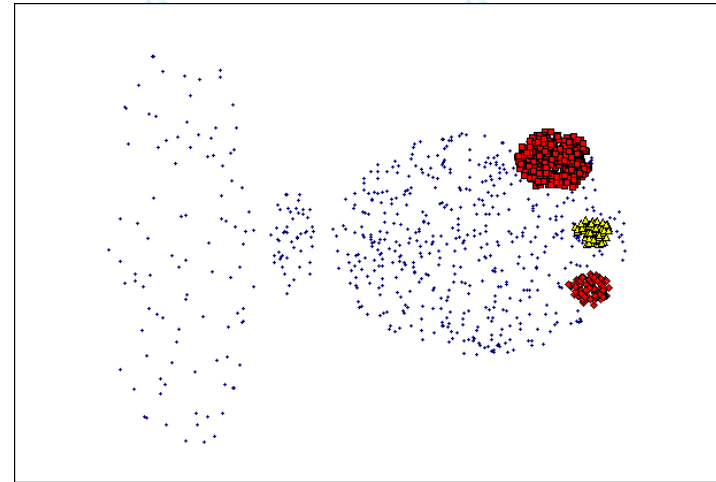


Original Points

- Varying densities
- High-dimensional data



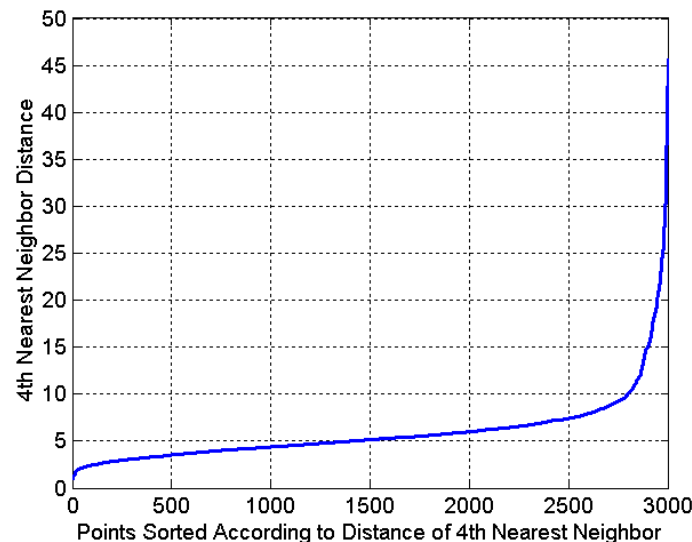
(MinPts=4, Eps=9.75).



(MinPts=4, Eps=9.92)

DBSCAN: Determining EPS and MinPts

- Idea is that for points in a cluster, their k^{th} nearest neighbors are at roughly the same distance
- Noise points have the k^{th} nearest neighbor at farther distance
- So, plot sorted distance of every point to its k^{th} nearest neighbor



Popular Clustering Tools

- Weka
- Hierarchical Clustering Explorer 3.0
 - <http://www.cs.umd.edu/hcil/hce/hce3.html>
- Cluster 3.0
 - <http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm>
- R Package
- Google a Matlab Clustering program yourself

Advanced Topics of Clustering

- Clustering with constraints
- Projected clustering
- Outlier detection
- Interactive clustering
- Clustering for data streams
- Database technology for clustering
- Categorical clustering.

Summary

- Hierarchical Clustering
- Different hierarchical clustering algorithms have different bias
- Density based clustering

قدردانی

- Dr. Jianjun Hu
<http://mleg.cse.sc.edu/edu/csce822/>
- University of South Carolina
- Department of Computer Science and Engineering