

Notes

- **Reminder: HW2 Due Today by 11:59PM**
 - ▣ **Problem 5: To make sure TA can run your code**, please provide a detailed ReadMe.txt file on how to run the program on the STD LINUX. If you installed/upgraded any package on STD LINUX, you should also mention it (with version number) in the ReadMe.txt file.
 - ▣ Make sure your handwriting is readable to TA (if you do not type).

- Review session on Thursday

- Midterm next Tuesday (10/08/2019)

CSE 5243 INTRO. TO DATA MINING

Cluster Analysis: Basic Concepts and Methods

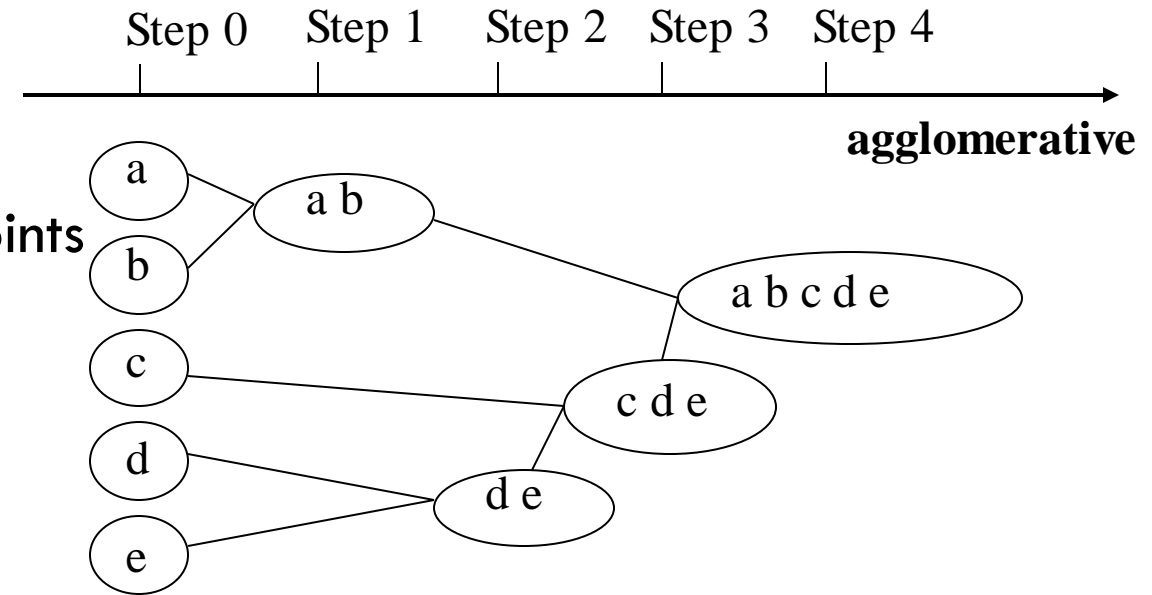
Huan Sun, CSE@The Ohio State University

Chapter 10. Cluster Analysis: Basic Concepts and Methods

- Cluster Analysis: An Introduction
- Partitioning Methods (K-means and its variants)
- **Hierarchical Methods**
- Density-based Methods
- Evaluation of Clustering
- Summary

Agglomerative Clustering Algorithm (Recap)

- More popular hierarchical clustering technique
- Basic algorithm is straightforward
 1. Compute the proximity matrix for all data points
 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

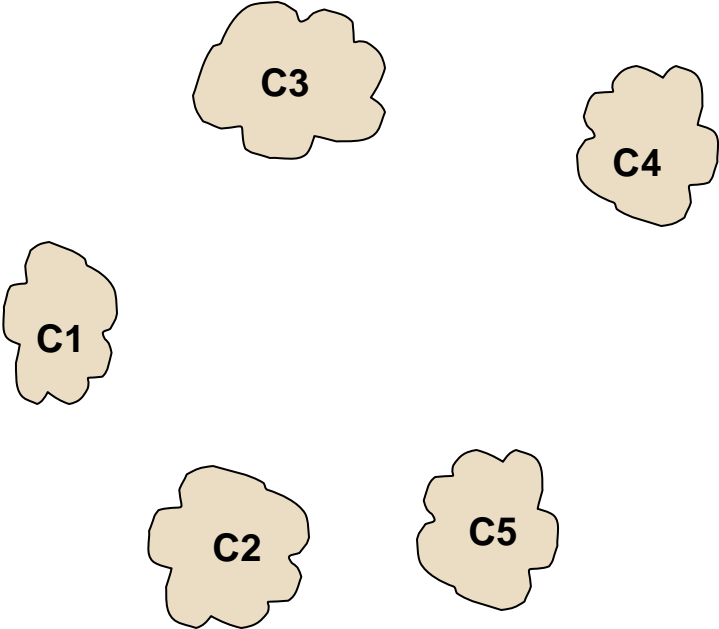


Agglomerative Clustering Algorithm (Recap)

- More popular hierarchical clustering technique
- Basic algorithm is straightforward
 1. Compute the proximity matrix for all data points
 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/similarity between clusters** distinguish the different algorithms

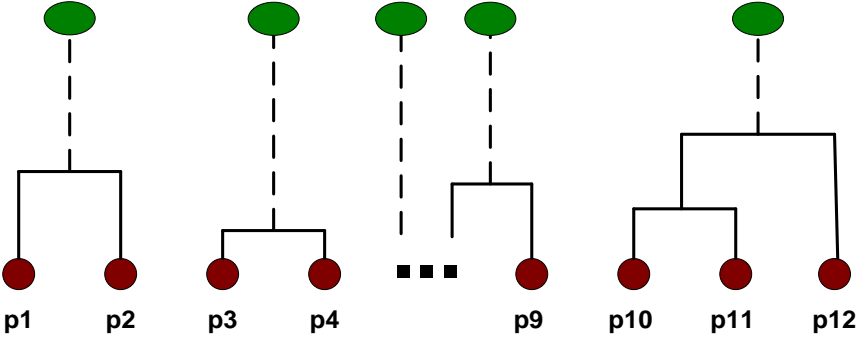
Intermediate Situation

□ For example, at a certain step, we have clusters



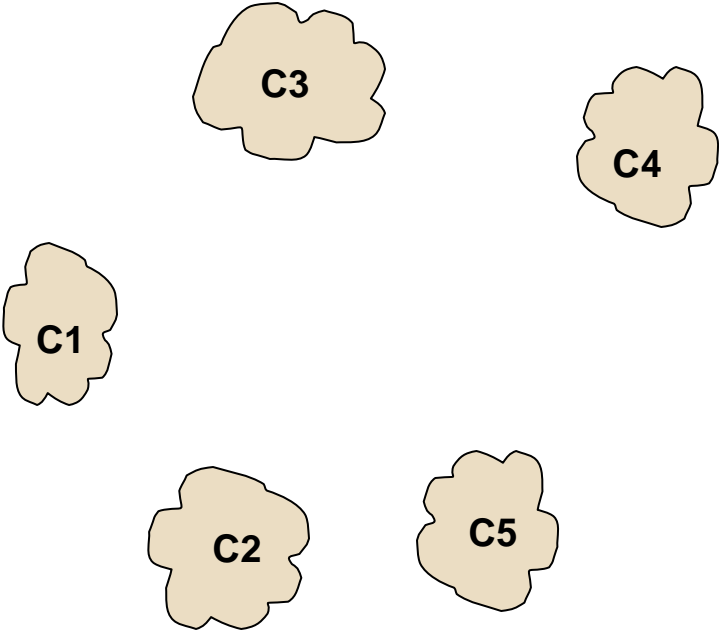
	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix



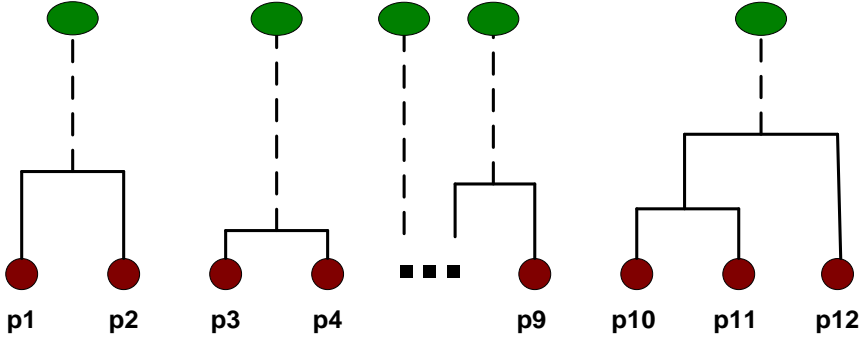
Intermediate Situation

□ How to compute the proximity matrix?

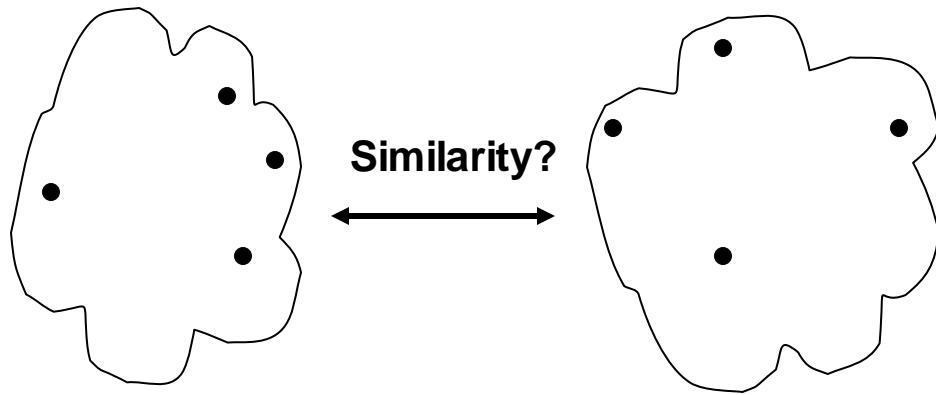


	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix



How to Define Inter-Cluster Similarity



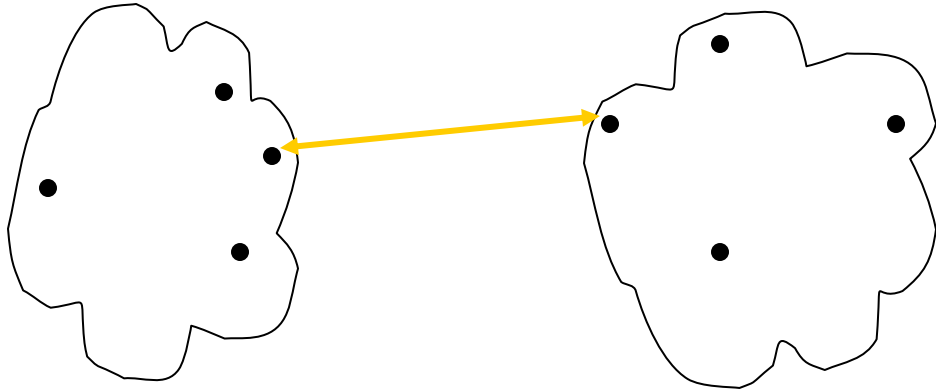
- MIN
- MAX
- Group Average
- Distance Between Centroids

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

· **Proximity Matrix**

·

How to Define Inter-Cluster Similarity



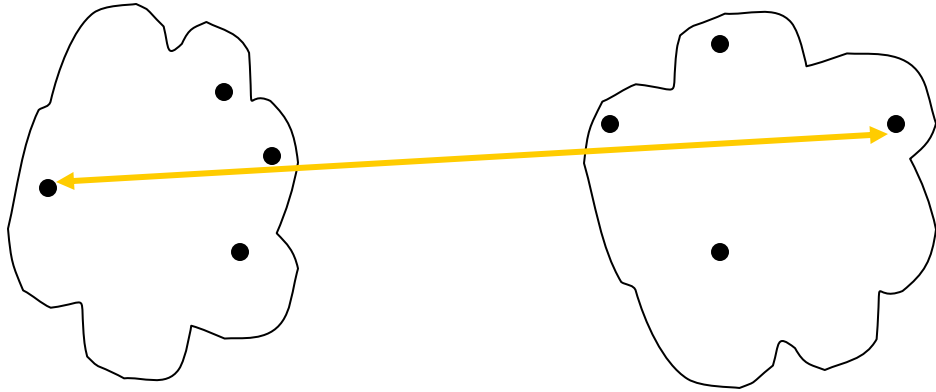
- MIN
- MAX
- Group Average
- Distance Between Centroids

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

· **Proximity Matrix**

·

How to Define Inter-Cluster Similarity



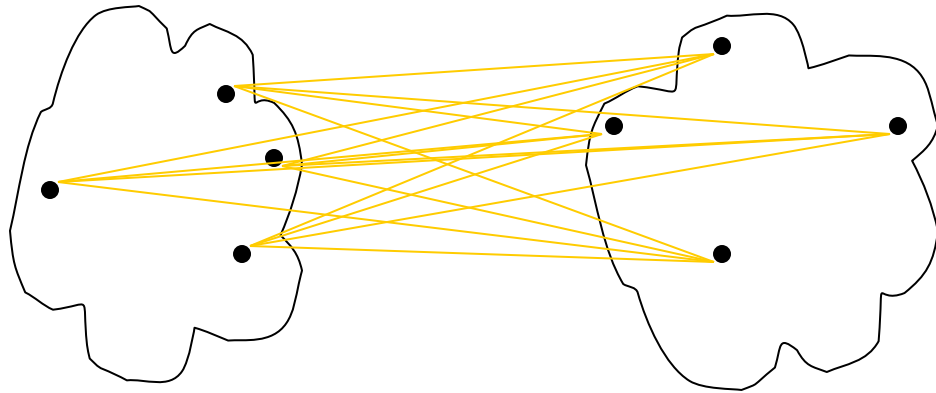
- MIN
- MAX
- Group Average
- Distance Between Centroids

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

· **Proximity Matrix**

·

How to Define Inter-Cluster Similarity



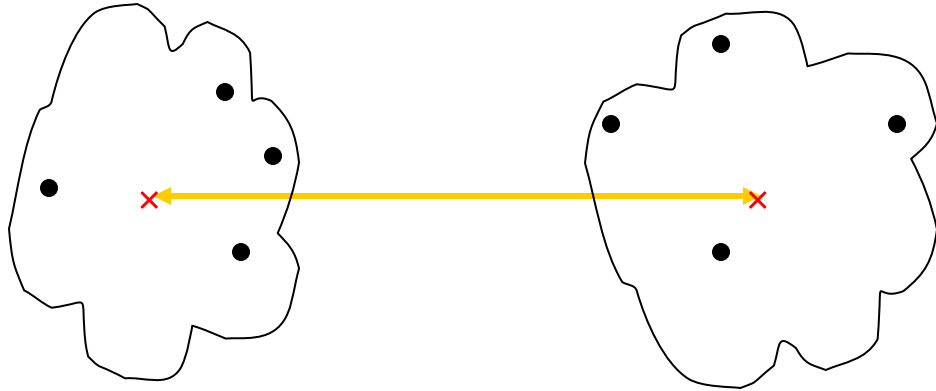
- MIN
- MAX
- **Group Average**
- Distance Between Centroids

	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

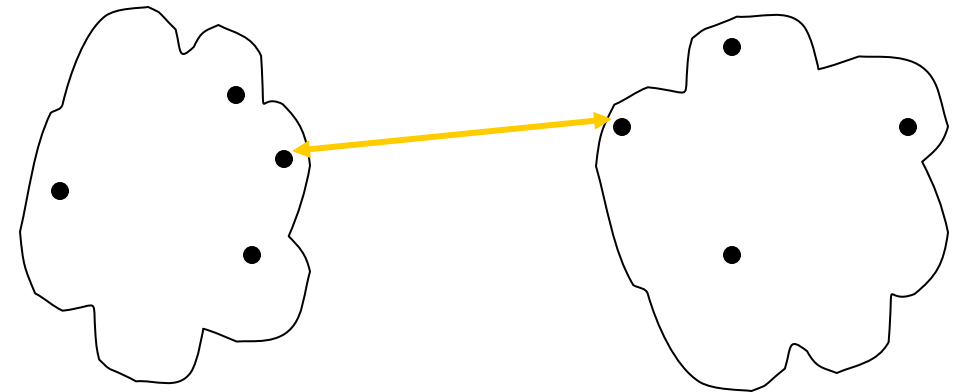
	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						

· Proximity Matrix

·

Cluster Similarity: MIN or Single Linkage

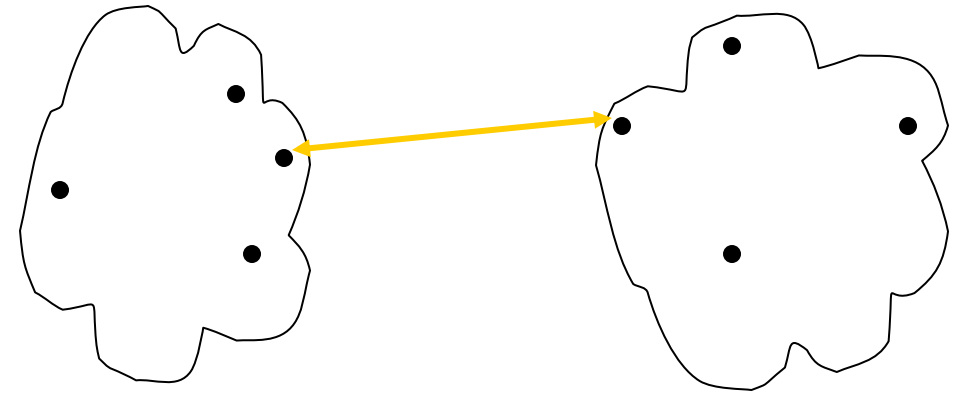
- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters



Why single linkage?

Cluster Similarity: MIN or Single Linkage

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



The name comes from the observation that if we connect two points in two clusters within this distance, typically only a single link would exist.

Cluster Similarity: MIN or Single Linkage

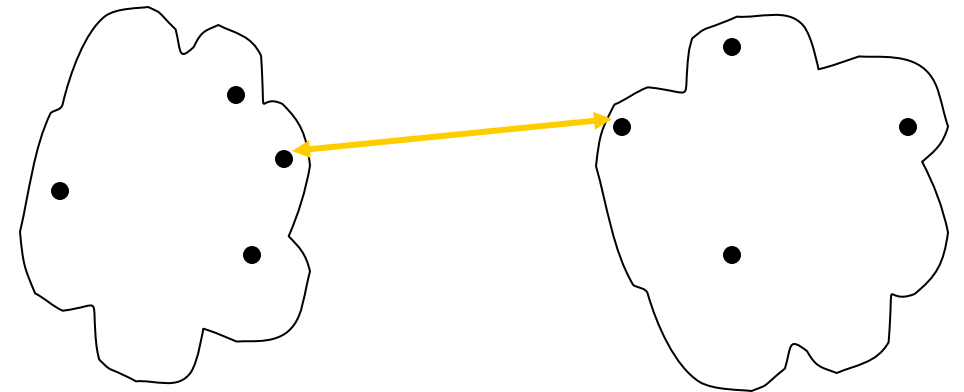
- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters

Let us define the distance between two points using Euclidean distance:

$$\delta(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2 = \left(\sum_{i=1}^d (x_i - y_i)^2 \right)^{1/2}$$

Using single link, the **distance** between two clusters C_i and C_j is then:

$$\delta(C_i, C_j) = \underline{\min}\{\delta(\mathbf{x}, \mathbf{y}) \mid \mathbf{x} \in C_i, \mathbf{y} \in C_j\}$$



The name comes from the observation that if we connect two points in two clusters within this distance, typically only a single link would exist.

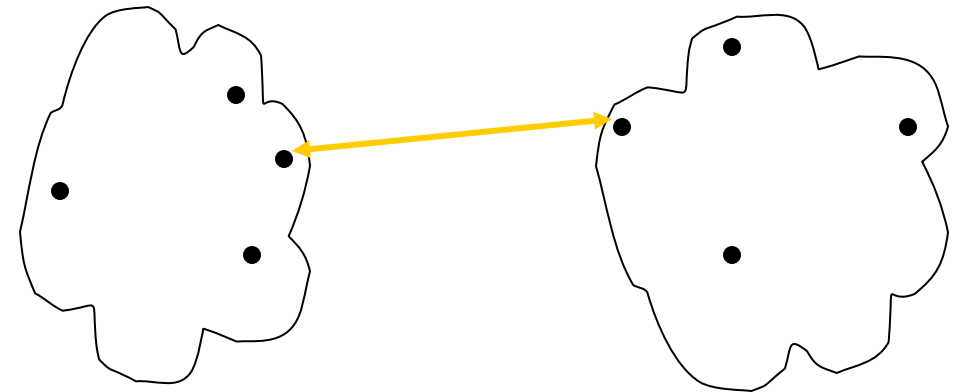
Cluster Similarity: MIN or Single Linkage

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

What if we define the similarity (**not distance**) between two points?

Using single link, the similarity between two clusters C_i and C_j is then:

$$\text{Sim}(C_i, C_j) = \underline{\text{max}}\{\text{sim}(x, y) \mid x \text{ in } C_i, y \text{ in } C_j\}$$



The name comes from the observation that if we connect two points in two clusters within this distance, typically only a single link would exist.

Cluster Similarity: MIN or Single Linkage

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

| | | | |
1 2 3 4 5

Example (**Similarity** Matrix)

Cluster Similarity: MIN or Single Linkage

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

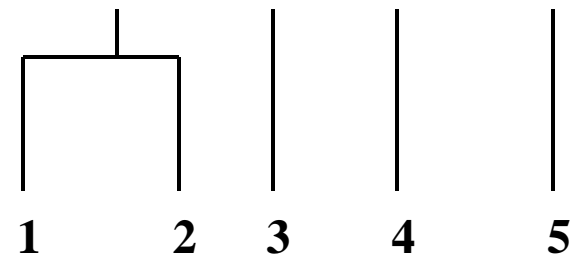
	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00

Example (**Similarity** Matrix)

Cluster Similarity: MIN or Single Linkage

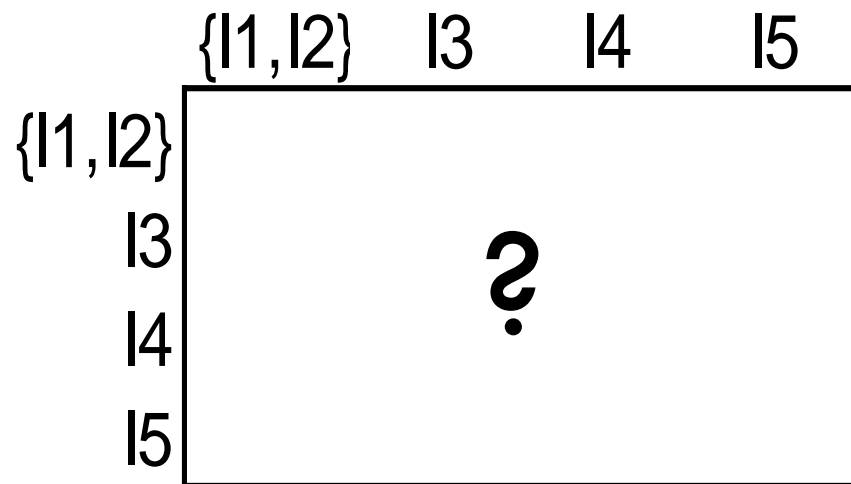
- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters

	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



Cluster Similarity: MIN or Single Linkage

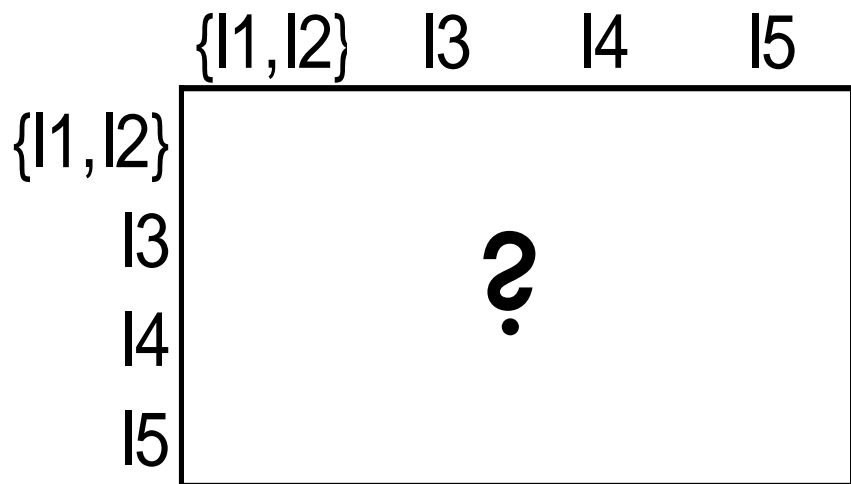
- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters



Update proximity matrix with new cluster {1, 12}

Cluster Similarity: MIN or Single Linkage

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



Update proximity matrix with new cluster $\{1,1,2\}$

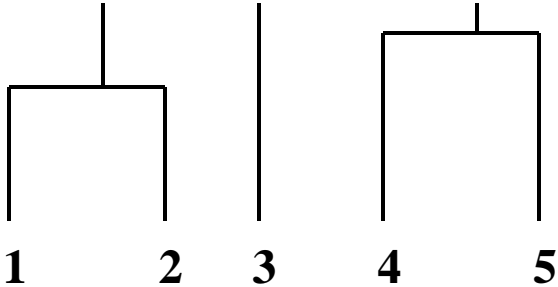
	11	12	13	14	15
11	1.00	0.90	0.10	0.65	0.20
12	0.90	1.00	0.70	0.60	0.50
13	0.10	0.70	1.00	0.40	0.30
14	0.65	0.60	0.40	1.00	0.80
15	0.20	0.50	0.30	0.80	1.00

Cluster Similarity: MIN or Single Linkage

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

	{1,12}	13	14	15
{1,12}	1.00	0.70	0.65	0.50
13	0.70	1.00	0.40	0.30
14	0.65	0.40	1.00	0.80
15	0.50	0.30	0.80	1.00

Update proximity matrix with new cluster {1, 12}



Cluster Similarity: MIN or Single Linkage

- 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**.

	{1,12}	13	{14,15}
{1,12}	1.00	0.70	?
13	0.70	1.00	?
{14,15}			

Update proximity matrix with new cluster {1, 12} and {14, 15}

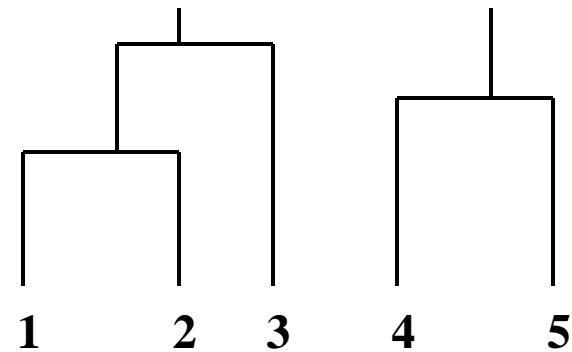
	{1,12}	13	14	15
{1,12}	1.00	0.70	0.65	0.50
13	0.70	1.00	0.40	0.30
14	0.65	0.40	1.00	0.80
15	0.50	0.30	0.80	1.00

Cluster Similarity: MIN or Single Linkage

- 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**.

	{1,12}	13	{14,15}
{1,12}	1.00	0.70	0.65
13	0.70	1.00	0.40
{14,15}	0.65	0.40	1.00

Update proximity matrix with new cluster {11, 12} and {14, 15}

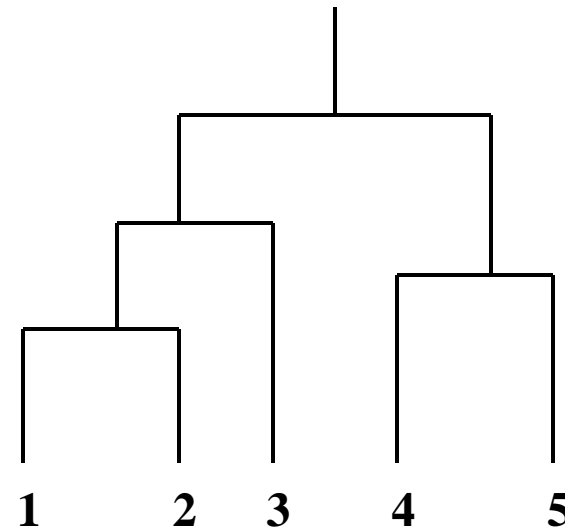


Cluster Similarity: MIN or Single Linkage

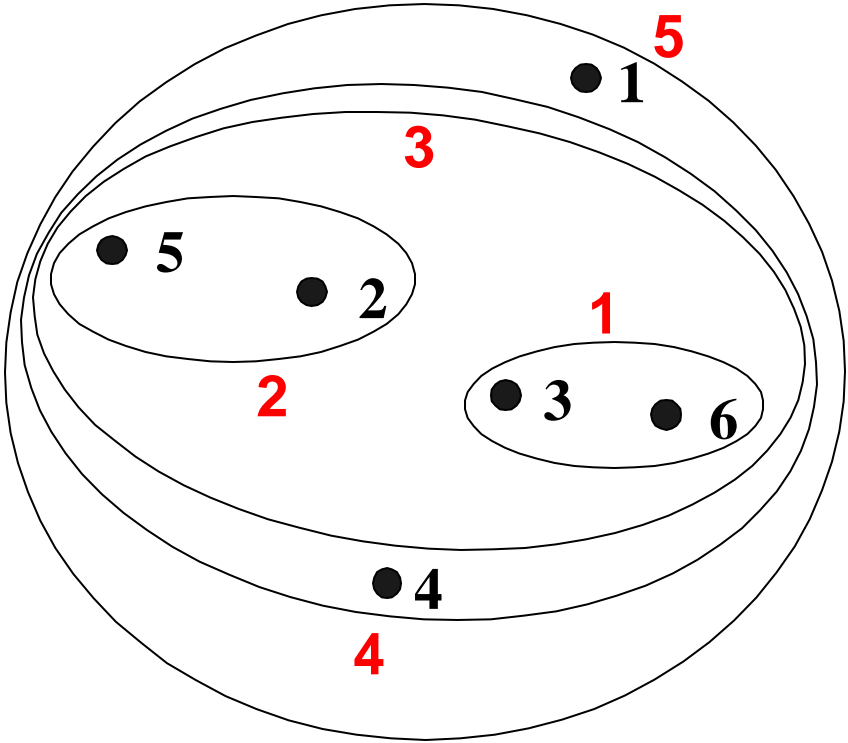
- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters
 - ▣ Determined by one pair of points, i.e., by **one link in the proximity graph**.

	{1, 2, 3}	{4, 5}
{1, 2, 3}	1.00	0.65
{4, 5}	0.65	1.00

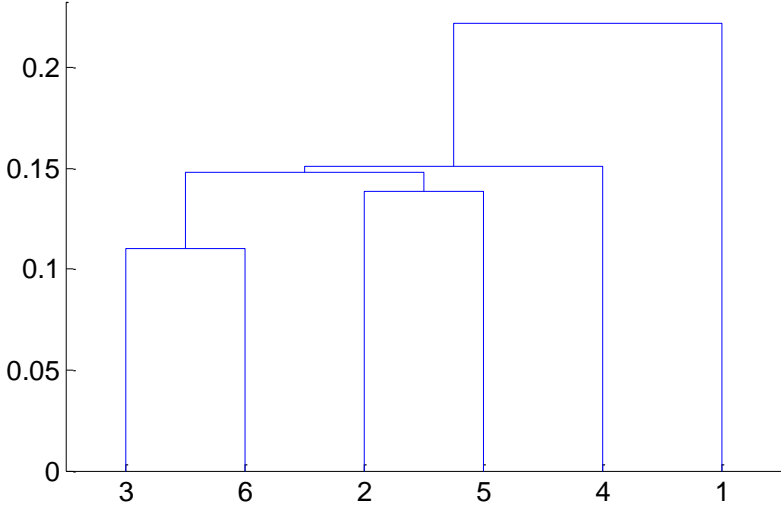
Only two clusters are left.



Hierarchical Clustering: MIN

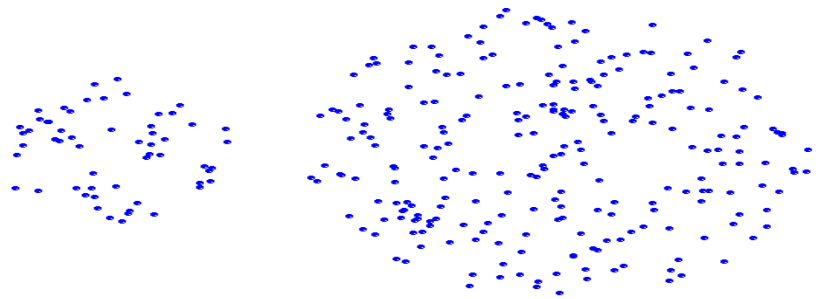


Nested Clusters

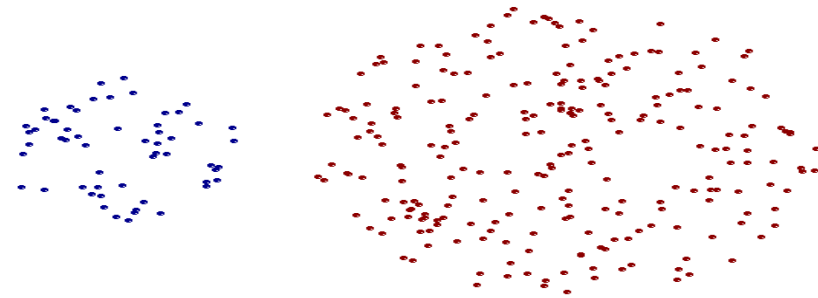


Dendrogram

Strength of MIN



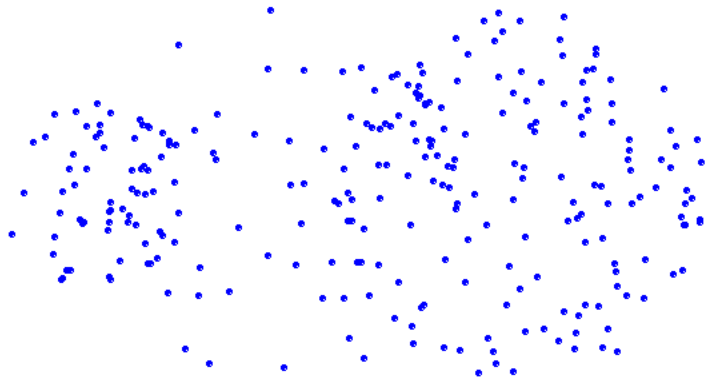
Original Points



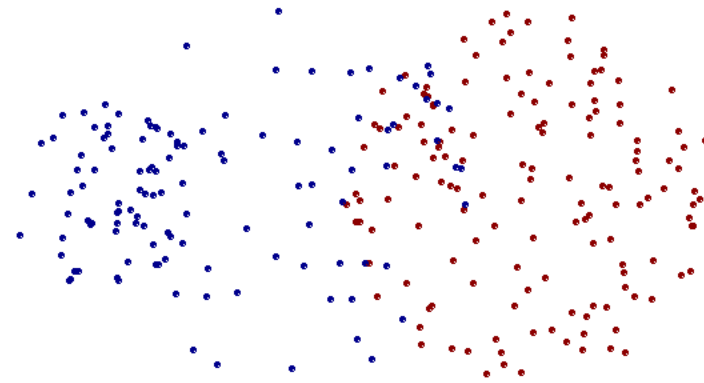
Two Clusters

- **Can handle non-globular or some irregular shapes**

Limitations of MIN



Original Points



Two Clusters

- **Sensitive to noise and outliers**

Cluster Similarity: MAX

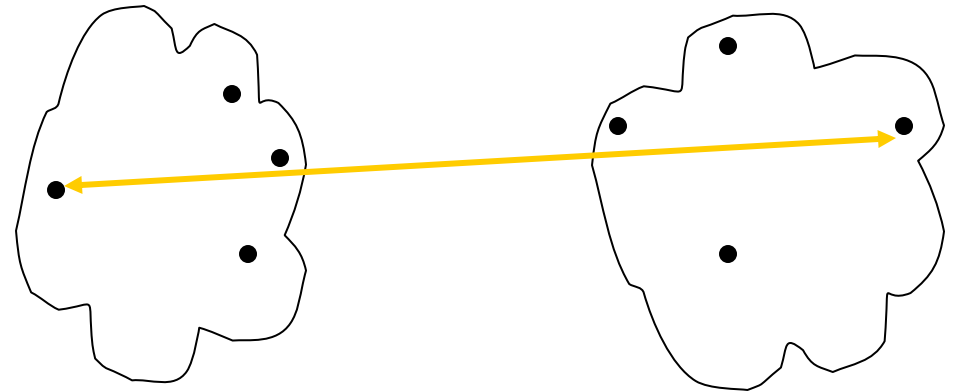
- Similarity of two clusters is based on **the two least similar (most distant)** points in the different clusters
 - ▣ Determined by all pairs of points in the two clusters

Let us define the distance between two points using Euclidean distance:

$$\delta(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2 = \left(\sum_{i=1}^d (x_i - y_i)^2 \right)^{1/2}$$

Using MAX link, the distance between two clusters C_i and C_j is then:

$$\delta(C_i, C_j) = \max\{\delta(\mathbf{x}, \mathbf{y}) \mid \mathbf{x} \in C_i, \mathbf{y} \in C_j\}$$



Cluster Similarity: MAX or Complete Linkage

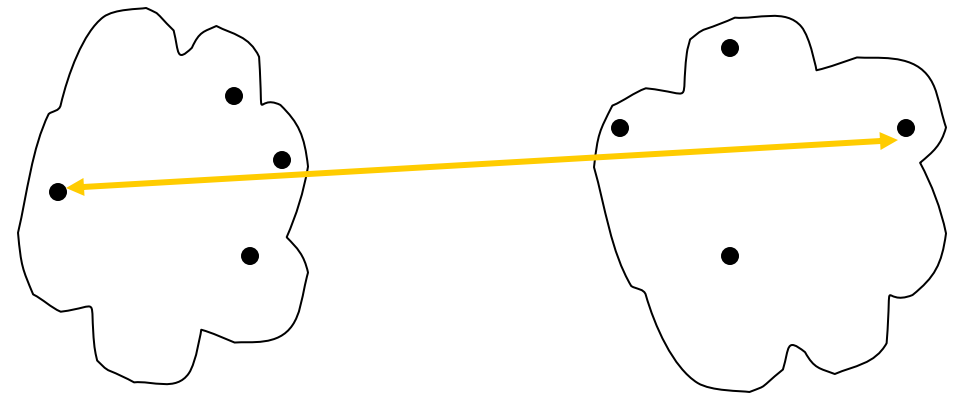
- Similarity of two clusters is based on **the two least similar (most distant) points** in the different clusters
 - ▣ Determined by all pairs of points in the two clusters

Let us define the distance between two points using Euclidean distance:

$$\delta(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2 = \left(\sum_{i=1}^d (x_i - y_i)^2 \right)^{1/2}$$

Using MAX link, the distance between two clusters C_i and C_j is then:

$$\delta(C_i, C_j) = \max\{\delta(\mathbf{x}, \mathbf{y}) \mid \mathbf{x} \in C_i, \mathbf{y} \in C_j\}$$



Why complete linkage?

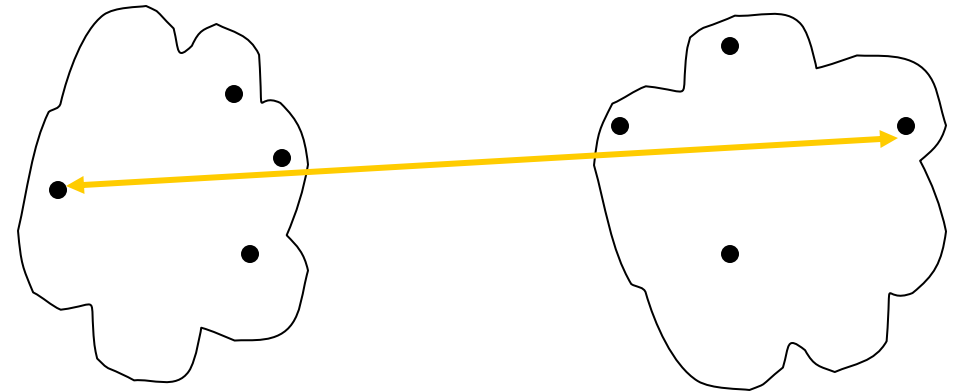
Cluster Similarity: MAX or Complete Linkage

- 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

What if we define the similarity (**not distance**) between two points?

Using MAX link, the similarity between two clusters C_i and C_j is then:

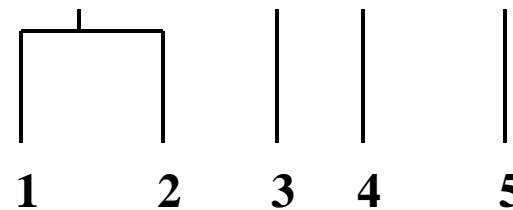
$$\text{Sim}(C_i, C_j) = \mathbf{min}\{\text{sim}(x, y) \mid x \text{ in } C_i, y \text{ in } C_j\}$$



Cluster Similarity: MAX or Complete Linkage

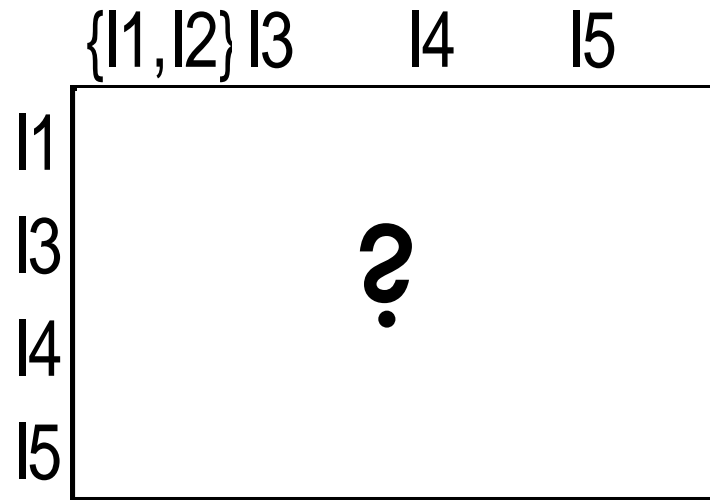
- Similarity of two clusters is based on **the two least similar (most distant)** points in the different clusters
 - ▣ Determined by all pairs of points in the two clusters

	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



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

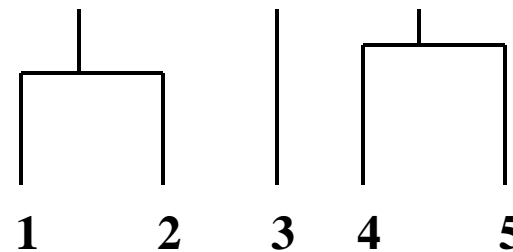


	1	2	3	4	5
1	1.00	0.90	0.10	0.65	0.20
2	0.90	1.00	0.70	0.60	0.50
3	0.10	0.70	1.00	0.40	0.30
4	0.65	0.60	0.40	1.00	0.80
5	0.20	0.50	0.30	0.80	1.00

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

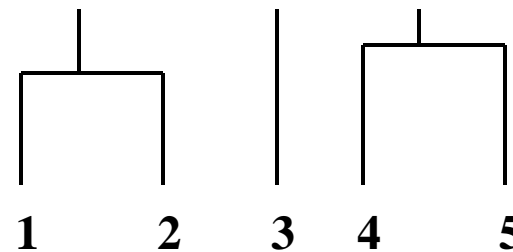
	{1,2}	3	4	5
1	1.00	0.10	0.60	0.20
3	0.10	1.00	0.40	0.30
4	0.60	0.40	1.00	0.80
5	0.20	0.30	0.80	1.00



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

	{1,2}	3	4	5
1	1.00	0.10	0.60	0.20
3	0.10	1.00	0.40	0.30
4	0.60	0.40	1.00	0.80
5	0.20	0.30	0.80	1.00

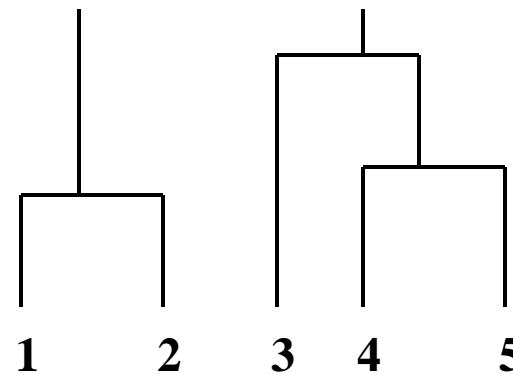


Which two clusters should be merged next?

Cluster Similarity: MAX or Complete Linkage

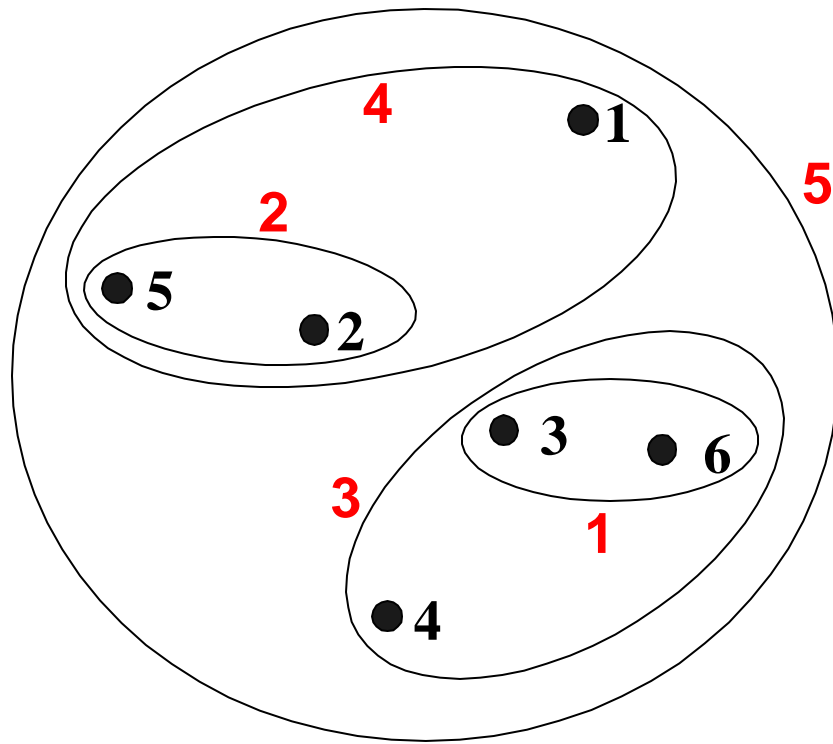
- Similarity of two clusters is based on **the two least similar (most distant)** points in the different clusters
 - ▣ Determined by all pairs of points in the two clusters

	{1,2}	3	4	5
1	1.00	0.10	0.60	0.20
3	0.10	1.00	0.40	0.30
4	0.60	0.40	1.00	0.80
5	0.20	0.30	0.80	1.00

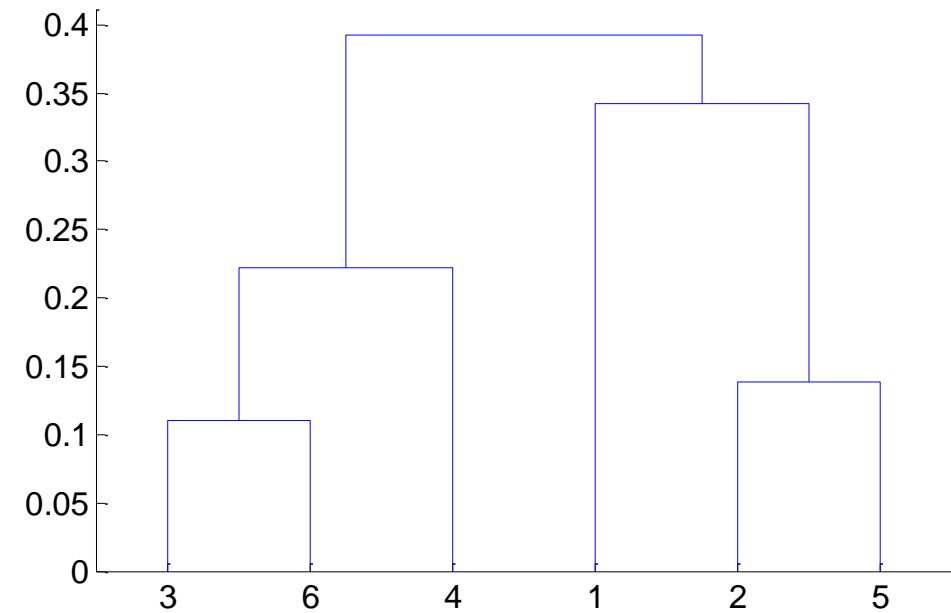


Merge {3} with {4,5}, why?

Hierarchical Clustering: MAX

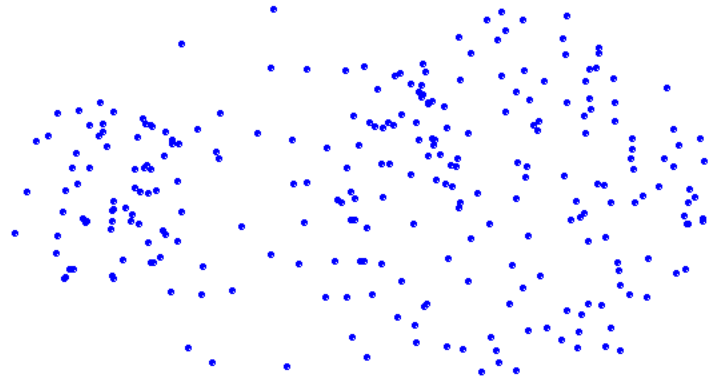


Nested Clusters

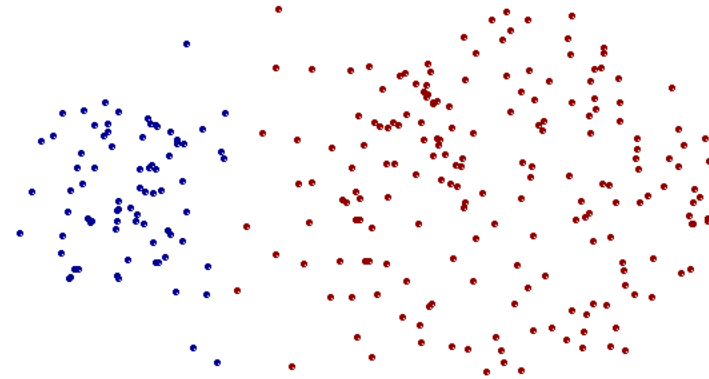


Dendrogram

Strength of MAX



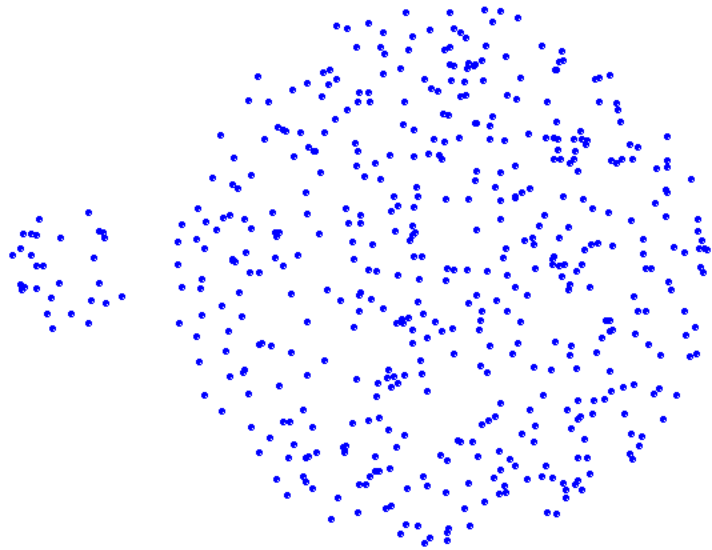
Original Points



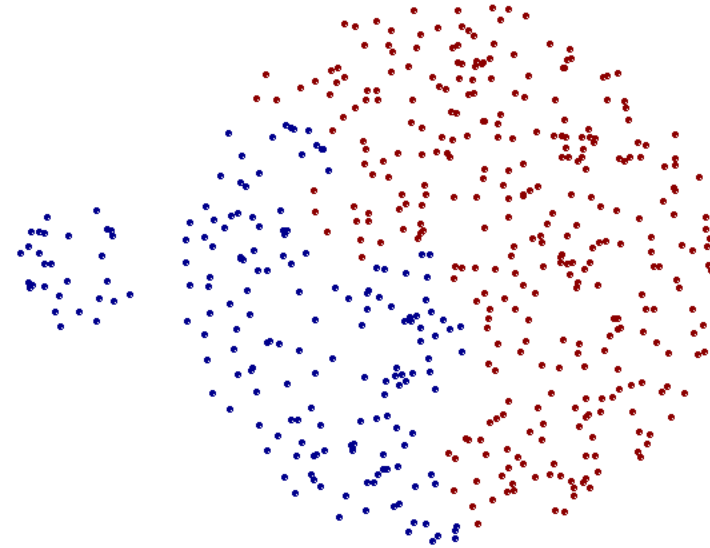
Two Clusters

- **Less susceptible to noise and outliers**

Limitations of MAX



Original Points



Two Clusters

- Tends to break large clusters
- Biased towards globular clusters

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

Hierarchical Clustering: Time and Space requirements

□ Space complexity?

A. $O(N^2)$

B. $O(N)$

C. $O(N \cdot \log(N))$

Hierarchical Clustering: Time and Space requirements

- Space complexity?

- A. $O(N^2)$

- B. $O(N)$

- C. $O(N \cdot \log(N))$

- Time complexity?

- ▣ N : the number of data points. $O(N^3)$ time in many cases, as 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

Clustering Algorithms

- K-means and its variants
- Hierarchical clustering
- **Density-based clustering**

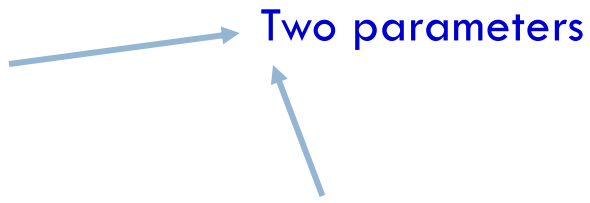
Density-Based Clustering Methods

- Clustering based on density (a local cluster criterion), such as density-connected points
- Major features:
 - ▣ Discover clusters of arbitrary shape
 - ▣ Handle noise
 - ▣ One scan (only examine the local region to justify density)
 - ▣ Need density parameters as termination condition
- Several interesting studies:
 - ▣ **DBSCAN**: Ester, et al. (KDD'96)
 - ▣ OPTICS: Ankerst, et al (SIGMOD'99)
 - ▣ DENCLUE: Hinneburg & D. Keim (KDD'98)
 - ▣ CLIQUE: Agrawal, et al. (SIGMOD'98) (also, grid-based)



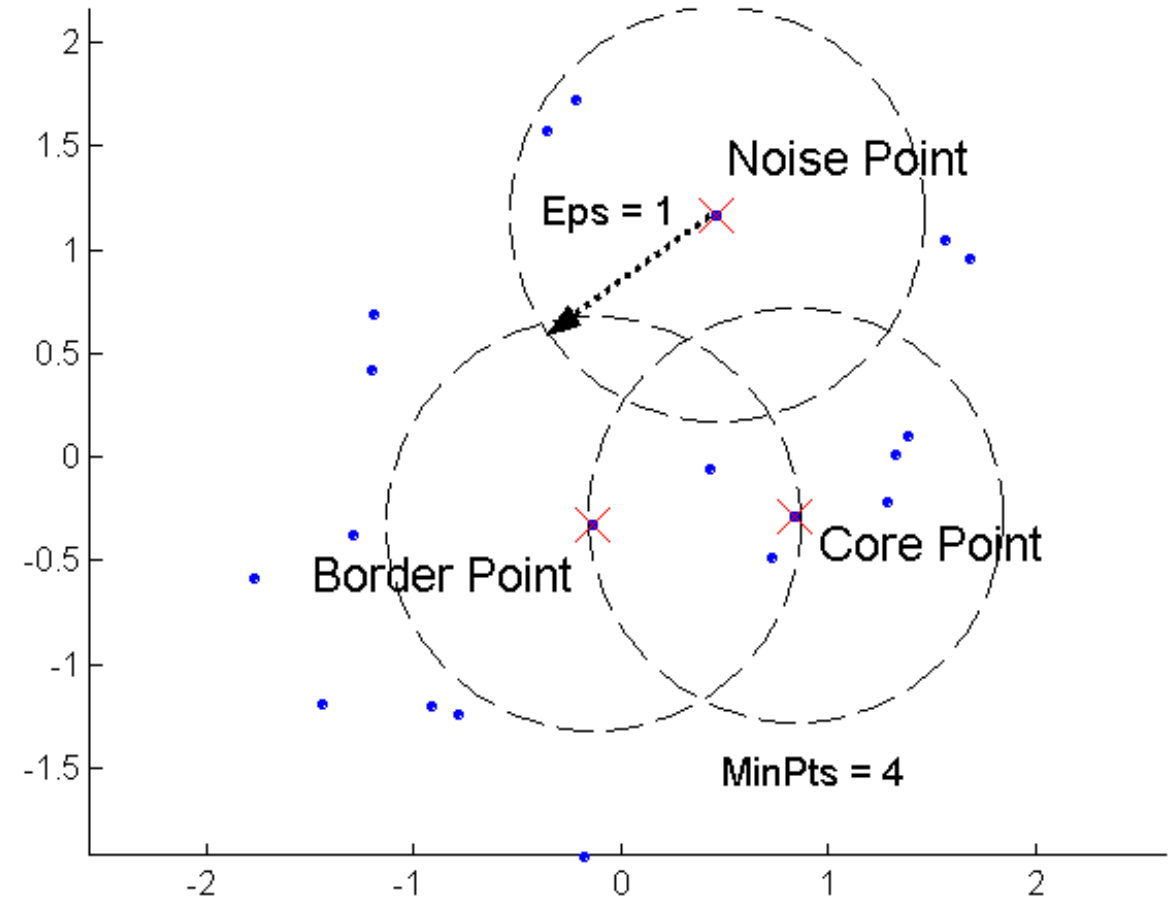
DBSCAN

- DBSCAN (M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, KDD'96)
 - ▣ Discovers clusters of arbitrary shape: Density-Based Spatial Clustering of Applications with Noise

 - DBSCAN is a density-based algorithm.
 - ▣ Density = number of points within a specified radius (**Eps**)
 - ▣ A point is a **core point** if it has at least 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

1. A point is a **core point** if it has at least a specified number of points (MinPts) within Eps
2. A **border point** has fewer than MinPts within Eps, but is in the neighborhood of a core point
3. A **noise point** is any point that is not a core point or a border point.



DBSCAN Algorithm

- Eliminate noise points
- Perform clustering on the remaining points

$current_cluster_label \leftarrow 1$

for all core points **do**

if the core point has no cluster label **then**

$current_cluster_label \leftarrow current_cluster_label + 1$

 Label the current core point with cluster label $current_cluster_label$

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 $current_cluster_label$

end if

end for

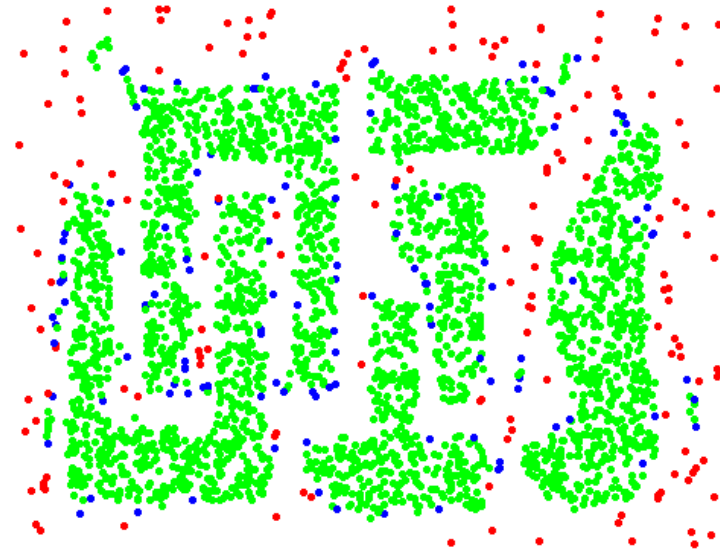
end for

The Eps -neighborhood of a point q :
 $N_{Eps}(q): \{p \text{ belongs to } D \mid \text{dist}(p, q) \leq Eps\}$

DBSCAN: Core, Border and Noise Points



Original Points



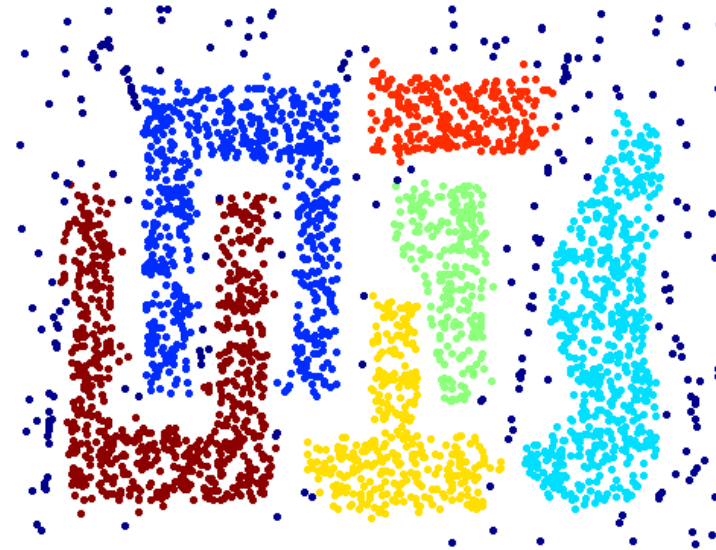
Point types: **core**, **border** and **noise**

Eps = 10, MinPts = 4

When DBSCAN Works Well



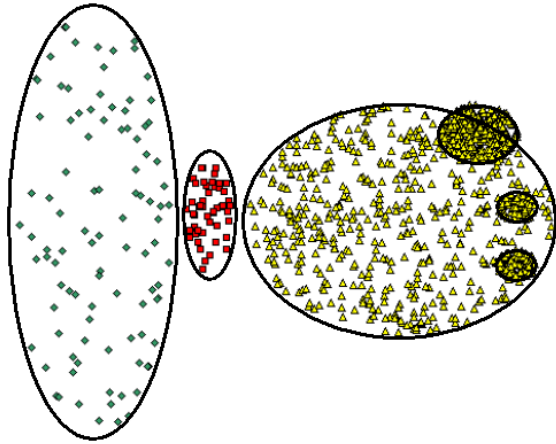
Original Points



Clusters

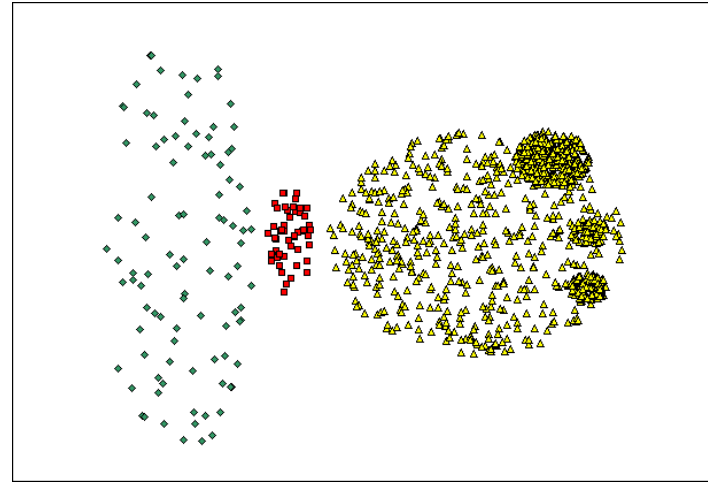
- Resistant to Noise
- Can handle clusters of different shapes and sizes

When DBSCAN Does NOT Work Well



Original Points

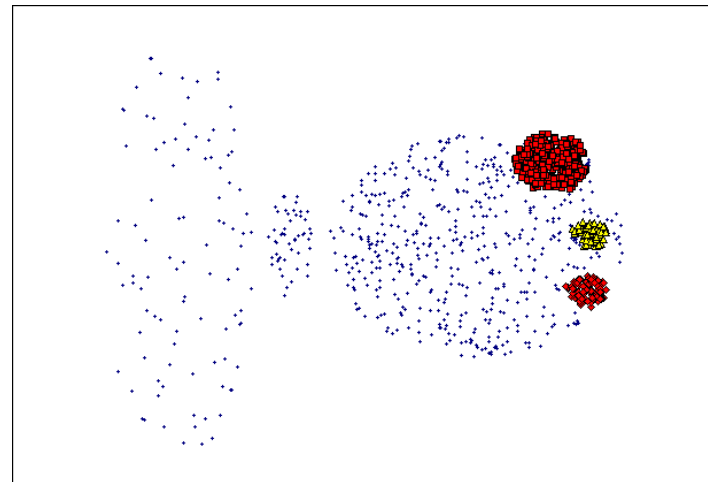
- Varying densities
- High-dimensional data



(MinPts=4, Eps=9.75).



Sensitive to parameters!



(MinPts=4, Eps=9.92)

Cluster Validity

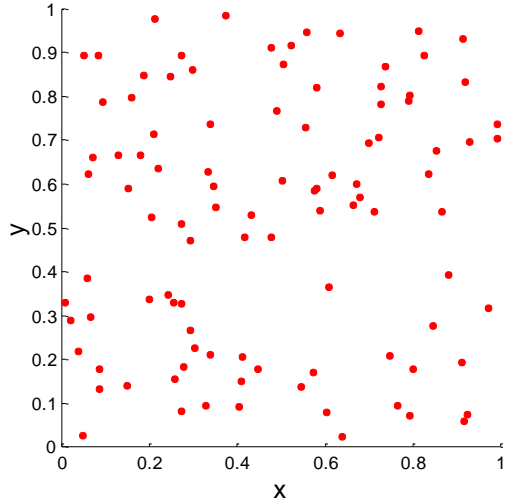
- For supervised classification we have a variety of measures to evaluate how good our model is
 - ▣ Accuracy, precision, recall
- For cluster analysis, the analogous question is **how to evaluate the “goodness” of the resulting clusters?**
 - **One measure mentioned before...**

Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is
 - Accuracy, precision, recall
- For cluster analysis, the analogous question is how to evaluate the “goodness” of the resulting clusters?
- But “clusters are in the eye of the beholder”!
- Then **why do we want to evaluate them?**
 - **To avoid finding patterns in noise**
 - **To compare clustering algorithms**
 - **To compare two sets of clusters**
 - **To compare two clusters**

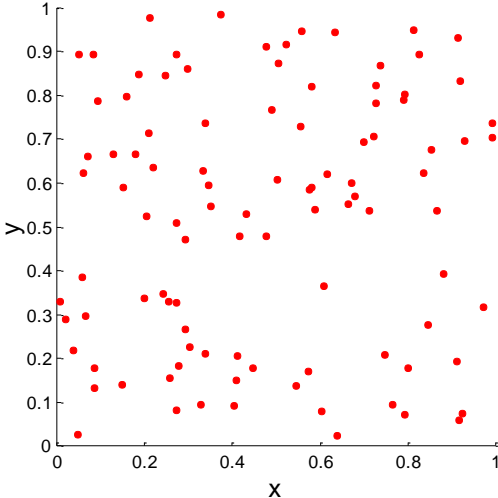
Clusters found in Random Data

Random Points

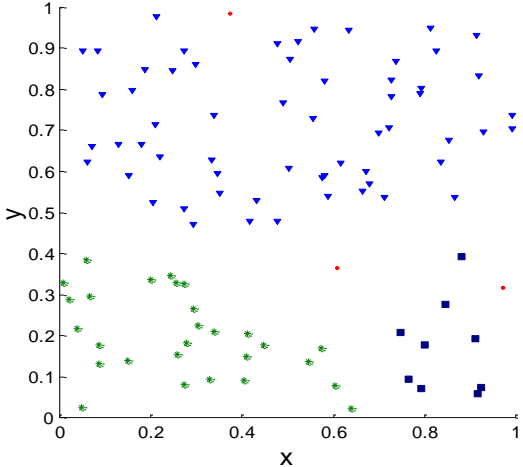


Clusters found in Random Data

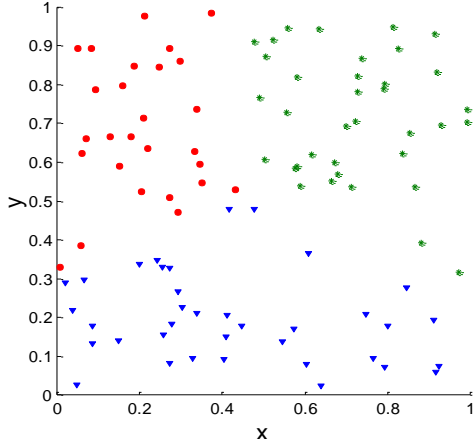
Random Points



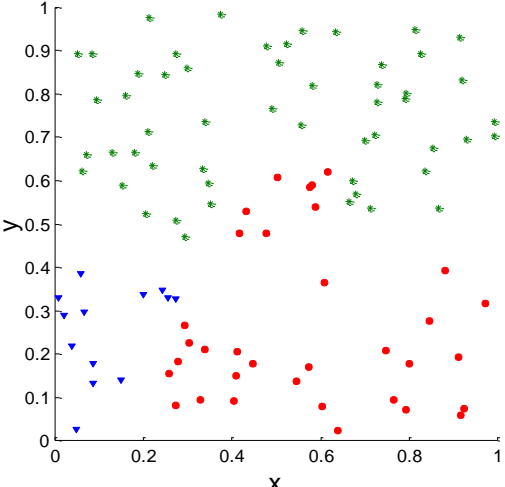
DBSCAN



K-means

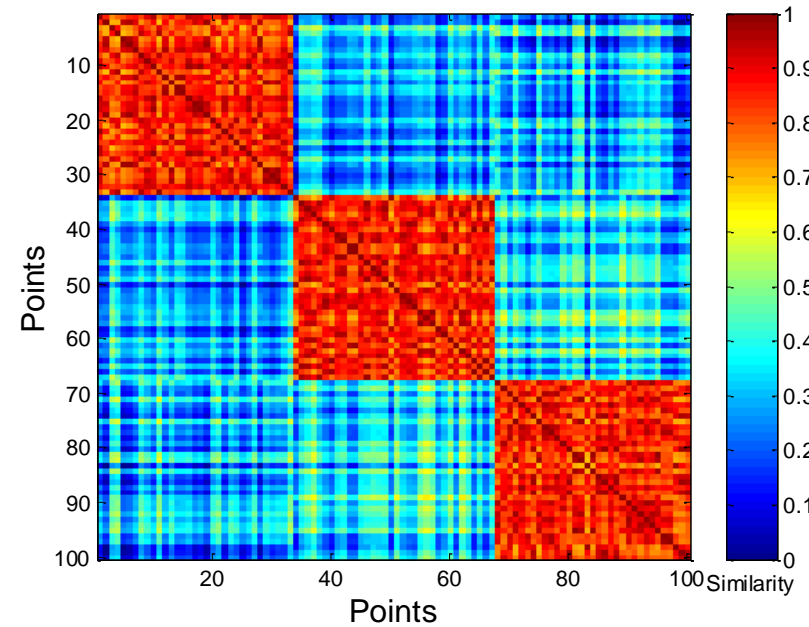
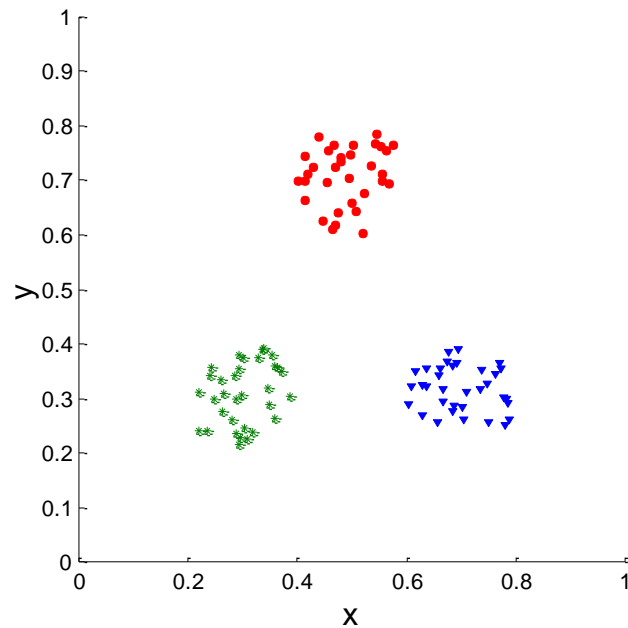


Complete Link



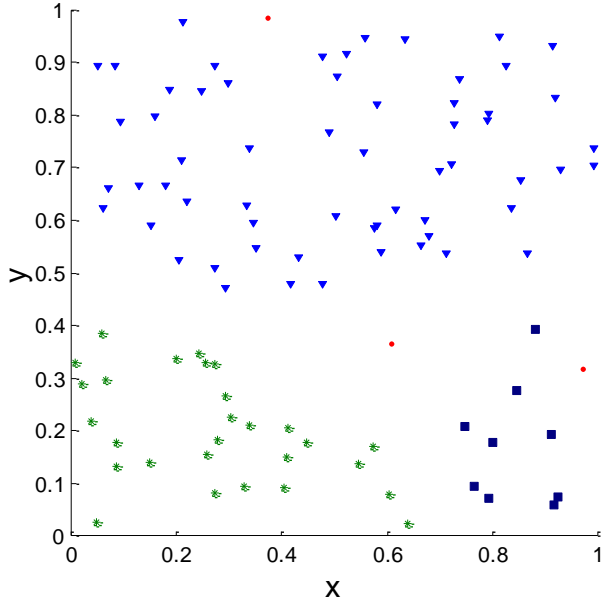
Using Similarity Matrix for Cluster Validation

- Order the similarity matrix with respect to cluster labels and inspect visually.



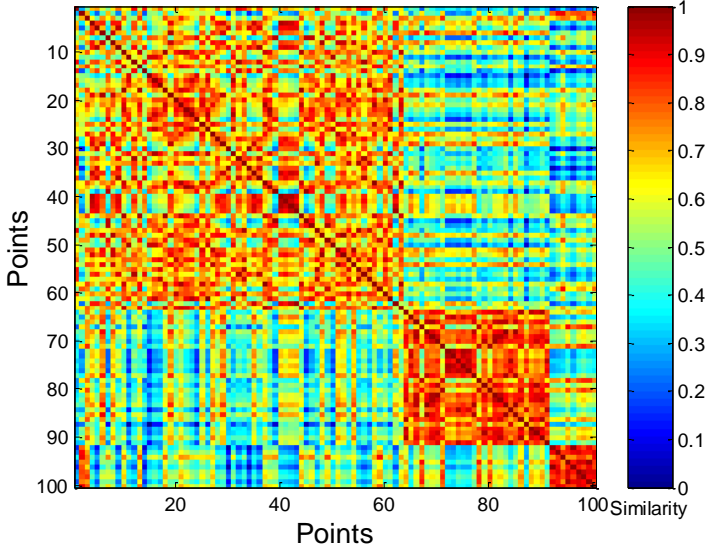
Using Similarity Matrix for Cluster Validation

□ Clusters in random data are not so crisp

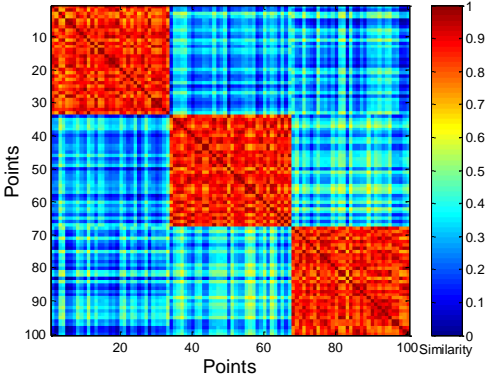


DBSCAN

Visualization
→



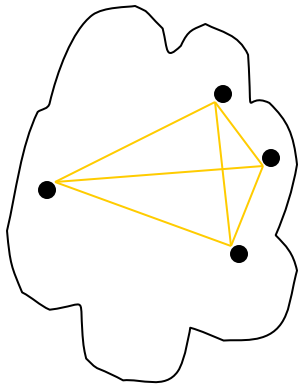
Visualization of Similarity Matrix



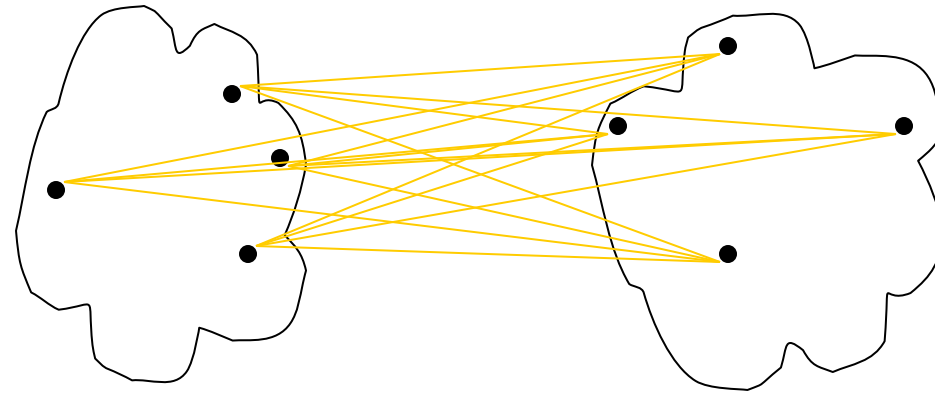
What good clustering results look like...

Cohesion and Separation

- A proximity graph based approach can also be used for cohesion and separation.
 - ▣ Cluster cohesion is the sum of the weight of all links within a cluster.
 - ▣ Cluster separation is the sum of the weights between nodes in the cluster and nodes outside the cluster.



cohesion

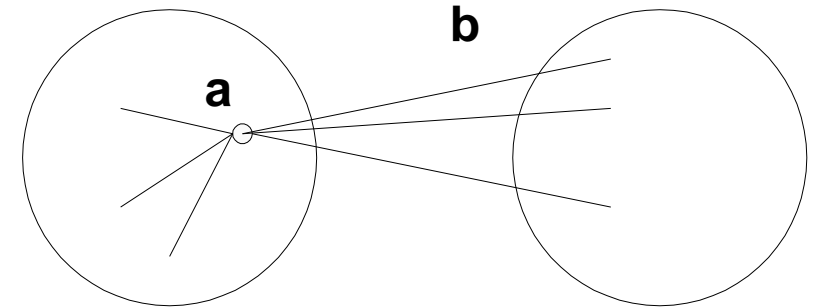


separation

Silhouette Coefficient

- Silhouette Coefficient **combine ideas of both cohesion and separation**, but for individual points, as well as clusters and clusterings
- For an individual point, i
 - Calculate a = average distance of i to the points in its cluster
 - Calculate b = min (average distance of i to points in another cluster)
 - The silhouette coefficient for a point is then given by

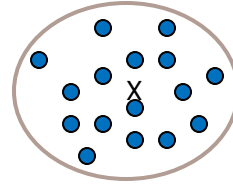
$$s = 1 - a/b \quad \text{if } a < b, \quad (\text{or } s = b/a - 1 \quad \text{if } a > b, \text{ not the usual case})$$



- Can calculate the Average Silhouette width for a cluster or a clustering

Measures of Cluster: Centroid, Radius and Diameter

- Centroid: \vec{x}_0
 - the “middle” of a cluster
 - n : number of points in a cluster
 - \vec{x}_i is the i -th point in the cluster



$$\vec{x}_0 = \frac{\sum_i^n \vec{x}_i}{n}$$

- Radius: R
 - Average distance from member objects to the centroid
 - The square root of average distance from any point of the cluster to its centroid

$$R = \sqrt{\frac{\sum_i^n (\vec{x}_i - \vec{x}_0)^2}{n}}$$

- Diameter: D
 - Average pairwise distance within a cluster
 - The square root of average mean squared distance between all pairs of points in the cluster

$$D = \sqrt{\frac{\sum_i^n \sum_j^n (\vec{x}_i - \vec{x}_j)^2}{n(n-1)}}$$

Other Measures of Cluster Validity

□ Entropy/Gini (Please review how to calculate it)

- If **there is a class label** – you can use the entropy/gini of the class label – similar to what we did for classification (Check problem III in sample midterm)
- If **there is no class label** – one can compute the entropy w.r.t each attribute (dimension) and sum up or weighted average to compute the disorder within a cluster

□ Classification Error

- If there is a class label one can compute this in a similar manner

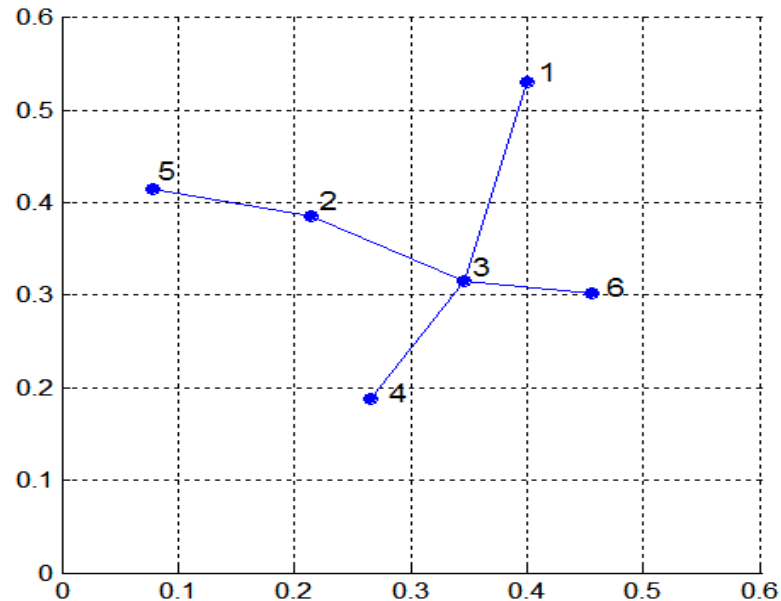
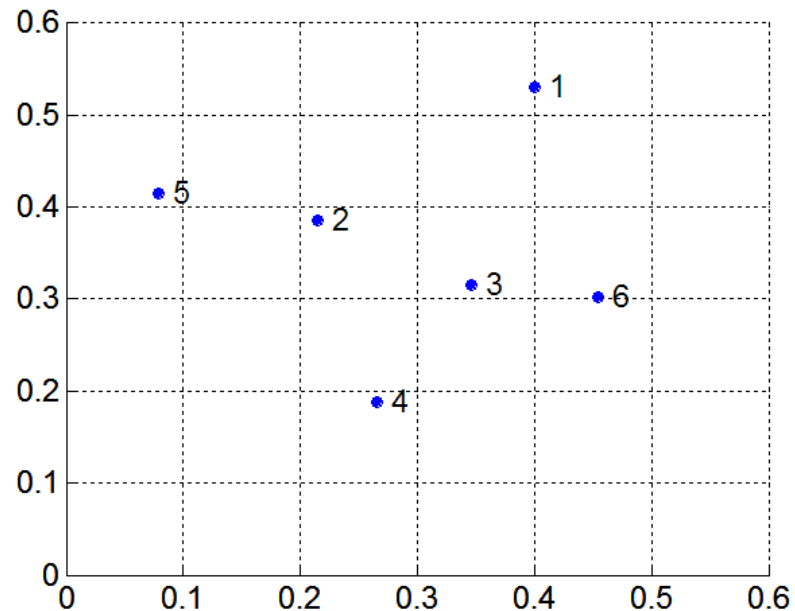
91

Backup slides

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
- Add q to the tree and put an edge between p and q



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
-