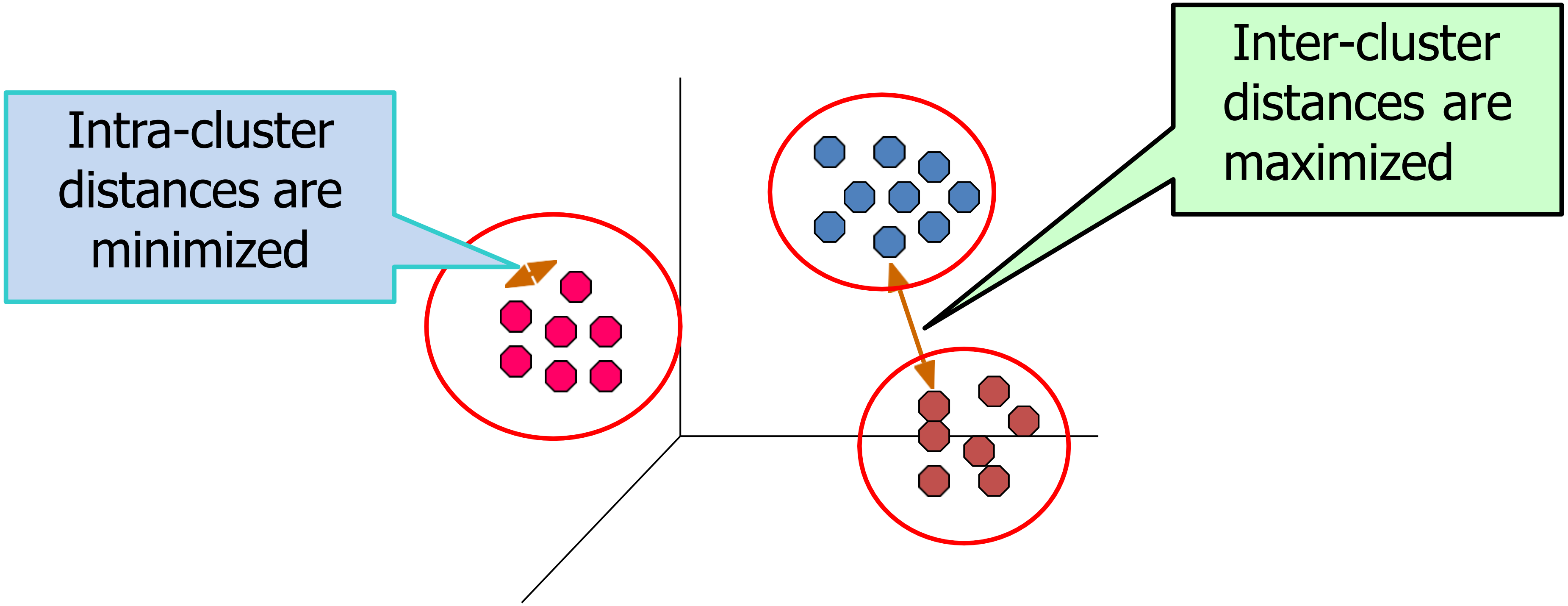
Ch. 10 - Clustering Analysis: Basic Concepts and Methods

Definition of Clustering

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



Input Data Types for Clustering

- Data matrix
  - (two modes)

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

- Dissimilarity matrix
  - (one mode)
  - Example:
    - Kernel matrix (for kernel clustering algorithms)
    - Adjacency or relationship matrix in social graph
  - Or you can compute the distance matrix from every pair of data points

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ : & : & : & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

Euclidean

2 2

2 1/2

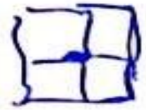
Minkowski w_1 p w_2 p

^ wp ^ p $1/p$

^

$p = 2 \rightarrow$ Euclidean — 

$p = 1 \rightarrow$ city-block — 

Uniformity — 

Classification → supervised

clustering → unsupervised

SPSS
SAS

→ scalability
→ different attribute types
(database records have mixed attribute types)
→ high dimensionality
→ order dependence vs. order independence

Types of clustering

i) Partitioning-based Clustering
ii) Hierarchical methods
iii) Density-based methods
iv) Grid-based methods.

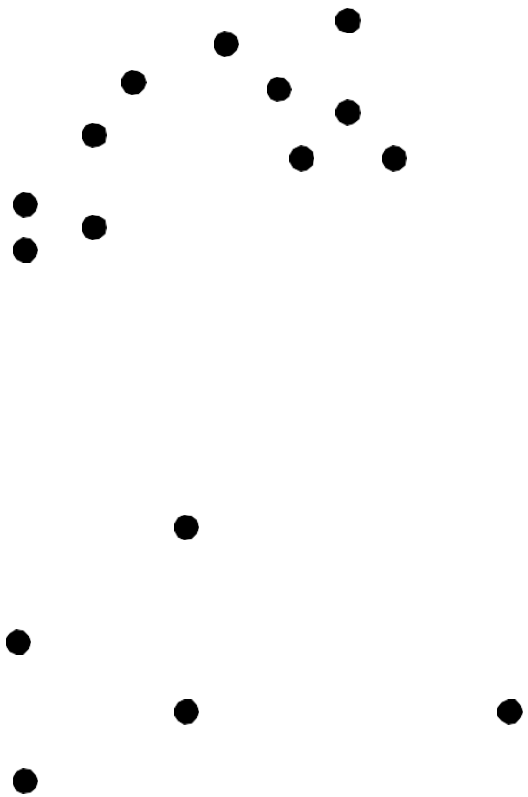
Types of Clustering

- Partitional Clustering
  - A **division** of data objects into non-overlapping subsets (clusters)
    - This division is called a partition
  - Can also be overlapping (soft clusters)
    - Fuzzy clustering
    - Most probabilstic/ model-based clustering

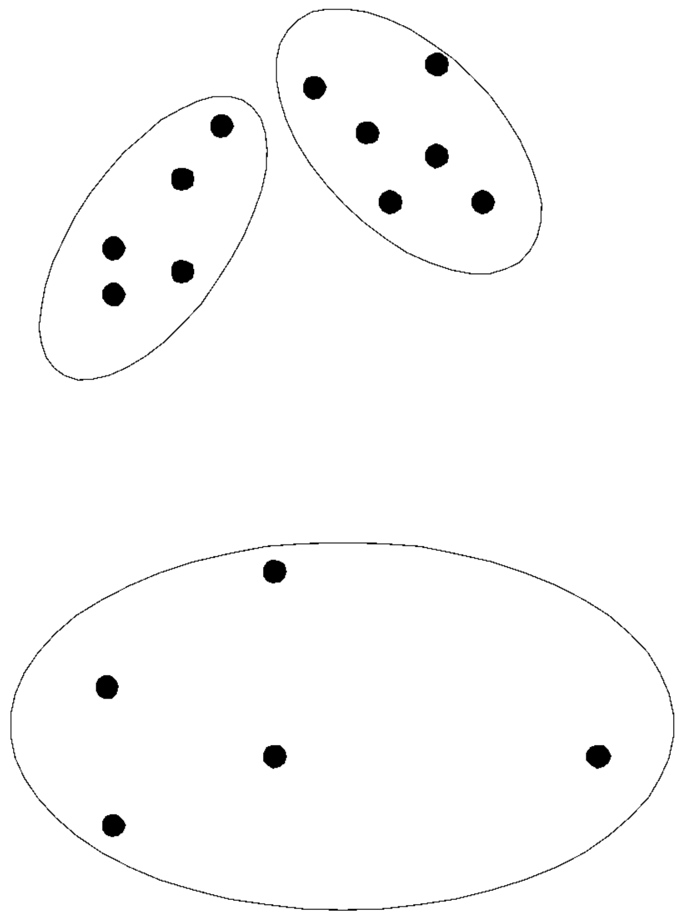
- Hierarchical clustering
  - A set of nested clusters organized as a **hierarchical tree**,
  - Tree is called dendogram

- Density-based clustering
  - A cluster is a **dense region** of points, which is **separated by low-density** regions, **from other regions** of high density.
  - Used when the clusters are irregular or intertwined, and when noise and outliers are present.

Partitional Clustering

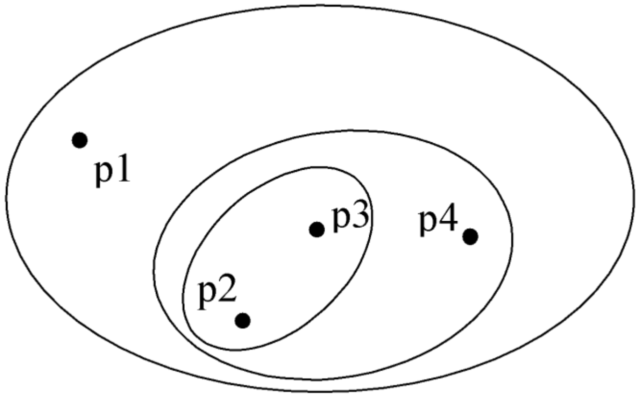


Original Points

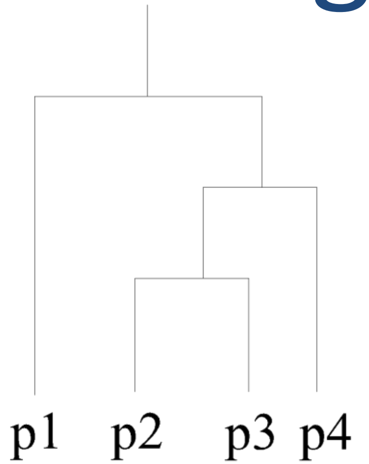


A Partitional Clustering

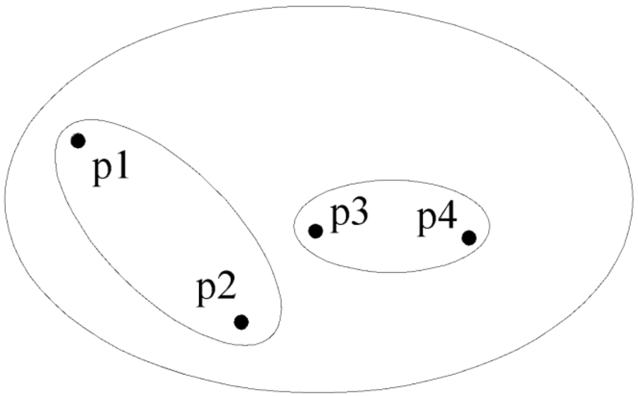
Hierarchical Clustering



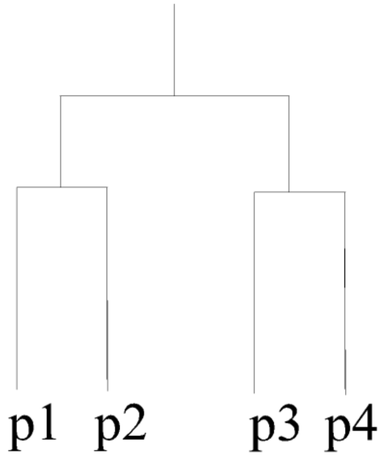
Traditional Hierarchical Clustering



Traditional Dendogram

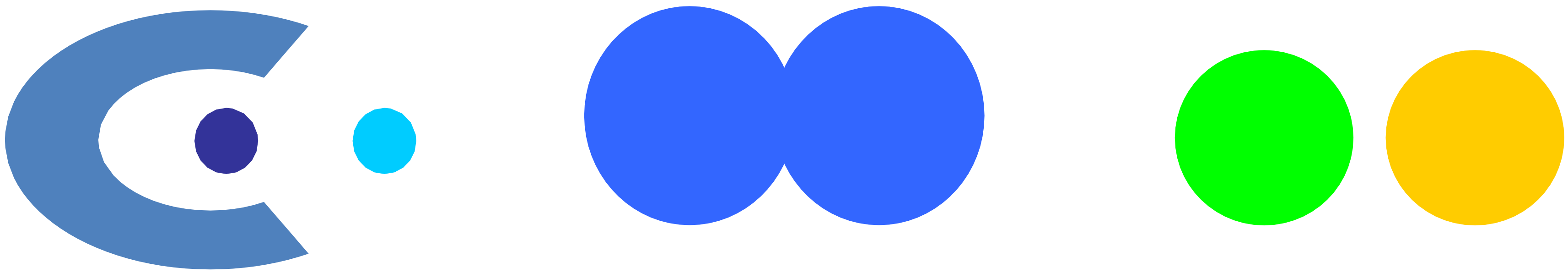


Non-traditional Hierarchical Clustering



Non-traditional Dendogram

Density Based Clusters



Partitional Clustering Methods: K-Means Algorithm

Algorithm: *k*-means. The *k*-means algorithm for partitioning based on the mean value of the objects in the cluster.

Input: The number of clusters *k* and a database containing *n* objects.

Output: A set of *k* clusters that minimizes the squared-error criterion.

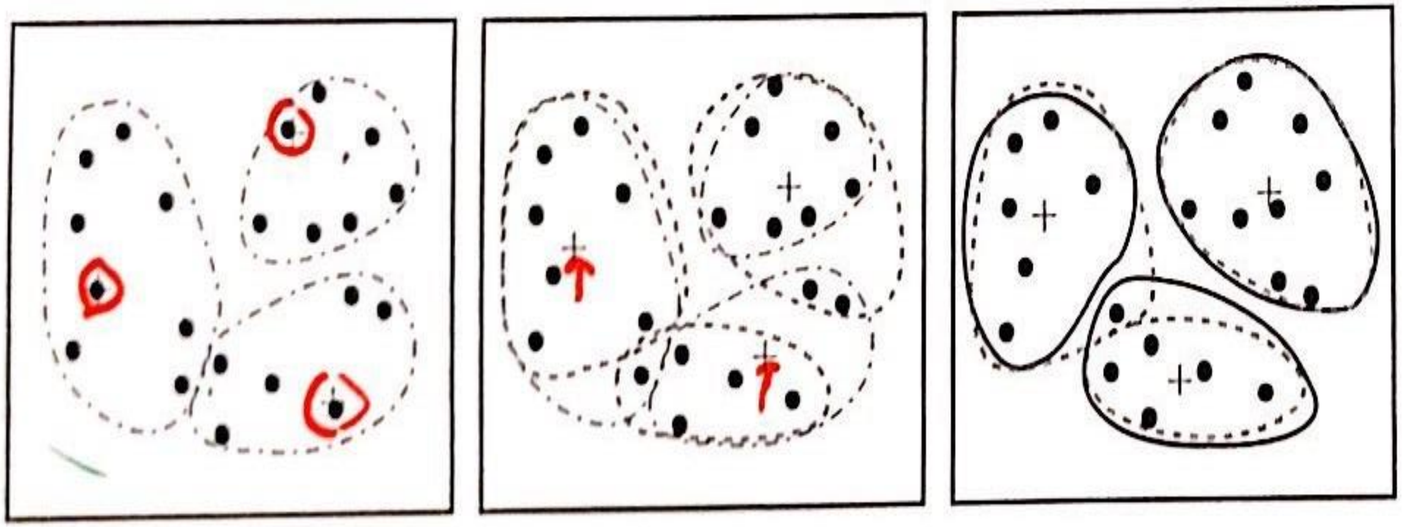
Method:

(1) arbitrarily choose *k* objects as the initial cluster centers;
(2) repeat
(3) (re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster;
(4) update the cluster means, i.e., calculate the mean value of the objects for each cluster;
(5) until no change;

7.2

Figure 8.1 The *k*-means algorithm.

$$E = \sum_{i=1}^{k} \left[\sum_{p \in C_i} |p - m_i|^2 \right] \quad 7.18$$



(a) (b) (c)

7.3

Figure ~~8.2~~ Clustering of a set of objects based on the *k*-means method. (The mean of each cluster is marked by a "+".)

$O(ktn)$

Partitional Clustering Methods:
K-Mediods Algorithm

Algorithm: k-**medoids.** A typical k-medoids algorithm for partitioning based on medoid or central objects.

Input: The number of clusters k and a database containing n objects.

Output: A set of k clusters that minimizes the sum of the dissimilarities of all the objects to their nearest medoid.

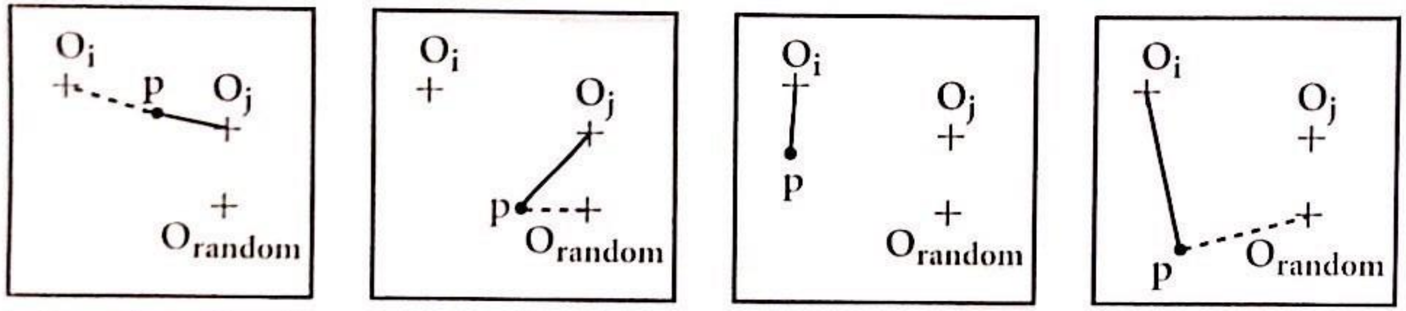
Method:

(1) arbitrarily choose k objects as the initial medoids;
(2) repeat
(3) assign each remaining object to the cluster with the nearest medoid;
(4) randomly select a nonmedoid object, o_{random};
(5) compute the total cost, S, of swapping o_j with o_{random};
(6) **if** $S < 0$ **then** swap o_j with o_{random} to form the new set of k medoids;
(7) until no change;

Figure ~~8.1~~ 7.5 The k-medoids algorithm.

$$E = \sum_{j=1}^{k} \sum_{p \in C_j} |p - o_j|$$ absolute error

$$O(k(n-k)^2)$$



1. Reassigned to O_i
2. Reassigned to O_{random}
3. No change
4. Reassigned to O_{random}

- • data object
- + cluster center
- — before swapping
- - - - after swapping

	p is currently closest to O_i	O_j
p is closest to O_i	3	1
closest to O_{random}	4	2

Fig. 7.4

3 Four cases of the cost function for k-medoids clustering.

should O_{random} take the place of O_j?

Hierarchical Clustering Methods

AGglomerative NESting Clustering Algorithm

- Most popular hierarchical clustering technique
- Basic AGNES 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

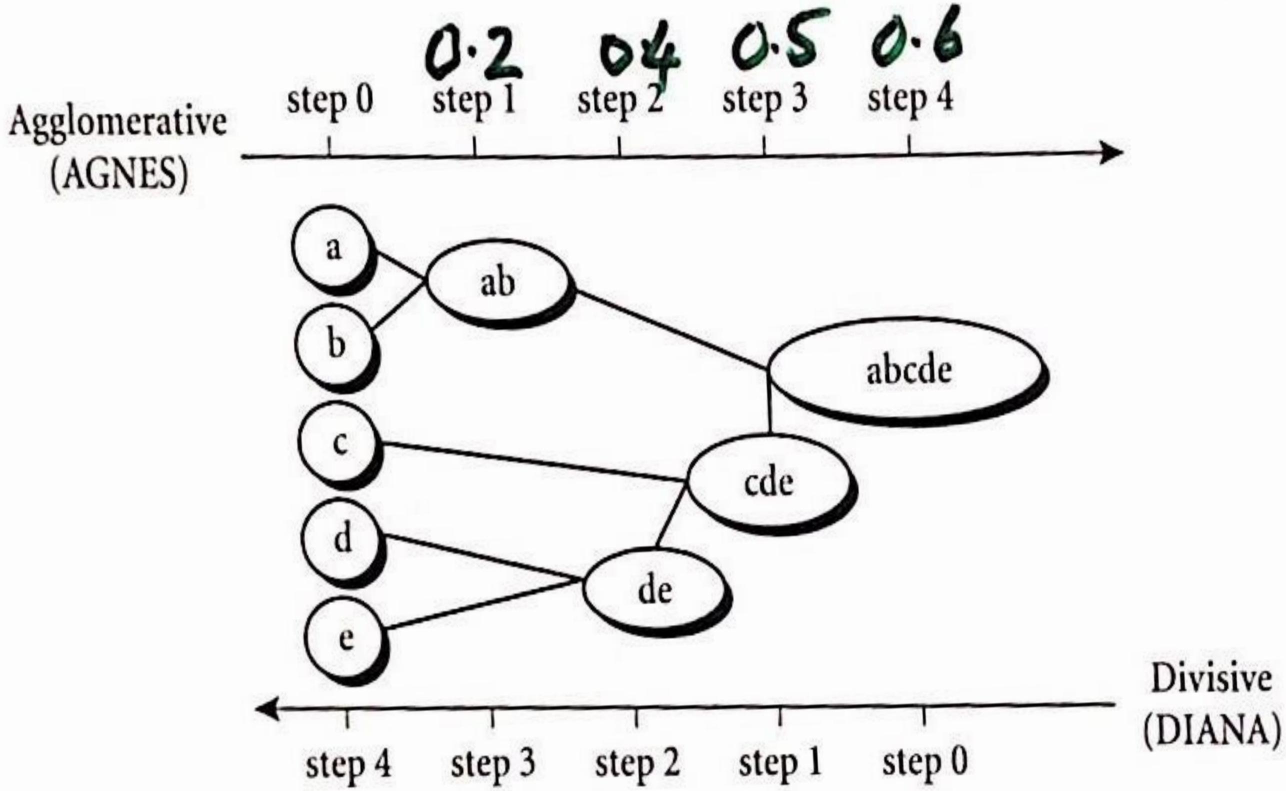
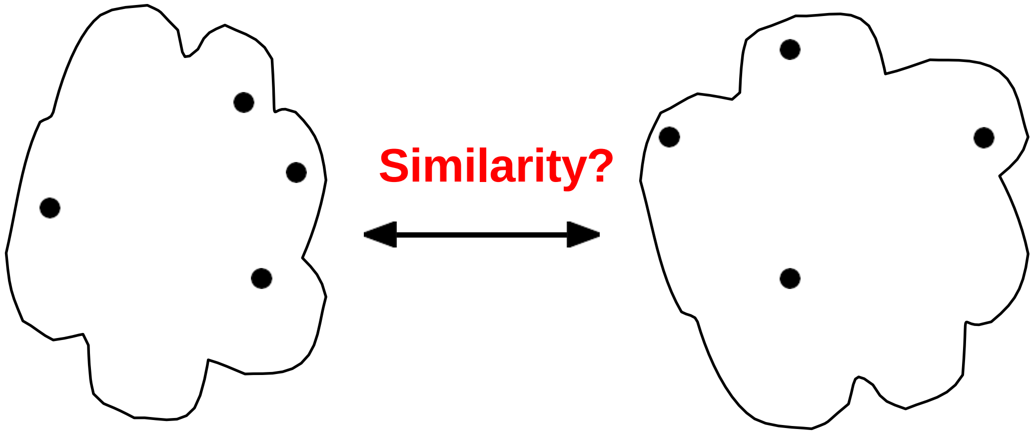


Figure 8.5 Agglomerative and divisive hierarchical clustering on data objects $\{a, b, c, d, e\}$.

How to Define Inter-Cluster Similarity

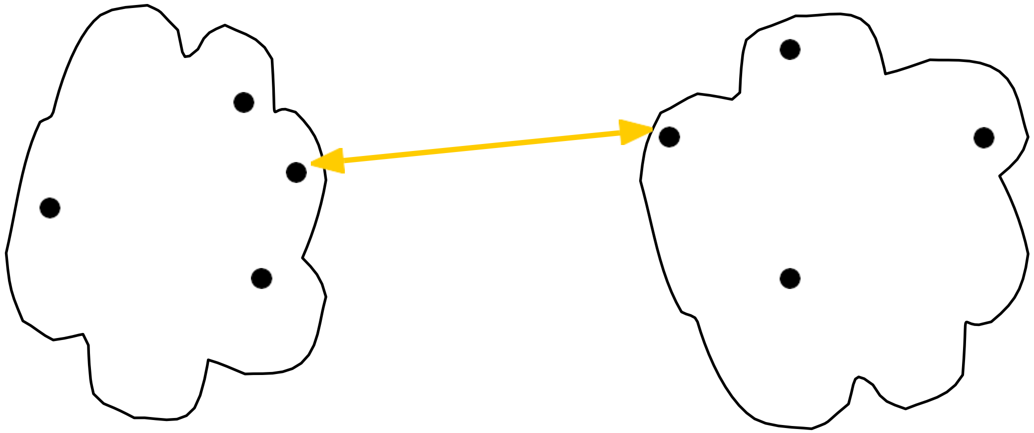


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

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

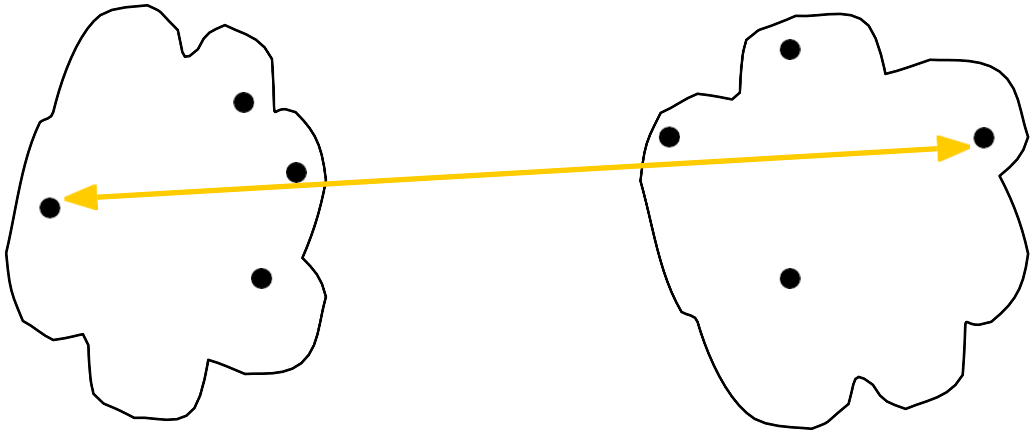


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

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

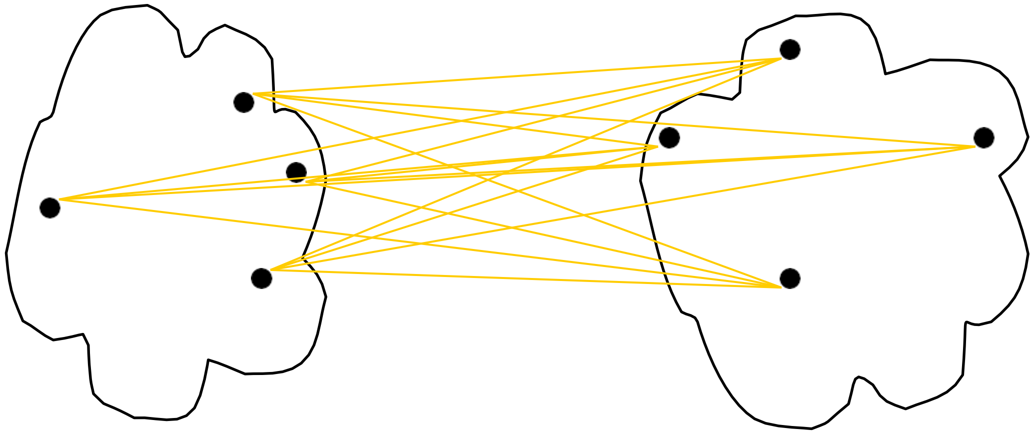


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

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

How to Define Inter-Cluster Similarity

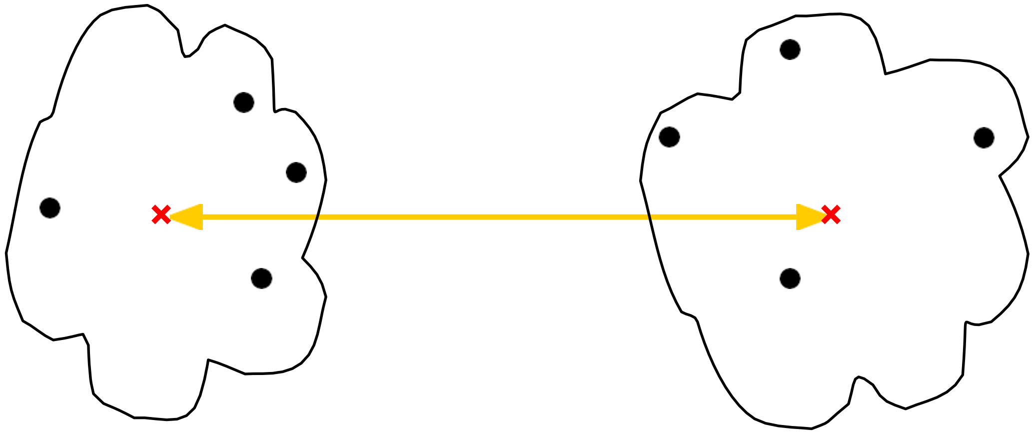


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

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

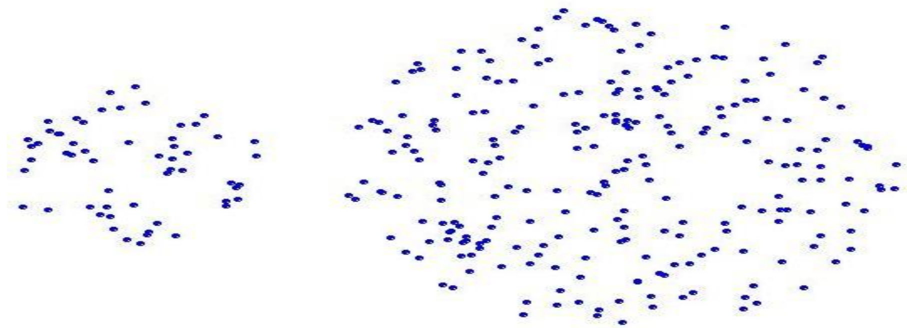


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

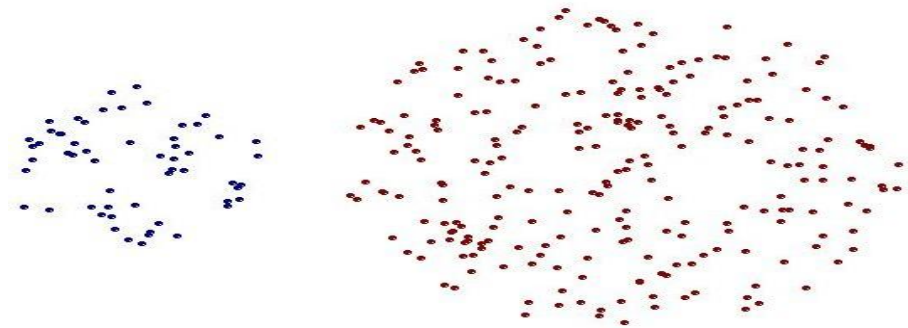
Proximity Matrix

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

Strength of MIN



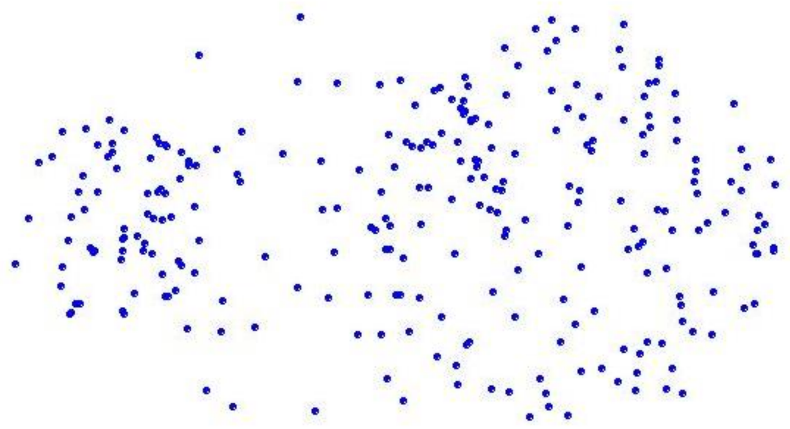
Original Points



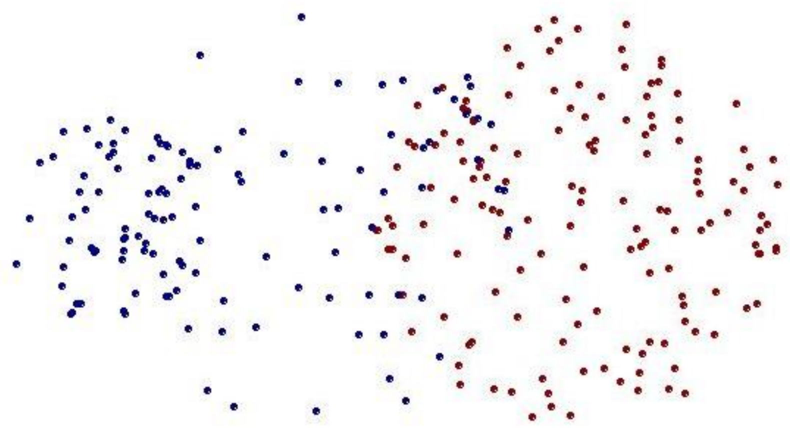
Two Clusters

- **Can handle non-elliptical shapes**

Limitations of MIN



Original Points



Two Clusters

- **Sensitive to noise and outliers**

The purpose of a dendrogram is to show the level at which two or more objects combine to form a common cluster. To illustrate, let us consider 5 objects whose object-object similarity matrix is as given below:

O_2	0.6			
O_3	0.4	0.8		
O_4	0.1	0.5	0.7	
O_5	0.1	0.2	0.2	0.3
	O_1	O_2	O_3	O_4

Suppose that the clusters corresponding to a given threshold are defined (borrowing a graph theoretic terminology) as the connected components (CC's) of the associated graph. Then, the dendrogram for this situation is as shown in Figure 2. In a dendrogram the abscissa has no particular meaning. The ordinate, on the other hand, represents similarity values. In the example given, O_2 and O_3 join at level 0.8, O_4 combines with O_2 and O_3 at level 0.7, O_1 combines with O_2, O_3 and O_4 at level 0.6 and, finally, all the objects form a single cluster at level 0.3.

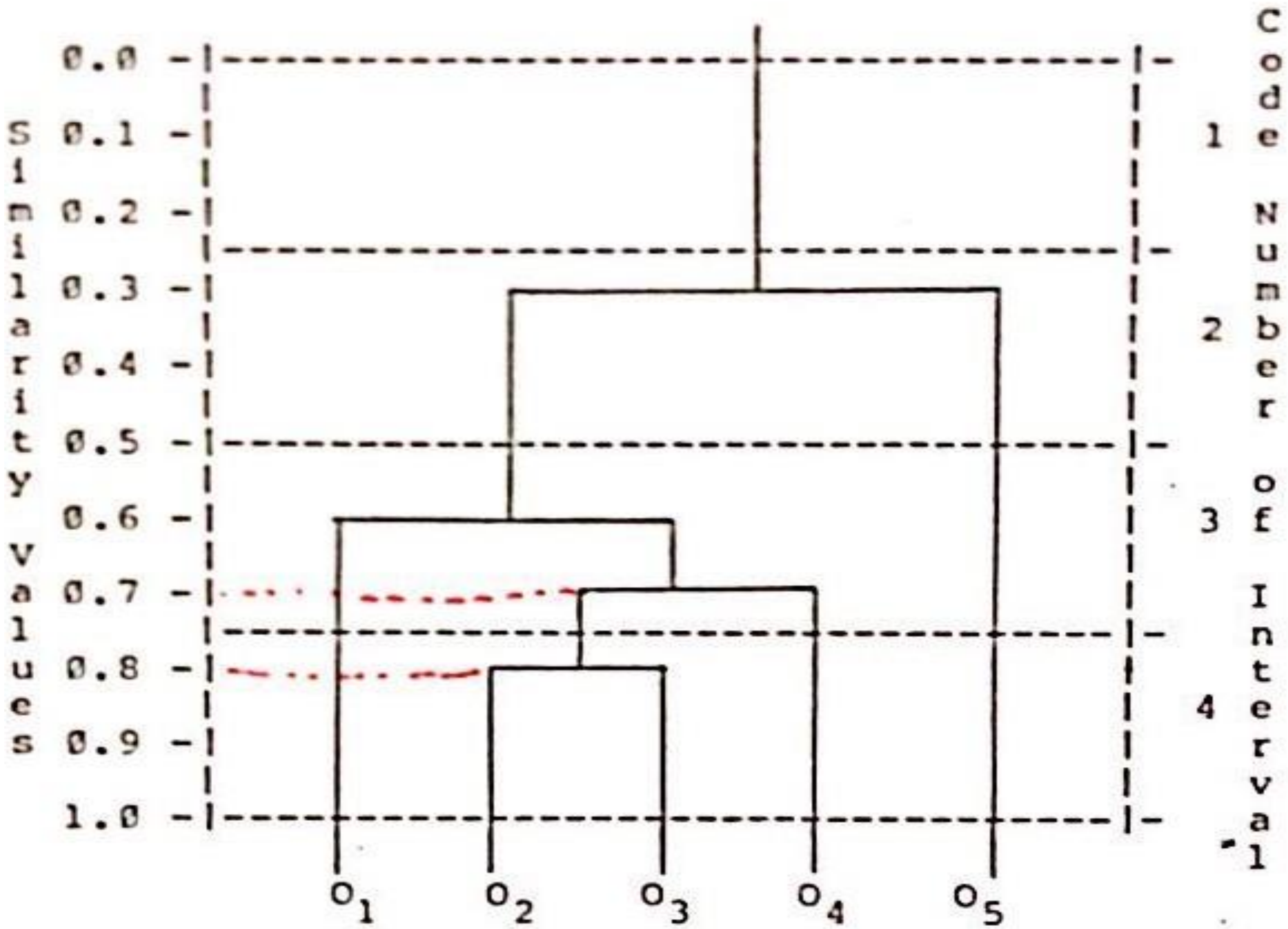
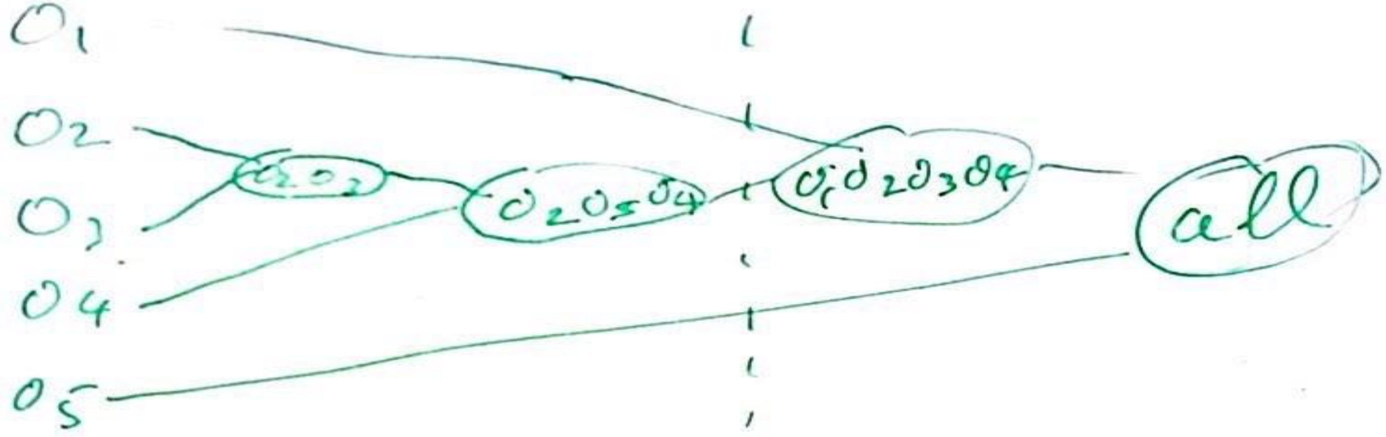


FIGURE 2. A dendrogram to illustrate the computation of cophenetic values.

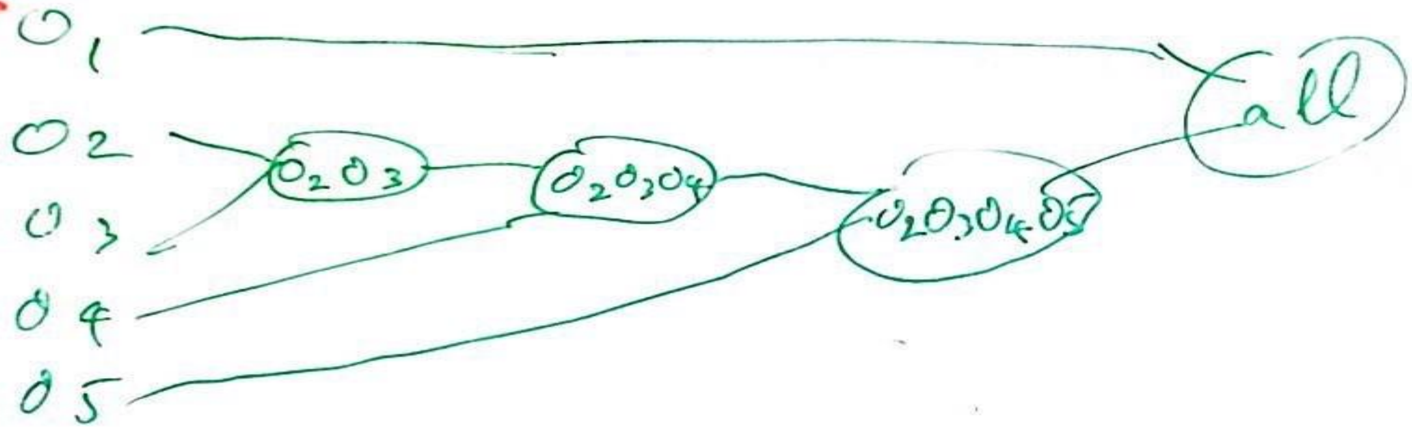
d_{min} ⇒ Similarities 0.8 0.7 0.6 0.3

Smax



d_{max} 0.8 0.5 0.2 0.1

Smin



Save / ave: 0.8 0.6 0.36 0.2

Evaluation of Clustering

Measures of Cluster Validity

- Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following three types.
  - External Index: Used to measure the extent to which cluster labels match externally supplied class labels.
    - Entropy
  - Internal Index: Used to measure the goodness of a clustering structure *without* external information.
    - E.g. Sum of Squared Error (SSE)
  - Relative Index: Used to compare two different clusterings or clusters.
    - Often an external or internal index is used for this function, e.g., SSE or entropy

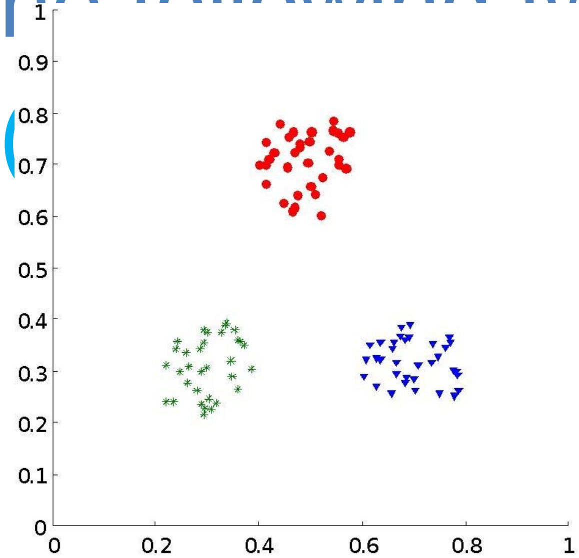
Measuring Cluster Validity Via Correlation

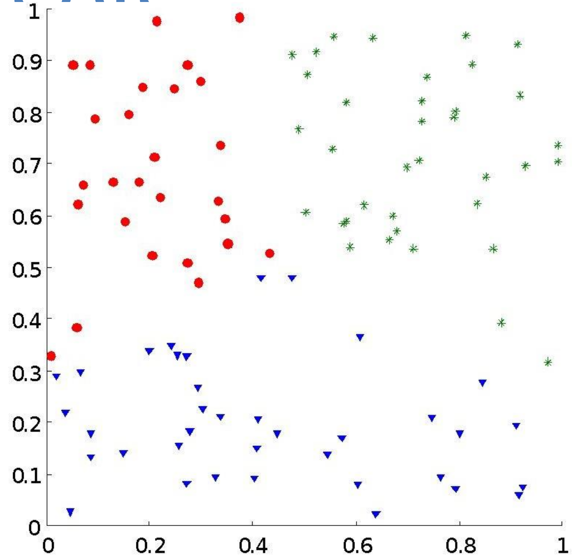
- Two matrices
  - Proximity Matrix (P(i,j) = similarity or distance between pt i and pt j)
  - "Incidence" Matrix (I(i,j) =1 if in same cluster)
    - One row and one column for each data point
    - An entry is 1 if the associated pair of points belong to the same cluster
    - An entry is 0 if the associated pair of points belongs to different clusters
- Compute the correlation between the two matrices P and I
  - Since the matrices are symmetric, only the correlation between n(n-1) / 2 entries needs to be calculated.
- High correlation indicates that points that belong to the same cluster are close to each other.
- Not a good measure for some density or contiguity based clusters.

Measuring Cluster Validity Via Correlation

- Correlation of incidence and proximity matrices for the K-means clusterings of the following two data sets.

- (



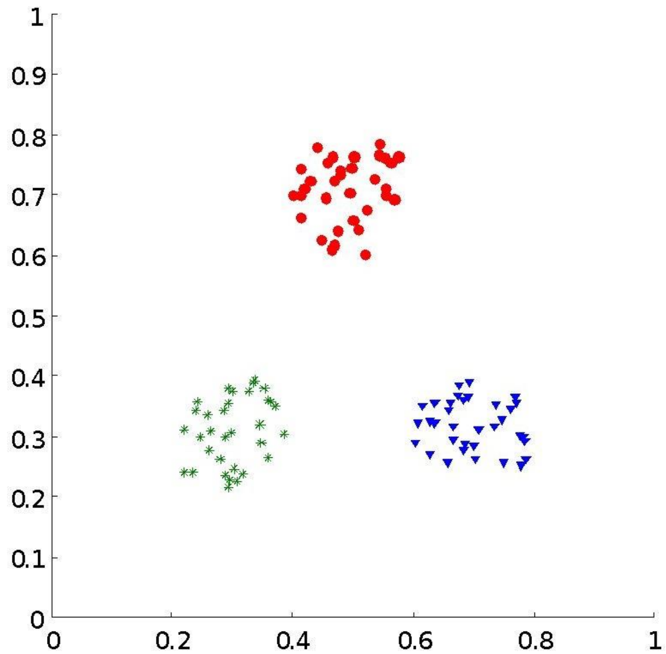


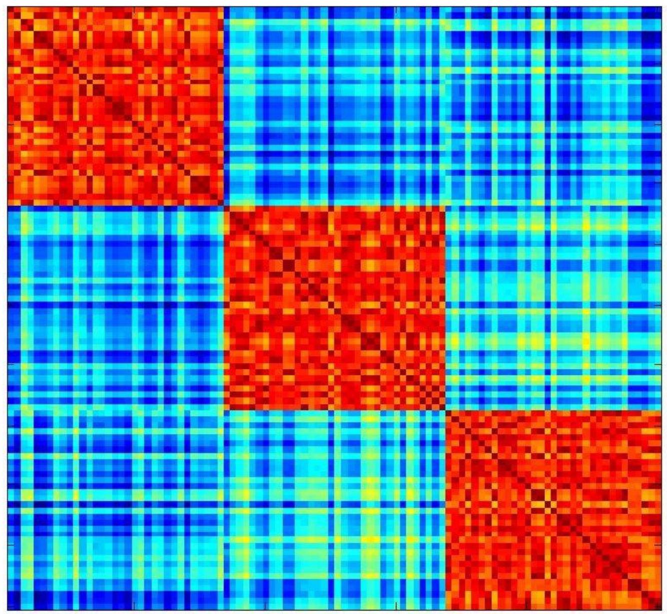
Corr = -0.9235 **(close to -1 because distance instead of similarity was used as proximity)**

Corr = -0.5810 **(closer to 0)**

Using Similarity Matrix for Cluster Validation

- Order the similarity matrix with respect to cluster labels and inspect visually. ➔ you see clusters/**<u>blocks</u>** in matrix





Validation

- http://scikit-learn.org/stable/modules/clusteri ng.html#clustering-evaluation
  - Several cluster validation metrics including
    - RAND index,
    - Silhouette,
    - Mutual Information Scores,
    - Homogeneity, Compactness, V-Measure
  - Gap statistic also available

Application Examples

- Marketing: Discover groups of customers with similar purchase patterns or similar demographics
- Land use: Find areas of similar land use in an earth observation database
- City-planning: Identify groups of houses according to their house type, value, and geographical location
- Web Usage Profiling: find groups of users with similar usage or interests on a website
- Document Organization: find groups of documents about similar topics
- **Summarization or Data Reduction:** Reduce size of large data sets
  - **can be better than random sampling**
  - 1) Cluster into large # of clusters
  - 2) then use cluster centroids
- **Discretize Numerical attributes:**
  - 1) Cluster numerical attribute into partitions
  - 2) Then consider each partition (group) ⇔ 1 categorical value
- **Imputate Missing Values :**
  - Cluster attribute values
  - Replace missing value with cluster center

1. What are Outliers?

- Outlier is a pattern in the data that does not conform to the expected behavior
- Also referred to as anomalies, exceptions, peculiarities, discordant observations, aberrations, surprises or contaminants
- Outliers translate to significant (often critical) real life entities
  - Cyber intrusions
  - Credit card fraud

36

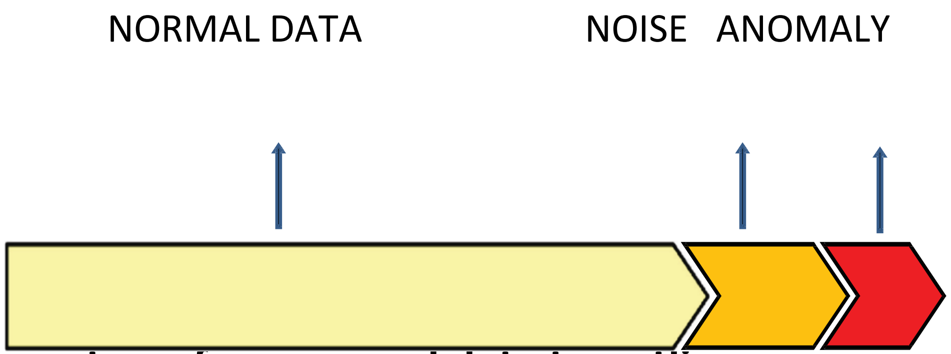
Real World Outliers

- Credit Card Fraud
  - An abnormally high purchase made on a credit card

- Cyber Intrusions
  - A web server involved in *ftp* traffic

More Formal Terminology

- Weak or Strong Outliers
- Increasing Outlierness Score from left to right



The spectrum from normal data to outliers

Outlier Detection = Anomaly Detection + Noise Removal

38

2. Key Challenges

- Defining a representative normal region is challenging
- The boundary between normal and anomalous behavior is often not precise
- The exact notion of an anomaly is different for different application domains
- Availability of labeled data for training/validation
- Multiple generating mechanisms (for both normal and anomalous instances)
- Normal behavior keeps evolving (Malicious adversary)

39

3. Type of Anomalies

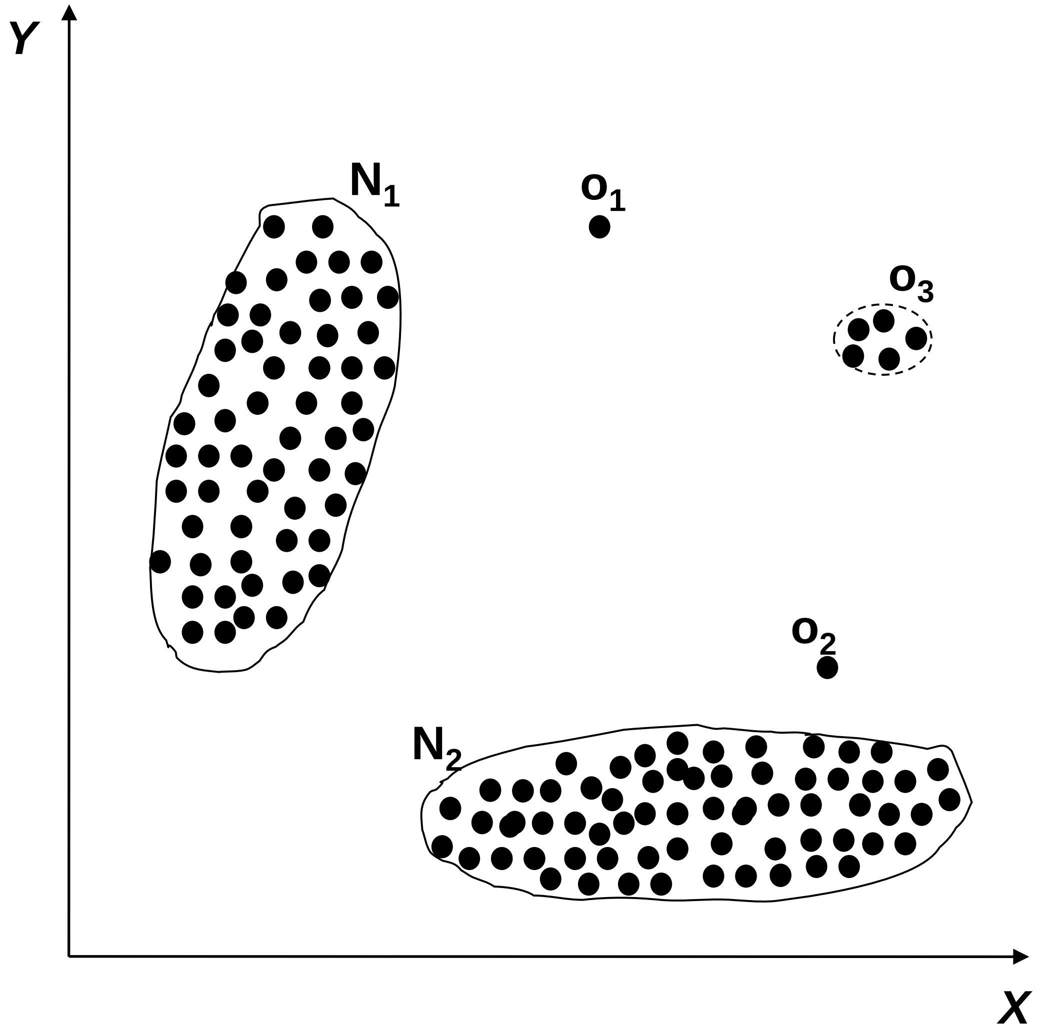
- Point Anomalies
- Contextual Anomalies
- Collective Anomalies

40

3.1 Point Anomalies

o1 and **o2** represent point anomalies

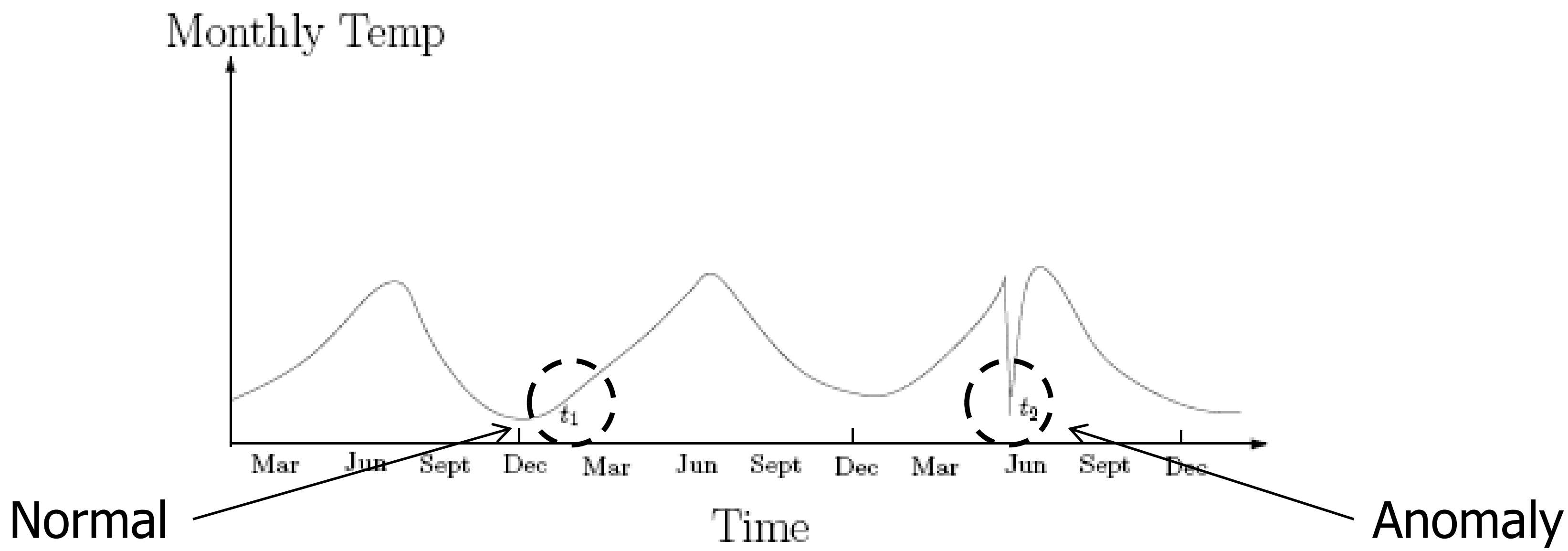
Region **o3** also contains point anomalies



41

3.2 Contextual Anomalies

- A point anomaly, but within a context
- Requires a notion of context
- Also referred to as ***conditional*** anomalies*

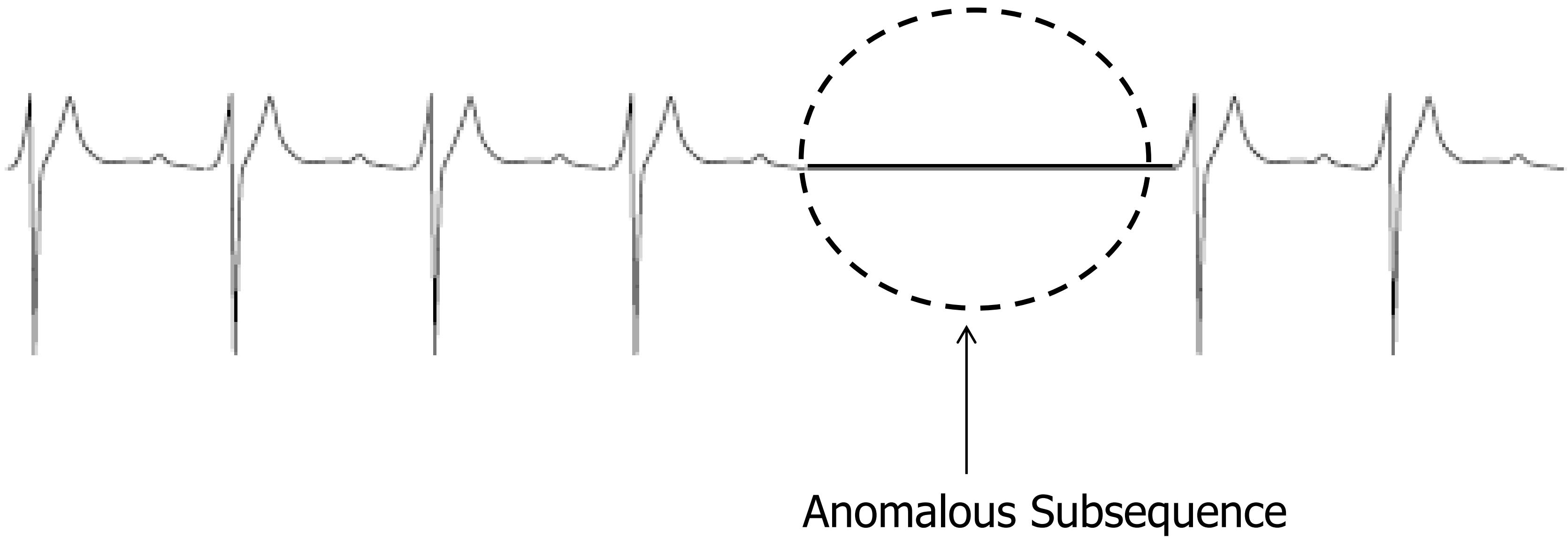


* Xiuyao Song, Mingxi Wu, Christopher Jermaine, Sanjay Ranka, Conditional Anomaly Detection, IEEE Transactions on Data and Knowledge Engineering, 2006.

42

3.3 Collective Anomalies

- A collection of related data instances is anomalous
- Requires a relationship among data instances
  - Sequential Data
  - Spatial Data
  - Graph Data
- The *individual instances* within a collective anomaly are *not* anomalous by themselves



43