

Clustering Analysis Basics

Ke Chen

Outline

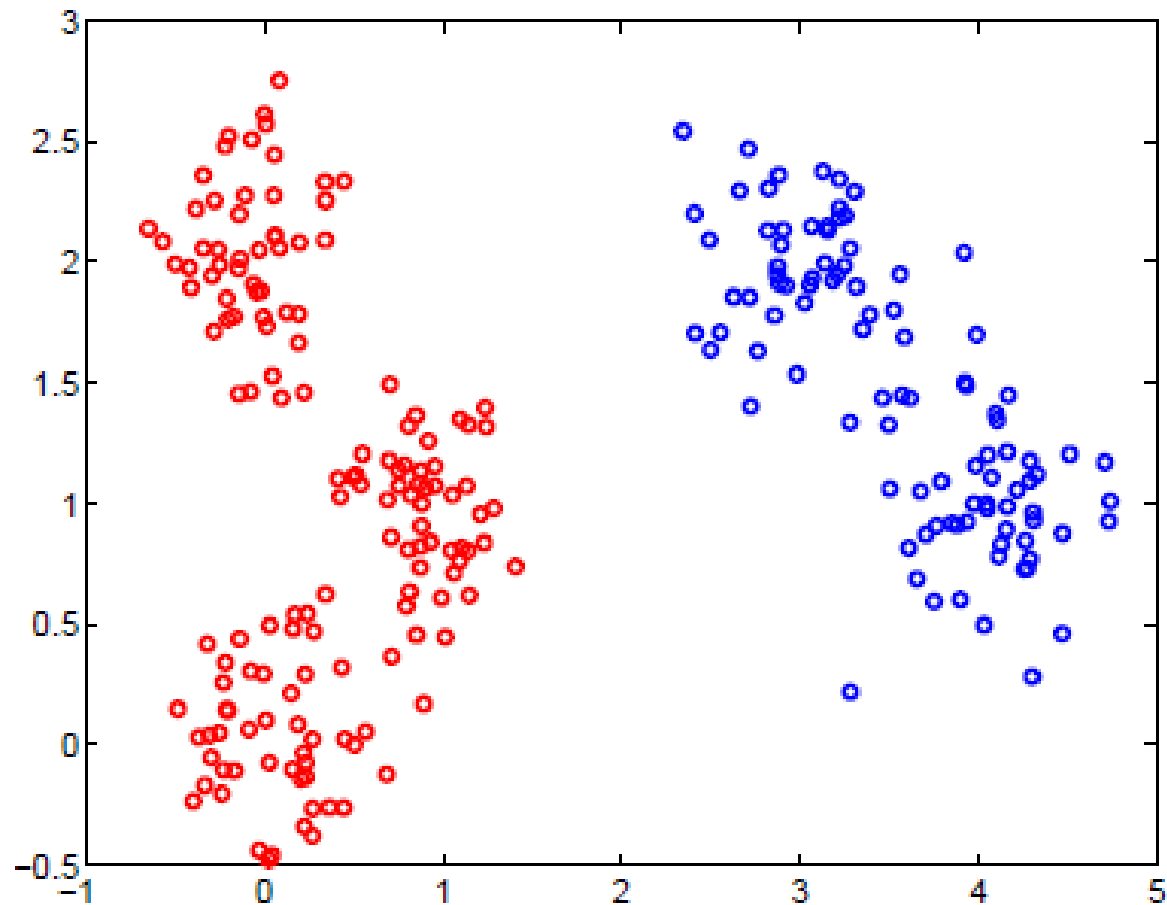
- Introduction
- Data Types and Representations
- Distance Measures
- Major Clustering Approaches
- Summary

Introduction

- Cluster: A collection/group of data objects/points
 - similar (or related) to one another within the same group
 - dissimilar (or unrelated) to the objects in other groups
- Cluster analysis
 - find *similarities* between data according to characteristics underlying the data and grouping similar data objects into clusters
- Clustering Analysis: Unsupervised learning
 - no predefined classes for a training data set
 - Two general tasks: identify the “natural” clustering number and properly grouping objects into “sensible” clusters
- Typical applications
 - as a stand-alone tool to gain an insight into data distribution
 - as a preprocessing step of other algorithms in intelligent systems

Introduction

- Illustrative Example 1: how many clusters?



Introduction

- Illustrative Example 2: are they in the same cluster?

Blue shark,
sheep, cat,
dog

Lizard, sparrow,
viper, seagull, gold
fish, frog, red
mullet

1. Two clusters
2. Clustering criterion:
How animals bear
their progeny

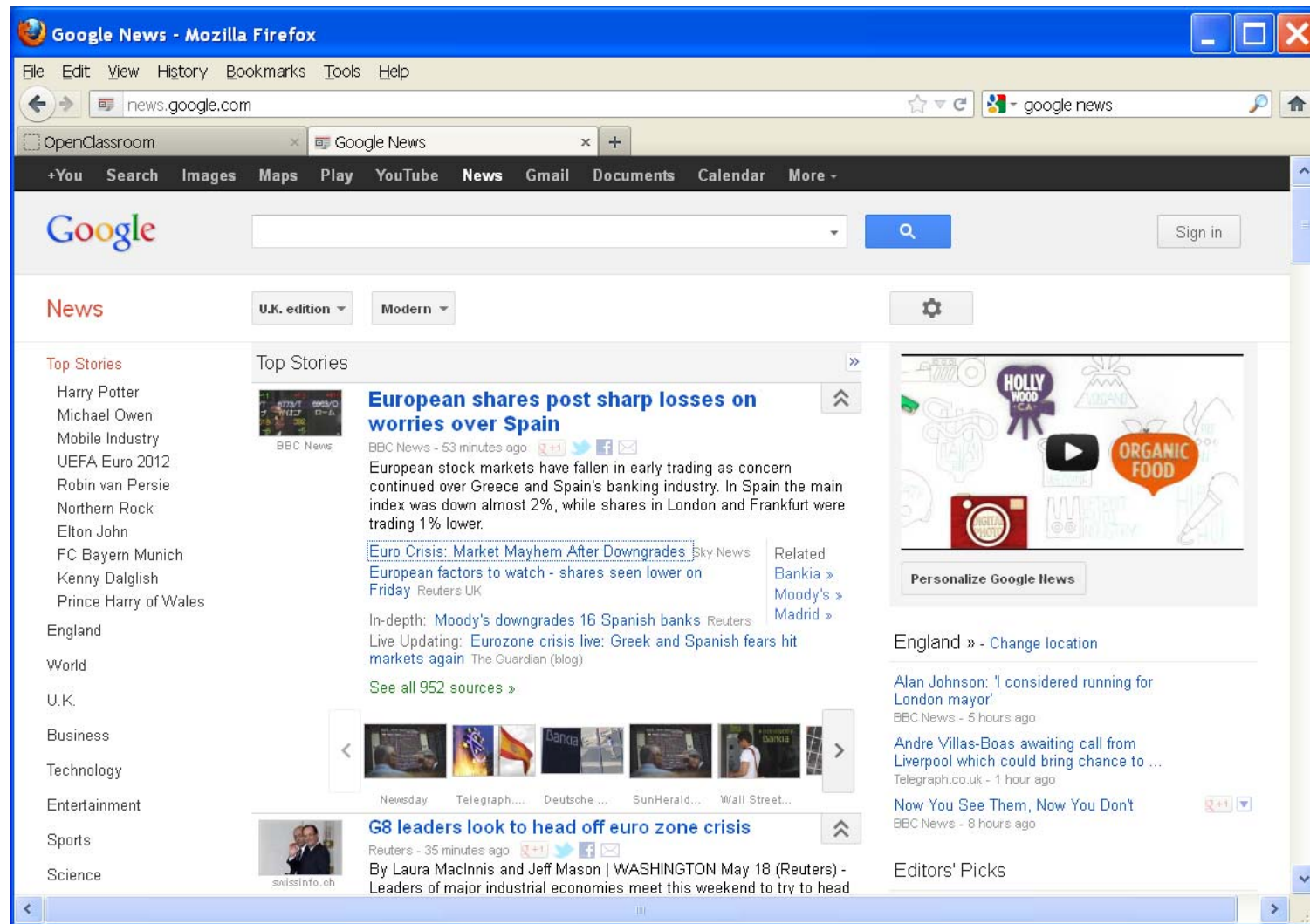
Gold fish, red
mullet, blue
shark

Sheep, sparrow,
dog, cat, seagull,
lizard, frog, viper

1. Two clusters
2. Clustering criterion:
Existence of lungs

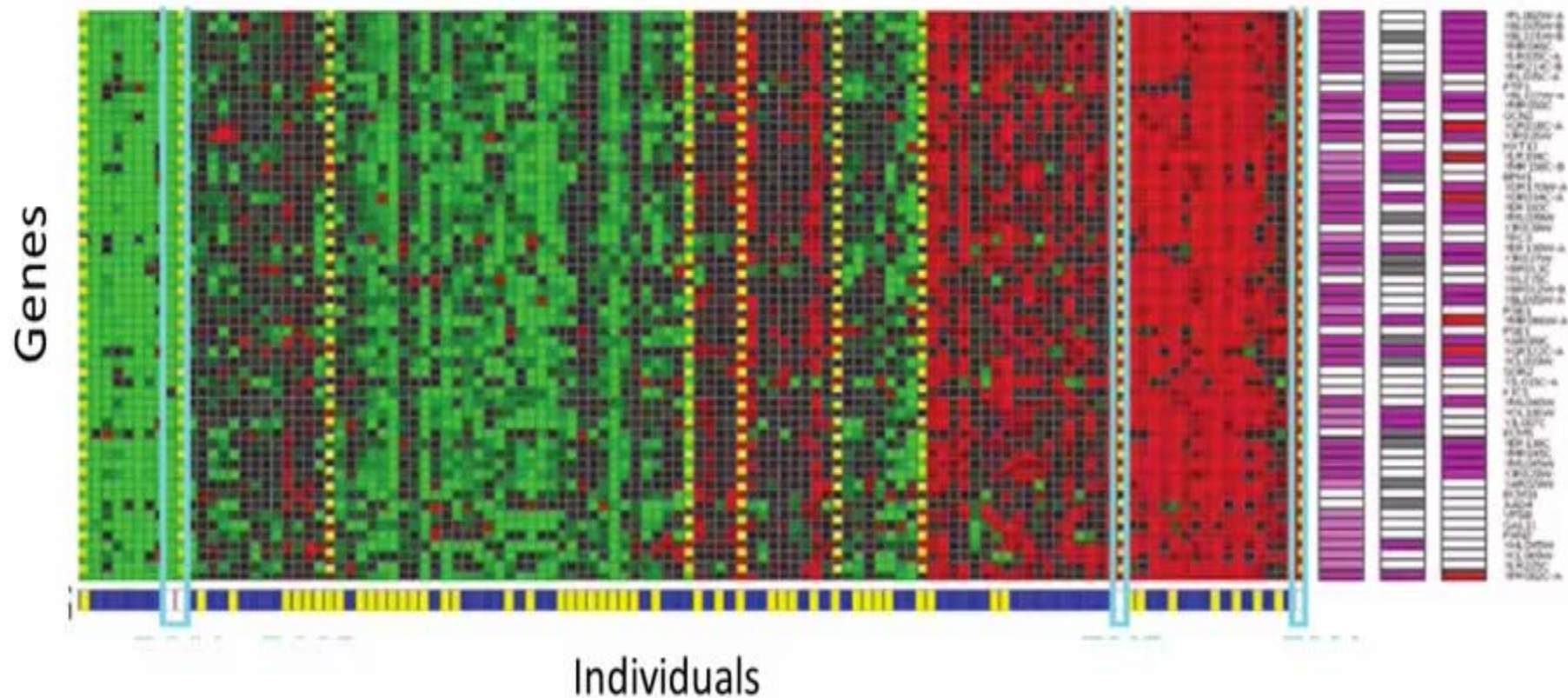
Introduction

- Real Applications: [Google News](https://news.google.com)



Introduction

- Real Applications: Genetics Analysis

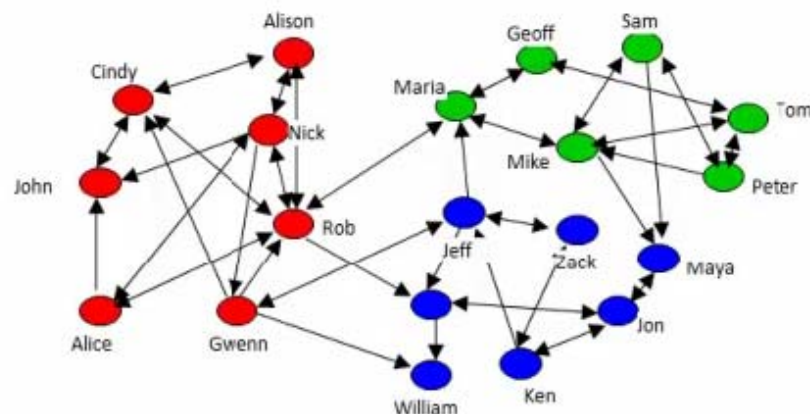


Introduction

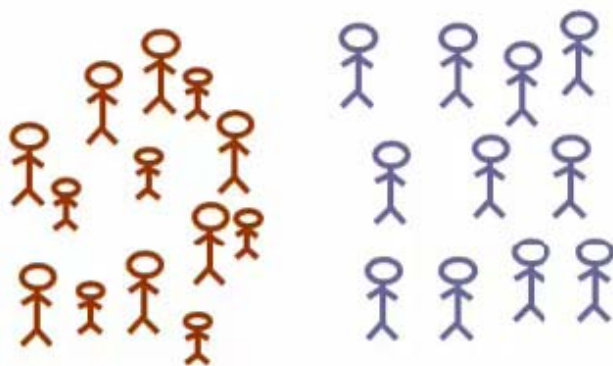
- Real Applications: Emerging Applications



Organize computing clusters



Social network analysis



Market segmentation



Astronomical data analysis

Introduction

- A technique demanded by many real world tasks
 - **Bank/Internet Security**: fraud/spam pattern discovery
 - **Biology**: taxonomy of living things such as kingdom, phylum, class, order, family, genus and species
 - **City-planning**: Identifying groups of houses according to their house type, value, and geographical location
 - **Climate change**: understanding earth climate, find patterns of atmospheric and ocean
 - **Finance**: stock clustering analysis to uncover correlation underlying shares
 - **Image Compression/segmentation**: coherent pixels grouped
 - **Information retrieval/organisation**: Google search, topic-based news
 - **Land use**: Identification of areas of similar land use in an earth observation database
 - **Marketing**: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
 - **Social network mining**: special interest group automatic discovery

Quiz

Of the following examples, which would you address using an unsupervised learning algorithm? (Check all that apply.)

- ☐ Given email labeled as spam/not spam, learn a spam filter.
- ☒ Given a set of news articles found on the web, group them into set of articles about the same story.
- ☒ Given a database of customer data, automatically discover market segments and group customers into different market segments.
- ☐ Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not.

Data Types and Representations

- Discrete vs. Continuous
 - Discrete Feature
 - Has only a finite set of values
e.g., zip codes, rank, or the set of words in a collection of documents
 - Sometimes, represented as integer variable
 - Continuous Feature
 - Has real numbers as feature values
e.g, temperature, height, or weight
 - Practically, real values can only be measured and represented using a finite number of digits
 - Continuous features are typically represented as floating-point variables

Data Types and Representations

- Data representations

- Data matrix (object-by-feature structure)

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

- n data points (objects) with p dimensions (features)
- **Two modes:** row and column represent different entities

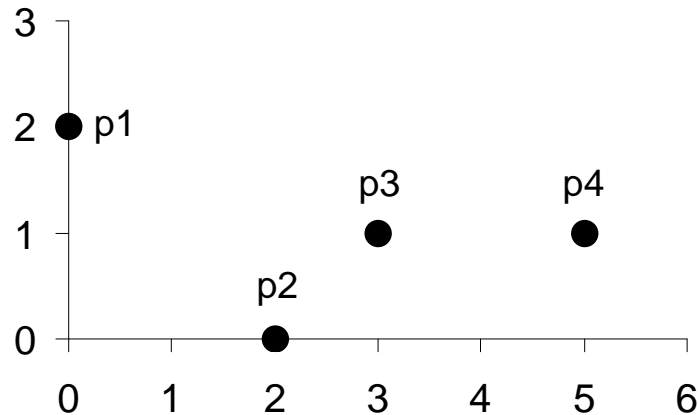
- Distance/dissimilarity matrix (object-by-object structure)

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

- n data points, but registers only the distance
- A symmetric/triangular matrix
- **Single mode:** row and column for the same entity (distance)

Data Types and Representations

- Examples



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

Data Matrix

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix (i.e., Dissimilarity Matrix) for Euclidean Distance

Distance Measures

- Minkowski Distance (http://en.wikipedia.org/wiki/Minkowski_distance)

For $\mathbf{x} = (x_1 \ x_2 \ \cdots \ x_n)$ and $\mathbf{y} = (y_1 \ y_2 \ \cdots \ y_n)$

$$d(\mathbf{x}, \mathbf{y}) = \left(|x_1 - y_1|^p + |x_2 - y_2|^p + \cdots + |x_n - y_n|^p \right)^{\frac{1}{p}}, \quad p > 0$$

- $p = 1$: Manhattan (city block) distance

$$d(\mathbf{x}, \mathbf{y}) = |x_1 - y_1| + |x_2 - y_2| + \cdots + |x_n - y_n|$$

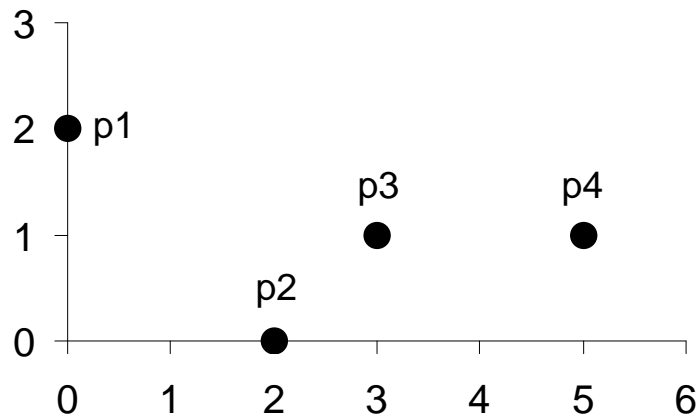
- $p = 2$: Euclidean distance

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 + \cdots + |x_n - y_n|^2}$$

- Do not confuse p with n , i.e., all these distances are defined based on all numbers of features (dimensions).
- A generic measure: use appropriate p in different applications

Distance Measures

- Example: Manhattan and Euclidean distances



L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

Distance Matrix for Manhattan Distance

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

Data Matrix

L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix for Euclidean Distance

Distance Measures

- Cosine Measure (Similarity vs. Distance)

For $\mathbf{x} = (x_1 \ x_2 \ \cdots \ x_n)$ and $\mathbf{y} = (y_1 \ y_2 \ \cdots \ y_n)$

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{x_1 y_1 + \cdots + x_n y_n}{\sqrt{x_1^2 + \cdots + x_n^2} \sqrt{y_1^2 + \cdots + y_n^2}}$$
$$d(\mathbf{x}, \mathbf{y}) = 1 - \cos(\mathbf{x}, \mathbf{y})$$

- Property: $0 \leq d(\mathbf{x}, \mathbf{y}) \leq 2$
- Nonmetric vector objects: keywords in documents, gene features in micro-arrays, ...
- Applications: information retrieval, biologic taxonomy, ...

Distance Measures

- Example: Cosine measure

$$\mathbf{x}_1 = (3, 2, 0, 5, 2, 0, 0), \mathbf{x}_2 = (1, 0, 0, 0, 1, 0, 2)$$

$$3 \times 1 + 2 \times 0 + 0 \times 0 + 5 \times 0 + 2 \times 1 + 0 \times 0 + 0 \times 2 = 5$$

$$\sqrt{3^2 + 2^2 + 0^2 + 5^2 + 2^2 + 0^2 + 0^2} = \sqrt{42} \approx 6.48$$

$$\sqrt{1^2 + 0^2 + 0^2 + 0^2 + 1^2 + 0^2 + 2^2} = \sqrt{6} \approx 2.45$$

$$\cos(\mathbf{x}_1, \mathbf{x}_2) = \frac{5}{6.48 \times 2.45} \approx 0.32$$

$$d(\mathbf{x}_1, \mathbf{x}_2) = 1 - \cos(\mathbf{x}_1, \mathbf{x}_2) = 1 - 0.32 = 0.68$$

Distance Measures

- Distance for Binary Features
 - For binary features, their value can be converted into 1 or 0.
 - Contingency table for binary feature vectors, \mathbf{x} and \mathbf{y}

		\mathbf{y}	
		1	0
\mathbf{x}	1	a	b
	0	c	d

a : number of features that equal 1 for both \mathbf{x} and \mathbf{y}

b : number of features that equal 1 for \mathbf{x} but that are 0 for \mathbf{y}

c : number of features that equal 0 for \mathbf{x} but that are 1 for \mathbf{y}

d : number of features that equal 0 for both \mathbf{x} and \mathbf{y}

Distance Measures

- Distance for Binary Features

- Distance for **symmetric** binary features

Both of their states equally valuable and carry the same weight; i.e., no preference on which outcome should be coded as 1 or 0 , e.g. gender

$$d(\mathbf{x}, \mathbf{y}) = \frac{b + c}{a + b + c + d}$$

- Distance for **asymmetric** binary features

Outcomes of the states not equally important, e.g., the *positive* and *negative* outcomes of a disease test ; the rarest one is set to 1 and the other is 0.

$$d(\mathbf{x}, \mathbf{y}) = \frac{b + c}{a + b + c}$$

Distance Measures

- Example: Distance for binary features

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	1	0	1	0	0	0
Mary	F	1	0	1	0	1	0
Jim	M	1	1	0	0	0	0

- “Y”: yes
- “P”: positive
- “N”: negative

- gender is a symmetric feature (less important)
- the remaining features are asymmetric binary
- set the values “Y” and “P” to 1, and the value “N” to 0

Mary	
Jack	2
	0
	1
	3

$$d(\text{Jack}, \text{Mary}) = \frac{0 + 1}{2 + 0 + 1} = 0.33$$

Jim	
Jack	1
	1
	1
	3

$$d(\text{Jack}, \text{Jim}) = \frac{1 + 1}{1 + 1 + 1} = 0.67$$

Mary	
Jim	1
	1
	2
	2

$$d(\text{Jim}, \text{Mary}) = \frac{1 + 2}{1 + 1 + 2} = 0.75$$

Distance Measures

- Distance for nominal features
 - A generalization of the binary feature so that **it can take more than two states/values**, e.g., red, yellow, blue, green,
 - There are two methods to handle variables of such features.

- **Simple mis-matching**

$$d(\mathbf{x}, \mathbf{y}) = \frac{\text{number of mis-matching features between } \mathbf{x} \text{ and } \mathbf{y}}{\text{total number of features}}$$

- **Convert it into binary variables**

creating new binary features for all of its nominal states

e.g., if an feature has three possible nominal states: red, yellow and blue, then this feature will be expanded into three binary features accordingly.

Thus, distance measures for binary features are now applicable!

Distance Measures

- Distance for nominal features (cont.)
 - Example: Play tennis

	Outlook	Temperature	Humidity	Wind
D_1	010	100	10	10
D_2	100	100	01	10

- Simple mis-matching

$$d(D_1, D_2) = \frac{2}{4} = 0.5$$

- Creating new binary features

- Using the same number of bits as those features can take

Outlook = {Sunny, Overcast, Rain} \longrightarrow (100, 010, 001)

Temperature = {High, Mild, Cool} \longrightarrow (100, 010, 001)

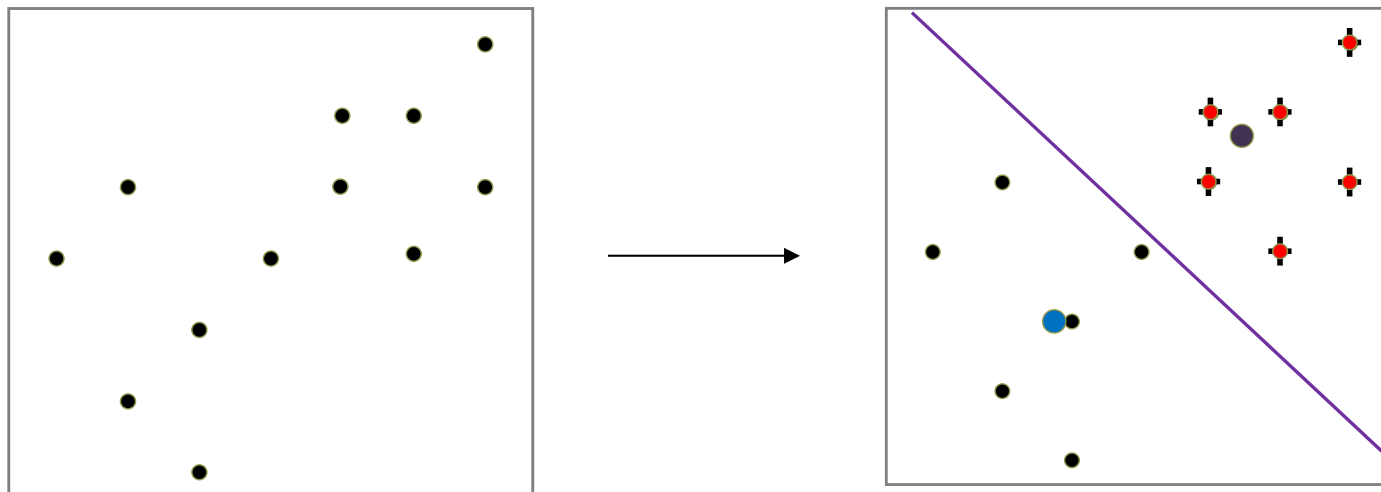
Humidity = {High, Normal} \longrightarrow (10, 01)

Wind = {Strong, Weak} \longrightarrow (10, 01)

$$d(D_1, D_2) = \frac{2+2}{10} = 0.4$$

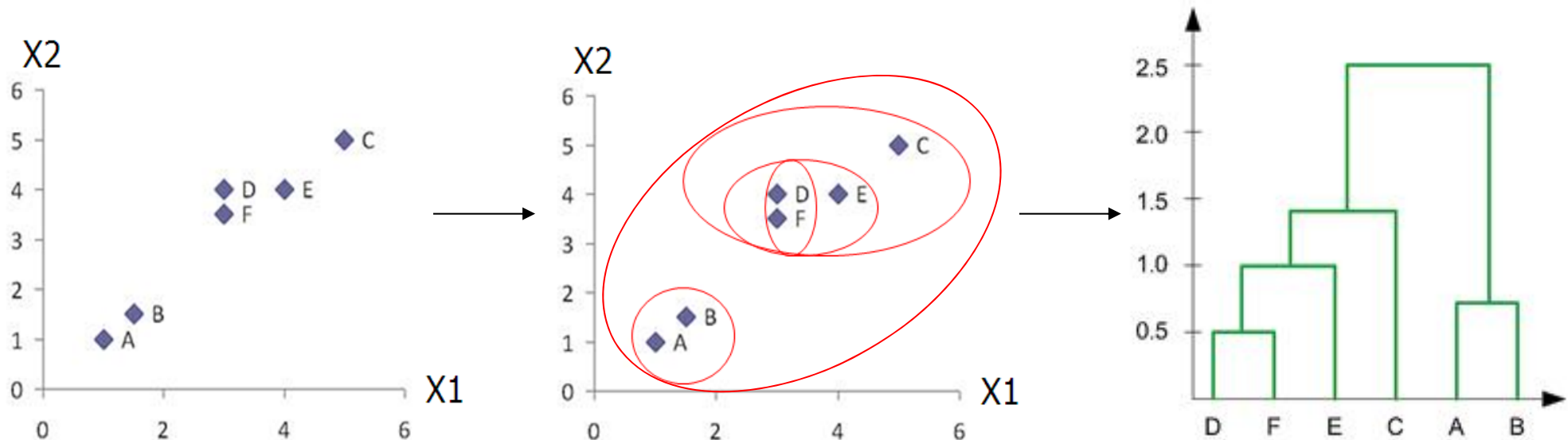
Major Clustering Approaches

- Partitioning approach
 - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square distance cost
 - Typical methods: **k-means**, k-medoids, CLARANS,



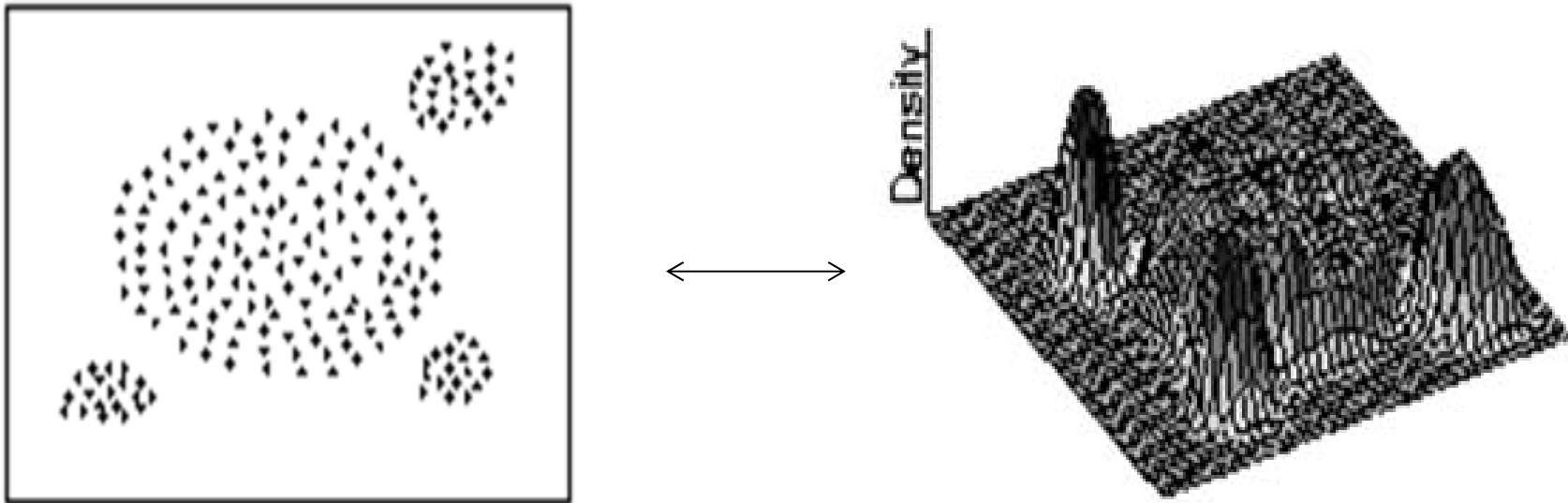
Major Clustering Approaches

- Hierarchical approach
 - Create a hierarchical decomposition of the set of data (or objects) using some criterion
 - Typical methods: **Agglomerative**, Diana, Agnes, BIRCH, ROCK,



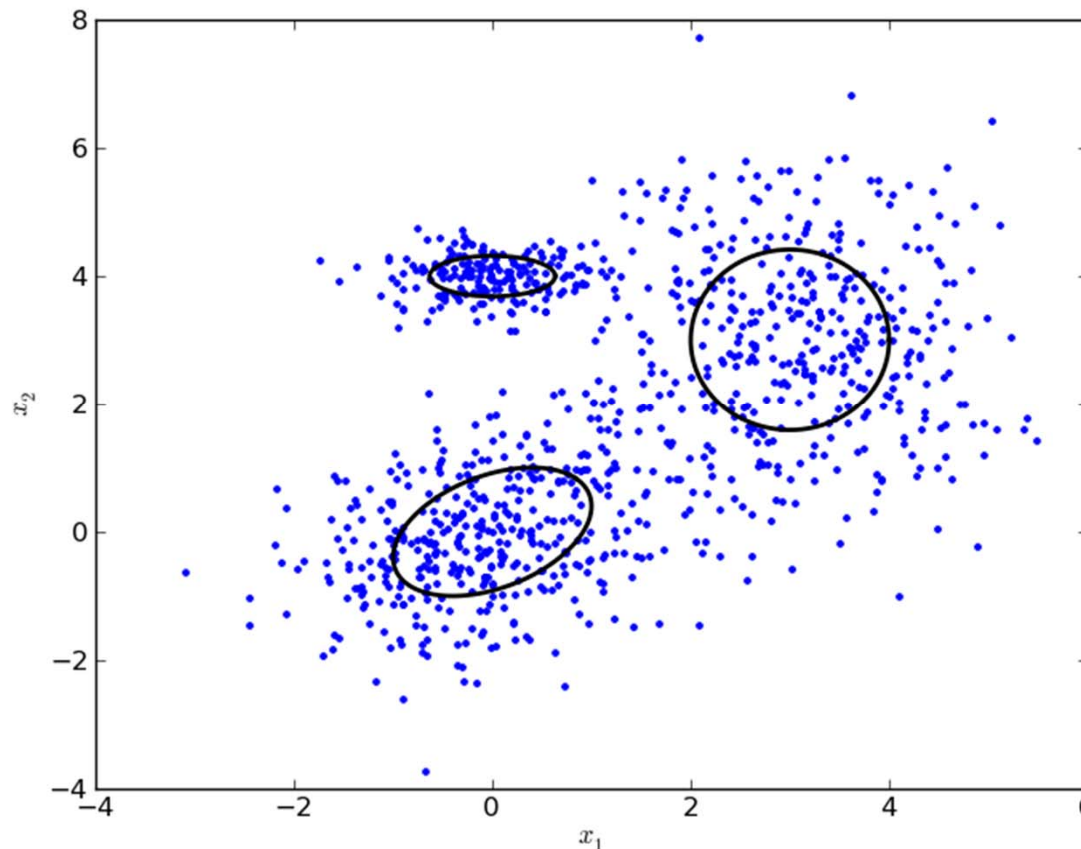
Major Clustering Approaches

- Density-based approach
 - Based on connectivity and density functions
 - Typical methods: DBSACN, OPTICS, DenClue,



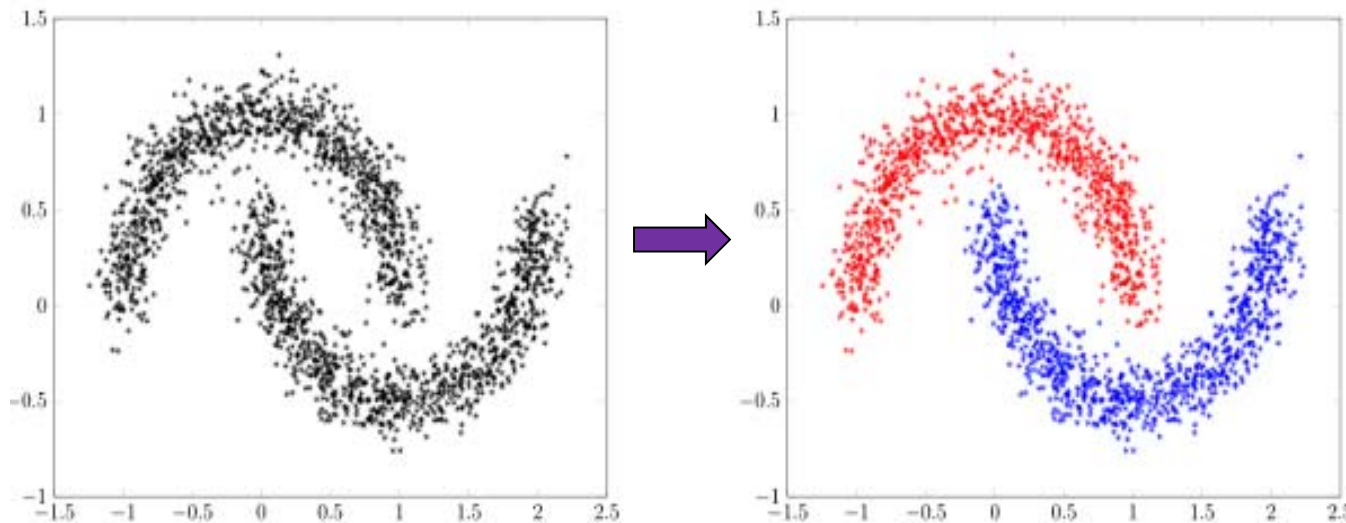
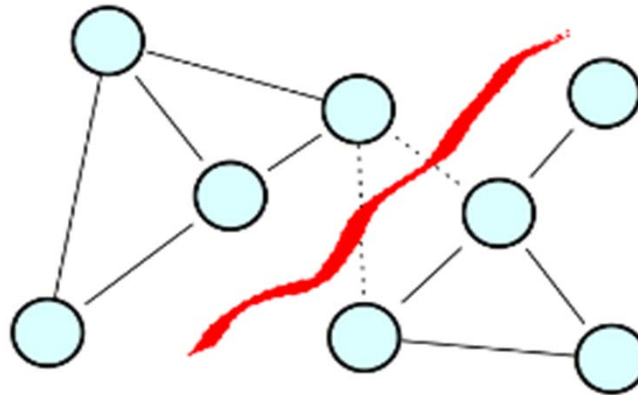
Major Clustering Approaches

- Model-based approach
 - A generative model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
 - Typical methods: Gaussian Mixture Model (GMM), COBWEB,



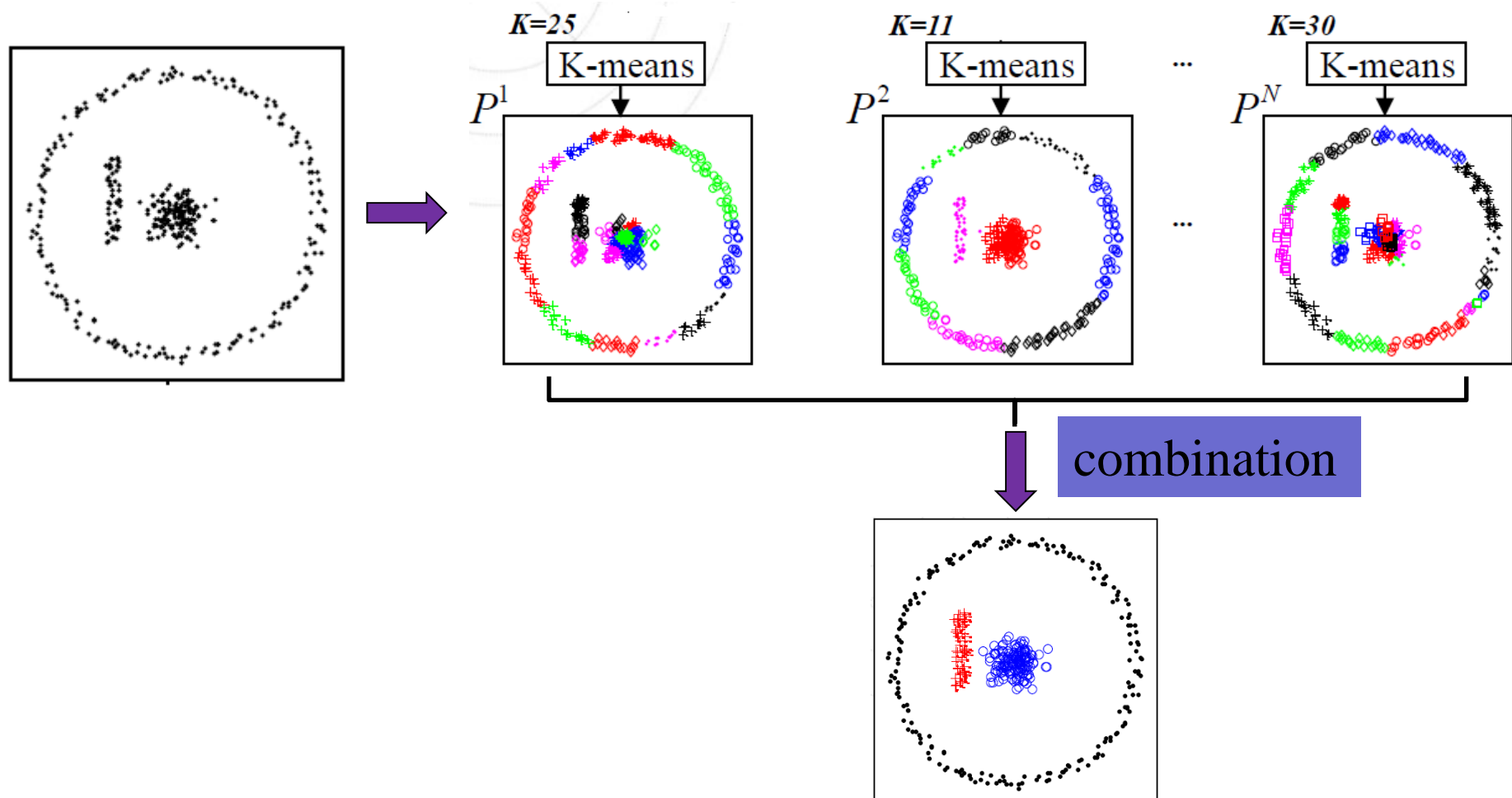
Major Clustering Approaches

- Spectral clustering approach
 - Convert data set into weighted graph (vertex, edge), then cut the graph into sub-graphs corresponding to clusters via spectral analysis
 - Typical methods: Normalised-Cuts



Major Clustering Approaches

- Clustering ensemble approach
 - Combine multiple clustering results (different partitions)
 - Typical methods: Evidence-accumulation based, graph-based



Summary

- **Clustering analysis** groups objects based on their (dis)similarity and has a broad range of applications.
- Measure of **distance** (or **similarity**) plays a critical role in clustering analysis and distance-based learning.
- Clustering algorithms can be categorized into partitioning, hierarchical, density-based, model-based, spectral clustering as well as ensemble approaches.
- There are still lots of research issues on cluster analysis;
 - finding the number of “natural” clusters with arbitrary shapes
 - dealing with mixed types of features
 - handling massive amount of data – Big Data
 - coping with data of high dimensionality
 - performance evaluation (especially when no ground-truth available)