Course. Introduction to Machine Learning
Lecture 2. Introduction to unsupervised learning and Cluster Analysis (Part I)

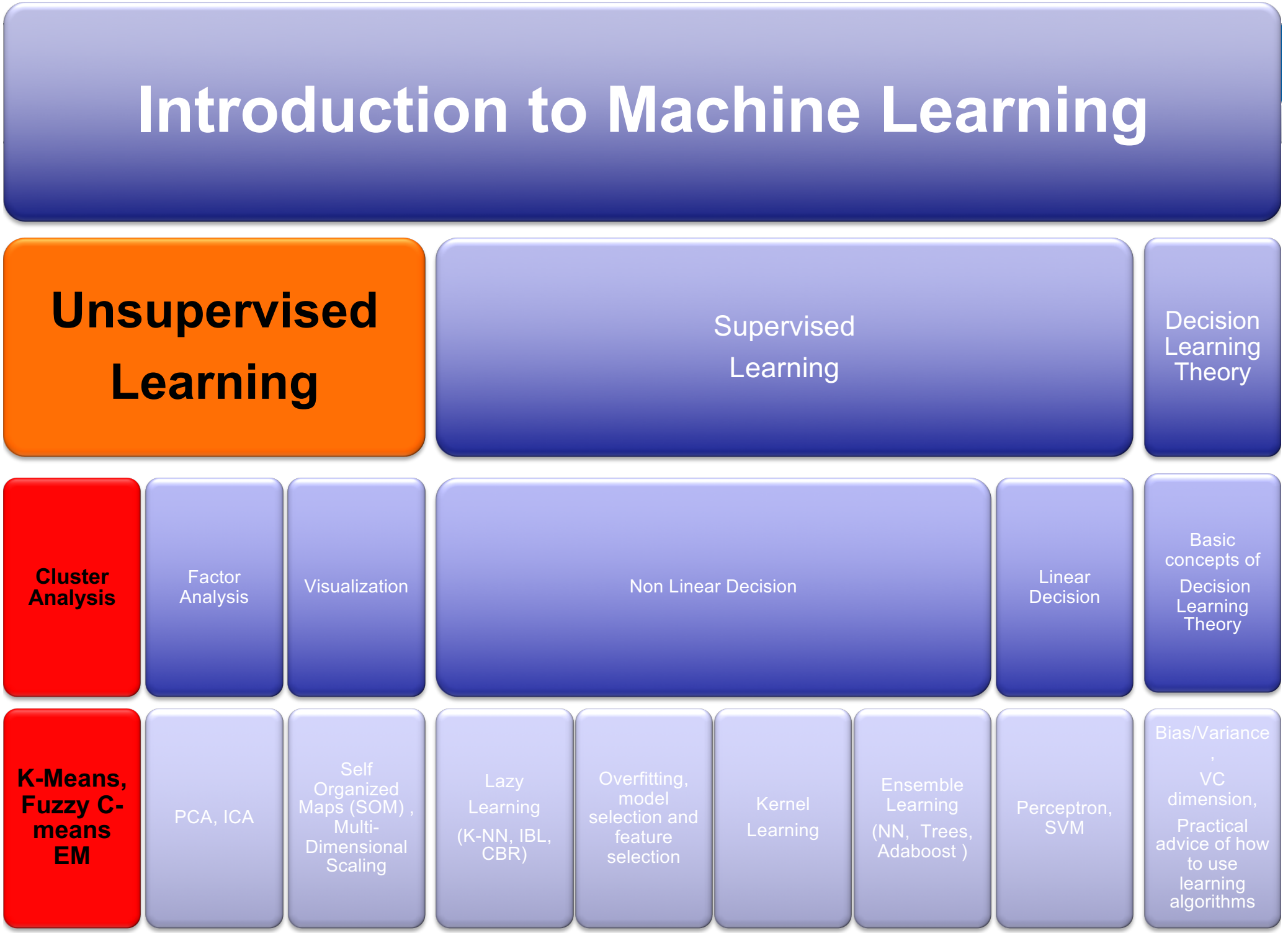
Dr. Maria Salamó Llorente

Dept. Mathematics and Informatics,

Faculty of Mathematics and Informatics,

University of Barcelona

Introduction to Machine Learning

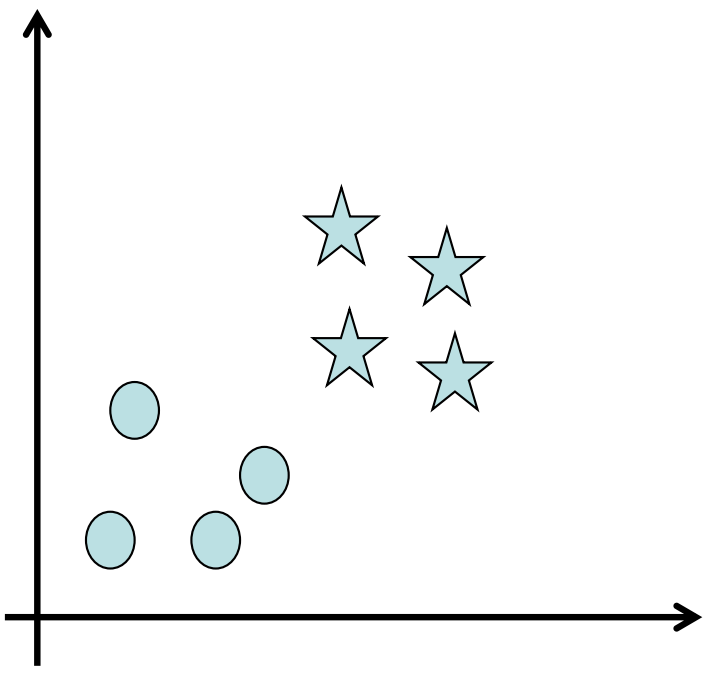


UNIVERSITAT DE BARCELONA

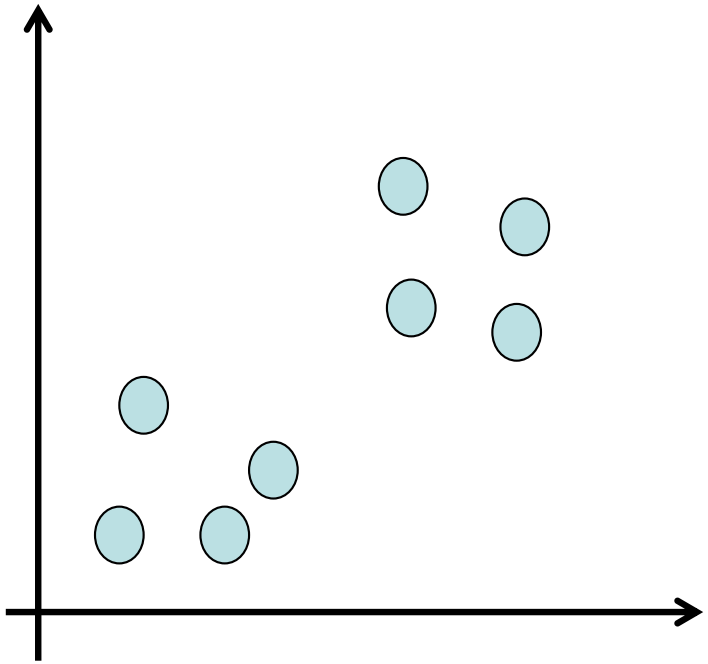
Contents

Introduction to Unsupervised learning

Supervised



Unsupervised



Unsupervised learning

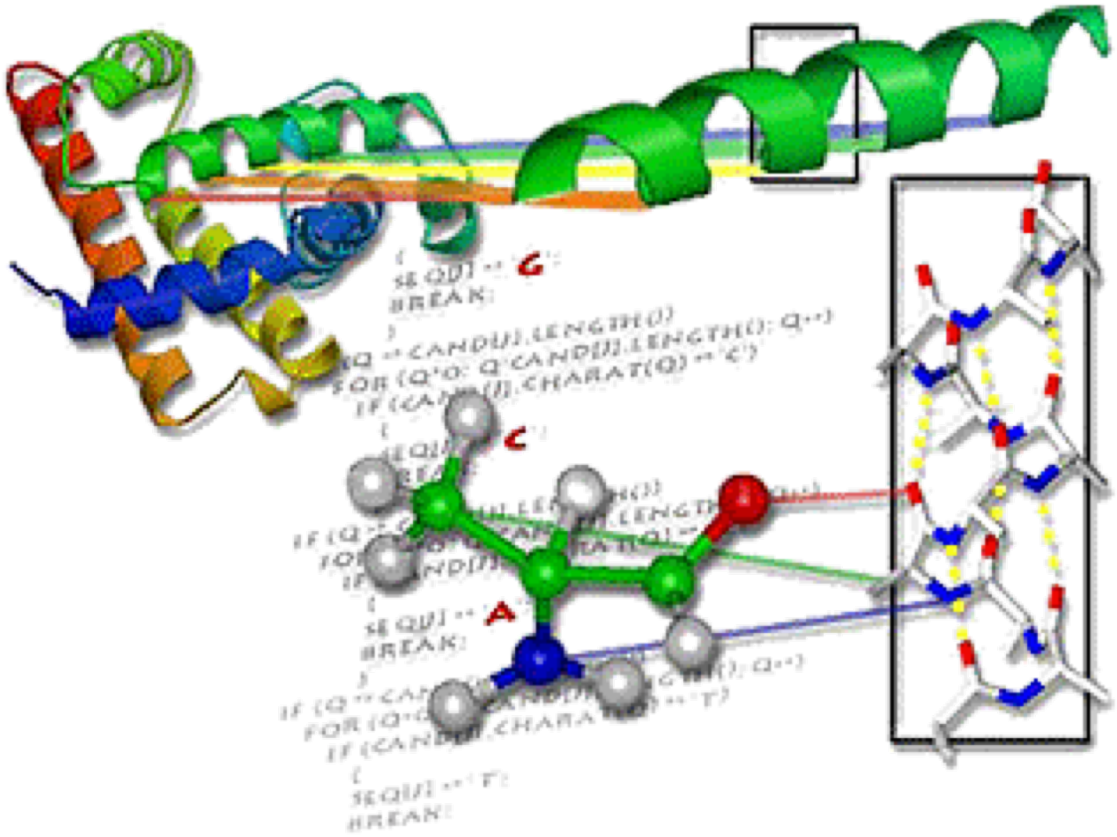
- **Main Goals**:
  - **Summarization**: To obtain representations that describe an unlabeled dataset
  - **Understanding**: To discover the key concepts inside the data
- These tasks are difficult because the discovery process is biased by context
  - Different answers can be valid depending of the discovery goal or the domain
  - There are few criterion to validate the results
- Representation of the clusters:
  - Unstructured (partitions)
  - Relational (hierarchies)

Examples of unsupervised learning

- Bioinformatics
- Medicine
- Market research
- Social network analysis
- NLP: document clustering, text mining, concept extraction
- Image segmentation
- Educational data mining
- Climatology
- …

UNIVERSITAT DE BARCELONA

Bioinformatics



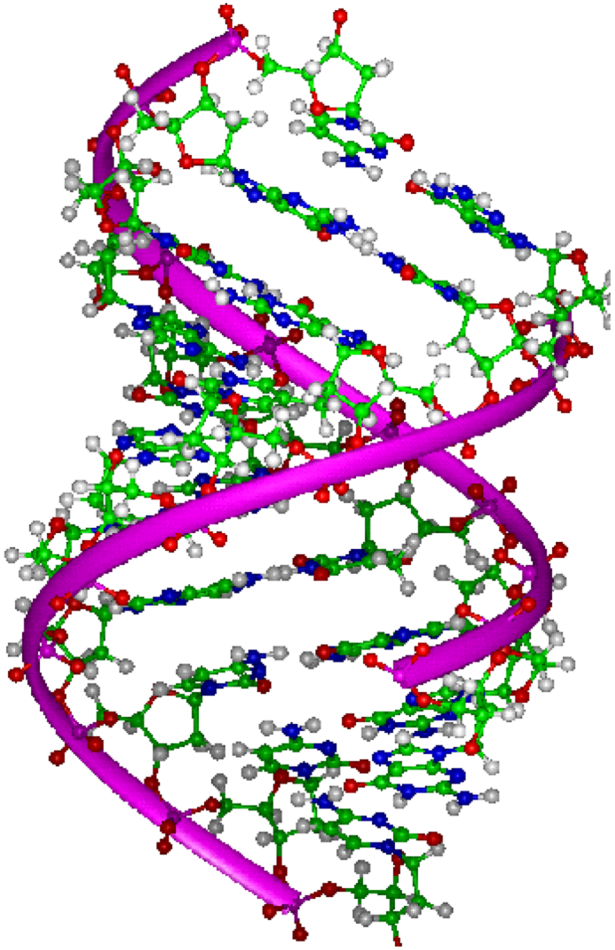
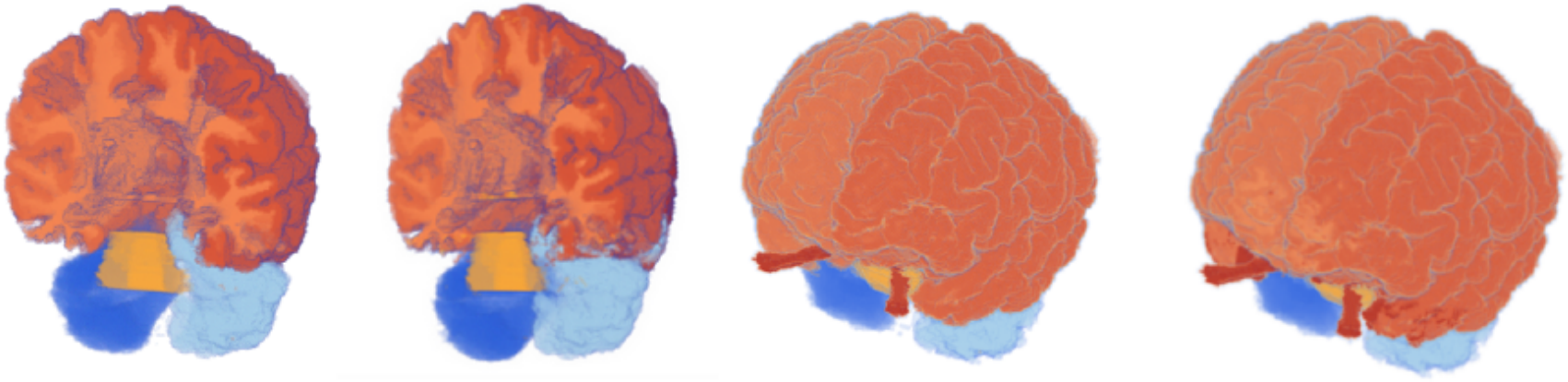
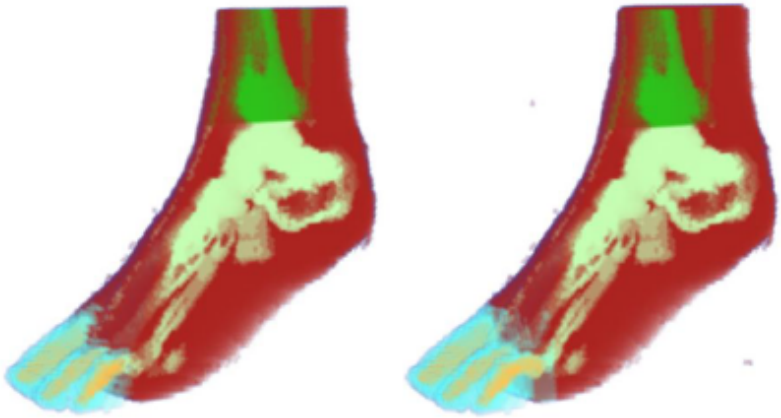


Illustration by Brian Haas, Phillips lab

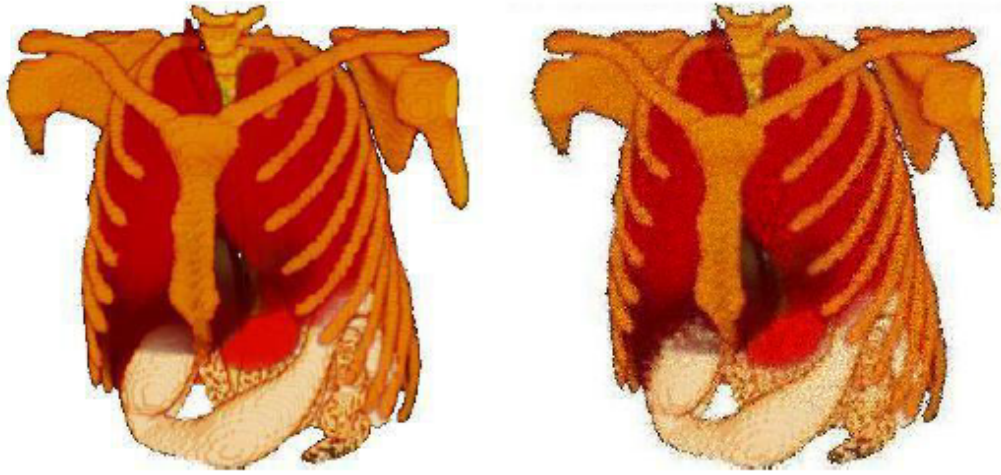
Medicine



(a)



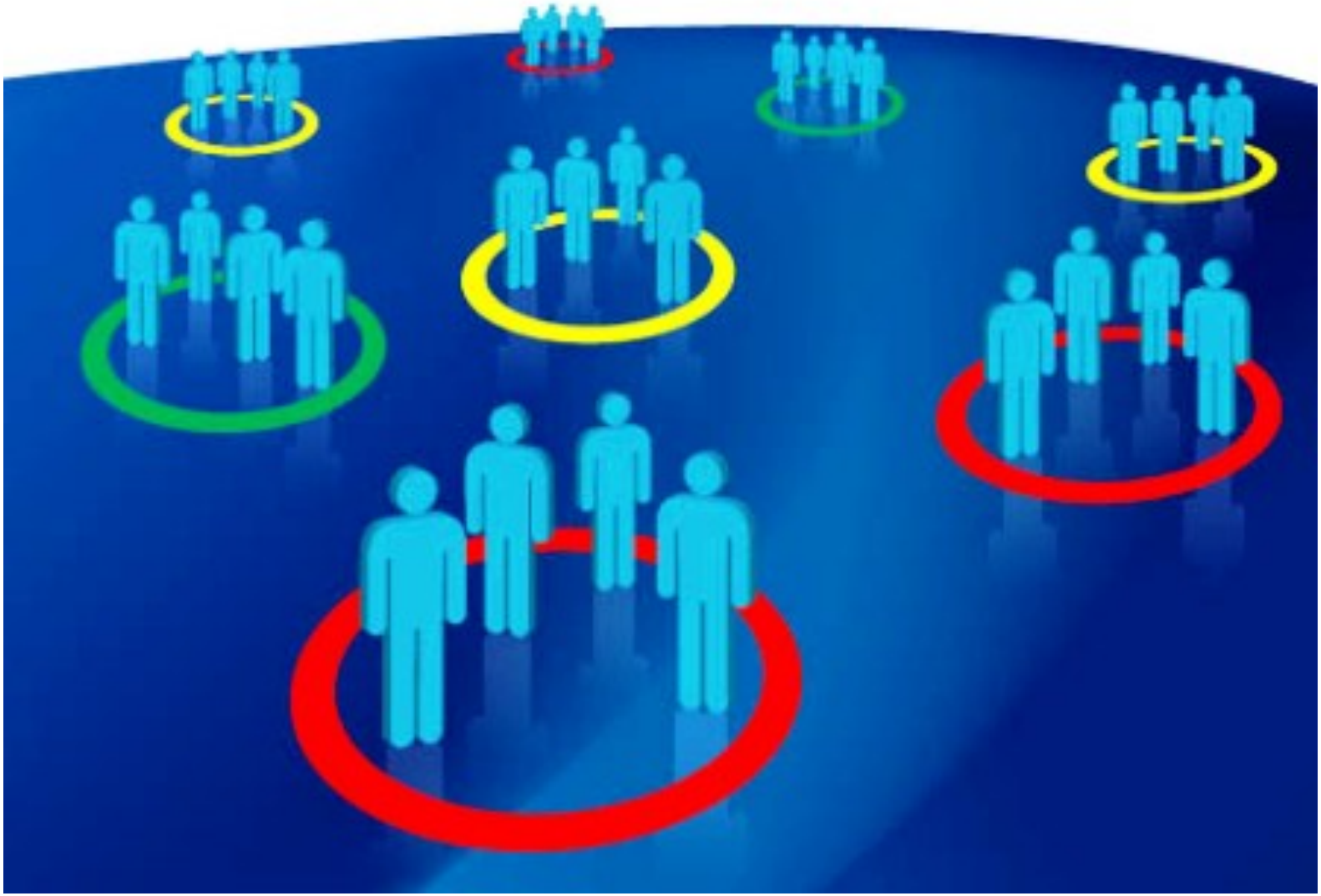
(b)



(c)

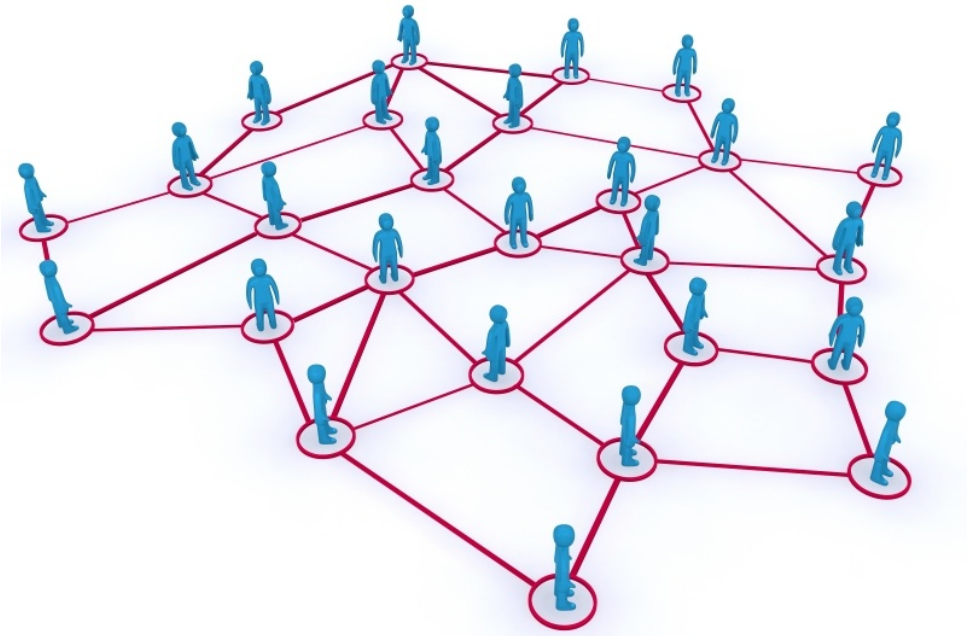
UNIVERSITAT DE BARCELONA

Market research



UNIVERSITAT DE BARCELONA

Social network analysis



Unsupervised learning approaches

- **Clustering**
  - It is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense or another) to each other than to those in other groups (clusters)
- **Factor Analysis**
  - Statistical methods used to describe variability among observed, correlated variables in terms of a potentially lower number of unobserved variables called factors.
- **Visualization**
  - Study of (interactive) visual representations of abstract data to reinforce human cognition.

Unsupervised learning approaches

- **Clustering**: method by which large sets of data is grouped into clusters of smaller sets of similar data
  - **Based on connectivity**: Hierarchical clustering
  - **Based on centroids**: K-means
  - **Distribution-based models**: Mixture models, Expectation-Maximization
  - Density models: DBScan, Optics
  - Subspace models: Biclustering
  - Group models
  - Graph-based models

Unsupervised learning approaches

- **Factor analysis**: blind signal separation using *feature extraction* techniques for *dimensionality reduction*

  - **Principal components analysis (PCA)**
  - **Independent component analysis (ICA)**
  - Non-negative matrix factorization
  - Singular value decomposition (SVD)
  - ...

Unsupervised learning approaches

- **Visualization**: a set of techniques often used in information visualization for exploring similarities and dissimilarities in data
  - **Neural network models**:
    - **Self-organized maps (SOM)**
    - Adaptive resonance theory (ART)
  - **Multi-dimensional scaling (MDS)**
    - **Classical multidimensional scaling**
    - Metric multidimensional scaling
    - Non metric multidimensional scaling
    - Generalized multidimensional scaling

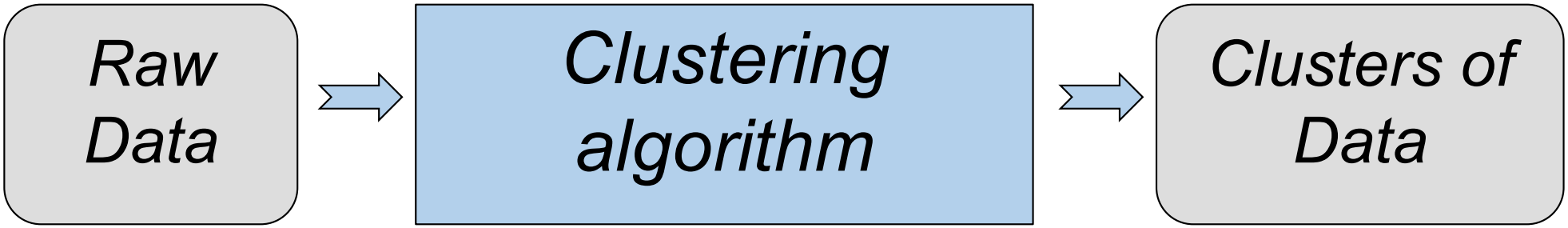
UNIVERSITAT DE BARCELONA

Contents

1. Introduction to unsupervised learning
    1. Introduction to unsupervised learning
    2. Examples
    3. Definition of unsupervised learning
    4. Unsupervised learning approaches
2. **Cluster analysis**
    1. Defining clustering analysis
    2. Areas that apply clustering
    3. Classification of clustering algorithms
    4. K-Means
    5. EM
    6. Other algorithms
    7. Applications

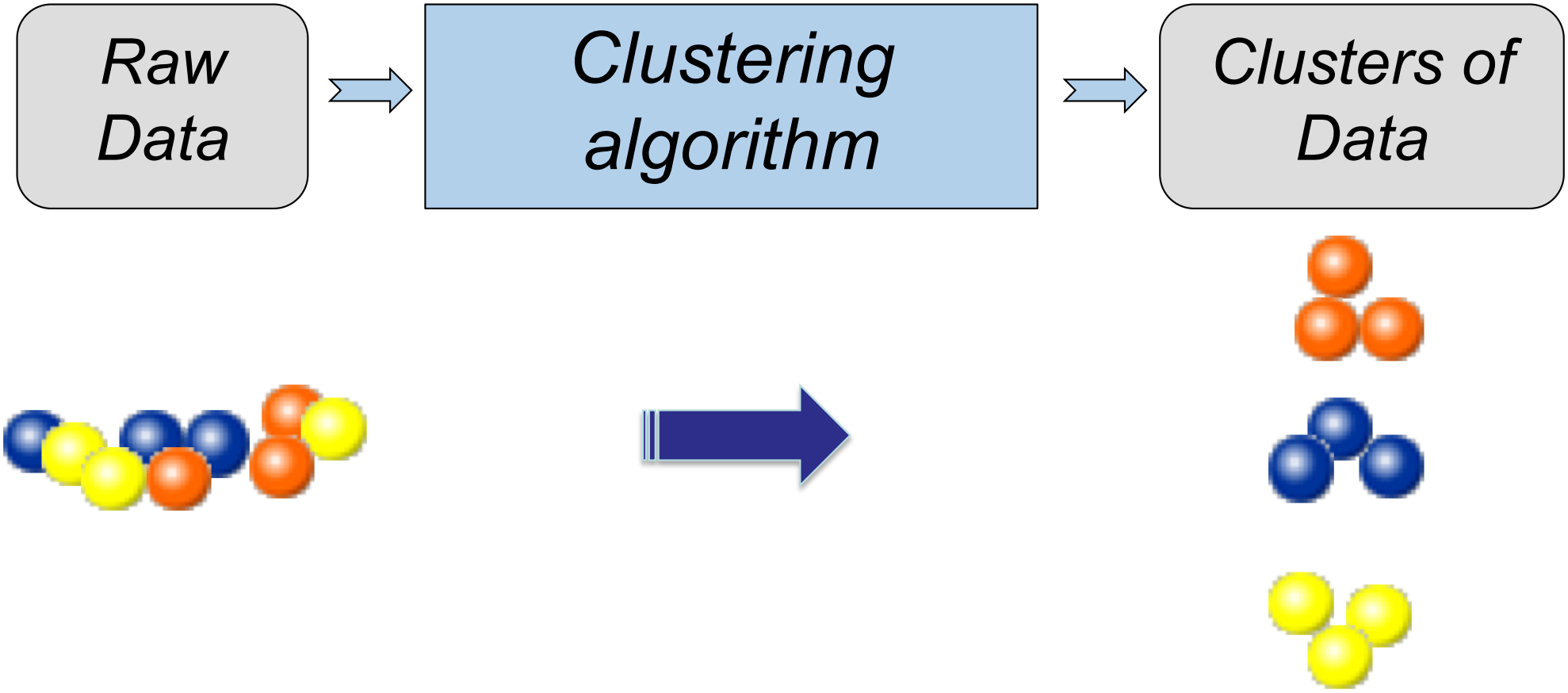
Defining clustering analysis

- **Cluster analysis** or **clustering** is the task of assigning a set of unlabelled objects into groups (**clusters**) so that the objects in the same cluster are very similar (in some sense or another) to each other than those of the other clusters.
- Cluster analysis discover new categories in an **unsupervised** manner



Defining clustering analysis

- A clustering algorithm attempts to find natural groups of components (or data) based on some similarity.



Areas that apply clustering

Clustering is ...

- Main task of explorative **data mining**
- Common technique for **statistical data** analysis used in many fields:
  - Machine learning
  - Pattern recognition
  - Information retrieval
  - Bioinformatics
  - Natural language processing
  - Recommender systems
  - Data mining
  - ...

Clustering Example

How many clusters?

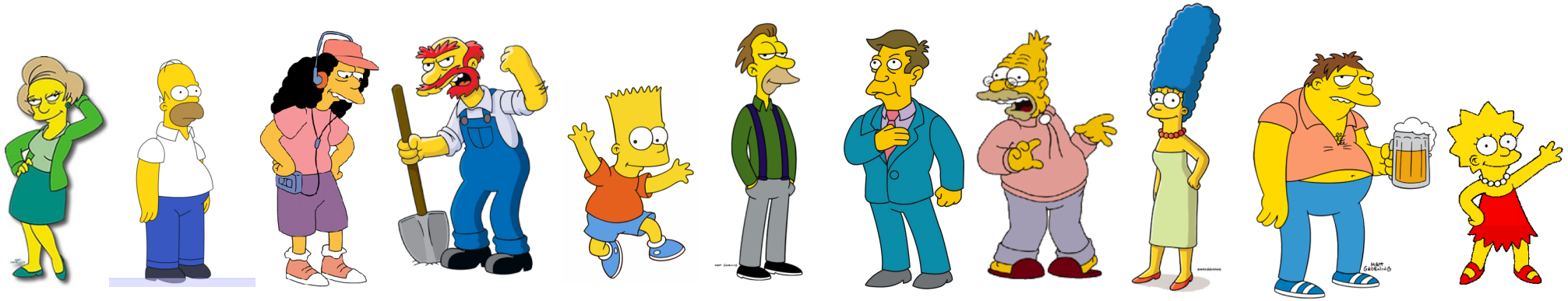
UNIVERSITAT DE BARCELONA

Issues for clustering

- What is a natural grouping among these objects?
  - Definition of “groupness”
- What makes object “related”?
  - Definition of “similarity/distance”
- Representation of objects
  - Vector space? Normalization?
- How many clusters
  - Fixed a priori?
  - Completely data driven?
- Clustering algorithms
  - Hierarchical algorithms
  - Partitional algorithms
- Formal foundation and convergence

UNIVERSITAT DE BARCELONA

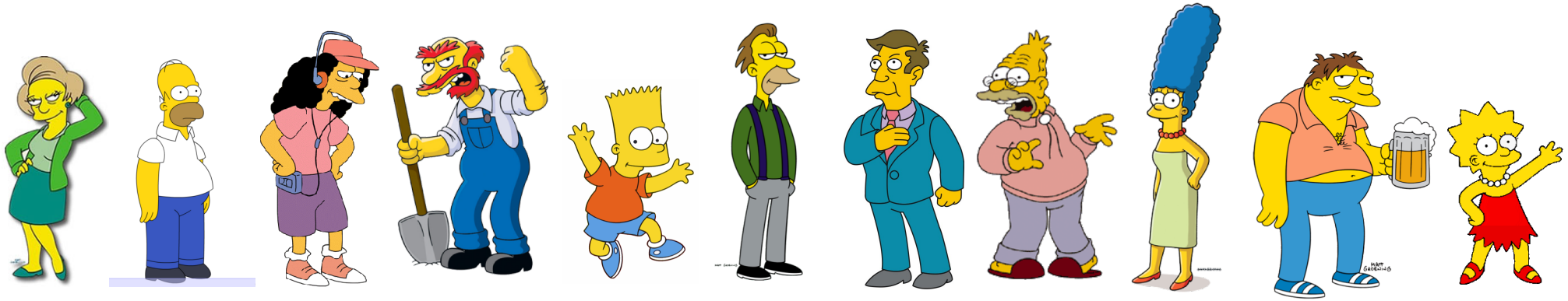
What is a natural grouping?



Decide how to group them !!!

UNIVERSITAT DE BARCELONA

What is a natural grouping?



Decide how to group them, in a diferent way !!!

What is similarity?

UNIVERSITAT DE BARCELONA

Desirable distance measure properties

- $D(A, B) = D(B,A)$ Symmetry
- $D(A,A)= 0$ Constancy of Self-Similarity
- $D(A,B) = 0$ | if $A=B$ Positivity Separation
- $D(A,B) \leq D(A,C)+ D(B,C)$ Triangular Inequality

Distance measure: Minkowski Metric

- Suppose two object *x* and *y* both have *p* features

$$x = (x_1, x_2, \cdots, x_p)$$

$$y = (y_1, y_2, \cdots, y_p)$$

- The Minkowski metric is defined by

$$d(x, y) = \sqrt[r]{\sum_{i=1}^{p} | x_i - y_i |^r}$$

Distance measures:

Pearson correlation

- Pearson correlation coefficient:

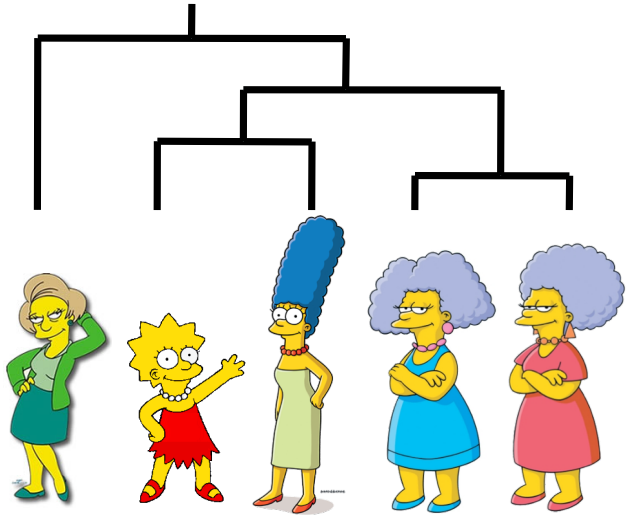
$$s(x,y) = \frac{\sum_{i=1}^{p}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{p}(x_i - \bar{x})^2 \times \sum_{i=1}^{p}(y_i - \bar{y})^2}}$$

$$\text{where } \bar{x} = \frac{1}{p}\sum_{i=1}^{p} x_i \text{ and } \bar{y} = \frac{1}{p}\sum_{i=1}^{p} y_i .$$

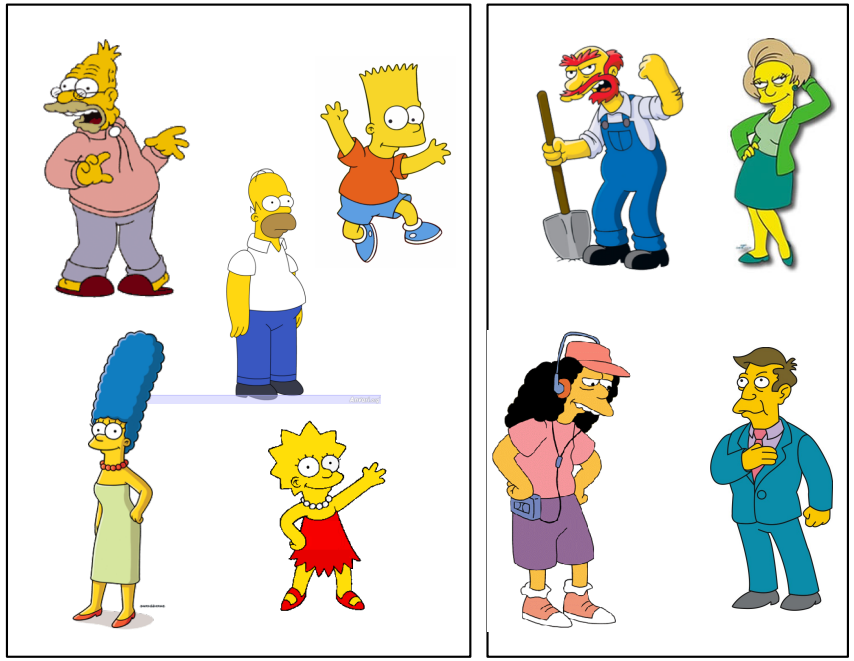
$$\left| s(x,y) \right| \leq 1$$

Taxonomy

- **Hierarchical algorithms**
  - Examples are organized as a binary tree
  - No explicit division in groups
    - **Bottom-up**
    - Top-down

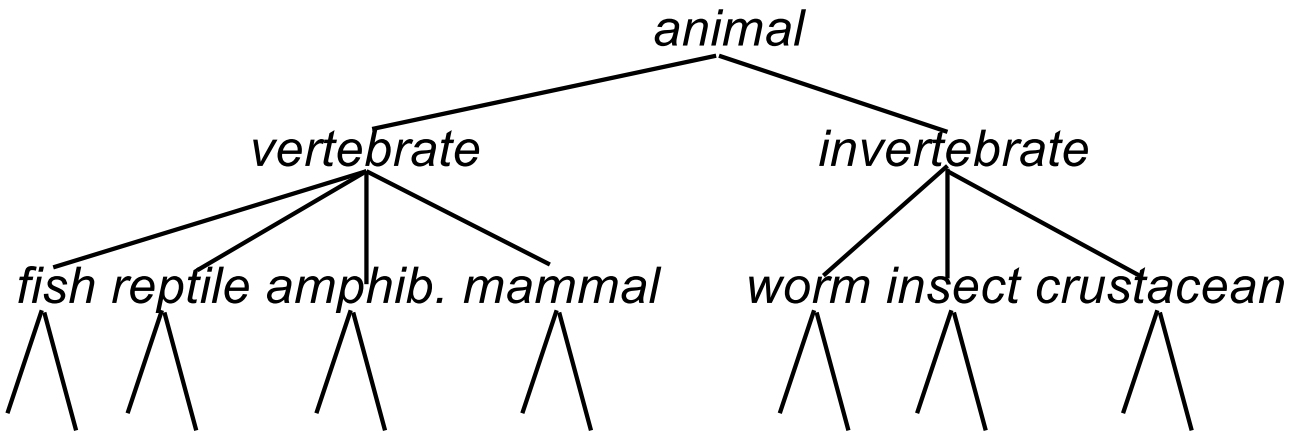


- **Partitional algorithms**
  - Usually start with a random (partial) partitioning
  - Refine it iteratively:
    - **K-means clustering**
    - **Mixture-model based clustering**



Hierarchical Clustering

- Build a tree-based hierarchical taxonomy (*dendrogram*) from a set of unlabeled examples



- Recursive application of a standard clustering algorithm can produce a hierarchical clustering

Hierarchical Clustering Algorithms

- *Agglomerative* (*bottom-up*)
  - Methods start with each example in its own cluster
  - Iteratively combine them to form larger and larger clusters
- *Divisive* (*partitional, top-down*)
  - Methods start with all the examples in a single cluster
  - Consider all the possible way to divide the cluster into two. Choose the best division
  - Recursively operate on both sides

Hierarchical Agglomerative Clustering

(HAC algorithm)

Basic HAC algorithm:

1. Compute the similarity matrix between the input data points
2. Start with all instances in their own cluster
3. **Repeat**
4. Among the current clusters, determine the two clusters, c_i and c_j, that are most similar
5. Merge them and replace c_i and c_j with a single cluster $c_i \cup c_j$
6. Update the similarity matrix
7. **until** there is only one single cluster

- **Key operation is the computation of the similarity between two clusters**
  - Different definitions of the similarity between clusters lead to different algorithms

Cluster Similarity

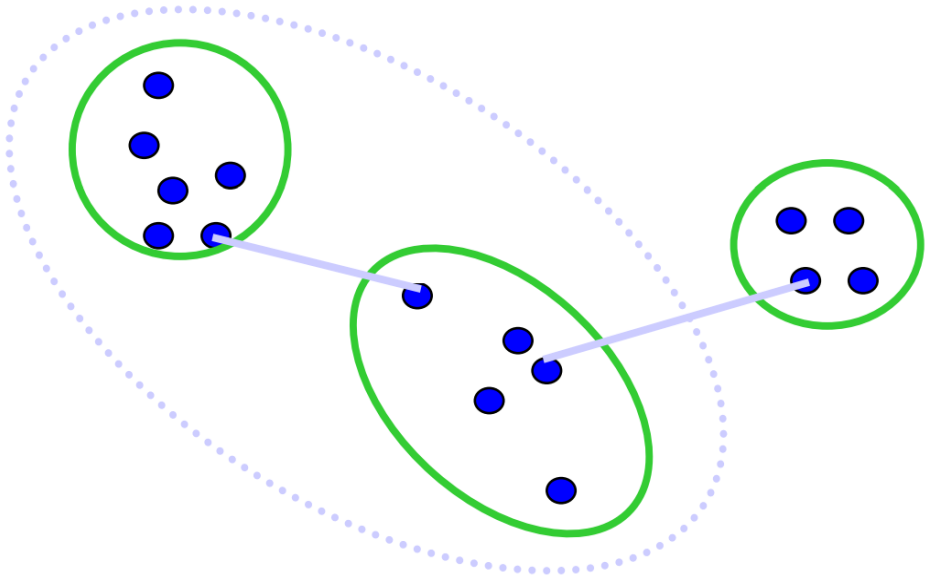
- Assume a similarity function that determines the similarity of two instances: ***sim*(*x*,*y*)**
  - For example, Cosine similarity of document vectors
- How to compute similarity of two clusters each possibly containing multiple instances?
  - **Single Link**: Similarity of two most similar members
  - **Complete Link**: Similarity of two least similar members
  - **Group Average**: Average similarity between members
  - Centroid: clusters whose centroids are the most cosine similar

Single-Link Agglomerative Clustering

- Use maximum similarity of pairs:

$$sim(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y)$$

- Can result in "straggly" (long and thin) clusters due to *chaining effect*.
  - Appropriate in some domains, such as clustering islands

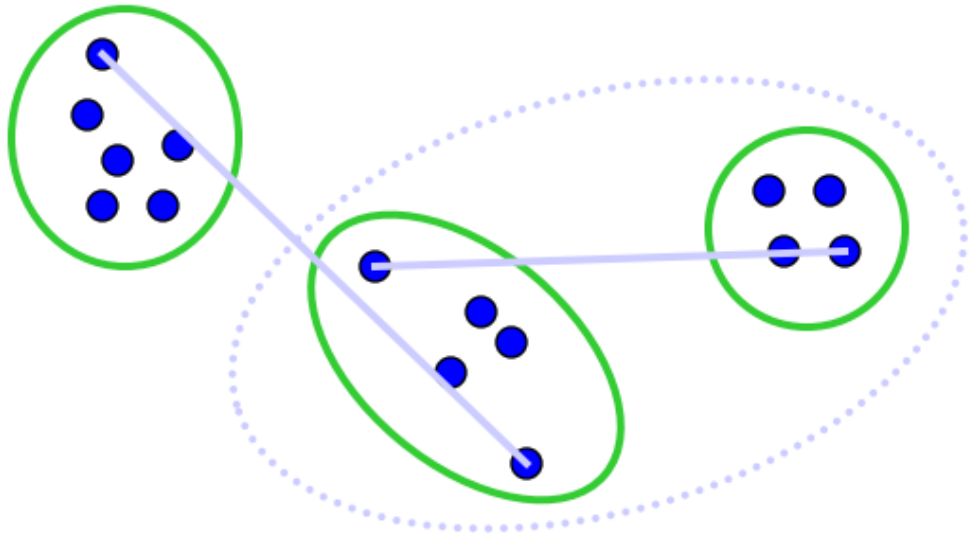


Complete-Link Agglomerative Clustering

- Use minimum similarity of pairs:

$$sim(c_i, c_j) = \min_{x \in c_i, y \in c_j} sim(x, y)$$

- Makes more "tight," spherical clusters that are typically preferable



Computational Complexity

- In the first iteration, all HAC methods need to compute similarity of all pairs of ***n*** individual instances which is **O(n^2)**.
- In each of the subsequent $n-2$ merging iterations, it must compute the distance between the most recently created cluster and all other existing clusters.
- In order to maintain an overall O(n^2) performance, computing similarity to each other cluster must be done in constant time.
- Else **O($n^2 \log n$) or O(n^3)** if done naively

Computing Cluster Similarity

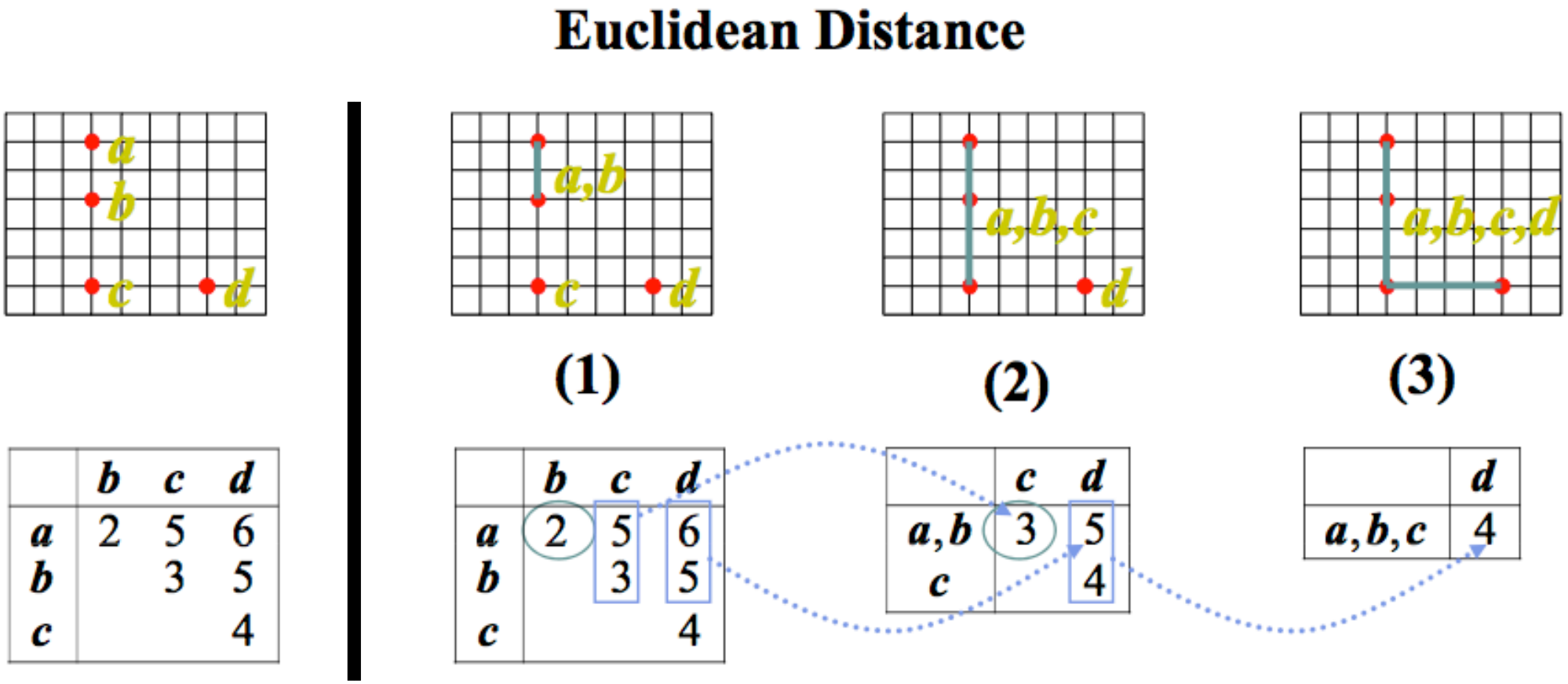
- After merging c_i and c_j, the similarity of the resulting cluster to any other cluster, c_k, can be computed by:
  - **Single-Link**:

$$sim((c_i \cup c_j), c_k) = \max(sim(c_i, c_k), sim(c_j, c_k))$$

  - **Complete-Link**:

$$sim((c_i \cup c_j), c_k) = \min(sim(c_i, c_k), sim(c_j, c_k))$$

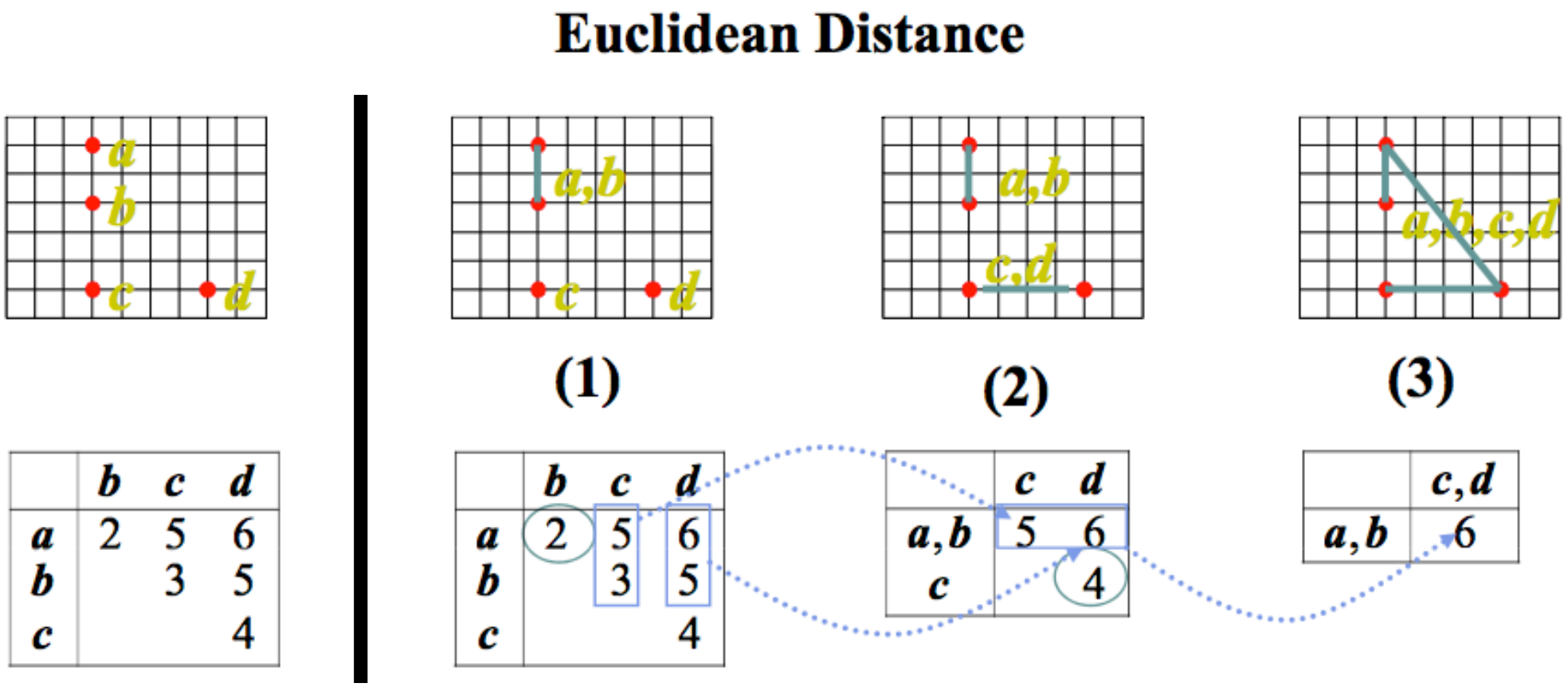
Single-Link Example



Distance Matrix

Illustration by Eric Xing @ CMU

Complete-Link Example

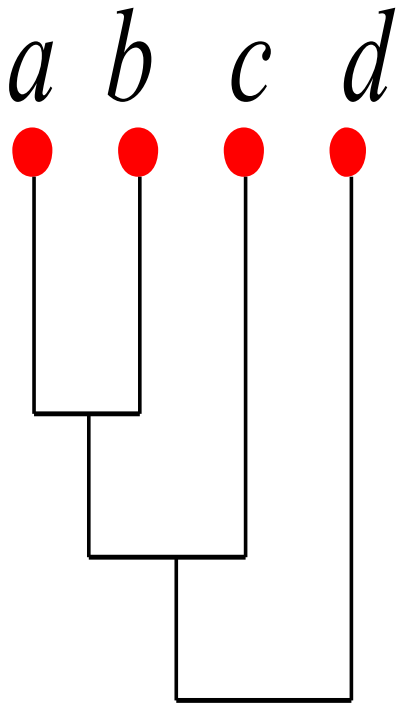


Distance Matrix

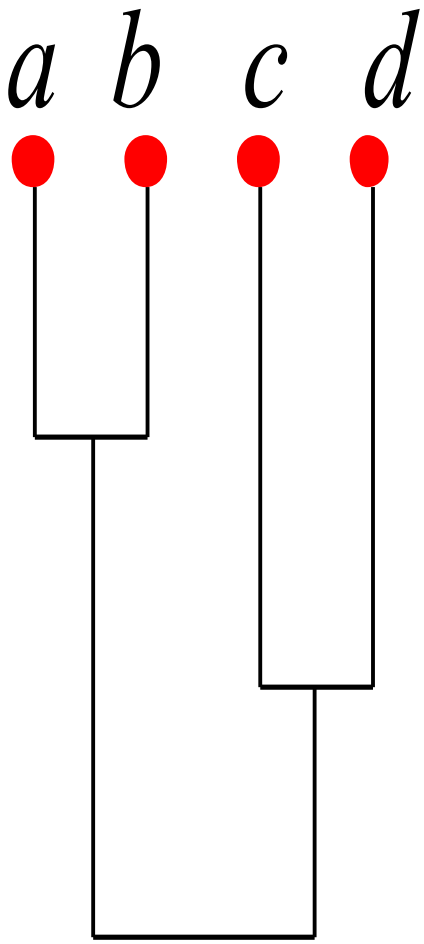
Illustration by Eric Xing @ CMU

Dendrograms Comparison

Single-Link



Complete-Link

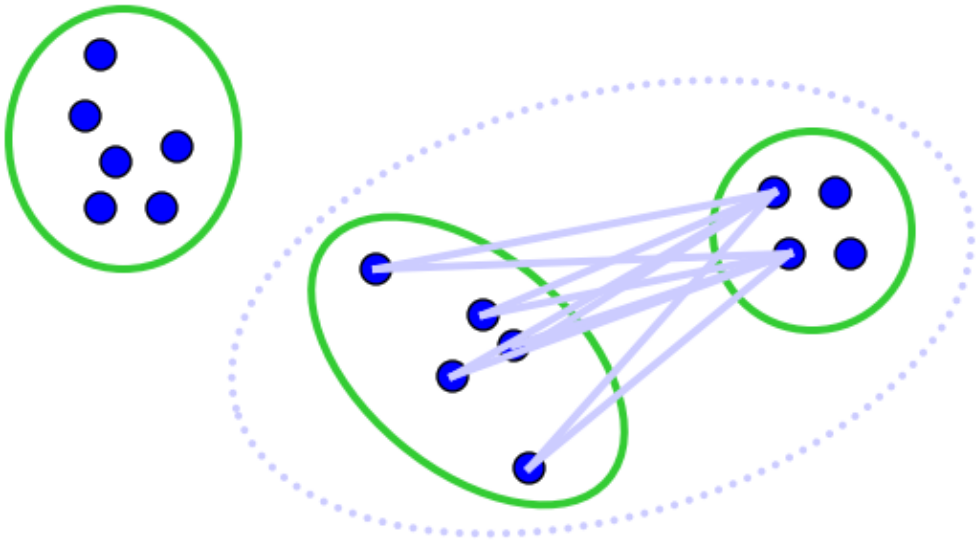


Group Average Agglomerative Clustering

- Use average similarity across all pairs within the merged cluster to measure the similarity of two clusters.

$$sim(c_i, c_j) = \frac{1}{\left|c_i \cup c_j\right|\left(\left|c_i \cup c_j\right| - 1\right)} \sum_{\vec{x} \in (c_i \cup c_j)} \sum_{\vec{y} \in (c_i \cup c_j): \vec{y} \neq \vec{x}} sim(\vec{x}, \vec{y})$$

- Compromise between Single and Complete link.



Computing Group Average Similarity

- Assume cosine similarity and normalized vectors with unit length.
- Always maintain sum of vectors in each cluster.

$$\vec{s}(c_j) = \sum_{\vec{x} \in c_j} \vec{x}$$

- Compute similarity of clusters in constant time:

$$sim(c_i, c_j) = \frac{(\vec{s}(c_i) + \vec{s}(c_j)) \bullet (\vec{s}(c_i) + \vec{s}(c_j)) - (|c_i| + |c_j|)}{(|c_i| + |c_j|)(|c_i| + |c_j| - 1)}$$