

Week 2

Course. Introduction to Machine Learning

Theory 2. Introduction to unsupervised learning and Cluster Analysis (Part I)

Dr. Maria Salamó Llorente

maria.salamo@ub.edu

Dept. Mathematics and Informatics,
Faculty of Mathematics and Informatics,
University of Barcelona (UB)

Introduction to Machine Learning

Unsupervised Learning

Cluster Analysis

Factor Analysis

Visualization

Supervised Learning

Non Linear Decision

Linear Decision

Decision Learning Theory

Basic concepts of Decision Learning Theory

K-Means,
Fuzzy C-means
EM

PCA, ICA

Self Organized Maps (SOM) ,
Multi-Dimensional Scaling

Lazy Learning
(K-NN, IBL, CBR)

Overfitting,
model selection and feature selection

Kernel Learning

Ensemble Learning
(NN, Trees, Adaboost)

Perceptron,
SVM

Bias/Variance
,
VC dimension,
Practical advice of how to use learning algorithms

Contents

1. Introduction to unsupervised learning (Theory 2)

1. Introduction to unsupervised learning
2. Examples
3. Definition of unsupervised learning
4. Unsupervised learning approaches

2. Introduction to Cluster analysis (Theory 2)

1. Defining clustering analysis
2. Areas that apply clustering
3. Classification of clustering algorithms

3. Hierarchical clustering (Theory 2)

4. Partitional clustering (Theory 3)

1. K-Means algorithm,
2. Bisecting K-Means,
3. Fuzzy C-Means
4. EM (expectation maximization algorithm)

Introduction to unsupervised learning

Contents

1. Introduction to unsupervised learning (Theory 2)

1. Introduction to unsupervised learning
2. Examples
3. Definition of unsupervised learning
4. Unsupervised learning approaches

2. Introduction to Cluster analysis (Theory 2)

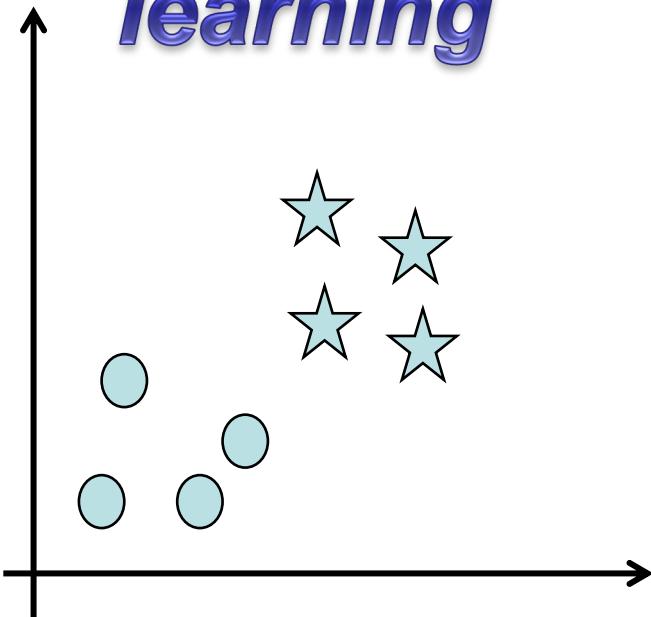
1. Defining clustering analysis
2. Areas that apply clustering
3. Classification of clustering algorithms

3. Hierarchical clustering (Theory 2)

4. Partitional clustering (Theory 3)

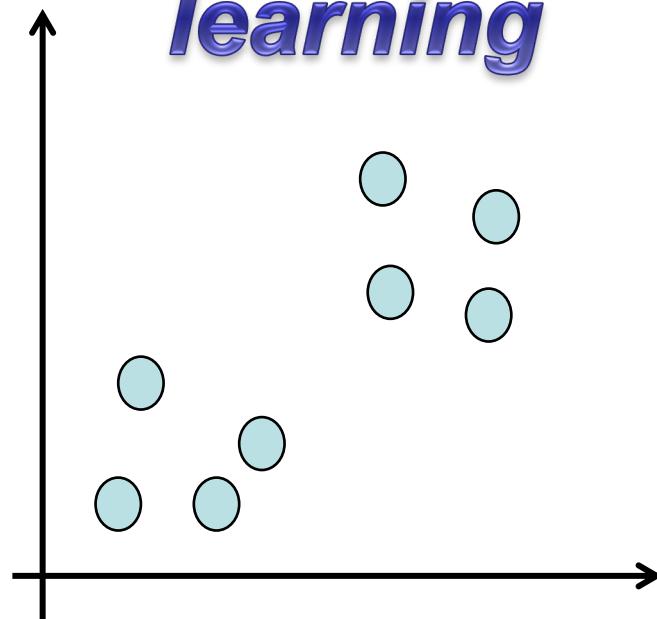
1. K-Means algorithm,
2. Bisecting K-Means,
3. Fuzzy C-Means
4. EM (expectation maximization algorithm)

Supervised learning



Inferring a function
from labelled
training data

Unsupervised learning



Try to find hidden
structure in
unlabeled data



<https://youtu.be/ukzFl9rgwfU>

There are two **main goals** in unsupervised learning

1. **Summarization**: To obtain representations that describe an unlabeled dataset
 2. **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:
 - Relational (hierarchies)
 - Unstructured (partitions)

Examples of unsupervised learning

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



<https://theappsolutions.com/blog/development/unsupervised-machine-learning/>

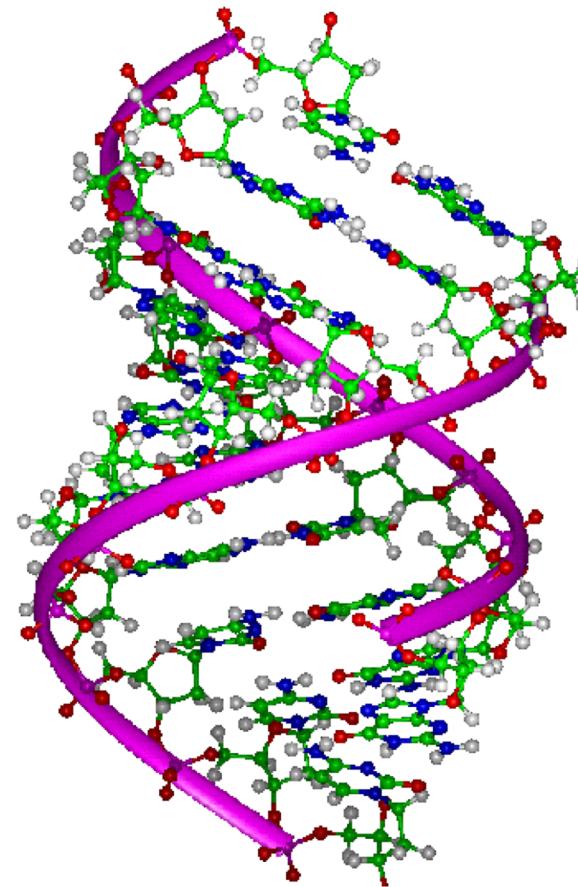
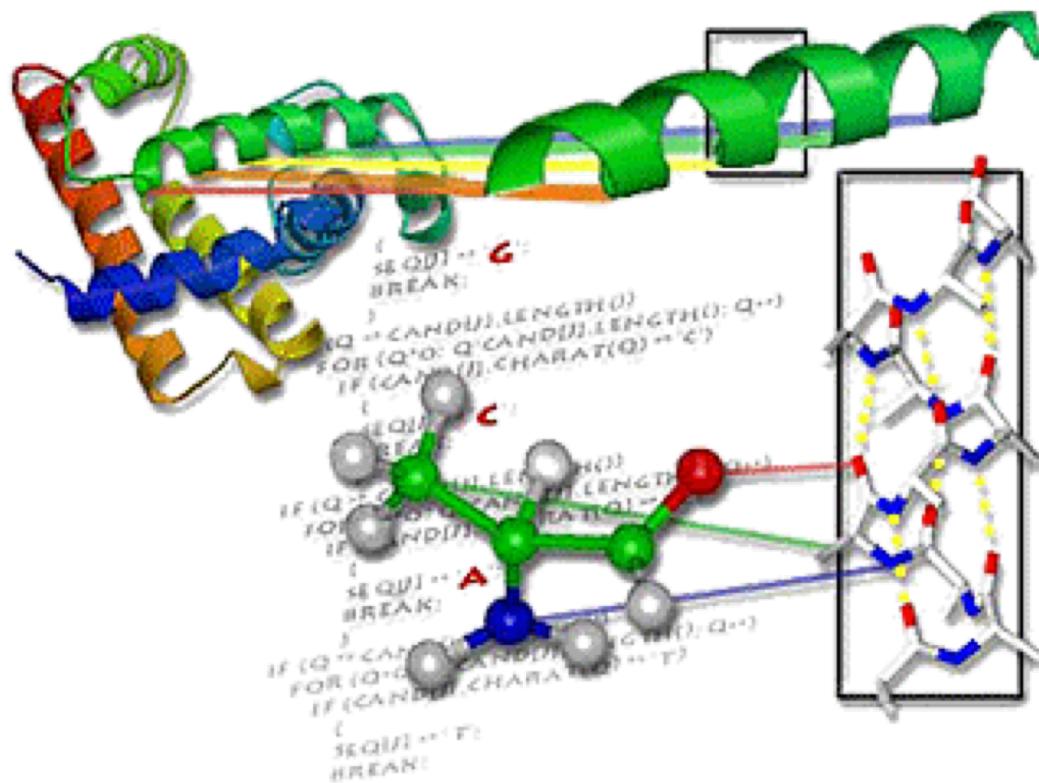
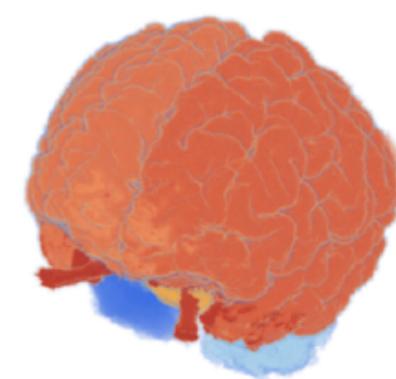
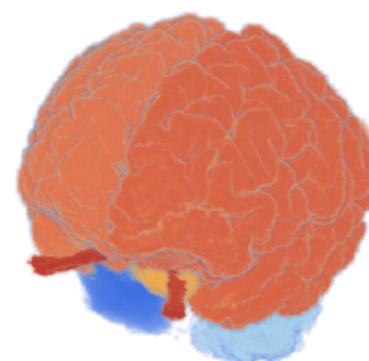
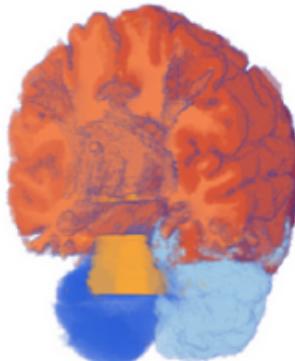
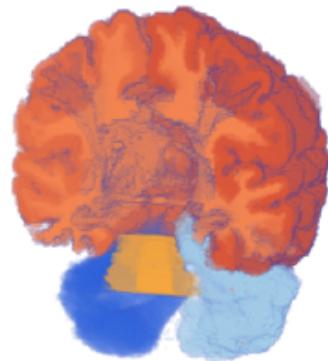


Illustration by Brian Haas, Phillips lab

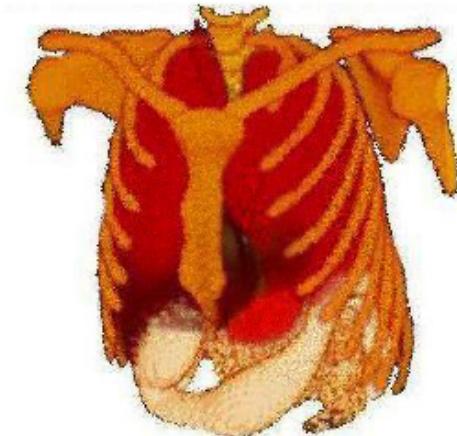
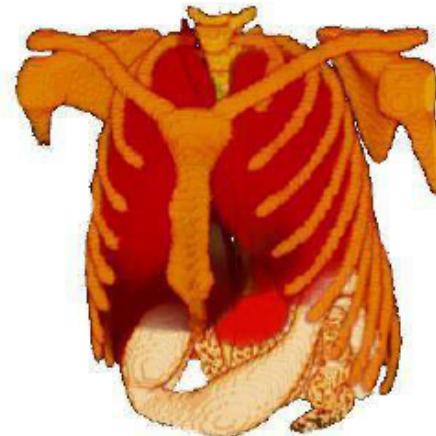
- **Objective**
 - To extract or recognize patterns that share common characteristics
- **It could be:**
 - **Sequence analysis** (also known as sequence alignment)
 - Clustering is used to group homologous sequences into gene families
 - **Genome annotation** (i.e., gene finding)



(a)



(b)

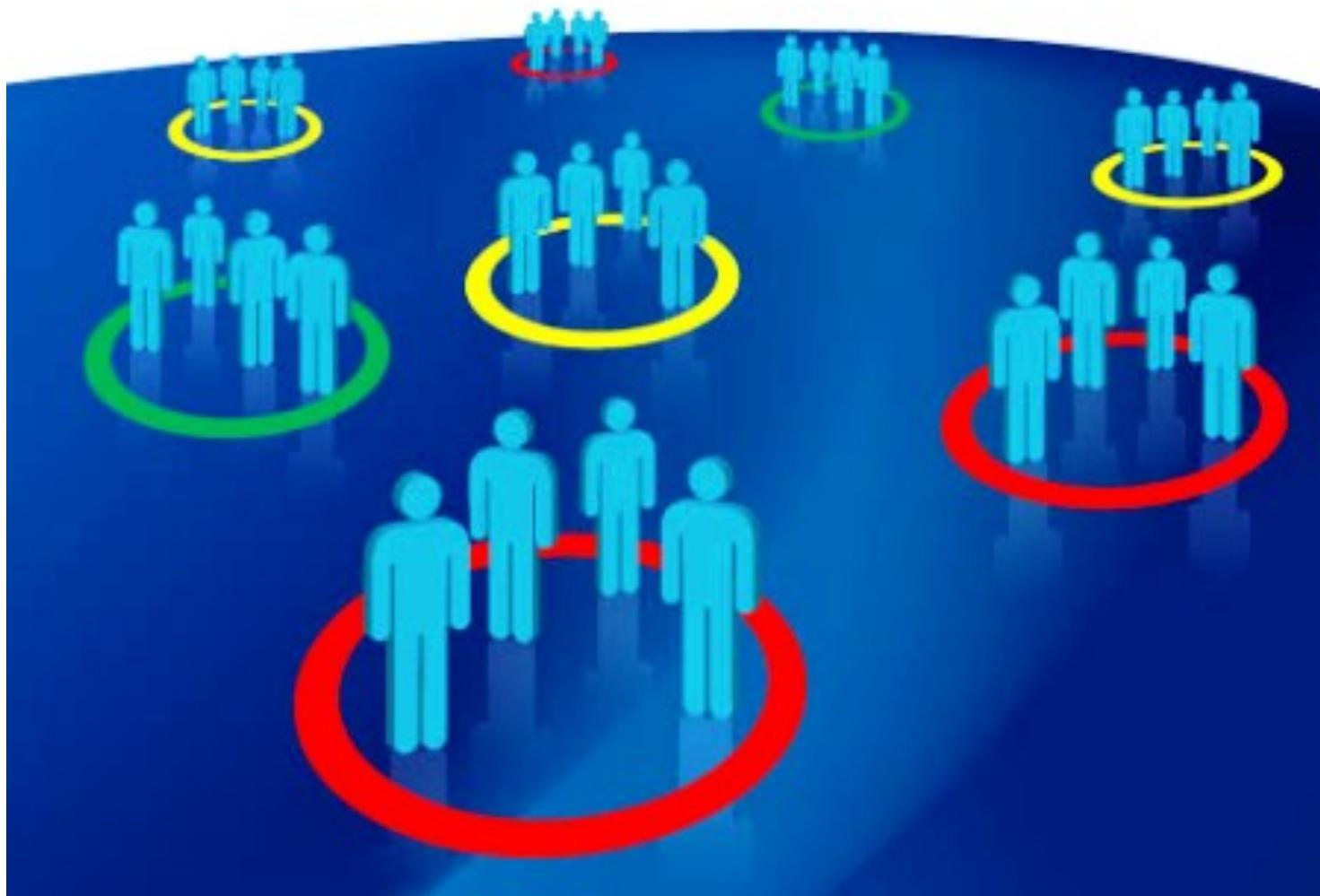


(c)

- Objective
 - On PET scans, cluster analysis can be used to differentiate between different types of tissue and blood in a three dimensional image
- Note that in this application, actual position does not matter, but the voxel intensity is considered as a vector, with a dimension for each image that was taken over time.
- This technique allows, for example, accurate measurement of the rate a radioactive tracer is delivered to the area of interest, without a separate sampling of arterial blood, an intrusive technique that is most common today.

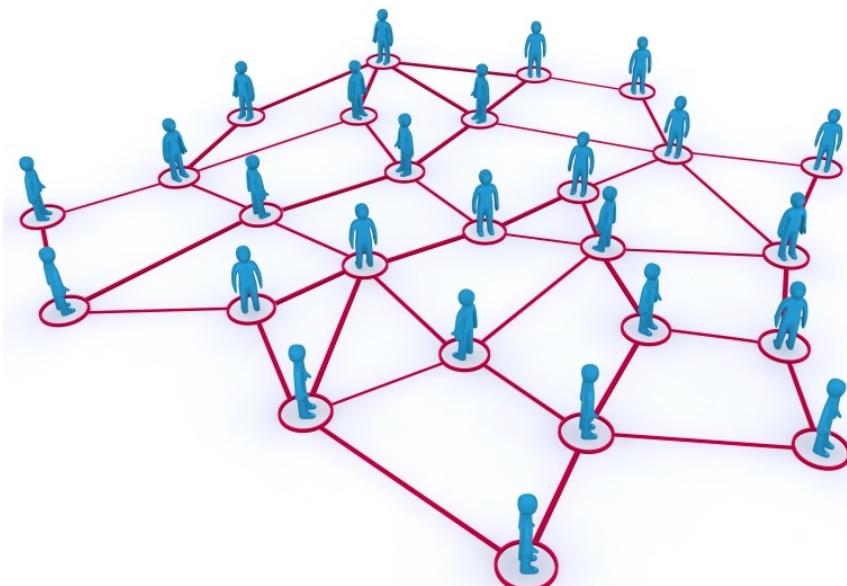


Market research



- **Objective**
 - Obtain valuable information about your customers and potential customers because it is essential to the success of any business
- **Market research data can be used**
 - Before starting any new business
 - When expanding an existing one
- **The idea is to perform an extensive market research to establish whether there is a need for your product or service**

Social network analysis



In the study of social networks, clustering may be used to recognize communities within large groups of people.



- **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.

- **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

- **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**)
 - ...

- **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

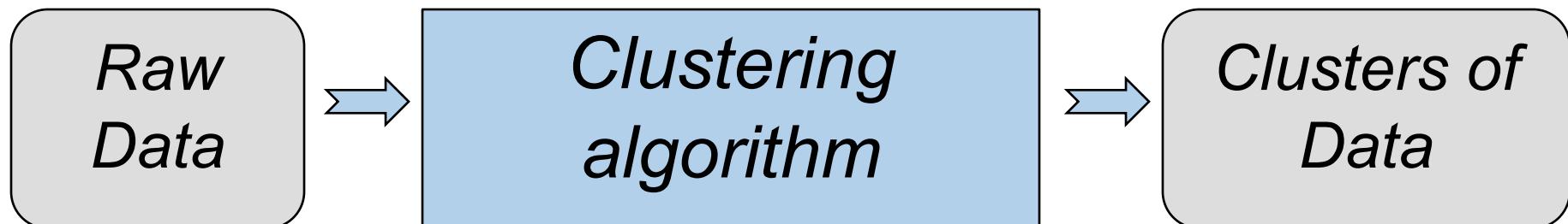
Introduction to cluster analysis

Contents

- 1. Introduction to unsupervised learning (Theory 2)**
 1. Introduction to unsupervised learning
 2. Examples
 3. Definition of unsupervised learning
 4. Unsupervised learning approaches
- 2. Introduction to Cluster analysis (Theory 2)**
 1. Defining clustering analysis
 2. Areas that apply clustering
 3. Classification of clustering algorithms
- 3. Hierarchical clustering (Theory 2)**
- 4. Partitional clustering (Theory 3)**
 1. K-Means algorithm,
 2. Bisecting K-Means,
 3. Fuzzy C-Means
 4. EM (expectation maximization algorithm)

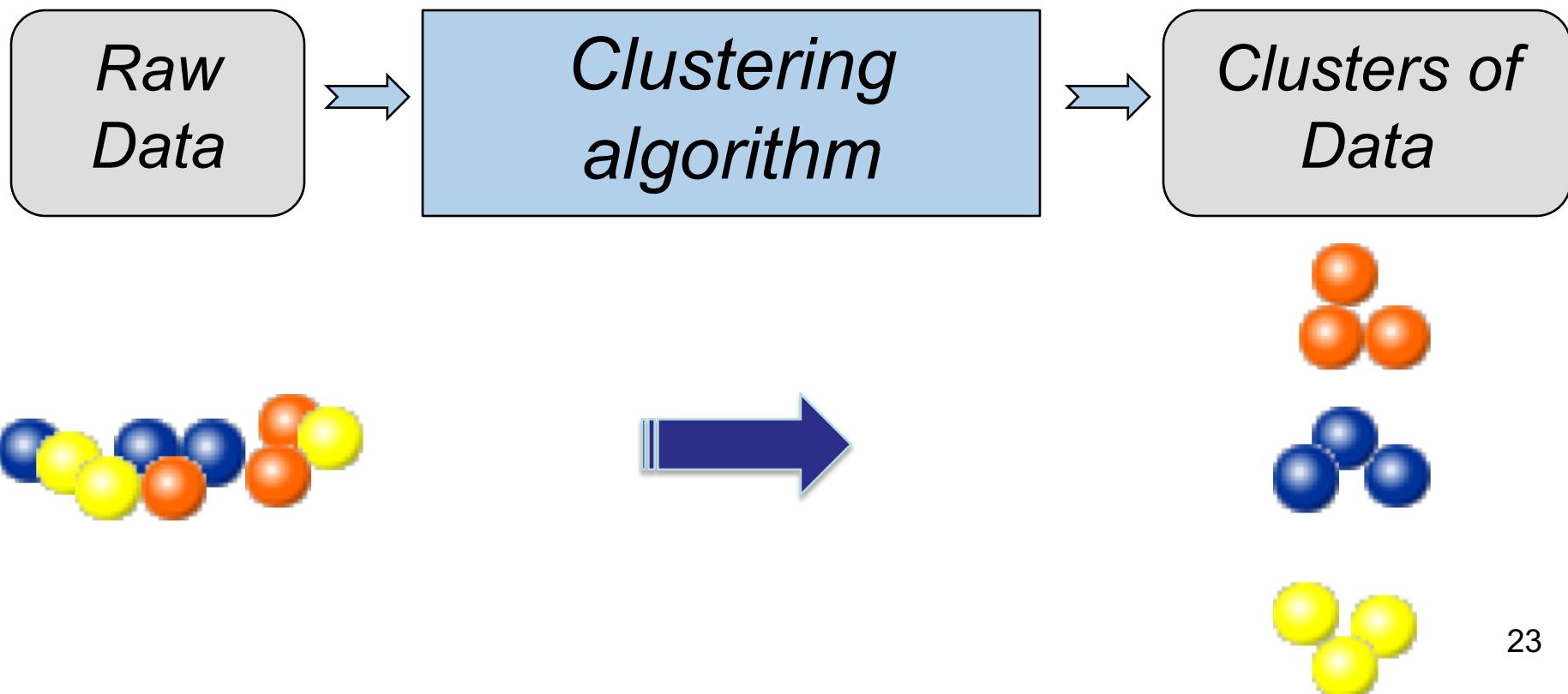
Defining clustering analysis

- **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** (in the example below, based on the colour)



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?



Take a moment to think about it

....

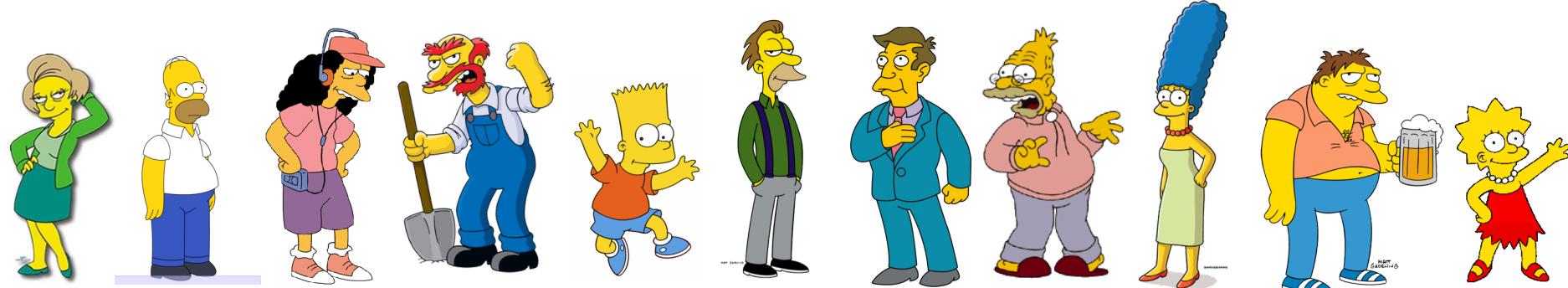
*think a bit about the number of clusters
you see in the example*

(You will have the answer in the theoretical class)

- 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



What is a natural grouping?



Decide how to group them !!!

Take a moment to think about it

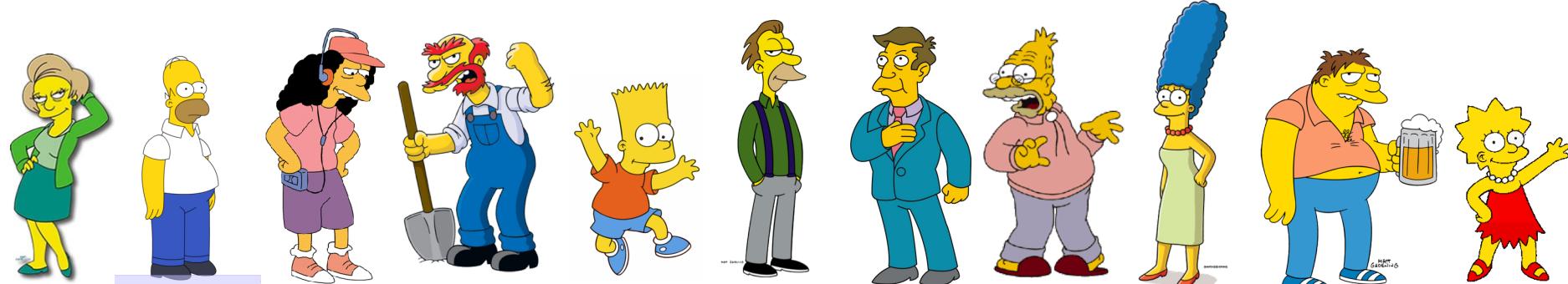
....

Think in a way to group them and make a note about it

(You will have the answer in the theoretical class)



What is a natural grouping?



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

Take a moment to think about it

....

Think in another way to group them and make an additional note about it

(You will have the answer in the theoretical class)





What is similarity?

Take a moment to think about it

....

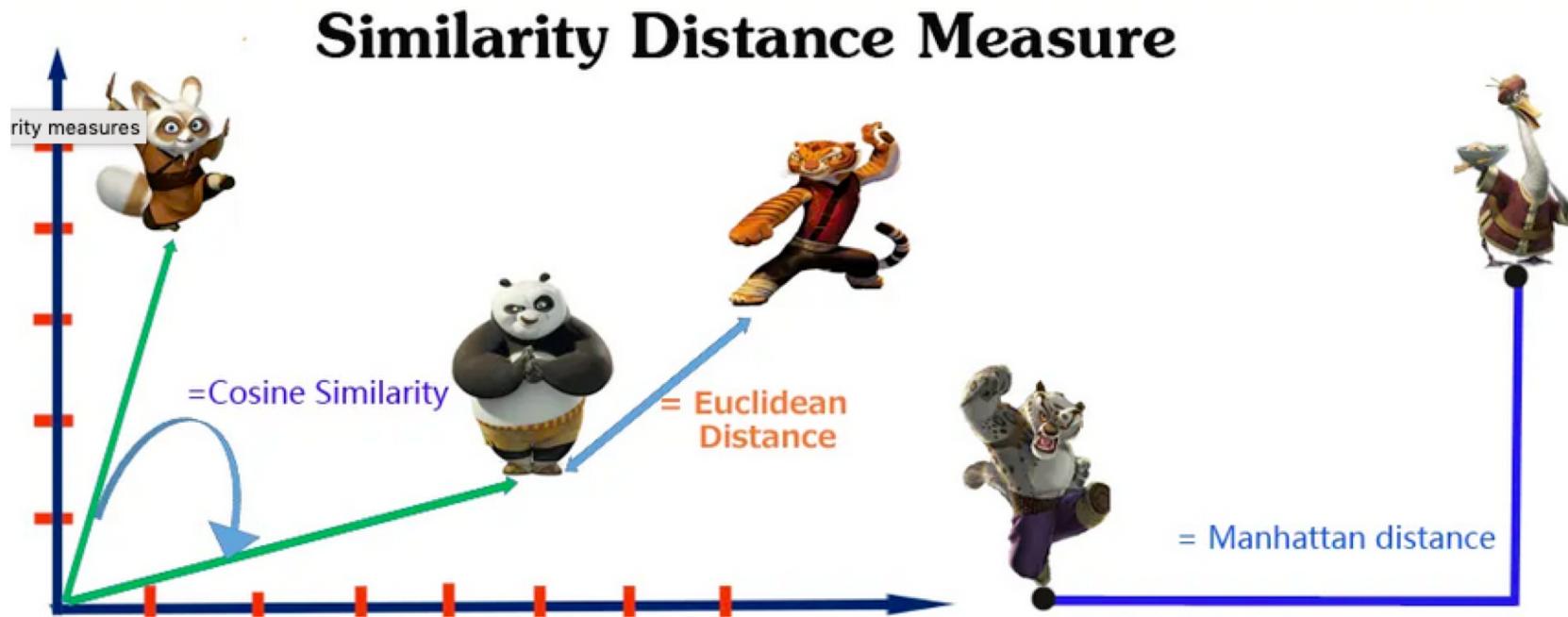
How do you consider there is great similarity between two objects

(You will have the answer in the theoretical class)



Similarity measures in ML

Similarity in ML is computed using distance measures



There are many in ML, here you will find some examples



<https://dataaspirant.com/five-most-popular-similarity-measures-implementation-in-python/>

- $D(A, B) = D(B, A)$ **Symmetry**
 - Otherwise you could claim “Alex looks like bob but bob looks like nothing like Alex”
- $D(A, A) = 0$ **Constancy of Self-Similarity**
 - Otherwise you could claim “Alex looks more like bob than bob does
- $D(A, B) = 0 \mid \text{if } A=B$ **Positivity Separation**
 - Otherwise there are objects in your world that are different but you cannot tell apart
- $D(A, B) \leq D(A, C) + D(B, C)$ **Triangular Inequality**
 - Otherwise you could claim “Alex is very like Bob, and Alex is very like Carl, but Bob is very unlike Carl”



Distance measure: Minkowski Metric

- Suppose two objects x and y both have p features

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

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

- The Minkowski metric is defined by

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

- Most Common Minkowski Metrics

1, $r = 2$ (Euclidean distance)

$$d(x, y) = \sqrt[2]{\sum_{i=1}^p |x_i - y_i|^2}$$

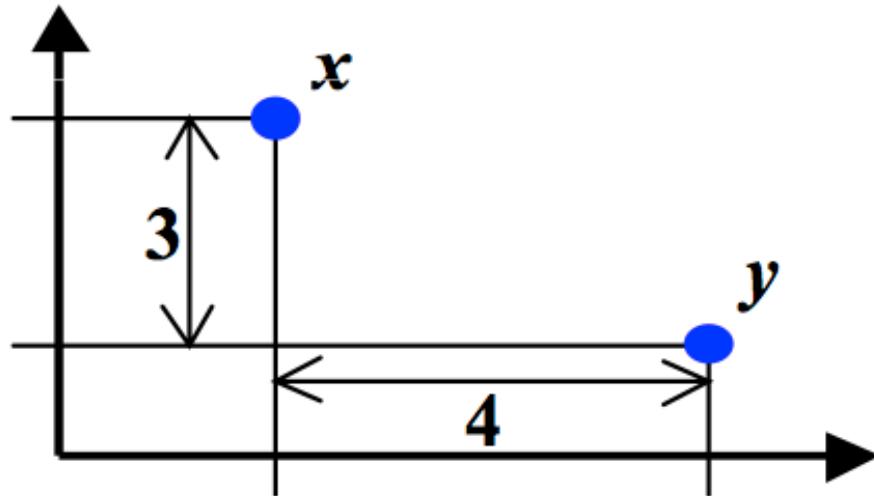
2, $r = 1$ (Manhattan distance)

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

3, $r = +\infty$ ("sup" distance)

$$d(x, y) = \max_{1 \leq i \leq p} |x_i - y_i|$$

An example



1: Euclidean distance : $\sqrt{4^2 + 3^2} = 5.$

2: Manhattan distance : $4 + 3 = 7.$

3: "sup" distance : $\max\{4, 3\} = 4.$

- Pearson correlation coefficient:

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

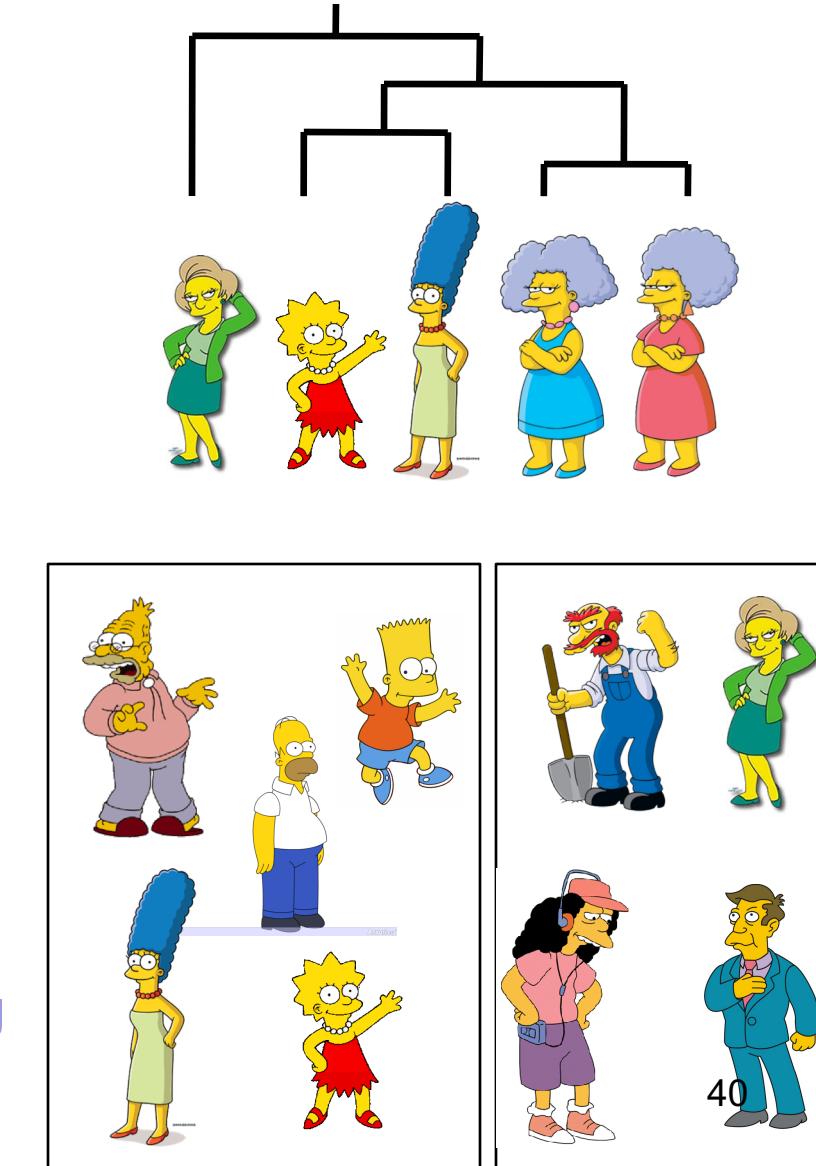
where $\bar{x} = \frac{1}{p} \sum_{i=1}^p x_i$ and $\bar{y} = \frac{1}{p} \sum_{i=1}^p y_i$.

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

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

Contents

1. Introduction to unsupervised learning (Theory 2)

1. Introduction to unsupervised learning
2. Examples
3. Definition of unsupervised learning
4. Unsupervised learning approaches

2. Introduction to Cluster analysis (Theory 2)

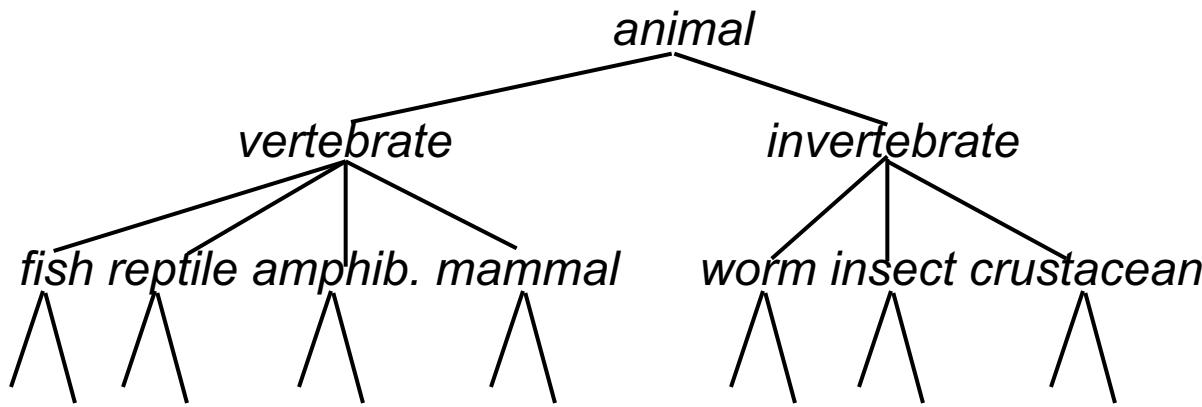
1. Defining clustering analysis
2. Areas that apply clustering
3. Classification of clustering algorithms

3. Hierarchical clustering (Theory 2)

4. Partitional clustering (Theory 3)

1. K-Means algorithm,
2. Bisecting K-Means,
3. Fuzzy C-Means
4. EM (expectation maximization algorithm)

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

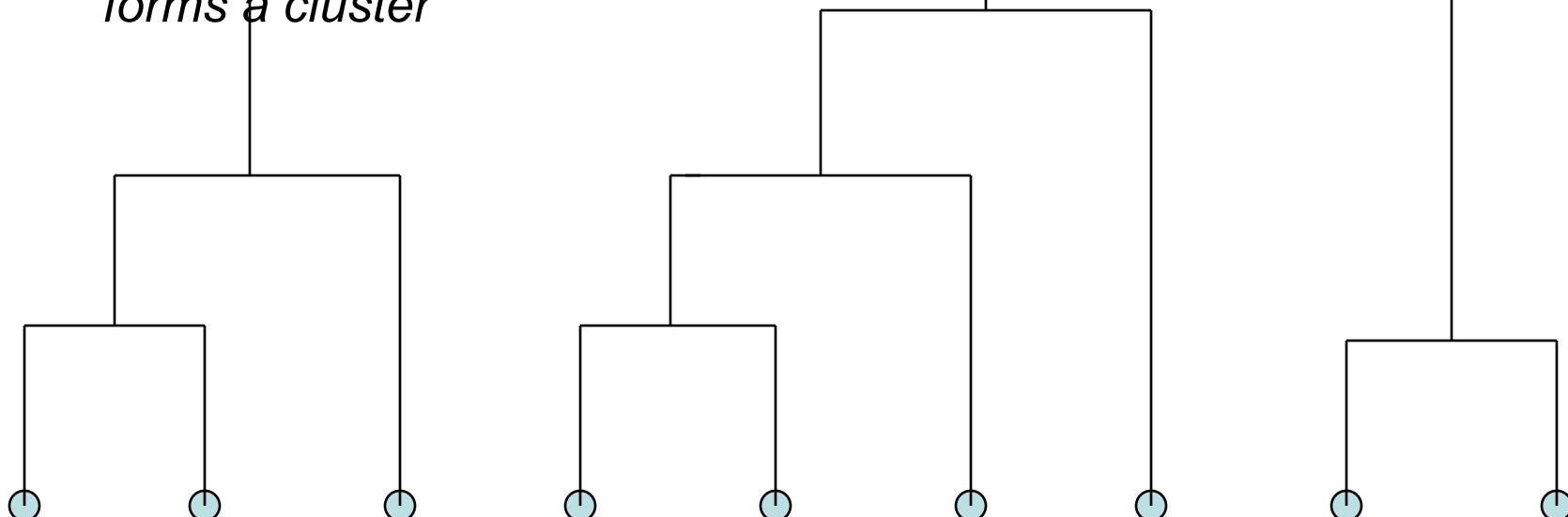


- Recursive application of a standard clustering algorithm can produce a hierarchical clustering

Dendrogram: Shows How Clusters are Merged

Decompose data objects into a several levels of nested partitioning (tree of clusters), called a dendrogram

A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster



Basic approaches for generating a hierarchical clustering

- ***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



<https://youtu.be/EUQY3hL38cw>

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

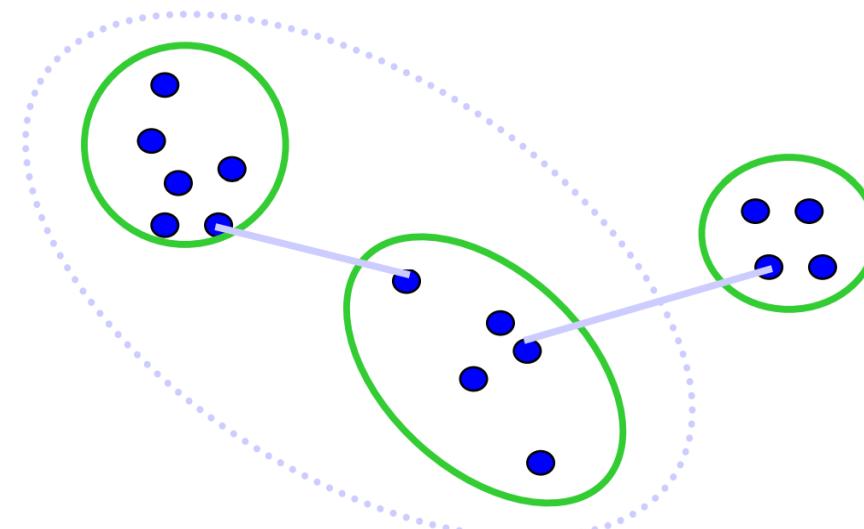
Cluster Similarity

- Assume a similarity function that determines the similarity of two instances: $\text{sim}(x,y)$
 - 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

- Use maximum similarity of pairs:

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

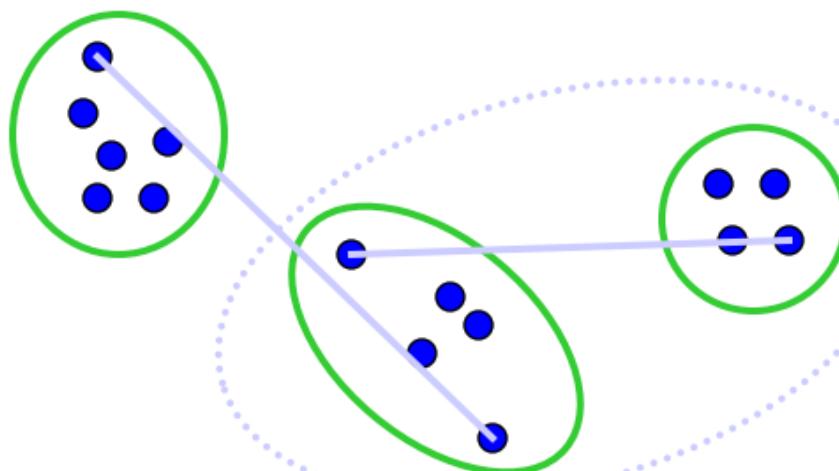
- Can result in “straggly” (long and thin) clusters due to *chaining effect*.
 - Appropriate in some domains, such as clustering islands



- 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:**

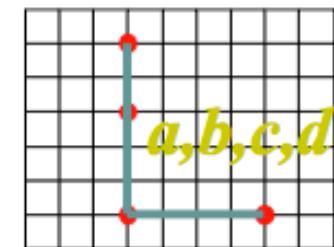
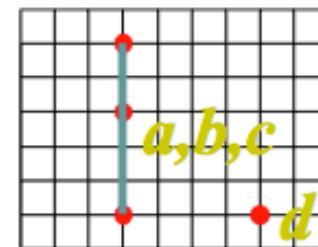
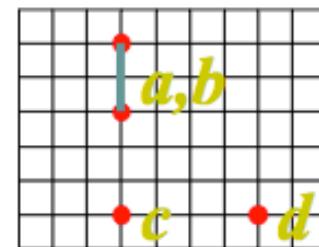
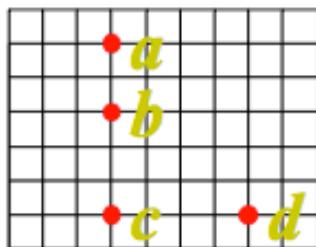
$$\text{sim}((c_i \cup c_j), c_k) = \max(\text{sim}(c_i, c_k), \text{sim}(c_j, c_k))$$

- **Complete-Link:**

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

Single-Link Example

Euclidean Distance



(1)

(2)

(3)

	b	c	d
a	2	5	6
b		3	5
c			4

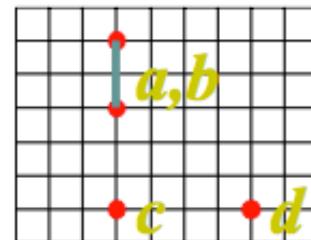
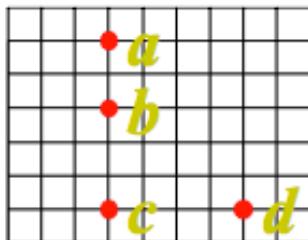
	b	c	d
a	2	5	6
b		3	5
c			4

	d
a, b, c	4

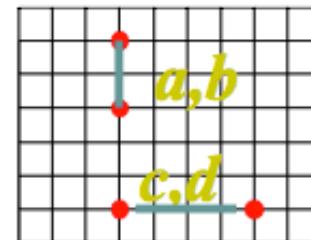
Distance Matrix

Complete-Link Example

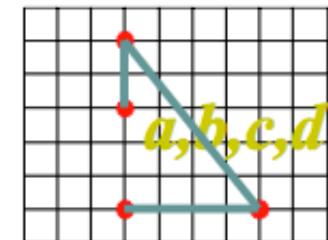
Euclidean Distance



(1)



(2)



(3)

	<i>b</i>	<i>c</i>	<i>d</i>
<i>a</i>	2	5	6
<i>b</i>		3	5
<i>c</i>			4

A distance matrix showing the initial distances between points. The diagonal values are 0. The matrix is:

	<i>b</i>	<i>c</i>	<i>d</i>
<i>a</i>	2		
<i>b</i>		3	
<i>c</i>			4

The matrix after pairing *a* and *b*. The value 2 is circled in green, and the row and column for *a* and *b* are highlighted in blue. The matrix is:

	<i>c</i>	<i>d</i>
<i>a, b</i>	5	6
<i>c</i>		4

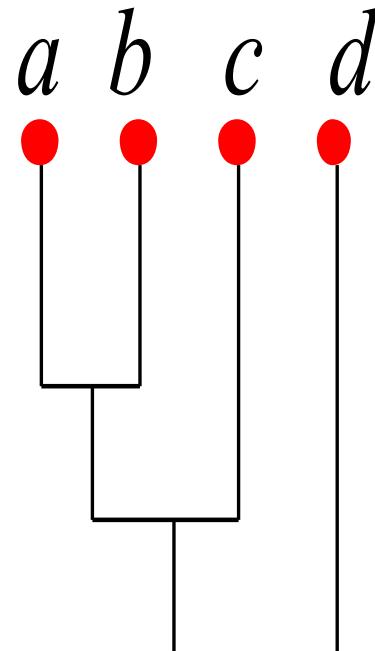
The final distance matrix showing the total distance between the two clusters. The value 6 is circled in green. The matrix is:

	<i>c, d</i>
<i>a, b</i>	6

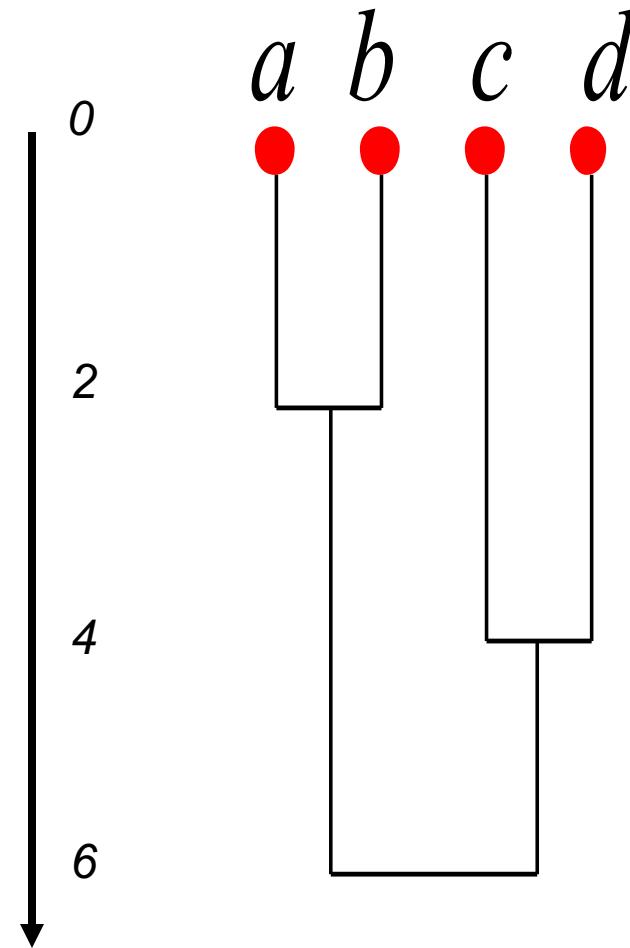
Distance Matrix

Dendograms Comparison

Single-Link



Complete-Link

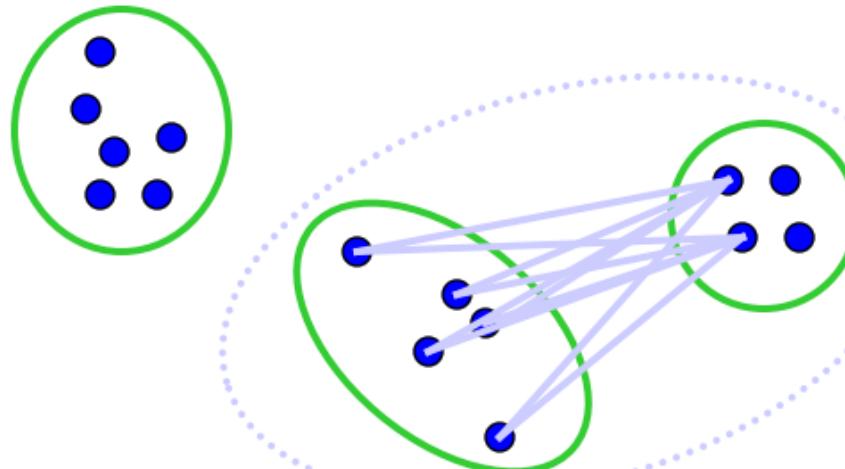


Group Average Agglomerative Clustering

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

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

- Compromise between Single and Complete link.



- 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:

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

Arriving at this point:



TO READ

Chapter 8 - Cluster Analysis: Basic concepts and algorithms

- Pages 487, 488, 489, 490
- Section 8.1: Overview
 - Section 8.1.1
 - Section 8.1.2
 - Section 8.1.3 (optional)
- Section 8.3: Agglomerative Hierarchical Clustering
 - Read all the subsections



Week 2

Course. Introduction to Machine Learning

Theory 2. Introduction to unsupervised learning and Cluster Analysis (Part I)

Dr. Maria Salamó Llorente

maria.salamo@ub.edu

Dept. Mathematics and Informatics,
Faculty of Mathematics and Informatics,
University of Barcelona (UB)