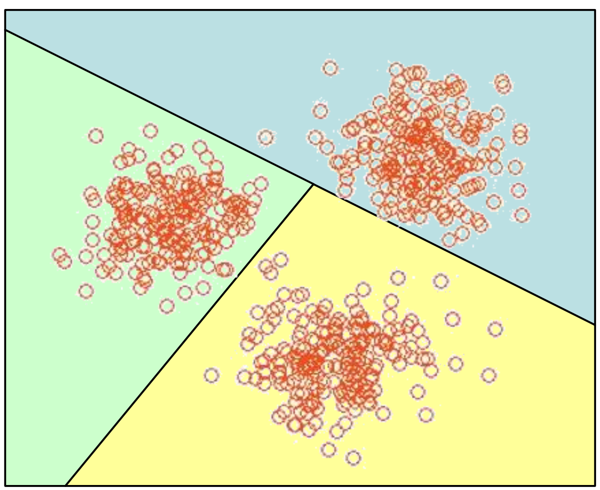
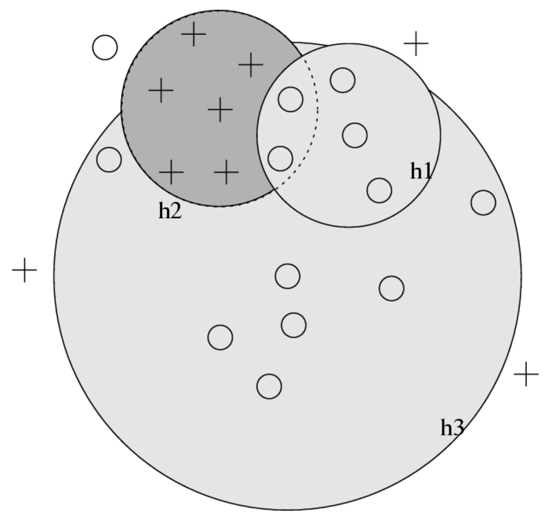
Clustering

May 2008

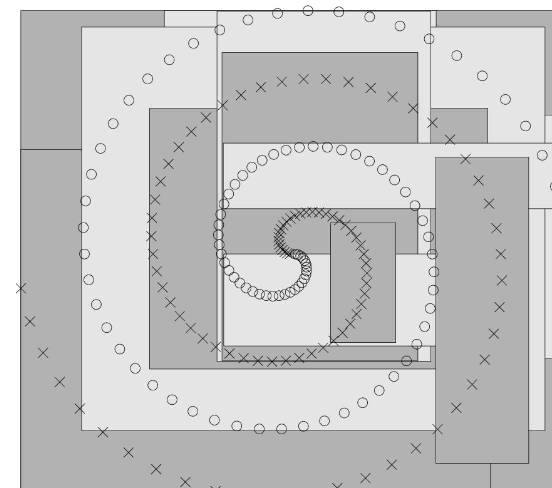
Arnaud QUIRIN – aquirin@gmail.com



Clustering non supervised



With constraints



A challenge : the spiral

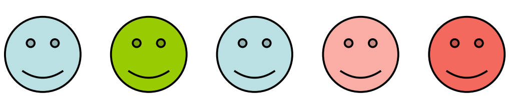
Objectives

- Non-supervised learning
- Goal : to organise objects in groups (= clusters)
- We need a similarity measure

Clustering (categorisation) $\neq$ Classification

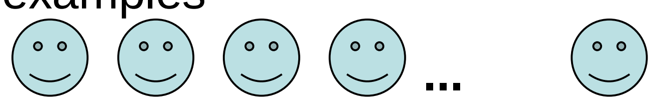
Clustering (categorisation)

- Non supervised
- The algorithm needs only a similarity measure



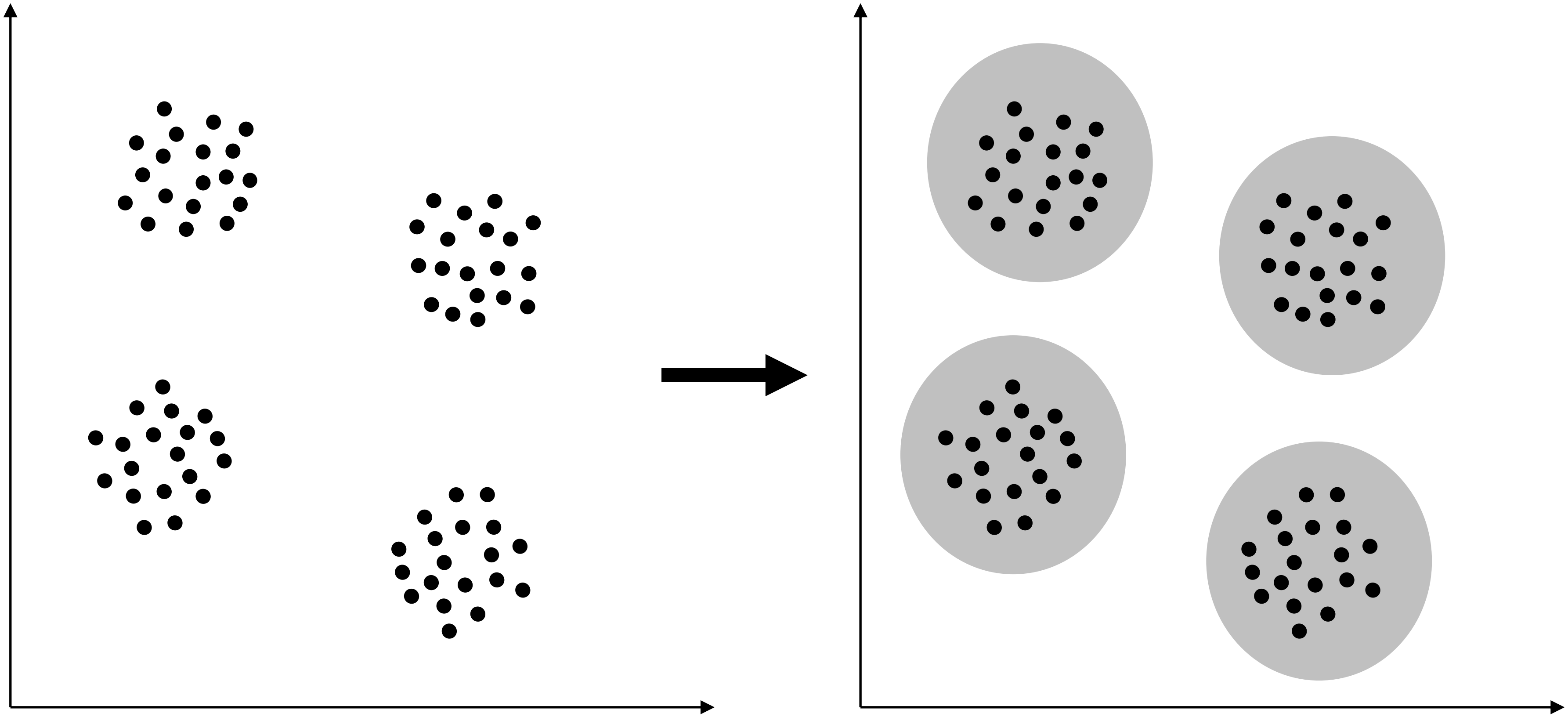
Classification

- Supervised
- It needs *already classified* examples



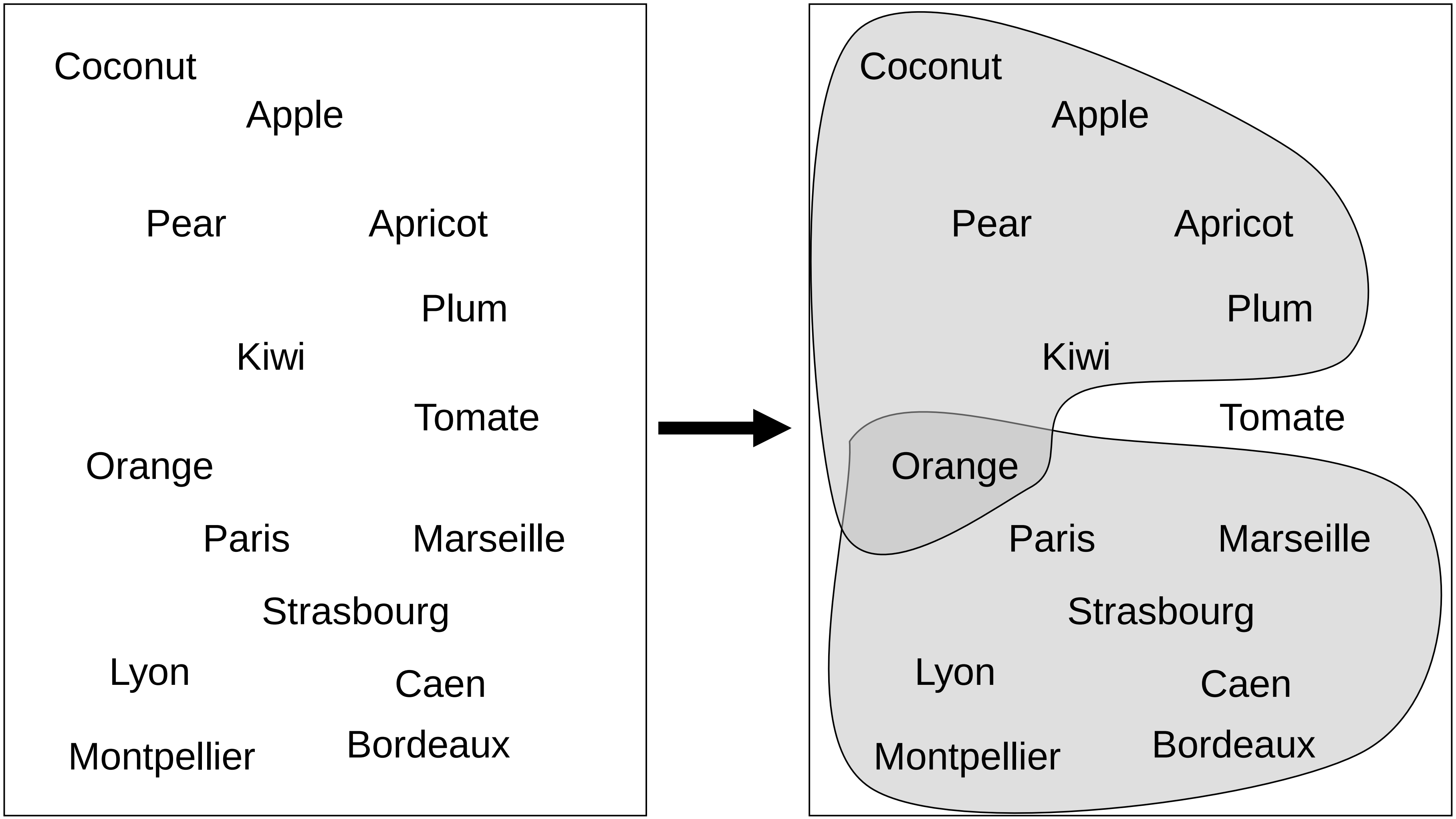
A A B C … ?

Example



Based on the distance

Example



Based on the concept

Goal of the categorisation

- Group the data into a homogeneity criterion
- Criterion complex to define, can depend on :
  - the data
  - the target application
  - the subjectivity of the user

A good clustering ?

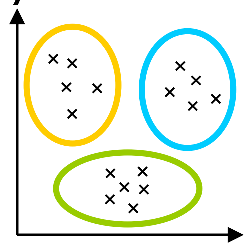
- Produce categories with a high quality
  - The intra-cluster similarity should be large
  - The inter-cluster similarity should be low

- The quality of the results depends on
  - The similarity measure and its implementation
  - The definition and the representation of a cluster

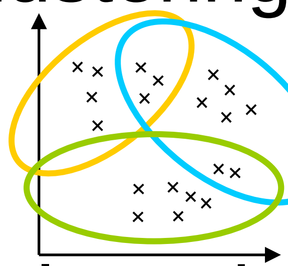
- The method could be evaluated using its ability to discover hidden patterns

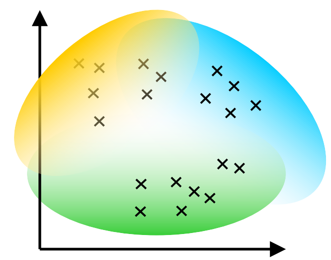
Several kinds

- Exclusive clustering, hard clustering

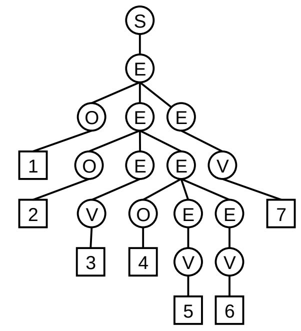


- Soft, fuzzy clustering



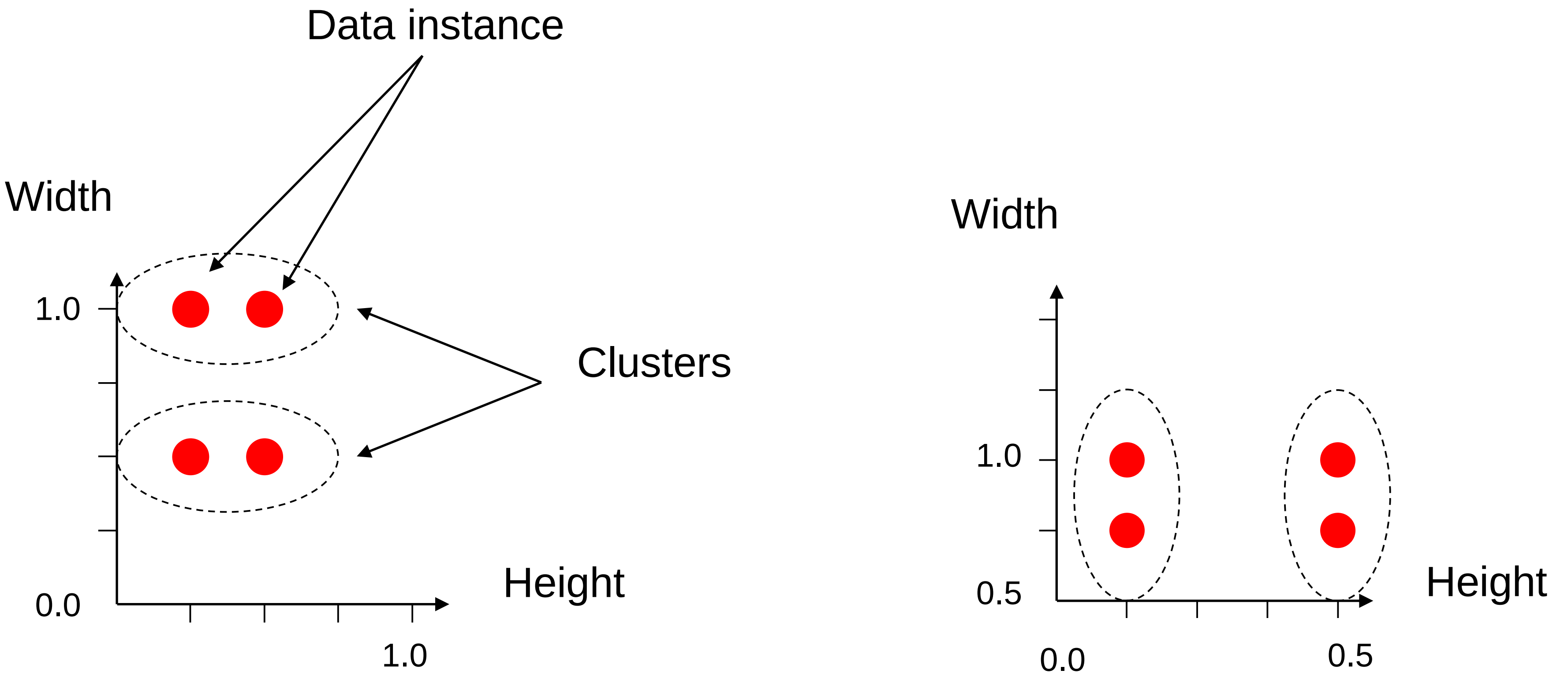


- Hierarchical clustering



- Probabilistic clustering

Distance measure



Before scaling

After scaling

Measures

Notation

- $\{ o_0, ..., o_N \}$: N data samples
- K : number of the clusters
- g_k : *gravity center* of the cluster C_k
- σ_k : variance of the cluster C_k
- σ : variance of the full dataset

Minkowski measure

$$d_p(o_i, o_j) = \left(\sum_{k=1}^{d} \left| o_{i,k} - o_{j,k} \right|^p \right)^{\frac{1}{p}}$$

p=1 Manhattan distance

p=2 Euclidian distance

Used for : data with a large number of attributes (d >> 3)

Distance measure

Criterion	Formula	Summary
Intra-cluster inertia	$I_t = \sum_{k=1}^{K} \sum_{i \in C_k} \mathrm{dist}(o_i, g_k)^2 = \sum_{k=1}^{K} \sigma_k$	Variance between the samples and the gravity center of their clusters
Compacity	$Cmp = \frac{1}{K} \cdot \sum_{k=1}^{K} \frac{\sigma_k}{\sigma}$	Grouping degree
Xie-Beni criterion (1991)	$XB = \frac{I_t}{N \cdot \min_{i,j \in K, i \neq j}(\mathrm{dist}(g_j, g_i))}$	Measure of the separation of the clusters, scale independent
Wemmert-Gançarski criterion (1999)	$WG = \frac{I_t}{N \cdot \min_{o_i \in C_k, k' \neq k}(\mathrm{dist}(o_i, g_{k'}))}$	Separation and compacity of the clusters, scale independent

Hard-Clustering : K-Means

K-Means (MacQueen, 1967)

Goal : minimize ***J***

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \underbrace{\left\| x_i^{(j)} - c_j \right\|^2}$$

Distance measure between a data $x_i^{(j)}$ and the center of the cluster c_j

J is the distance between the *n* data points from the centers of their respective clusters

K-Means

The algorithm :

K-MEANS (K)

1. Place randomly K centers (centroids) in the space of the objects we have to categorize. Each cluster is represented by its corresponding centroid.

DO

2a. Assign to each object the cluster for which its centroid is the closest one

2b. When all the objects have be assigned, recompute all the K locations of the centroids, using the barycenter

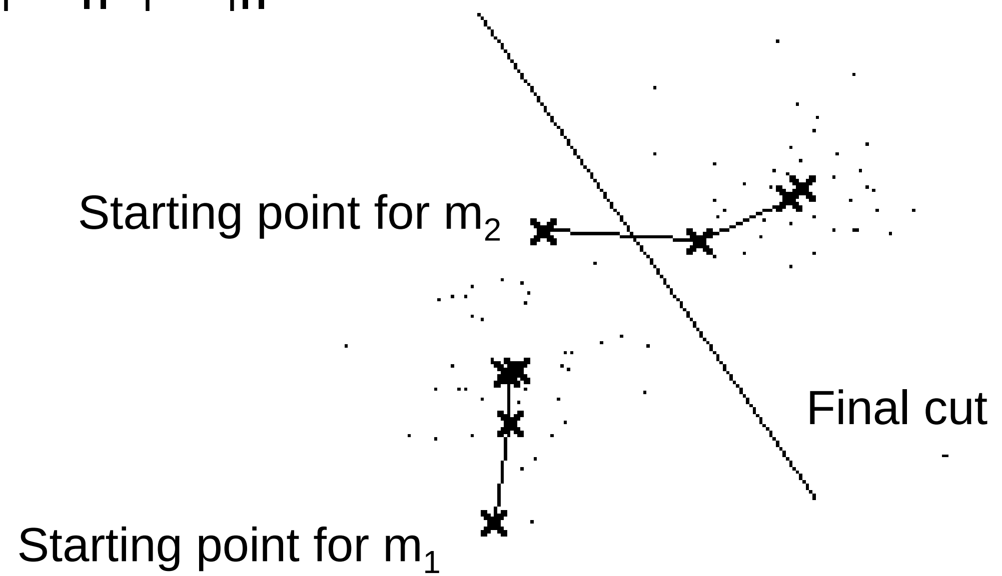
UNTIL the locations converge

K-Means

Justification :

- N data sample : $[X_1, ..., X_n]$
- K clusters, with k<n
- m_i is the barycenter of the examples of the cluster i

X_i is in C_i if $||x_i-m_i||$ is minimal.



Soft-Clustering : Fuzzy C-Means

Fuzzy C-Means (FCM) [Dunn, 1973 ; Bezdek, 1981]

- Now, a data can belong to two clusters or more
- Used very frequently for pattern recognition (ex: OCR)
- Goal : minimize this objective function $\boldsymbol{J_m}$

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \left\| x_i - c_j \right\|^2, \quad 1 \leq m < \infty$$

Soft-Clustering : Fuzzy C-Means

FUZZY-C-MEANS (K, m)

- Initialize randomly a matrix $U=[u_{ij}]$ $U^{(0)} = U$ (membership matrix)
- At the step k, do
  - Compute the centroids $C^{(k)} = [c_j]$ using $U^{(k)}$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m x_i}{\sum_{i=1}^{N} u_{ij}^m}$$

  - Update $U^{(k)}$, which becomes $U^{(k+1)}$

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{\frac{2}{m-1}}}$$

While $|| U^{(k+1)} - U^{(k)} || > \varepsilon$

Soft-Clustering : Fuzzy C-Means

The membership matrix $U = [u_{ij}]$

The K clusters

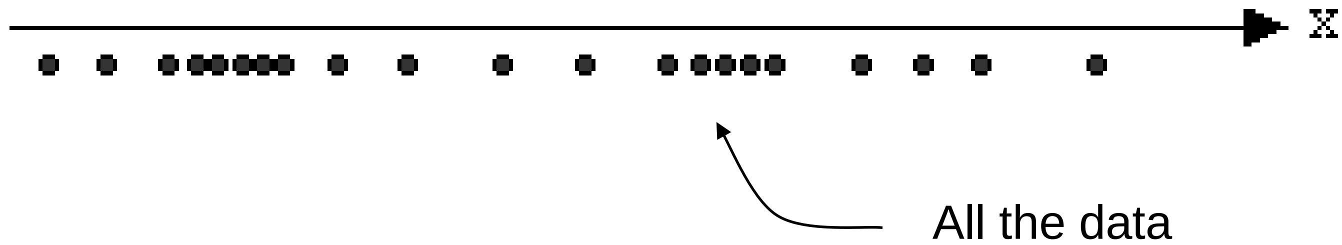
The N data points

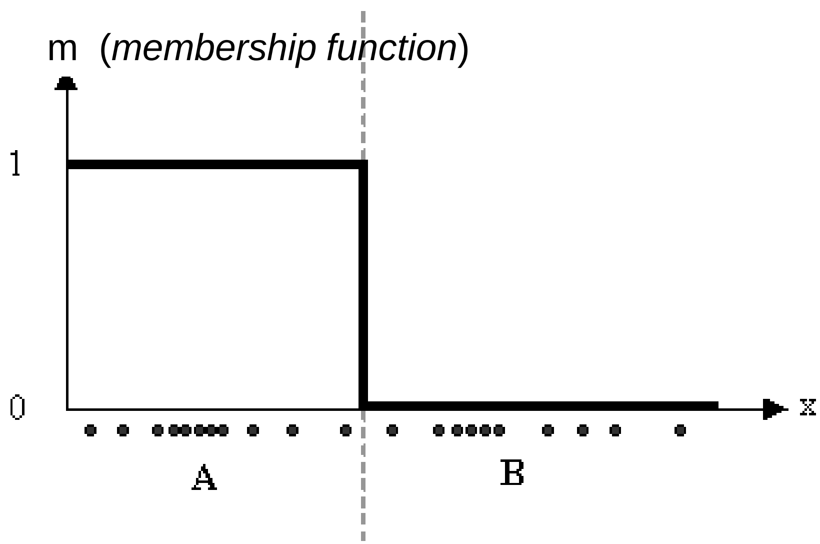
$$\begin{pmatrix} 0.1 & 0.3 & 0.4 & 0.2 \\ 0.1 & 0.0 & 0.3 & 0.6 \\ 0.2 & 0.4 & 0.0 & 0.4 \\ 0.4 & 0.0 & 0.1 & 0.5 \\ 0.4 & 0.4 & 0.1 & 0.1 \\ 0.6 & 0.3 & 0.1 & 0.0 \end{pmatrix}$$

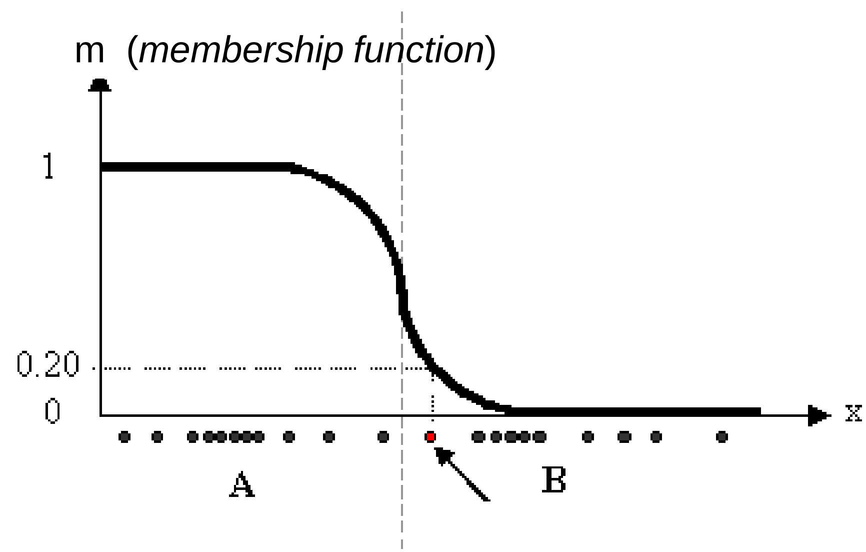
The data '4' belongs at 50% to the cluster C=4

Comparison with K-Means

Example :







K = 2

Hierarchical Categorization

Basic concepts from (S.C. Johnson,1967)

Given :

- N examples,
- A similarity matrix between all the examples N*N

Algorithm :

1. Start to assign each data into its own cluster. We define the distance between the clusters as the distance between the data they represent

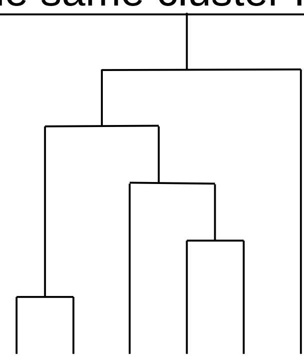
DO

2a. Find the closest pair of clusters, and join them into the same cluster

2b. Compute the distance *d* between the remaining clusters (the new one, and all the remainings)

WHILE all the examples are not in the same cluster R (root), having the size N

Sortie :



Etape 6 : 1 clusters

Etape 5 : 2 clusters

Etape 4 : 3 clusters

Etape 3 : 4 clusters

Etape 2 : 5 clusters

Etape 1 : 6 clusters

Hierarchical Categorization

How to compute the distance *d(i,j)* ?

single-linkage (ou minimum) :

d is the shortest distance between any element in the cluster *i* and any element in the cluster *j*.

complete-linkage (ou diameter) :

d is the biggest distance between any element in the cluster *i* and any element in the cluster *j*.

average-linkage :

d is the average distance between the elements of the two clusters.

UCLUS Method [D'Andrade 1978] :

Use of the median, and not the mean.

$$[0.0\ \ 0.1\ \ 0.1\ \ \underline{0.1}\ \ 0.3\ \ 0.4\ \ 2.5]$$

median

mean = 0.5

Advantage : more robust for datasets having a lot of *outliers*.

Hierarchical Categorization

Grouping categorization (ex: Johnson, 1967) :
join iteratively the clusters

$$\neq$$

Spliting categorization (really rare) :
start from one big cluster, and split it

Hierarchical Categorization

Algorithm of (Johnson, 1967)

- Proximity matrix D = [d(i,j)] for two objects *i* and *j*.
- d(i,i) = 0
- d(i,j) = d(j,i)
- D : square matrix (NxN)
- Sequence of n clusters : (0), (1), ..., (n-1)

  i, index of the cluster

  (i), the content of the cluster I

- L(k) : k-th level of the categorization

  (the distance intra-cluster are represented with a horizontal edge in the tree)

- Proximity between (a) and (b) : d[(a),(b)]

Hierarchical Categorization

Algorithm of (Johnson, 1967)

HIERARCHICAL-CLUSTERING (D)

Put all the objects into a cluster. L(c) = 0 for any cluster c. m = 0

DO

- Find the pair of clusters (a) and (b) the most similar. d[(a),(b)] = min d[(i)(j)] for any cluster i, j.

- Increment the counter m = m+1. Join (a) et (b) and call it (m). Assign the level L(m) = d[(a),(b)].

- Update the proximity matrix D. Remove the lines/rows corresponding to (a) et (b) and create a line/row for (m). Compute the new values of D :

  d[(k), (m)] = min { d[(k),(a)], d[(k),(b)] } for any cluster (k) != (a) or (b)

WHILE L(m) < K

Hierarchical Categorization

Problems with the grouping categorization :

- Not efficient with a lot of data.
- Time complexity: $O(n^2)$
- Cannot reset the previous grouping

Applications

• Marketing : find similar consumers (same products bought, same behavior) in a supermarket database

• Biology : classification of animals, plants, …

• Library : find groups of books, reviews, CD/DVD, …

• Insurance, banking : find stocks with the same trends, identify similar appartments, markets, … Identify strange behavior (cheaters)

• Urban plannification : identify parcels given their values, their types, their geographical location

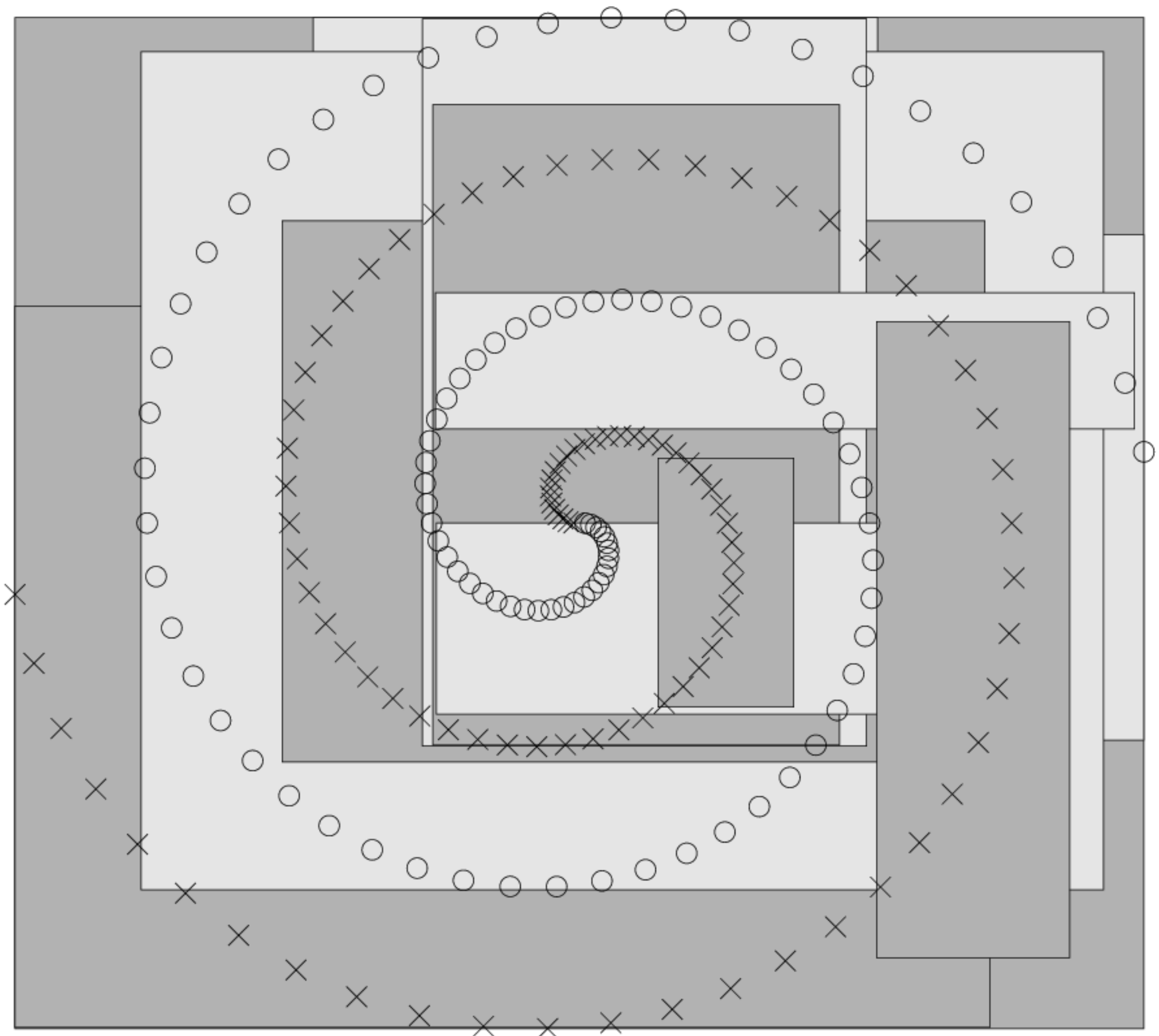
• Terrestrial studies : groups and identify dangerous areas (tidal wave, tsunami, earthquake, ...)

• Web : automatic classification of a document base, using keywords (texts, images, musics, videos, ...)

Examples

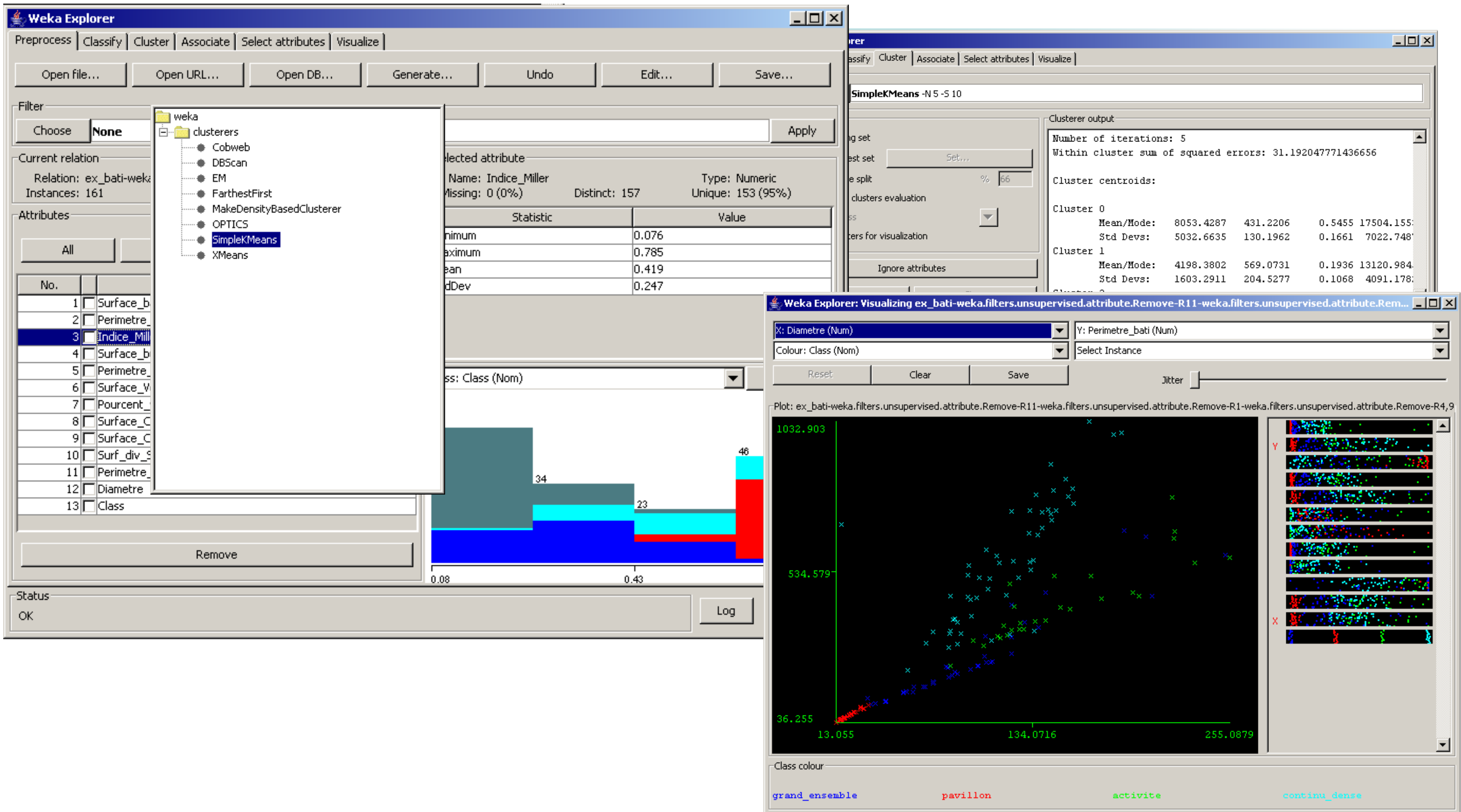
- Thesis of Jean-Pierre Novak (2000)

(hyper-rectangular neural nets)



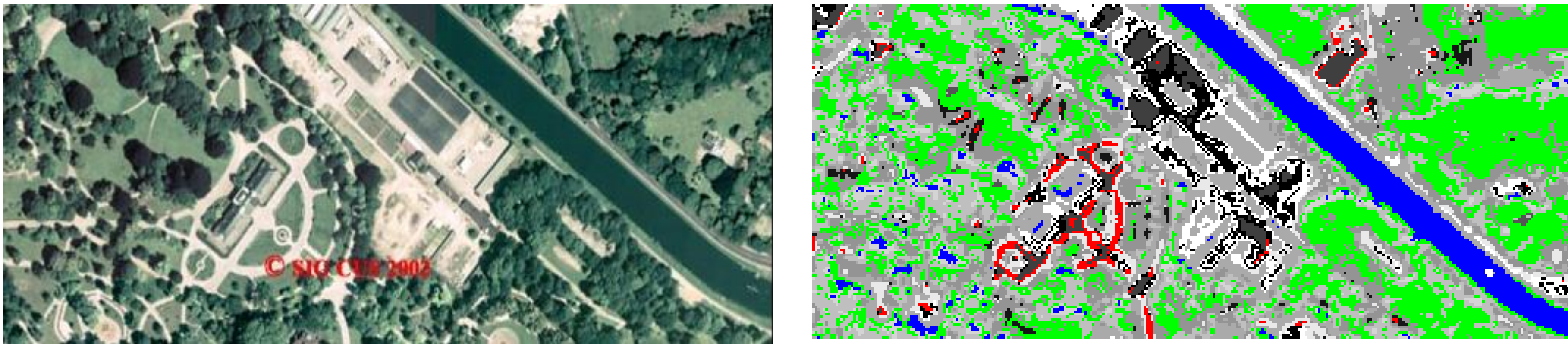
Examples

- Weka (free software for classification in Java)

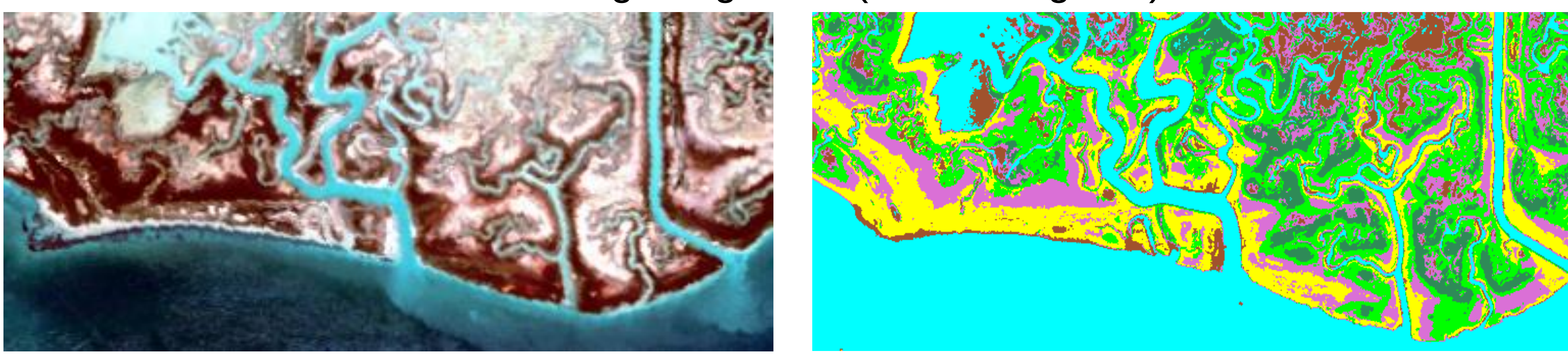


Examples

- K-Means on real pictures



The Orangerie garden (Strasbourg, FR)



The saltmarshes of San Felice (Venice, IT)