



Clustering : une vision unifiée pour une utilisation éclairée - Partie 1 : Méthodes exploratoires en clustering

Christophe Biernacki

► To cite this version:

Christophe Biernacki. Clustering : une vision unifiée pour une utilisation éclairée - Partie 1 : Méthodes exploratoires en clustering. Doctoral. Paris, France. 2023. hal-04370746

HAL Id: hal-04370746

<https://inria.hal.science/hal-04370746>

Submitted on 3 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Clustering : une vision unifiée pour une utilisation éclairée

—

Partie 1

Méthodes exploratoires en clustering

C. Biernacki

Ateliers statistiques de la SFdS, 15 et 16 juin 2023, IHP, Paris



Preamble

What is this course?

- Be able to perform practically some methods
- Use them with discernment

What is not this course?

- Not an exhaustive list of clustering methods (and related bibliography)
- Do not make specialists of clustering methods

This preamble is valid for all four lessons:

- 1 Méthodes exploratoires en clustering
- 2 Evaluation d'une méthode de clustering
- 3 Formalisation par modèles de mélange
- 4 Traitement de la grande dimension & co-clustering

Lectures

- **General overview of data mining** (contain some pretreatments before clustering):
Gérard Govaert et al. (2009). Data Analysis. Wiley-ISTE, ISBN: 978-1-848-21098-1.
<https://www.wiley.com/en-fr/Data+Analysis-p-9781848210981>
- **More advanced material on clustering:**
 - Christian Hennig, Marina Meila, Fionn Murtagh, Roberto Rocci (2015). Handbook of Cluster Analysis. Chapman and Hall/CRC, ISBN 9781466551886, Series: Chapman & Hall/CRC Handbooks of Modern Statistical Methods.
<https://www.crcpress.com/Handbook-of-Cluster-Analysis/Hennig-Meila-Murtagh-Rocci/p/book/9781466551886>
 - Christophe Biernacki. Mixture models. J-J. Droesbeke; G. Saporta; C. Thomas-Agnan. Choix de modèles et agrégation, Technip, 2017.
<https://hal.inria.fr/hal-01252671/document>
 - Christophe Biernacki, Cathy Maugis. High-dimensional clustering. Choix de modèles et agrégation, Sous la direction de J-J. DROESBEKE, G. SAPORTA, C. THOMAS-AGNAN Edition: Technip.
<https://hal.archives-ouvertes.fr/hal-01252673v2/document>
- **Advanced material on co-clustering:**
 - Gérard Govaert, Mohamed Nadif (2013). Co-Clustering: Models, Algorithms and Applications. Wiley-ISTE, ISBN-13: 978-1848214736.
<https://www.wiley.com/en-fr/Co+Clustering:+Models,+Algorithms+and+Applications-p-9781848214736>
 - C. Biernacki, J. Jacques, C. Keribin (2023). A Survey on Model-Based Co-Clustering: High Dimension and Estimation Challenges, *Journal of Classification*, in press
<https://hal.science/hal-03769727/document>
- **Basic to more advanced R book:** Pierre-Andre Cornillon, Arnaud Guyader, Francois Husson, Nicolas Jegou, Julie Josse, Maela Kloareg, Eric Matzner-Lober, Laurent Rouvière (2012). R for Statistics. Chapman and Hall/CRC, ISBN 9781439881453.
<https://www.crcpress.com/R-for-Statistics/Cornillon-Guyader-Husson-Jegou-Josse-Kloareg-Matzner-Lober-Rouviere/p/book/9781439881453>

Take home message

cluster
clustering

define both!

Outline

- 1 Data
- 2 Classifications(s)
- 3 Clustering: motivation
- 4 Empirical procedures
- 5 Automatic procedures
- 6 Experiments
- 7 To go further

Data
●○○○
○○○○○○○

Classifications(s)
○○○○○○○○○○○○○○○

Clustering: motivation
○○○○○○

Empirical procedures
○○○○○○○○○○○○○○○

Automatic procedures
○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○○○

Experiments
○○○○○○

To go further
○○○○○

Everything begins from data!

Today's data (1/2)

Today, it is easy to collect many features, so it favors

- data variety and/or mixed
- data missing
- data uncertainty (or interval data)

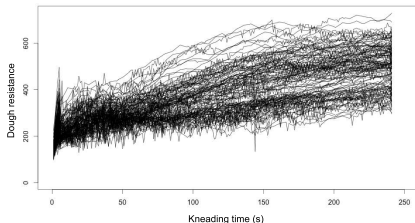
Mixed, missing, uncertain

Observed individuals x^O			
?	0.5	?	5
0.3	0.1	green	3
0.3	0.6	{red,green}	3
0.9	[0.25 0.45]	red	?
↓	↓	↓	↓
continuous	continuous	categorical	integer

Today's data (2/2)

And also

- Ranking data (like Olympics games)
- Directional data (like angle of wind direction)
- Ordinal data (a score like $A > B > C$)
- Functional data (like time series)¹
- Graphical data (like social networks, biological network)
- ...



¹See a specific lesson for clustering times series

Today's features: full mixed/missing



categorical

Marital status
married

integer
Chi

Children 3

missing

Size (m)
?

rank

Drink preference
beer > soda > water

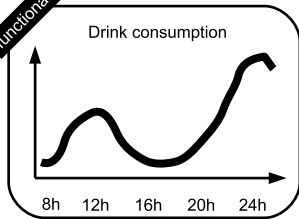
ordinal

Intelligence
low

continuous

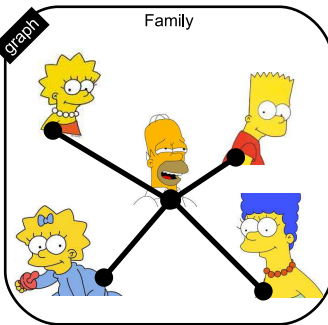
Weight (kg)
119.5

functional

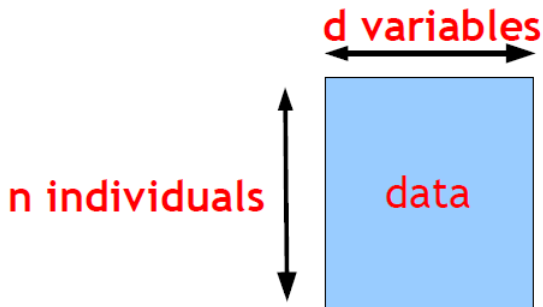


And so on...

graph



Data sets structure



Coding for data $\mathbf{x}$

- A **set** of n individuals

$$\mathbf{x} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$$

with $\mathbf{x}_i$ a set of (possibly non-scalar) d variables

$$\mathbf{x}_i = \{\mathbf{x}_{i1}, \dots, \mathbf{x}_{id}\}$$

where $\mathbf{x}_{ij} \in \mathcal{X}_j$

- A **n -uplet** of individuals

$$\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$$

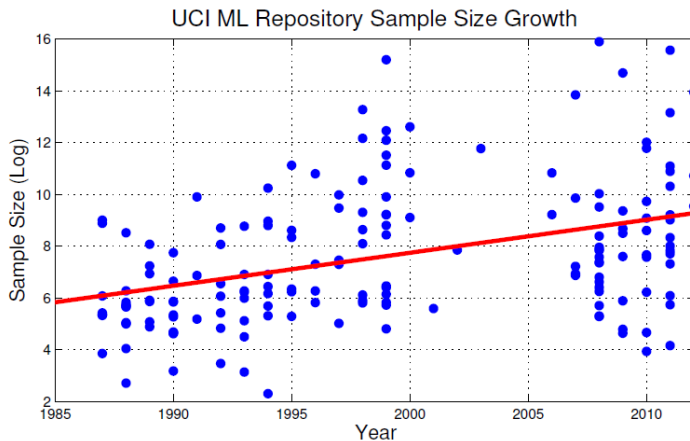
with $\mathbf{x}_i$ a d -uplet of (possibly non-scalar) variables

$$\mathbf{x}_i = (\mathbf{x}_{i1}, \dots, \mathbf{x}_{id}) \in \mathcal{X}$$

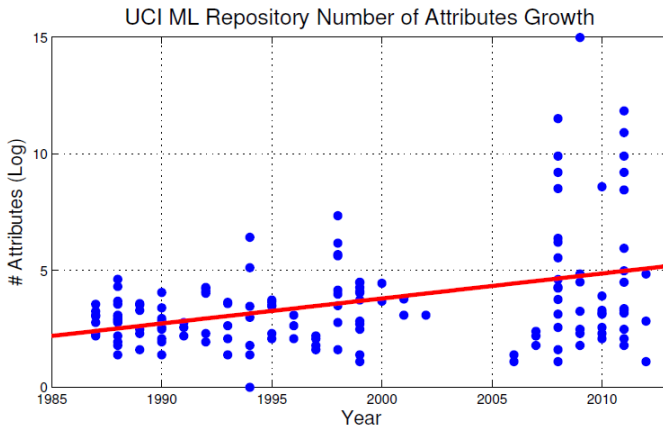
where $\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_d$

We will pass from a coding to another, depending on the practical utility (useful for some calculus to have matrices or vectors for instance)

Large data sets (n^2)



High-dimensional data (d)³



³S. Alelyani, J. Tang and H. Liu (2013). Feature Selection for Clustering: A Review. *Data Clustering: Algorithms and Applications*, 29

Genesis of “Big Data”

The Big Data phenomenon mainly originates in the increase of computer and digital resources at an ever lower cost

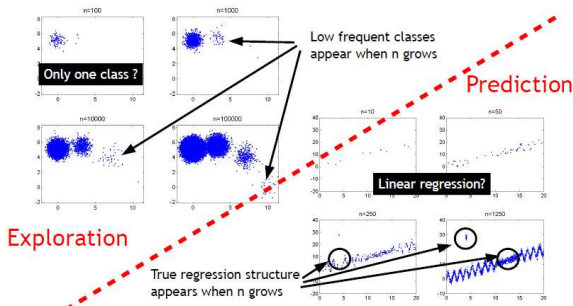
- **Storage cost per MB**: 700\$ in 1981, 1\$ in 1994, 0.01\$ in 2013
→ price divided by 70,000 in thirty years
- **Storage capacity of HDDs**: ≈ 1.02 Go in 1982, ≈ 8 To today
→ capacity multiplied by 8,000 over the same period
- **Computer processing speed**: 1 gigaFLOPS⁴ in 1985, 33 petaFLOPS in 2013
→ speed multiplied by 33 million

⁴FLOP = FLoating-point Operations Per Second

More data for what?

Opportunity to improve accuracy of traditional questionings

Synthetic examples



- Here is just illustrated the **effect of n**
- In further lessons will be illustrated the **effect of d** (be patient)

Supervised classification (1/3)

- **Data:** learning dataset $\mathcal{D} = \{\mathbf{x}^O, \mathbf{z}\}$
 - n individuals: $\mathbf{x} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\} = \{\mathbf{x}^O, \mathbf{x}^M\}$, each $\mathbf{x}_i \in \mathcal{X}$
 - Observed individuals $\mathbf{x}^O$
 - Missing individuals $\mathbf{x}^M$
 - Partition in K groups $G_1, \dots, G_K$: $\mathbf{z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)^6$, $\mathbf{z}_i = (z_{i1}, \dots, z_{iK})'$

$$\mathbf{x}_i \in G_k \Leftrightarrow z_{ih} = \mathbb{I}_{\{h=k\}}$$

- **Aim:** estimation of an allocation rule r from $\mathcal{D}$

$$\begin{aligned} r: \quad \mathcal{X} &\longrightarrow \{1, \dots, K\} \\ \mathbf{x}_{n+1}^O &\longmapsto r(\mathbf{x}_{n+1}^O). \end{aligned}$$

Other possible coding for $\mathbf{z}$

Sometimes it can be more convenient in formula to code $\mathbf{z} = (z_1, \dots, z_n)^a$ where

$$\mathbf{x}_i \in G_k \Leftrightarrow z_i = k$$

^aSometimes it would be more convenient to use a set also: $\mathbf{z} = \{\mathbf{z}_1, \dots, \mathbf{z}_n\}$

⁶Sometimes it would be more convenient to use a set also: $\mathbf{z} = \{\mathbf{z}_1, \dots, \mathbf{z}_n\}$

Supervised classification (2/3)

1st coding for z

Mixed, missing, uncertain

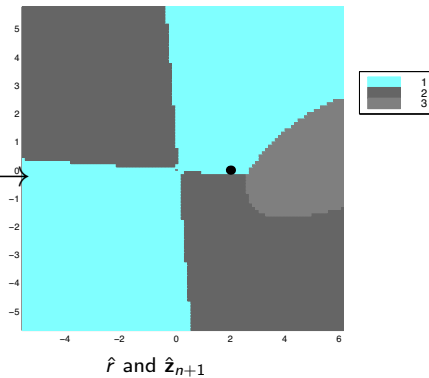
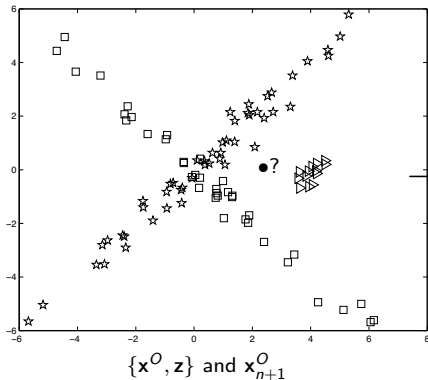
Individuals x^O				Partition z			$\Leftrightarrow$	Group
?	0.5	red	5	0	1	0	$\Leftrightarrow$	G_2
0.3	0.1	green	3	1	0	0	$\Leftrightarrow$	G_1
0.3	0.6	{red,green}	3	1	0	0	$\Leftrightarrow$	G_1
0.9	[0.25 0.45]	red	?	0	0	1	$\Leftrightarrow$	G_3
↓	↓	↓	↓					
continuous	continuous	categorical	integer					

2nd coding for z

Mixed, missing, uncertain

Individuals x^O				Partition z		$\Leftrightarrow$	Group
?	0.5	red	5	2		$\Leftrightarrow$	G_2
0.3	0.1	green	3	1		$\Leftrightarrow$	G_1
0.3	0.6	{red,green}	3	1		$\Leftrightarrow$	G_1
0.9	[0.25 0.45]	red	?	3		$\Leftrightarrow$	G_3
↓	↓	↓	↓				
continuous	continuous	categorical	integer				

Supervised classification (3/3)



Semi-supervised classification (1/3)

- **Data:** learning dataset $\mathcal{D} = \{\mathbf{x}^O, \mathbf{z}^O\}$
 - n individuals: $\mathbf{x} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\} = \{\mathbf{x}^O, \mathbf{x}^M\}$ belonging to a space $\mathcal{X}$
 - Observed individuals $\mathbf{x}^O$
 - Missing individuals $\mathbf{x}^M$
 - Partition: $\mathbf{z} = \{\mathbf{z}_1, \dots, \mathbf{z}_n\} = \{\mathbf{z}^O, \mathbf{z}^M\}$
 - Observed partition $\mathbf{z}^O$
 - Missing partition $\mathbf{z}^M$
- **Aim:** estimation of an allocation rule r from $\mathcal{D}$

$$\begin{array}{ccc} r: & \mathcal{X} & \longrightarrow \{1, \dots, K\} \\ & \mathbf{x}_{n+1}^O & \longmapsto r(\mathbf{x}_{n+1}^O). \end{array}$$

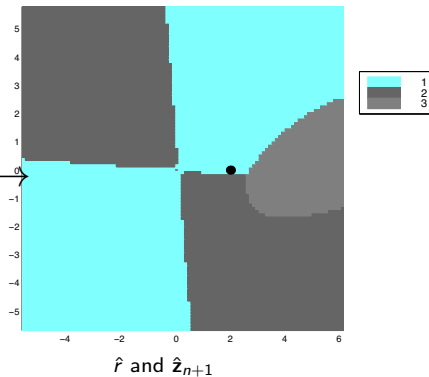
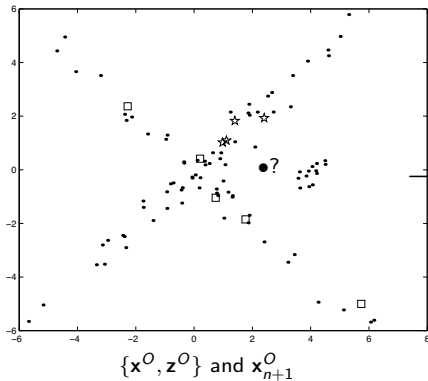
- **Idea:** $\mathbf{x}$ is cheaper than $\mathbf{z}$ so $\#\mathbf{z}^M \gg \#\mathbf{z}^O$

Semi-supervised classification (2/3)

Mixed, missing, uncertain

Individuals x^O				Partition z^O			$\Leftrightarrow$	Group
?	0.5	red	5	0	?	?	$\Leftrightarrow$	G_2 or G_3
0.3	0.1	green	3	1	0	0	$\Leftrightarrow$	G_1
0.3	0.6	{red, green}	3	?	?	?	$\Leftrightarrow$	???
0.9	[0.25 0.45]	red	?	0	0	1	$\Leftrightarrow$	G_3
↓	↓	↓	↓					
continuous	continuous	categorical	integer					

Semi-supervised classification (3/3)



Unsupervised classification (1/3)

- **Data:** learning dataset $\mathcal{D} = \mathbf{x}^O$, so $\mathbf{z}^O = \emptyset$
- **Aim:** estimation of the partition $\mathbf{z}^7$ and the number of groups K
- **Also known as:** clustering

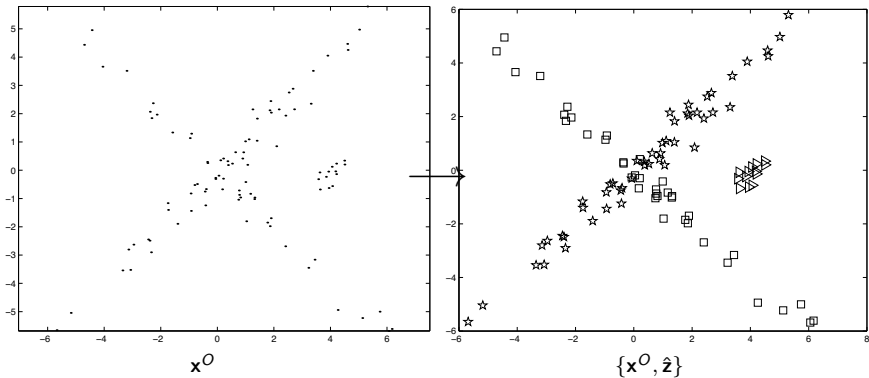
⁷We will see other clustering structures further, here to fix ideas with the main one.

Unsupervised classification (2/3)

Mixed, missing, uncertain

Individuals x^O				Partition z^O			$\Leftrightarrow$	Group
?	0.5	red	5	?	?	?	$\Leftrightarrow$	???
0.3	0.1	green	3	?	?	?	$\Leftrightarrow$	???
0.3	0.6	{red,green}	3	?	?	?	$\Leftrightarrow$	???
0.9	[0.25 0.45]	red	?	?	?	?	$\Leftrightarrow$	???
↓	↓	↓	↓					
continuous	continuous	categorical	integer					

Unsupervised classification (3/3)



Traditional solutions (1/3)

Two main model-based frameworks⁸

- **Generative models**

- Model $p(\mathbf{x}, \mathbf{z})$
- Thus direct model for $p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{z})$
- Easy to take into account some missing $\mathbf{z}$ and $\mathbf{x}$

- **Predictive models**

- Model $p(\mathbf{z}|\mathbf{x})$ or sometimes $\mathbf{1}_{\{p(\mathbf{z}|\mathbf{x}) > 1/2\}}$ or also ranking on $p(\mathbf{z}|\mathbf{x})$
- Avoid assumptions on $p(\mathbf{x})$, thus avoids associated error model
- difficult to take into account some missing $\mathbf{z}$ and $\mathbf{x}$

⁸Sometimes presented as distance-based instead of model-based: see some links in my 3th lesson

Traditional solutions (2/3)

No mixed, missing or uncertain data:

- **Supervised classification**⁹
 - **Generative models:** linear/quadratic discriminant analysis
 - **Predictive models:** logistic regression, support vector machines (SVM), k nearest neighbourhood, classification trees...
- **Semi-supervised classification**¹⁰
 - **Generative models:** mixture models
 - **Predictive models:** low density separation (transductive SVM), graph-based methods...
- **Unsupervised classification**¹¹
 - **Generative models:** k -means like criteria, hierarchical clustering, mixture models
 - **Predictive models:** -

⁹Govaert *et al.*, Data Analysis, Chap.6, 2009

¹⁰Chapelle *et al.*, Semi-supervised learning, 2006

¹¹Govaert *et al.*, Data Analysis, Chap.7-9, 2009

Data
○○○○○
○○○○○○○○

Classifications(s)
○○○○○○○○○○○○○○

Clustering: motivation
●○○○○○

Empirical procedures
○○○○○○○○○○○○○○

Automatic procedures
○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○○○

Experiments
○○○○○○

To go further
○○○○○

Outline

1 Data

2 Classifications(s)

3 Clustering: motivation

4 Empirical procedures

5 Automatic procedures

6 Experiments

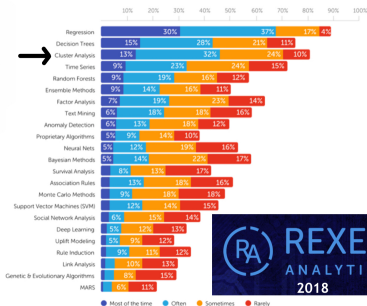
7 To go further

Clustering everywhere¹²

Regression, Cluster Analysis, & Decision Trees: The Core Algorithm Triad



What algorithms/analytic methods do you TYPICALLY use?



© 2018 Rexer Analytics

● Most of the time ● Often ● Sometimes ● Rarely

Rexer Analytics' surveys since 2007 have consistently shown that data scientists primarily work with the algorithm triad of regression, decision trees, and cluster analysis. In every survey since 2007, over half of respondents reported using each of these methods in the prior year. Among these three, regression is clearly dominant, with more than two thirds (67%) of respondents indicating that they use regression "often" or "most of the time".

On average 2017 respondents use 11 different algorithms in the course of their work (slightly down from 12 in 2015). Despite extensive media hype about AI, Cognitive Computing, Deep Learning, and the rise of machine learning and its related algorithms, no algorithms showed substantial increased usage since the 2015 survey.

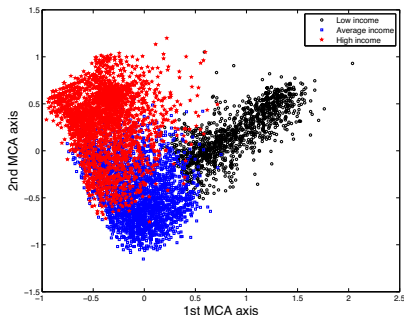
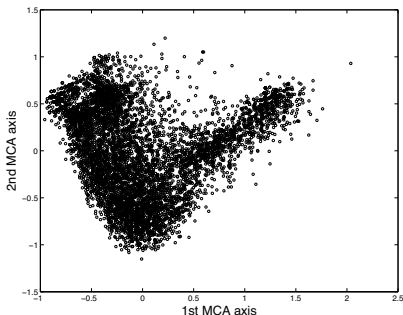


Why such a success despite its complexity relatively to other classification purposes?

¹²Rexer Analytics's Annual Data Miner Survey is the largest survey of data mining, data science, and analytics professionals in the industry (survey of 2018)

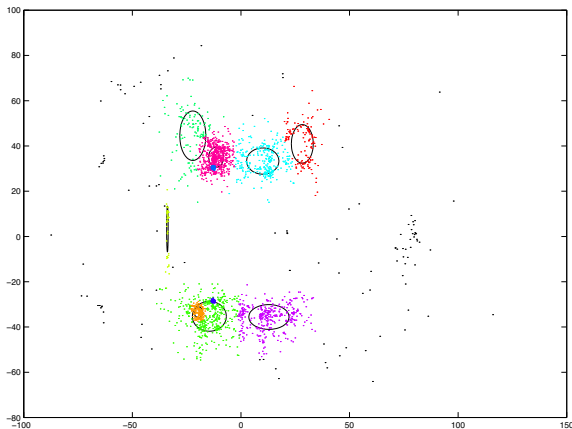
A 1st aim: explanatory task

- A clustering for a **marketing study**
- **Data**: $d = 13$ demographic attributes (nominal and ordinal variables) of $n = 6\,876$ shopping mall customers in the San Francisco Bay (SEX (1. Male, 2. Female), MARITAL STATUS (1. Married, 2. Living together, not married, 3. Divorced or separated, 4. Widowed, 5. Single, never married), AGE (1. 14 thru 17, 2. 18 thru 24, 3. 25 thru 34, 4. 35 thru 44, 5. 45 thru 54, 6. 55 thru 64, 7. 65 and Over), etc.)
- **Partition**: retrieve less than 19 999\$ (group of “low income”), between 20 000\$ and 39 999\$ group of “average income”), more than 40 000\$ (group of “high income”)



Another explanatory example: acoustic emission control

- **Data:** $n = 2\,061$ event locations in a rectangle of $\mathbb{R}^2$ representing the vessel
- **Groups:** sound locations = vessel defects



A 2nd aim: preprocessing step (1/2)

A synthetic example in [supervised classification](#): predict two groups (blue and black)¹³

- Logit model:

- Not very flexible since linear borderline
- Unbiased ML estimate but asymptotic variance $\sim (\mathbf{x}'\mathbf{w}\mathbf{x})^{-1}$ is influenced by correlations

$$\text{logit}(p(\text{blue}|\mathbf{x}_{n+1})) = \beta' \mathbf{x}_{n+1}$$

- A clustering may [improve logistic regression prediction](#)

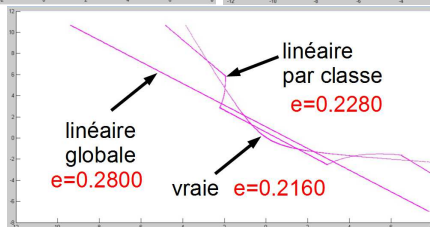
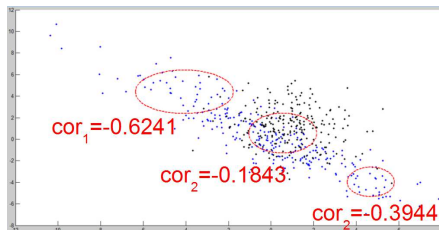
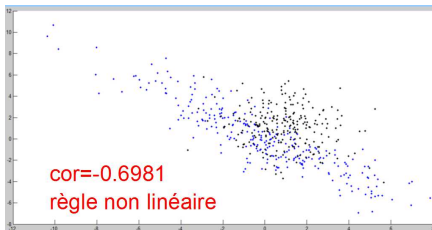
- [More flexible borderline](#): piecewise linear
- Decrease correlation so [decrease variance](#)

$$\text{logit}(p(\text{blue}|\mathbf{x}_{n+1})) = \beta'_{z_{n+1}} \mathbf{x}_{n+1}$$

[Do not confound](#) groups blue and black and clusters z_{n+1} !!

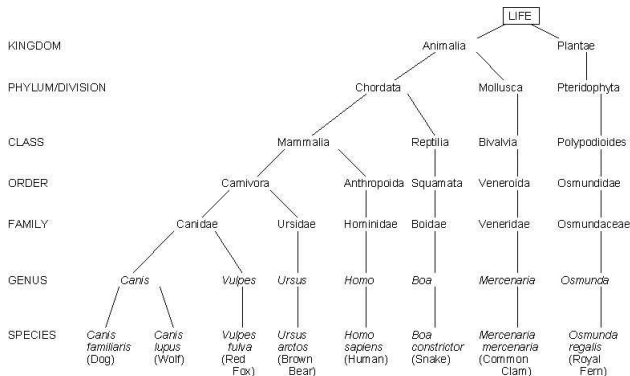
¹³Another lesson will retain a similar idea through a mixture of regressions

A 2nd aim: preprocessing step (2/2)



A first systematic attempt

- Carl von Linné (1707–1778), Swedish botanist, physician, and zoologist
- Father of modern taxonomy based on the most **visible similarities** between species
- *Linnaeus's Systema Naturae* (1st ed. in 1735) lists about 10,000 species of organisms (6,000 plants, 4,236 animals)

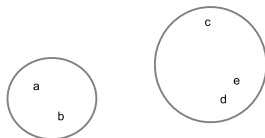


Four main clustering structures: partition

Each object belongs to exactly one cluster

- **Definition:** a set of K non-empty parts of $\mathbf{x}$: $P = (G_1, \dots, G_K)$:
 - For all $k \neq k'$, $G_k \cap G_{k'} = \emptyset$
 - $G_1 \cup \dots \cup G_K = \mathbf{x}$
- **Example:** $\mathbf{x} = \{a, b, c, d, e\}$

$$P = \underbrace{\{a, b\}}_{G_1}, \underbrace{\{c, d, e\}}_{G_2}$$



- **Notation:** $\mathbf{z} = (z_1, \dots, z_n)$, with $z_i \in \{1, \dots, K\}$

Four main clustering structures: partition with outliers

Each object belongs to one cluster or less

■ **Definition:** a set of K non-empty parts of x : $P = (G_1, \dots, G_K)$:

■ For all $k \neq k'$, $G_k \cap G_{k'} = \emptyset$

■ $G_1 \cup \dots \cup G_K \subset x$

■ **Example:** $x = \{a, b, c, d, e\}$

$$P = \underbrace{\{\{a, b\}\}}_{G_1}, \underbrace{\{\{d, e\}\}}_{G_2}, \quad c \text{ excluded}$$



■ See more in Lesson 3

Four main clustering structures: multiclass partition

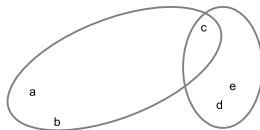
Each object belongs to one cluster or more

■ **Definition:** a set of K non-empty parts of x : $P = (G_1, \dots, G_K)$:

■ $G_1 \cup \dots \cup G_K = x$

■ **Example:** $x = \{a, b, c, d, e\}$

$$P = \{\underbrace{\{a, b, c\}}_{G_1}, \underbrace{\{c, d, e\}}_{G_2}\}$$



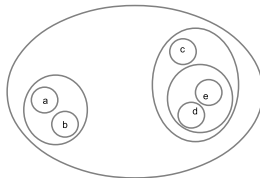
■ Not considered in lessons

Four main clustering structures: hierarchy

Each object belongs to nested partitions

- **Definition:** H is a hierarchy of x if
 - $x \in H$
 - For any $x_i \in x$, $\{x_i\} \in H$
 - For any $G_1, G_2 \in H$, $G_1 \cap G_2 \in \{G_1, G_2, \emptyset\}$
- **Example:** $x = \{a, b, c, d, e\}$

$$H = \{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{d, e\}, \{a, b\}, \{c, d, e\}, \{a, b, c, d, e\}\}$$



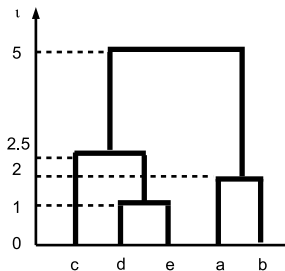
- Case of the species taxonomy

Four main clustering structures: indexed hierarchy

Each element of a hierarchy is associated to an index value

- **Definition:** $\iota : H \rightarrow \mathbb{R}^+$ is an **index** of a hierarchy H if
 - ι is a non-decreasing mapping: $(G_1, G_2) \in H^2$, $G_1 \subset G_2$, $G_1 \neq G_2 \Rightarrow \iota(G_1) < \iota(G_2)$
 - For all $x_i \in \mathbf{x}$, $\iota(\{x_i\}) = 0$
- The couple (H, ι) is an **indexed hierarchy** and can be visualized by a **dendrogram**
- **Example** (continued):

$$H = \underbrace{\{a\}}_0, \underbrace{\{b\}}_0, \underbrace{\{c\}}_0, \underbrace{\{d\}}_0, \underbrace{\{e\}}_0, \underbrace{\{d, e\}}_1, \underbrace{\{a, b\}}_2, \underbrace{\{c, d, e\}}_{2.5}, \underbrace{\{a, b, c, d, e\}}_5$$



Limit of visualization (1/3)

Prostate cancer data¹⁴

- **Individuals:** 506 patients with prostatic cancer grouped on clinical criteria into two Stages 3 and 4 of the disease
- **Variables:** $d = 12$ pre-trial variates were measured on each patient, composed by **eight continuous** variables (age, weight, systolic blood pressure, diastolic blood pressure, serum haemoglobin, size of primary tumour, index of tumour stage and histologic grade, serum prostatic acid phosphatase) and **four categorical** variables with various numbers of levels (performance rating, cardiovascular disease history, electrocardiogram code, bone metastases)
- Some **missing data:** 62 missing values ($\approx 1\%$)

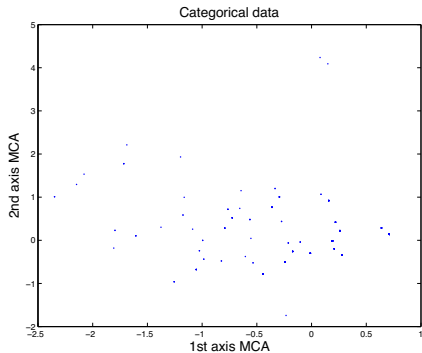
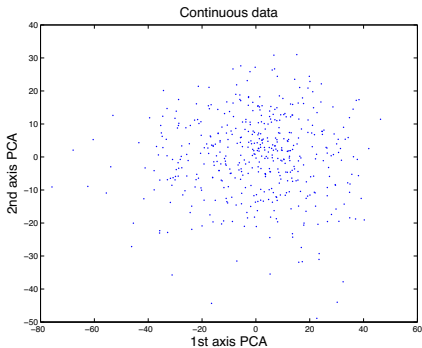
Questions

- Doctors say two stages of the disease are **well separated** with the variables
- We forget the classes (Stages of the disease) to **retrieve clustering by visualizing**

¹⁴Byar DP, Green SB (1980): Bulletin Cancer, Paris 67:477-488

Limit of visualization (2/3)

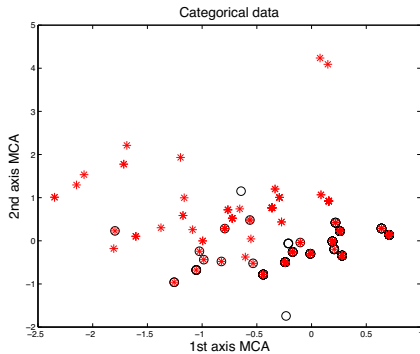
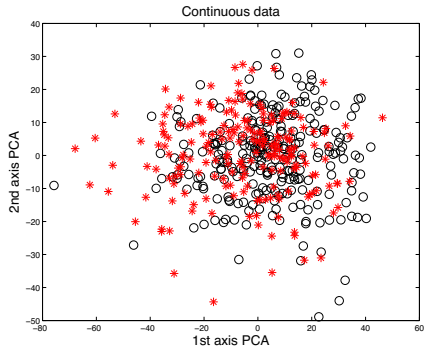
- Perform PCA for continuous variables, MCA for categorical ones
- Discard missing values since not convenient for traditional PCA and MCA



We do not “see” anything...

Limit of visualization (2/3)

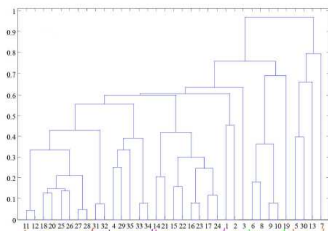
If we have a look at the true partition, indeed visualizing was not enough...



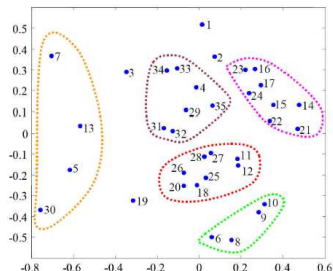
Need to define more “automatic” methods acting **directly in the space $\mathcal{X}$**

Clustering of clustering algorithms¹⁵

- Jain *et al.* (2004) hierarchical clustered 35 different clustering algorithms into 5 groups based on their partitions on 12 different datasets.
- It is not surprising to see that the related algorithms are clustered together.
- For a visualization of the similarity between the algorithms, the 35 algorithms are also embedded in a two-dimensional space obtained from the 35x35 similarity matrix.



(a)



(b)

Figure 10 Clustering of clustering algorithms. (a) Hierarchical clustering of 35 different algorithms; (b) Sammon's mapping of the 35 algorithms into a two-dimensional space, with the clusters highlighted for visualization. The algorithms in the group (4, 29, 31-35) correspond to K-means, spectral clustering, Gaussian mixture models, and Ward's linkage. The algorithms in group (6, 8-10) correspond to CHAMELEON algorithm with different objective functions.

¹⁵A.K. Jain (2008). Data Clustering: 50 Years Beyond K-Means.

Dissimilarities, distances, similarities: definitions

General idea

- Put in the **same cluster** individuals x_1 and x_2 which are **similar**
- Put in **different clusters** individuals x_1 and x_2 which are **dissimilar**

- Definition of a **dissimilarity** $\delta : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$
 - Separation: for any $(x_1, x_2) \in \mathcal{X}^2$, $\delta(x_1, x_2) = 0 \Leftrightarrow x_1 = x_2$
 - Symmetry: for any $(x_1, x_2) \in \mathcal{X}^2$, $\delta(x_1, x_2) = \delta(x_2, x_1)$
- Definition of a **distance** $\delta : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$: it is a dissimilarity with a new property
 - Triangular inequality: for any $(x_1, x_2, x_3) \in \mathcal{X}^3$, $\delta(x_1, x_3) \leq \delta(x_1, x_2) + \delta(x_2, x_3)$
- Definition of a **similarity** (or affinity) $s : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$
 - For any $x_1 \in \mathcal{X}$, $\delta(x_1, x_1) = s_{\max}$ with $s_{\max} \geq s(x_1, x_2)$ for any $(x_1, x_2) \in \mathcal{X}^2$, $x_1 \neq x_2$
 - Symmetry: for any $(x_1, x_2) \in \mathcal{X}^2$, $s(x_1, x_2) = s(x_2, x_1)$

Often dissimilarities (or similarities) are enough for clustering

Similarities, dissimilarities, distances: examples for quantitative data

$$\mathcal{X} = \mathbb{R}^d$$

- Euclidean distance (or L_1 norm) with metric $\mathbf{M}$:

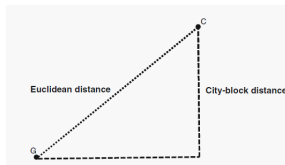
$$\delta_{\mathbf{M}}(\mathbf{x}_1, \mathbf{x}_2) = \|\mathbf{x}_1 - \mathbf{x}_2\|_{\mathbf{M}} = \sqrt{(\mathbf{x}_1 - \mathbf{x}_2)' \mathbf{M} (\mathbf{x}_1 - \mathbf{x}_2)}$$

with a metric $\mathbf{M}$: symmetric and positive definite $d \times d$ matrix

- Manhattan distance or city-block distance or L_1 norm:

$$\delta(\mathbf{x}_1, \mathbf{x}_2) = \sum_{j=1}^d \|\mathbf{x}_{1j} - \mathbf{x}_{2j}\|$$

- ...



Similarities, dissimilarities, distances: examples for binary data

$$\mathcal{X} = \{0, 1\}^d \text{ thus } \mathbf{x}_{ij} \in \{0, 1\}$$

- The Hamming distance counts the dimensions on which both vectors are different:

$$\delta(\mathbf{x}_1, \mathbf{x}_2) = \sum_{j=1}^d \mathbb{I}_{\{\mathbf{x}_{1j} \neq \mathbf{x}_{2j}\}}$$

- The matching coefficient counts dimensions on which both vectors are non-zero:

$$\delta(\mathbf{x}_1, \mathbf{x}_2) = \sum_{j=1}^d \mathbb{I}_{\{\mathbf{x}_{1j} = \mathbf{x}_{2j} = 1\}}$$

- ...

Similarities, dissimilarities, distances: similarities proposals

- It is easy to construct similarities from dissimilarities or distances
- A typical example:

$$s(\mathbf{x}_1, \mathbf{x}_2) = \exp(-\delta(\mathbf{x}_1, \mathbf{x}_2))$$

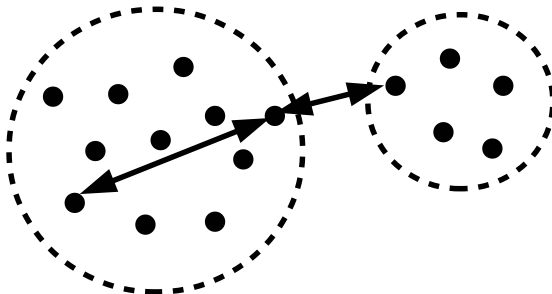
Thus $s_{\max} = 1$

- The **Gaussian similarity** is an important case ($\sigma > 0$):

$$s(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2\sigma^2}\right)$$

Weaken a two restrictive requirement

The property that two individuals in the same cluster are closer than any other individual in another cluster is too strong



Relax this **local** criterion by a **global** criterion

Within-cluster inertia criterion

Select the partition $\mathbf{z}$ minimizing the criterion

$$W_{\mathbf{M}}(\mathbf{z}) = \sum_{i=1}^n \sum_{k=1}^K z_{ik} \|\mathbf{x}_i - \bar{\mathbf{x}}_k\|_{\mathbf{M}}^2$$

- Look for compact clusters (individuals of the same cluster are close from each other)
- $\|\cdot\|_{\mathbf{M}}$ is the Euclidian distance with **metric** $\mathbf{M}$ in $\mathbb{R}^d$
- $\bar{\mathbf{x}}_k$ is the **mean** (or center) of the k th cluster

$$\bar{\mathbf{x}}_k = \frac{1}{n_k} \sum_{i=1}^n z_{ik} \mathbf{x}_i$$

and $n_k = \sum_{i=1}^n z_{ik}$ indicates the **number of individuals** in cluster k

Between-cluster inertia criterion

Select the partition $\mathbf{z}$ maximizing the criterion

$$B_{\mathbf{M}}(\mathbf{z}) = \sum_{k=1}^K n_k \|\bar{\mathbf{x}} - \bar{\mathbf{x}}_k\|_{\mathbf{M}}^2$$

- Look for clusters far from each other
- $\bar{\mathbf{x}}$ the mean (or center) of the whole data set $\mathbf{x}$:

$$\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$$

- It is in fact equivalent to minimize $W_{\mathbf{M}}(\mathbf{z})$ since

$$B_{\mathbf{M}}(\mathbf{z}) + W_{\mathbf{M}}(\mathbf{z}) = \text{cste}$$

A combinatorial problem

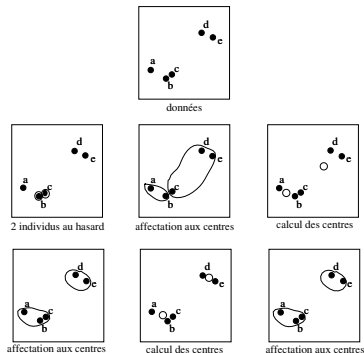
- Number of possible partitions $\mathbf{z}$: $\#\{\mathbf{z}\} \approx \frac{K^n}{K!}$
- For instance : $n = 40$, $K = 3$, $\#\{\mathbf{z}\} \approx 2.10^{18}$ (two billion of billion), thus about 64000 years for a computer calculating one million of partitions per second
- Thus impossible to minimize directly $W_{\mathbf{M}}(\mathbf{z})$
- But equivalent to minimize on $\mathbf{z}$ and $\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K\}$ with $\boldsymbol{\mu}_k \in \mathbb{R}^d$, the criterion

$$\tilde{W}_{\mathbf{M}}(\mathbf{z}, \boldsymbol{\mu}) = \sum_{i=1}^n \sum_{k=1}^K z_{ik} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|_{\mathbf{M}}^2$$

It leads to the famous K -means algorithm

K-means algorithm

Alternating optimization between the partition z and the center μ of clusters



Algorithme des centres mobiles

- Start from a random μ among x (avoid to start from a random z)
- Each iteration decreases the criterion $\tilde{W}_M(z, \mu)$ (so also $W_M(z)$)
- Converge into a finite (and often low) number of iterations
- Memory complexity $O((n + K)d)$, time complexity $O(\text{nb.iter.}Kdn)$

Fuzzy within-cluster inertia criterion

Select the fuzzy partition $\mathbf{c}$ minimizing the criterion

$$W_{\mathbf{M},m}^{\text{fuzzy}}(\mathbf{c}) = \sum_{i=1}^n \sum_{k=1}^K c_{ik}^m \|\mathbf{x}_i - \bar{\mathbf{x}}_k\|_{\mathbf{M}}^2$$

- Fuzzy partition $\mathbf{c} = \{\mathbf{c}_1, \dots, \mathbf{c}_K\}$, $\mathbf{c}_k = (c_{1k}, \dots, c_{nk})$, $c_{ik} \in [0, 1]$ defined by

$$c_{ik} = \frac{1}{\sum_{k'=1}^K \left(\frac{\|\mathbf{x}_i - \bar{\mathbf{x}}_k\|_{\mathbf{M}}}{\|\mathbf{x}_i - \bar{\mathbf{x}}_{k'}\|_{\mathbf{M}}} \right)^{\frac{2}{m-1}}}$$

- $m \geq 1$ determines the level of cluster fuzziness: m large produces fuzzier clusters; $m = 1$ (in the limit) produces hard clustering (m converges to 0 or 1)
- $\bar{\mathbf{x}}_k$ is the **fuzzy mean** (or center) of the k th cluster

$$\bar{\mathbf{x}}_k = \frac{1}{n_k} \frac{\sum_{i=1}^n c_{ik}^m \mathbf{x}_i}{\sum_{i=1}^n c_{ik}^m}$$

and $n_k = \sum_{i=1}^n c_{ik}$ indicates the **fuzzy number of individuals** in cluster k

- Optimize $W_{\mathbf{M},m}^{\text{fuzzy}}(\mathbf{c})$ (through $\tilde{W}_{\mathbf{M},m}^{\text{fuzzy}}(\mathbf{c}, \mu)$) by the **fuzzy K-means** algorithm

Kernel clustering: the idea

- Original (input) space: **non-linearly separable** clusters
- Feature space: a **non-linear mapping ϕ** to a higher-dimensional space

$$\begin{aligned} \phi: \mathcal{X} &\mapsto \mathcal{Y} \\ \mathbf{x}_i &\rightarrow \phi(\mathbf{x}_i) \end{aligned}$$

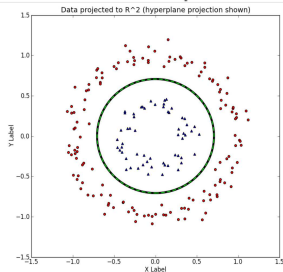
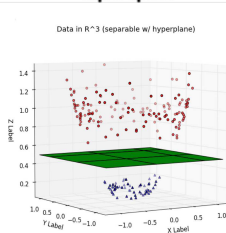
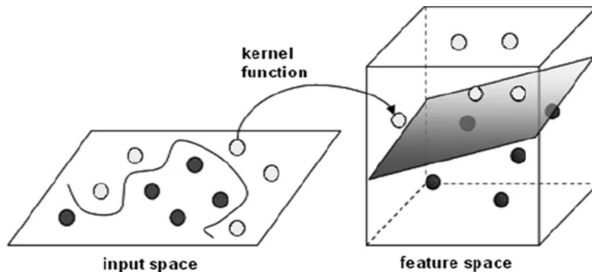
- Expected property: **linearly separable** clusters in the feature space
- **Difficulty**: define ϕ
- The “kernel trick”: not necessary to define ϕ , **define just an inner product**

$$\kappa(\mathbf{x}_i, \mathbf{x}_{i'}) = \phi(\mathbf{x}_i)' \phi(\mathbf{x}_{i'})$$

κ is so-called a **kernel**

- Lesson 4: more arguments on the discriminant interest of higher dimension

Kernel clustering: illustration



Kernel clustering: fuzzy within-cluster rewriting

- W^{fuzzy} with the data set $\phi(\mathbf{x}) = \{\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_n)\}$ (we drop $\mathbf{M}$ and m)

$$W^{\text{fuzzy}}(\mathbf{c}) = \sum_{i=1}^n \sum_{k=1}^K c_{ik}^m \|\phi(\mathbf{x}_i) - \bar{\phi}_k(\mathbf{x})\|^2$$

with

$$\bar{\phi}_k(\mathbf{x}) = \frac{1}{n_k} \frac{\sum_{i=1}^n c_{ik}^m \phi(\mathbf{x}_i)}{\sum_{i=1}^n c_{ik}^m} \text{ and } c_{ik} = \frac{1}{\sum_{k'=1}^K \left(\frac{\|\phi(\mathbf{x}_i) - \bar{\phi}_{k'}(\mathbf{x})\|}{\|\phi(\mathbf{x}_i) - \bar{\phi}_k(\mathbf{x})\|} \right)^{\frac{2}{m-1}}}$$

- After some algebra, only appears inside an inner product since:

$$\begin{aligned} \|\phi(\mathbf{x}_i) - \bar{\phi}_k(\mathbf{x})\|^2 &= \phi(\mathbf{x}_i)' \phi(\mathbf{x}_i) - 2 \frac{\sum_{i'=1}^n c_{ik}^m \phi(\mathbf{x}_i)' \phi(\mathbf{x}_{i'})}{\sum_{i'=1}^n c_{i'k}^m} + \frac{\sum_{i''=1}^n \sum_{i'=1}^n c_{i'k}^m c_{i''k}^m \phi(\mathbf{x}_{i'})' \phi(\mathbf{x}_{i'')}}{(\sum_{i'=1}^n c_{i'k}^m)^2} \\ &= \kappa(\mathbf{x}_i, \mathbf{x}_i) - 2 \frac{\sum_{i'=1}^n c_{ik}^m \kappa(\mathbf{x}_i, \mathbf{x}_{i'})}{\sum_{i'=1}^n c_{i'k}^m} + \frac{\sum_{i''=1}^n \sum_{i'=1}^n c_{i'k}^m c_{i''k}^m \kappa(\mathbf{x}_{i'}, \mathbf{x}_{i'')}}{(\sum_{i'=1}^n c_{i'k}^m)^2} \end{aligned}$$

Consequently, **need only to define κ** for performing the (fuzzy) K -means...

Kernel clustering: kernel examples

- Polynomial

$$\kappa(\mathbf{x}_i, \mathbf{x}_{i'}) = (\mathbf{x}_i' \mathbf{x}_{i'} + a)^b$$

- Gaussian

$$\kappa(\mathbf{x}_i, \mathbf{x}_{i'}) = \exp(-\|\mathbf{x}_i - \mathbf{x}_{i'}\|/(2\sigma^2))$$

- Sigmoid

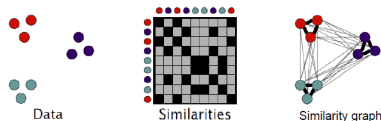
$$\kappa(\mathbf{x}_i, \mathbf{x}_{i'}) = \tanh(a\mathbf{x}_i' \mathbf{x}_{i'} + b)$$

Spectral clustering: the idea¹⁷

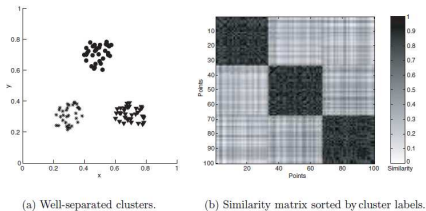
- Build a **similarity matrix S** $n \times n$ from $\mathbf{x}$:

$$S_{ii'} = s_{ii'} \text{ if } i \neq i', \quad S_{ii} = 0$$

- Similarities can be viewed as proximities between individuals (nodes) in a **graph**



- If some disconnected sub-graphs, **S** can be **reordered as block diagonal**



- We “can see” the clusters thus **clustering is expected to be easy**

¹⁷Figures from [A. Jain *et al.*, Data Clustering: A Review.]

Spectral clustering: Laplacian matrix

- For technical reasons, we study (unnormalized graph) **Laplacian $\mathbf{L}$** instead of **$\mathbf{S}$**

$$\mathbf{L} = \mathbf{D} - \mathbf{S}$$

where $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ is the diagonal **degree matrix** with $d_i = \sum_{i'=1}^n \mathbf{S}_{ii'}$

- Have in mind **$\mathbf{S}$** if easier ; the structure of **$\mathbf{L}$** is similar

Properties of $\mathbf{L}$

- 1 For every vector $\mathbf{u} = (u_1, \dots, u_n)' \in \mathbb{R}^n$, we have

$$\mathbf{u}'\mathbf{L}\mathbf{u} = \frac{1}{2} \sum_{i=1}^n \sum_{i'=1}^n s_{ii'} (u_i - u_{i'})^2$$

- 2 $\mathbf{L}$ is symmetric and positive semi-definite
- 3 The smallest eigenvalue of $\mathbf{L}$ is 0, the corresponding eigenvector is the constant one vector $\mathbf{1} = (1, \dots, 1)' \in \mathbb{R}^n$
- 4 $\mathbf{L}$ has n non-negative, real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$

Spectral clustering: specific spectral structure of $\mathbf{L}$

Linking clusters and $\mathbf{L}$

- 1 The multiplicity K of the eigenvalue 0 of $\mathbf{L}$ equals the number of clusters $G_1, \dots, G_K$ (or the connected components in the graph)
- 2 The eigenspace of eigenvalue 0 is spanned by the indicator vectors $\mathbf{1}_{G_1}, \dots, \mathbf{1}_{G_K}$ of those components, where $\mathbf{1}_{G_k}(i) = 1$ if $\mathbf{x}_i \in G_k$, zero otherwise

[A. Singh (2010). Spectral Clustering. Machine Learning.]

- If graph is connected, first Laplacian vec is constant (all 1s)
- If graph is disconnected (k connected components), Laplacian is block diagonal and first k Laplacian vecs are:



OR



$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_1 & & 0 \\ & \mathbf{L}_2 & \\ 0 & & \mathbf{L}_3 \end{bmatrix}$$

$$\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

First three eigenvectors

Spectral clustering: eigenvectors as new data to cluster

- Define a new data set $\mathbf{y} = \{\mathbf{y}_1, \dots, \mathbf{y}_n\} \in \mathbb{R}^K$ where

$$\mathbf{y}_i = (\mathbf{1}_{G_1}(i), \dots, \mathbf{1}_{G_K}(i))$$

- From the previous property, $\mathbf{y}_i$ s in the same clusters are very similar. . .
- . . . whereas $\mathbf{y}_i$ s in different clusters are very different
- Thus just run a K -means (for instance) to finish the clustering job!

[A.K. Jain (2008). Data Clustering: 50 Years Beyond K-Means]

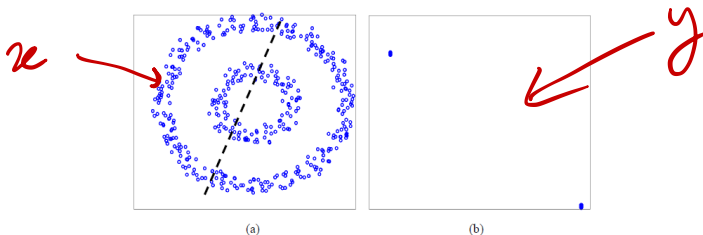
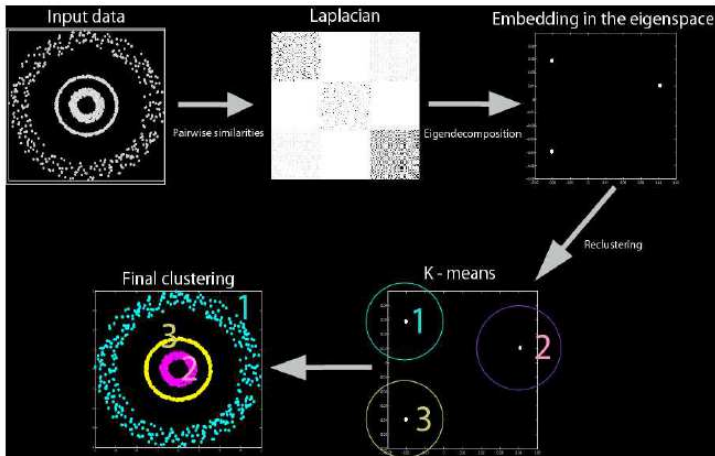


Figure 5 Importance of a good representation. (a) “Two rings” dataset where K-means fails to find the two “natural” clusters; the dashed line shows the linear cluster separation boundary obtained by running K-means with $K = 2$. (b) a new representation of the data in (a) based on the top 2 eigenvectors of the graph Laplacian of the data, computed using an RBF kernel; K-means now can easily detect the two clusters

Spectral clustering: summary¹⁸



¹⁸[J. Suykens and C. Alzate (2011). Kernel spectral clustering: model representations, sparsity and out-of-sample extensions.]

Spectral clustering: some last elements

- Other Laplacian choices are possible, as the normalized one

$$\mathbf{L}^{\text{norm}} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{S} \mathbf{D}^{-1/2}$$

- Its overall complexity is $O(n^2)$ thus problem when n large
- It is in fact [equivalent to kernel \$K\$ -means](#) modulo relaxed version¹⁹

¹⁹See details in [I.S. Dhillon *et al.* (2004). Kernel kmeans, Spectral Clustering and Normalized Cuts. KDD'04.]

Hierarchical agglomerative clustering: algorithm

Aim

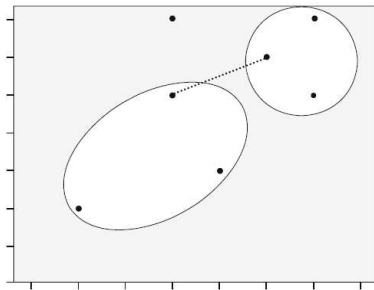
Design a procedure to build an indexed hierarchy

- **Start:** make a **dissimilarity matrix from δ** between singletons $\{\mathbf{x}_i\}$ of $\mathbf{x}$
- **Iterations:**
 - Merge two clusters that are the closest from a **linkage criterion Δ**
 - Compute dissimilarities between new clusters
- **End:** as soon as a unique cluster

- Need to define $\Delta : H \times H \rightarrow \mathbb{R}^+$ from δ
- If Δ is non-decreasing, it defines an index on the final hierarchy

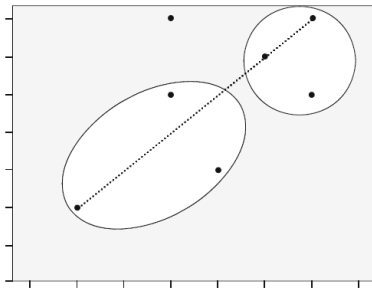
Single linkage criterion

$$\Delta(G_1, G_2) = \min\{\delta(\mathbf{x}_i, \mathbf{x}_{i'}) : \mathbf{x}_i \in G_1, \mathbf{x}_{i'} \in G_2\}$$



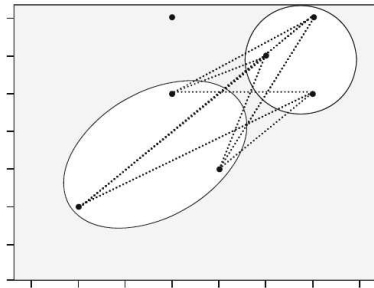
Complete linkage criterion

$$\Delta(G_1, G_2) = \max\{\delta(\mathbf{x}_i, \mathbf{x}_{i'}) : \mathbf{x}_i \in G_1, \mathbf{x}_{i'} \in G_2\}$$

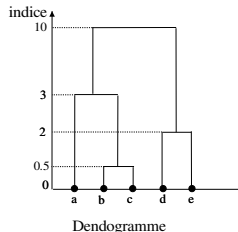
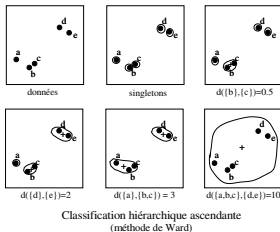


Average linkage criterion

$$\Delta(G_1, G_2) = \frac{1}{n_1 n_2} \sum_{x_i \in G_1} \sum_{x_{i'} \in G_2} \delta(x_i, x_{i'})$$



Hierarchical agglomerative clustering: Ward example



- A partition is obtained by cutting the dendrogram
- A dissimilarity matrix between pairs of individuals is enough
- Memory complexity $O(n^2 \ln n)$, space complexity $O(n^2)$ (do not scale well. . .)
- Suboptimal optimisation of $W_M(\cdot)$ (see next slide)

Link between Ward and K -means

Two successive Ward hierarchical algorithm iterations minimize W

- Partition at a given iteration: $P = \{G_1, \dots, G_K, G_{K+1}, G_{K+2}\}$

$$W(P) = \sum_{k=1}^K \sum_{\mathbf{x}_i \in G_k} \delta(\mathbf{x}_i, \boldsymbol{\mu}_k) + \sum_{\mathbf{x}_i \in G_{K+1}} \delta(\mathbf{x}_i, \boldsymbol{\mu}_{K+1}) + \sum_{\mathbf{x}_i \in G_{K+2}} \delta(\mathbf{x}_i, \boldsymbol{\mu}_{K+2})$$

- Partition at a the next iteration: $P^+ = \{G_1, \dots, G_K, \underbrace{G_{K+1} \cup G_{K+2}}_{G_{K+1}^+}\}$

$$W(P^+) = \sum_{k=1}^K \sum_{\mathbf{x}_i \in G_k} \delta(\mathbf{x}_i, \boldsymbol{\mu}_k) + \sum_{\mathbf{x}_i \in G_{K+1} \cup G_{K+2}} \delta(\mathbf{x}_i, \boldsymbol{\mu}_{K+1}^+)$$

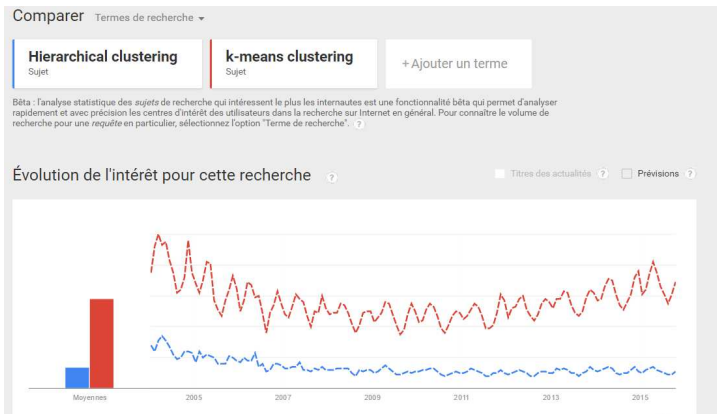
- After some algebra and using the Koenig-Huygens theorem, we obtain

$$W(P^+) - W(P) = \frac{n_{K+1}n_{K+2}}{n_{K+1} + n_{K+2}} \delta(\boldsymbol{\mu}_{K+1}, \boldsymbol{\mu}_{K+2})$$

- Thus, it is like performing K -means under (partial) partition P constraints

Popularity of K -means and hierarchical clustering

Even K -means was first proposed over 50 years ago, it is still one of the most widely used algorithms for clustering for several reasons: ease of implementation, simplicity, efficiency, empirical success. . . and model-based interpretation (see later)



Data set used for illustration

US State Facts and Figures

Description

Data sets related to the 50 states of the United States of America.

state.x77 : matrix with 50 rows and 8 columns giving the following statistics in the respective columns.

Population population estimate as of July 1, 1975

Income : per capita income (1974)

Illiteracy illiteracy (1970, percent of population)

Life Exp : life expectancy in years (1969–71)

Murder : murder and non-negligent manslaughter rate per 100,000 population (1976)

HS Grad : percent high-school graduates (1970)

Frost : mean number of days with minimum temperature below freezing (1931–1960) in capital or large city

Area : land area in square miles

Source

U.S. Department of Commerce, Bureau of the Census (1977) *Statistical Abstract of the United States*.

U.S. Department of Commerce, Bureau of the Census (1977) *County and City Data Book*.

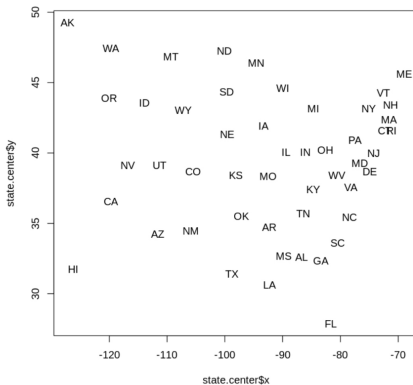
References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

HCA & kmeans example

Load data US census

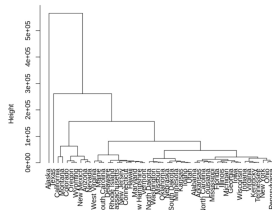
```
5]: data(state)
plot(state.center,type='n') # affiche position etats...
text(state.center,state.abb) # ... avec abréviations
```



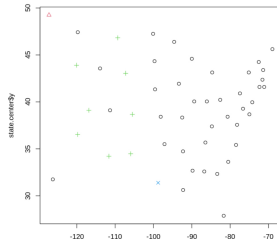
CAH avec "complete" linkage

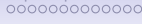
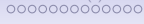
```
R> hc <- hclust(dist(state.x77),method='complete') # CAH...
plot(hc,hang=1) # ... + dendrogramme
partitionUS <- cutree(hc,4) # partition en 4 classes
plot(state.center,pch=partitionUS,col=partitionUS) # répartition géographique
```

Cluster Dendrogram



dist(state.x77)
hclust("complete")

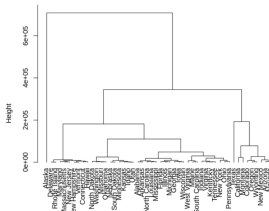




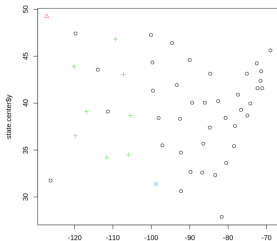
CAH avec "Ward" linkage

```
[8]: hc <- hclust(dist(state.x77),method='ward.D2') # CAH...
      plot(hc,hang=-1) # ... + dendrogramme
      partitionUS <- cutree(hc,4) # partition en 4 classes
      plot(state.center,pch=partitionUS,col=partitionUS) # répartition géographique
```

Cluster Dendrogram

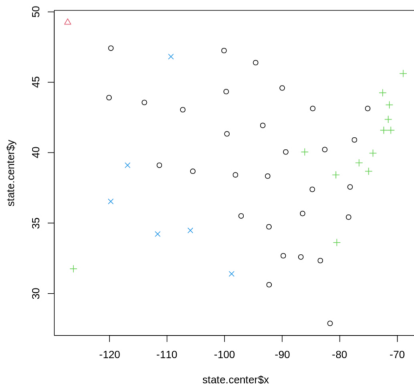


```
dist(state.x77)
hclust ("ward.D2")
```



kmeans en 4 classes

```
[9]: km <- kmeans(state.x77, 4, nstart=5)
      plot(state.center,pch=km$cluster,col=km$cluster) # répartition géographique
```



Semi-supervised clustering²⁰

- The user has to provide any external information he has on the partition
- Pair-wise constraints:
 - A **must-link constraint** specifies that the point pair connected by the constraint belong to the same cluster
 - A **cannot-link constraint** specifies that the point pair connected by the constraint do not belong to the same cluster
- Attempts to derive constraints from domain ontology and other external sources into clustering algorithms include the usage of WordNet ontology, gene ontology, Wikipedia, etc. to guide clustering solutions

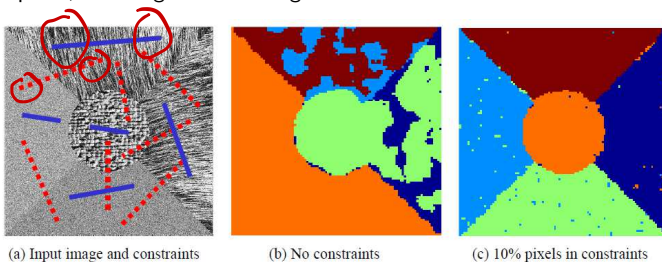
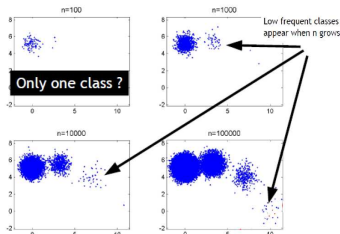


Figure 12 Semi-supervised learning. (a) Input image with must-link (solid blue lines) and must not link (broken red lines) constraints. (b) Clustering (segmentation) without constraints. (c) Improved clustering with 10% of the data points included in the pair-wise constraints [6].

²⁰O. Chapelle *et al.* (2006), A.K. Jain (2008). Data Clustering: 50 Years Beyond K-Means.

Online clustering

- **Dynamic data** are quite recent: blogs, Web pages, retail chain, credit card transaction streams, network packets received by a router and stock market, etc.
- As the data gets modified, **clustering must be updated** accordingly: ability to detect emerging clusters, etc.
- Often all data **cannot be stored on a disk**
- This imposes additional requirements to traditional clustering algorithms to **rapidly process and summarize** the massive amount of continuously arriving data
- Data stream clustering a significant challenge since they are expected to involved **single-pass algorithms**



Next lesson

Clustering : une vision unifiée pour une utilisation éclairée
Evaluation d'une méthode de clustering