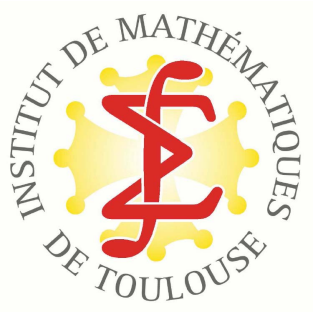


Institut de Mathématiques de Toulouse, INSA Toulouse

Introduction, Exploration, Unsupervised classification

Data Mining
October, 2024

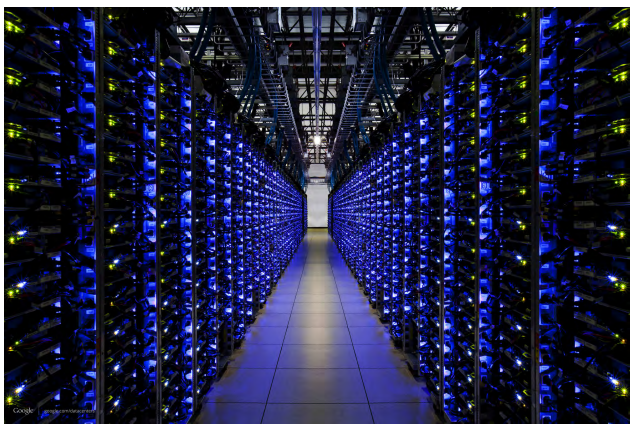
Béatrice Laurent - Philippe Besse - Olivier Roustant

Outline

- Introduction
- Multidimensional Exploration
- Unsupervised classification

Introduction

- Statistical learning plays a key role in many fields of sciences, medicine, industry, marketing, finance ..
- The development of data storage and computing resources leads to the storage of a huge amount of data from which the data scientist will try to learn crucial informations to better understand the underlying phenomena or to provide predictions.



- Many fields are impacted, here are some examples of learning problems :
  - **Signals :** Aerospace industry produces a huge amont of signal measurements obtained from thousand of on-board sensors. It is particularly important to detect possible anomalies before launching the satellite. Similarly, many sensors are involved in planes and it is important to detect a abnormal behavior on a sensor.
    $\hookrightarrow$ curve clustering or classification and anomaly detections in a set of curves for predictive maintenance purposes.
  - **Images :** Convolutional neural networks and deep learning lead to important progresses for image classification. Many fields are concerned : medical images (e.g. tumor detection), earth observation satellite images, photos, video surveillance images, handwritten text images ...
  - **Geolocalisation data :** Machine learning based on geolocalisation data has also many potential applications : targeted advertising, road traffic forecasting, monotoring the behavior of fishing vessels ...

- **Consumers preferences data :** Websites and supermarkets collect a huge amount of data on the behavior of consumers. Machine learning algorithms are used to valorize these data (gathered sometimes with personal data such as age, sex, job, adress ..) for recommandation systems ..

- **Microarray data :** DNA microarrays allow to measure the expression of thousands of genes simultaneously on a single individual. It is, for example, a challenge to try to infer from those kind of data which genes are involved in a certain type of cancer. The number p of genes measured on a microarray is generally much larger than the number n of individuals in the study.
  $\hookrightarrow$ Variable selection in high dimension.

From Statistic to AI through *Data Science*

1930-70 h-Octets Statistical inference

1950 Beginnings of Artificial Intelligence : Allan Turing

1970s kO Data analysis and *exploratory data analysis*

1980s MO Neural networks, functional data analysis

1990s GO *Data mining* : pre-acquired data

2000s TO Bioinformatics : $p \gg n$, *Machine Learning*

2008 Data Science

2010s PO Big Data p and n very large

2012 *Deep Learning*

2016 Artificial Intelligence (IA) : AlphaGo

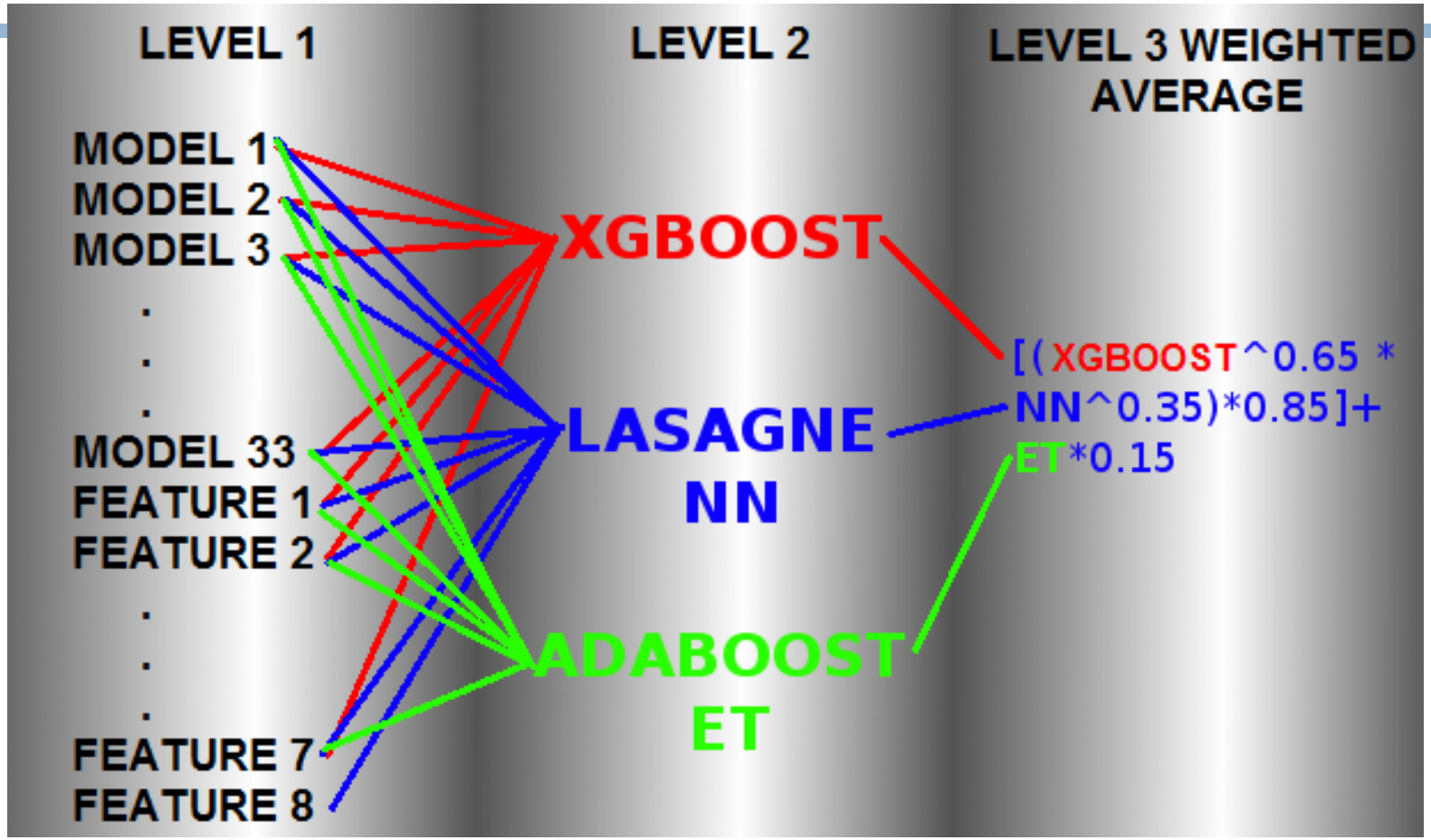
VVV...VV : Volume, Variety, Velocity... Veracity, Valorization

Objectives

- Exploration : description, visualisation, clustering (taxonomy)
- Explanation $Y = f(\boldsymbol{X})$ (supervised learning)
- Prediction and selection, explanable and interpretable models
- "Raw" forecasting (black box models)
- Anomaly Detection

Aim

- Academic publication (*Benchmarks — UCI repository*)
- Valorisation, Industrial solutions
- Kaggle type competition.



Kaggle competition : Identify people who have a high degree of Psychopathy based on Twitter usage.

Usecase Ozone

Aim : Prediction of the ozone concentration for the next day at 5 PM (max. of the day) from a learning sample composed of the explanatory variables $X^1, \dots X^p$:

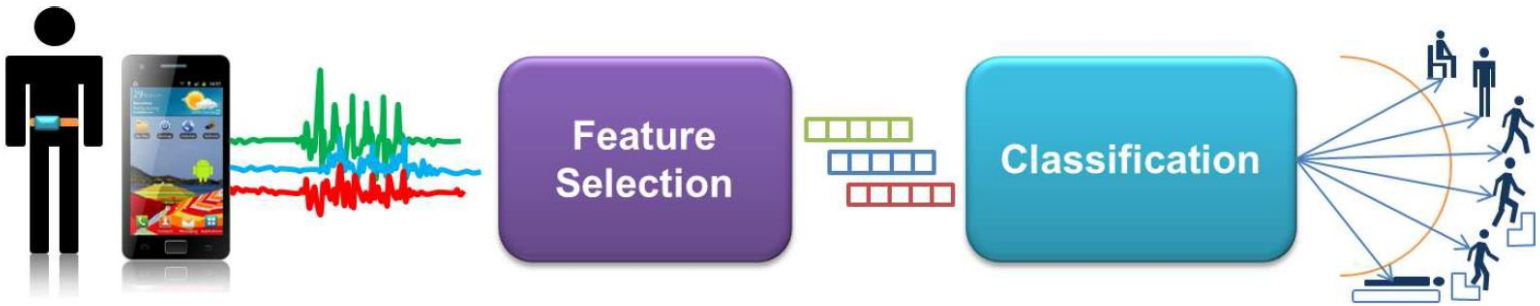
- MOCAGE (deterministic model of Meteo France),
- NO2, NO3,
- H2O,
- Temperature,
- Wind speed and orientation,
- Station,
- Type of day (holiday or not)

and the variable to explain :

- Y : Ozone concentration

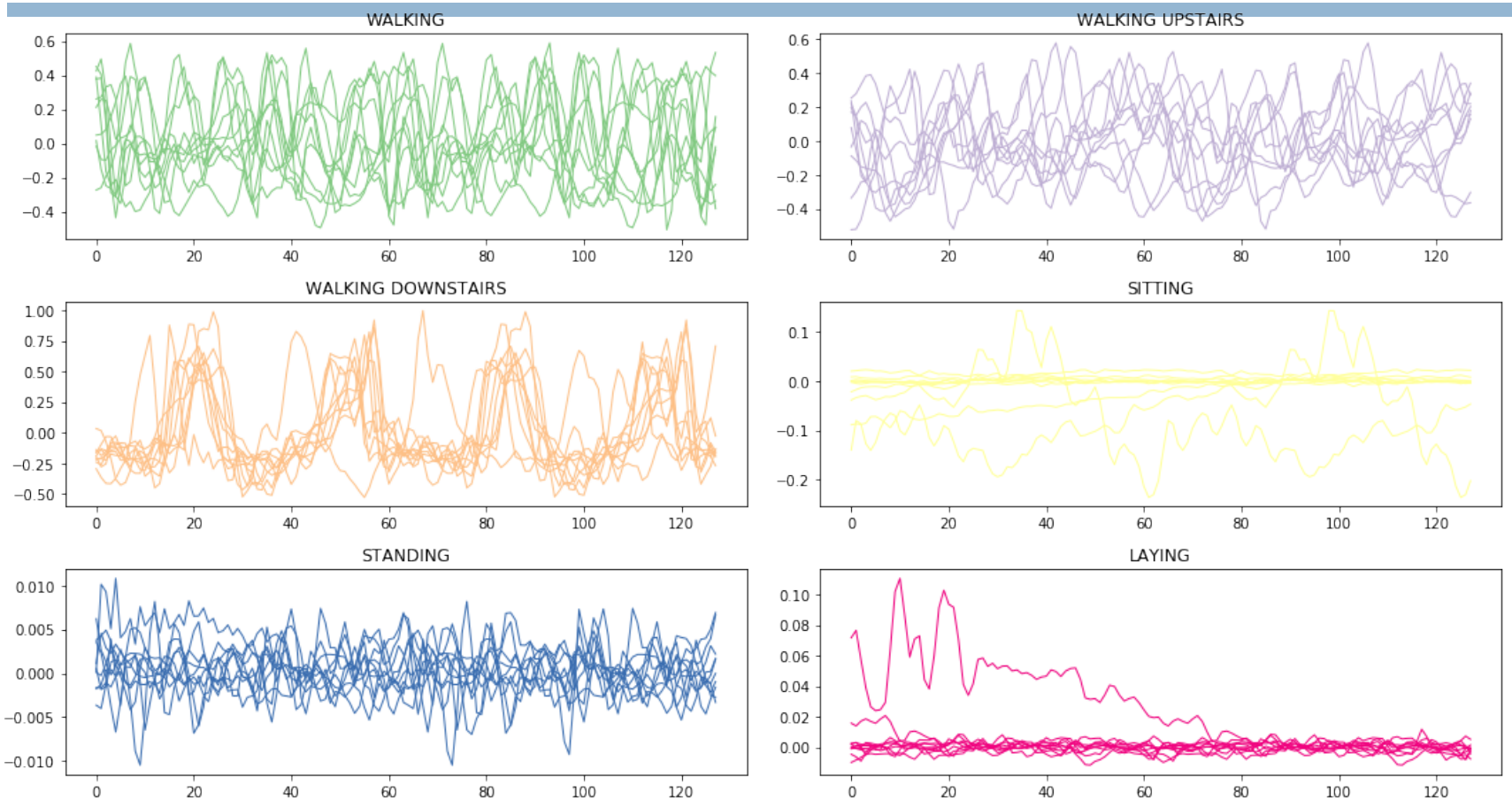
$\hookrightarrow$ **Statistical adaptation**

Usecase HAR



Human activity recognition HAR

- Public data available on *UCI repository*
- 9 signals per individual : accelerations in x, y, z, those by subtracting the natural gravity and angular accelerations in x, y, z obtained from the gyroscope.
- Each signal contains $p = 128$ measures sampled at 64Hz during 2s.
- 7352 samples for learning and 2947 for testing.
- Objectives : Activity recognition (6 classes) standing, sitting, lying, walking, walking upstairs or walking downstairs.



Human activity recognition : acceleration in y by class

HAR First step : "features" variables obtained from signal processing

- $p = 561$ new variables (*features*)
  - Time domain : min, max, means, variances, correlations...
  - Frequency domain : largest, mean, energy per frequency band...

HAR ... to be continued

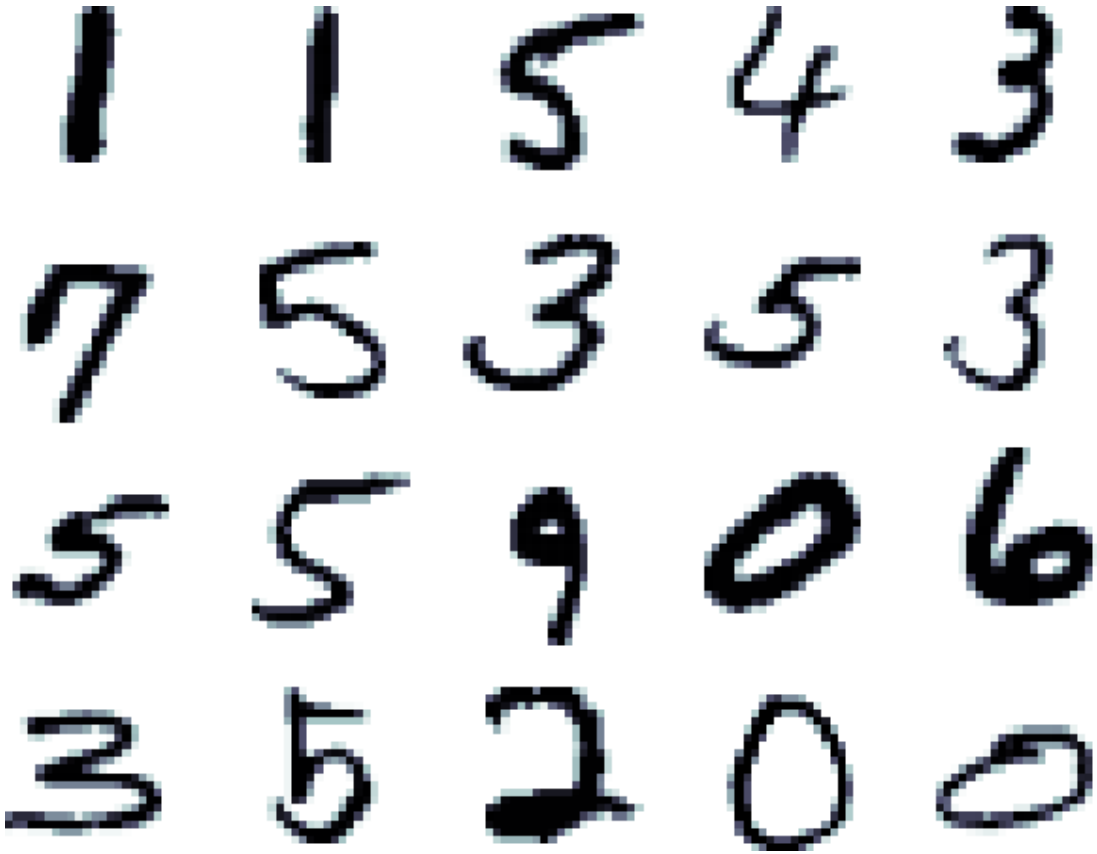
- raw signals and *deep learning*

Usecase MNIST

MNIST dataset

- Yann le Cun website
- 60 000 handwritten digits, $28 \times 28 = 784$ pixels
- Test : 10 000 images
- Classical methods (k-nn, Random Forests)
- Preprocessing : normalisation of the images
- Specific Distance with invariance properties
- Deep learning : *TensorFlow, Keras*

Usecase MNIST



MNIST : some examples of handwritten digits

Unsupervised vs Supervised learning

- In the framework of **supervised learning**, we have a **learning sample** composed of observation data of the type **input/output** :

$$d_1^n = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$$

where $\boldsymbol{x}_i = (x_i^1, x_i^2, \ldots, x_i^p)$ is a p-dimensional variable (quantitative or qualitative or both), $y_i \in \mathcal{Y}$ for $i = 1 \ldots n$ is the variable to explain (label).

Objectives : From the learning sample, we want to

- **Estimate** the link between the input vector $\boldsymbol{x}$ (explanary variables) and the output y :

$$y = f(x^1, x^2, \ldots, x^p)$$

- **Predict** the output y associated to a new entry $\boldsymbol{x}$,
- **Select** the important explanatory variables among $x^1, \ldots, x^p$.

Unsupervised vs Supervised learning

- **Supervised learning**,

quantitative output	qualitative output
$\mathcal{Y} \subset \mathbb{R}^p$	$\mathcal{Y}$ finite
$\downarrow$	$\downarrow$
regression	**classification**

- In the framework of **unsupervised learning**, we only observe $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$.

  Objectives :
  - Find underlying structures in these unlabeled data.
  - Clustering.

Steps of Data Science

1. Data management
2. Exploration
3. Modeling, supervised or unsupervised learning
4. Scaling up, industrialization

Step 0 : Data management

Data management

- Acquisition
- Archiving
- Extractions
- Fusion
- Quantification : frequency of occurence, discretization ...

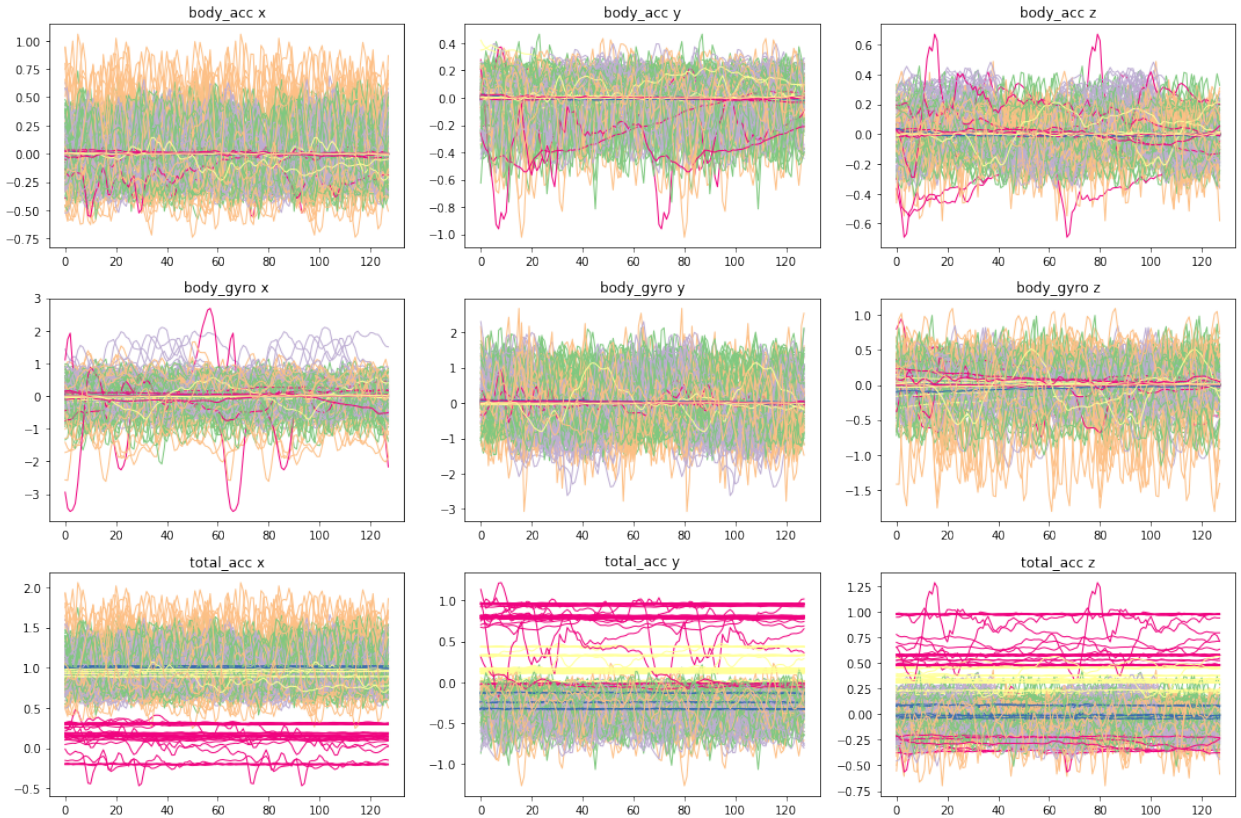
Data (matrix)

- p variables $X = (X^1, \ldots, X^p)$
- n individuals or observations
- Y variable to explain

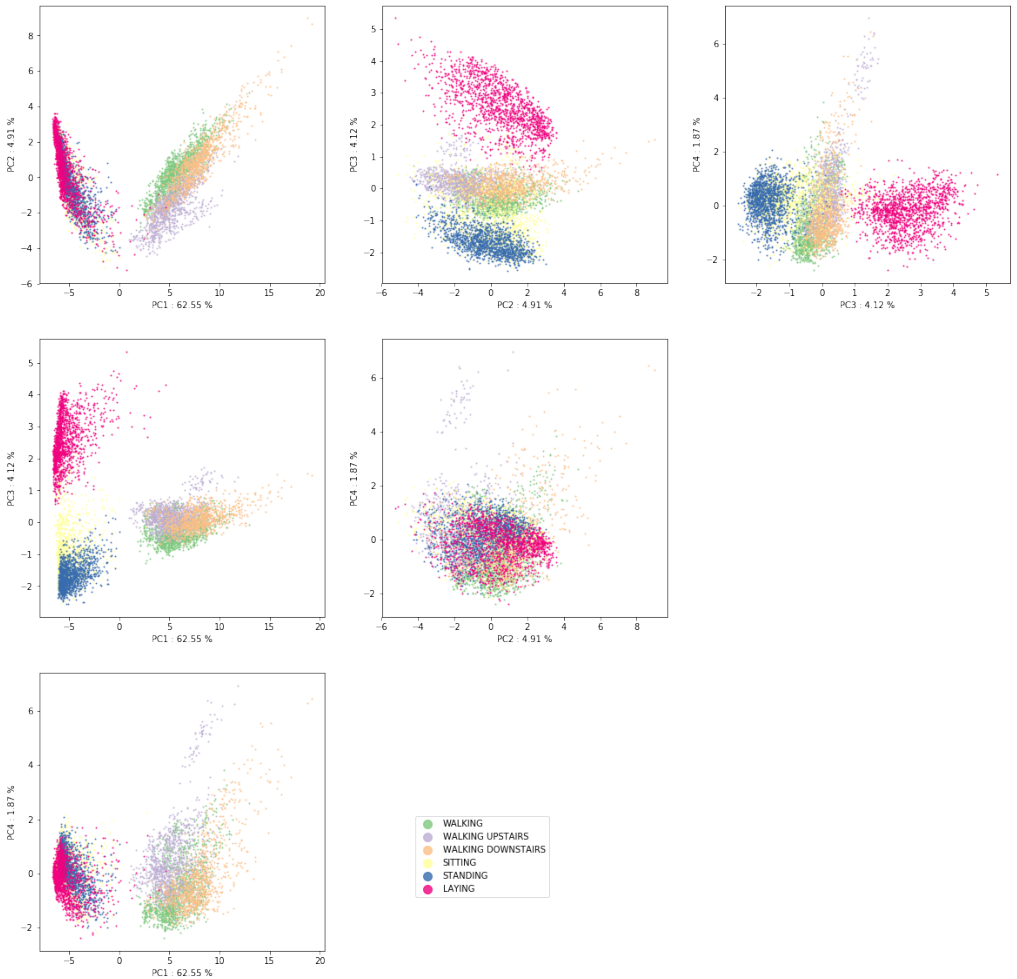
Step 1 : Exploration

Data munging

- Extraction, cleaning, consistency check
- Elementary statistics : univariate, bivariate
- Distribution, transformation of the variables
- New variables or features
- Outlier detection
- Missing data (imputation ?)
- Dimension reduction : Principal Component Analysis, Factorial Discriminant Analysis
- Visualisation



HAR : raw signals



HAR : Principal component analysis of the signal processing features

Step 2 : Modeling, supervised or unsupervised learning

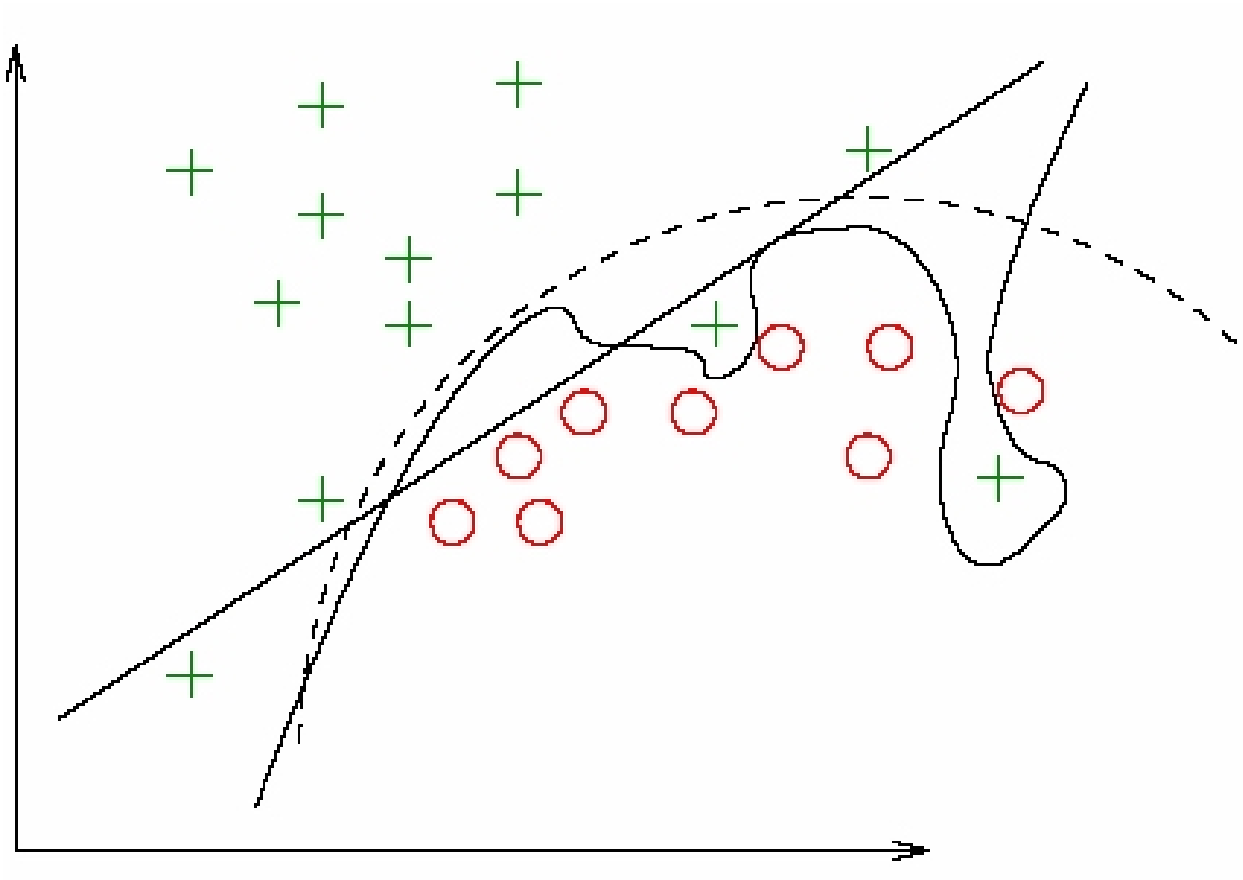
Unsupervised Classification (*clustering*)

- HCA : Hierarchical Cluster Analysis
- *k*-means, *k*-medoids (PAM)
- Gaussian mixtures
- DBSCAN, Self Organizing Map

Supervised Learning (Classification)

- Logistic Regression
- Discriminant analysis, k-nearest neighbours
- Support Vector Machine
- Classification And regression Trees (CART)
- Bagging, Random Forests
- Boosting
- Neural Networks, Deep Learning

Supervised learning : risk of overfitting

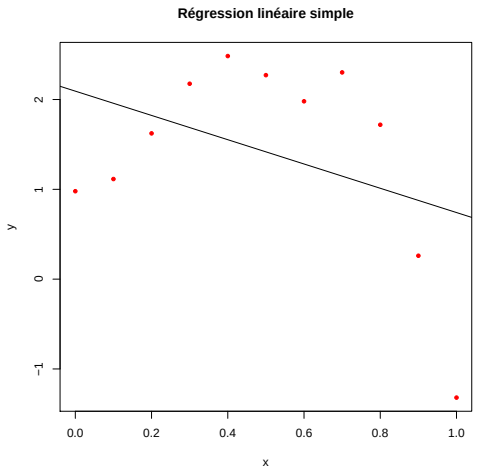


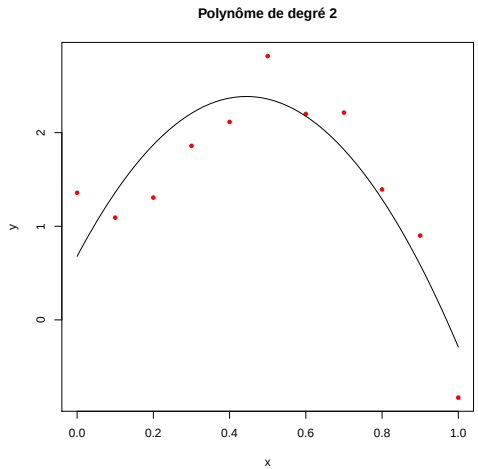
Model complexity in supervised classification

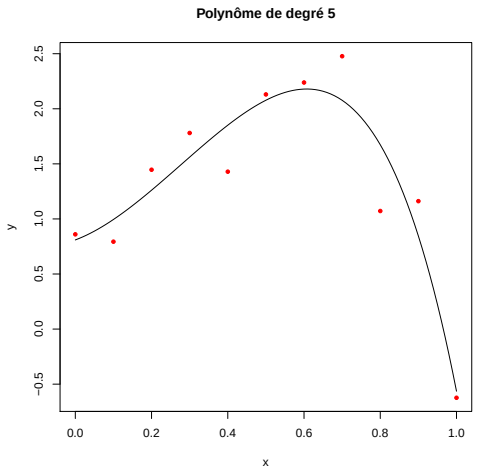
Supervised Learning (Regression)

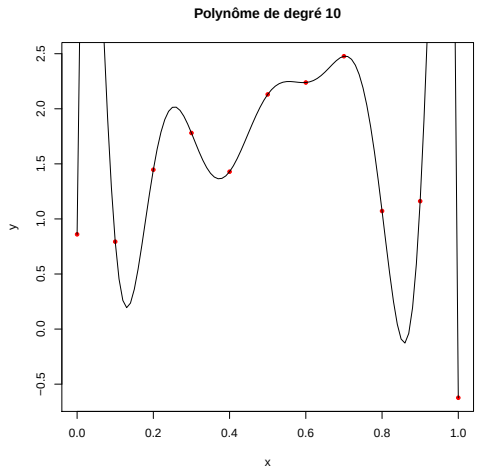
- Linear regression
- k nearest neighbours
- Support Vector Regression
- Classification And Regression Trees (CART)
- Bagging, Random Forests
- Boosting
- Neural Networks, Deep Learning

Supervised learning : risk of overfitting









Model complexity in polynomial regression

Strategy for supervised *Learning*

1. Random Partition of the sample : learning, (validation), test
2. **For** each method that we consider :
   - Learning (estimation) depending on θ (complexity)
   - Optimization of θ : validation set or cross-validation with the learning set
3. Comparison of the methods : prediction error on the test sample
4. Eventual Iteration (*Monte Carlo*)
5. Choice of the method (prevision *vs.* interpretability).
6. Estimation of the selected model with all the sample, exploitation

Possibly : Aggregation of several models

Outline

- Introduction
- Multidimensional Exploration
  - Basic tools
  - Principal components analysis
  - Factorial discriminant analysis
- Unsupervised classification

Multidimensional Exploration

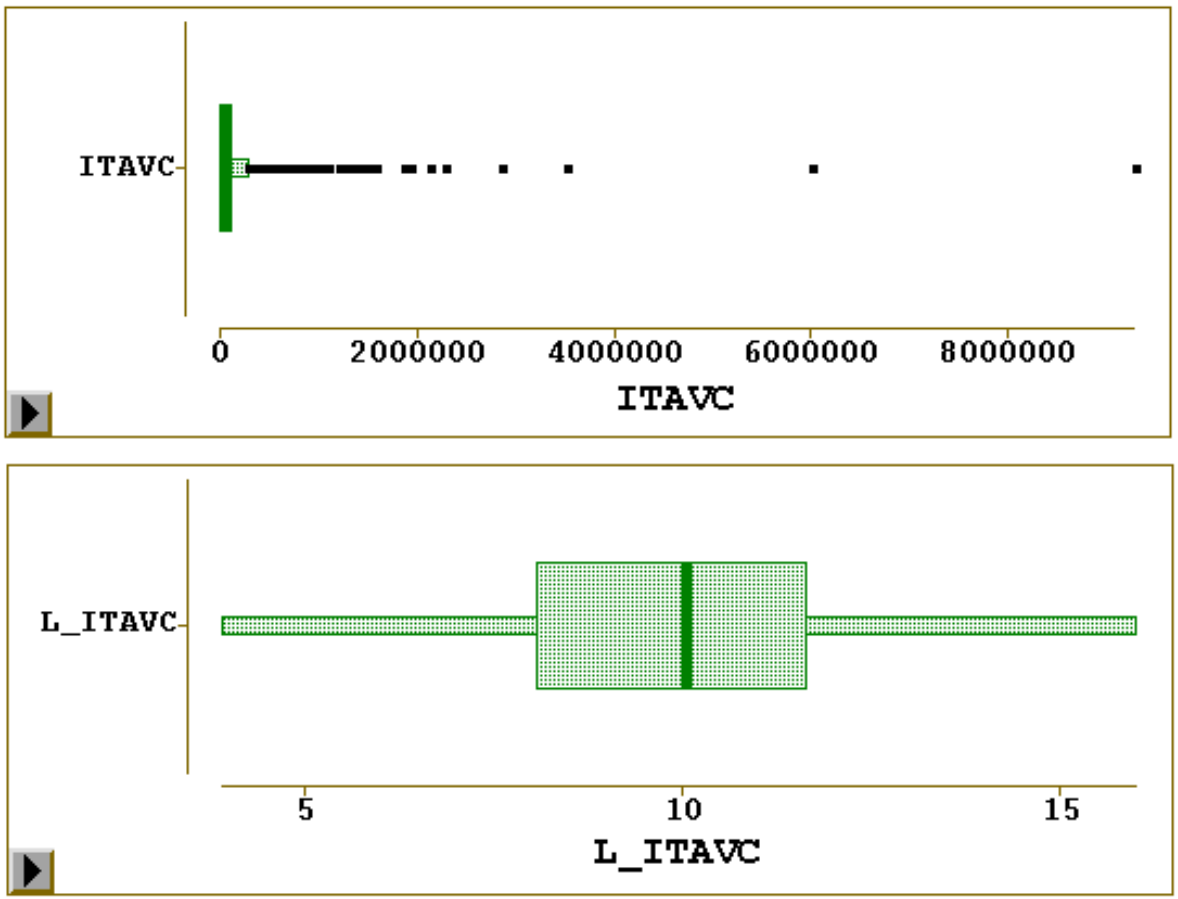
Some basic tools

- Univariate and bivariate description
- Principal components analysis

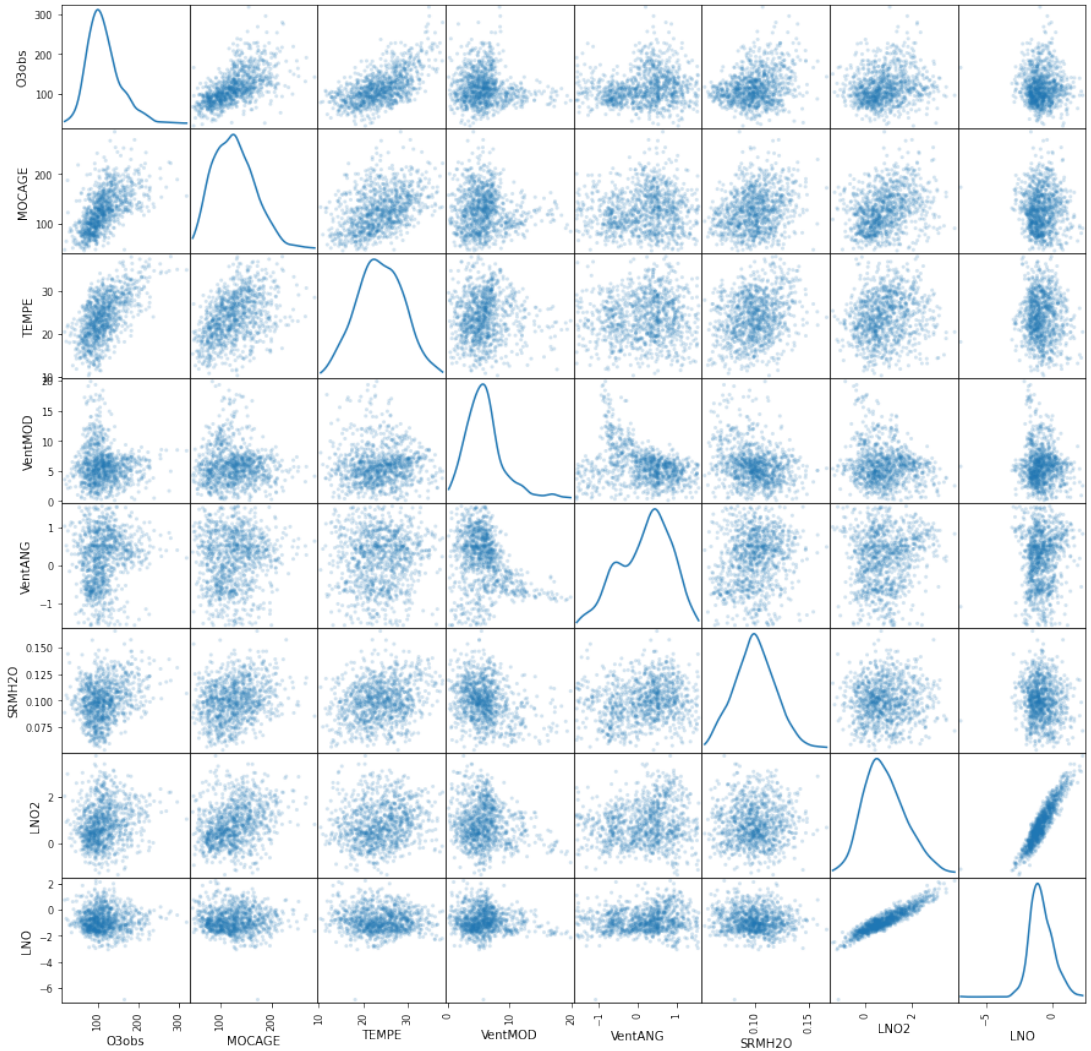
Diagnosis

- Error detection, inconsistency
- Detection of outliers
- Normality of the distributions

Log-normal variable transformation



ScatterPlot



Ozone data set

Outline

- Introduction
- Multidimensional Exploration
  - Basic tools
  - Principal components analysis
  - Factorial discriminant analysis
- Unsupervised classification

Principal Component Analysis : objectives

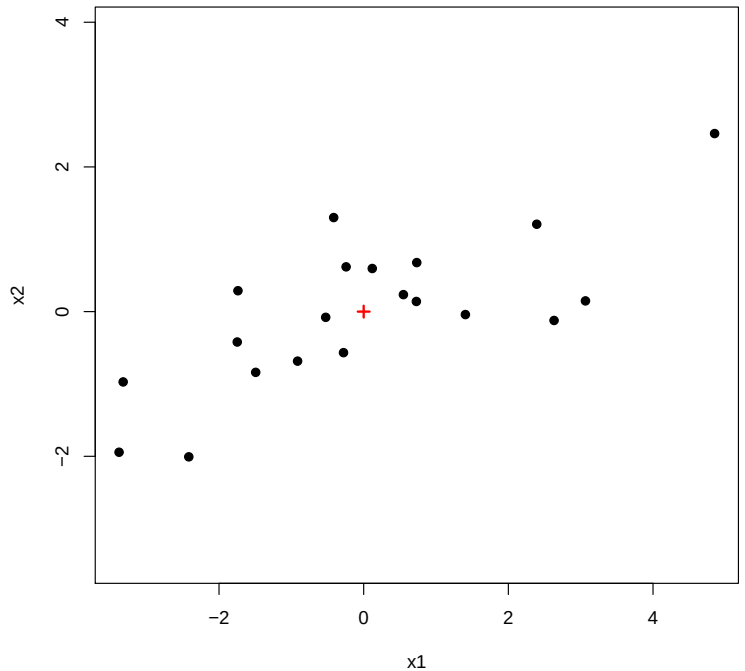
- We have a dataset composed with n individuals on which we observed p variables.
- If $p > 2$ (or 3), it is impossible to represent the individuals.

Objectives of PCA :

- Perform a dimension reduction to visualize the data.
- Identify trends on individuals (profiles) and "typical individuals" (e.g. in marketing, or in clinical trials)
- Have a synthetic visualization of the main links between the variables.
- Represent the data in smaller dimensions, avoiding "redundancies".

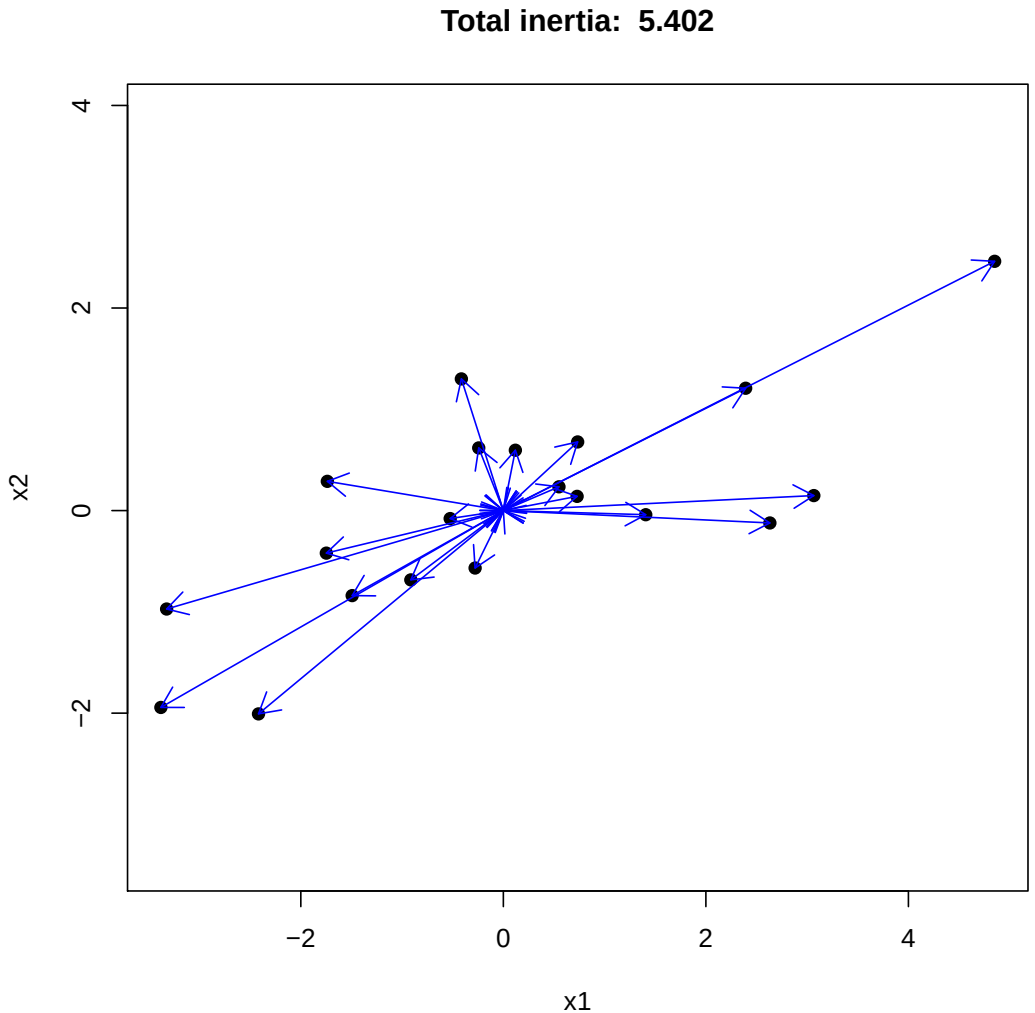
PCA only works on quantitative variables.

Illustration in 2 dimensions



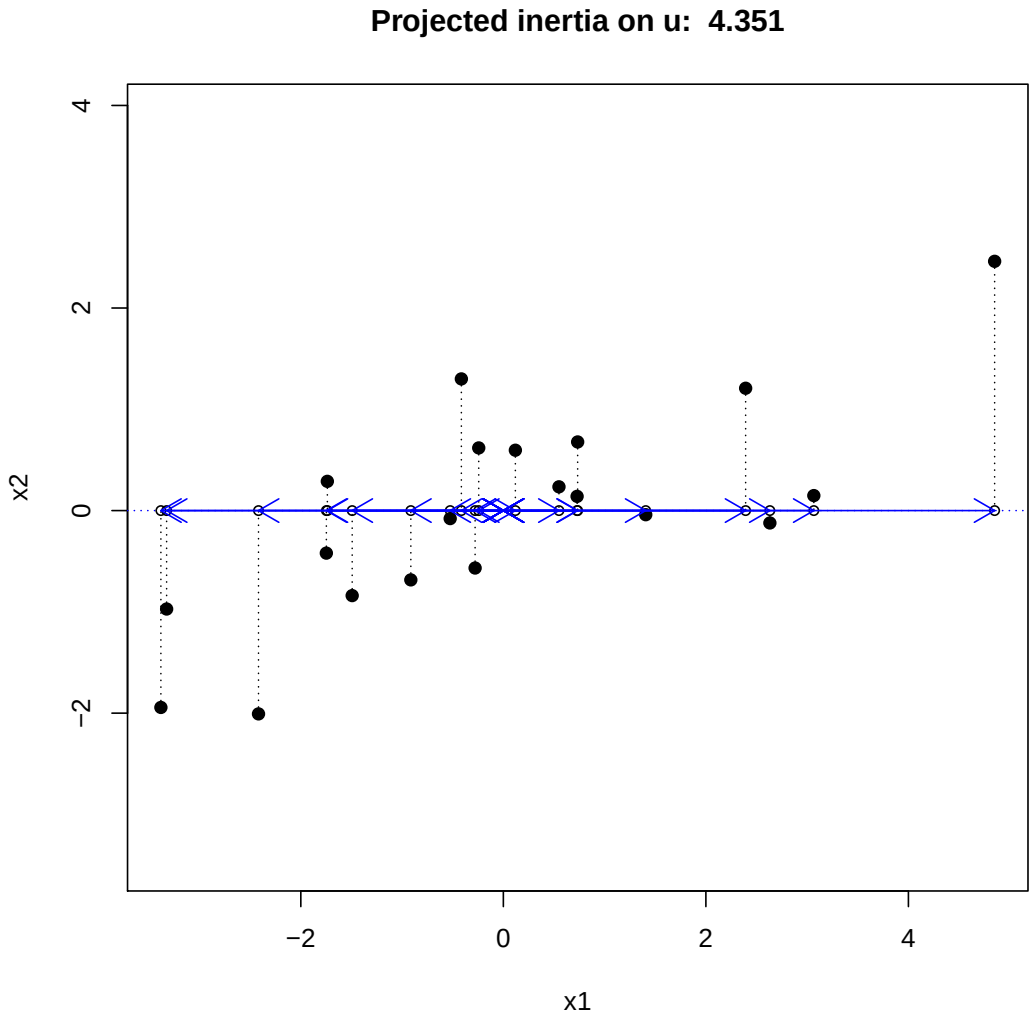
This is a $2D$ cloud of points, centered at 0.
Can you find a 1D axis 'containing' the maximum of information ?

Inertia



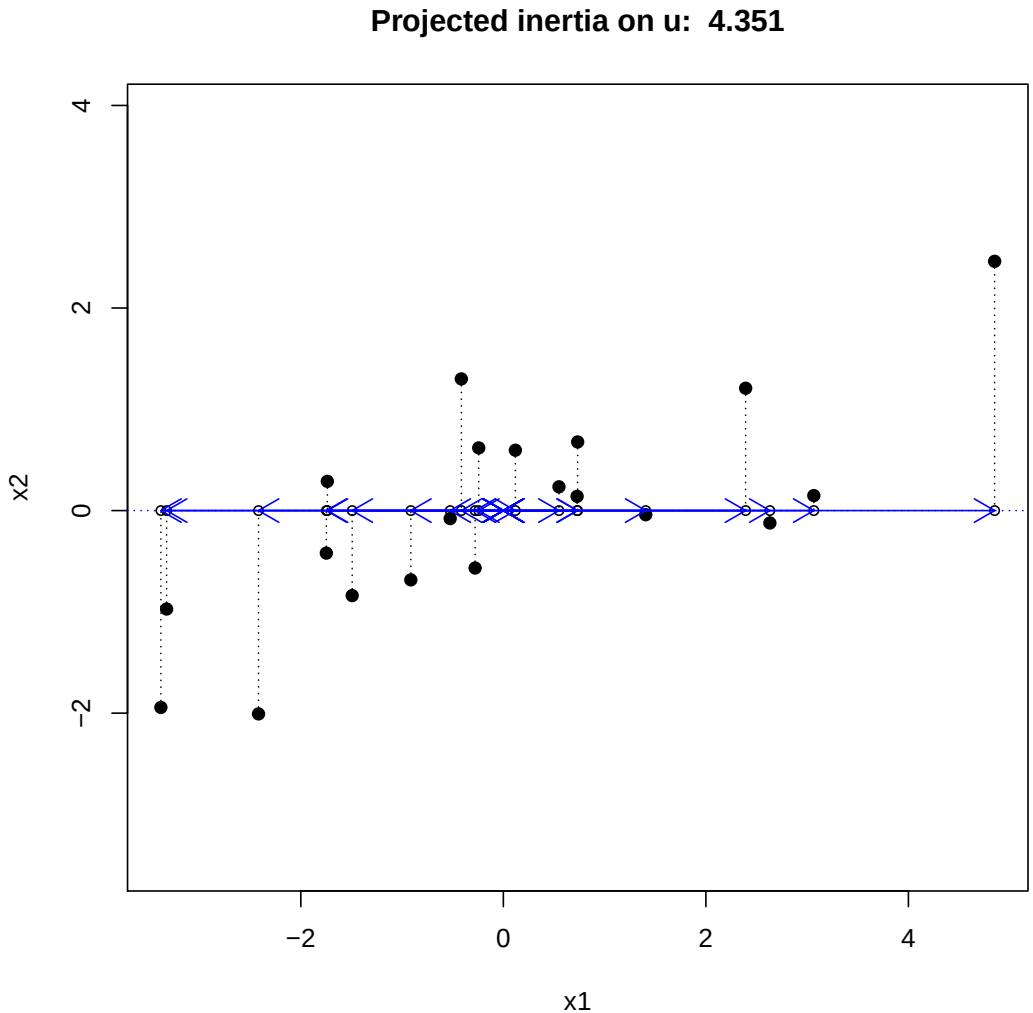
Total inertia : mean square of distances to the center.

Inertia



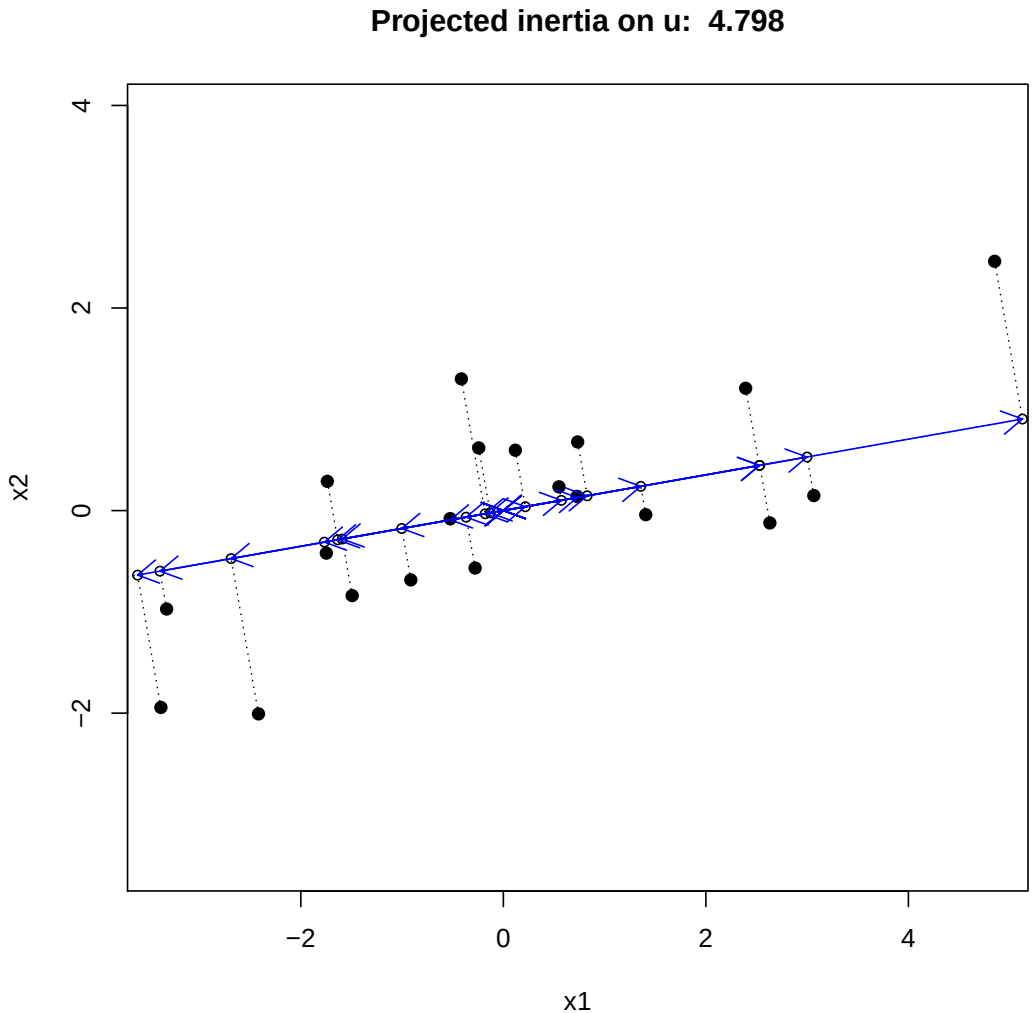
Projected inertia : inertia of projections. How much do we lose ?

Maximizing the projected inertia



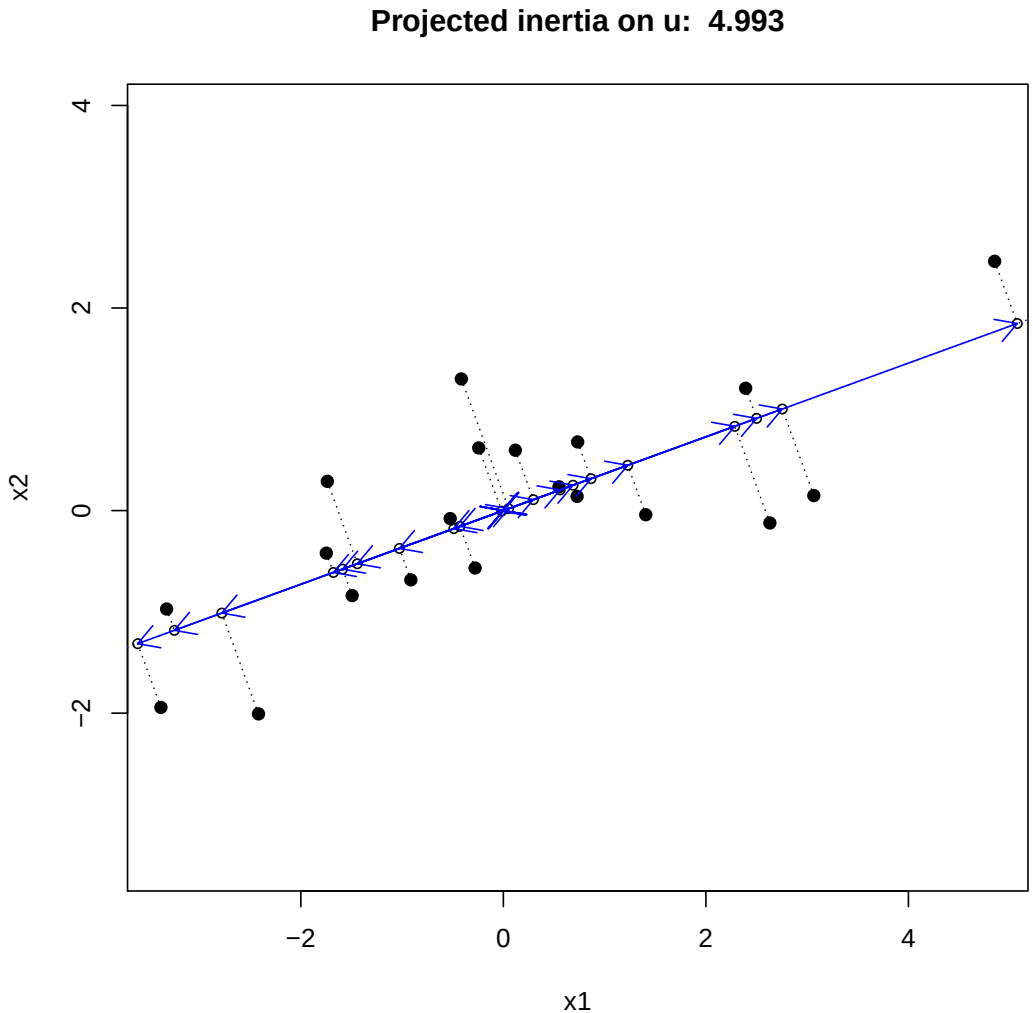
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



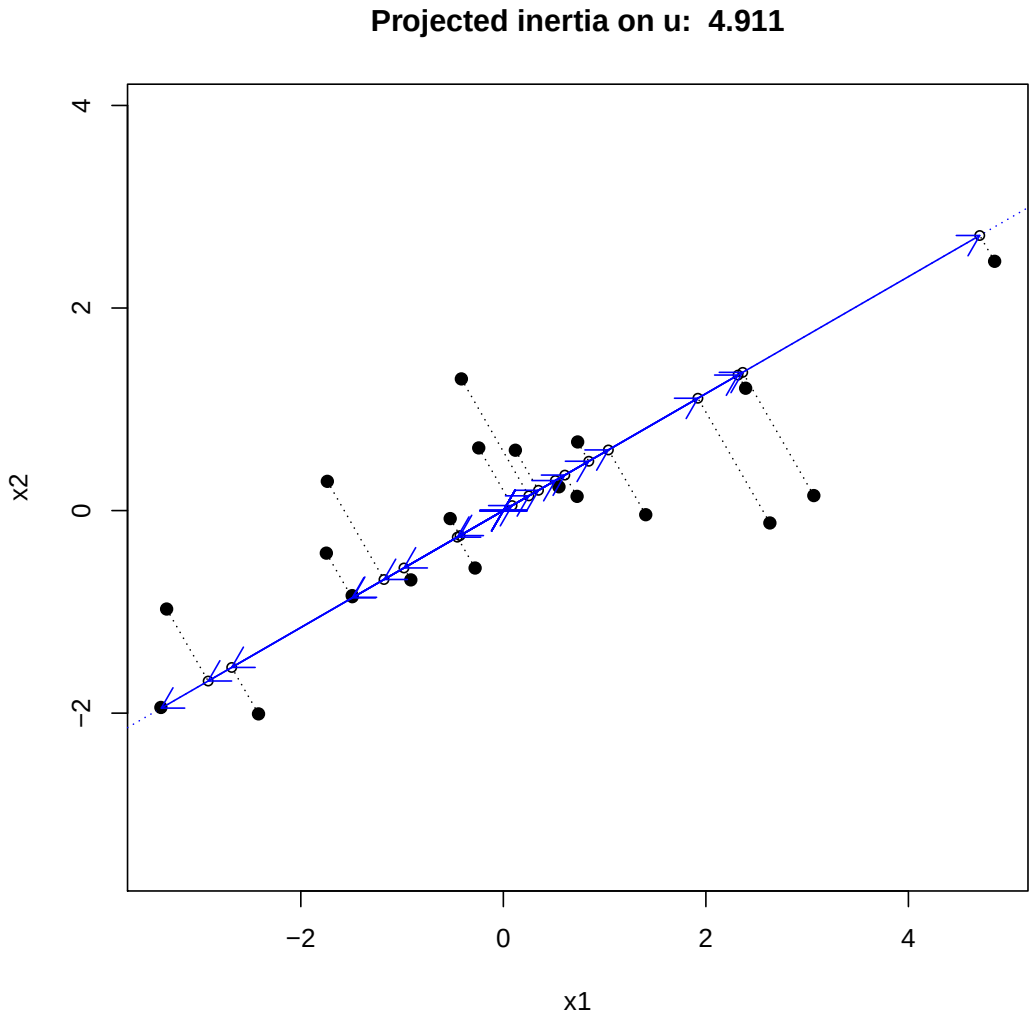
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



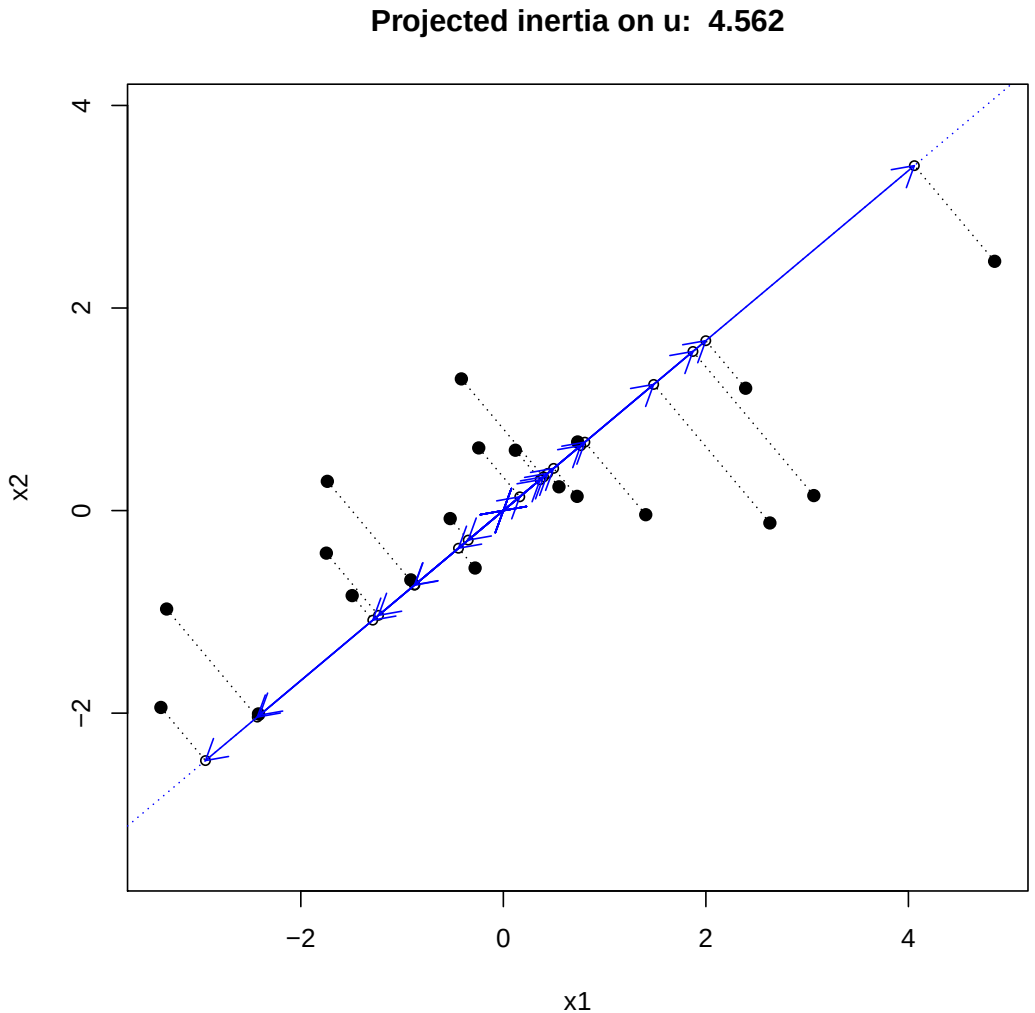
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



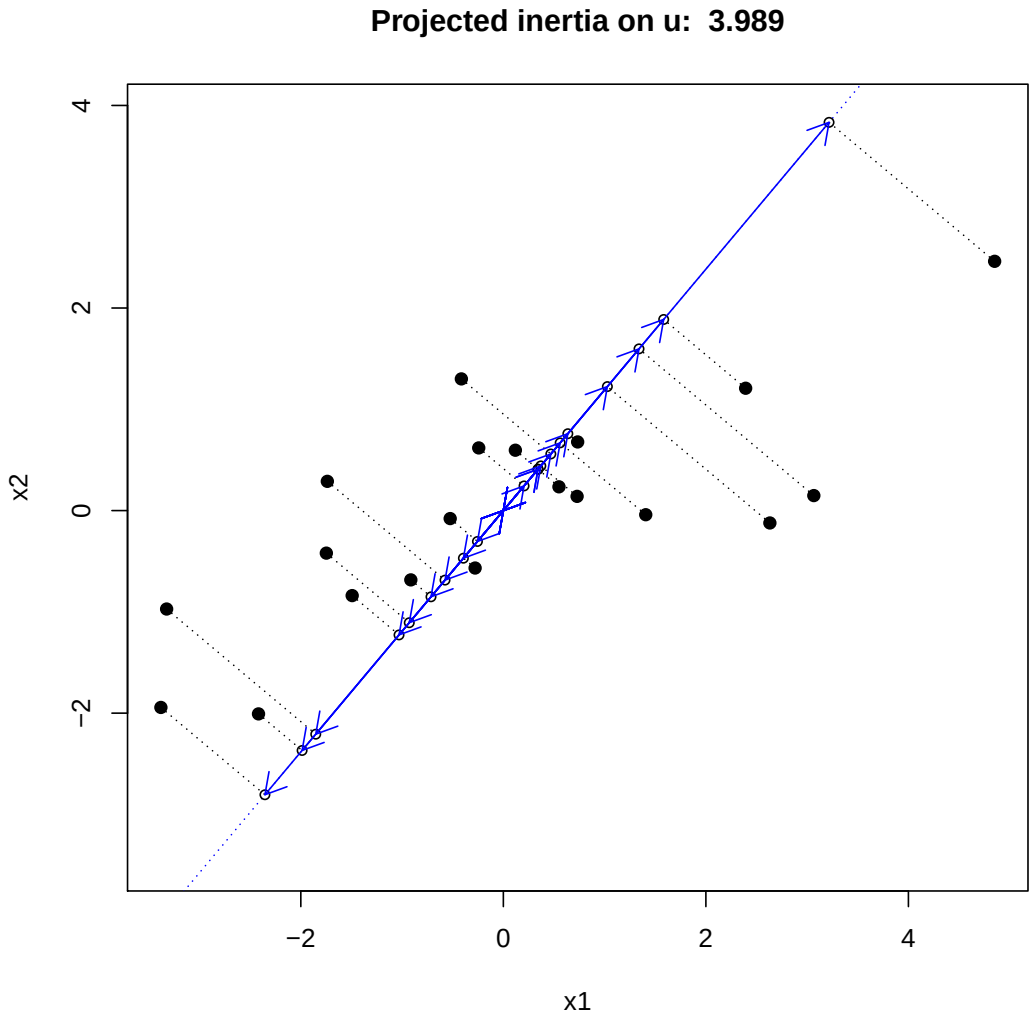
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



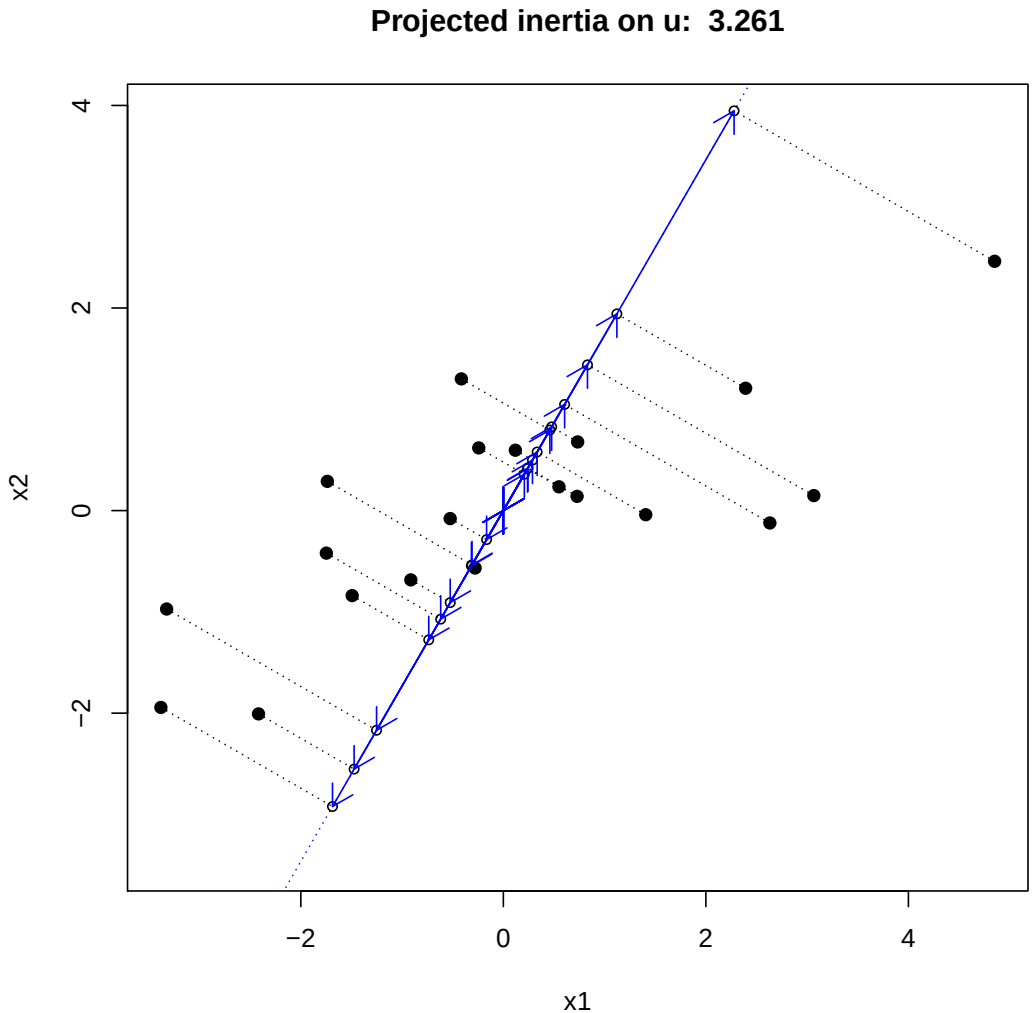
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



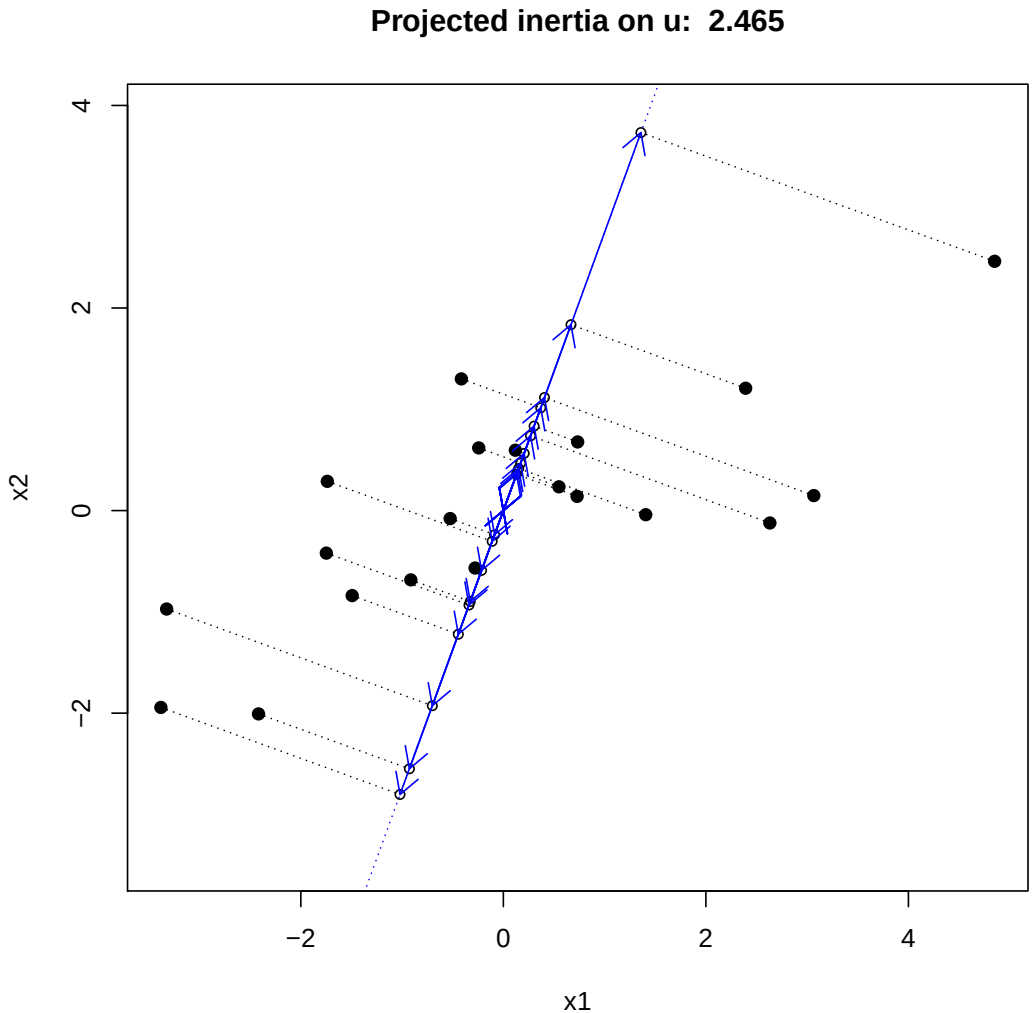
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



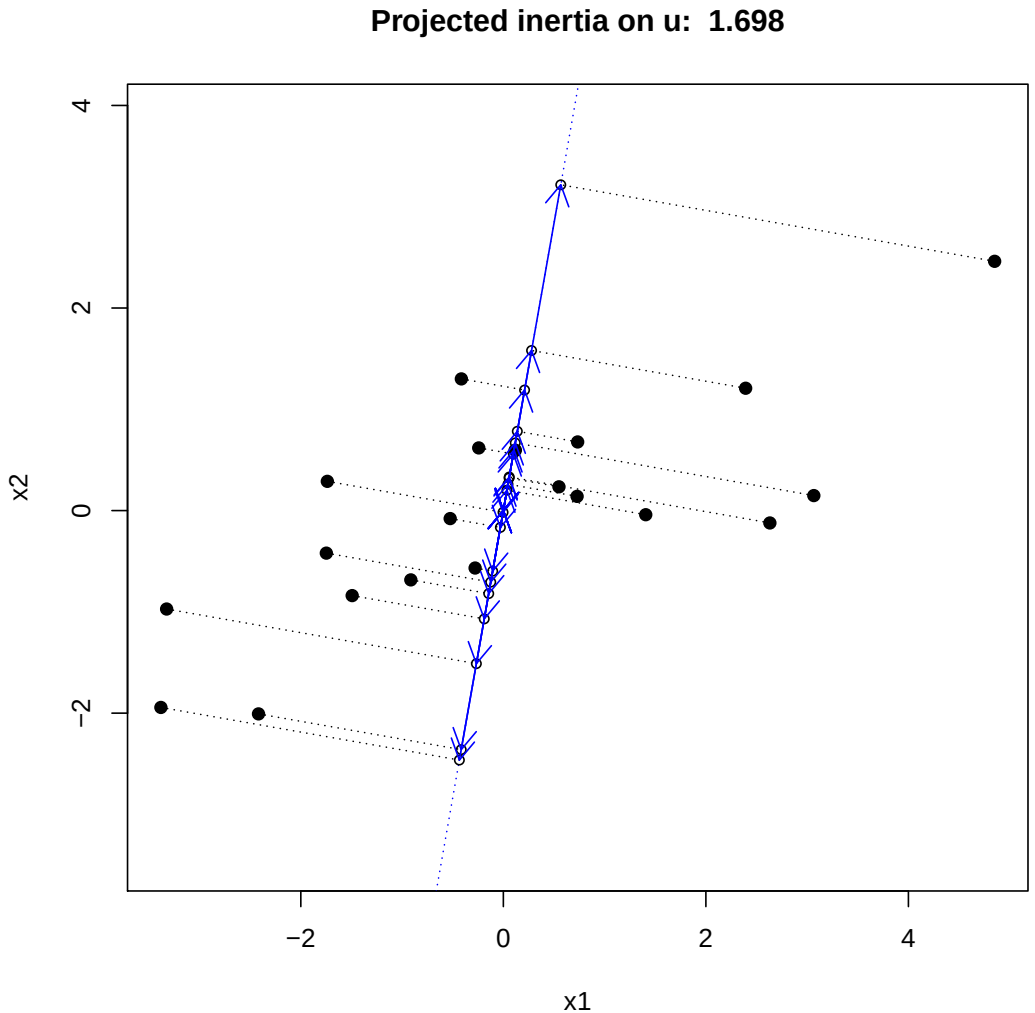
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



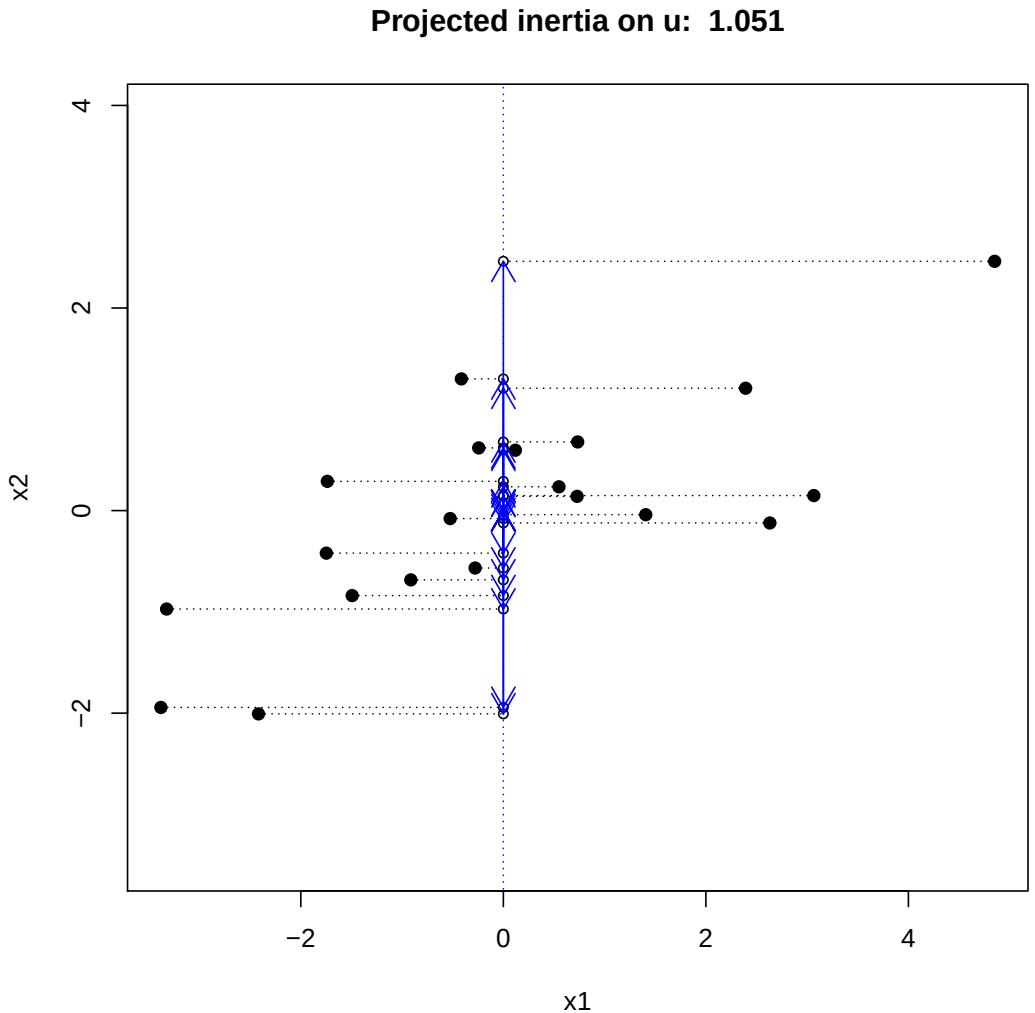
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



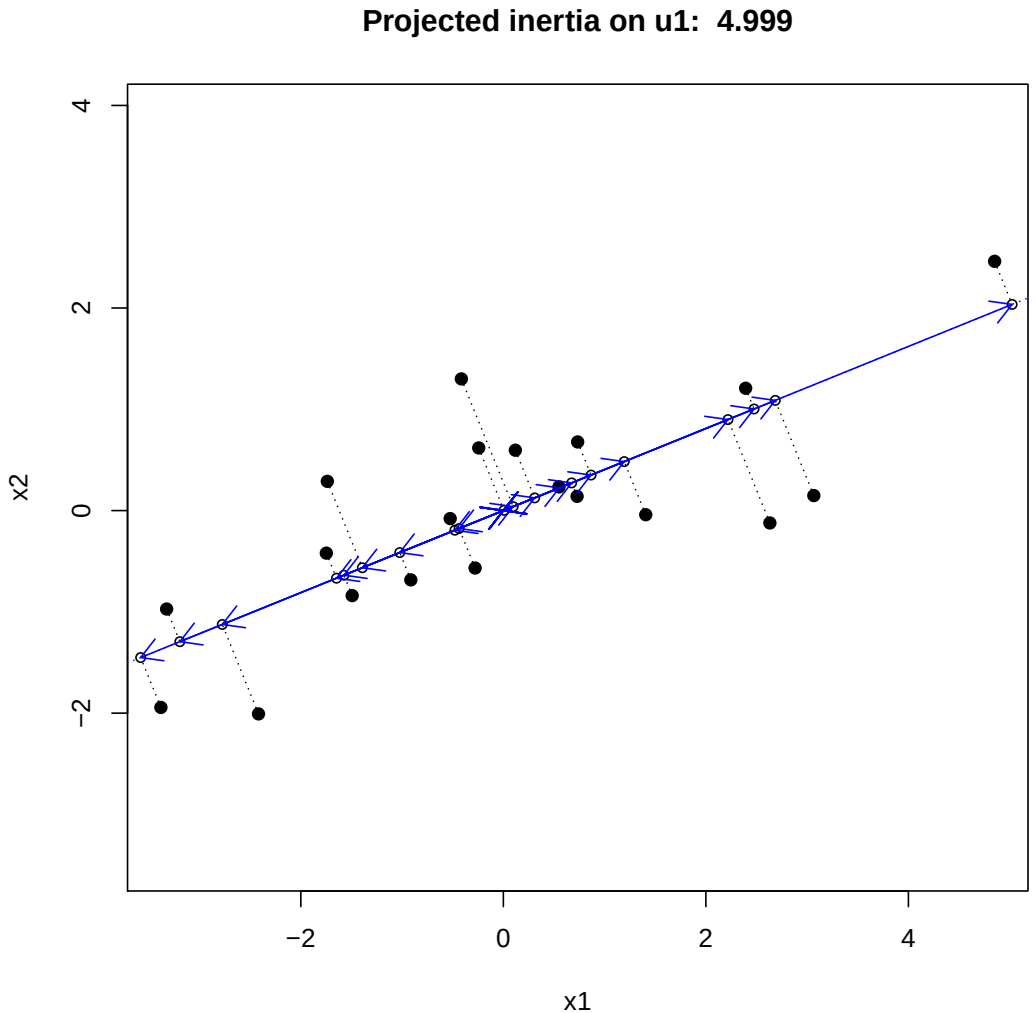
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



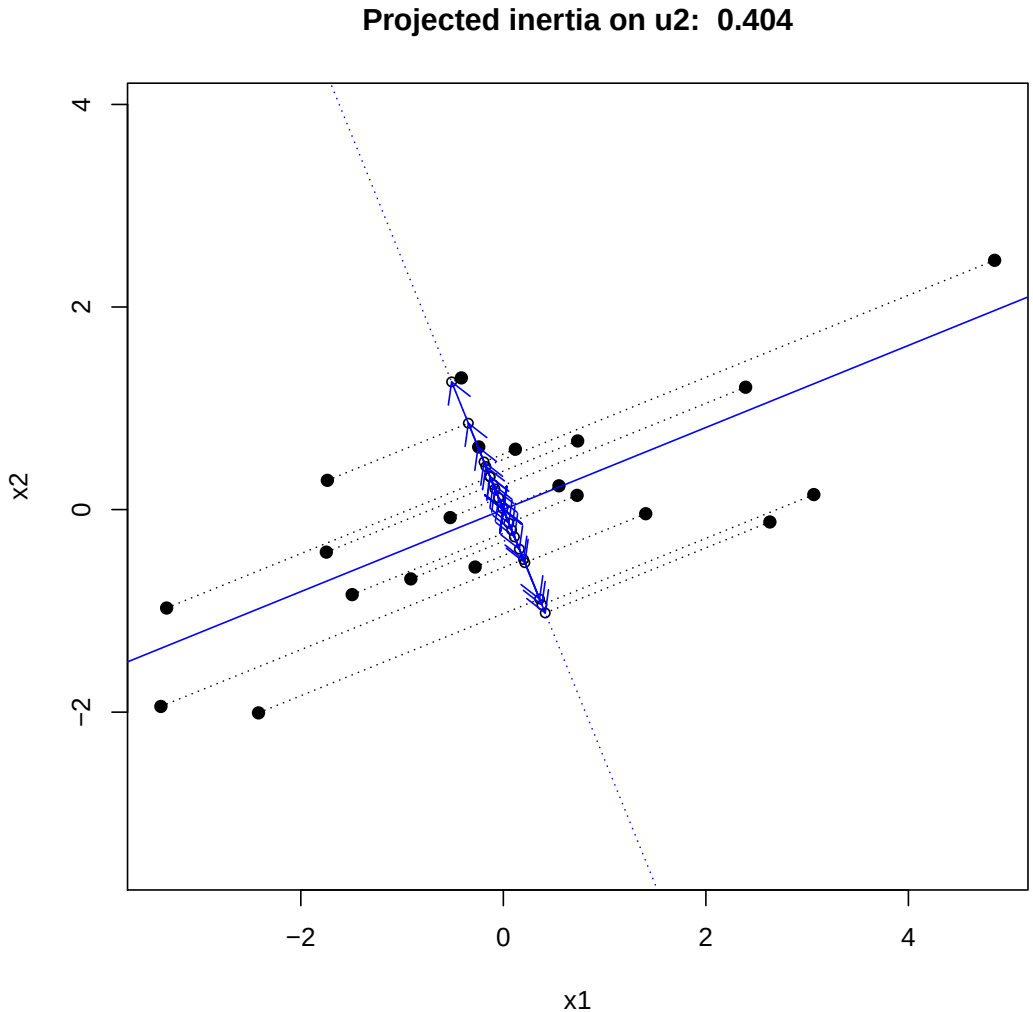
Projected inertia : For what axis is it maximal ?

Maximizing the projected inertia



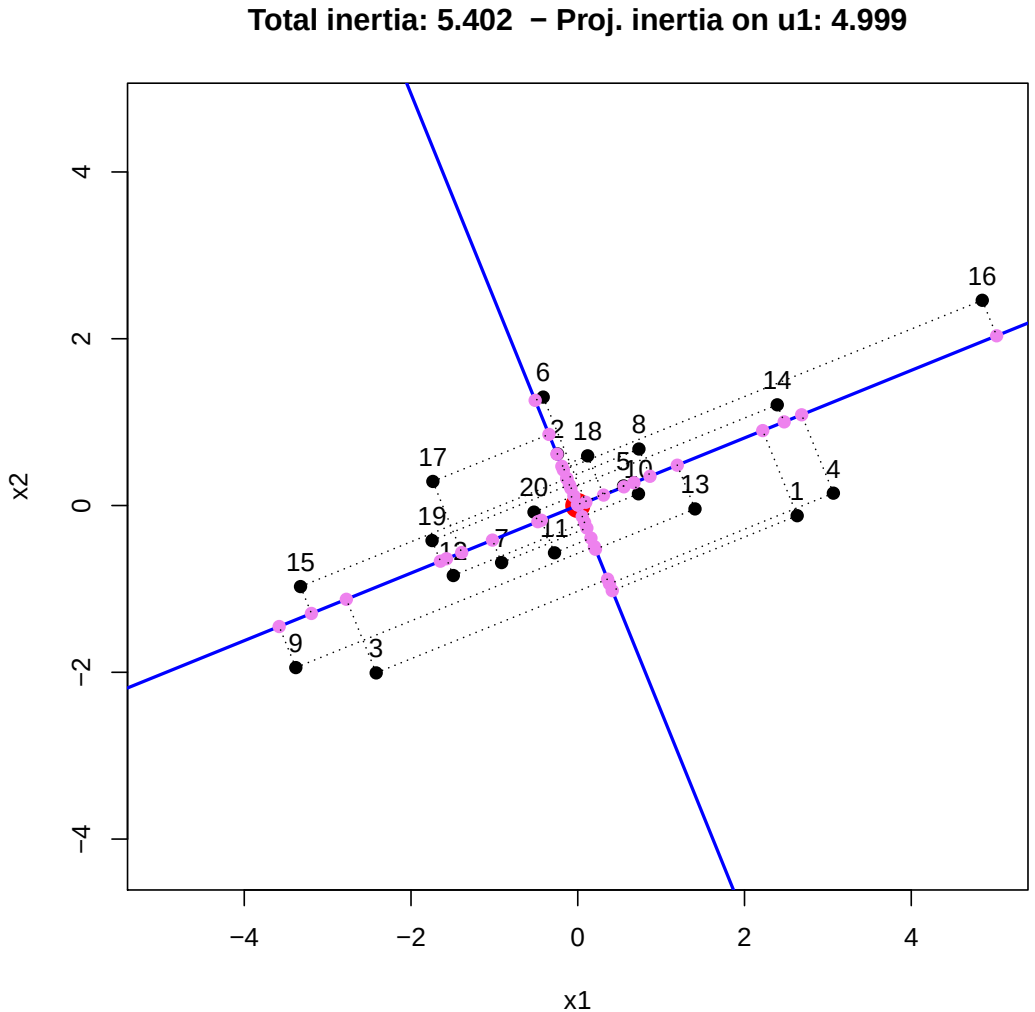
Projected inertia : Maximal for the largest eigenvalue of the covariance matrix

Maximizing the projected inertia, recursion



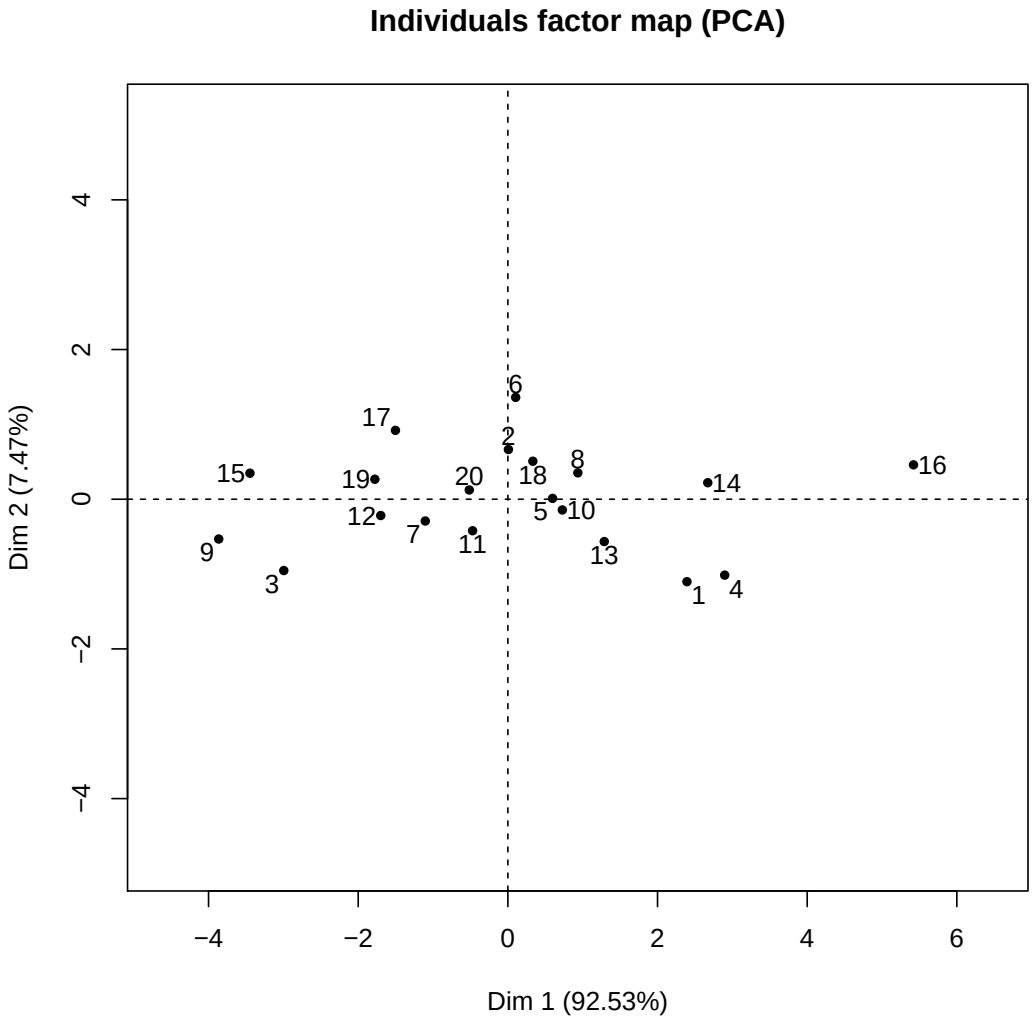
The second largest eigenvalue maximizes the projected inertia in the orthogonal of the first

Maximizing the projected inertia, summary



Projected points on the first two 'principal components'

Maximizing the projected inertia, summary



Representation with package FactoMineR. Percentages are inertia ratio w.r.t. total inertia

Notations and assumption

- $\mathbf{X}$: a matrix of size $n \times p$, representing the data :

	$\mathbf{x}^1$	$\dots$	$\mathbf{x}^j$	$\dots$	$\mathbf{x}^p$
$\mathbf{x}_1$	x_1^1	$\dots$	x_1^j	$\dots$	x_1^p
$\vdots$	$\vdots$		$\vdots$		$\vdots$
$\mathbf{x}_i$	x_i^1	$\dots$	x_i^j	$\dots$	x_i^p
$\vdots$	$\vdots$		$\vdots$		$\vdots$
$\mathbf{x}_n$	x_n^1	$\dots$	x_n^j	$\dots$	x_n^p

- $\mathbf{g}$: center of gravity (empirical mean), $\mathbf{g} = \bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \, (\in \mathbb{R}^p)$.

$\mathbf{g}$	$\overline{\mathbf{x}^1}$	$\dots$	$\overline{\mathbf{x}^j}$	$\dots$	$\overline{\mathbf{x}^p}$

We assume that $\mathbf{g} = \mathbf{0}$, i.e. the data have been centered.

Notations and assumption

- The rows of $\mathbf{X}$ lie in $\mathbb{R}^p$, and form the indivuals space.
  It is an Euclidean space, equipped with the usual ℓ^2 norm $\|.\|$.

- The columns of $\mathbf{X}$ lie in $\mathbb{R}^n$, and form the variables space.
  It is an Euclidean space. Instead of choosing the usual ℓ^2 norm, we rescale it by $1/n$. Indeed, as the data are centered, it corresponds to the empirical covariance :

$$\langle \mathbf{x}^j, \mathbf{x}^k \rangle_{\mathbb{R}^n} := \frac{1}{n} \sum_{i=1}^{n} x_i^j x_i^k = \widehat{\mathrm{cov}}(\mathbf{x}^j, \mathbf{x}^k).$$

  Notice that orthogonal variables = uncorrelated variables.
  Γ denotes the $p \times p$ empirical covariance matrix :

$$\Gamma = \left(\widehat{\mathrm{cov}}(\mathbf{x}^j, \mathbf{x}^k)\right)_{1 \leq j,k \leq p} = \frac{1}{n} \mathbf{X}^\top \mathbf{X} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top.$$

Notations and assumption

- Inertia : mean squared distance of the data to their center (here $\mathbf{0}$),

$$\mathcal{I} = \frac{1}{n} \sum_{i=1}^{n} \|\mathbf{x}_i\|^2$$

- Projected inertia on a subspace $F \subseteq \mathbb{R}^p$. Same definition for the projected points onto F (we denote by Π_F the projection operator) :

$$\mathcal{I}_F = \frac{1}{n} \sum_{i=1}^{n} \|\Pi_F(\mathbf{x}_i)\|^2$$

Properties of inertia

Link with variance, and inertia decomposition.

Consider a $1D$ axis spanned by a unit vector $\mathbf{a}$, and denote $\mathcal{I}_{\mathbf{a}} = \mathcal{I}_{\mathbb{R}\mathbf{a}}$. Then :

$$\mathcal{I}_{\mathbf{a}} = \mathbf{a}^\top \mathbf{\Gamma} \mathbf{a}, \qquad \text{and} \qquad \mathcal{I} = \mathcal{I}_{\mathbf{a}} + \mathcal{I}_{\mathbf{a}^\perp}$$

Moreover, $\mathcal{I}_{\mathbf{a}}$ and $\mathcal{I}$ are interpreted in terms of variances :

- $\mathcal{I}_{\mathbf{a}}$ is the empirical variance of the projected points onto $\mathbb{R}\mathbf{a}$,
- $\mathcal{I}$ is the sum of the empirical variances of the p variables :

$$\mathcal{I}_{\mathbf{a}} = \frac{1}{n}\sum_{i=1}^{n} \langle \mathbf{x}_i, \mathbf{a} \rangle^2, \qquad \mathcal{I} = \sum_{j=1}^{p} \hat{\sigma}_j^2, \quad \text{with} \quad \hat{\sigma}_j^2 = \frac{1}{n}\sum_{i=1}^{n} (x_i^j)^2$$

Remark : The empirical variances are computed here by dividing by n the sum of squares, contrarily to unbiased statistical estimates (division by $n-1$).

Main result

Theorem (principal component analysis)

As the covariance matrix Γ is real symmetric, it admits a spectral decomposition in orthogonal eigenspaces. Denote $\lambda_1 > \cdots > \lambda_p > 0$ the eigenvalues, and $\mathbf{v}_1, \ldots, \mathbf{v}_p$ orthogonal eigenvectors. Then :

- $\mathbf{v}_1$ maximizes $\mathcal{I}_{\mathbf{a}}$ over $\mathbf{a}$, which is then equal to λ_1.
- $\mathbf{v}_2$ maximizes $\mathcal{I}_{\mathbf{a}}$ over $\mathbf{a}$ in $(\mathbf{v}_1)^{\perp}$, which is then equal to λ_2.
- $\mathbf{v}_3$ maximizes $\mathcal{I}_{\mathbf{a}}$ over $\mathbf{a}$ in $(\mathbf{v}_1, \mathbf{v}_2)^{\perp}$, which is then equal to λ_3.
- ...

Furthermore the inertia (called *total inertia*) is decomposed :

$$\mathcal{I} = \mathcal{I}_{\mathbf{v}_1} + \cdots + \mathcal{I}_{\mathbf{v}_p} = \lambda_1 + \cdots + \lambda_p$$

Principal components

- The eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_p$ define a new orthonormal basis in $\mathbb{R}^p$.
- The change of variables is defined by :

$$\mathbf{C} = \mathbf{XP}, \qquad \text{with} \quad \mathbf{P} = [\mathbf{v_1}, \ldots, \mathbf{v_p}] .$$

The $n \times p$ matrix $\mathbf{C}$ is called matrix of principal components. The columns of $\mathbf{C}$ are called principal variables. They contain the coordinates of the individuals in the new space.

Principal components

- Principal variables are linear combinations of the original variables, with coefficients given by the eigenvectors :

$$\mathbf{C}^j = \mathbf{X}\mathbf{P}_j = \sum_{k=1}^{p} (\mathbf{v}_j)_k \mathbf{x}^k$$

- Principal variables are uncorrelated and $\widehat{\mathrm{var}}(\mathbf{C}^k) = \lambda_k$:

$$\left(\widehat{\mathrm{cov}}(\mathbf{C}^j, \mathbf{C}^k)\right)_{1 \leq j,k \leq p} = \frac{1}{n}\mathbf{C}^\top \mathbf{C} = \mathbf{P}^\top \mathbf{\Gamma} \mathbf{P} = \mathrm{diag}(\lambda_1, \ldots, \lambda_p).$$

Remark : singular value / spectral decomposition

PCA can be done with Singular Value Decomposition (SVD), which decomposes a rectangular matrix $n \times m$ or rank r as

$$\mathbf{X} = \mathbf{U}\Lambda^{1/2}\mathbf{V}^\top,$$

where Λ is the diagonal matrix containing the r non-zero eigenvalues of $\mathbf{X}^\top\mathbf{X}$ (or $\mathbf{X}\mathbf{X}^\top$), ranked by decreasing order, and $\mathbf{U}$ (resp. $\mathbf{V}$) is an orthogonal matrix for $\|.\|_{\mathbb{R}^n}$ (resp. for $\|.\|_{\mathbb{R}^m}$) containing the eigenvectors of $\mathbf{X}\mathbf{X}^\top$ (resp. $\mathbf{X}^\top\mathbf{X}$).

In the frequent case when $p = r$ (e.g. $n > p$), we have :

$$\mathbf{V} = \mathbf{P}, \qquad \Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_p).$$

(In the general case, $\mathbf{V}$ contains the r columns of $\mathbf{P}$ corresponding to non-zero eigenvalues.) Then, SVD can be used to obtain efficient formula, such as

$$\mathbf{C} = \mathbf{X}\mathbf{P} = \mathbf{U}\Lambda^{1/2}\mathbf{P}^\top\mathbf{P} = \mathbf{U}\Lambda^{1/2}.$$

Variations (metric, weights)

Changing the metric in the individuals space

Consider a new norm on $\mathbb{R}^p$, called metric, defined by a positive definite matrix $\mathbf{M}$, of size p :

$$\|\mathbf{x}\|_M^2 = \mathbf{x}^\top \mathbf{M}\mathbf{x}.$$

Let $\mathbf{R}$ be an invertible matrix s.t. $\mathbf{R}^\top\mathbf{R} = \mathbf{M}$ (e.g. square root, Choleski decomposition). Then, the map

$$\mathbf{R} : \begin{array}{ccc} (\mathbb{R}^p, \|.\|_M) & \to & (\mathbb{R}^p, \|.\|) \\ \mathbf{x} & \mapsto & \mathbf{R}\mathbf{x} \end{array}$$

is an isometry, and thus preserves distances and orthogonality.

Indeed : $\|\mathbf{R}\mathbf{x}\|^2 = (\mathbf{R}\mathbf{x})^\top(\mathbf{R}\mathbf{x}) = \mathbf{x}^\top\mathbf{M}\mathbf{x} = \|\mathbf{x}\|_M^2$.

Changing the metric in the individuals space

Due to the isometry property, we deduce immediately :

PCA with / without metric

$\mathbf{v}$ max. projected inertia for original data $\mathbf{x}_1, \ldots, \mathbf{x}_n$ with metric $\|.\|_M$

$\Leftrightarrow$

$\mathbf{Rv}$ max. proj. inertia for transformed data $\mathbf{Rx}_1, \ldots, \mathbf{Rx}_n$ with $\|.\|$

$\Leftrightarrow$

$\mathbf{Rv}$ is an eigenvector of $\frac{1}{n}\sum_{i=1}^{n}(\mathbf{Rx}_i)(\mathbf{Rx}_i)^\top = \mathbf{R}\left(\frac{1}{n}\mathbf{X}^\top\mathbf{X}\right)\mathbf{R}^\top$

$\Leftrightarrow$

$\mathbf{v}$ is an eigenvector of $\left(\frac{1}{n}\mathbf{X}^\top\mathbf{X}\right)\mathbf{M} = \Gamma\mathbf{M}$

Changing the metric in the individuals space

Recall that the data are assumed to be centered.

Example. Standardize (centered) data.

$$\mathbf{M} = \mathrm{diag}\left(\frac{1}{\hat{\sigma}_1^2}, \ldots, \frac{1}{\hat{\sigma}_p^2}\right)$$

Then we can choose $\mathbf{R} = \mathrm{diag}\left(\frac{1}{\hat{\sigma}_1}, \ldots, \frac{1}{\hat{\sigma}_p}\right)$. Thus doing PCA with the metric $\mathbf{M}$ is equivalent to doing usual PCA on the standardized data.

Changing the weights in the variable space

In the standard formulation, each individual $\mathbf{x}_1, \ldots, \mathbf{x}_n$ has weight $\frac{1}{n}$.

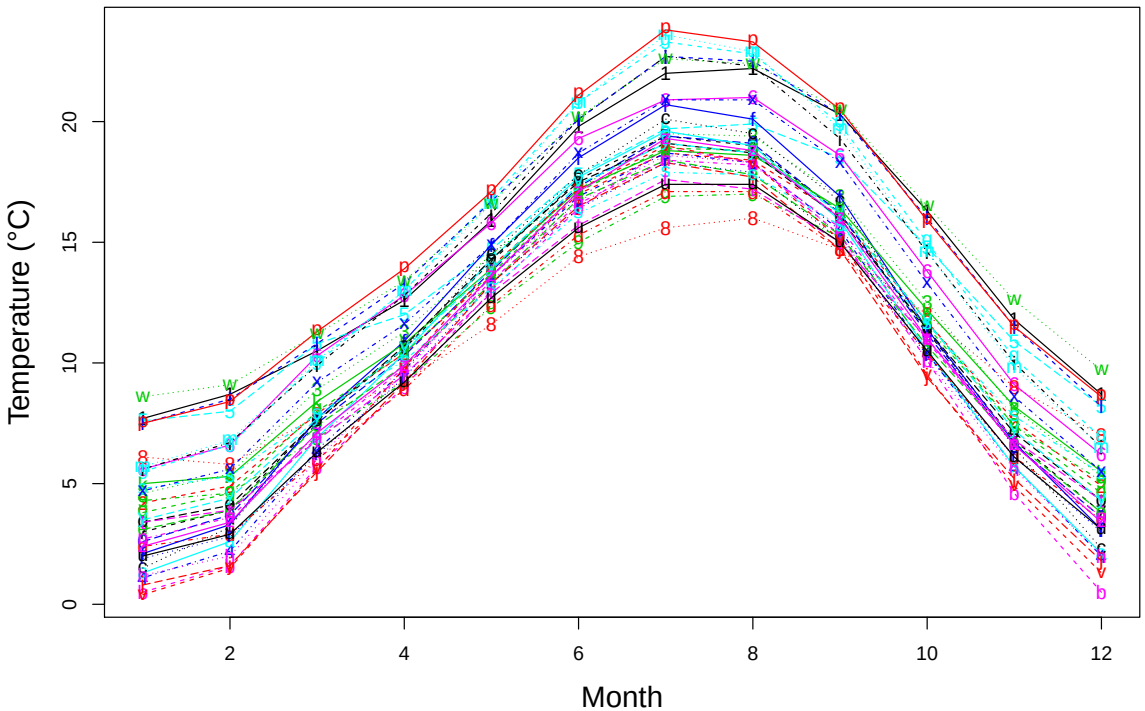
Obviously, one can use positive weights $\omega_1, \ldots, \omega_n$ that sum to one. It can be useful if some individuals have more importance.

This can be viewed as an isometric transformation in the space $\mathbb{R}^n$ by the diagonal matrix containing the square roots of ω_i.
The theory is immediately adapted, by modifying the definitions, e.g. :

$$\mathcal{I} = \sum_{i=1}^{n} \omega_i \|\mathbf{x}_i\|^2, \qquad \Gamma = \sum_{i=1}^{n} \omega_i \mathbf{x}_i \mathbf{x}_i^\top.$$

Results interpretation

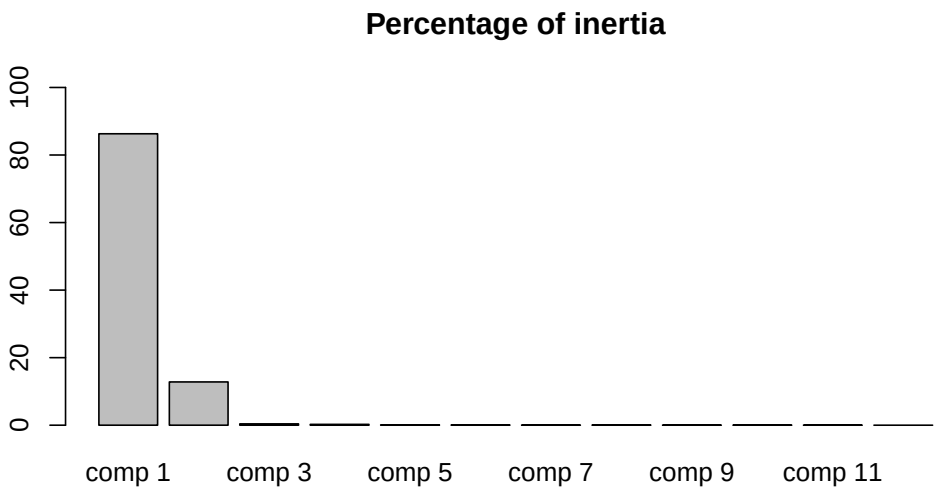
Example on a temperature dataset



Dataset : Temperature at $n = 36$ cities (individuals) for $p = 12$ months (variables).

Dimension reduction

Here the variables are highly correlated, and a strong dimension reduction is expected. The decrease of inertia shows that 2 dimensions explain approx 99% of the variance.

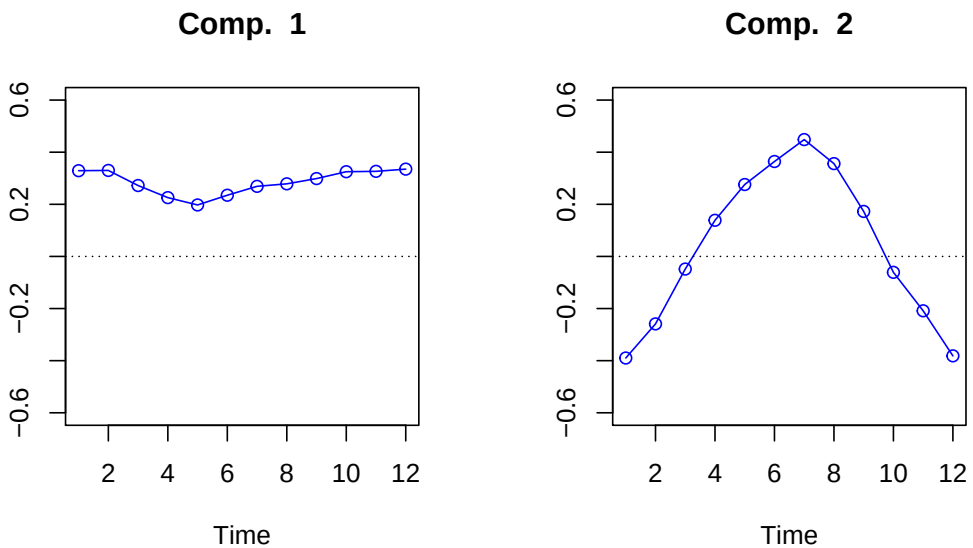


Interpretation of principal components

Remember that $\mathbf{C}^j = \sum_{k=1}^{p} (\mathbf{v}_j)_k \mathbf{x}^k$.

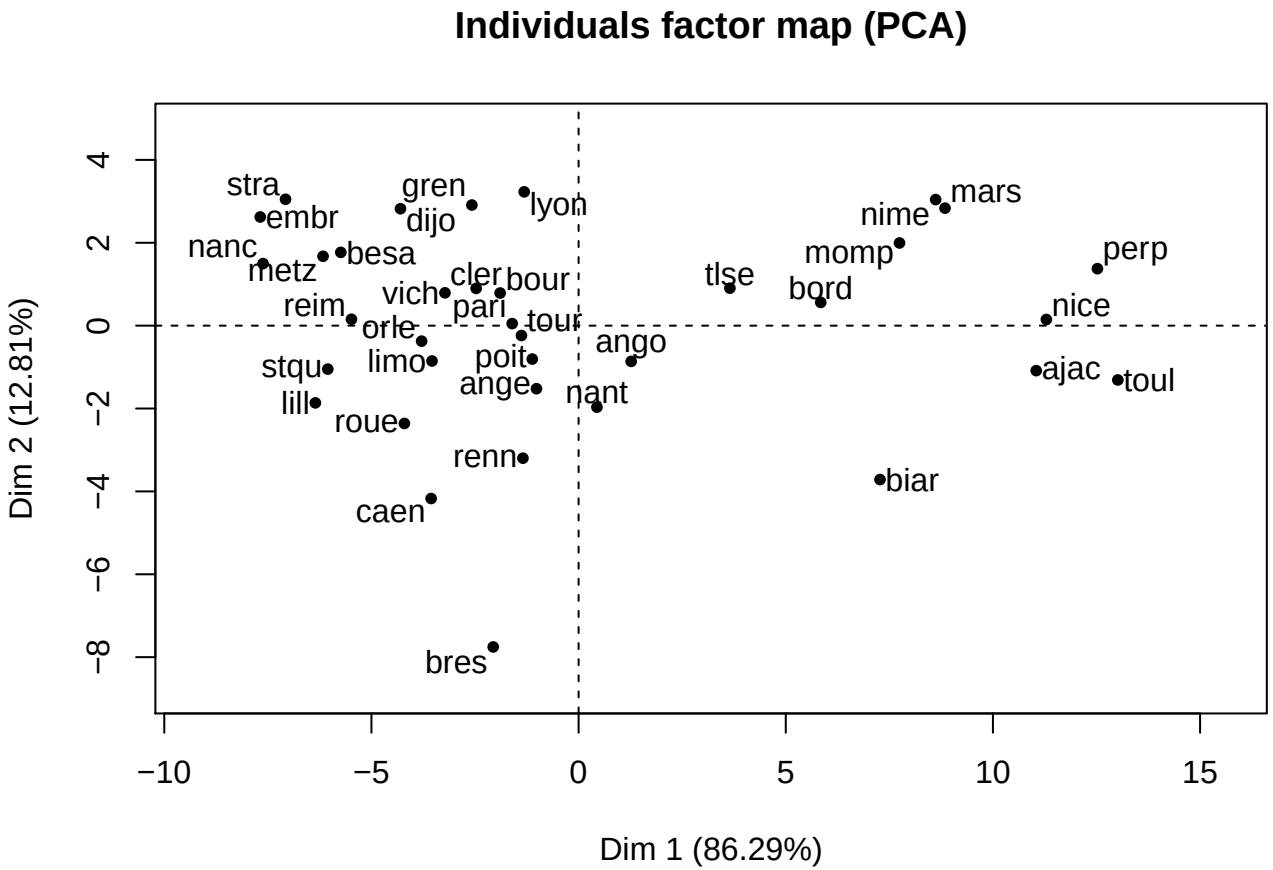
To interpret $\mathbf{C}^j$, look at $\mathbf{v}_j$. Here we can plot them as a function of time.

- $\mathbf{C}^1 \propto (\mathbf{x}^1 + \cdots + \mathbf{x}^{12})$, proportional to the annual temperature
- $\mathbf{C}^2 \propto (\mathbf{x}^5 + \ldots + \mathbf{x}^8) - (\mathbf{x}^1 + \mathbf{x}^2 + \mathbf{x}^{11} + \mathbf{x}^{12})$, contrast summer/winter



Coordinates of the first 2 eigenvectors in $\mathbb{R}^{12}$.

Graphics for individuals



PCA : Projection on the first 2 principal axis.

Graphics for variables

- The principal variables $\mathbf{C}^k$ are orthogonal with variance λ_k. Thus, they define an orthonormal basis $\mathbf{U}^k = \mathbf{C}^k / \sqrt{\lambda_k}$.

- Consider the coordinates $a_{j,k}$ of the original variables in this basis

$$a_{j,k} = \mathrm{cov}(\mathbf{X}^j, \mathbf{U}^k).$$

  We thus have, $\|\mathbf{x}^j\|^2_{\mathbb{R}^n} = \hat{\sigma}_j^2 = \sum_k a_{j,k}^2$.

- The idea is to plot these coordinates for two principal components.

Graphics for variables, case of unit variance

- When the variables have been normalized (unit variance),

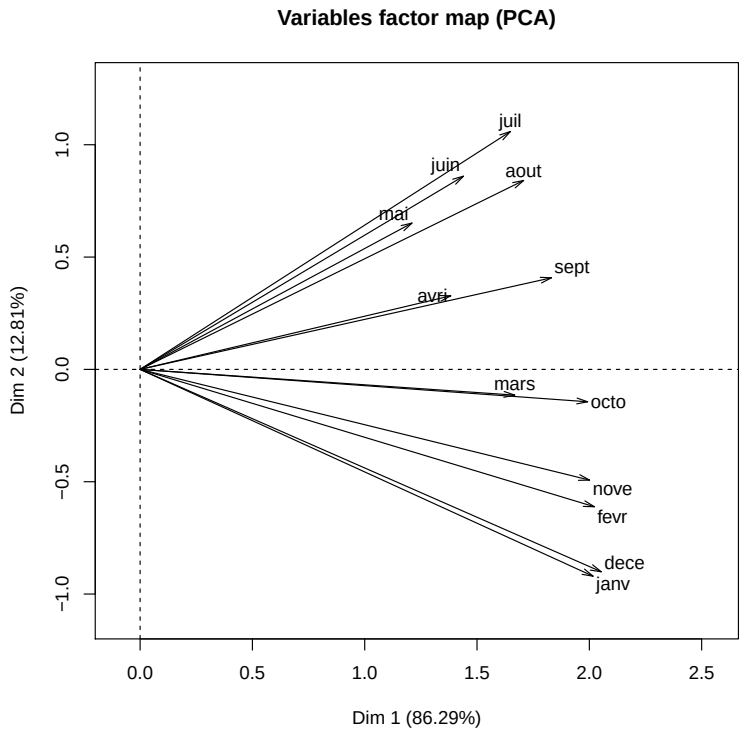
$$a_{j,k} = \mathrm{cor}(\mathbf{X}^j, \mathbf{U}^k) = \cos(\widehat{\mathbf{X}^j, \mathbf{U}^k})$$

  and $\sum_{k=1}^{p} a_{j,k}^2 = 1$.

- Thus the coordinates $(a_{j,k})_k$ belong to a p-dimensional sphere.

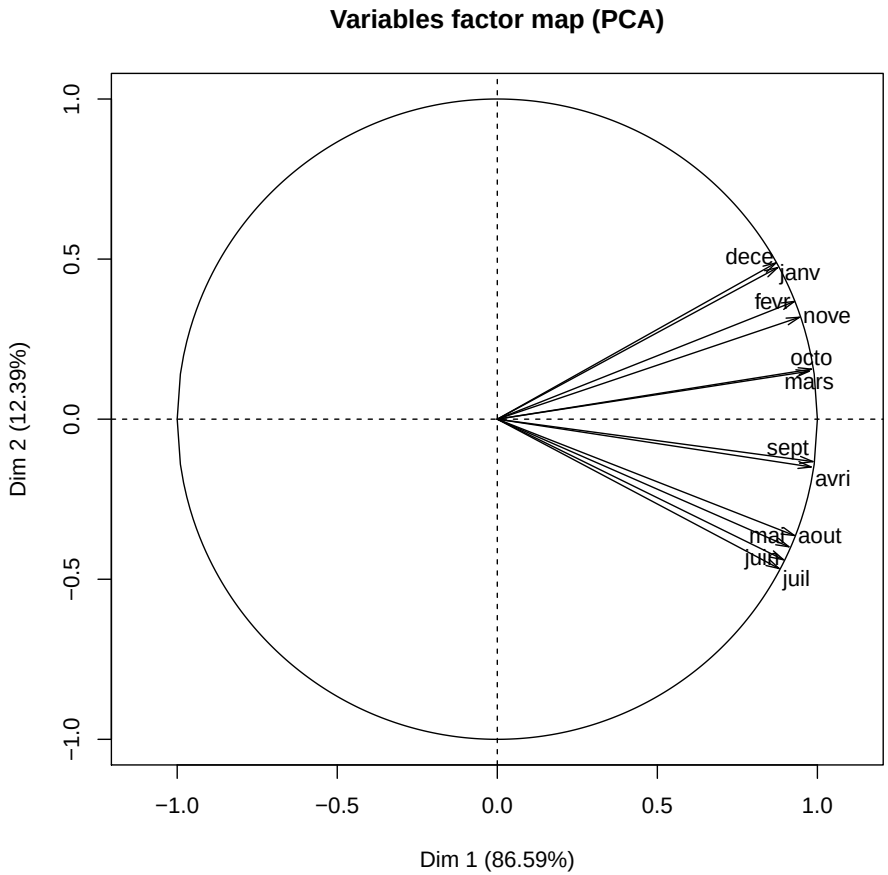
- Further $(a_{j,1}, a_{j,2})$ belongs to the unit disk : $a_{j,1}^2 + a_{j,2}^2 \leq 1$.
  It is closed to the unit circle if $a_{j,3}, \ldots, a_{j,p}$ are nearly zero.
  In that case, $\mathbf{X}^j$ is well-represented by $\mathbf{C}^1, \mathbf{C}^2$.
  This is the circle of correlations for components (1, 2).

Interpretation of principal components



Coordinates of the variables in the orthonormal basis of principal variables. We see again that Axis 1 weigths all months nearly equally, whereas Axis 2 exhibits a contrast summer / winter.

Interpretation of principal components



Circle of correlation (normalized variables). Here all variables are well-represented by the first 2 principal components.

Graphics for variables

- The attentive reader may have remarked that the coordinates k on slide 61 and on slide 57 (plot of eigenvector coordinates) look similar.
- It can be shown that they are indeed proportional, by the factor $\sqrt{\lambda_k}$.
- The graph of variables (circle of correlation for normalized variables) thus gives a complementary 2D visualization.

Conclusion and further readings

- PCA is a dimension reduction technique which finds uncorrelated variables, called principal variables, that are linear combination of the original ones, which approximate the best the data in the mean-square sense.
- PCA = spectral decomposition of the covariance matrix
  - Up to isometric transformations (metric, weights)
- Several graphs can be used to interpret principal components : projection of individuals, circle of correlation (normalized case).
  - Mind that what you visualize is only a projection. Several tools quantify the quality of the representation.

Outline

- Introduction
- Multidimensional Exploration
  - Basic tools
  - Principal components analysis
  - Factorial discriminant analysis
- Unsupervised classification
  - Hierarchical Cluster Analysis
  - k-means

Factorial Discriminant Analysis

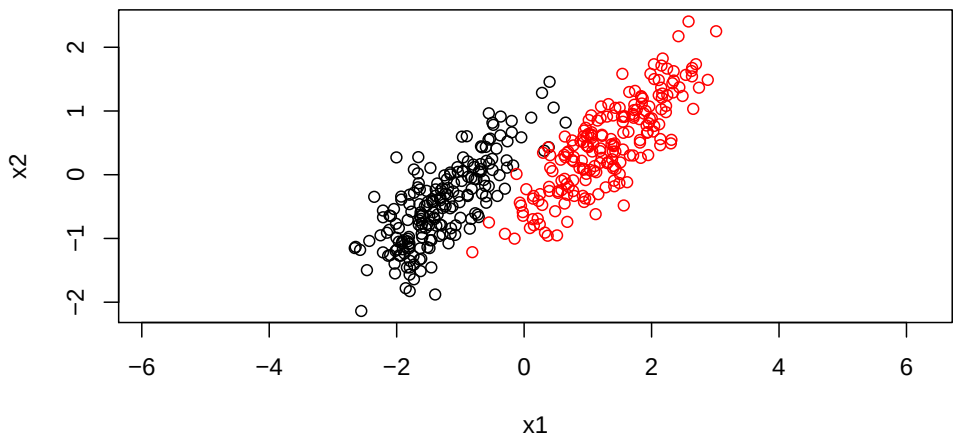


Figure – This is a cloud of points, with two classes, in dimension 2 (higher in general)

Can you find two 1D axis 'suitable' to identify classes ?

FDA, as an exploratory tool

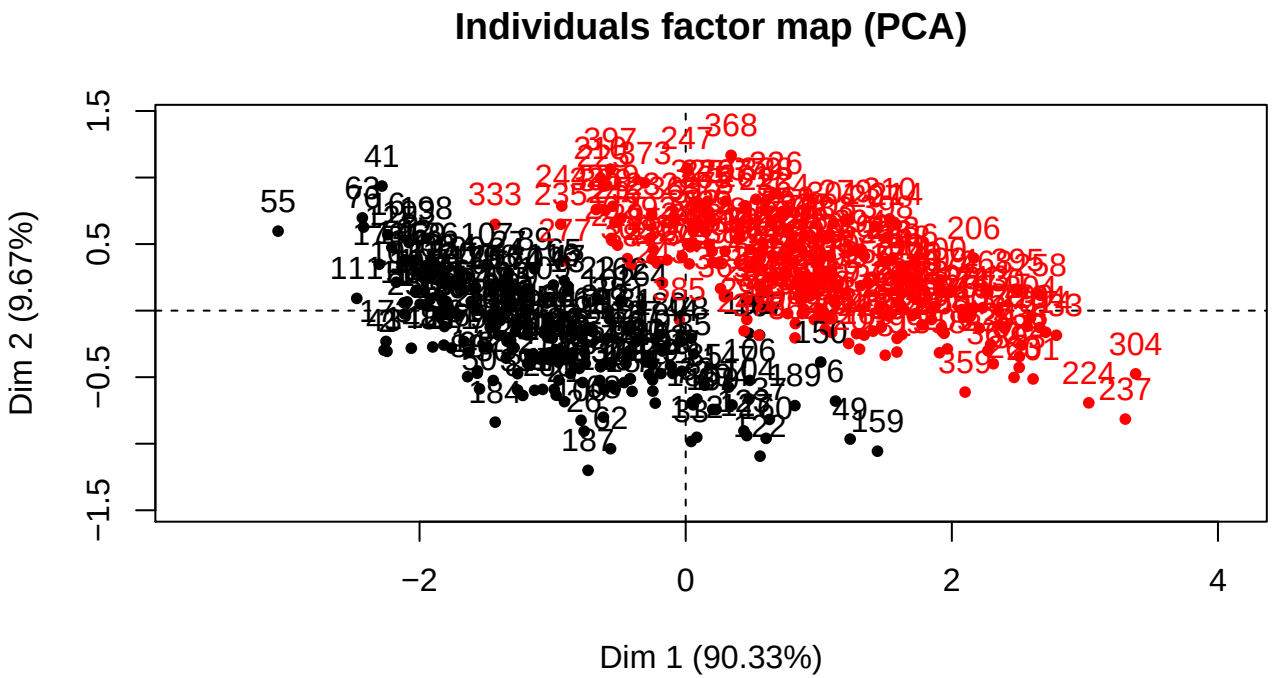


Figure – Result of the PCA analysis. Can we do better ?

FDA, as an exploratory tool

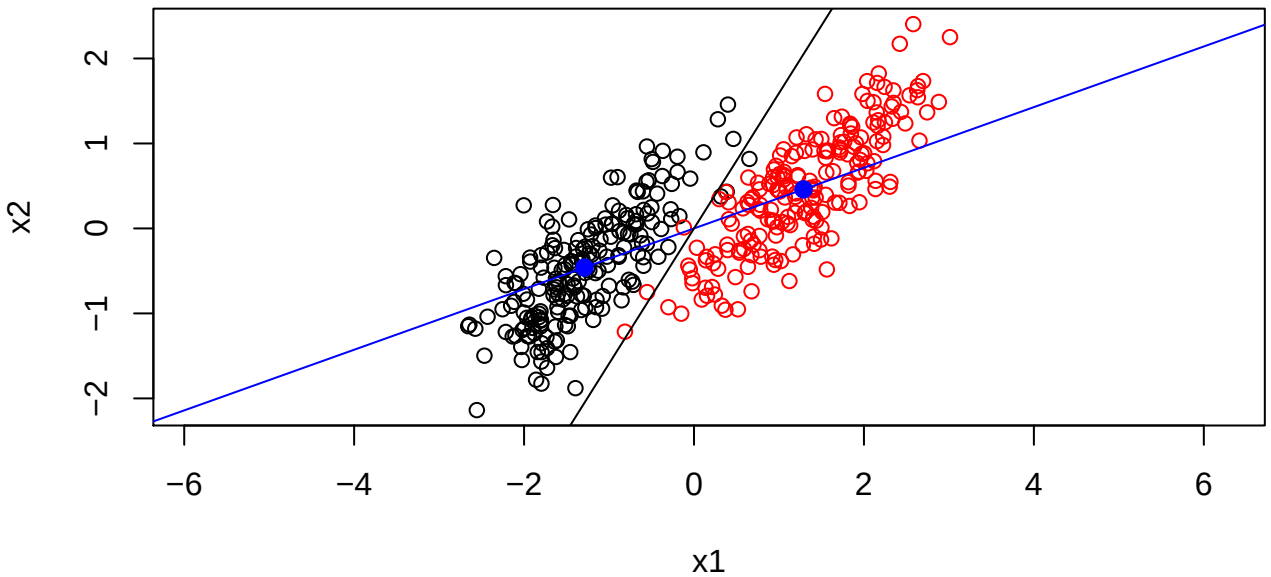


Figure – Result of the FDA analysis, actually PCA for the centroids : two data only ! The two axes are orthogonal... for a specific ('Mahalanobis') metric !

FDA, as an exploratory tool

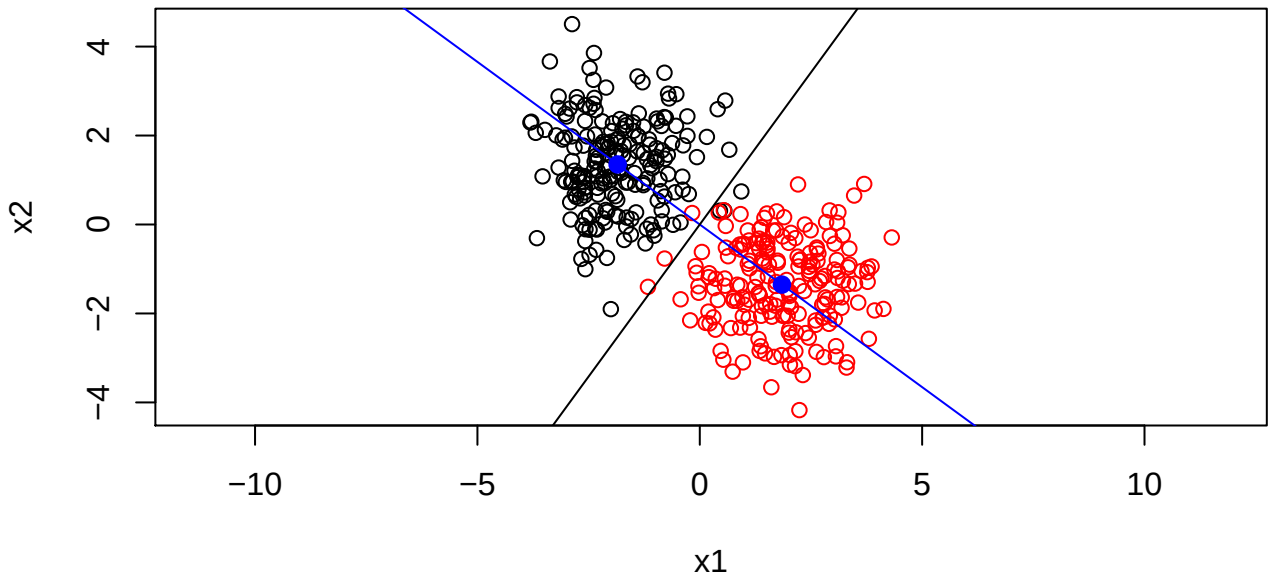


Figure – Result of the FDA analysis : visualization for tranformed data. The two axes are orthogonal for the usual metric.

Mahalanobis metric and 'sphered' data

The Mahalanobis metric is such that the covariance matrix is identity. This is equivalent to (matricially) reduce or 'sphere' the data :

$$\boldsymbol{x} \mapsto \mathrm{Cov}^{-1/2}\boldsymbol{x}$$

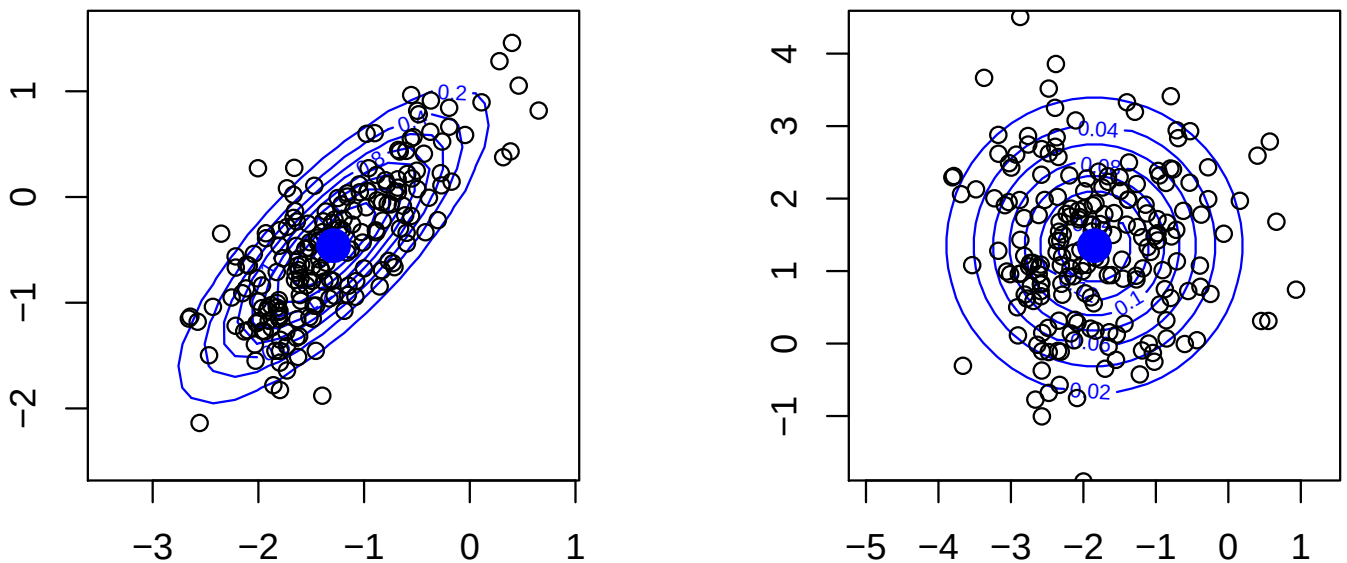
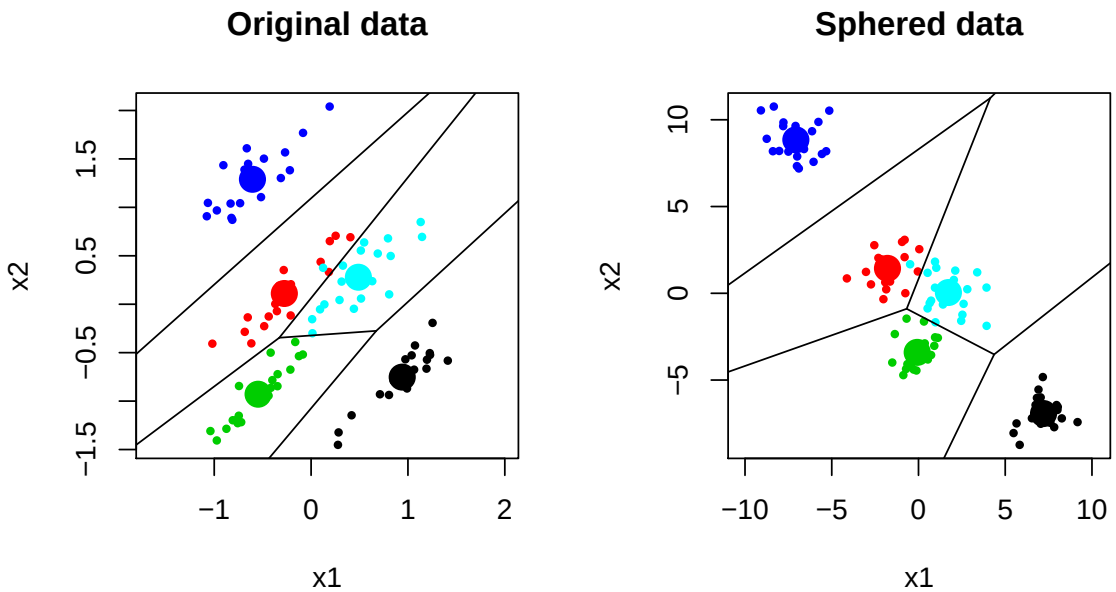


Figure – Left : Original data. Right : Reduced data. Level sets are for the multinormal distribution with corresponding covariance matrix.

Remark : FDA for supervised classification

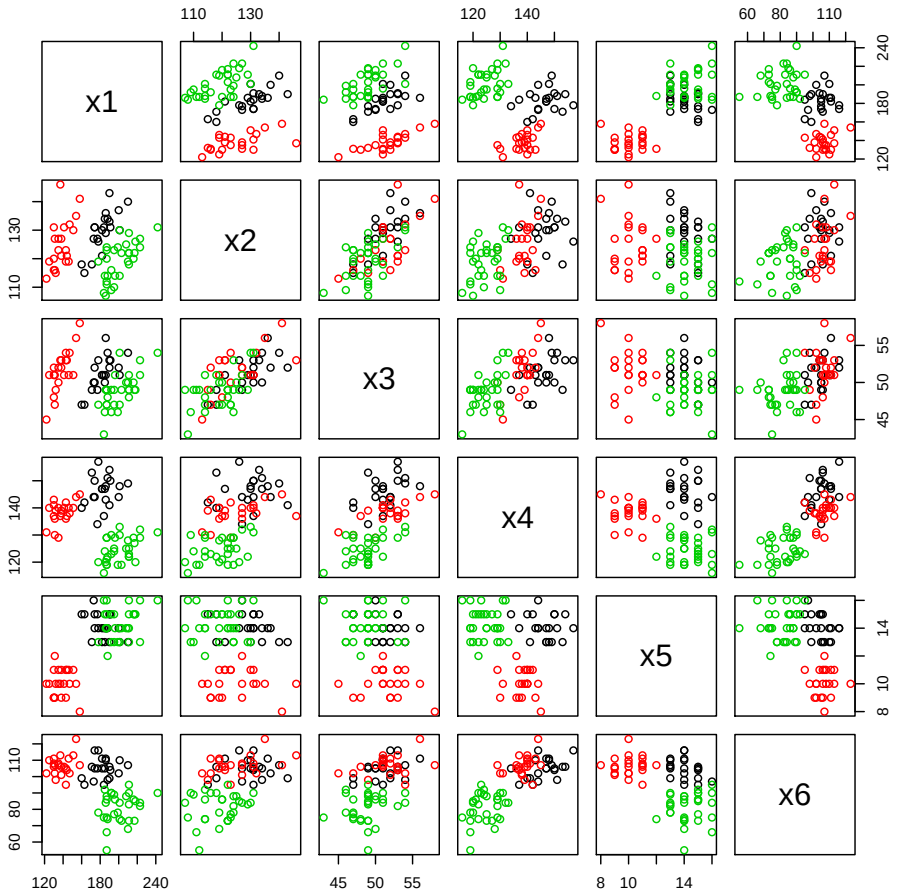
- FDA can be used for classification. When classes have equal sizes, one chooses the class corresponding to the closest centroid. This gives linear frontiers.



- Test other configurations with the applet : https://roustant.shinyapps.io/lda-app/

A six dimensional example

We consider the Lubitsch data for insects. There are 74 data, 6 variables, and 3 classes.



A six dimensional example

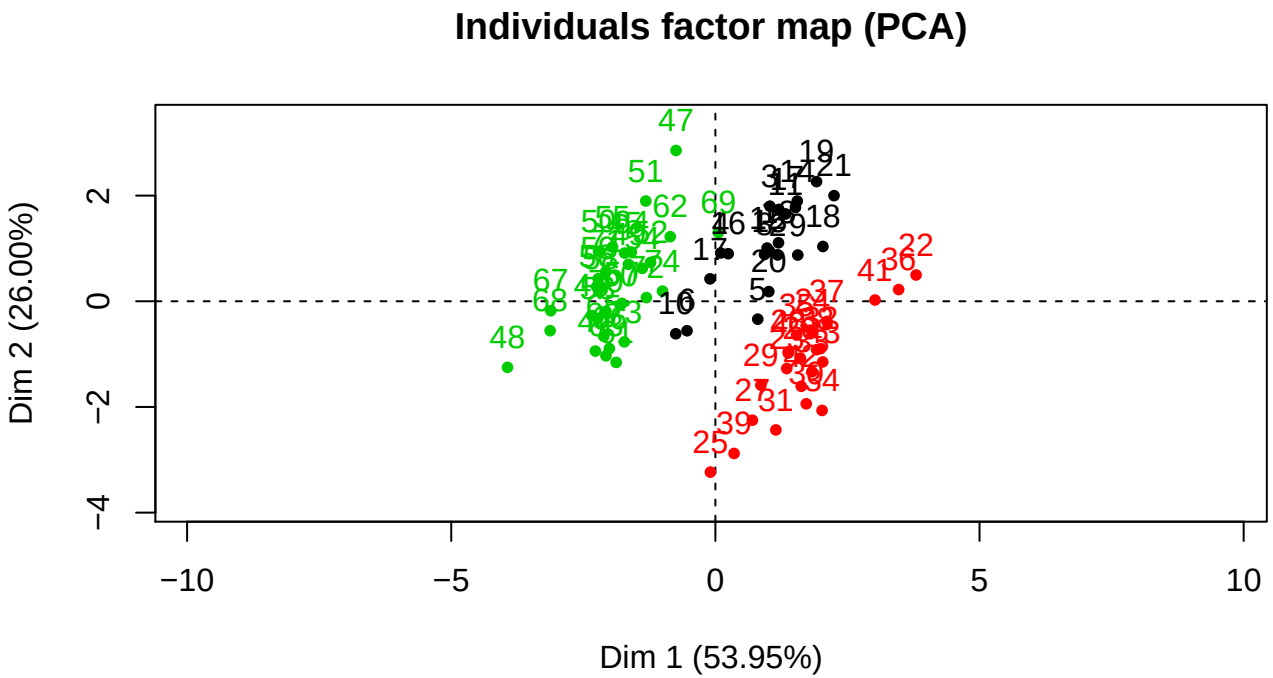


Figure – Insect dataset. Result of the PCA analysis.

A six dimensional example

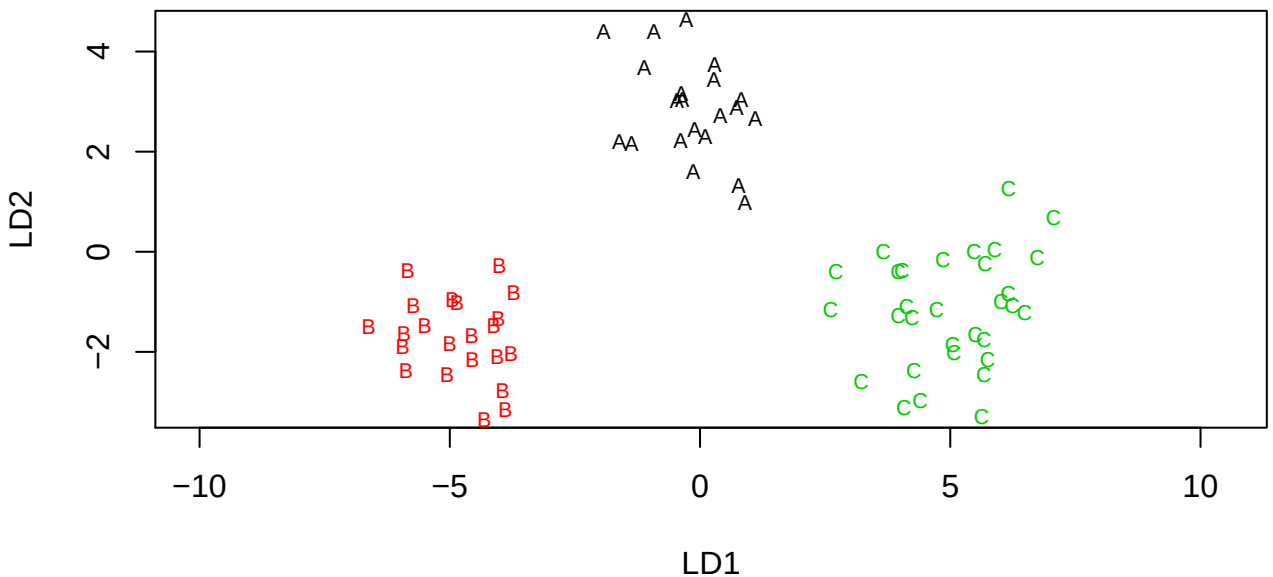
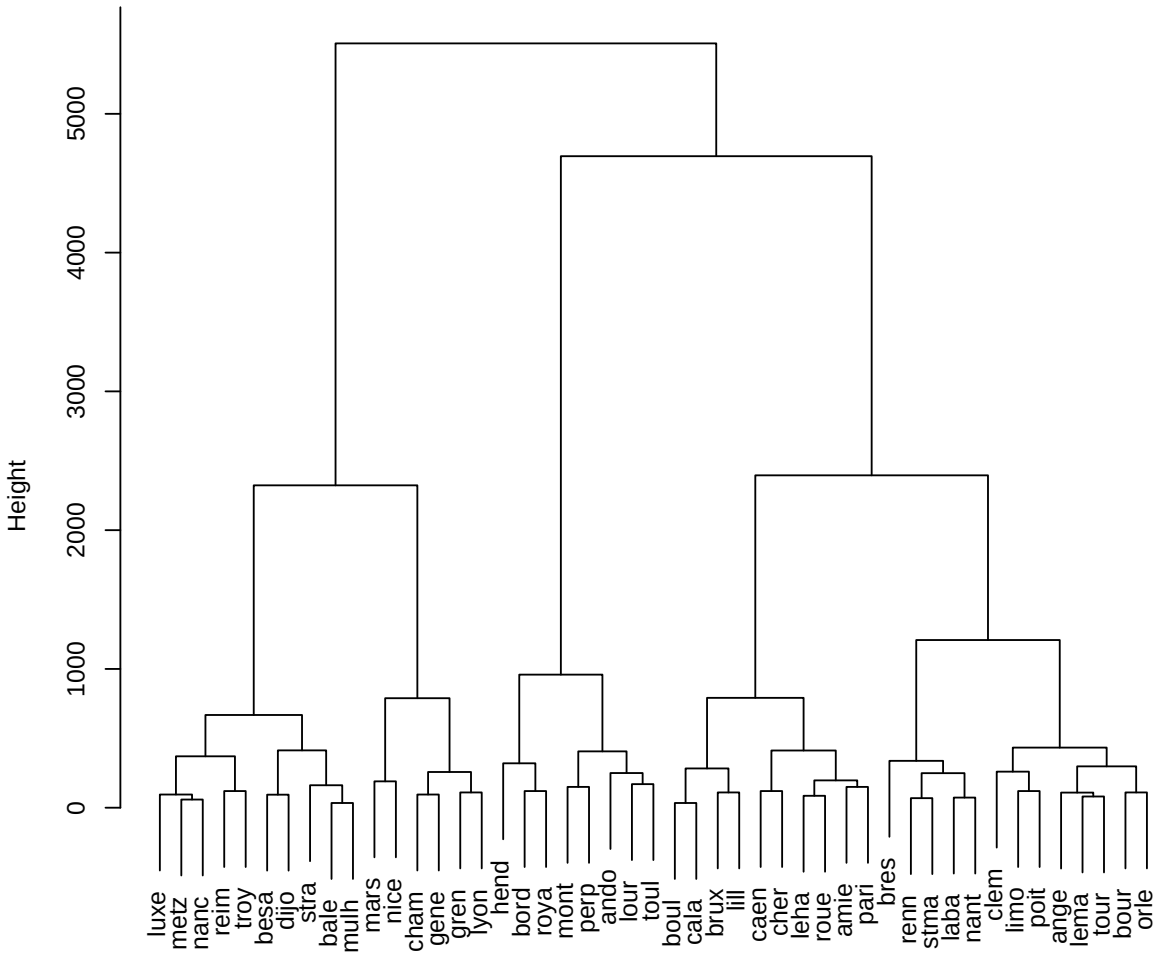


Figure – Insect dataset. Result of the FDA analysis.

Outline

- Introduction
- Multidimensional Exploration
  - Basic tools
  - Principal components analysis
  - Factorial discriminant analysis
- Unsupervised classification
  - Hierarchical Cluster Analysis
  - k-means

Cities : dendrogram example



Algorithm for Hierarchical Cluster Analysis

- Initialization : singletons, distances between pairs of singletons
- Iterate until aggregation in a single class :
  1. gather the two closest classes within the meaning of the chosen "distance" between groups
  2. update the distance table by replacing the two classes with the new one and by calculating its "distance" with the other classes.

Distance between groups

Linkage functions

The distances between two sets A and B, can be computed in various way. Here are some examples, denoting by g_A, g_B the centroids of A, B :

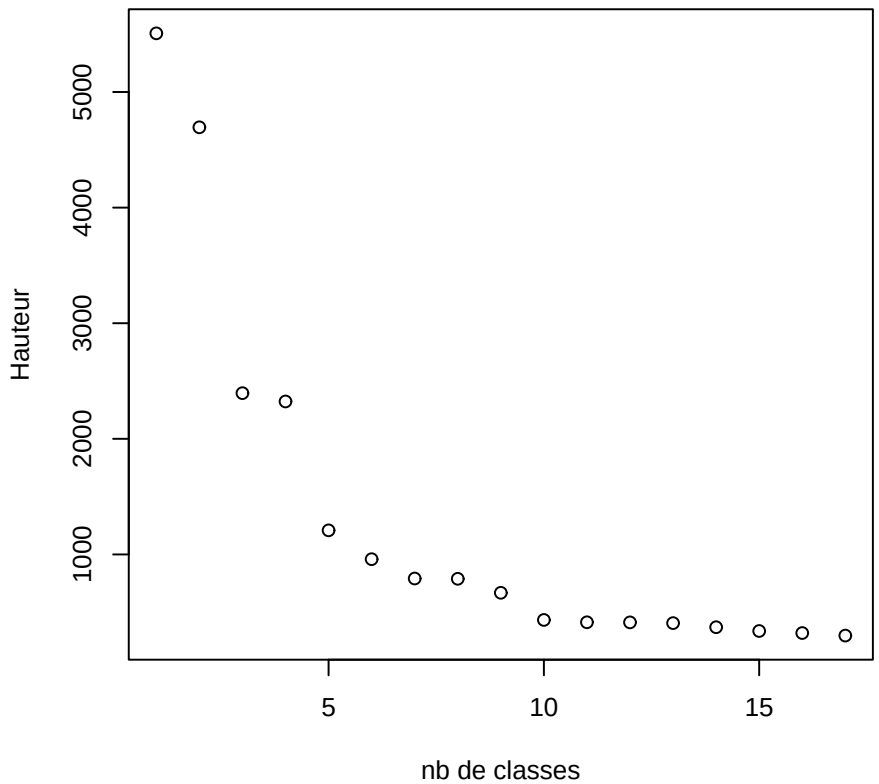
linkage name	linkage function $d(A,B)$
single	$\min_{i\in A, j\in B} d_{ij}$
complete	$\sup_{i\in A, j\in B} d_{ij}$
average	$\frac{1}{\lvert A\rvert.\lvert B\rvert}\sum_{i\in A, j\in B} d_{ij}$
centroid	$d(g_A, g_B)$
Ward	$\frac{\lvert A\rvert.\lvert B\rvert}{\lvert A\rvert+\lvert B\rvert} d(g_A, g_B)^2$

Ward's criterion is equal to the **between-class inertia**,

$$|A|d(g_A, g)^2 + |B|d(g_B, g)^2$$

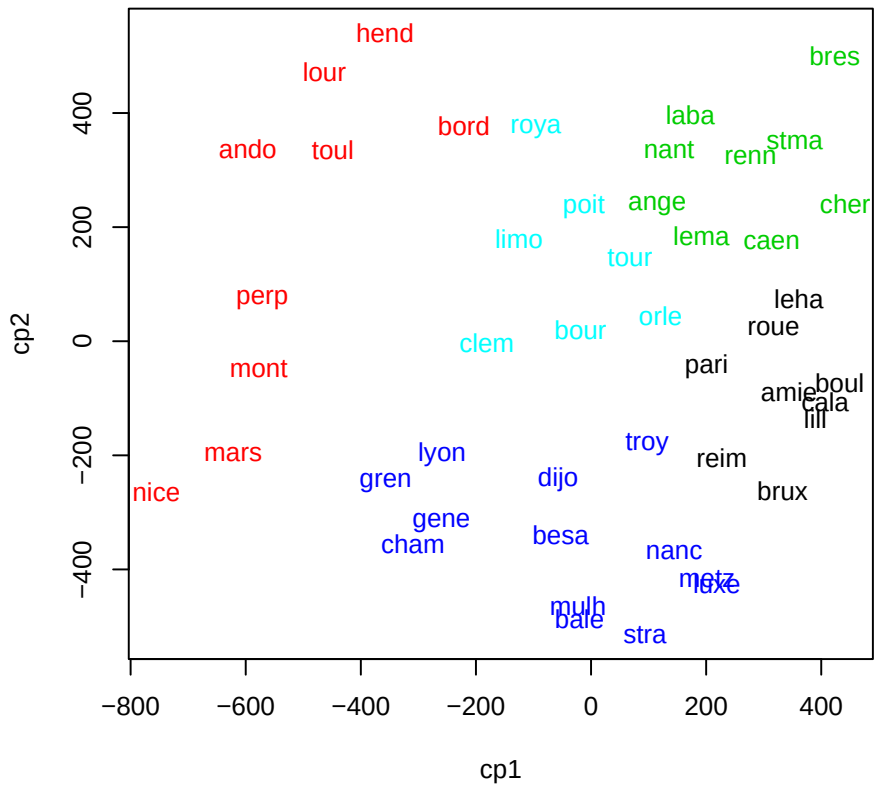
where g is the centroid of the set $\{A, B\}$ obtained after merging A, B.

Number of classes



Cities : Decrease of the between-class inertia

Representation of the classes



Outline

- Introduction
- Multidimensional Exploration
  - Basic tools
  - Principal components analysis
  - Factorial discriminant analysis
- Unsupervised classification
  - Hierarchical Cluster Analysis
  - k-means

Principle of moving center

- dynamic reallocation of individuals to classes
- The number of classes k is fixed *a priori*

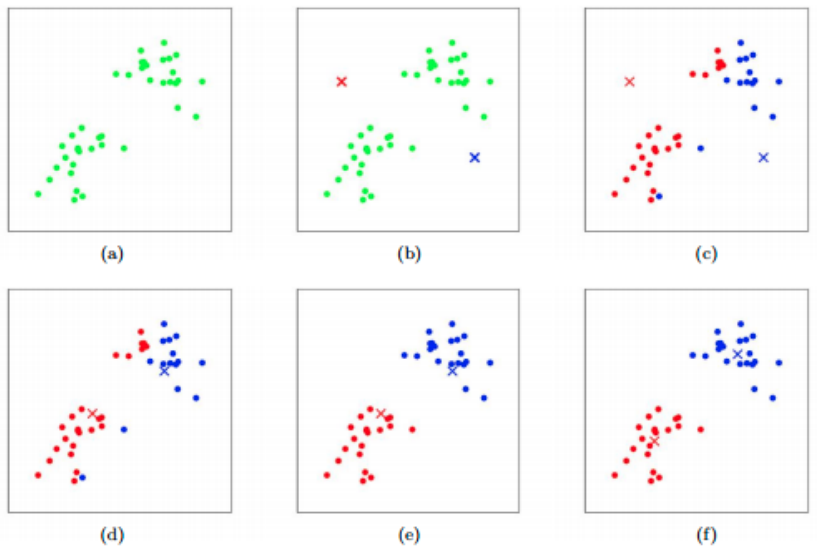
Principle of Forgy algorithm

- Initialisation Draw - or randomly select - k points in the space of individuals called centers.
- Iterate until stabilization of the between-class inertia
  1. Allocate each individual to a center, i.e. to a class
  2. Compute the centroid of each class, it becomes the new center.

Variants

- k-means Algorithme : the center of the classes, here centroids, are recalculated at each allocation of a point to a class : more efficient algorithm.
- Partitionning Around Medoïds (PAM) or k-medoïds.
  - Medoïd : individual minimizing the mean of the distances to the other points (more robust algorithm).

Illustration of k-means



Conclusion

- We have seen several tools to visualize data.
- Concerning unsupervised classification or clustering, many other algorithms are available.
- The choice among all possible algorithms is often difficult.
- The next steps will be devoted to supervised classification where an estimation of the prediction error is possible.
- To go further... read the textbooks on wikistat.fr and execute the notebooks on the Github wikistat.