



# Algorithms. Matching

Part II. Preference model.

B9 - Algorithms Matching

M-ALG-102

...

## └ Introduction

# Introduction

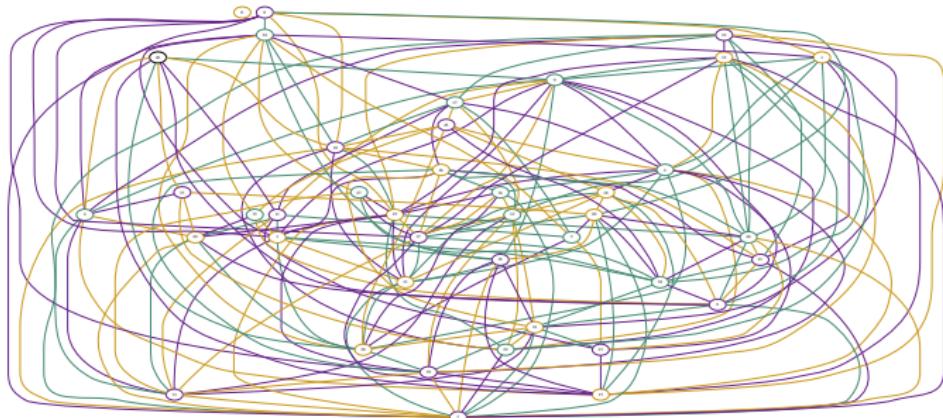


Figure – Graph

# Introduction

<https://github.com/nlehir/ALGO2>

We will need (the installation of these packages should work with pip) :

- ▶ networkx
- ▶ matplotlib
- ▶ pandas
- ▶ sklearn
- ▶ optionnally : ipdb

## Day 2

### Compatibility graphs

- Simple geometrical data

- Complex data

### Probability distributions

- Reminders on probabilities

- Analyzing a distribution

- Optimization and Maximum Likelihood

- Gradients

### Multivariate analysis and clustering

- Correlation

- Dimension reduction

- Correlation and causality

- Scatter matrix

### Clustering

- Kmeans clustering

- Similarities

...

## Compatibility graphs

- ▶ Yesterday we processed graphs describing **relationship between data**
- ▶ If two nodes were related, they were linked by an edge in the graph.

...

# Compatibility graphs

- ▶ Yesterday we processed graphs describing **relationship between data**
- ▶ If two nodes were related, they were linked by an edge in the graph.
- ▶ Today we are interested in building such graphs directly from the data, we call them **compatibility graphs**.

...

└ Compatibility graphs

## Compatibility graphs

We are interested in building **compatibility graphs**.

Given two nodes in a graph, should there be an edge between them ?

...

## Compatibility graphs

We are interested in building **compatibility graphs**.

Given two nodes in a graph, should there be an edge between them ?

**Note :** it is not the same problem as the matching problem. In the matching problem, the edges are already defined.

...

└ Compatibility graphs

## Compatibility graphs

We are interested in building **compatibility graphs**.

Given two nodes in a graph, should there be an edge between them ?

**Note :** it is not the same problem as the matching problem. In the matching problem, the edges are already defined.

However, once the edges are built, we can apply a matching to it.

...

└ Compatibility graphs

## Example applications

- ▶ Social networks management
- ▶ Recommendations

...

└ Compatibility graphs

## Building compatibility graphs

- ▶ We will build graphs first from simple data
- ▶ Then from more complex data.

...

└ Compatibility graphs

  └ Simple geometrical data

## Building a graph from simple data

- ▶ We will first build a graph from simple data in the 2D space.

...

└ Compatibility graphs

└ Simple geometrical data

## Euclidian distance and compatibility

Consider the following data :

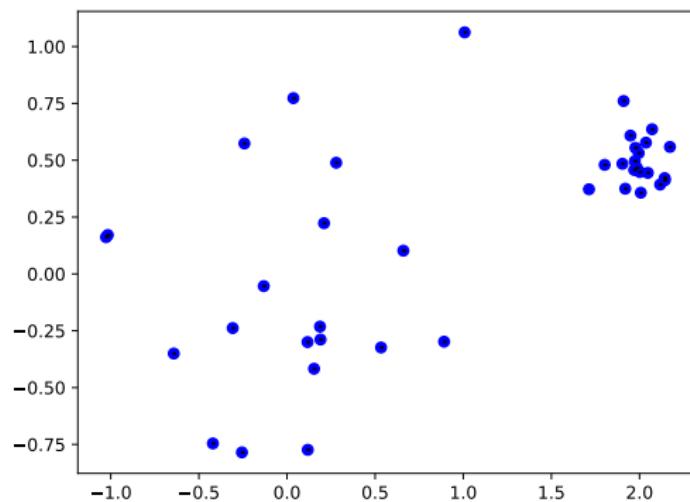
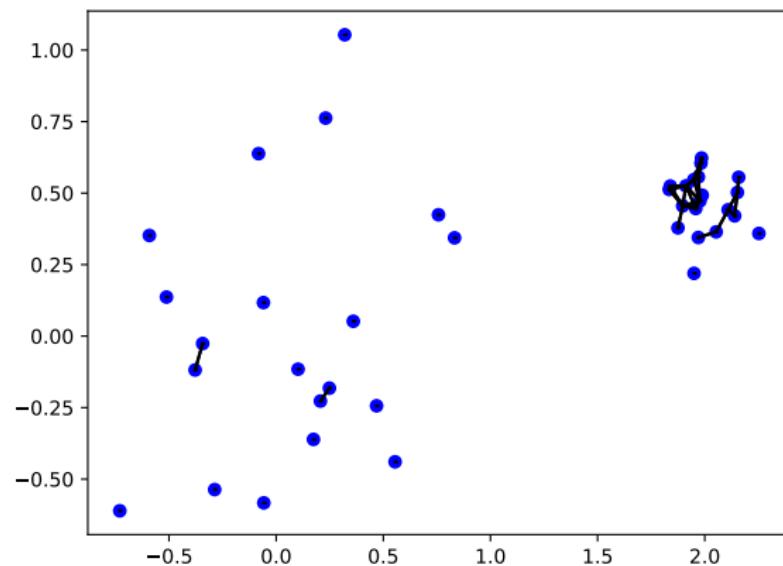


Figure – Data : we would like to define **edge** between some of them

Is this set of edges a good solution?



## Figure – Some definition of edges

...

└ Compatibility graphs

└ Simple geometrical data

Is this set of edges a good solution ?

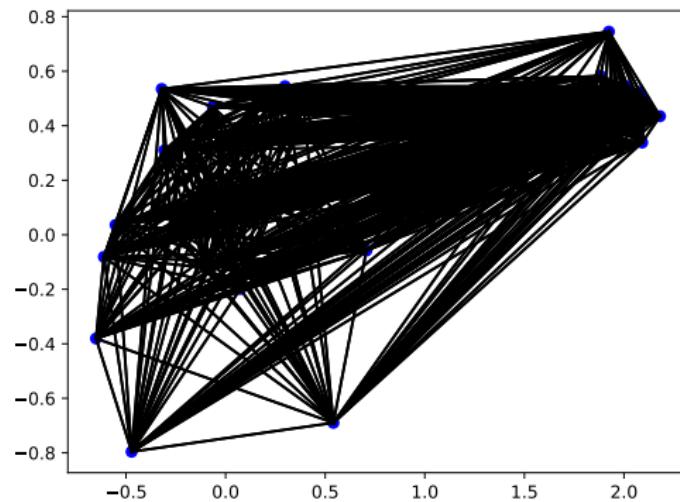


Figure – Some definition of edges

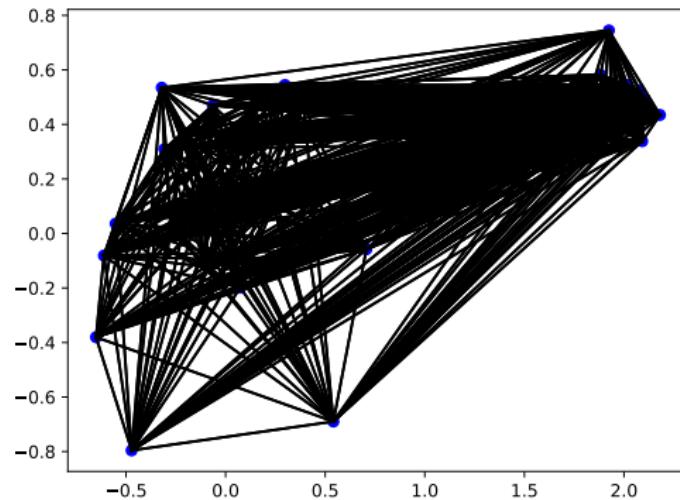
...

└ Compatibility graphs

└ Simple geometrical data

## Euclidian distance and compatibility

Here, all we know about the data is their **euclidian distance** :



...

└ Compatibility graphs

└ Simple geometrical data

This one looks ok

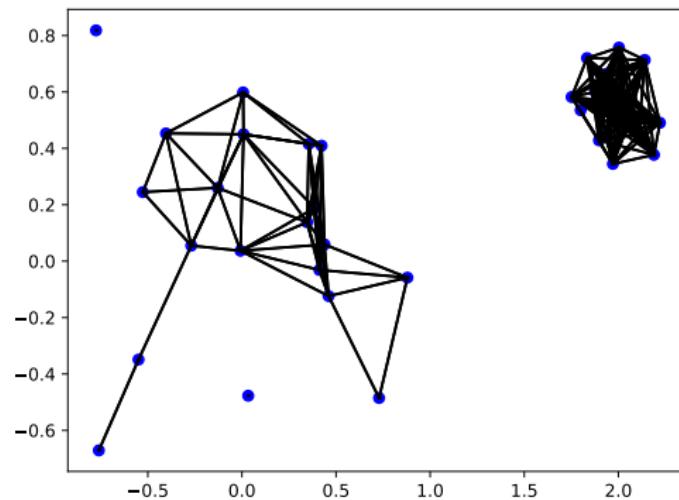


Figure – A proposition of edges

...

└ Compatibility graphs

  └ Simple geometrical data

# Backboard

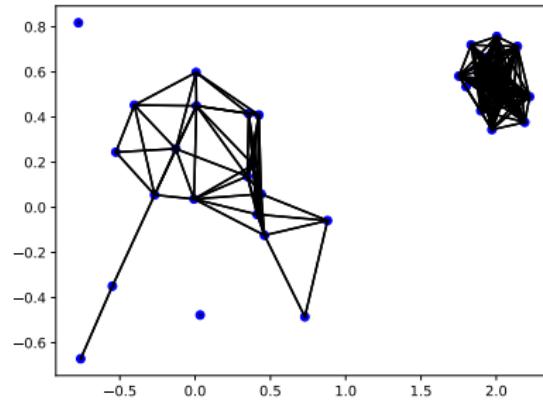
- ▶ Euclidian distance and threshold.

...

└ Compatibility graphs

└ Simple geometrical data

**Exercice 1:** Setting a threshold cd **compatibility\_simple** and set the threshold used in **build\_graph.py** to draw relevant edges between the nodes. Feel free to use another dataset !



...

└ Compatibility graphs

└ Simple geometrical data

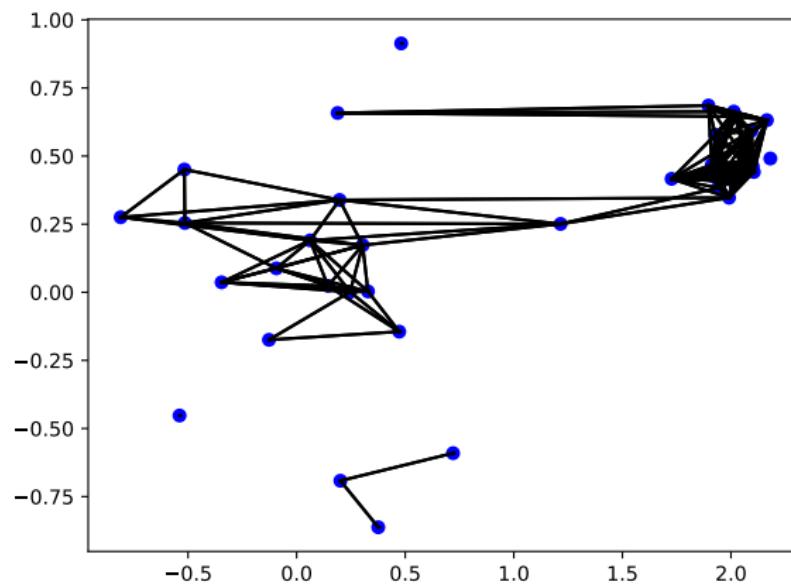
## Exercice 2 : Changing the distance

- ▶ Assess the impact of changing the distance used. Possible choices :
  - ▶  $L_1$  distance (Manhattan)
  - ▶  $\| \cdot \|_\infty$  distance (backboard)
  - ▶ custom distance
- ▶ use `build_graph_other_distance.py` and edit the distances used at the end of the file.
- ▶ Try several values for the threshold.

...

## └ Compatibility graphs

## └ Simple geometrical data



...

└ Compatibility graphs

└ Simple geometrical data

## General notion of a distance

- ▶ Let us generalize what we experimentally studied.

...

└ Compatibility graphs

└ Simple geometrical data

## Examples of distances

$x = (x_1, \dots, x_p)$  and  $y = (y_1, \dots, y_p)$  are  $p$ -dimensional **vectors**.

...

- └ Compatibility graphs

- └ Simple geometrical data

## Examples of distances

$x = (x_1, \dots, x_p)$  and  $y = (y_1, \dots, y_p)$  are  $p$ -dimensional **vectors**.

- ▶ L2 :  $\|x - y\|_2^2 = \sum_{k=1}^p (x_k - y_k)^2$  (Euclidian distance)

...

- └ Compatibility graphs

- └ Simple geometrical data

## Examples of distances

$x = (x_1, \dots, x_p)$  and  $y = (y_1, \dots, y_p)$  are  $p$ -dimensional **vectors**.

- ▶ L2 :  $\|x - y\|_2^2 = \sum_{k=1}^p (x_k - y_k)^2$  (Euclidian distance)
- ▶ L1 :  $\|x - y\|_1 = \sum_{k=1}^p |x_k - y_k|$  (Manhattan distance)

...

└ Compatibility graphs

└ Simple geometrical data

## Examples of distances

$x = (x_1, \dots, x_p)$  and  $y = (y_1, \dots, y_p)$  are  $p$ -dimensional vectors.

- ▶ L2 :  $\|x - y\|_2^2 = \sum_{k=1}^p (x_k - y_k)^2$  (Euclidian distance)
- ▶ L1 :  $\|x - y\|_1 = \sum_{k=1}^p |x_k - y_k|$  (Manhattan distance)
- ▶ weighted L1 :  $\sum_{k=1}^p w_k |x_k - y_k|$

...

└ Compatibility graphs

└ Simple geometrical data

## Hamming distance

- ▶  $\#\{x_i \neq y_i\}$  (Hamming distance)

...

└ Compatibility graphs

└ Simple geometrical data

## Hamming distance and edit distance

- ▶  $\#\{x_i \neq y_i\}$  (Hamming distance)
- ▶ linked to **edit distance** : used to quantify how dissimilar two strings are by counting the number of operations needed to transform one into the other (several variants exist)

...

└ Compatibility graphs

└ Simple geometrical data

## General definition of a distance

A **distance** on a set  $E$  is an application  $d : E \times E \rightarrow \mathbb{R}_+$  that must :

...

- └ Compatibility graphs

- └ Simple geometrical data

## General definition of a distance

A **distance** on a set  $E$  is an application  $d : E \times E \rightarrow \mathbb{R}_+$  that must :

- ▶ be **symmetrical** :  $\forall x, y, d(x, y) = d(y, x)$

...

└ Compatibility graphs

└ Simple geometrical data

## General definition of a distance

A **distance** on a set  $E$  is an application  $d : E \times E \rightarrow \mathbb{R}_+$  that must :

- ▶ be **symmetrical** :  $\forall x, y, d(x, y) = d(y, x)$
- ▶ **separate the values** :  $\forall x, y, d(x, y) = 0 \Leftrightarrow x = y$

...

└ Compatibility graphs

└ Simple geometrical data

## General definition of a distance

A **distance** on a set  $E$  is an application  $d : E \times E \rightarrow \mathbb{R}_+$  that must :

- ▶ be **symmetrical** :  $\forall x, y, d(x, y) = d(y, x)$
- ▶ **separate the values** :  $\forall x, y, d(x, y) = 0 \Leftrightarrow x = y$
- ▶ respect the **triangular inequality**

$$\forall x, y, z, d(x, y) \leq d(x, z) + d(y, z)$$

...

- └ Compatibility graphs

- └ Complex data

## Building compatibility graphs for more complex data

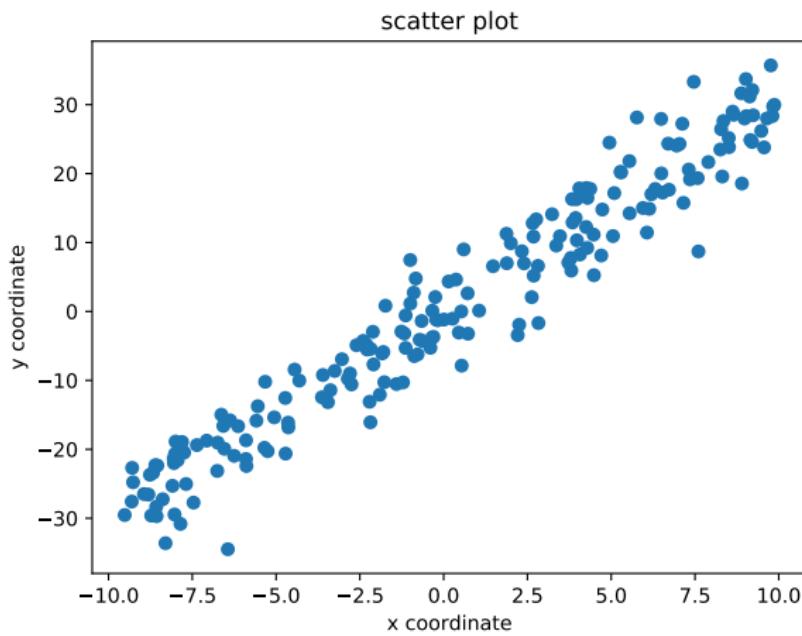
- ▶ We will do the same with more complex data :
  - ▶ possibly more dimensions
  - ▶ possibility categorical variables

...

- Probability distributions

- Reminders on probabilities

## Random variables

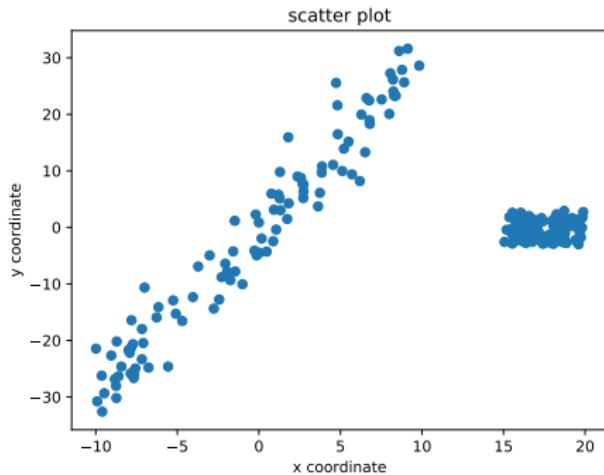


...

- Probability distributions

- Reminders on probabilities

## Random variables



We want to analyse how the data are **distributed**. For instance the  $x$  coordinate, the  $y$  coordinate.

...

└ Probability distributions

  └ Reminders on probabilities

## Random variables

- ▶ A **random variable** is a quantity that can take several values

...

- Probability distributions

- Reminders on probabilities

## Random variables

- ▶ A **random variable** is a quantity that can take several values
- ▶ For instance :
  - ▶ the result of a dice throw



Figure – Dice

...

└ Probability distributions

  └ Reminders on probabilities

## Random variables

- ▶ A **random variable** is a quantity that can take several values
- ▶ For instance :
  - ▶ the result of a dice throw
  - ▶ waiting time with RATP



Figure – Some metro station

...

- Probability distributions

- Reminders on probabilities

## Random variables

- ▶ A **random variable** is a quantity that can take several values
- ▶ For instance :
  - ▶ the result of a dice throw
  - ▶ waiting time with RATP
  - ▶ weather



Figure – Weather in November

...

└ Probability distributions

  └ Reminders on probabilities

## Random variables

- ▶ A **random variable** is a quantity that can take several values
- ▶ For instance :
  - ▶ the result of a dice throw
  - ▶ waiting time with RATP
  - ▶ weather
  - ▶ number of cars taking the périphérique at the same time

...

└ Probability distributions

  └ Reminders on probabilities

## Random variables

What are the differences between these random variables ?

...

└ Probability distributions

  └ Reminders on probabilities

## Random variables

What are the differences between these random variables ?

- ▶ Some are **continuous**, others **discrete**
- ▶ **continuous** :

...

└ Probability distributions

  └ Reminders on probabilities

## Random variables

What are the differences between these random variables ?

- ▶ Some are **continuous**, others **discrete**
- ▶ **continuous** : weather, RATP

...

- Probability distributions

- Reminders on probabilities

## Random variables

What are the differences between these random variables ?

- ▶ Some are **continuous**, others **discrete**
- ▶ **continuous** : weather, RATP
- ▶ **discrete** : dice (6 possibilities), number of cars (> 10000)

...

└ Probability distributions

  └ Reminders on probabilities

## Probability distributions

- ▶ A random variable is linked to a **probability distribution**.

...

└ Probability distributions

  └ Reminders on probabilities

## Probability distributions

- ▶ A random variable is linked to a **probability distribution**.
- ▶ It quantifies the probability of observing one outcome.

...

- Probability distributions

- Reminders on probabilities

## Probability distributions

- ▶ A random variable is linked to a **probability distribution**, which is a function  $P$
- ▶ It quantifies the probability of observing one outcome.
- ▶ For a discrete variable : each possible outcome is associated with a number between 0 and 1

...

- Probability distributions

- Reminders on probabilities

## Probability distributions

- ▶ For a dice game, the possible outcomes are in the set  $\{1, 2, 3, 4, 5, 6\}$
- ▶ For a dice game :  $P(1) = ?$   $P(2) = ?$   $P(3) = ?$   $P(4) = ?$   
 $P(5) = ?$   $P(6) = ?$

...

└ Probability distributions

└ Reminders on probabilities

# Probability distributions

- ▶ For a dice game, the possible outcomes are in the set  $\{1, 2, 3, 4, 5, 6\}$
- ▶ For a dice game :  $P(1) = \frac{1}{6}$ ,  $P(2) = \frac{1}{6}$ ,  $P(3) = \frac{1}{6}$ ,  $P(4) = \frac{1}{6}$ ,  
 $P(5) = \frac{1}{6}$ ,  $P(6) = \frac{1}{6}$
- ▶ This is called a **uniform distribution**

...

└ Probability distributions

  └ Reminders on probabilities

# Probability distributions

- ▶ Peripherique :

...

└ Probability distributions

  └ Reminders on probabilities

## Probability distributions

- ▶ Peripherique : probably a time-dependent very complicated distribution

...

└ Probability distributions

  └ Reminders on probabilities

## Continuous variables

- ▶ How would you model a continuous variable ? Can you assign a number to a waiting time or a weather ?

...

- Probability distributions

- Reminders on probabilities

## Continuous variables

- ▶ How would you model a continuous variable? Can you assign a number to a waiting time or a weather?
- ▶ One needs to use **probability densities**. Formally, the probability of being between  $x$  and  $x + dx$  is  $p(x)dx$ .

...

- Probability distributions

- Reminders on probabilities

## Continuous variables

- ▶ How would you model a continuous variable? Can you assign a number to a waiting time or a weather?
- ▶ One needs to use **probability densities**. Formally, the probability of being between  $x$  and  $x + dx$  is  $p(x)dx$ .
- ▶ Let's see some examples

...

- Probability distributions

- Reminders on probabilities

## Uniform discrete

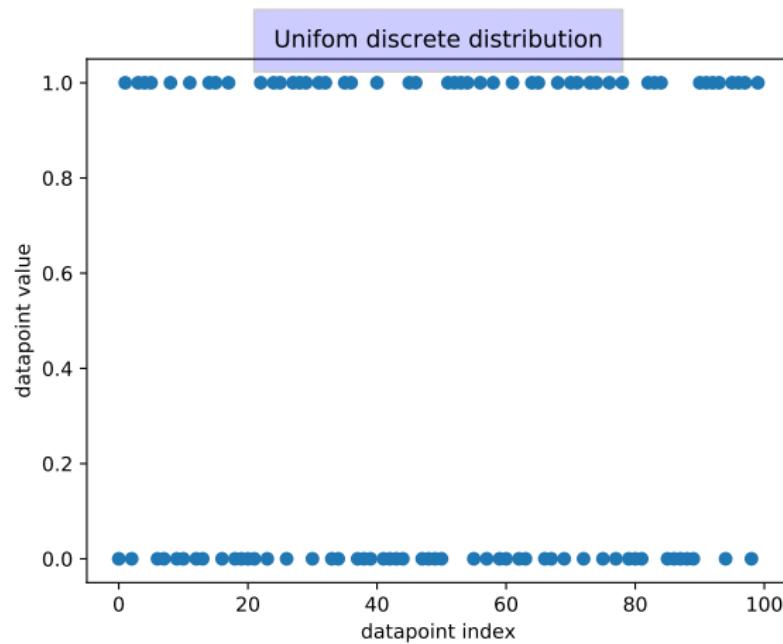


Figure – Uniform discrete distribution

...

- Probability distributions

- Reminders on probabilities

## Bernoulli

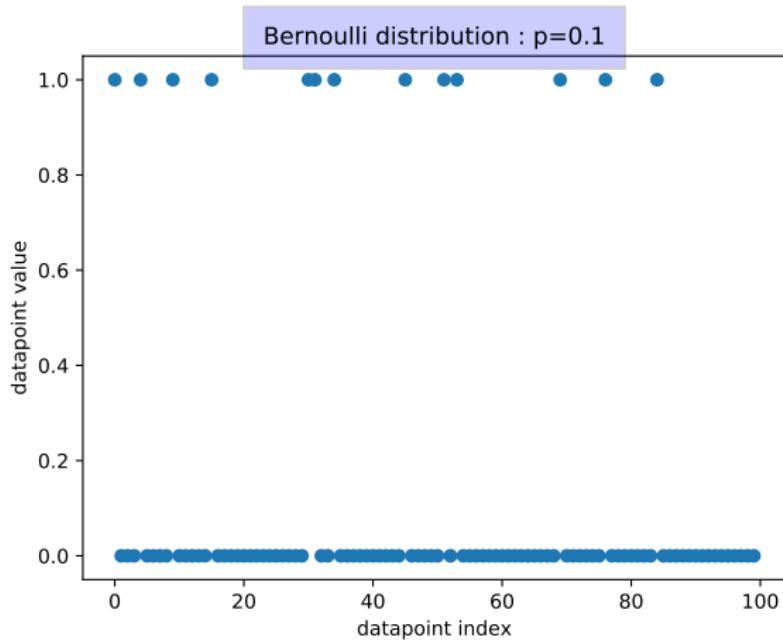


Figure – Bernoulli distribution

...

- └ Probability distributions

- └ Reminders on probabilities

## Bernoulli p

- ▶ With probability  $p$ ,  $X = 1$
- ▶ With probability  $1 - p$ ,  $X = 0$

...

- Probability distributions

- Reminders on probabilities

## Bernoulli

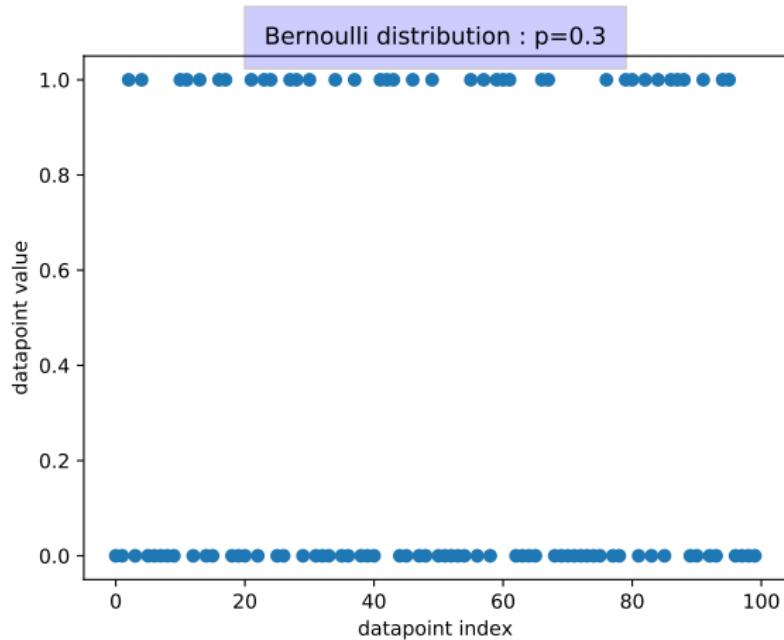
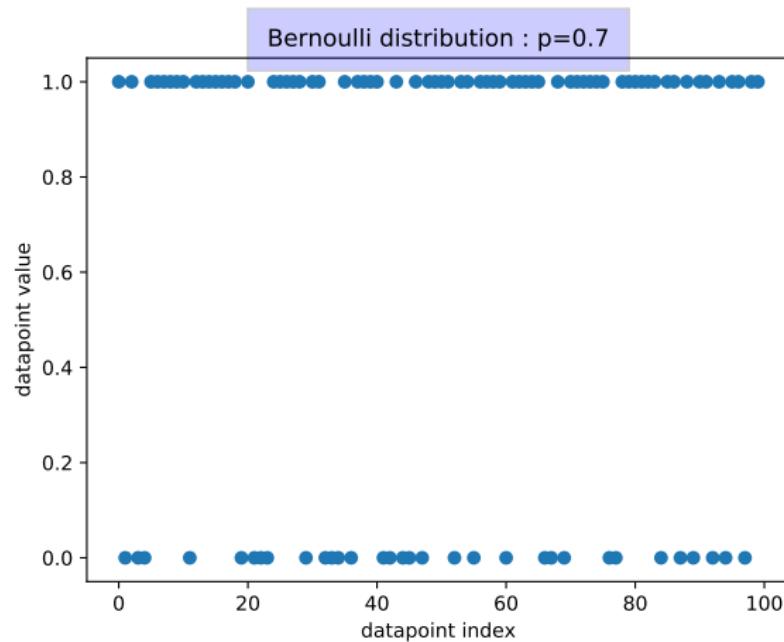


Figure – Bernoulli Distribution

Bernoulli



## Figure – Bernoulli Distribution

...

- Probability distributions

- Reminders on probabilities

## Uniform continuous

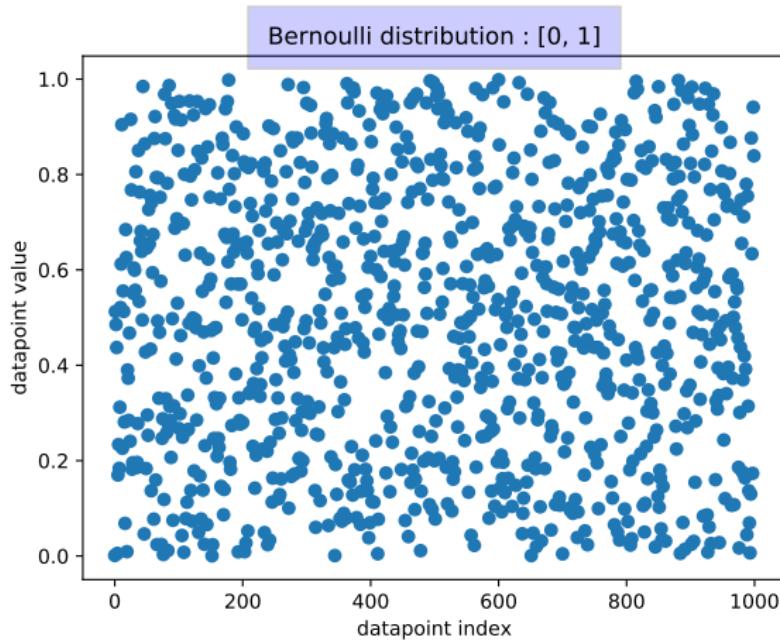


Figure – Uniform continuous distribution

...

- Probability distributions

- Reminders on probabilities

## Uniform continuous

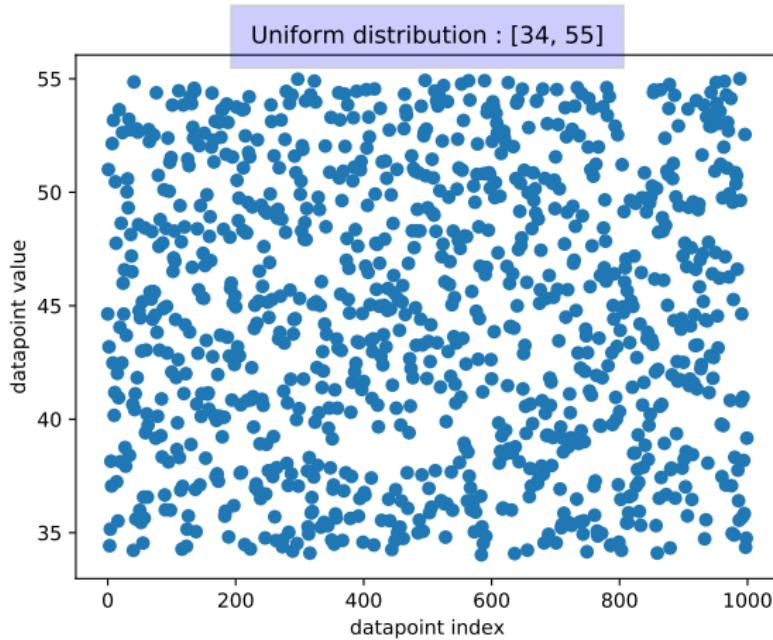


Figure – Uniform continuous distribution

...

- Probability distributions

- Reminders on probabilities

## Normal

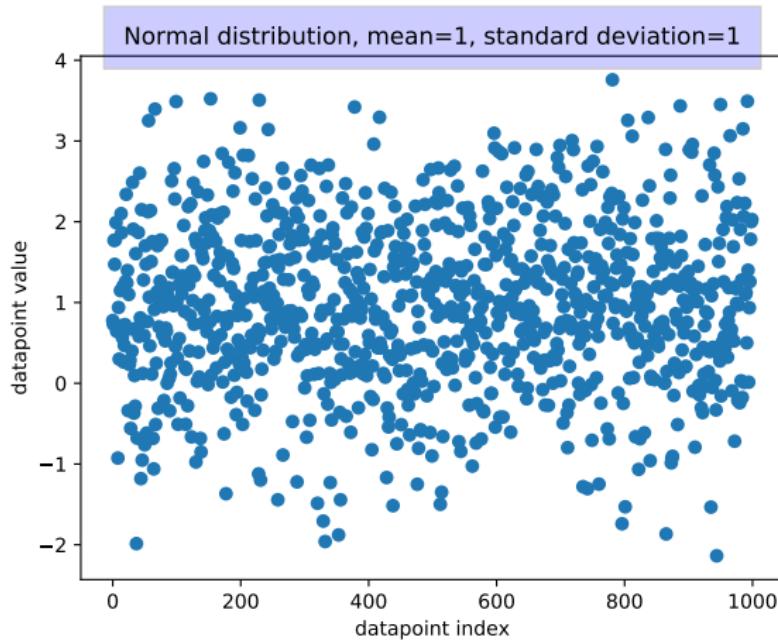


Figure – Normal distribution

...

- Probability distributions

- Reminders on probabilities

## Normal

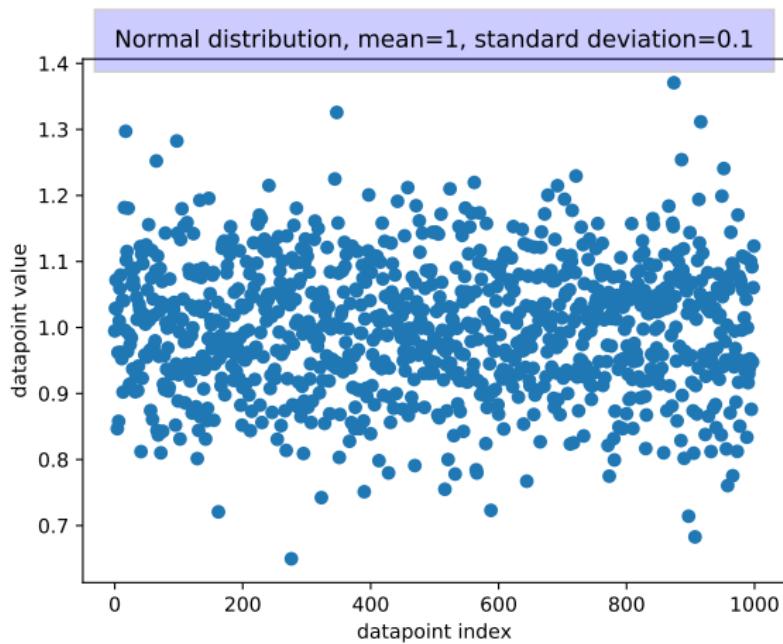


Figure – Normal distribution

...

- Probability distributions

- Reminders on probabilities

## Normal

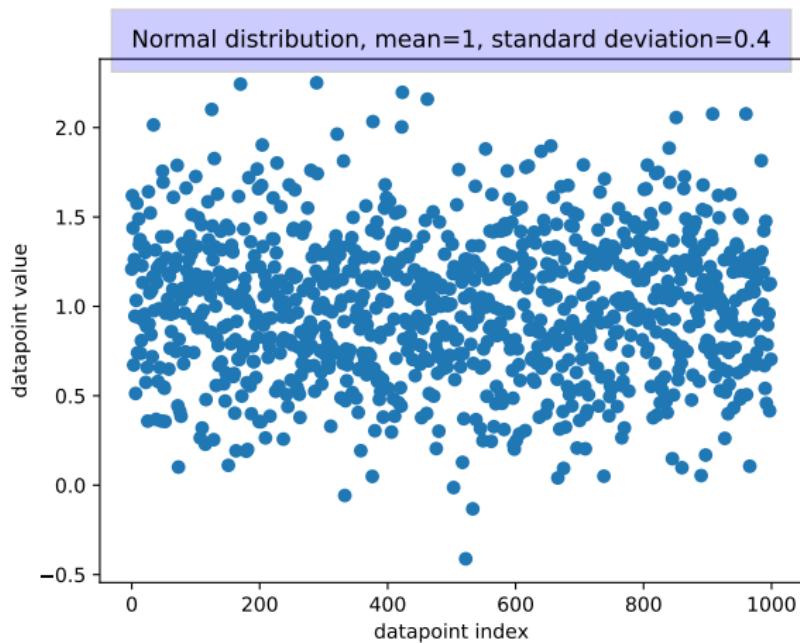


Figure – Normal distribution

...

- Probability distributions

- Reminders on probabilities

## White noise

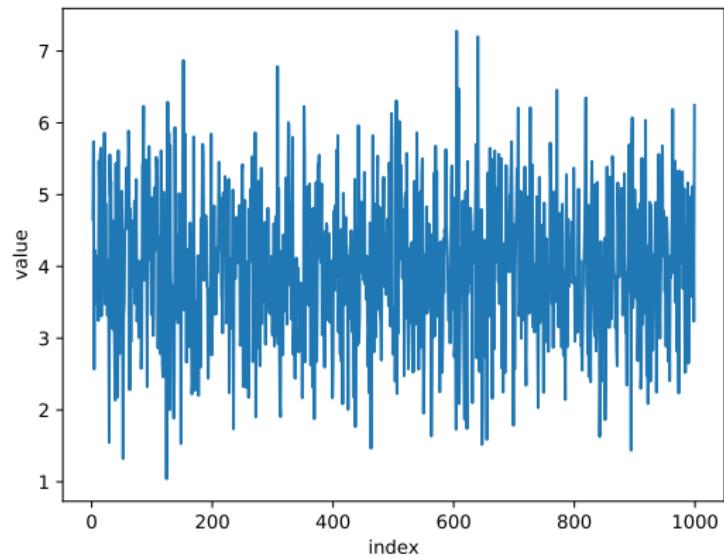


Figure – White noise

...

└ Probability distributions

  └ Reminders on probabilities

## Histograms

Is looking at the raw dataset really **informative** ?

...

- └ Probability distributions
- └ Reminders on probabilities

## Histograms

Is looking at the raw dataset really **informative** ?  
It is informative, but often a **histogram** tells more.

...

- Probability distributions

- Reminders on probabilities

## Uniform discrete

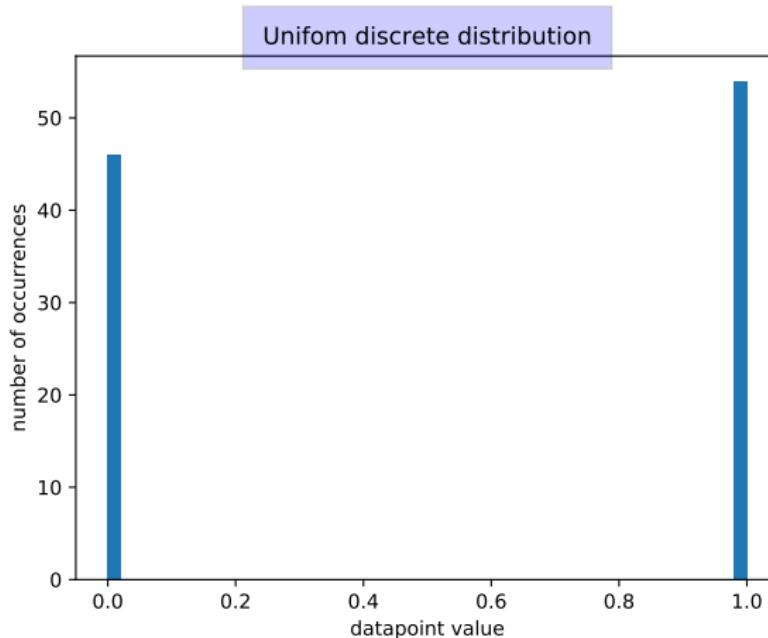


Figure – Historgram 1

...

- Probability distributions

- Reminders on probabilities

## Bernoulli

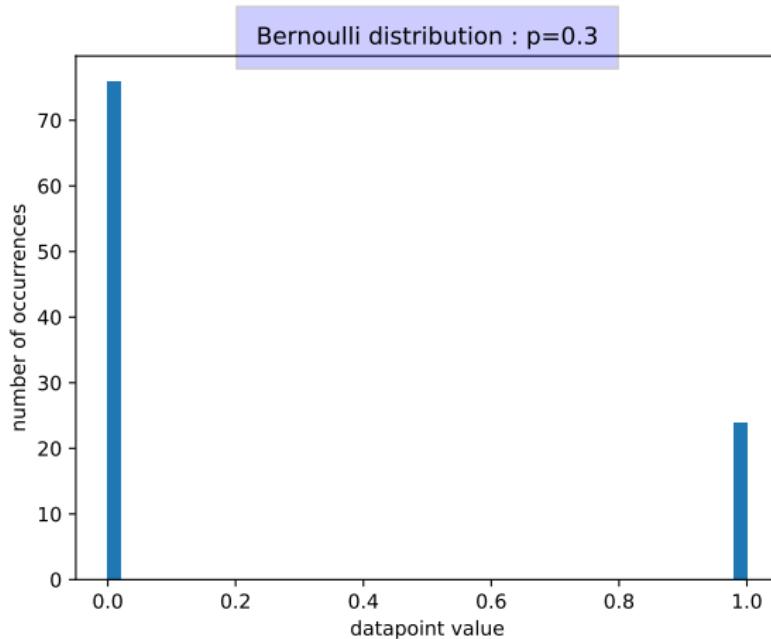


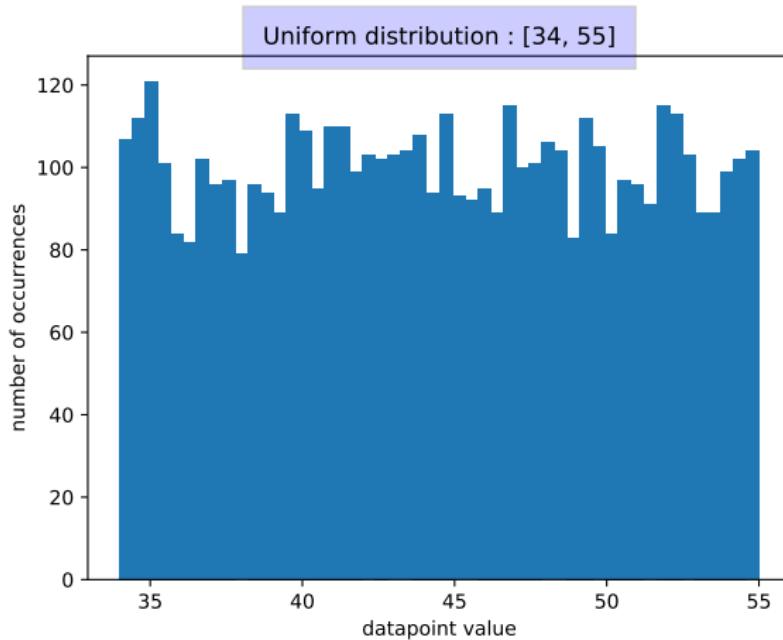
Figure – Histogram 2

...

- Probability distributions

- Reminders on probabilities

## Uniform continuous



...

- Probability distributions

- Reminders on probabilities

## Normal

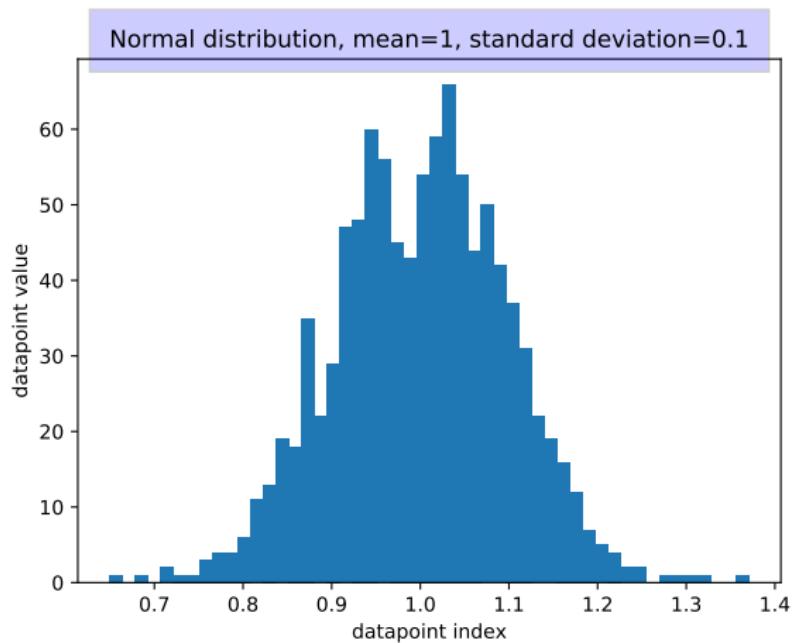


Figure – Histogram 4

...

└ Probability distributions

  └ Analyzing a distribution

Exercice 3 : Analyzing a distribution I put values in the file  
**mysterious\_distro\_1.csv**

...

- └ Probability distributions
  - └ Analyzing a distribution

Exercice 3 : Analyzing a distribution I put values in the file  
**mysterious\_distro\_1.csv**

Can you analyze these values in terms of a **distribution** ?

Use **read\_myst\_1.py** to analyze the distribution (suggestion :  
change the number of bins used)

...

- └ Probability distributions
  - └ Analyzing a distribution

**Exercice 4 :** Analyzing a distribution When you have guessed the kind of distribution it is, you need to finds its **parameters**.

- ▶ its mean
- ▶ its standard deviation

This is called **fitting** a distribution to a dataset : it's a classical machine learning problem.

To do so, uncomment the last section of the script

**read\_myst\_1.py**

...

- └ Probability distributions
  - └ Analyzing a distribution

## Distribution 1

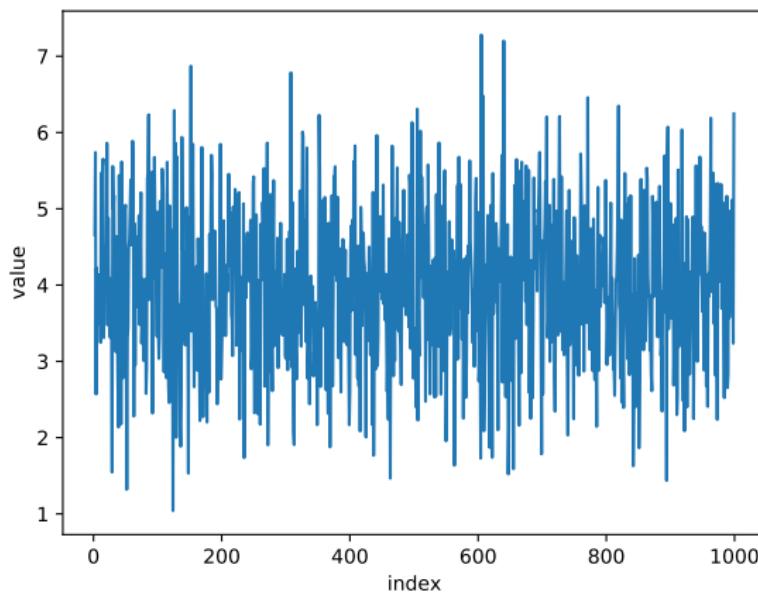


Figure – The data we analyze

...

- └ Probability distributions
  - └ Analyzing a distribution

# histograms

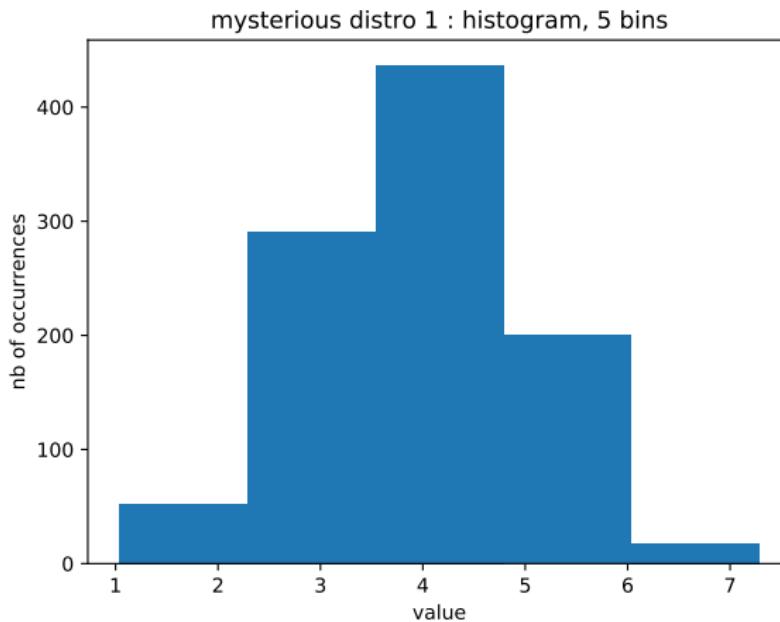


Figure – 5 bins

...

- └ Probability distributions
  - └ Analyzing a distribution

# histograms

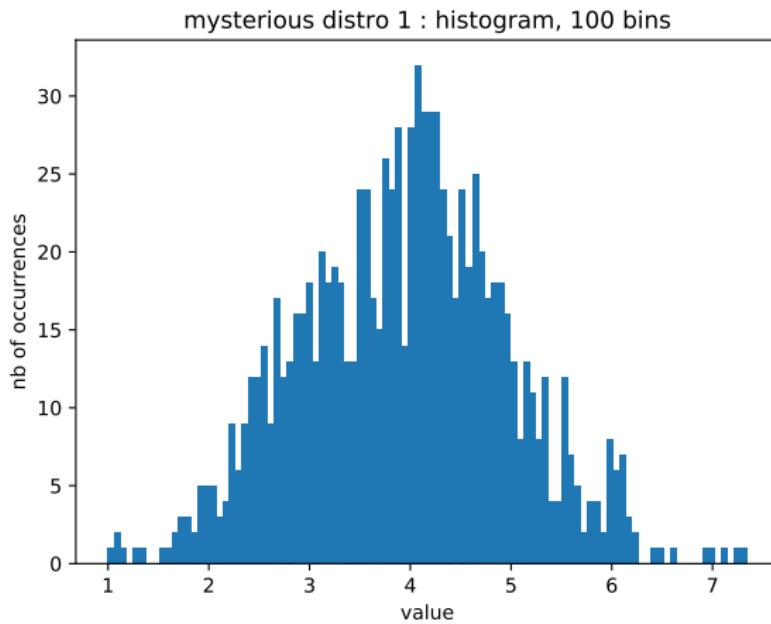


Figure – 100 bins

...

- └ Probability distributions
  - └ Analyzing a distribution

# histograms

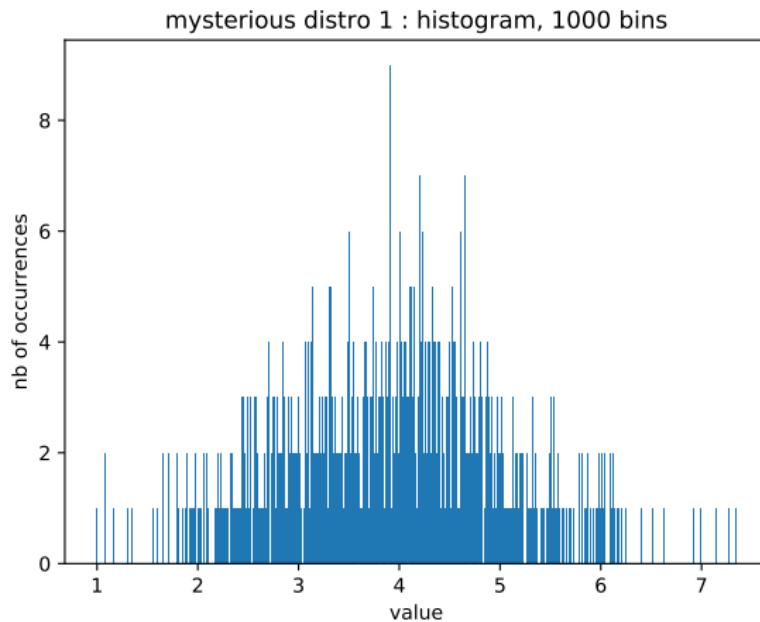


Figure – 1000 bins (too many)

## Normal distribution

```
import csv
import numpy as np

file_name = 'mysterious_distro_1.csv'

mean = 4
std_dev = 1
nb_point = 1000

with open('csv_files/' + file_name, 'w') as csvfile:
    filewriter = csv.writer(csvfile, delimiter=',')
    for point in range(1, nb_point):
        random_variable = np.random.normal(loc=mean, scale=std_dev)
        filewriter.writerow([str(point), str(random_variable)])
```

Figure – **create\_normal.py** : Creation of the distribution

...

└ Probability distributions

  └ Analyzing a distribution

**Exercice 4 :** Second example Let's try to perform the same analysis on the file **mysterious\_distro\_2.csv** using **read\_myst\_2.py**.

...

- └ Probability distributions
  - └ Analyzing a distribution

## Second example

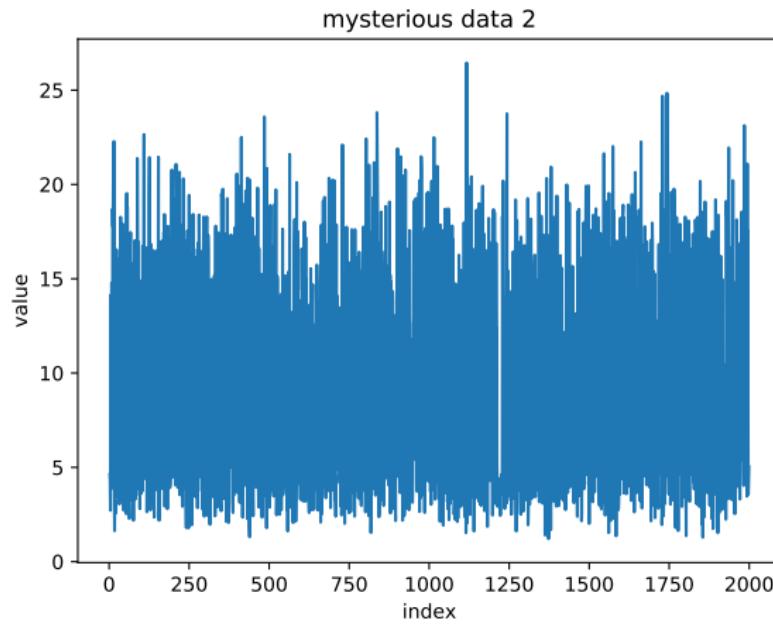


Figure – Second distribution

...

- └ Probability distributions
  - └ Analyzing a distribution

## Multimodal distribution

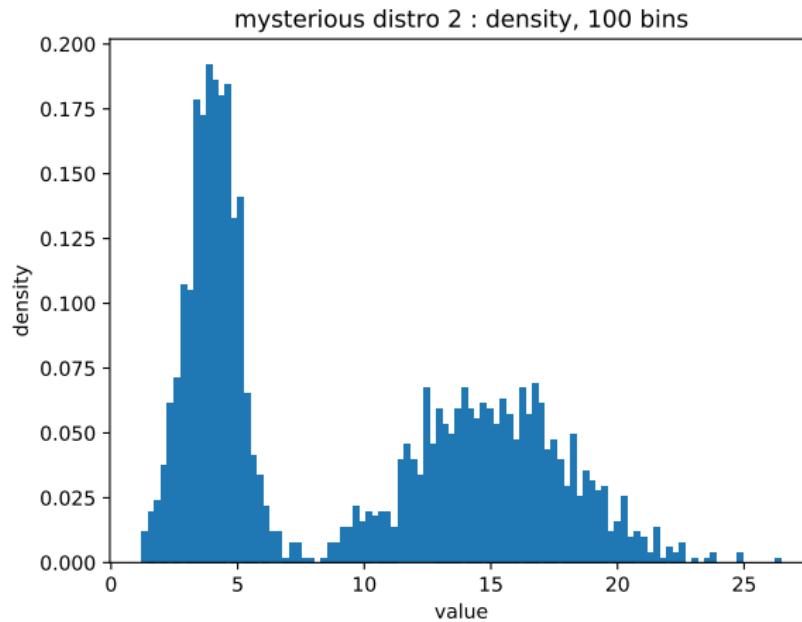


Figure – This distribution has several **modes**

## Multimodal distribution

```
mean_1 = 4
std_dev_1 = 1
nb_point_1 = 1000

mean_2 = 15
std_dev_2 = 3
nb_point_2 = 1000

nb_point = nb_point_1 + nb_point_2

with open('csv_files/' + file_name, 'w') as csvfile:
    filewriter = csv.writer(csvfile, delimiter=',')
    for point in range(1, nb_point):
        if random.randint(1, 2) == 1:
            random_variable = np.random.normal(loc=mean_1, scale=std_dev_1)
            filewriter.writerow([str(point), str(random_variable)])
        else:
            random_variable = np.random.normal(loc=mean_2, scale=std_dev_2)
            filewriter.writerow([str(point), str(random_variable)])
```

Figure – **create\_bimodal.py** : Generation of multimodal distribution

...

└ Probability distributions

  └ Optimization and Maximum Likelihood

## Fitting

In most cases, it won't be that straightforward to fit a distribution :

## Fitting

In most cases, it won't be that straightforward to fit a distribution :

- ▶ what distribution do we want to use ?
- ▶ even if we know the right shape of the distribution, how to choose the parameters ?

## Maximum Likelihood

The **Maximum Likelihood** method is one example method used in Machine Learning.

Say you observe a dataset  $(x_1, \dots, x_n)$ .

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Maximum Likelihood

The **Maximum Likelihood** method is one example method used in Machine Learning.

Say you observe a dataset  $(x_1, \dots, x_n)$ .

You first need to choose a **model** (which is the distribution) of your dataset,  $p$ .

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Maximum Likelihood

The **Maximum Likelihood** method is the one used in Machine Learning.

Say you observe a dataset  $(x_1, \dots, x_n)$ .

You first need to choose a **model** (which is the distribution) of your dataset,  $p$ .

Then, you must optimize the **parameters of this model**, noted  $\theta$ .

...

└ Probability distributions

└ Optimization and Maximum Likelihood

# Maximum Likelihood

The **likelihood** (vraisemblance) of your model is

$$L(\theta) = \prod_{i=1}^n p(x_i|\theta) \quad (1)$$

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Maximum Likelihood

The **likelihood** (vraisemblance) of your model is

$$L(\theta) = \prod_{i=1}^n p(x_i|\theta) \tag{2}$$

This is the function that you want to **maximise**.

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Remark on max-likelihood

Most of the time it's written this way : "minimise  $-\log L(\theta)$ "

Why ?

...

- Probability distributions

- Optimization and Maximum Likelihood

## Remark on max-likelihood

Most of the time it's written this way : "minimise  $-\log L(\theta)$ "

Because the log **transforms the product into a sum**, which is easier to **derivate**.

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Remark on max-likelihood

$$-\log L(\theta) = - \sum_{i=1}^n \log(p(x_i|\theta)) \quad (3)$$

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Example 1

**Exercice 5 :** We observe the data  $(1, 0)$ . We assume that these data come from a random variable that follows a Bernoulli distribution of parameter  $p$ . What is the likelihood of these observations as a function of  $p$ ?

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Example 1

**Exercice 5:** We observe the data  $(1, 0)$ . We assume that these data come from a random variable that follows a Bernoulli distribution of parameter  $p$ . What is the likelihood of these observations as a function of  $p$ ?

$$L = p(1|p)p(0|p) \quad (4)$$

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Example 1

**Exercice 5:** We observe the data  $(1, 0)$ . We assume that these data come from a random variable that follows a Bernoulli distribution of parameter  $p$ . What is the likelihood of these observations as a function of  $p$ ?

$$L = p(1|p)p(0|p) \quad (5)$$

For which value of  $p$  is this likelihood **maximum**?

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Example 2

**Exercice 6:** We observe the data  $(2.5, 3.5)$ . We assume that these data come from a normal law of parameters  $\mu$  and  $\sigma$ .  
What is the likelihood of  $(\mu, \sigma)$ ?

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Example 2

**Exercice 6 :** We observe the data  $(2.5, 3.5)$ . We assume that these data come from a normal law of parameters  $\mu$  and  $\sigma$ .

$$L = p(2.5|\mu, \sigma)p(3.5|\mu, \sigma) \quad (6)$$

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Example 2

Exercice 6 : We observe the data  $(2.5, 3.5)$ . We assume that these data come from a normal law of parameters  $\mu$  and  $\sigma$ .

$$\begin{aligned} L &= p(2.5|\mu, \sigma)p(3.5|\mu, \sigma) \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{2.5-\mu}{\sigma}\right)^2} \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{3.5-\mu}{\sigma}\right)^2} \end{aligned} \tag{7}$$

...

└ Probability distributions

└ Optimization and Maximum Likelihood

## Example 2

**Exercice 6 :** We observe the data  $(2.5, 3.5)$ . We assume that these data come from a normal law of parameters  $\mu$  and  $\sigma$ .

$$\begin{aligned} L &= p(2.5|\mu, \sigma)p(3.5|\mu, \sigma) \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{2.5-\mu}{\sigma}\right)^2} \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{3.5-\mu}{\sigma}\right)^2} \end{aligned} \tag{8}$$

We wan show that the likelihood is maximum for :

- ▶  $\hat{\mu} = \frac{2.5+3.5}{2}$
- ▶  $\hat{\sigma}^2 = \frac{(2.5-\hat{\mu})^2 + (3.5-\hat{\mu})^2}{2}$

...

└ Probability distributions

  └ Gradients

## Max Likelihood

In the case of very large datasets, and large numbers of parameters (tens, hundredths, more), most of the time an **analytic solution** is not available.

...

- Probability distributions

- Gradients

## Max Likelihood

In the case of very large datasets, and large numbers of parameters (tens, hundredths, more), most of the time an **analytic solution** is not available. So how can we **maximize** the likelihood ?

...

└ Probability distributions

└ Gradients

## Max Likelihood

In the case of very large datasets, and large numbers of parameters (tens, hundredths, more), most of the time an **analytic solution** is not available. So how can we **maximize** the likelihood ?

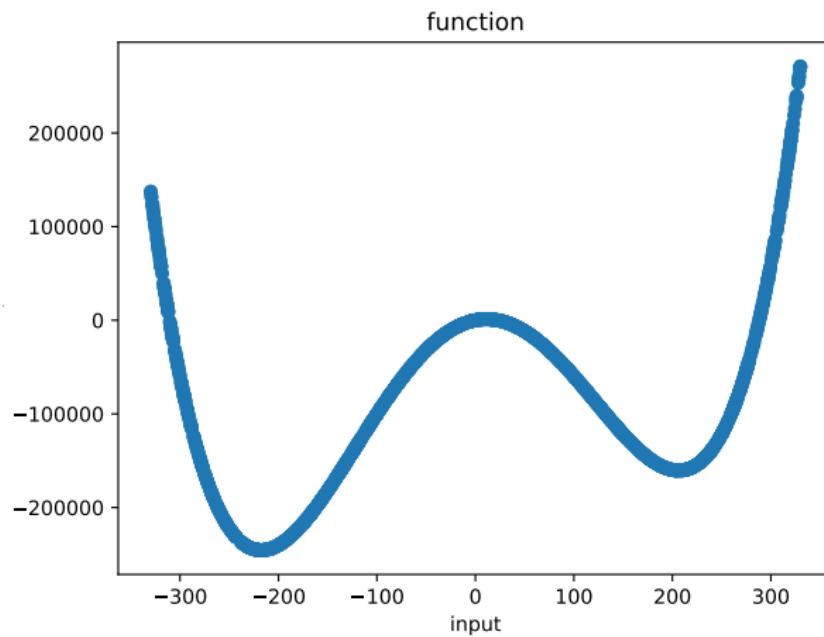
Most common method : **gradient descent**.

...

Probability distributions

Gradients

## Notion of optimization



...

└ Probability distributions

└ Gradients

## Gradient descent

- ▶ In the case a function  $f : \mathbb{R} \rightarrow \mathbb{R}$ , we can study its variations by computing its derivative  $f'$ , **if it exists**

...

└ Probability distributions

└ Gradients

## Gradient descent

- ▶ In the case a function  $f : \mathbb{R} \rightarrow \mathbb{R}$ , we can study its variations by computing its derivative  $f'$ , **if it exists**
- ▶ If  $f'(x) > 0$ , the function grows around  $x$ .
- ▶ If  $f'(x) < 0$ , the function decreases around  $x$ .
- ▶ If  $x$  is a local extremum,  $f'(x) = 0$

...

└ Probability distributions

└ Gradients

## Gradient descent

- ▶ In the case a function  $f : \mathbb{R} \rightarrow \mathbb{R}$ , we can study its variations by computing its derivative  $f'$ , **if it exists**
- ▶ If  $f'(x) > 0$ , the function grows around  $x$ .
- ▶ If  $f'(x) < 0$ , the function decreases around  $x$ .
- ▶ If  $x$  is a local extremum,  $f'(x) = 0$
- ▶ Is the reciprocal true ?

...

└ Probability distributions

└ Gradients

# Gradient

- ▶ The **gradient** is similar to a derivative but in the case of a function with several inputs, such as  $(\mu, \theta)$ .
- ▶ Then we store the **partial derivative** with respect to each input in a **vector** called the gradient.

## Gradient descent

Consider a function  $f$  that has 2 parameters as inputs.

$$\nabla_f(x, y) = \left( \frac{\delta f}{\delta x}, \frac{\delta f}{\delta y} \right) \quad (9)$$

We want  $x$  to **minimise**  $f$ . We perform, until some criteria is satisfied :

$$x \leftarrow x - \alpha \nabla_f(x) \quad (10)$$

$\alpha$  is a small parameter called the learning rate.

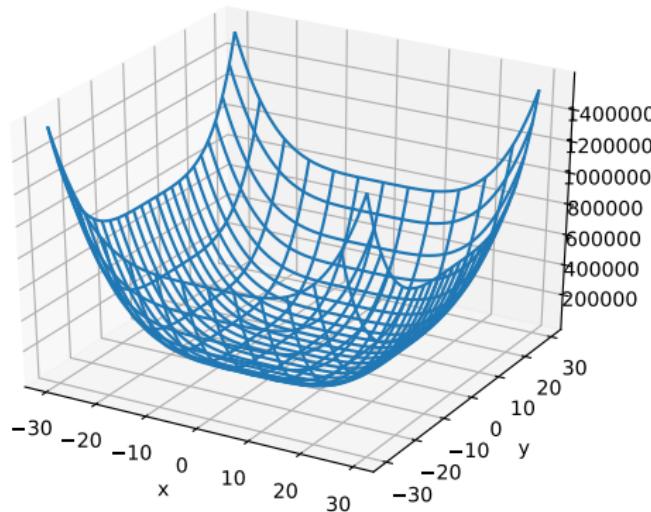
...

Probability distributions

Gradients

# Gradient

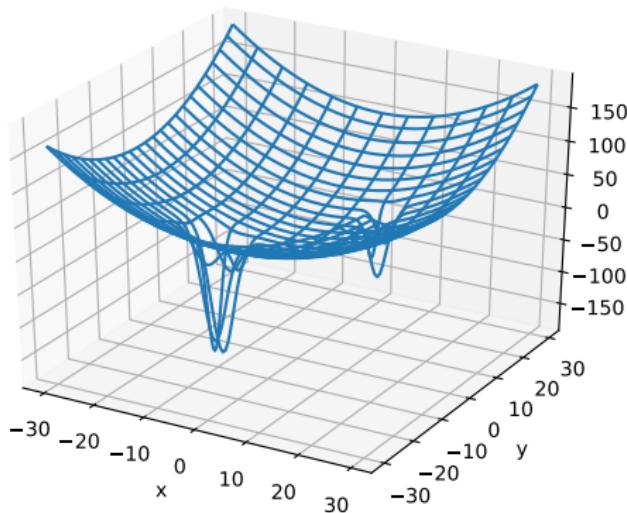
**Exercice 6 : Using the gradient algorithm** We will use the algorithm on two functions.



...

# Gradient

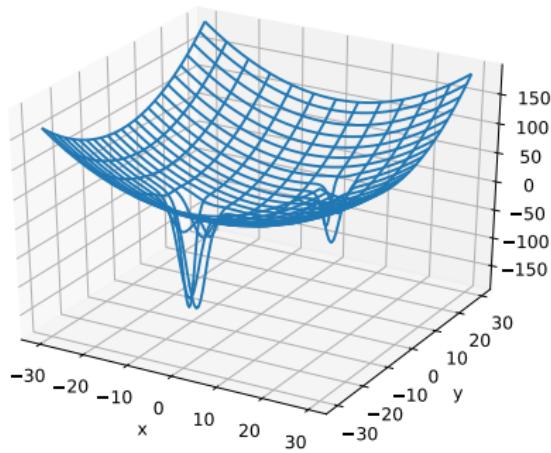
**Exercice 6 : Using the gradient algorithm** We will use the algorithm on two functions.



...

## Gradient

**Exercice 6:** Using the gradient algorithm `cd ./gradient` and use the files `gradient.py` and `gradient_2.py` in order to implement the algorithm to find **minima**.



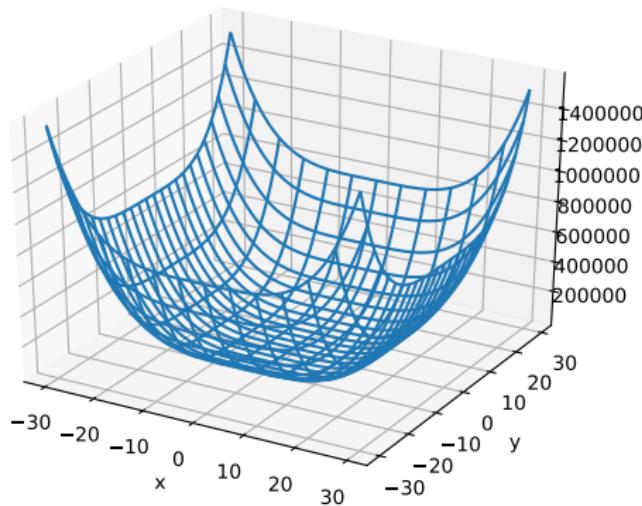
...

Probability distributions

Gradients

## The gradient descent

Experiment with it, try to change all the parameters and to break it again. Is it stable?



## Multidimensional vectors

We can consider data that live in higher dimensional spaces than 2.

...

└ Multivariate analysis and clustering

  └ Correlation

## Multidimensional vectors

We can consider data that live in higher dimensional spaces than 2.  
Examples ?

## Multidimensional vectors

We can consider data that live in higher dimensional spaces than 2.  
Examples ?

- ▶ images
- ▶ sensor that receives **multimodal information**

## Correlation

Sometimes the components of a multidimensional vector  $(x_1, \dots, x_n)$  are not independent.

...

## Correlation

Sometimes the components of a multidimensional vector  $(x_1, \dots, x_n)$  are not independent.

To study this, we can use the **covariance** of the two components, or the **correlation** which is actually clearer.

...

└ Multivariate analysis and clustering

└ Correlation

## Expected value (espérance)

- ▶ For a discrete random variable  $X$  that takes the values  $x_i$  with probability  $p_i$  :

$$E(X) = \sum_{i=1}^n p_i x_i \quad (11)$$

- ▶ For a continuous random variable  $X$  with density  $p(x)$  :

$$E(X) = \int p(x) dx \quad (12)$$

## Expected value (espérance)

Exercice 6 : Computing an expected value

- ▶ For a discrete random variable  $X$  that takes the values  $x_i$  with probability  $p_i$  :

$$E(X) = \sum_{i=1}^n p_i x_i \tag{13}$$

- ▶ For a continuous random variable  $X$  with density  $p(x)$  :

$$E(X) = \int p(x) dx \tag{14}$$

Compute the expected value of the dice game.

## Variance

$$\text{var}(X) = E((X - E(X))^2) \quad (15)$$

...

- Multivariate analysis and clustering

- Correlation

## Variance and Covariance

$$\text{var}(X) = E((X - E(X))^2) \quad (16)$$

$$\text{cov}(X, Y) = E((X - E(X))(Y - E(Y))) \quad (17)$$

## Example

Look at the data contained in **mysterious\_distro\_3.csv**  
They contain a random variable with 5 dimensions. Some of these dimensions are correlated.  
Think for instance to physics : temperature and pressure, etc. If you have measurements of temperature and pressure, the two would probably be **correlated**.

## Correlation

Exercice 7 : Which dimensions of the distribution are correlated ?

...

- Multivariate analysis and clustering

- Correlation

## Correlation matrix

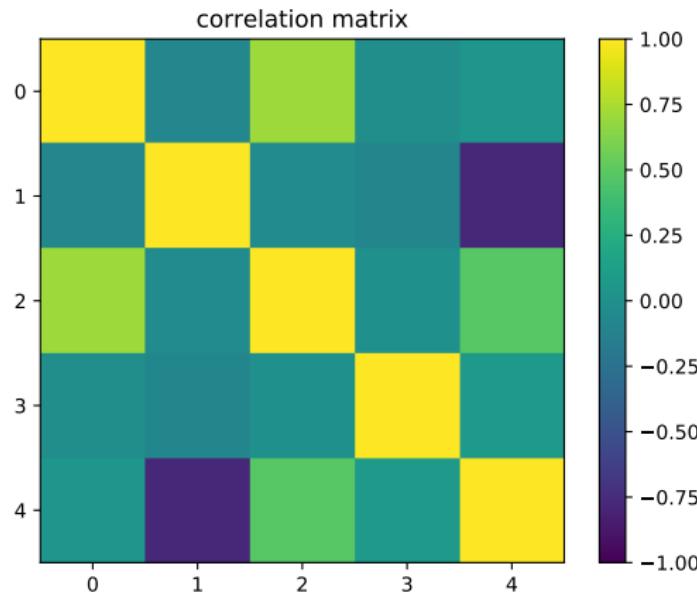


Figure – Correlation matrix for the distribution

## Generation of the data

```
mean_1 = 4
std_dev_1 = 1

mean_2 = 15
std_dev_2 = 3

mean_3 = -5
std_dev_3 = 2

mean_noise = 0
noise_std_dev = 1

nb_point = 1000

with open('csv_files/' + file_name, 'w') as csvfile:
    filewriter = csv.writer(csvfile, delimiter=',')
    for point in range(1, nb_point):
        noise = np.random.normal(loc=mean_noise, scale=noise_std_dev)
        random_variable_1 = np.random.normal(loc=mean_1, scale=std_dev_1)
        random_variable_2 = np.random.normal(loc=mean_2, scale=std_dev_2)
        random_variable_3 = random_variable_1 + noise
        random_variable_4 = np.random.normal(loc=mean_3, scale=std_dev_3)
        random_variable_5 = -0.4 * random_variable_2 + noise
        filewriter.writerow([str(point),
                            str(random_variable_1),
                            str(random_variable_2),
                            str(random_variable_3),
                            str(random_variable_4),
                            str(random_variable_5)])
```

Figure – Multidimensional random variable

...

└ Multivariate analysis and clustering

  └ Dimension reduction

## Removing dimensions

- ▶ Sometimes given a question and a dataset, not all dimensions of the data are **relevant**
- ▶ It is possible that only one or two of them are sufficient to answer the given question

...

- Multivariate analysis and clustering

- Dimension reduction

## Removing dimensions

- ▶ Sometimes given a question and a dataset, not all dimensions of the data are **relevant**
- ▶ It is possible that only one or two of them are sufficient to answer the given question
- ▶ We will illustrate this with the titanic dataset and the pandas library

## Install pandas

Install pandas

# Titanic Dataset

- The dataset contains the list of passengers and several informations on each of them :

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25	S	
2	1	1	Curtis, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17399	71.2033	C86	C
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 3101282	7.925	S	
4	1	1	Holtedahl, Mrs. Jacques Heude (July May Heude)	female	35	1	0	113833	33.1	C133	S
5	0	3	Allen, Mr. William Henry	male	35	0	0	373493	8.95	S	
6	0	3	Horan, Ms. James	male	0	0	0	330877	8.4983	Q	
7	0	1	McCarthy, Ms. Timothy J	male	54	0	0	17460	51.8625	C46	S
8	0	3	Palsson, Master Gustaf Leonard	male	2	3	1	349699	21.075	S	
9	1	3	Johnson, Mrs. Oscar W (Elsieast Vilhelmina Berg)	female	27	0	2	347742	11.1333	S	
10	1	2	Nasser, Mrs. Nicholas (Adele Ahola)	female	14	1	0	237734	30.0708	C	
11	1	3	Sandstrom, Miss. Marguerite Rut	female	4	1	1	MM 35648	16.7	G8	S
12	1	3	Burnett, Miss. Elizabeth	female	58	0	0	113983	26.95	C103	S
13	0	3	Saundercock, Mr. William Henry	male	20	0	0	A/5 215	8.95	S	
14	0	3	Andersen, Mr. Anderson John	male	39	1	5	347082	31.275	S	
15	0	3	Vestrum, Miss. Hilda Aneade Adolfa	female	14	0	0	350486	7.8542	S	
16	1	2	Hewlett, Mrs. (Mary D KIngcome)	female	55	0	0	248785	16	S	
17	0	3	Rice, Master Eugene	male	2	4	1	302623	26.125	Q	
18	1	2	Williams, Mr. Charles Eugene	male	0	0	0	244373	13	S	
19	0	3	Vander Valken, Mrs. Julia (Elinia Maria Vandervoordt)	female	31	1	0	349783	18	S	
20	1	3	Masseveldt, Mrs. Patricia	female	0	0	0	2649	7.225	C	
21	0	2	Payne, Mr. Joseph J	male	35	0	0	239885	26	S	
22	1	2	Beauchamp, Mr. Lawrence	male	34	0	0	249698	13	D56	S
23	1	3	McGowan, Miss. Anna "Annie"	female	15	0	0	330923	0.8202	Q	
24	1	1	Shaper, Mr. William Thompson	male	26	0	0	113788	35.5	A6	S
25	0	3	Peterson, Miss. Tatanya Daniels	female	8	3	1	346699	21.075	S	
26	1	3	Asplund, Mrs. Carl Oscar (Selma Augusta (Selma Johanna) Asplund)	female	36	1	5	341037	31.3675	S	
27	0	3	Erick, Mr. Farred Charles	male	0	0	0	2631	7.225	C	
28	0	1	Perkins, Mr. Charles Alexander	male	19	5	2	19988	281	C25 C25 C27	S
29	1	3	O'Dwyer, Miss. Ellen "Nellie"	female	0	0	0	330999	7.8792	Q	
30	0	3	Todoroff, Mr. Lalin	male	0	0	0	349218	7.8958	S	
31	0	1	Urschutz, Dr. Max E	male	40	0	0	PC 17601	27.7508	C	
32	1	3	Spicer, Mrs. William Augustus (Maria Eugenia)	female	1	0	0	PC 17569	146.5208	B76	C
33	1	3	Glynn, Miss. Mary Augusta	female	0	0	0	235677	7.75	Q	
34	0	2	Wheeler, Miss. Edward H	male	66	0	0	CA 24579	16.5	S	
35	0	1	Hoyos, Mr. Bolivar Jose	male	28	1	0	PC 17604	82.1708	C	
36	0	1	Hansen, Mr. Alexander Oskar	male	42	1	0	113799	52	S	
37	1	3	Horne, Mr. Harry	male	0	0	2077		7.2292	C	
38	0	3	Caro, Mr. Ernest Charles	male	21	0	0	A/5 2123	8.95	S	
39	0	3	Vander Valken, Miss. Augusta Maria	female	18	2	0	345784	18	S	
40	1	3	Nicole-Yerrell, Miss. Junilia	female	14	1	0	2651	11.2417	C	

...

## Titanic Dataset

- ▶ The dataset contains the list of passengers and several informations on each of them :

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket
1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 31
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803
5	0	3	Allen, Mr. William Henry	male	35	0	0	373450
6	0	3	Moran, Mr. James	male	0	0	0	330877
7	n	1	McCarthy, Mr. Timothy J	male	34	n	n	17463

Figure – Titanic dataset : can be found here  
<https://www.kaggle.com/c/titanic/data>

## Pandas

- ▶ The pandas library is used to study large datasets with python
- ▶ We will use the **Dataframe** structure to process the titanic dataset
- ▶ **cd multivariate\_analysis/** and use the file **pandas\_infos.py** to load the dataset to a dataframe and print general information on the dataframe.

# Pandas

```
---  
all info on passenger 25  
PassengerId                      26  
Survived                           1  
Pclass                             3  
Name      Asplund, Mrs. Carl Oscar (Selma Augusta Emilia...  
Sex                                female  
Age                               38  
SibSp                            1  
Parch                            5  
Ticket                          347077  
Fare                            31.3875  
Cabin                           NaN  
Embarked                         S  
Name: 25, dtype: object  
---  
age of passenger 25  
38.0
```

Figure – Some passenger that survived

## Exercice 8 : Prediction

We would like to know if there is a criterion that can help to predict if a passenger survived or not.

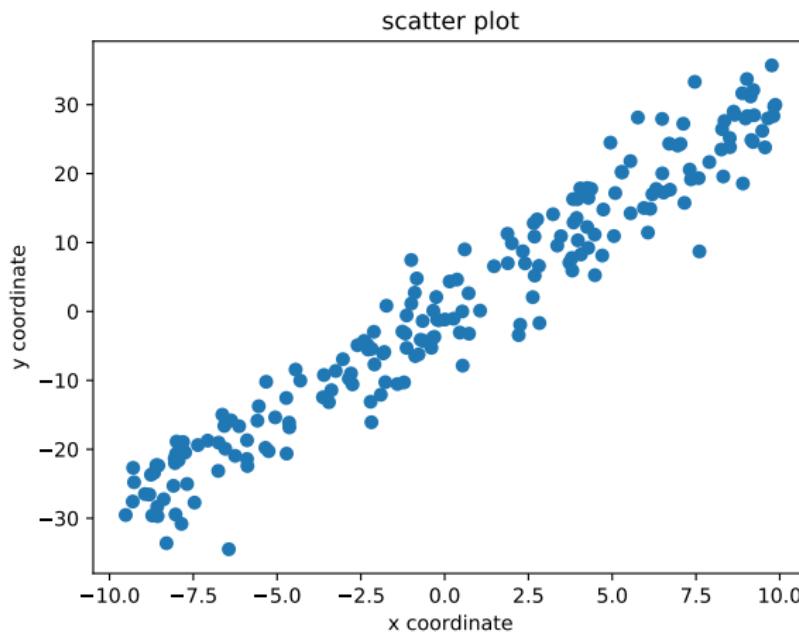
Ideally, we would like to remove dimensions that are useless for this problem.

...

- Multivariate analysis and clustering

- Dimension reduction

## Scatter plot (nuage de points)

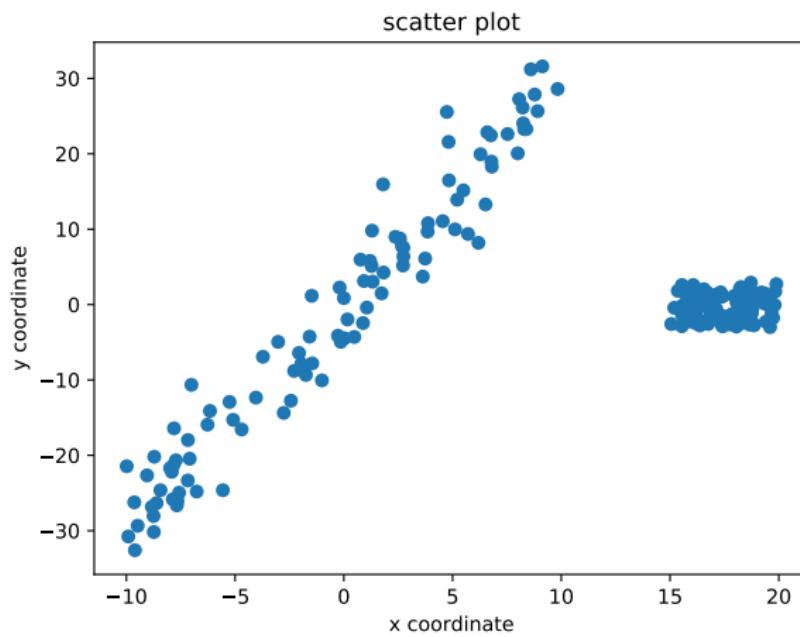


...

- Multivariate analysis and clustering

- Dimension reduction

## Scatter plot (nuage de points)



## Exercice 8 : Prediction

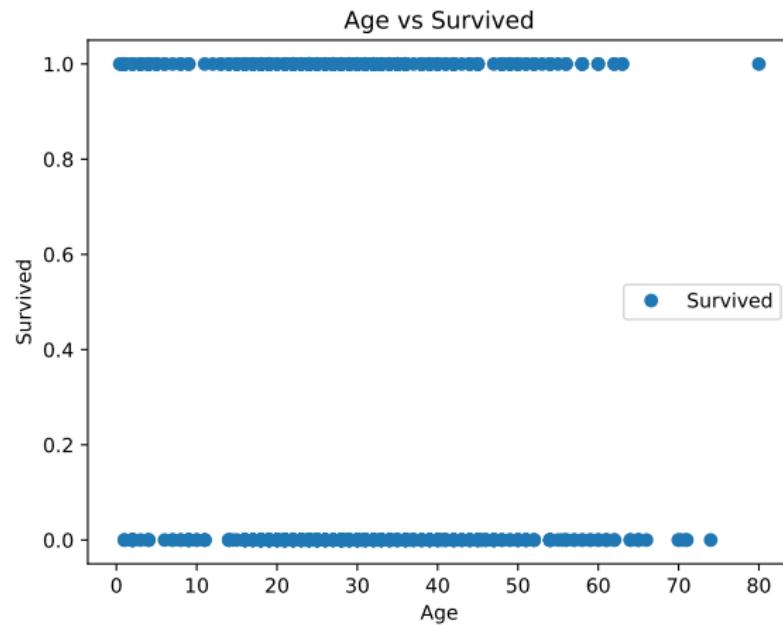
`cd multivariate_analysis/titanic/.`

Use the file `scatter_titanic.py` to see if one column is sufficient to predict survival. We use the `seaborn` lib.

...

- Multivariate analysis and clustering

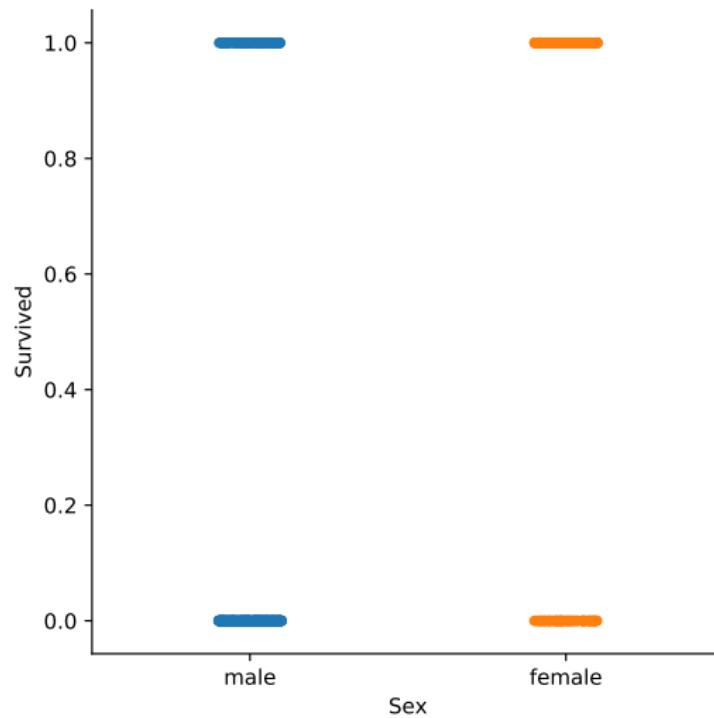
- Dimension reduction



...

- Multivariate analysis and clustering

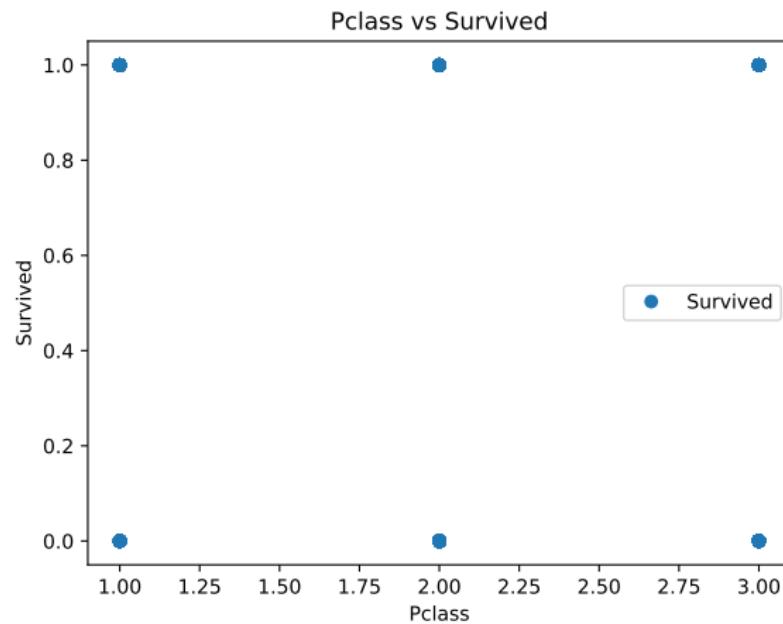
- Dimension reduction



...

- Multivariate analysis and clustering

- Dimension reduction



## Exercice 8 : Prediction

How could we plot 3 variables on the same graph ?

### Exercice 8 : Prediction

How could we plot 3 variables on the same graph ?

Use the file **scatter\_titanic\_color.py** in order to color the datapoint in a 2D space as a function of survival.

## Exercice 8 : Prediction

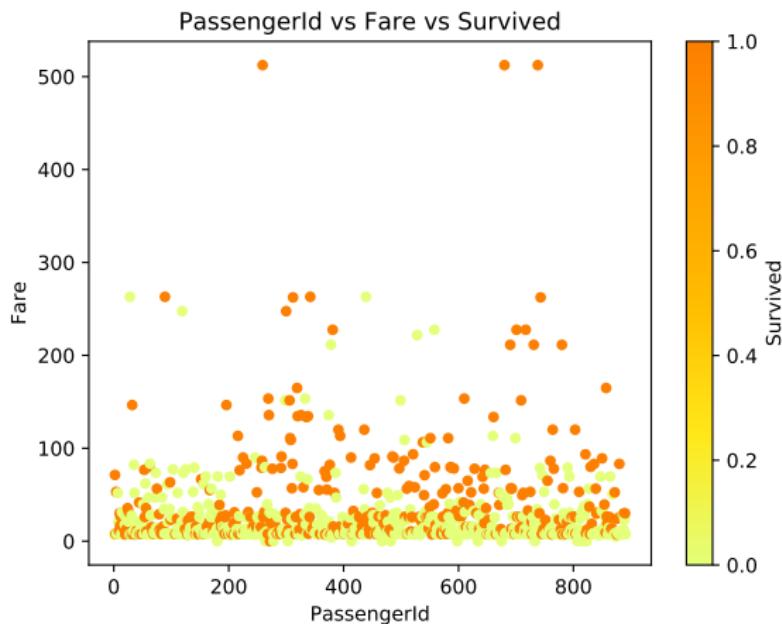


Figure – Not much information

## Exercice 8 :

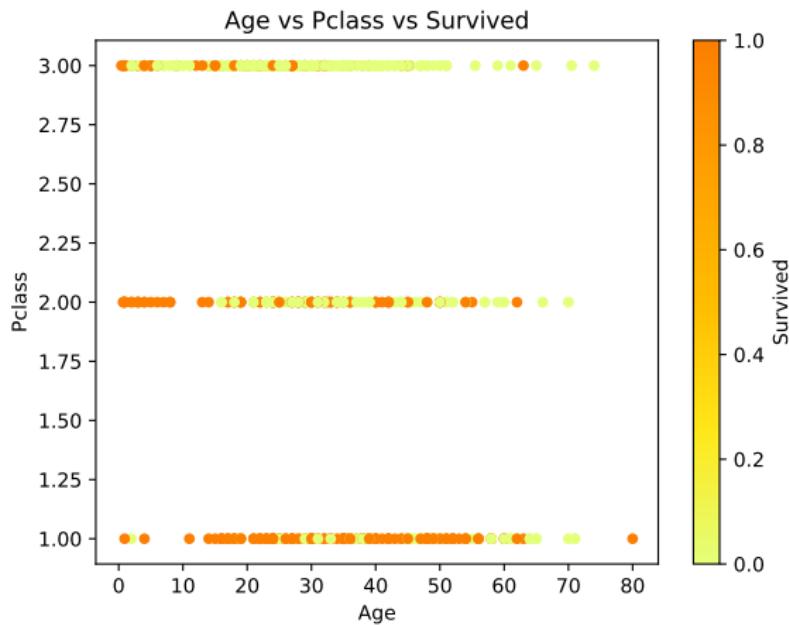


Figure – There seems to be a link between class and survival

Remark : we did not have to use pandas to solve this problem.

## Warning

- ▶ Some times removing dimensions can lead to a **misinterpretation**

...

- └ Multivariate analysis and clustering
  - └ Correlation and causality

### Exercice 9 :

`cd multivariate_analysis/causality/.`

- ▶ Load the dataset `grades.csv` and study it using `process_grades.py`
- ▶ What columns are correlated ?

## Exercice 9 :

- ▶ **cd causality**
- ▶ Load the dataset **grades.csv** and study it using **process\_grades.py**
- ▶ What columns are correlated ?
- ▶ Plot the grade as a function of the height.

...

- Multivariate analysis and clustering

- Correlation and causality

## Exercice 9 :

- ▶ Load the dataset `grades.csv` and study it using `process_grades.py`
- ▶ What columns are correlated ?
- ▶ Plot the grade as a function of the height.
- ▶ Plot the grade as a function of the age.

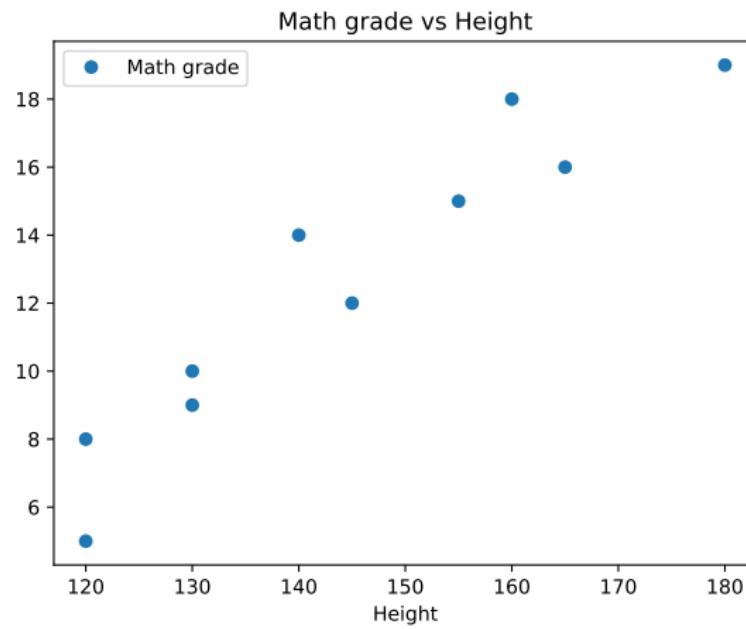
## Exercice 9 :

- ▶ Load the dataset **grades.csv** and study it using **process\_grades.py**
- ▶ What columns are correlated ?
- ▶ Plot the grade as a function of the height.
- ▶ Plot the grade as a function of the age.
- ▶ Which one of these plots does not make sense ?

...

- └ Multivariate analysis and clustering
  - └ Correlation and causality

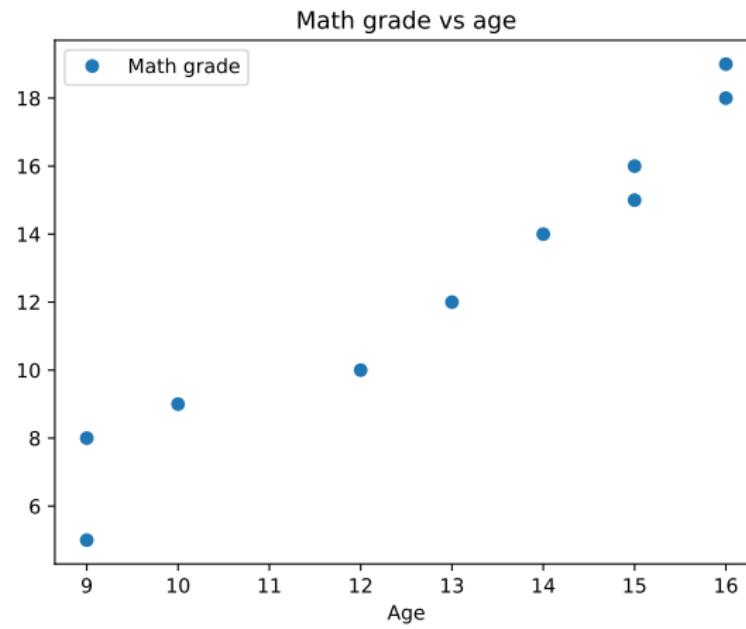
## Correlation and causality



...

- └ Multivariate analysis and clustering
  - └ Correlation and causality

## Correlation and causality



## Conclusion

- ▶ The height of a student is actually not linked to his/her grade
- ▶ If we plot the grade as a function of the height There is a **hidden variable** which is the age of the student
- ▶ Correlation is different than causality.

## Scatter matrix

- ▶ Let us generalize the idea of a scatter plot.

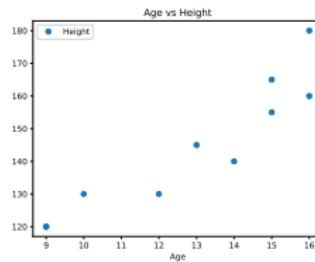


Figure – Exemple scatter plot

- ▶ Given a dataset, we can build all the possible scatter plots and store them in a matrix, called the **scatter matrix**.

## Scatter matrix

- ▶ Let us generalize the idea of a scatter plot.

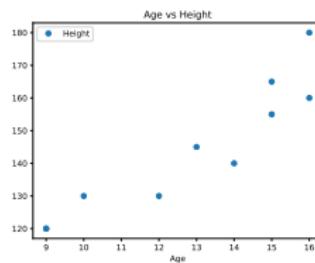


Figure – Exemple scatter plot

- ▶ Given a dataset, we can build all the possible scatter plots and store them in a matrix, called the **scatter matrix**.
- ▶ What will happen on the diagonal of the matrix ?

## Scatter matrix

- ▶ Let us generalize the idea of a scatter plot.

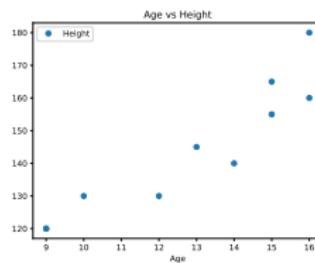


Figure – Exemple scatter plot

- ▶ Given a dataset, we can build all the possible scatter plots and store them in a matrix, called the **scatter matrix**.
- ▶ What will happen on the diagonal of the matrix ?

## Scatter matrix

- ▶ Let us generalize the idea of a scatter plot.
- ▶ Given a dataset, we can build all the possible scatter plots and store them in a matrix, called the **scatter matrix**.
- ▶ What will happen on the diagonal of the matrix ?

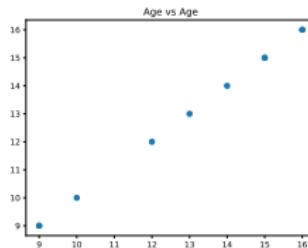


Figure – Variable plotted against itsself : All the points are on the  $y = x$  line

...

└ Multivariate analysis and clustering

└ Scatter matrix

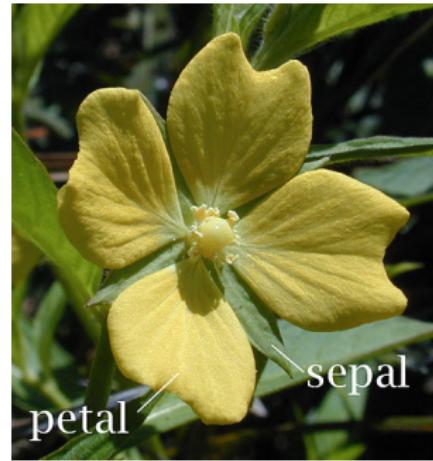
## Scatter matrix

- ▶ Let us generalize the idea of a scatter plot.
- ▶ Given a dataset, we can build all the possible scatter plots and store them in a matrix, called the **scatter matrix**.
- ▶ On the diagonal, one can plot histograms or the density probability
- ▶ The scatter plot can be a good way to start analyzing a dataset when we don't know which variables could be correlated

...

## Iris dataset

- ▶ 150 samples of iris flower
- ▶ 3 species
- ▶ 4 attributes : petal width and length, sepal width and length



Exercice 10 : Iris dataset : scatter matrix  
`cd multivariate_analysis/titanic/.`

- ▶ use `iris_scatter_matrix.py` to plot the scatter matrix of the iris dataset with seaborn.
- ▶ Is there a variable that can discriminate between the species ?

## Exercice 10 : Iris dataset : scatter matrix

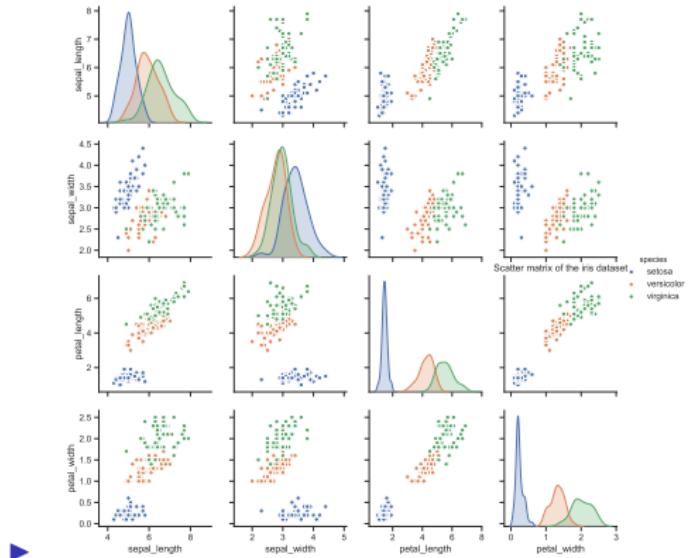


Figure – Plotted in `iris_scatter_matrix.py`

## Exercice 10 : Iris dataset : scatter matrix

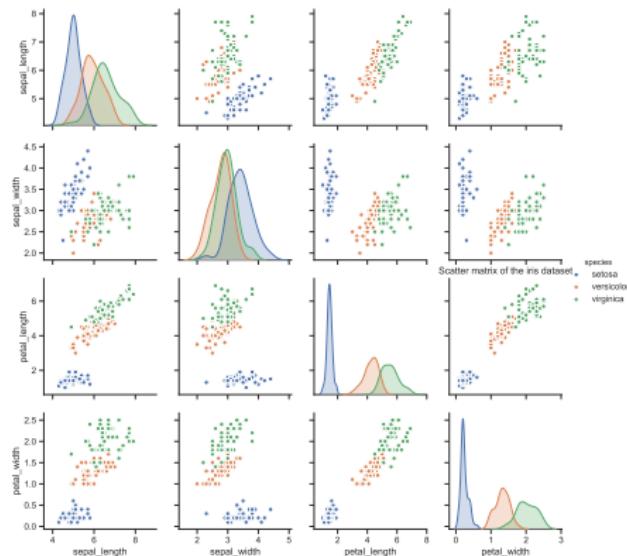


Figure – It seems that **petal width** is a parameter that separates the three species. On the contrary, **sepal width** is not able to do so.

## Titanic dataset : scatter matrix

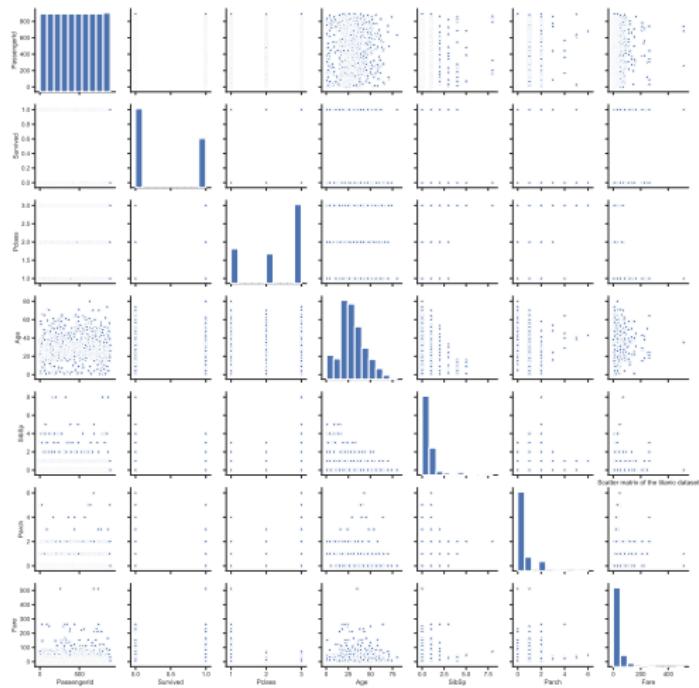


Figure – Titanic dataset scatter matrix

## K means clustering

- ▶ A famous unsupervised clustering method

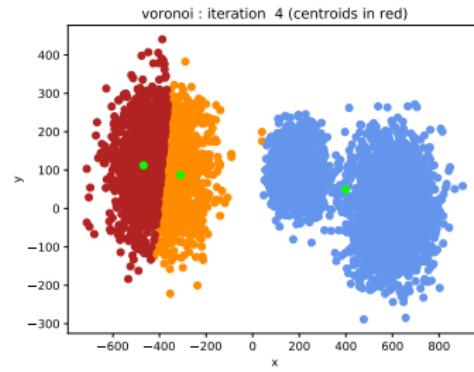


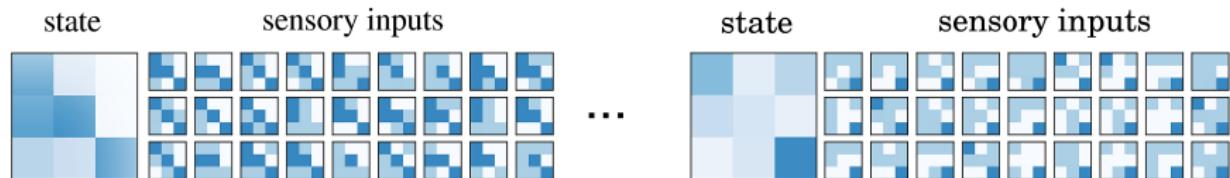
Figure – K means clustering

...

└ Clustering

└ Kmeans clustering

## Kmeans



**Figure –** Other example of kmeans clustering, this time in 9 dimensions  
[Le Hir et al., 2018]

## Kmeans : Expectation Maximisation algorithm

- ▶ Classical Machine Learning algorithm (EM)
- ▶ Blackboard
- ▶ What could be the drawbacks of this algorithm ?

...

└ Clustering

└ Kmeans clustering

## Kmeans clustering

Exercice 11: **Implementing kmeans**

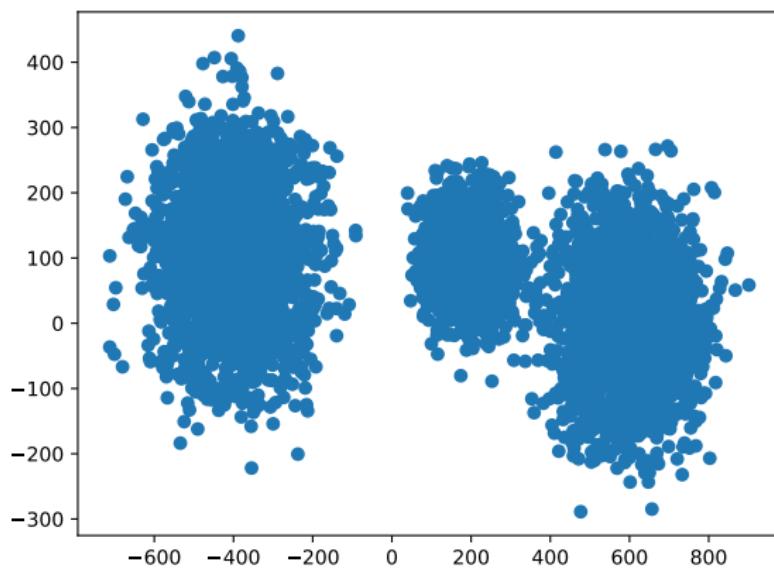


Figure 11: Data points to cluster

## Kmeans clustering

### Exercice 11 : Implementing kmeans cd ./kmeans

- ▶ Modify the `k_means.py` file so that it performs the kmeans algorithm.
- ▶ There are **two mistake series :**
  - ▶ line 64
  - ▶ around line 84

you will need to fix them.

You should obtain something like this :

...

└ Clustering

└ Kmeans clustering

# Kmeans

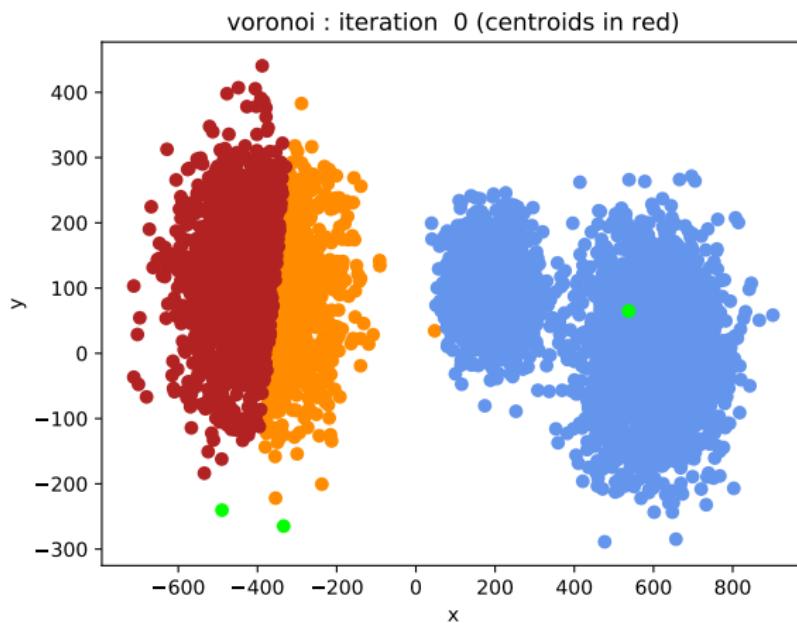


Figure – Voronoi 0th iteration

...

└ Clustering

└ Kmeans clustering

# Kmeans

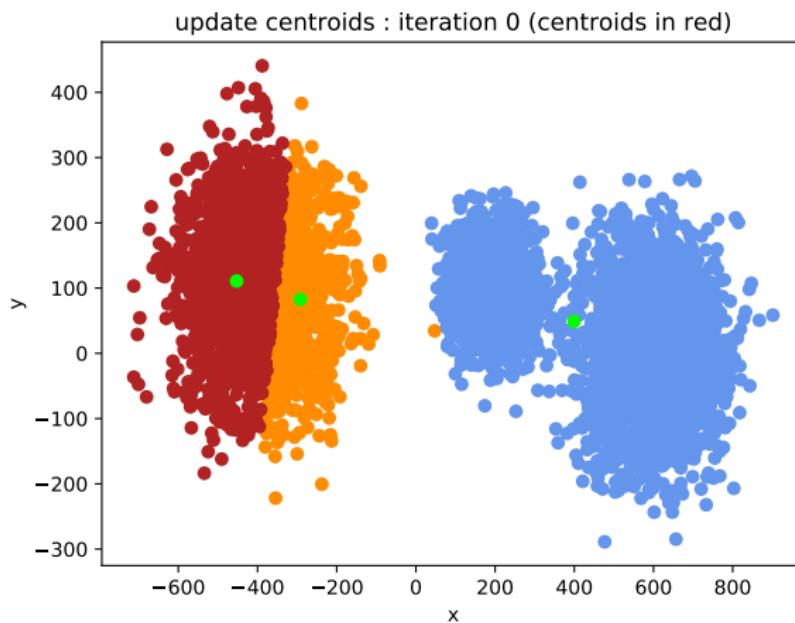


Figure – Centroids 0th iteration

...

└ Clustering

└ Kmeans clustering

## Kmeans

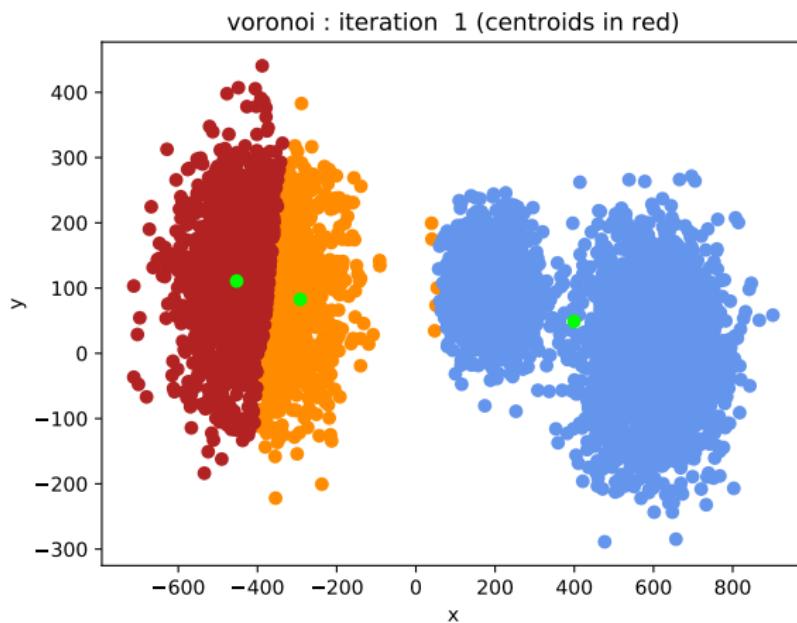


Figure – Voronoi 1st iteration

...

└ Clustering

└ Kmeans clustering

# Kmeans

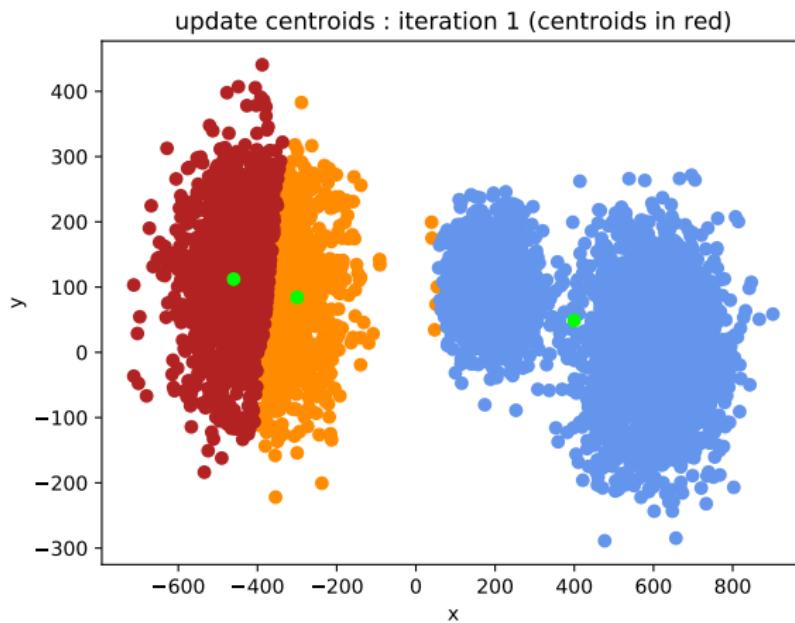


Figure – Centroids 1st iteration

...

└ Clustering

  └ Kmeans clustering

## Kmeans and initialization

Note that when launching the algorithm several times, the result may differ.

...

└ Clustering

  └ Kmeans clustering

# Sklearn

Exercice 12: Perform the kmeans algorithm using **sklearn**.

...

└ Clustering

  └ Similarities

## Similarities

- ▶ The kmeans were based on a notion of **distance between points**

...

└ Clustering

└ Similarities

## Similarities

- ▶ The kmeans were based on a notion of **distance between points**
- ▶ But sometimes you do not have access to a distance between the points.

...

└ Clustering

└ Similarities

## Similarities

- ▶ The kmeans were based on a notion of **distance between points**
- ▶ But sometimes you do not have access to a distance between the points.
- ▶ You might need to work with something that is more general, for instance a **similarity**.

...

└ Clustering

└ Similarities

## Similarities

- ▶ When working with distances, two points that "look the same" should be separated by a **small distance** .
- ▶ When working with a similarity, two points that "look the same" should have a **high similarity**.

...

└ Clustering

└ Similarities

## Example of similarity : adjacency

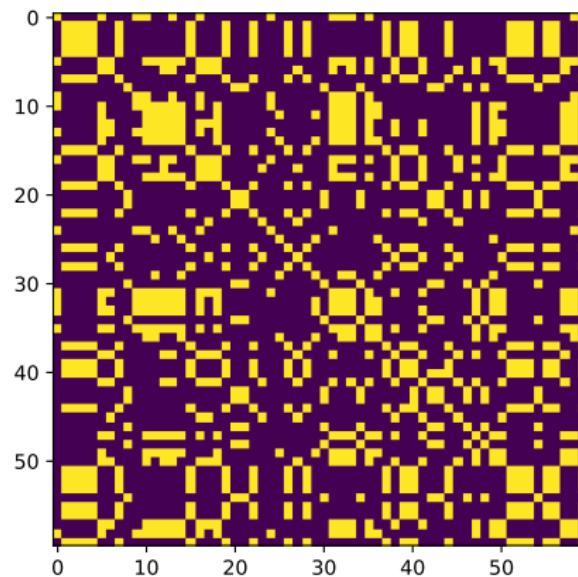
- ▶ An example of similarity is the relationship of **adjacency**.
- ▶ If  $i$  and  $j$  are related by an edge,  $S_{ij} = 1$ .
- ▶ Otherwise  $S_{ij} = 0$ .

...

└ Clustering

└ Similarities

## Adjacency matrix



...

└ Clustering

└ Similarities

## Similarities

Differences between similarities and distances :

- ▶ A similarity  $S$  is not always symmetrical.

...

└ Clustering

└ Similarities

## Similarities

Differences between similarities and distances :

- ▶ A similarity  $S$  is not always symmetrical.
- ▶ Indeed, in a **directed graph**, having a directed edge between  $i$  and  $j$  does not mean that we have an edge between  $j$  and  $i$ .

...

└ Clustering

└ Similarities

## Similarities

Differences between similarities and distances :

- ▶ A similarity  $S$  is not always symmetrical.
- ▶ Indeed, in a **directed graph**, having a directed edge between  $i$  and  $j$  does not mean that we have an edge between  $j$  and  $i$ .
- ▶  $S_{ij} = 0$  does not mean that  $i = j$ , it is rather the contrary.

...

└ Clustering

└ Similarities

## Similarities

- ▶ A similarity is a more general notion than a distance. Given a distance between two points, we can deduce a similarity.

...

└ Clustering

└ Similarities

## Similarities

- ▶ A similarity is a more general notion than a distance. Given a similarity between two points, we can deduce a similarity.
- ▶ For instance this way, if  $d_{ij}$  is the distance between  $i$  and  $j$  :

$$S_{ij} = \exp(-d_{ij}) \quad (18)$$

...

└ Clustering

└ Spectral Clustering

## Spectral Clustering

- ▶ A clustering method that works with similarities
- ▶ It performs a low dimensional embedding of the similarity matrix, followed by a Kmeans

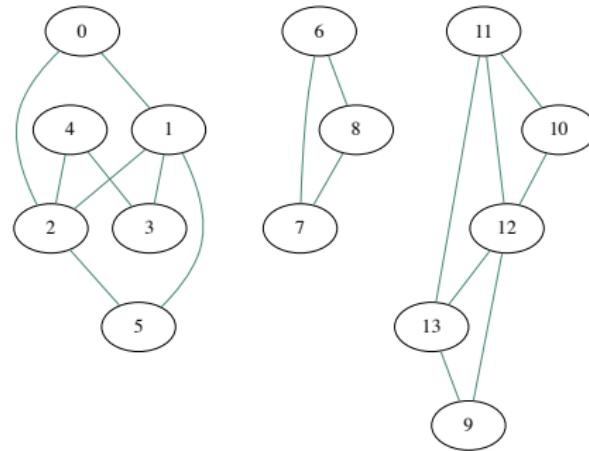
...

└ Clustering

└ Spectral Clustering

## Exercise

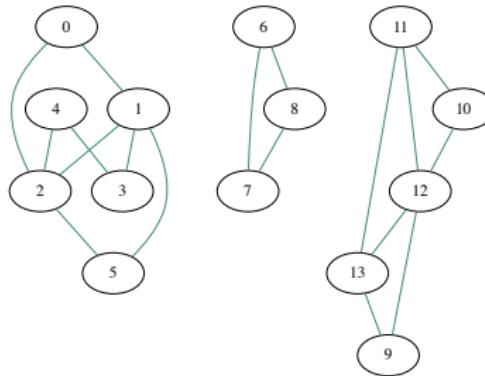
We will perform Spectral Clustering on this graph :



...

└ Clustering

└ Spectral Clustering



Please `cd spectral_clustering` and use `vanilla_spectral_clustering.py` in order to apply spectral clustering. You first need to input the right **affinity matrix** or **similarity matrix** and then use the **sklearn** library. You also need to **tune the number of clusters**. doc : check the **sklearn** page for Spectral Clustering.

...

└ Clustering

  └ Spectral Clustering

## Spectral clustering

Can you guess some drawbacks of the method ?

...

└ Clustering

└ Spectral Clustering

## Spectral clustering

Can you guess some drawbacks of the method ?

- ▶ Need to provide the number of clusters.
- ▶ Not adapted to a large number of clusters.
- ▶ kmeans step : so depends on a random initialization.

...

└ Clustering

  └ Spectral Clustering

## Heuristic

- ▶ We would like a criterion in order to justify the number of clusters used.

...

└ Clustering

└ Spectral Clustering

Normalized cut : a measurement of the quality of a clustering

- ▶ The **cut of a cluster** is the number of outside connections (connections with other clusters).
- ▶ The **degree** of a node is its number of adjacent edges
- ▶ The **degree of a cluster** is the sum of the degrees of its nodes.
- ▶ The **normalized cut** of a clustering is :

$$NCut(\mathcal{C}) = \sum_{k=1}^K \frac{Cut(C_k, V \setminus C_k)}{d_{C_k}} \quad (19)$$

...

└ Clustering

  └ Spectral Clustering

## Normalization

- ▶ The normalization is useful in order to take the **weight** (degree) of a cluster into account.

...

└ Clustering

  └ Spectral Clustering

## Normalized cut and clustering

Let's see how the normalized cut can help us choose the right number of clusters (backboard).

...

└ Clustering

└ Spectral Clustering

## Heuristic

**Exercice 13:** Exercise : normalized but elbow Please use the criterion in the file **normalized\_cut.py** in order to guess the relevant number of clusters in order to process the data contained in **data/**

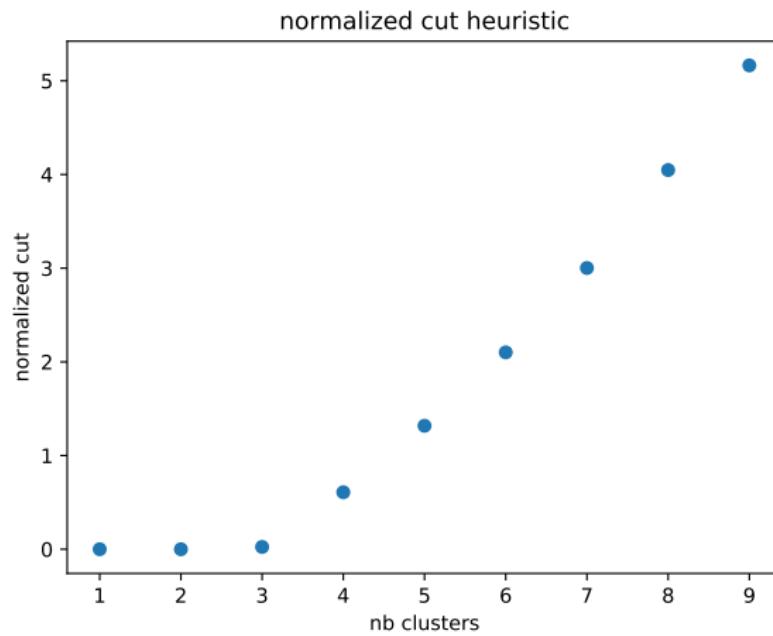


...

└ Clustering

└ Spectral Clustering

## Normalized cuts

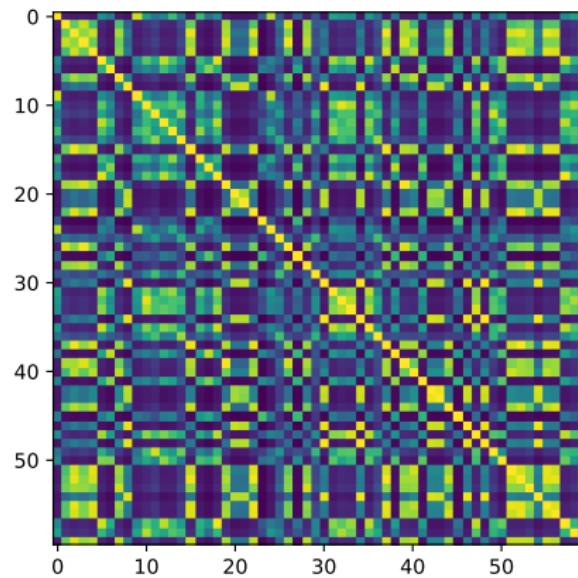


...

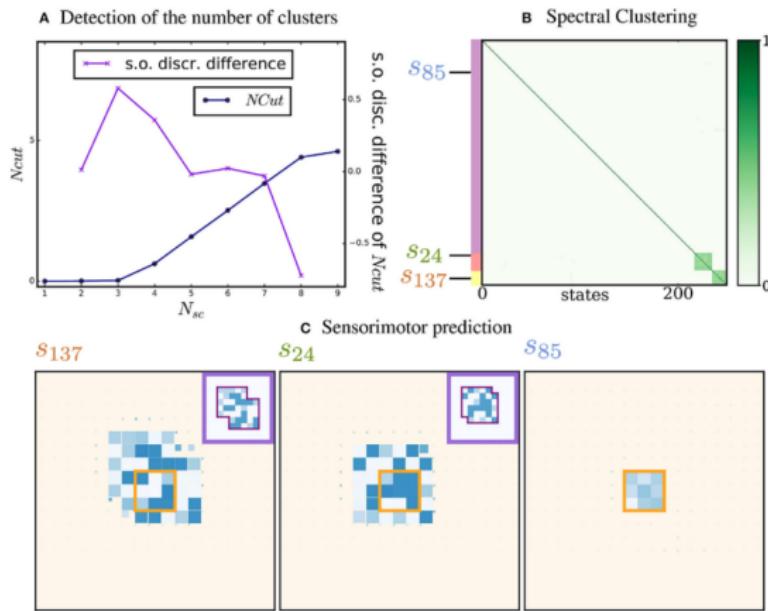
└ Clustering

└ Spectral Clustering

# Similarity



# Example



**c Sensorimotor prediction**



**Figure –** In a), the elbow method is used to choose the number of clusters. [Le Hir et al., 2018]

...

## Other methods to evaluate the quality of a clustering

- ▶ Stability of the result when launching the algorithm many times
- ▶ Separation of the clusters (the mean distance between pairs of centroids is large)
- ▶ Ratio inter / intra
- ▶ Silhouette coefficient

...

└ Additional considerations and conclusions

## Other interesting notions

- ▶ Agglomerative clustering (CHA : classification Hierarchique Ascendante)
- ▶ Xmeans : improvement of k means
- ▶ If you know more about probabilities or are curious :
  - ▶ Latent variables and variational learning
  - ▶ Auto Encoders
  - ▶ Boltzmann Machines

...

# Conclusion

Different kinds of problems exist :

- ▶ P problems where exact polynomial solutions exist (max matching)
- ▶ For other problems :
  - ▶ exhaustive search works but is too slow
  - ▶ to solve the problem a balance between rapidity and quality must be found.
- ▶ Evaluating the quality of a result is not an easy task.

...

└ Additional considerations and conclusions

# Project

- ▶ Description of the project

...

└ Additional considerations and conclusions

# Questions ?

...

└ Additional considerations and conclusions

## References

-  Le Hir, N., Sigaud, O., and Laflaqui  re, A. (2018).  
Identification of Invariant Sensorimotor Structures as a  
Prerequisite for the Discovery of Objects.  
*Frontiers in Robotics and AI*, 5(June) :1–14.