



# 3I026 - INTRODUCTION À L'INTELLIGENCE ARTIFICIELLE ET DATA SCIENCE

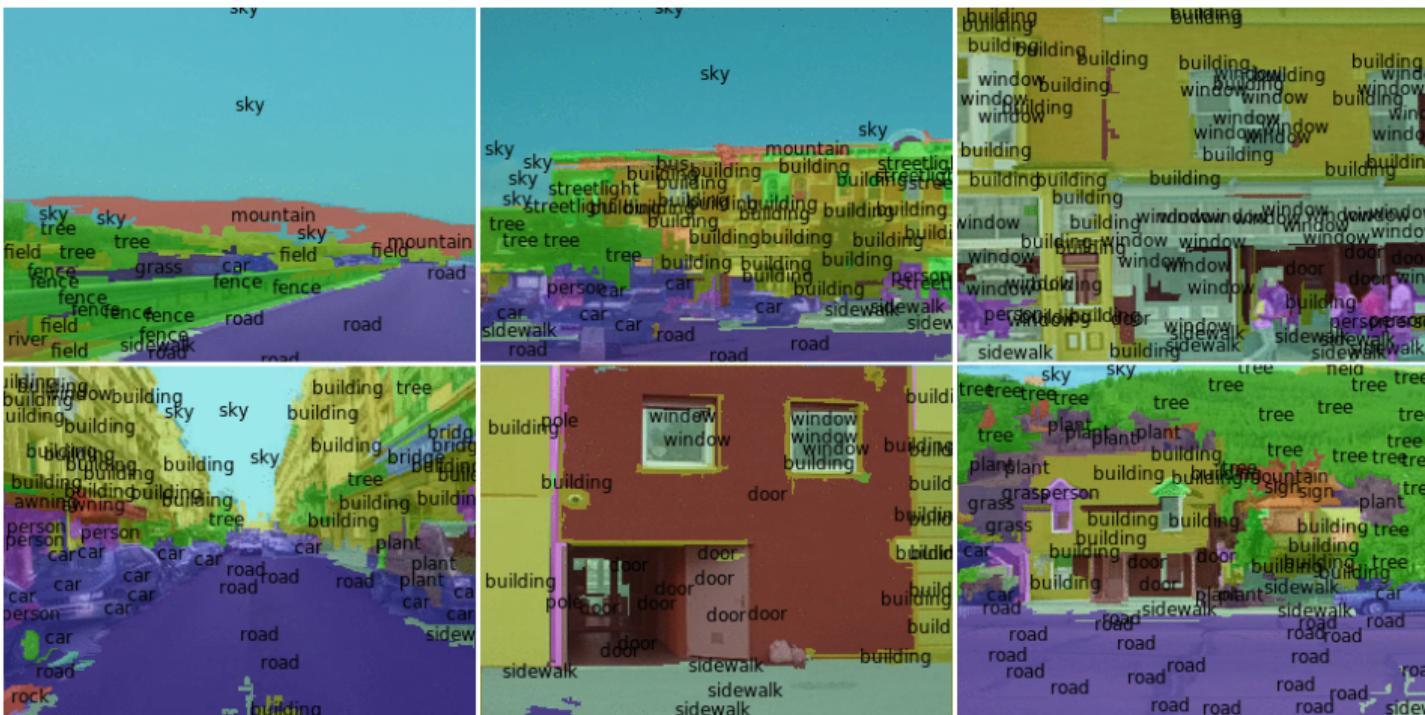
Vincent Guigue  
Christophe Marsala

Sorbonne Université

- 1 Comprendre les données en **ENTREE** du problème
- 2 Comprendre ce qui est attendu en **SORTIE** du problème



- 1 Comprendre les données en **ENTREE** du problème
- 2 Comprendre ce qui est attendu en **SORTIE** du problème



- 1 Comprendre les données en **ENTREE** du problème
- 2 Comprendre ce qui est attendu en **SORTIE** du problème



- 1 Comprendre les données en **ENTREE** du problème
- 2 Comprendre ce qui est attendu en **SORTIE** du problème



# Appréhension des données : Lecture, description

# Qu'est ce que c'est que la donnée ?

En général, un fichier...

Issu de :

- une base de données,
- un site web, une API,
- ...

... Qu'est ce que je veux en faire ?

- 1 L'importer,
- 2 comprendre ce qu'il y a dedans,
- 3 expliquer à d'autres personnes ce qu'il y a dedans,
- 4 mettre en forme pour pouvoir traiter ces données,
- 5 appliquer des (pré-)traitements sur ces données,
- 6 classer, scorer, analyser automatiquement

# Données : phase 1, acquisition

## ■ C'est quoi un CSV

Accueil Insertion Mise en page Formulaire Classeur Chercher dans la feuille

Presse-papiers Police Alignement Numérique Mise en forme conditionnelle Mettre sous forme de tableau Styles de cellule Cellules Modification

N28

	A	B	C	D	E	F	G	H	I	J
1	region	total_population_percent_who_percent_blue_percent_asia_percent_blue_per_capita_ij_modus_rent_modus_ape								
2	alabera	4799277	67	26	1	4	23890	501	38.1	
3	alaska	720356	63	3	5	6	32651	978	33.6	
4	aristana	6479793	57	6	3	30	25358	747	36.3	
5	arizona	2031818	74	15	1	2	22176	400	37.5	
6	arkansas	3785918	40	6	13	38	29527	1119	35.4	
7	colorado	5133239	70	6	9	22	31209	825	36.1	
8	connecticut	3583561	70	9	4	14	37892	890	40.2	
9	delaware	908446	65	21	3	8	29819	828	38.9	
10	district of col	639371	35	49	3	10	45290	1154	33.8	
11	florida	19081156	57	15	2	23	26236	838	41	
12	georgia	981277	55	30	3	9	25182	479	35.8	
13	hawaii	1278298	23	2	57	9	22025	120	38.5	
14	idaho	1583344	84	1	1	11	23548	607	34.9	
15	illinois	12848554	63	14	5	36	29666	759	36.8	
16	indiana	6514861	81	9	2	6	24635	577	37.1	
17	iowa	3062553	88	3	2	5	27027	534	38.1	
18	kansas	2883207	78	6	2	11	28922	551	36	
19	kentucky	4361333	86	8	1	3	23463	509	38.2	
20	louisiana	4461389	69	34	2	4	24521	410	36	
21	maine	1338320	94	1	1	3	36824	664	43.2	
22	maryland	5834299	54	29	6	8	36354	1034	38	
23	massachusetts	6605058	76	6	6	10	35763	936	39.2	
24	michigan	988095	76	14	3	5	25881	623	39.1	
25	minnesota	5347740	83	5	4	5	30913	734	37.6	
26	mississippi	2911212	50	37	1	3	29613	510	36.2	
27	missouri	6007382	81	11	2	4	25599	549	36	
28	montana	998554	87	0	1	3	25373	577	39.9	
29	nebraska	1841625	82	4	2	9	26893	563	36.3	
30	nevada	2730066	53	8	7	27	26589	840	36.6	
31	new hampshir	1313971	92	1	2	3	33134	878	41.5	
32	new jersey	8832408	59	13	9	18	36027	1024	39.1	
33	new mexico	2644306	40	2	7	47	27253	859	36.7	
34	new york	16487053	58	14	8	18	32382	963	38.1	
35	north carolina	9651380	65	21	2	9	25284	602	37.6	
36	north dakota	6887981	88	1	1	2	23732	564	36.4	
37	ohio	115459590	81	12	2	3	26046	562	39	
38	oklahoma	3785742	68	7	2	9	24203	525	36.2	
39	oregon	3868721	78	2	4	12	26809	749	38.7	

Prêt

# Données : phase 1, acquisition

## ■ C'est quoi un CSV

Accueil Insertion Mise en page Formules G: Chercher dans la feuille

Presse-papiers Police Alignement Numérique Mettre sous forme de tableau Styles de cellule

N28 ✓ fx

	A	B	C	D	E	F	G	H	I	J
1	region	total_population	percent_whi	percent_bla	percent_asia	percent_hi	per_capita_gdp	median_rent	median_ape	
2	alabama	4799277	67	26	1	4	23890	501	38.1	
3	alaska	720316	63	3	5	6	32651	978	33.6	
4	arizona	6479793	57	4	3	30	25368	747	36.3	
5	arkansas	2018128	74	15	1	7	22176	400	37.5	
6	california	3789181	40	6	13	28	23027	1119	35.4	
7	colorado	5133232	70	6	9	22	31209	825	38.1	
8	connecticut	3583516	70	9	4	14	37892	890	40.2	
9	delaware	908446	65	21	3	8	29819	828	38.9	
10	district of col	639371	35	49	3	10	45290	1154	33.8	
11	florida	19091156	57	15	2	23	26236	838	41	
12	georgia	981477	59	30	3	9	25182	479	35.6	
13	hawaii	1378298	23	2	57	9	23025	120	38.5	
14	idaho	1583344	84	1	1	11	23548	607	34.9	
15	illinois	1284854	63	14	5	36	29666	759	36.8	
16	indiana	6514861	81	9	2	6	24635	577	37.1	
17	iowa	3062553	88	3	2	5	23027	534	38.1	
18	kansas	2883207	78	6	2	11	28922	551	36	
19	kentucky	4361333	86	8	1	3	23461	509	38.2	
20	louisiana	4561497	69	34	2	4	24522	410	36	
21	maine	1338320	94	1	1	3	36824	664	43.2	
22	maryland	5834299	29	6	8	36354	1034	38		
23	massachusetts	6605058	76	6	6	10	35763	936	39.2	
24	michigan	988059	76	14	3	5	25881	623	39.1	
25	minnesota	83	5	4	5	30913	734	37.6		
26	mississippi	29112	50	37	1	3	29612	510	36.2	
27	missouri	6007182	81	11	2	4	23599	549	36	
28	montana	988554	87	0	1	3	25373	577	39.9	
29	nevada	1841625	82	4	2	9	26893	563	36.3	
30	nevada	2730066	53	8	7	27	26589	840	36.6	
31	new hampshire	1339371	92	1	2	3	33134	878	41.5	
32	new jersey	8832406	59	13	9	18	36027	1024	39.1	
33	new mexico	2640306	40	2	7	47	37263	859	34.7	
34	new york	16487053	58	14	8	18	32382	963	38.1	
35	north carolina	9651380	65	21	2	9	25284	602	37.6	
36	north dakota	6887981	88	1	1	2	23732	564	36.4	
37	ohio	115459590	81	12	2	3	26046	562	39	
38	oklahoma	3785742	68	7	2	9	24203	525	36.2	
39	oregon	3868721	78	2	4	12	26809	749	38.7	

Demographic\_State.csv

Demographics\_State.csv UNREGISTERED

```

1 "region","total_population","percent_white","percent_black","percent_asian","percent_hispanic","per_capita_income","median_rent","median_ape"
2 "alabama",4799277,67,26,1,4,23680,501,38.1
3 "alaska",720316,63,3,5,6,32651,978,33.6
4 "arizona",6479783,57,4,3,38,25358,747,36.3
5 "arkansas",2933369,74,15,1,7,22178,480,37.5
6 "california",37659181,48,6,13,38,29527,1119,35.4
7 "colorado",5119329,70,4,3,21,31109,825,36.1
8 "connecticut",3583561,78,9,4,14,37892,888,40.2
9 "delaware",988446,65,21,3,8,29819,828,38.9
10 "district of columbia",639371,35,49,3,10,45290,1154,33.8
11 "florida",19091156,57,15,2,23,26236,838,41
12 "georgia",981477,59,30,3,9,25182,479,35.6
13 "hawaii",1376298,23,2,37,9,29305,120,38.3
14 "idaho",1583364,84,1,1,11,22568,607,34.9
15 "illinois",1284854,63,14,5,16,29666,759,36.8
16 "indiana",6514861,81,9,2,6,24635,577,37.1
17 "iowa",3062553,88,3,2,5,23027,534,38.1
18 "kansas",2883207,78,6,2,11,26929,551,36
19 "kentucky",4361333,86,8,1,3,23461,509,38.2
20 "louisiana",4561497,69,34,2,4,24522,410,36
21 "maine",1328320,94,1,1,1,28224,664,43.2
22 "maryland",5834299,54,29,6,8,36354,1034,38
23 "massachusetts",6605058,76,6,6,10,35763,936,39.2
24 "michigan",988059,76,14,5,16,29612,510,36.2
25 "minnesota",5347746,83,5,4,5,30913,734,37.6
26 "mississippi",5347746,83,5,4,5,30913,734,37.6
27 "missouri",6007182,81,11,2,4,25649,549,38
28 "montana",988554,87,8,1,3,25373,577,39.9
29 "nevada",1841625,82,4,2,9,26893,563,36.3
30 "nevada",2730066,53,8,7,27,26589,840,36.6
31 "new hampshire",1339371,92,1,2,3,33134,878,41.5
32 "new jersey",8832406,59,13,9,18,36027,1024,39.1
33 "new mexico",2640306,40,2,7,47,37263,859,34.7
34 "new york",16487053,58,14,8,18,32382,963,38.1
35 "north carolina",9651380,65,21,2,9,25284,602,37.6
36 "north dakota",6887981,88,1,1,2,23732,564,36.4
37 "ohio",115459590,81,12,2,3,26046,562,39
38 "oklahoma",3785742,68,7,2,9,24203,525,36.2
39 "oregon",3868721,78,2,4,12,26809,749,38.7

```

Line 1, Column 1 Tab Size: 4 Plain Text

## ■ Comment le faire rentrer dans mon programme ?

Lecture de fichier + liste  
python

pandas + read\_csv

numpy + load\_txt

J'ai fait rentrer les données dans mon programme...

Et je fais quoi ensuite ?

Et je fais quoi ensuite ?

- Fast-checking :

- dimension des structures de données
- entêtes des colonnes
- tracé de quelques colonnes
- print...

1	region	total_population	percent_white	percent_black	percent_asian
2	0 alabama	4799277	67	26	1
3	1 alaska	720316	63	3	5
4	2 arizona	6479703	57	4	3
5	3 arkansas	2933369	74	15	1
6	4 california	37659181	40	6	13
7					
8	percent_hispanic	per_capita_income	median_rent	median_age	
9	0 4	23680	501	38.1	
10	1 6	32651	978	33.6	
11	2 30	25358	747	36.3	
12	3 7	22170	480	37.5	
13	4 38	29527	1119	35.4	

```
1 RangelIndex: 51 entries , 0 to 50
2 Data columns (total 9 columns):
3   region           51 non-null object
4   total_population 51 non-null int64
5   percent_white    51 non-null int64
6   percent_black    51 non-null int64
7   percent_asian    51 non-null int64
8   percent_hispanic 51 non-null int64
9   per_capita_income 51 non-null int64
10  median_rent      51 non-null int64
11  median_age       51 non-null float64
12  dtypes: float64(1), int64(7), object(1)
```

... Mais c'est parfois plus compliqué

### Chargement de données de chiffres manuscrits :

```

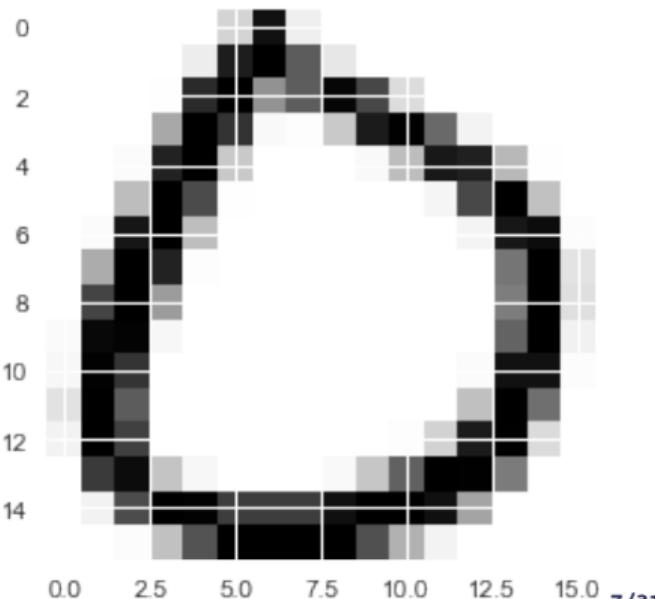
1 [0.      0.      0.      0.      0.      0.546   1.879   0.255   0.      0.
 0.      0.
2 0.      0.      0.      0.      0.      0.      0.      0.
 0.284  1.804  2.      1.42
3 0.336  0.      0.      0.      0.      0.      0.      0.      0.      0.
 0.      0.022
4 1.713  2.      1.027  1.408  1.947  1.56    0.462   0.      0.      0.
 0.      0.
5 0.      0.      0.      0.882  2.      1.665   0.098   0.031   0.64
 1.805  1.987  1.327
6 0.203  0.      0.      0.      0.      0.065   1.764   2.
 0.633  0.      0.
7 0.      0.086  0.744  1.833  1.778  0.78    0.008   0.      0.      0.
 0.744  2.
8 1.538  0.014  0.      0.      0.      0.      0.163   1.551   2.
 0.715  0.
9 0.      0.064  1.844  2.      0.737  0.      0.      0.      0.      0.
 0.      0.
10 0.173  1.841  1.913  0.052  0.      0.855   2.      1.765   0.025   0.
 0.      0.
11 0.      0.      0.      0.      0.      1.23    2.      0.383   0.      1.57
 2.      0.959
12 0.      0.      0.      0.      0.      0.      0.      0.      0.
 1.179  2.      0.434
13 0.078  1.939  1.976  0.135  0.      0.      0.      0.      0.      0.
 0.      0.
14 0.      1.36   1.998  0.221  0.133  1.983  1.656   0.007   0.      0.
 0.      0.
15 0.      0.      0.      0.      0.064  1.878  1.878   0.043   0.4
 1.998  1.415  0.
16 0.      0.      0.      0.      0.      0.      0.      0.723   2.
 1.273  0.
17 0.217  1.996  1.6     0.      0.      0.      0.      0.      0.      0.
 0.034  0.562
18 1.807  1.985  0.474  0.      0.      1.629  1.912  0.687  0.118  0.
 0.      0.

```

... Mais c'est parfois plus compliqué

### Chargement de données de chiffres manuscrits :

```
1 [0.      0.      0.      0.      0.      0.546   1.879   0.255   0.      0.  
0.      0.  
2 0.      0.      0.      0.      0.      0.      0.      0.  
0.284  1.804  2.      1.42  
3 0.336  0.      0.      0.      0.      0.      0.      0.      0.  
0.      0.022  
4 1.713  2.      1.027  1.408  1.947  1.56    0.462   0.      0.      0.  
0.      0.  
5 0.      0.      0.      0.882  2.      1.665   0.098   0.031   0.64  
1.805  1.987  1.327  
6 0.203  0.      0.      0.      0.      0.065   1.764   2.  
0.633  0.      0.  
7 0.      0.086  0.744  1.833  1.778  0.78    0.008   0.      0.      0.  
0.744  2.  
8 1.538  0.014  0.      0.      0.      0.      0.163   1.551   2.  
0.715  0.  
9 0.      0.064  1.844  2.      0.737  0.      0.      0.      0.      0.  
0.      0.  
10 0.173  1.841  1.913  0.052  0.      0.855   2.      1.765   0.025   0.  
0.      0.  
11 0.      0.      0.      0.      0.      1.23    2.      0.383   0.      1.57  
2.      0.959  
12 0.      0.      0.      0.      0.      0.      0.      0.      0.  
1.179  2.      0.434  
13 0.078  1.939  1.976  0.135  0.      0.      0.      0.      0.      0.  
0.      0.  
14 0.      1.36   1.998  0.221  0.133  1.983  1.656   0.007   0.      0.  
0.      0.  
15 0.      0.      0.      0.      0.064  1.878  1.878   0.043   0.4  
1.998  1.415  0.  
16 0.      0.      0.      0.      0.      0.      0.      0.723   2.  
1.273  0.  
17 0.217  1.996  1.6     0.      0.      0.      0.      0.      0.      0.  
0.034  0.562  
18 1.807  1.985  0.474  0.      0.      1.629  1.912  0.687  0.118  0.
```



Transmettre, résumer, manipuler, décrire l'information

- + trouver les informations utiles dans le bruit

Transmettre, résumer, manipuler, décrire l'information

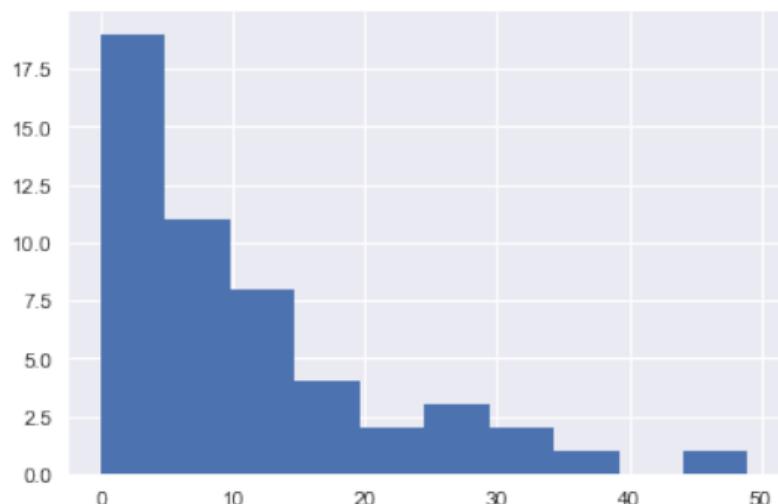
- + trouver les informations utiles dans le bruit

- Indicateurs statistiques

- Moyenne, écart-types, quantiles...
- Distributions (sur une ou plusieurs variables)

- Plots

- Histogrammes



Transmettre, résumer, manipuler, décrire l'information

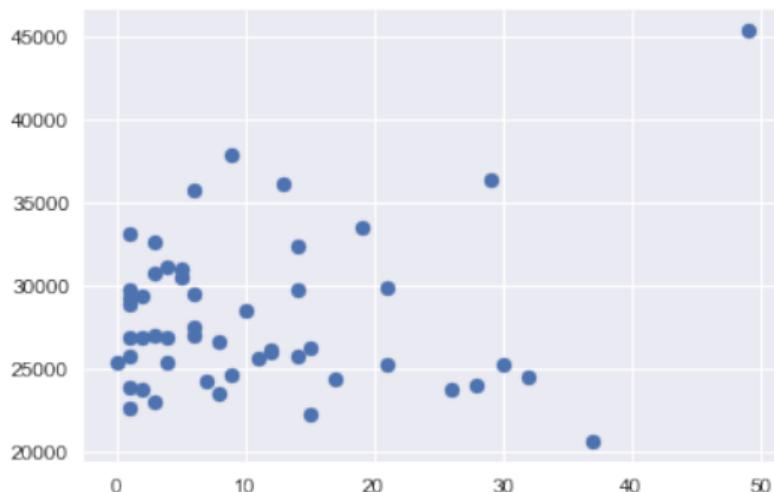
- + trouver les informations utiles dans le bruit

- Indicateurs statistiques

- Moyenne, écart-types, quantiles...
- Distributions (sur une ou plusieurs variables)

- Plots

- Histogrammes



## Description = stats

```
1      total_population  percent_white  percent_black  percent_asian  \
2 count      5.100000e+01      51.000000      51.000000      51.000000
3 mean       6.108561e+06     70.254902     10.823529      3.725490
4 std        6.904016e+06     16.116877     10.867761      5.355664
5 min        5.701340e+05     23.000000      0.000000      1.000000
6 25%        1.712494e+06     59.500000      3.000000      1.000000
7 50%        4.361333e+06     74.000000      7.000000      2.000000
8 75%        6.712318e+06     82.500000     14.500000      4.000000
9 max        3.765918e+07     94.000000     49.000000     37.000000
10
11      percent_hispanic  per_capita_income  median_rent  median_age
12 count      51.000000      51.000000      51.000000      51.000000
13 mean       10.803922    28053.803922    719.490196     37.639216
14 std        9.996038     4659.378182    189.820375     2.352367
15 min        1.000000     20618.000000    448.000000     29.600000
16 25%        4.500000     24908.500000    566.000000     36.300000
17 50%        8.000000     26824.000000    664.000000     37.600000
18 75%        12.500000    30144.000000    839.000000     38.950000
19 max        47.000000    45290.000000   1220.000000    43.200000
```

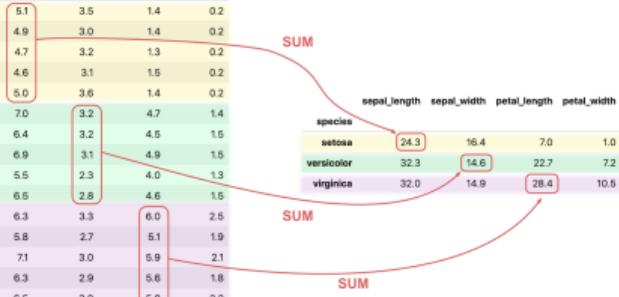
Comprendre et transmettre : le compromis entre la complétude et la compression des informations

# Manipuler les données : deux bibliothèques principales

pandas = philosophie BD

- Colonnes nommées
- Indexation, re-indexation
- Sélection par groupby
- Super outil pour les dates

	species	sepal_length	sepal_width	petal_length	petal_width
0	setosa	5.1	3.5	1.4	0.2
1	setosa	4.9	3.0	1.4	0.2
2	setosa	4.7	3.2	1.3	0.2
3	setosa	4.6	3.1	1.5	0.2
4	setosa	5.0	3.6	1.4	0.2
50	versicolor	7.0	3.2	4.7	1.4
51	versicolor	6.4	3.2	4.5	1.5
52	versicolor	6.9	3.1	4.9	1.5
53	versicolor	5.5	2.3	4.0	1.3
54	versicolor	6.5	2.8	4.6	1.5
100	virginica	6.3	3.3	6.0	2.5
101	virginica	5.8	2.7	5.1	1.9
102	virginica	7.1	3.0	5.9	2.1
103	virginica	6.3	2.9	5.6	1.8
104	virginica	6.5	3.0	5.8	2.2



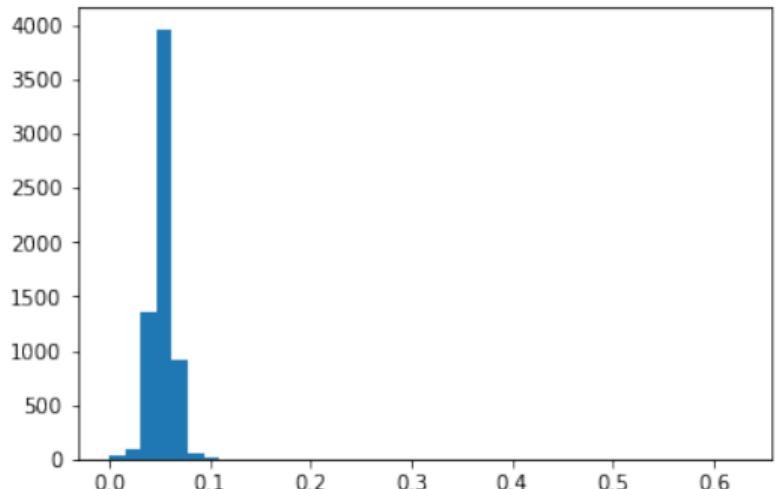
species	sepal_length	sepal_width	petal_length	petal_width
setosa	24.3	16.4	7.0	1.0
versicolor	32.3	14.6	22.7	7.2
virginica	32.0	14.9	28.4	10.5

numpy = philosophie math

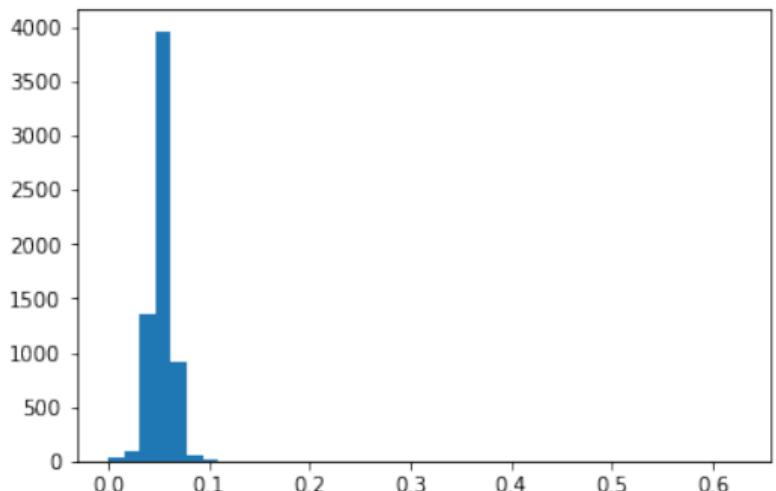
- Des matrices indexées en [ligne,col]
- Des opérateurs +, -, @...
- Proba (lois usuelles, vraisemblances, génération de nombres aléatoires)
- Algèbre linéaire (inverse, rang, résolution, ...)
- Sélection par np.where

+matplotlib dans les deux cas

**ex : prix au km des courses blablacar :**

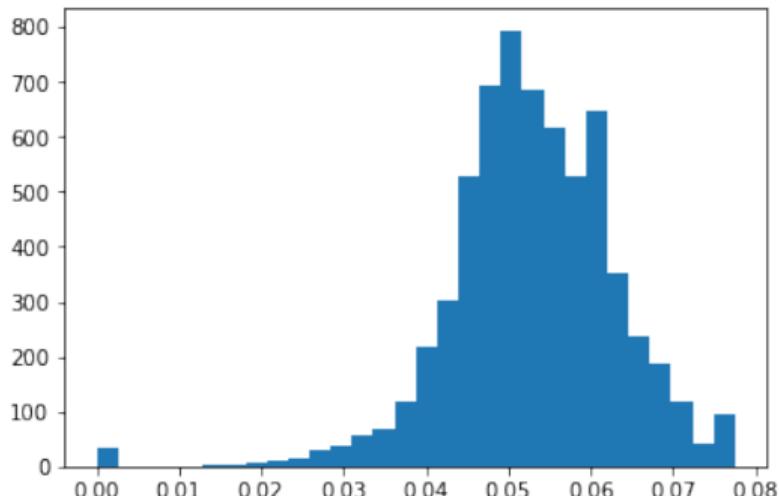


ex : prix au km des courses blablacar :



Avant suppression des valeurs anormalement hautes...

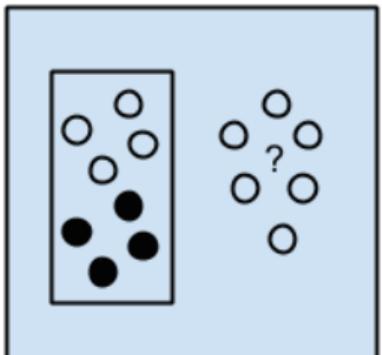
⇒ Mais comment je fais pour réaliser ça automatiquement ?



Et après

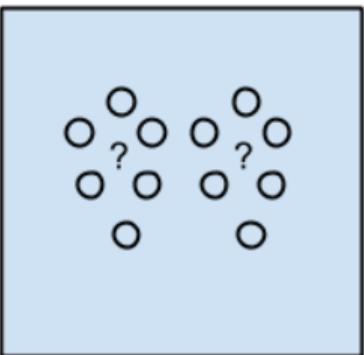
Rentrer dans les données :  
Mise en forme, modifications

Supervisé



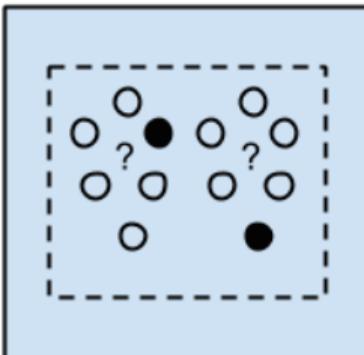
Supervised Learning  
Algorithms

Non-supervisé



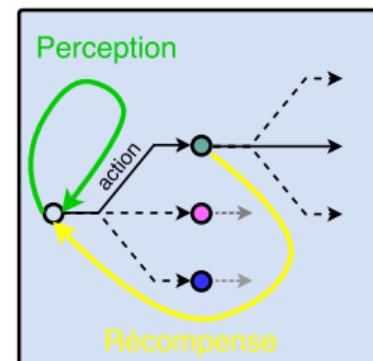
Unsupervised Learning  
Algorithms

Semi-supervisé



Semi-supervised  
Learning Algorithms

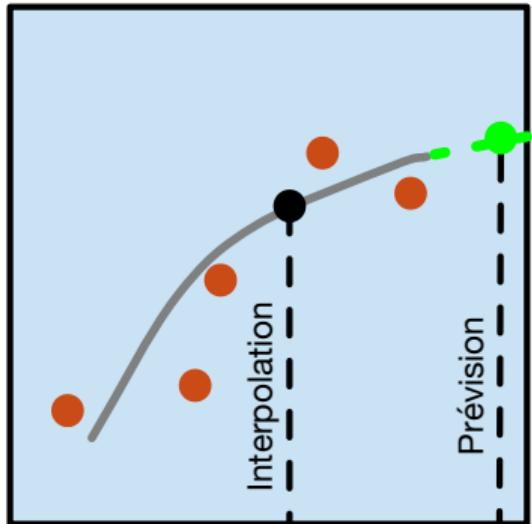
Renforcement



- Différents algorithmes... ... et différentes évaluations
- Différentes **données**, différents **coûts**...

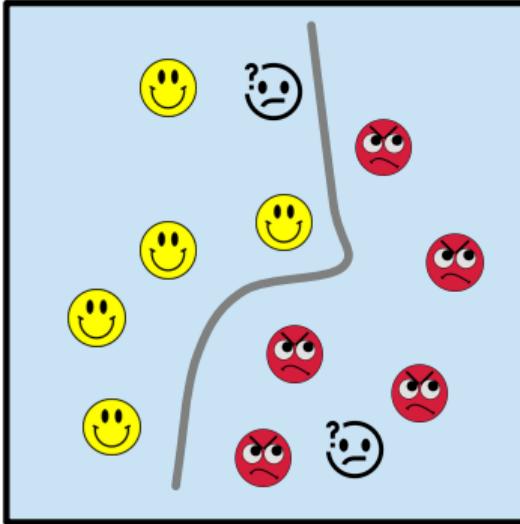
Et une nouvelle donne avec *Amazon Mechanical Turk*

## Régression



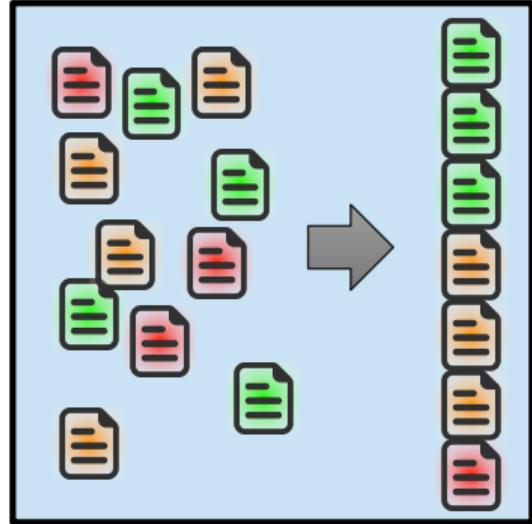
- Analyse de la qualité du vin
- Prédiction de ventes
- Maintenance prédictive

## Classification



- Analyse d'opinion
- Spam
- Reconnaissance d'écriture

## Ordonnancement



- Moteur de recherche
- Recommandation

Pour UN  $\mathbf{x}$  :

$$\text{data : } \mathbf{x} = [x_1, \dots, x_d], \text{ étiquette : } y \quad f(\mathbf{x}) = \sum_j x_j w_j \approx y$$

**Si**  $d = 1, y \in \mathbb{R}$

- Formaliser la matrice de données  $X$  + les indices
- Tracer les points  $(x, y)$
- Tracer  $f(x)$

Pour UN  $\mathbf{x}$  :

$$\text{data : } \mathbf{x} = [x_1, \dots, x_d], \text{ étiquette : } y \quad f(\mathbf{x}) = \sum_j x_j w_j \approx y$$

**Si  $d = 1, y \in \{1, -1\}$**

- Formaliser la matrice de données  $X$  + les indices
- Tracer les points  $(x, y)$
- Tracer  $f(x)$

Pour UN  $\mathbf{x}$  :

$$\text{data : } \mathbf{x} = [x_1, \dots, x_d], \text{ étiquette : } y \quad f(\mathbf{x}) = \sum_j x_j w_j \approx y$$

**Si  $d = 2, y \in \{-1, 1\}$**

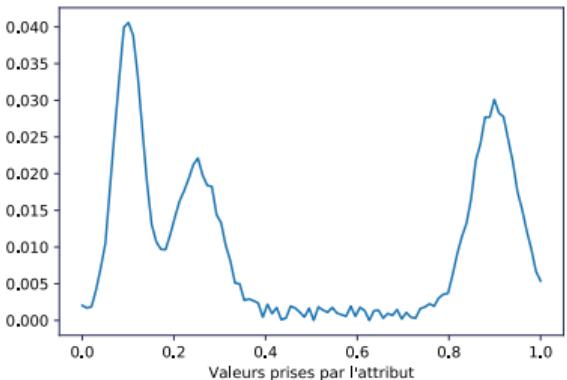
- Formaliser la matrice de données  $X$  + les indices
- Tracer les points  $(x, y)$
- Tracer  $f(x)$

$$\text{data : } \mathbf{x} = [x_1, \dots, x_d], \text{ étiquette : } y \quad f(\mathbf{x}) = \sum_j x_j w_j \approx y$$

- Si je prédis des affinités entre 1 et 5 étoiles...
  - Si je prédis des productions d'acier entre 1000T et 1100T
  - Si une colonne est dans les  $10^5$ ... Et une autre dans les  $10^{-5}$
- 1 Quelles fonctions d'évaluation, quelles astuces numériques ?
- 2 Attention à toujours pouvoir faire marche arrière !
- 3 C'est quoi une normalisation gaussienne (sur une variable) ?

$$\text{data : } \mathbf{x} = [x_1, \dots, x_d], \text{ étiquette : } y \quad f(\mathbf{x}) = \sum_j x_j w_j \approx y$$

- C'est quoi ?
- Pourquoi ça ne fait pas bon ménage avec un modèle linéaire ?
- Pourquoi il faut parfois passer au catégoriel même avec des données numériques ?



$$\text{data : } \mathbf{x} = [x_1, \dots, x_d], \text{ étiquette : } y \quad f(\mathbf{x}) = \sum_j x_j w_j \approx y$$

- Echelle de notation en 0 et 5 étoiles
- Prédiction = aimer un film  $m$  par rapport à votre historique  
Comment prendre en compte les avis négatifs ??

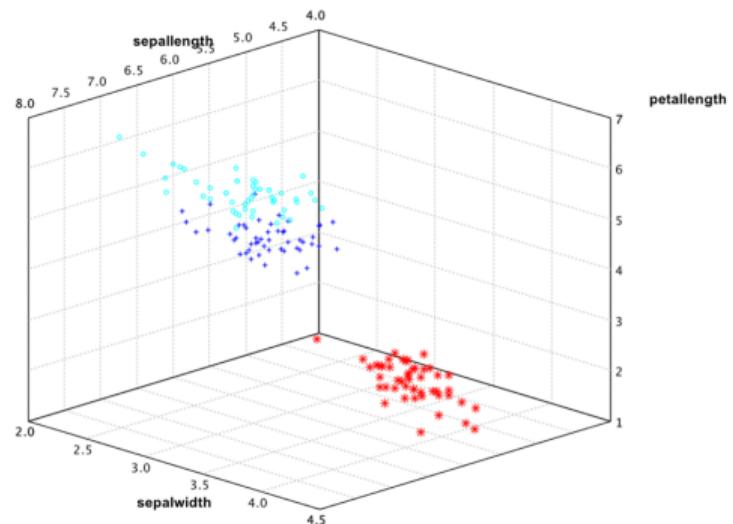
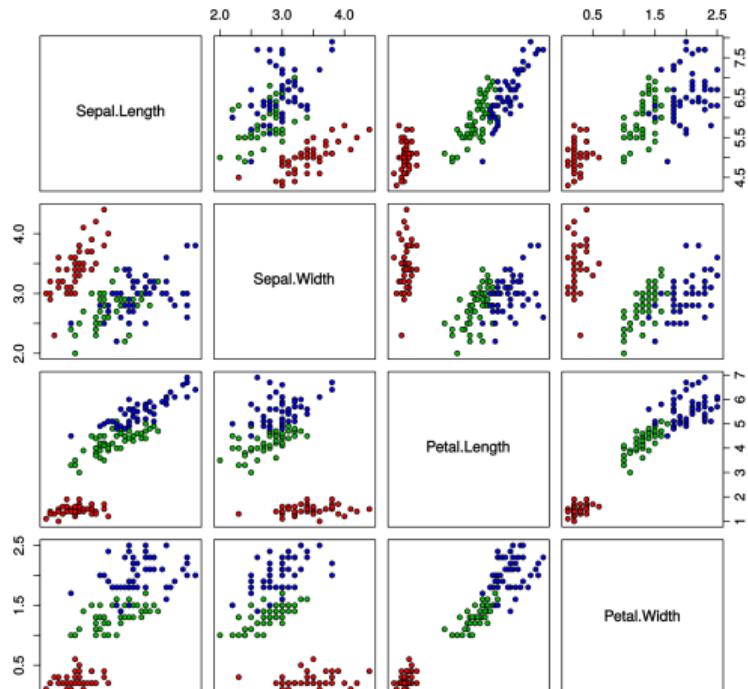
- C'est quoi une gaussienne ?
  - Pourquoi on aime normaliser selon une gaussienne ?
  - Quelles limites au modèle gaussien ?
- C'est quoi un produit scalaire ?
  - Pourquoi on s'en sert tout le temps ?
- C'est quoi un produit vectoriel ?
  - Pourquoi on ne s'en sert jamais ?
- C'est quoi un produit matriciel ?

# Outils mathématiques (2)

- Comment on recombine des colonnes ?
- Pourquoi ?

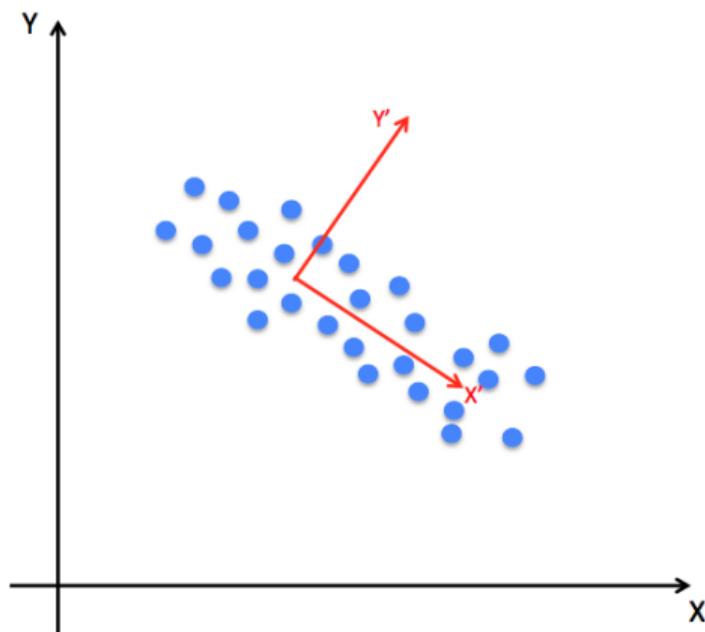
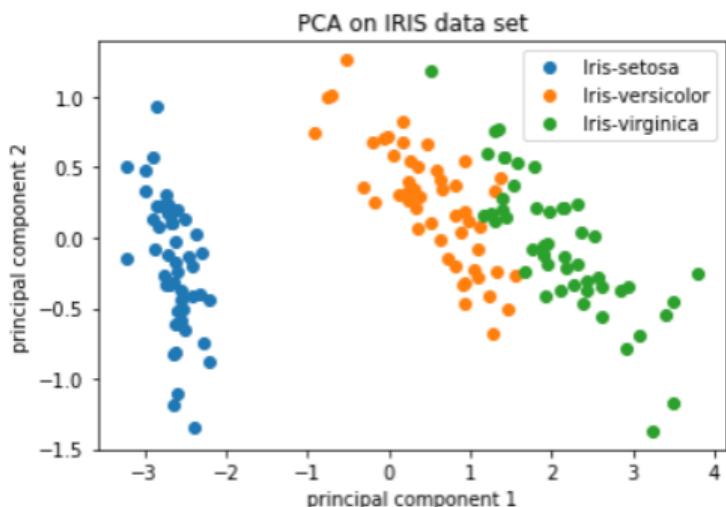
Base de données *Iris*

Iris Data (red=setosa,green=versicolor,blue=virginica)



- Comment on recombine des colonnes ?
- Pourquoi ?

Base de données *Iris*



## Information importante vs information redondante

- Qu'est ce que c'est qu'une corrélation ?
- Qu'est ce que c'est qu'une matrice de corrélation ?
- Qu'est ce que c'est qu'un coefficient de corrélation ?

- numpy
  - ambiguïté de la multiplication
- pandas
- matplotlib
  - scatter, plot
- Attention aux boucles
  - Additionner deux matrices, trier, comparer, compter les éléments...

Infos importantes pour les TME :

- Attentions aux quotas
- Bibliothèques

```
1 pip install paquet --user
2     --proxy=proxy.ufr-info-p6.jussieu.fr:3128
3 pip install scikit-learn --user
4     --proxy=proxy.ufr-info-p6.jussieu.fr:3128 --upgrade
```

- Scrapping
- accès aux API
  - `request + JSON`

K-plus proches voisins

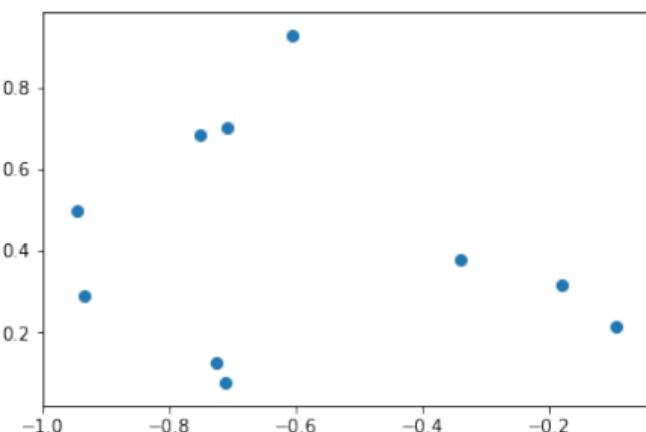
# Encore un algo !

## Idée du plus proche voisin

Lorsque je dois classer, évaluer, catégoriser un point... Je peux prendre la classe, la catégorie ou le score de son plus proche voisin !

- Seul besoin :

Disposer d'un fonction de calcul de la distance

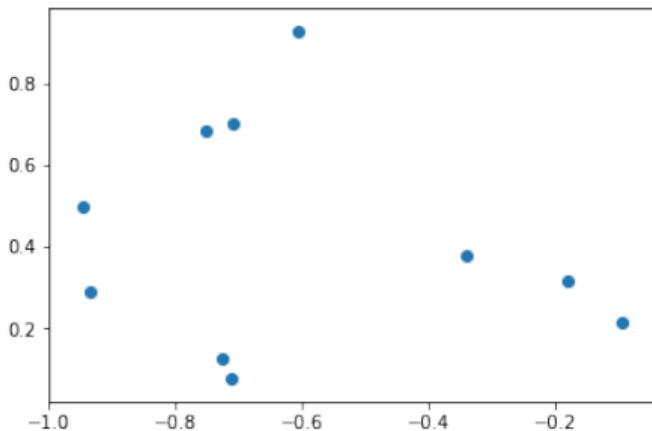


# Encore un algo !

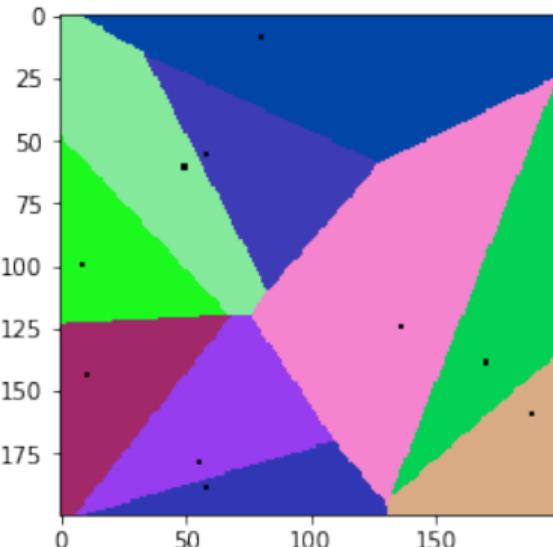
## Idée du plus proche voisin

Lorsque je dois classer, évaluer, catégoriser un point... Je peux prendre la classe, la catégorie ou le score de son plus proche voisin !

- Seul besoin :



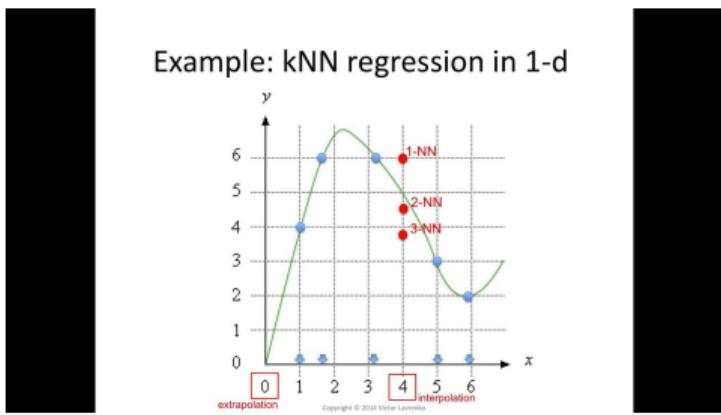
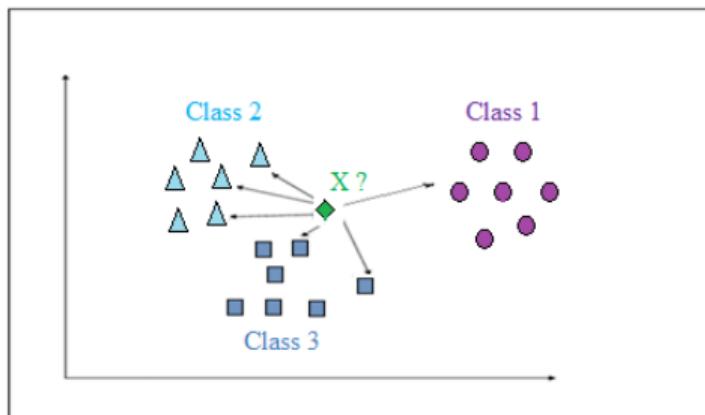
Disposer d'un fonction de calcul de la distance

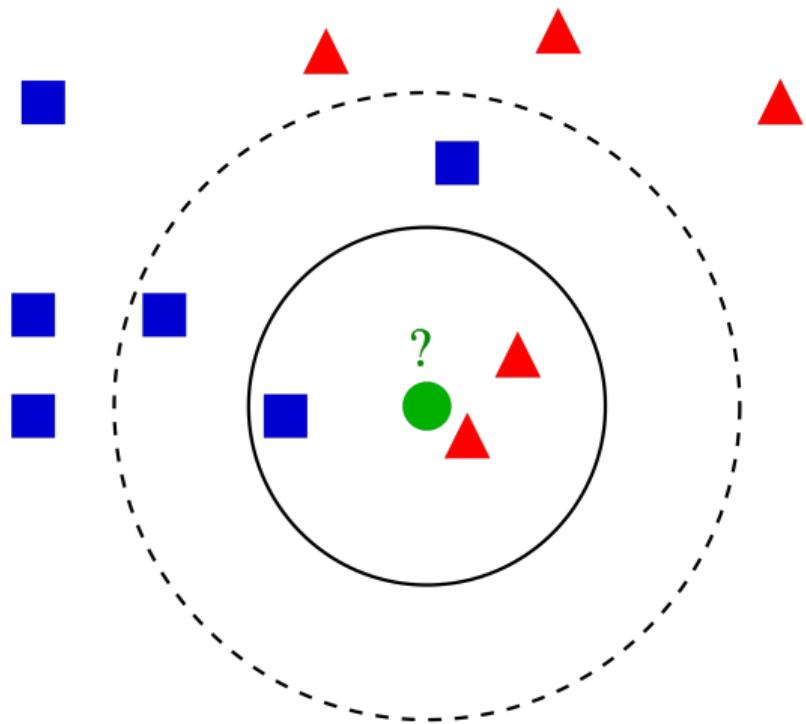


# Algorithme des kppv/knn

## Du plus proche voisin aux kppv (knn)

- Calcul des distances entre la cible et tous les points d'apprentissage
- Recherche des  $k$  plus proches voisins
- Agrégation d'une décision
  - moyenne (score)
  - vote (classification)
  - moyenne ou vote pondéré





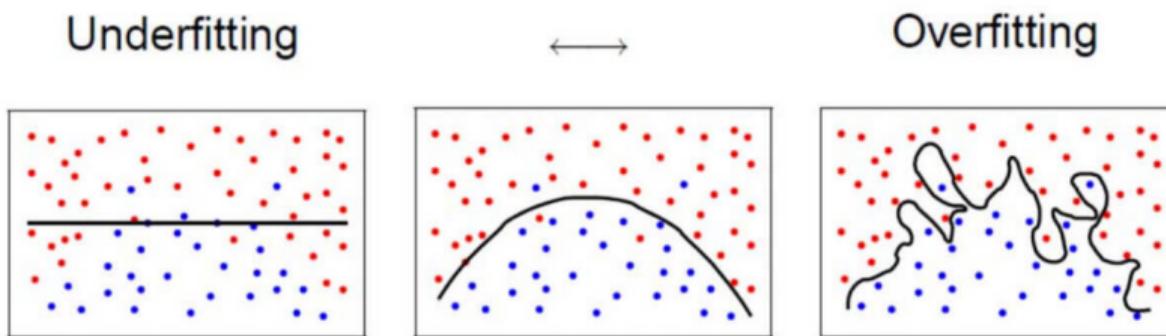
- Pour un point, la décision peut changer en fonction de  $k$
- Quid du cas  $k = 1$ 
  - Si le point  $x$  est nouveau
  - Si le point  $x$  est dans l'ensemble d'apprentissage

- **Evaluer** un modèle est aussi important que de l'**apprendre**
- **Evaluer** un modèle n'est pas trivial

Apprendre par cœur les données = sur-apprentissage (*overfitting*)

Evaluer un classifieur = statistique de performances sur **d'autres** données (tirées selon les mêmes paramètres)

Données **iid** : indépendantes et identiquement distribuées



## $k$ -ppv

### Apprentissage

Instantané... Il suffit de stocker les points d'apprentissage

### Inférence/prédiction sur $x$

- Calcul de la distance entre  $x$  et tous les points d'apprentissage
- Tri
- Agrégation (vote, moyenne,...)

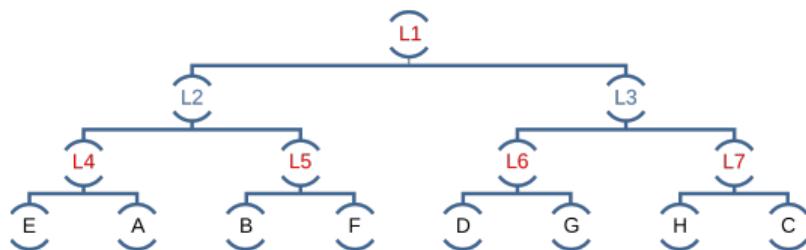
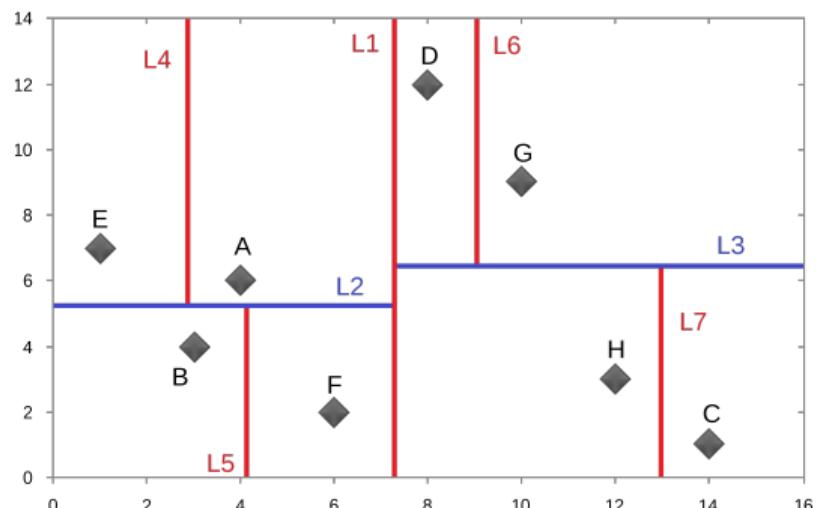
## Modèle linéaire

Il faut apprendre le vecteur  $w$  (cf cours 3)

Calcul de  $x \cdot w \approx$  instantané

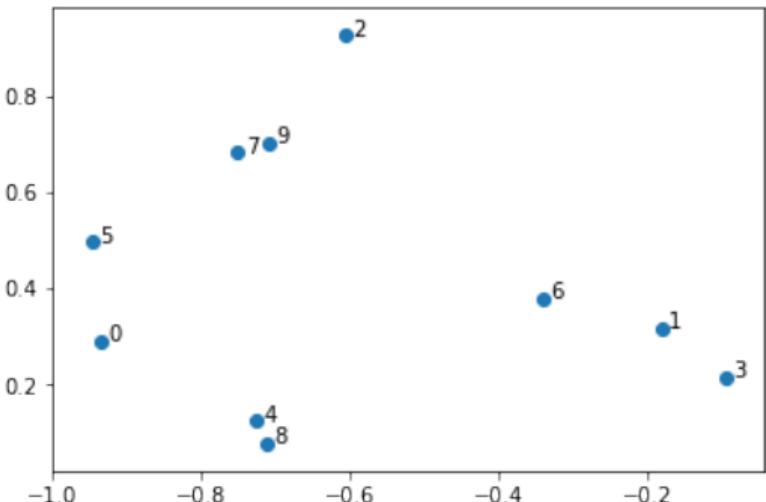
Comment sauver les knn ?  $\Rightarrow$  LSH / kd-tree

- local sensitive hashing
- kd tree

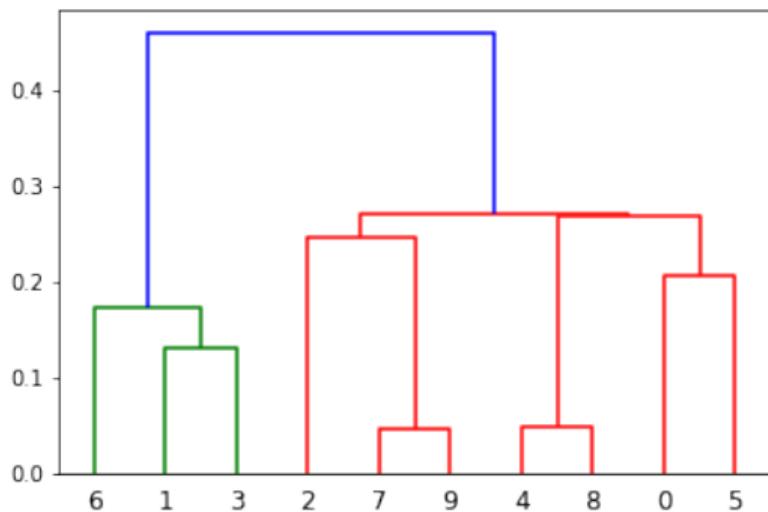
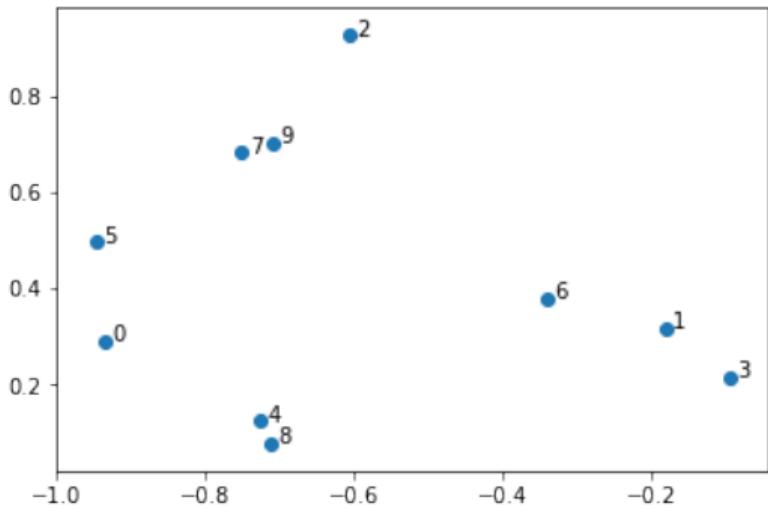


Ne calculer les distances QUE entre les voisins  $\Rightarrow$  Diviser l'espace

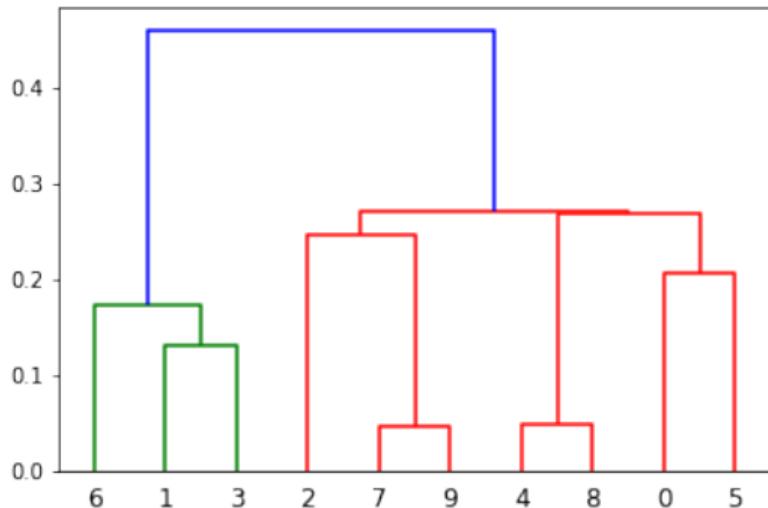
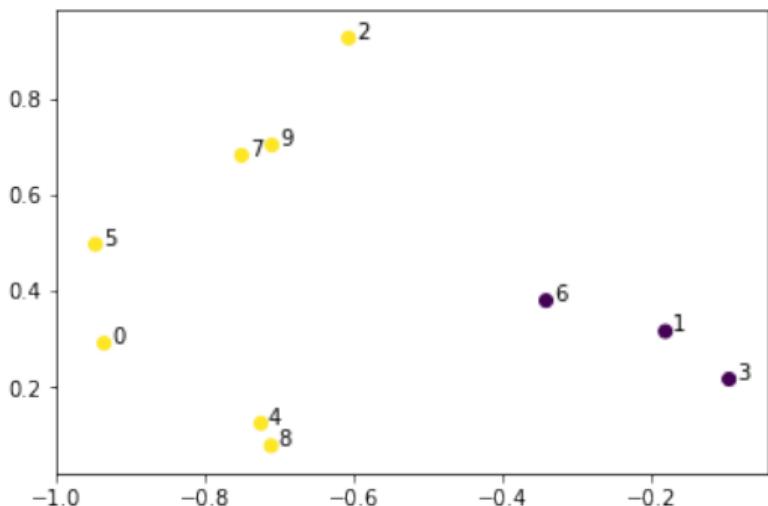
Un outil : le dendrogramme



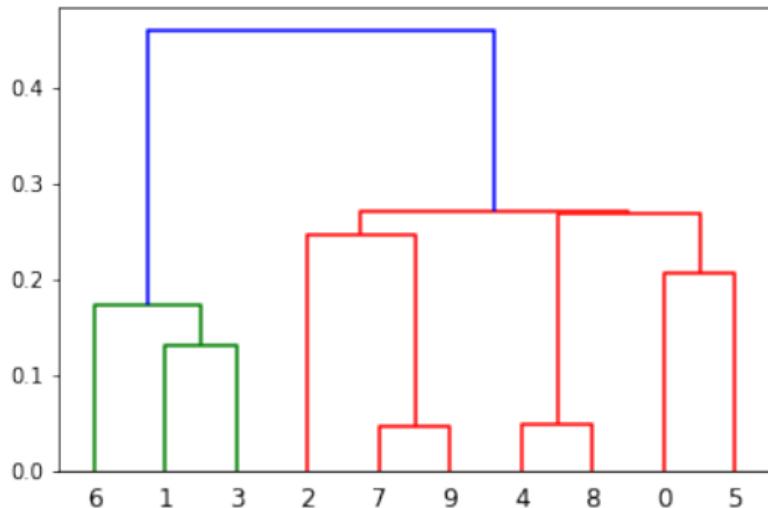
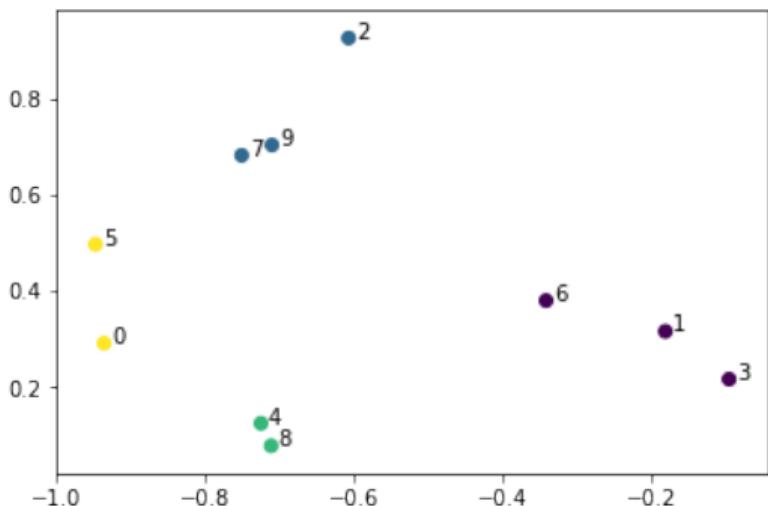
Un outil : le dendrogramme



Un outil : le dendrogramme

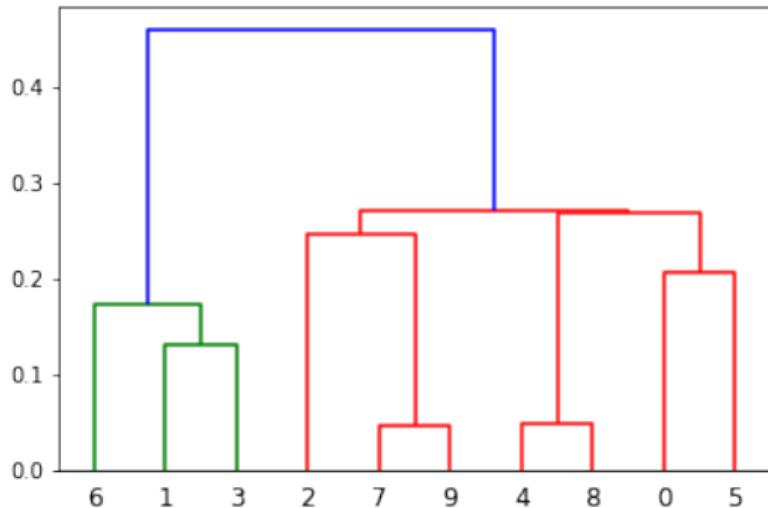
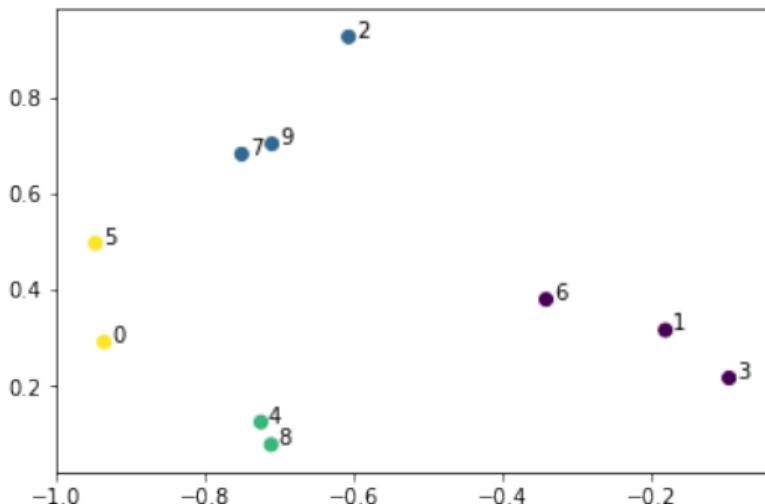


Un outil : le dendrogramme



# Et si je n'ai plus de supervision ?

Un outil : le dendrogramme



- + Très performant dans de nombreux cas
- + S'adapte à toutes les données en définissant de nouvelles distances
- Cher en temps de calcul

**Mêmes conclusions que pour les k-ppv !!!**