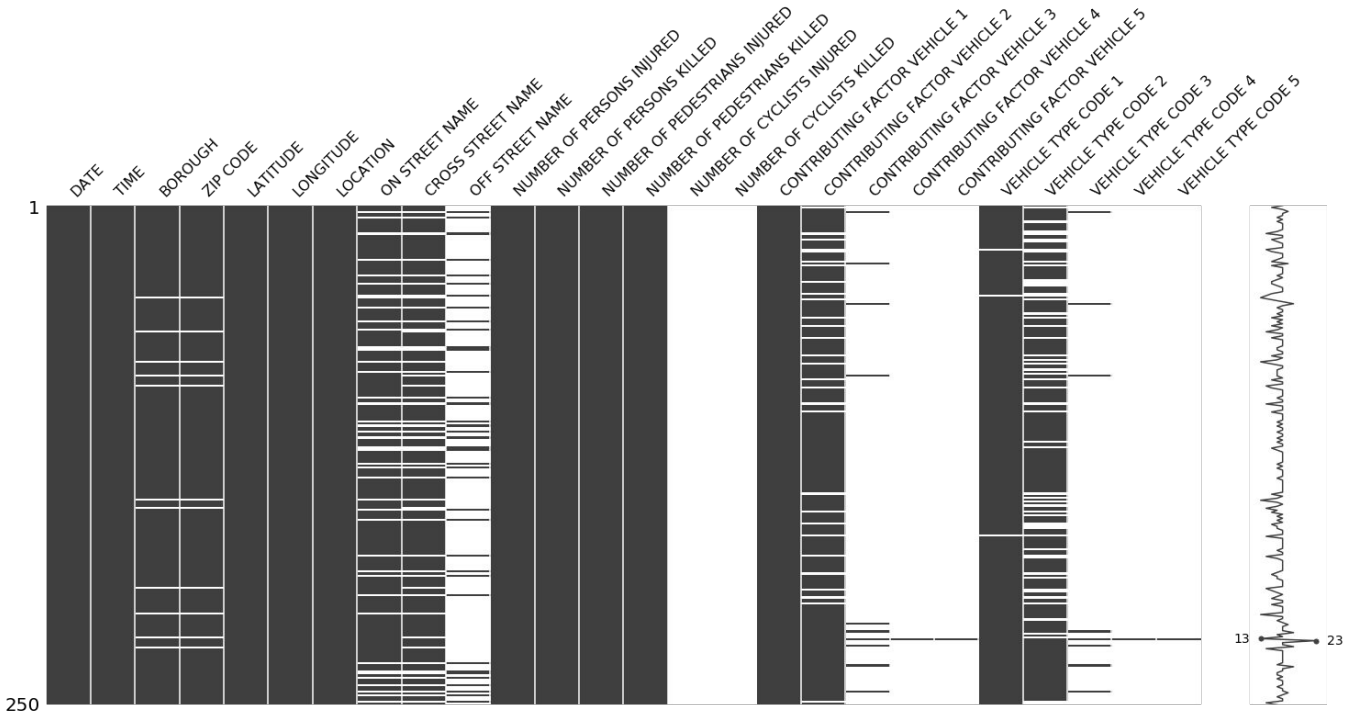
Machine Learning
& Intelligence Artificielle

Manuel Simoes
manuel.simoes@cpc-analytics.fr

- Valeurs Manquantes -

Les valeurs manquantes : module missingno



missingno pypi v0.4.2 | python 3.4+ | status stable | license MIT | doi 10.21105/joss.00547+

Messy datasets? Missing values? `missingno` provides a small toolset of flexible and easy-to-use missing data visualizations and utilities that allows you to get a quick visual summary of the completeness (or lack thereof) of your dataset. Just `pip install missingno` to get started.

https://github.com/ResidentMario/missingno

Visualisation des valeurs manquantes

Méthodes Pandas

- df.dropna()
- df.isna()
- df.isnull()
- df.fillna()

Les valeurs manquantes

Les valeurs manquantes

De nombreux algorithmes ne savent pas fonctionner avec des données manquantes.

- Affecter une valeur fixe (ex: moyenne, médiane...)
- Affecter une valeur provenant d'un modèle prédictif
- Éliminer les lignes (observations) incomplètes.
  Si le jeu de données est de taille conséquente et le taux de données bas
- Éliminer la colonne contenant les données manquantes.
  Si elle contient que peu d'information au regard du nombre d'observation

C'est aussi un processus empirique

essayer une approche, mesurer son efficacité et tenter d'améliorer le résultat, jusqu'à obtenir un résultat satisfaisant.

Attention

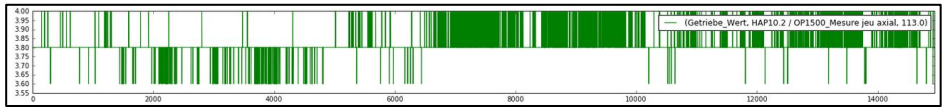
vous devez réfléchir à l'impact de vos modifications sur le fonctionnement de l'algorithme.

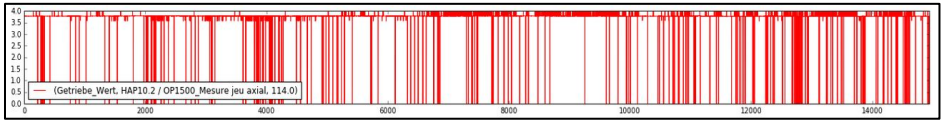
Scikit- Learn

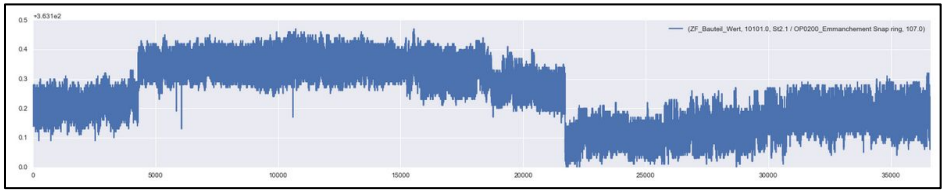
Doc https://scikit-learn.org/stable/modules/impute.html

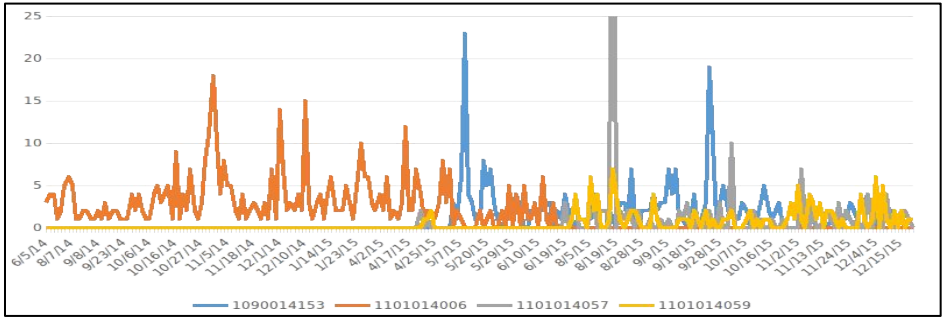
- Données Aberrantes -

Les valeurs et données aberrantes









Les données seront toujours vous étonner quand il s'agit de données aberrantes.

Cela peut aussi dépendre du sens de la colonne étudiée :

- Age négatif
- Erreur dans une étiquette
- …

Les données outlier

Une valeur aberrante (*outlier*) est une valeur extrême de la distribution d'une variable qui s'écarte considérablement des autres valeurs.

Deux options

- Il s'agit d'une erreur (capteur défectueux,...), dans ce cas nous pouvons la rejeter ou la corriger (valeur médiane, valeur moyenne...).

- Il s'agit d'une situation exceptionnelle mais bien réelle, qui doit être prise en compte lors de l'analyse des données (en fonction du problème donné, on peut aussi réduire le champ d'étude pour l'exclure). Exemple le salaire d'une personne très riche.

Quelques méthodes élaborées https://www.analyticsvidhya.com/blog/2019/02/outlier-detection-python-pyod/

An Awesome Tutorial to Learn Outlier Detection in Python using **PyOD Library**

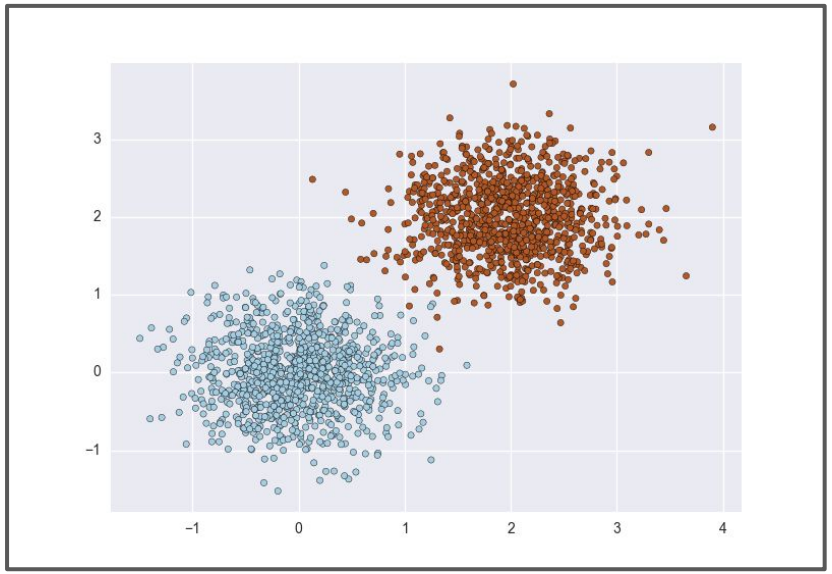
- Imputation des Données -

&

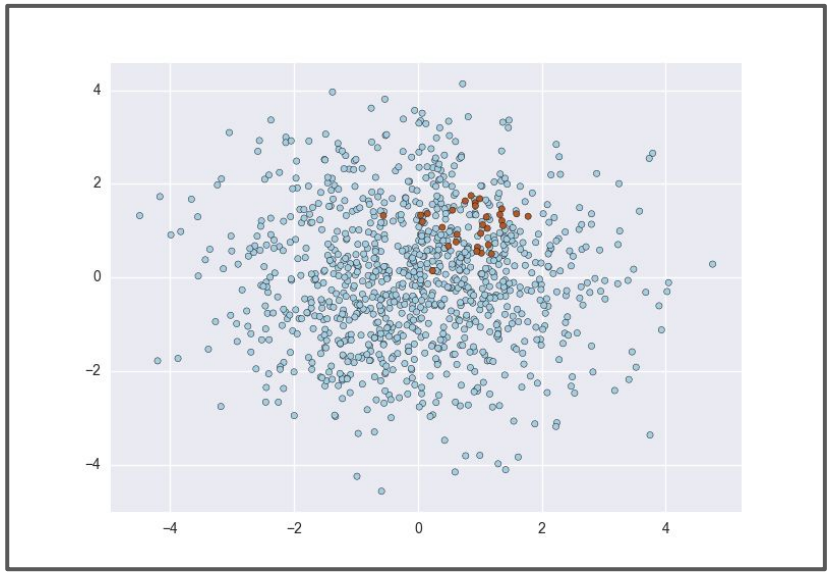
- Données non Balancées -

Imbalanced Data

La théorie



La vrai vie !



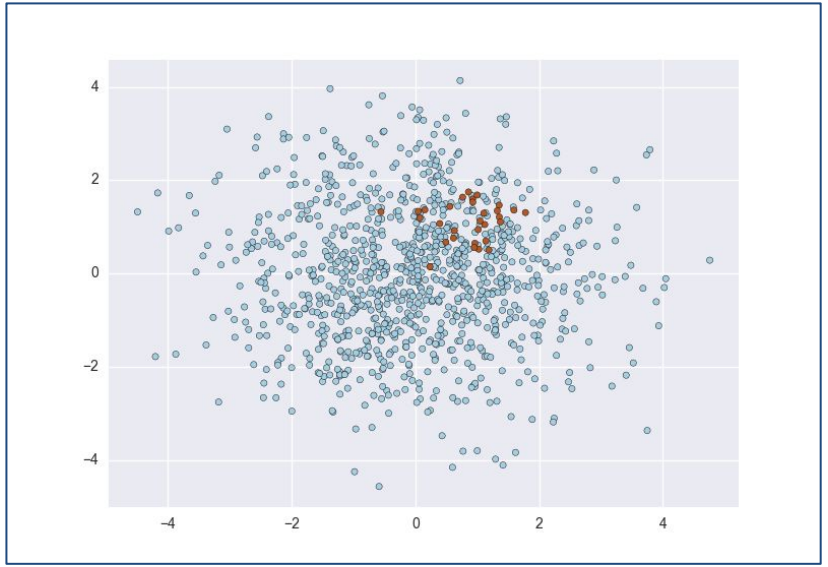
Une des classes est très majoritaire !! Le nombre de données de la classe minoritaire est en faible proportion.

Ici les problèmes c'est l'absence de données plutôt que l'absence de certaines features.

Imbalanced Data

Dans le cas d'une classification, il peut arriver qu'une ou plusieurs classes soient "surreprésentés" par rapport à d'autres. Dans ce cas, les données appartenant aux classes minoritaires risquent d'être ignorées, car leurs erreurs sur ces classes seront négligeables dans la moyenne.

Un travail sur les données est alors important.



Oversampling : Création de nouvelles données dans la classe minoritaire

Undersampling : Suppression de données dans la classe majoritaire.

Over-Sampling & Under-Sampling : combinaison des 2 méthodes précédentes

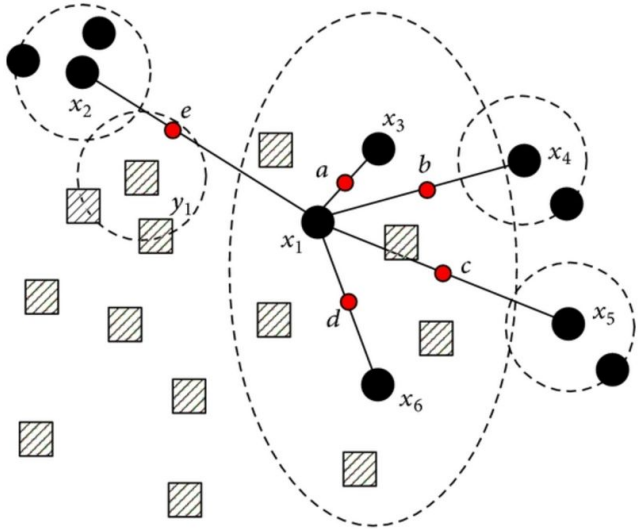
Librairie Python de fonctions **imbalanced-learn** : https://imbalanced-learn.readthedocs.io/en/stable/index.html

Imbalanced Data

https://imbalanced-learn.readthedocs.io

SMOTE (**S**ynthetic **M**inority **O**ver-sampling **TE**chnique) l'algorithme parcours toutes les observations de la classe minoritaire, cherche ses k plus proches voisins puis "synthétise"/définie aléatoirement de nouvelles données entre ces deux points.

ADASYN (**Ada**ptive **Syn**thetic Sampling Method for Imbalanced Data) est une version améliorée de SMOTE qui introduit une variation sur les données synthétiques.

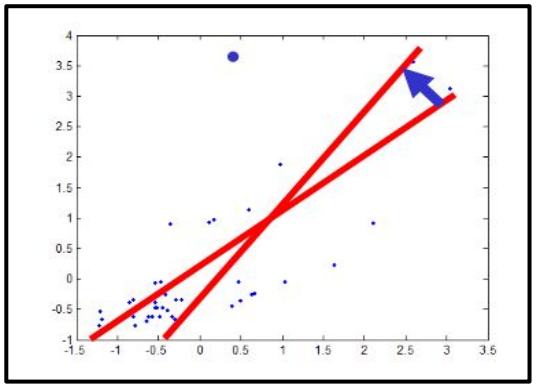


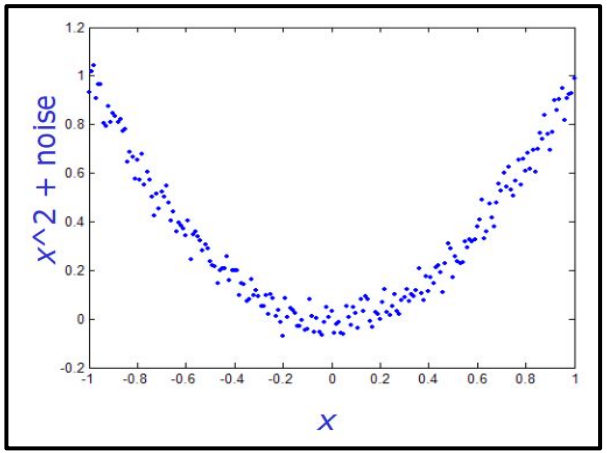
Site conseillé : http://rikunert.com/SMOTE_explained (R)

- Sélection des dimensions -

Évaluer les contributions des features à la Prédiction (Y)

Corrélation Linéaire entres les données : score R^2





Ne mesure pas les relations non linéaire

Correlation *R²*

$$r = \frac{\sum_{j=1}^{N}\left(\left(x^j - \bar{x}\right)\cdot\left(y^j - \bar{y}\right)\right)}{\sqrt{\sum_{j=1}^{N}\left(\left(x^j - \bar{x}\right)^2\right)\cdot\sum_{j=1}^{N}\left(\left(y^j - \bar{y}\right)^2\right)}}$$

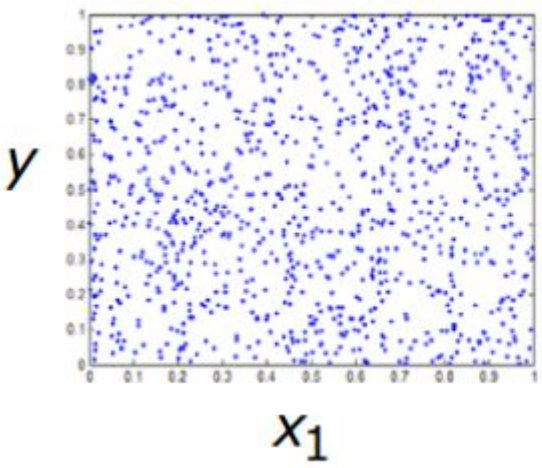
- Elle est linéaire.
- Elle est paramétrique (cela prend l'hypothèse d'un modèle linéaire…).
- Elle n'est pas explicative.
- Il est quasiment impossible de calculer une corrélation avec plus de 2 variables.
- Elle est sensible aux outliers.

R² = -1 et 1 pour des corrélations linéaire forte

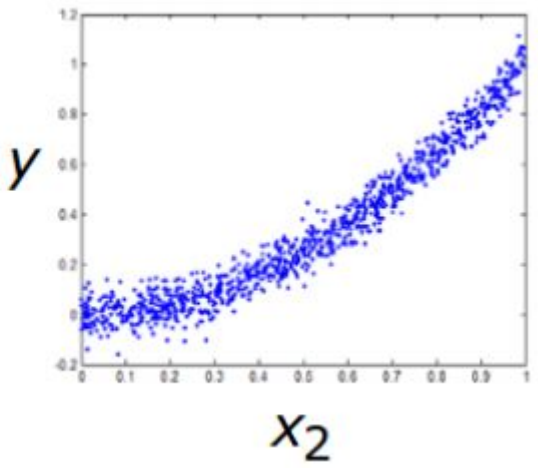
R² = 0 est l'absence de corrélation linéaire.

R² faible ne veux pas dire absence de corrélation.

Corrélation entre les features



A priori pas de corrélation !!



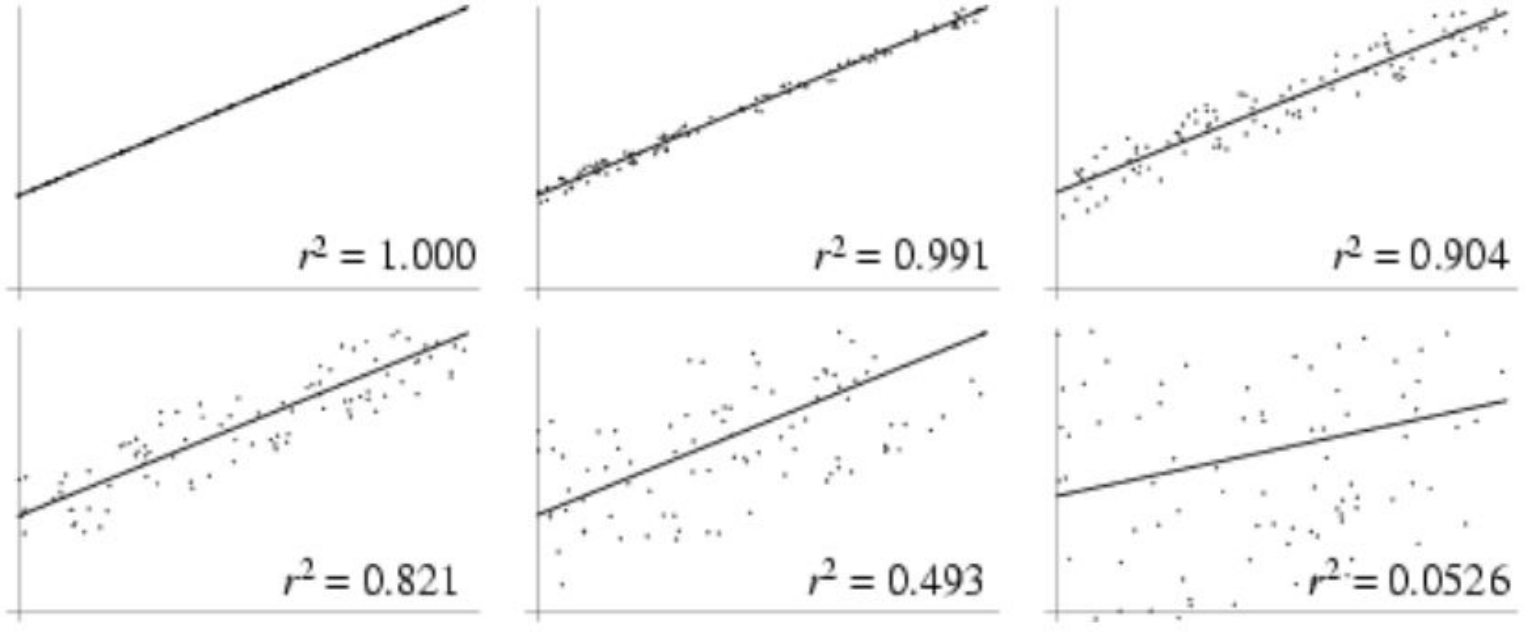
Corrélation évidente !!

Les corrélations sont souvent difficile à identifier

Corrélation linéaire entres les données : score

Dans le cas d'une régression linéaire

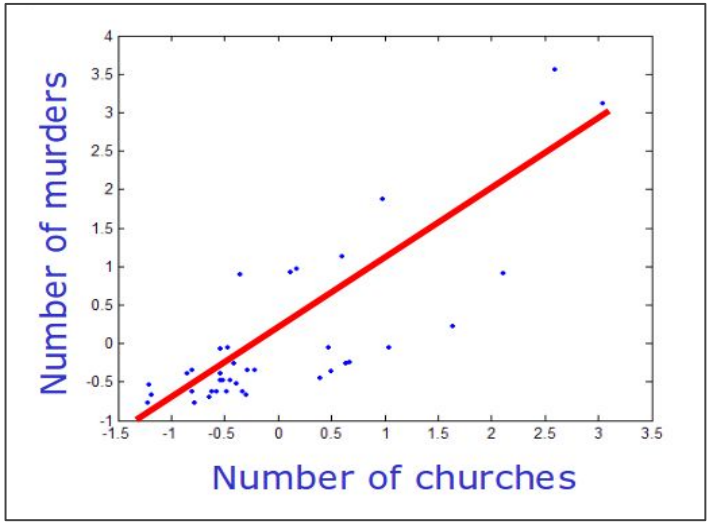
Forte corrélation



Faible corrélation

Corrélation entres les données : causalité

- Le nombre de meurtre dans la ville est fortement corrélé ($R^2 = 0,8$) avec le nombre d'église dans la ville
- C'est simplement quand la population augmente dans une ville, le nombre d'église et de meurtre augmente aussi...



Une forte corrélation ne veut pas dire **causalité**

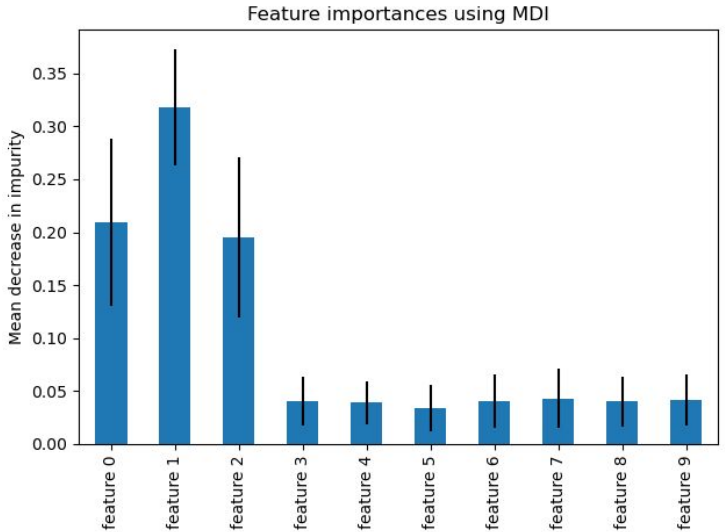
	christian chruches	murders 2002
Albuquerque	211	61
Atlanta	1500	152
Austin	353	25
Baltimore	466	253
Boston	370	60
Charlotte	505	67
Cleveland	980	80
Colorado Springs	400	25
Colombus	436	81
Denver	859	51
Detroit	1165	402
El paso	320	14
Fresno	450	42
Honolulu	39	18
Houston	1750	256
Indianapolis	1191	112
Jacksonville	21	3
Kansas city	1001	83
Long beach	236	67
Los Angeles	2000	654
Miami	911	65
milwaukee	411	111
Mnneapolis	419	47
New Orleans	712	258
New York	2233	587
oakland	374	108
Oklahoma City	25	38
Omaha	236	26
philadelphia	963	288
Portland	498	20
StLouis	900	111
San Diego	373	47
San Francisco	540	68
San Jose	403	26
Seattle	482	26
Tucson	382	47
Tulsa	330	26
Virginia Beach	248	3
Washington	742	264

Contribution des colonnes [arbre de décision]

Avec l'algo de l'arbre de décision, il est possible de calculer la contribution des features.
L'attribut **`feature_importances_`** donne, le calcul de la moyenne et la déviation standard de l'accumulation des impuretés contenues dans l'arbre de décision.

```
import time
import numpy as np

start_time = time.time()
importances = forest.feature_importances_
std = np.std([tree.feature_importances_ for tree in forest.estimators_], axis=0)
elapsed_time = time.time() - start_time

print(f"Elapsed time to compute the importances: {elapsed_time:.3f} seconds")
```

```
import pandas as pd

forest_importances = pd.Series(importances, index=feature_names)

fig, ax = plt.subplots()
forest_importances.plot.bar(yerr=std, ax=ax)
ax.set_title("Feature importances using MDI")
ax.set_ylabel("Mean decrease in impurity")
fig.tight_layout()
```



Corrélation entre les données

Apprentissage Supervisé

	Relation Linéaire	**Relation Non-Linéaire**
Selection	Correlation entre les inputs et l'output	Information mutuelle entre les inputs (algo glouton/greedy ou encore algo génétique).
Projection	Linear Discriminan Analyis, Partial least Square	Projection Pursuit

Apprentissage Non - Supervisé

	Relation Linéaire	**Relation Non-Linéaire**
Selection	Correlation entre les inputs	Information mutuelle entre les inputs [Entropie]
Projection	**P**rincipal **C**omponent **A**nalysis	Sammon Mapping Kohonen maps