

data visualization :

data describe() :

mean (mean of values)

std (ecart types) : ce qui indique la dispersion des valeurs par rapport à la moyenne.

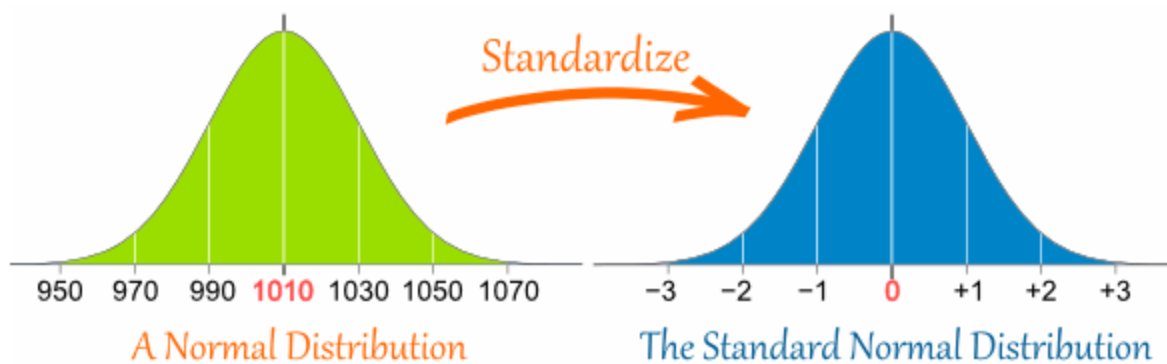
"25%" (premier quartile):

"50%" (médiane) :


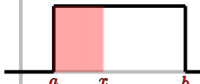
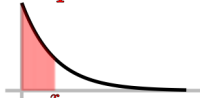

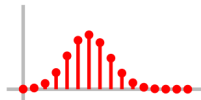
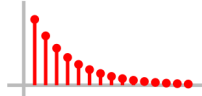
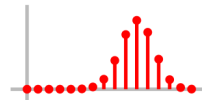
"75%" (3eme quartile):

about distribution :

normal



Histograms: It is a kind of bar graph which is an estimate of the probability distribution of a continuous variable

Probability Distributions					 Ace Tutors
Continuous	<div>Uniform</div>  $\mu = \frac{a+b}{2} \quad \sigma = \sqrt{\frac{(b-a)^2}{12}}$ $P(X < x) = \frac{x-a}{b-a}$	<div>Exponential</div>  $\mu = \frac{1}{\gamma} \quad \sigma = \frac{1}{\gamma}$ $P(X < x) = 1 - e^{-\gamma x}$	<div>Normal</div>  $z = \frac{x - \mu}{\sigma}$ $P(X < x) \Rightarrow \text{Use Z-Chart}$	<div>Key</div> <p>γ = rate parameter</p> <p>z = z-score</p> <p>p = probability of success</p> <p>n = # of trials</p> <p>N = population size</p> <p>K = # of success states</p>	
	Discrete	<div>Binomial</div>  $\mu = n \cdot p \quad \sigma = \sqrt{n \cdot p \cdot (1-p)}$ $P(X = x) = \binom{n}{x} p^x (1-p)^{n-x}$	<div>Geometric</div>  $\mu = \frac{1}{p} \quad \sigma = \frac{\sqrt{1-p}}{p}$ $P(X = x) = (1-p)^{x-1} p$	<div>Hypergeometric</div>  $\mu = n \frac{K}{N} \quad \sigma = \sqrt{n \frac{K(N-K)(N-K)}{N^2(N-1)}}$ $P(X = x) = \frac{\binom{K}{x} \binom{N-K}{n-x}}{\binom{N}{n}}$	

- uniform dis $p(X < x) = \frac{x-a}{b-a}$

Normalisation : La normalisation est utile lorsque la distribution des données n'est pas une distribution normale (gaussienne) ou lorsque vous voulez mettre toutes les variables sur la même échelle.

diff entre iloc et loc :

1. **.iloc** (index location) :

- **.iloc** est principalement basée sur des indices entiers.
- Vous pouvez utiliser **.iloc** pour sélectionner des lignes et des colonnes en spécifiant des indices entiers.
- Par exemple, `data.iloc[1, 2]` extraierait la valeur située à la deuxième ligne et à la troisième colonne de votre DataFrame.

2. **.loc** (label location) :

- **.loc** est principalement basée sur des labels ou des noms.
- Vous pouvez utiliser **.loc** pour sélectionner des lignes et des colonnes en spécifiant des noms de lignes et de colonnes.

- Par exemple, `data.loc['LigneA', 'ColonneB']` extraierait la valeur située à la ligne nommée 'LigneA' et à la colonne nommée 'ColonneB' de votre DataFrame.

nanvalues :

////////NaN values //////////

Lorsque vous rencontrez des valeurs manquantes (NaN) dans vos données de machine learning, vous pouvez utiliser différentes approches pour remplir ces valeurs en fonction du contexte et de la nature de vos données.

Voici quelques techniques couramment utilisées :

Suppression des lignes ou des colonnes : Si le nombre de valeurs manquantes est relativement faible par rapport à la taille totale de votre ensemble de donnée vous pouvez envisager de supprimer les lignes ou les colonnes contenant des valeurs manquantes.

Cependant, cela peut entraîner une perte d'informations précieuses, vous devez donc l'utiliser avec prudence.

Imputation par la moyenne ou la médiane : Vous pouvez remplacer les valeurs manquantes par la moyenne ou la médiane des valeurs existantes dans la même colonne

Cette méthode **est souvent utilisée pour les variables numériques continues**. La moyenne est sensible aux valeurs aberrantes (outliers), tandis que la médiane est plus robuste.

Imputation par la valeur la plus fréquente : **Pour les variables catégorielles**, vous pouvez remplacer les valeurs manquantes par la valeur la plus fréquente (mode) de la même colonne.

Cela fonctionne bien lorsque les valeurs manquantes sont peu nombreuses.

biblio :

Pandas is a library used for data analysis and organization. It provides flexible and powerful data structures like DataFrames and performs various data-related operations such as filtering, aggregation, and transformation.

NumPy is a library for scientific computations in Python. It offers data structures and functions for efficiently working with arrays and numerical data.

Seaborn is a library for statistical graphics in Python. It provides a convenient interface for creating beautiful visualizations and statistical plots to aid in data understanding.

Matplotlib.pyplot is a library for 2D data visualization in Python. It offers an interface for creating charts, plots, and detailed graphics.