Feature Engineering:

Feature extraction:

* **Imputation:**

↳ The problem is "Missing values"

due to ↳ Human errors
interruption in data flow
Privacy Concerns

⇒ Solutions: - Drop Row / column
- Imputation

Numerical imputation:
- put zero
- put Na
- Default medians of the columns

Categorical imputation:
- put the maximum occured value
- put other category like "other"

* **Outliers:**

Is a data which does not "fit in" with the other data that you are analysing.

⚠ So different from noise data
↳ is a data without signal
(unwanted data / wrong data)

Finding outliers:

We can find outliers using "statistical methodologies"

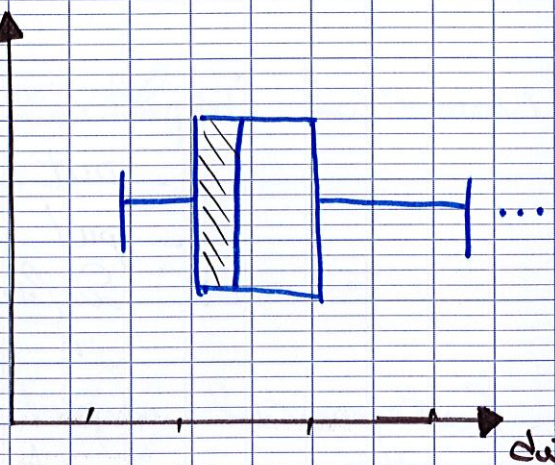
- Boxplot
- Scatter plot
- Z-score
- IQR
- Percentile.

1. Box plot:

```
import seaborn as sns
sns.boxplot(x = df["..."])
```

It's a graphical method capable of representing groups of data through their quartiles

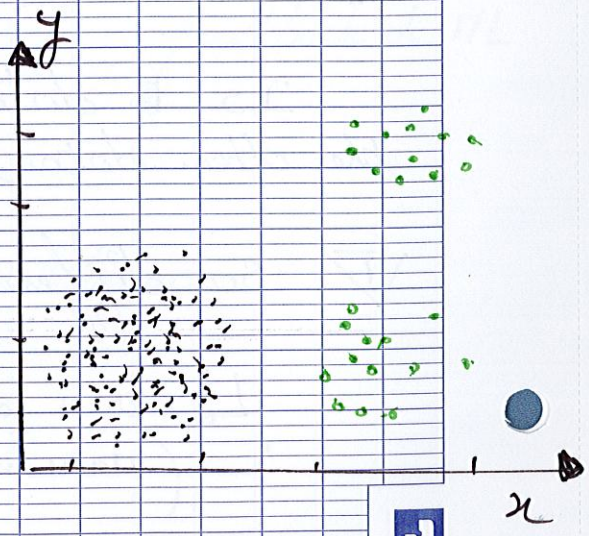
- The individual points are the outliers



2. Scatter plot

It's a mathematical diagram using coordinates to display values for two variables of the dataset.

- Looking to the plot we can see points which are far from the population like top right corner.



3. Standard deviation :

It's also reprented by the letter σ, which measure the amount of variation of values.

How to use Sd ??

we can use it to fix the upper and lower limit :

$up = df.mean() + df.std() * factor$

$low = df.mean() - df.std() * factor$

⇒ The factor is usually a value between 2 and 4, in the most of cases we put 3.

4. Z - score :

- It's the signed number of standard deviations, can be calculating using :

$$Z = \frac{X - \mu}{\sigma}$$

μ : mean()

σ : std()

- Z-score is capable of re-scaling the data and look for data points which are too far from zero 0.

In the most cases, its value is between 3 and -3 so the other values out of this marge are the outliers.

Fonction Python :

stats.zscore(df)

stats is a function of scipy.

5. IQR score :

The 'interquartile range', is a measure of statistical dispersion, equal to the difference between 75% and 25% percentile

$$IQR = Q_3 - Q_1$$

In this Case :

$$Lower\_limit = Q_1 - 1{,}5 * IQR$$

$$Upper\_limit = Q_3 + 1{,}5 * IQR$$

6. Percentiles :

with this method, you can assume a certain percent of the value from the top or the bottom as an outlier,

we have to calculate :

- quantile (0,95)
- quantile (0,05)

Handling Outliers:

• Binning:

This technic can be applied on both categorical and numerical data

- values → {low, mid, High}.
- Categories → {Euppe, africa ... } (exp)

- The purpose is to make the model more robust and prevent overfitting

• Log transformation:

- It helps to handle skewed data, the distribution becomes more approximate to normal.
- The model become more robust.
- The data have to be only positive values, and we can add 1 to ensure the output to be positive.

```
df["value"].transform(np.log)
```

• Grouping:

based on the use of "group by", the key point is to decide the aggregation functions of the features.

- For the categorical columns we use the max operation
- " numirical columns we use sum or mean

Scaling :

* The main problem here is that the numerical features do not have a certain range and they differ from each other.

* There are two common ways of scaling : Normalization and Standardization.

* Normalization :

It's scaling all values in a fixed range between 0 and 1.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

↳ This transformation do not change the distribution of the features.

* Standardization :

It's also called (Z-score normalization), can scale the values while taking in to account standard deviation

$$Z = \frac{x - \mu}{\sigma}$$

μ : th mean

σ : std.

Feature Encoding:

- It's about turning categorical features into numeric features to provide more fine-grained information.

Labeled Encoding:

This method consist in transforming categories into ordered integers.

A → 0
B → 1
C → 2

Python sklearn
↳ Label Encoder

One Hot Encoding:

Transform categories into individual binary (0 or 1)
↳ one of the most famous encoding methods.

- It consist in spreading the values in a column to multiple flag column and assigns 0 or 1 to them.

User	city
1	Rome
2	Madrid
1	France.
3	

→

User	Rome	madrid	France
1	1	0	0
2			
3			

The binary values expres the relation ship between grouped and encoded column.

Frequency Encoding:

Consists in encoding categorical levels of features to values between 0 and 1 based on their relative frequency.

A → 0,6
A → 0,6
A → 0,6
B → 0,4
B → 0,4

} → 3/5

} → 2/5

Target mean encoding:

Instead of increasing the number of features by encoding categorical variable, we can encode each level as the mean of the response.

		Mean encode
A	1	2/3
A	0	2/3
A	1	2/3
B	1	2/2
B	1	2/2.