**Handling the Missing Values**

In real-world data would certainly have missing values. This may be due to many reasons such as data transformation errors, wrong queries or data entry error problems. It is important to handle missing data because any statistical results based on a dataset with non-random missing values could be biased. Also, many ML algorithms do not support data with missing values.

**How to Identify Missing Values?**

We can check for null values in a derived dataset. But, sometimes, it might not be this simple to identify missing values. One needs to use the domain knowledge and look at the data description to understand the variables. There are variables that have a minimum value of zero. On some columns, a value of zero does not make sense and indicates an invalid or missing value.

**Quick Classification of Missing Data**

There are three types of missing data as below:

**Missing Completely At Random (MCAR):** It is the highest level of randomness. This means that the missing values in any features are not dependent on any other feature’s values. This is the desirable scenario in case of missing data.

**Missing At Random (MAR):** This means that the missing values in any feature are dependent on the values of other features.

**Missing Not At Random (MNAR):** Missing not at random data is a more serious issue and, in this case, it might be wise to check the data gathering process further and try to understand why the information is missing.

**What to Do with the Missing Values?**

We identified the missing values in a derived dataset, next we should decide the further course of action.

* **Ignore the missing values:**
* Missing data under 10% for an individual case or observation can generally be ignored, except when the missing data is a MAR or MNAR.
* The number of complete cases i.e., observation with no missing data must be sufficient for the selected analysis technique if the incomplete cases are not considered.

**Drop the missing values:**

Dropping a variable

\* If the data is MCAR or MAR and the number of missing values in a feature is very high, then that feature should be left out of the analysis.

If missing data for a certain feature or sample is more than 5% then you probably should leave that feature or sample out.

\* If the cases or observations have missing values for target variables(s), it is advisable to delete the dependent variable(s) to avoid any artificial increase in relationships with independent variables.

**Case Deletion:**

In this method, cases which have missing values for one or more features are deleted. If the cases having missing values are small in number,

it is better to drop them. Though this is an easy approach, it might lead to a significant decrease in the sample size. Also, the data may not

always be missing completely at random. This may lead to biased estimation of parameters.

**Imputation:**

Imputation is the process of substituting the missing data by some statistical methods. Imputation is useful in the sense that it preserves

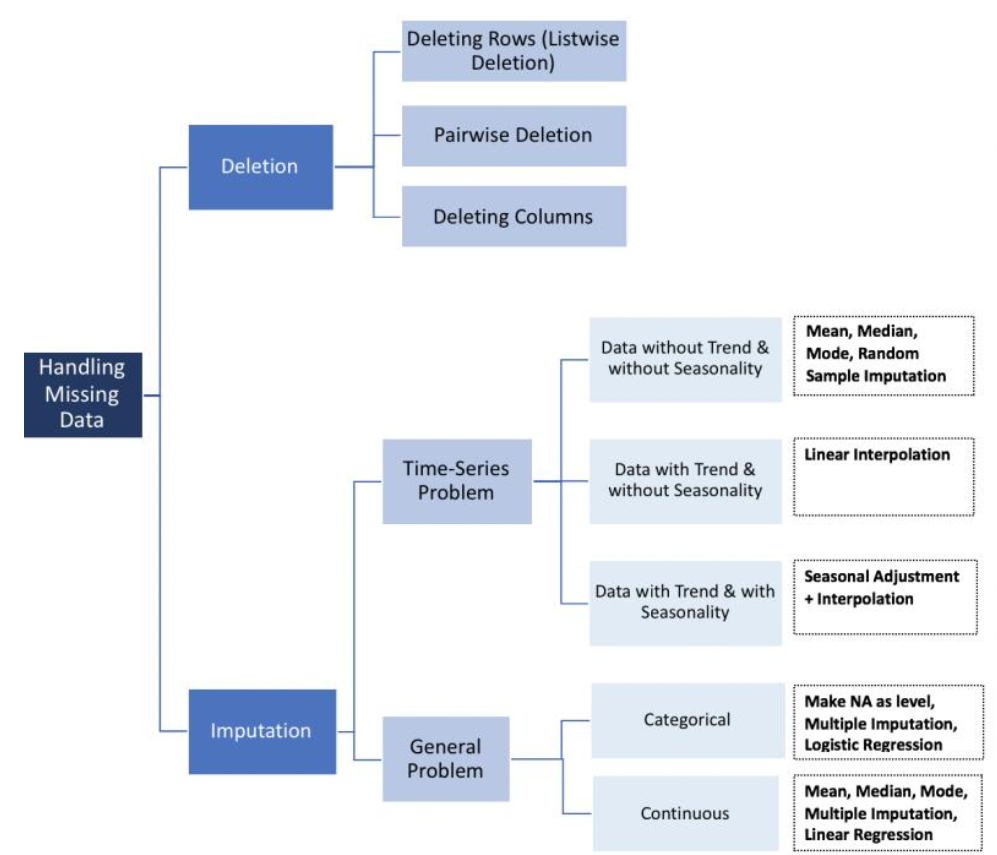
all cases by replacing missing data with an estimated value based on other available information. But imputation methods should be used

carefully as most of them introduce a large amount of bias and reduce variance in the dataset.

In Pandas missing data is represented by two values:

**None**: None is a Python singleton object that is often used for missing data in Python code.

**NaN**: NaN (an acronym for Not a Number), is a special floating-point value recognized by all systems that use the standard IEEE floating-point representation.



**Treating missing data:**

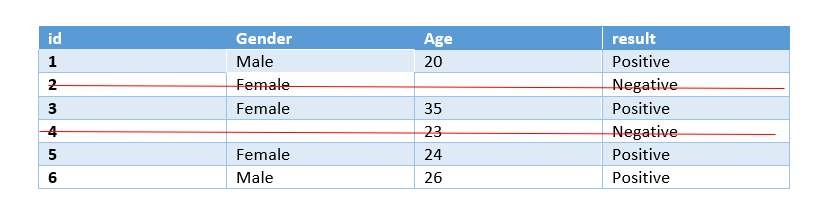
After classified the patterns in missing values, it needs to treat them.

**Deletion:**

The Deletion technique deletes the missing values from a dataset. followings are the types of missing data.

**Listwise deletion:**

Listwise deletion is preferred when there is a Missing Completely at Random case. In Listwise deletion entire rows (which hold the missing values) are deleted. It is also known as complete-case analysis as it removes all data that have one or more missing values.

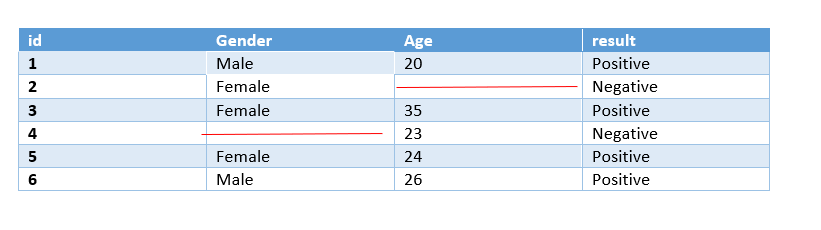


In python we use **dropna()** function for Listwise deletion.

Listwise deletion is not preferred if the size of the dataset is small as it removes entire rows if we eliminate rows with missing data then the dataset becomes very short and the machine learning model will not give good outcomes on a small dataset.

**Pairwise Deletion:**

Pairwise Deletion is used if missingness is missing completely at random i.e MCAR.



Pairwise deletion is preferred to reduce the loss that happens in Listwise deletion. It is also called an available-case analysis as it removes only null observation, not the entire row.

**Note:** All methods in pandas like mean, sum, etc. intrinsically skip missing values.

**Dropping complete columns**

If a column holds a lot of missing values, say more than 80%, and the feature is not meaningful, that time we can drop the entire column.