Data and It's Processing

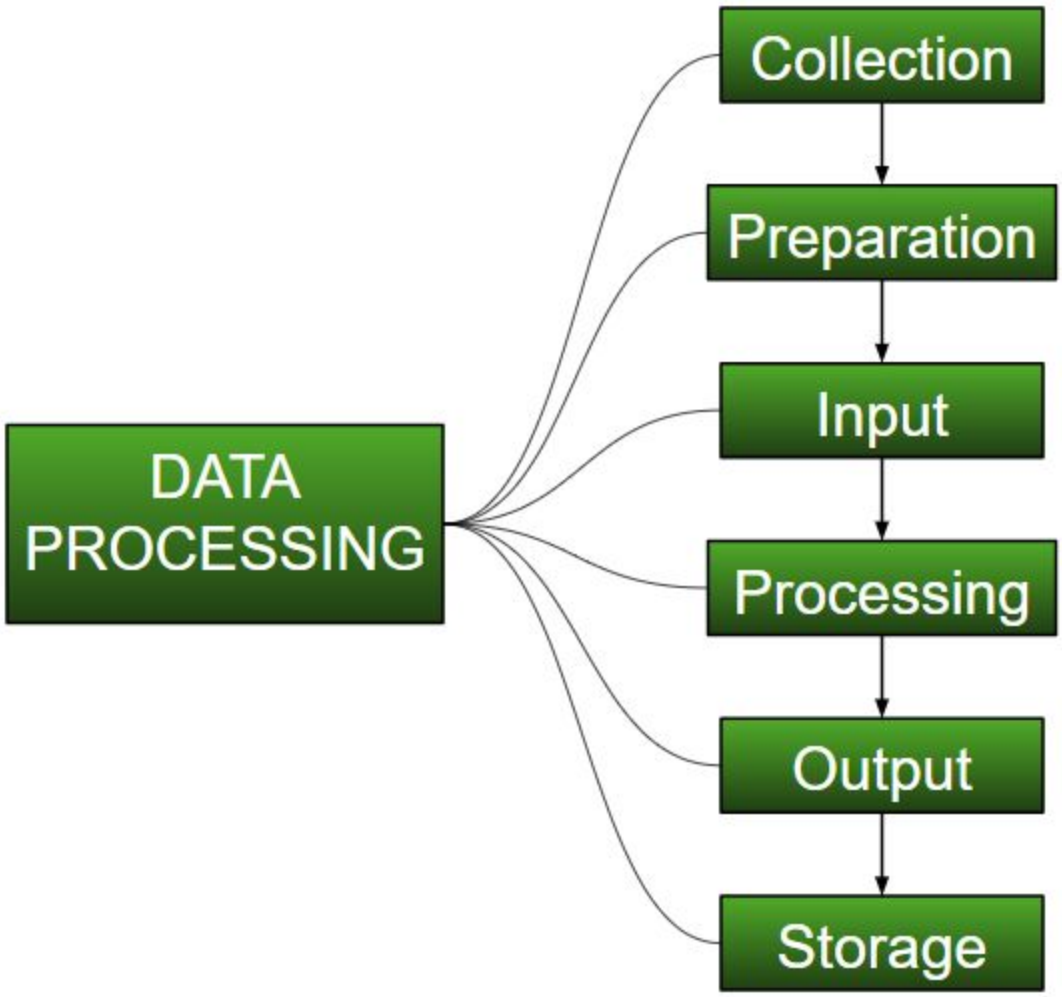
DATA : It can be any unprocessed fact, value, text, sound or picture that is not being interpreted and analyzed. Data is the most important part of all *Data Analytics, Machine Learning, Artificial Intelligence*. **Without** data, we **can't** train any model and all modern research and automation will go vain. Big Enterprises are spending lots of money just to gather as much certain data as possible.
Example: Why did Facebook acquire WhatsApp by paying a huge price of $19 billion? The answer is very simple and logical – it is to have access to the users' information that Facebook may not have but WhatsApp will have. This information of their users is of paramount importance to Facebook as it will facilitate the task of improvement in their services.
INFORMATION : Data that has been interpreted and manipulated and has now some meaningful inference for the users.
KNOWLEDGE : Combination of inferred information, experiences, learning and insights. Results in awareness or concept building for an individual or organization.

How we split data in Machine Learning?

- **Training Data:** The part of data we use to train our model. This is the data which your model actually sees(both input and output) and learn from.
- **Validation Data:** The part of data which is used to do a frequent evaluation of model, fit on training dataset along with improving involved hyperparameters (initially set parameters before the model begins learning). This data plays it's part when the model is actually training.
- **Testing Data:** Once our model is completely trained, testing data provides the unbiased evaluation. When we feed in the inputs of Testing data, our model will predict some values(without seeing actual output). After prediction, we evaluate our model by comparing it with actual output present in the testing data. This is how we evaluate and see how much our model has learned from the experiences feed in as training data, set at the time of training.

Properties of Data –

1. **Volume :** Scale of Data. With growing world population and technology at exposure, huge data is being generated each and every millisecond.
2. **Variety :** Different forms of data – healthcare, images, videos, audio clippings.
3. **Velocity :** Rate of data streaming and generation.
4. **Value :** Meaningfulness of data in terms of information which researchers can infer from it.
5. **Veracity :** Certainty and correctness in data we are working on.



- **Collection :**

  The most crucial step when starting with ML is to have data of good quality and accuracy. Data can be collected from any authenticated source like data.gov.in, Kaggle or UCI dataset repository.For example, while preparing for a competitive exam, students study from the best study material that they can access so that they learn the best to obtain the best results. In the same way, high-quality and accurate data will make the learning process of the model easier and better and at the time of testing, the model would yield state of the art results.

  A huge amount of capital, time and resources are consumed in collecting data. Organizations or researchers have to decide what kind of data they need to execute their tasks or research.

  Example: Working on the Facial Expression Recognizer, needs a large number of images having a variety of human expressions. Good data ensures that the results of the model are valid and can be trusted upon.

- **Preparation :**

  The collected data can be in a raw form which can't be directly fed to the machine. So, this is a process of collecting datasets from different sources, analyzing these datasets and then constructing a new dataset for further processing and exploration. This preparation can be performed either manually or from the automatic approach. Data can also be prepared in numeric forms also which would fasten the model's learning.

  Example: An image can be converted to a matrix of N X N dimensions, the value of each cell will indicate image pixel.

- **Input :**

  Now the prepared data can be in the form that may not be machine-readable, so to convert this data to readable form, some conversion algorithms are needed. For this task to be executed, high computation and accuracy is needed. Example: Data can be collected

through the sources like MNIST Digit data(images), twitter comments, audio files, video clips.

- **Processing :**
  This is the stage where algorithms and ML techniques are required to perform the instructions provided over a large volume of data with accuracy and optimal computation.
- **Output :**
  In this stage, results are procured by the machine in a meaningful manner which can be inferred easily by the user. Output can be in the form of reports, graphs, videos, etc
- **Storage :**
  This is the final step in which the obtained output and the data model data and all the useful information are saved for the future use.

The probability density for the Gaussian distribution is:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}},$$

Standard Deviation:

Say we have a bunch of numbers like 9, 2, 5, 4, 12, 7, 8, 11.

To calculate the standard deviation of those numbers:

- 1. Work out the Mean (the simple average of the numbers)
- 2. Then for each number: subtract the Mean and square the result
- 3. Then work out the mean of **those** squared differences.
- 4. Take the square root of that and we are done!

In the formula above μ (the greek letter "mu") is the mean of all our values ...

Example: 9, 2, 5, 4, 12, 7, 8, 11, 9, 3, 7, 4, 12, 5, 4, 10, 9, 6, 9, 4

The mean is:

$$\frac{9+2+5+4+12+7+8+11+9+3+7+4+12+5+4+10+9+6+9+4}{20}$$

$$= \frac{140}{20} = \mathbf{7}$$

So:

$$\mu = 7$$

Step 2. Then for each number: subtract the Mean and square the result

This is the part of the formula that says:

$$(x_i - \mu)^2$$

So what is x_i ? They are the individual x values 9, 2, 5, 4, 12, 7, etc...

In other words $x_1 = 9$, $x_2 = 2$, $x_3 = 5$, etc.

So it says "for each value, subtract the mean and square the result", like this

Example (continued):

$(9 - 7)^2 = (2)^2 = \mathbf{4}$

$(2 - 7)^2 = (-5)^2 = \mathbf{25}$

$(5 - 7)^2 = (-2)^2 = \mathbf{4}$

$(4 - 7)^2 = (-3)^2 = \mathbf{9}$

$(12 - 7)^2 = (5)^2 = \mathbf{25}$

$(7 - 7)^2 = (0)^2 = \mathbf{0}$

$(8 - 7)^2 = (1)^2 = \mathbf{1}$

... etc ...

And we get these results:

4, 25, 4, 9, 25, 0, 1, 16, 4, 16, 0, 9, 25, 4, 9, 9, 4, 1, 4, 9

Data Preprocessing:

1. Rescale Data

```
[[ 0.353  0.744  0.59   0.354  0.0    0.501  0.234  0.483]
 [ 0.059  0.427  0.541  0.293  0.0    0.396  0.117  0.167]
 [ 0.471  0.92   0.525  0.     0.0    0.347  0.254  0.183]
 [ 0.059  0.447  0.541  0.232  0.111  0.419  0.038  0.0  ]
 [ 0.0    0.688  0.328  0.354  0.199  0.642  0.944  0.2  ]]
```

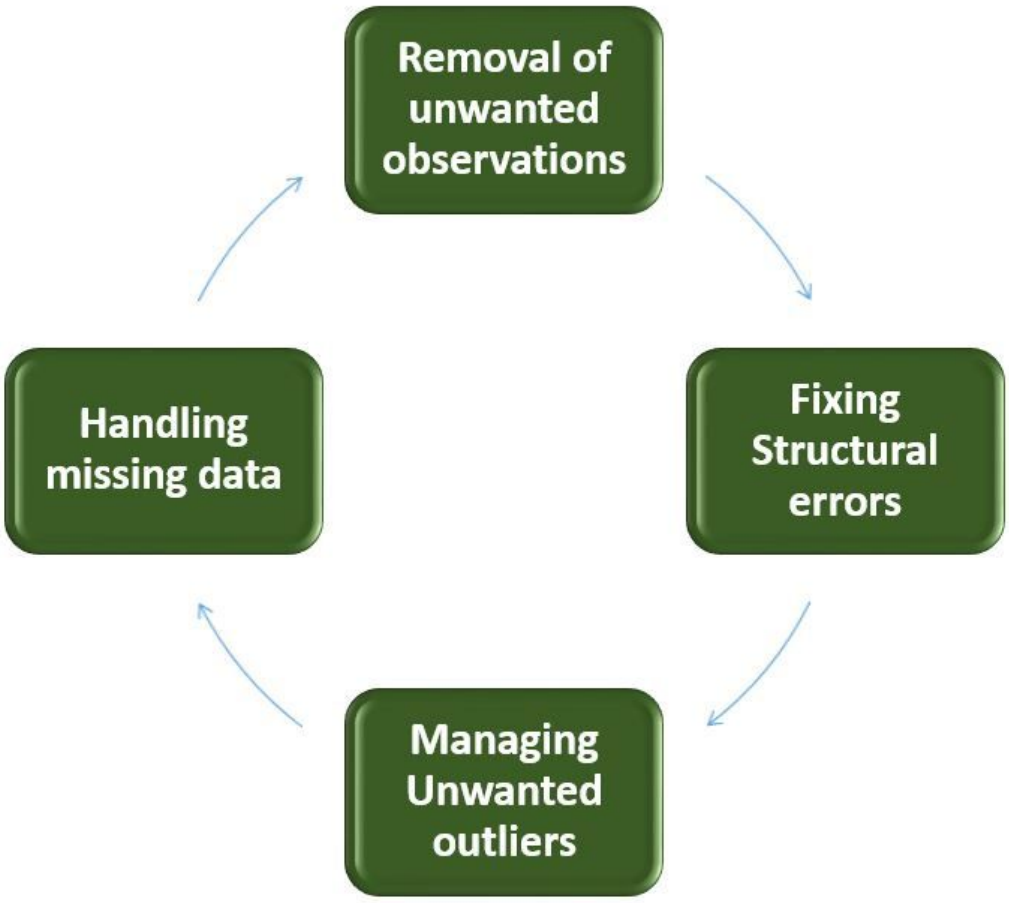
2. Binarize Data

```
[[ 1.  1.  1.  1.  0.  1.  1.  1.]
 [ 1.  1.  1.  1.  0.  1.  1.  1.]
 [ 1.  1.  1.  0.  0.  1.  1.  1.]
 [ 1.  1.  1.  1.  1.  1.  1.  1.]
 [ 0.  1.  1.  1.  1.  1.  1.  1.]]
```

3. Standardize Data

```
[[ 0.64   0.848  0.15   0.907 -0.693  0.204  0.468  1.426]
 [-0.845 -1.123 -0.161  0.531 -0.693 -0.684 -0.365 -0.191]
 [ 1.234  1.944 -0.264 -1.288 -0.693 -1.103  0.604 -0.106]
 [-0.845 -0.998 -0.161  0.155  0.123 -0.494 -0.921 -1.042]
 [-1.142  0.504 -1.505  0.907  0.766  1.41   5.485 -0.02 ]]
```

Data Cleansing:



1. **Removal of unwanted observations**

   This includes deleting duplicate/ redundant or irrelevant values from your dataset. Duplicate observations most frequently arise during data collection and Irrelevant observations are those that don't actually fit the specific problem that you're trying to solve.

   1. Redundant observations alter the efficiency by a great extent as the data repeats and may add towards the correct side or towards the incorrect side, thereby producing unfaithful results.
   2. Irrelevant observations are any type of data that is of no use to us and can be removed directly.

2. **Fixing Structural errors**

   The errors that arise during measurement, transfer of data or other similar situations are called structural errors. Structural errors include typos in the name of features, same attribute with different name, mislabeled classes, i.e. separate classes that should really be the same or inconsistent capitalization.

    1. For example, the model will treat America and america as different classes or values, though they represent the same value or red, yellow and red-yellow as different classes or attributes, though one class can be included in other two classes. So, these are some structural errors that make our model inefficient and gives poor quality results.

3. **Managing Unwanted outliers**

   Outliers can cause problems with certain types of models. For example, linear regression models are less robust to outliers than decision tree models. Generally, we should not remove outliers until we have a legitimate reason to remove them. Sometimes, removing them improves performance, sometimes not. So, one must have a good reason to remove the outlier, such as suspicious measurements that are unlikely to be the part of real data.

4. **Handling missing data**

   Missing data is a deceptively tricky issue in machine learning. We cannot just ignore or remove the missing observation. They must be handled carefully as they can be an indication of something important. The two most common ways to deal with missing data are:

    1. Dropping observations with missing values.
    2. Dropping missing values is sub-optimal because when you drop observations, you drop information.

        - The fact that the value was missing may be informative in itself.
        - Plus, in the real world, you often need to make predictions on new data even if some of the features are missing!
    3. Imputing the missing values from past observations.
    4. Imputing missing values is sub-optimal because the value was originally missing but you filled it in, which always leads to a loss in information, no matter how sophisticated your imputation method is.
        - Again, “missingness” is almost always informative in itself, and you should tell your algorithm if a value was missing.
        - Even if you build a model to impute your values, you’re not adding any real information. You’re just reinforcing the patterns already provided by other features.
5. Both of these approaches are sub-optimal because dropping an observation means dropping information, thereby reducing data and imputing values also is sub-optimal as we fil the values that were not present in the actual dataset, which leads to a loss of information.

Feature Scaling

Euclidean Distance : It is the square-root of the sum of squares of differences between the coordinates (feature values – Age, Salary, BHK Apartment) of data point and centroid of each class. This formula is given by Pythagorean theorem.

$$d(x, y) = \sqrt[r]{\sum_{k=1}^{n}(x_k - y_k)^r}$$

Manhattan Distance : It is calculated as the sum of absolute differences between the coordinates (feature values) of data point and centroid of each class.

$$d(x, y) = \sum_{k=1}^{n}|x_k - y_k|$$

Minkowski Distance : It is a generalization of above two methods. As shown in the figure, different values can be used for finding r.

$$d(x, y) = \sqrt[r]{\sum_{k=1}^{n}(x_k - y_k)^r}$$

Min-Max Normalization: This technique re-scales a feature or observation value with distribution value between 0 and 1.

$$X_{new} = \frac{X_i - \min(X)}{\max(x) - \min(X)}$$

Standardization: It is a very effective technique which re-scales a feature value so that it has distribution with 0 mean value and variance equals to 1.

$$X_{new} = \frac{X_i - X_{mean}}{\text{Standard Deviation}}$$

Note:
All the values of the given data after the successful min-max normalization is going to be in the given input range of the user.

After the successful standardisation, the sum of the given data values are going to be equal to 0.

Imbalanced Data Handling Techniques: There are mainly 2 mainly algorithms that are widely used for handling imbalanced class distribution.

1. SMOTE
2. Near Miss Algorithm