**Approach**

**Name:** Harish B

**Email:** [harishbharish6@gmail.com](mailto:harishbharish6@gmail.com)

**Step 1 :- Data Pre-processing**

Real-world data typically includes noise, missing values, and may be in an undesirable format, making it impossible to build machine learning models on it directly. Data pre-processing is necessary to clean the data and prepare it for a machine learning model, which also improves the model's accuracy and effectiveness.

1. **Get the Dataset:** A dataset is the first item needed to develop a machine learning model because data is the basis for all machine learning models. The dataset is the properly formatted collection of data for a certain issue.
2. **Importing Libraries:** Python predefined libraries must be imported in order to pre-process data using python.
3. **Importing Datasets:** The datasets that we have gathered for our machine learning research must now be imported.
4. **Handling Missing Data:** The handling of missing data in the datasets is the next stage of data pre-processing. Our machine learning model may run into serious issues if our dataset has some missing data. As a result, the dataset contains missing values, which must be handled.
5. **Splitting the Dataset into Training and Validation set:** One of the most important data pre-processing stages, since it allows us to improve the functionality of our machine learning model. Imagine that we trained our machine learning model using one dataset, and then we tested it using a completely other dataset. The understanding of the correlations between the models will then become challenging for our model. If we train our model really well and it has a high training accuracy, but we give it a fresh dataset, the performance will suffer. Therefore, we constantly strive to create a machine learning model that excels both with the training set and the test dataset.

**Step 2 :- Data Visualization**

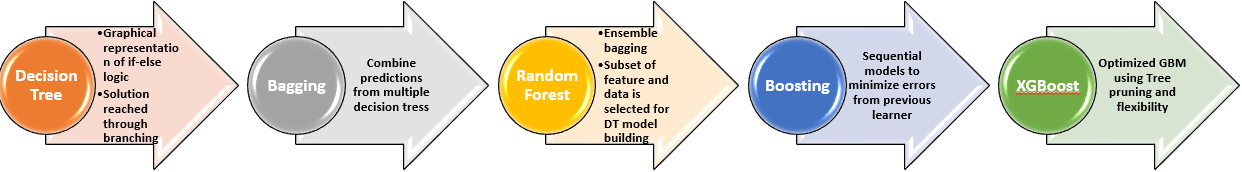
By organising the data into an understandable format and showing the trends and outliers, data visualisation aids in the telling of stories. A strong visualisation highlights important information while reducing data noise. Data visualization is the graphical representation of information and data in a pictorial or graphical format (Example: charts, graphs, and maps). Data visualization tools provide an accessible way to see and understand trends, patterns in data, and outliers. Data visualization tools and technologies are essential to analyzing massive amounts of information and making data-driven decisions. The concept of using pictures is to understand data that has been used for centuries. General types of data visualization are Charts, Tables, Graphs, Maps, Dashboards.

**Step 3 :- Predicting Output with Machine Learning Model**

**XGBoost Model**

Extreme Gradient Boosting (XGBoost) is a distributed, scalable gradient-boosted decision tree (GBDT) machine learning framework. The top machine learning library for regression, classification, and ranking issues, it offers parallel tree boosting. XGBoost, or Extreme Gradient Boosting, emerges as the obvious choice when it comes to a superfast machine learning algorithm that operates on tree-based models and attempts to achieve the best-in-class accuracy by optimally employing CPU resources. The XGBoost algorithm, developed by Tianqi Chen, has recently gained a lot of attention due to its widespread use in hackathons and Kaggle events.

Simply put, XGBoost may be officially defined as a decision tree-based ensemble learning framework that uses Gradient Descent as the underlying objective function, offers a great deal of flexibility, and efficiently utilises computing power to produce the required results.



**Evolution of XGBoost from Tree-based model**

The boosting algorithms generally work on three sequential steps as stated below:

* An initial model is built as F1 to predict the target variable y by giving equal importance to all samples
* The predicted outcome of the model is said y1. Error or residuals are estimated as y-y1
* A new model h1 is then built considering the error (y-y1) as the dependent variable
* Finally, a combined model is built using both F1 and h1 to produce a boosted variant of F1 that will result in a reduced error.