Ensemble Technique

What is Ensemble ?

→ Combining Multiple Models

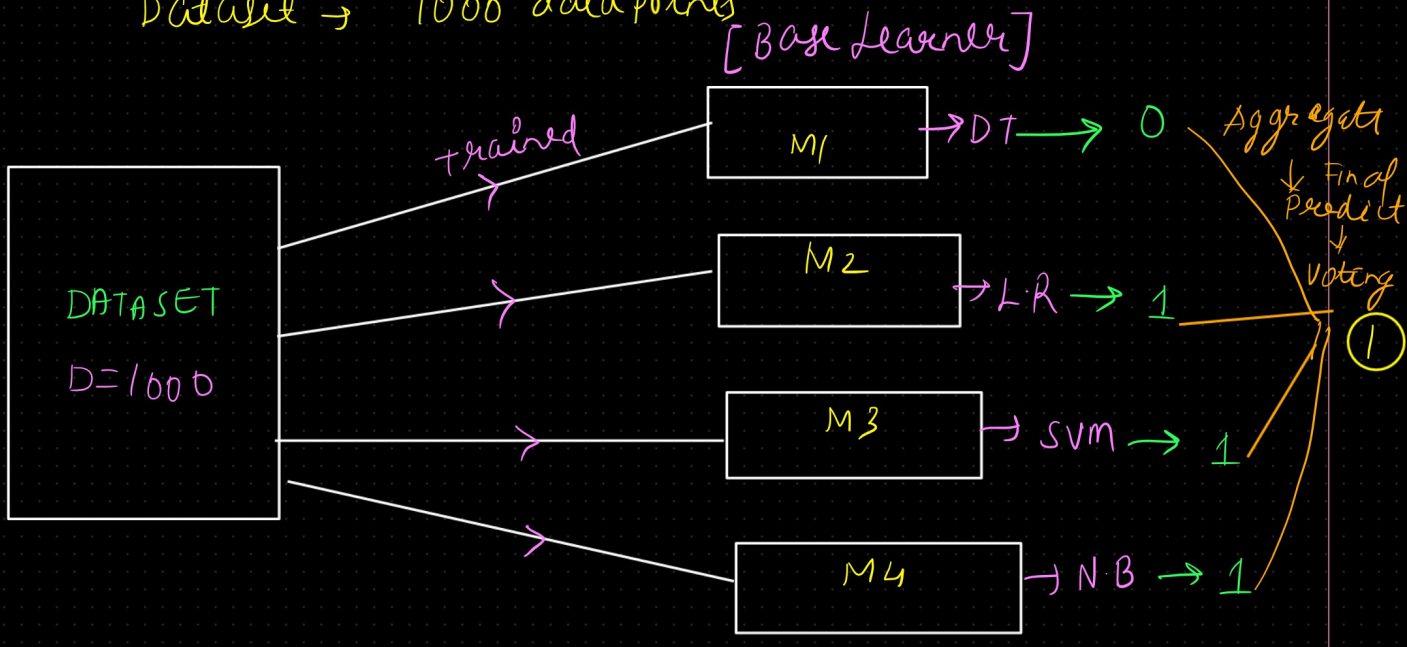
→ Ensemble techniques are a popular approach in machine learning where multiple models are combined to improve overall performance

Two Types :

1. Bagging (Bootstrap Aggregating)
2. Boosting (AdaBoost, Xgboost)

→ Bagging involves training multiple instances of the same base learning algorithm on different subsets of the training data. Each model is trained independently, and predictions are typically combined through averaging or voting.

Bagging Technique



Random Forest

Random Forest is an ensemble learning method that combines multiple decision trees. Each decision tree is trained on a random subset of the training data and a random subset of features.

Model → Decision Tree

Training data → Random Subset

Random Feature

Row sampling

Feature sampling

→ For instance, if you have a dataset with 1000 samples and 20 features, each decision tree in the random forest might be trained on a random selection of, say, 800 samples and 10 features.

↓

As a sample data

Dataset → 1000 samples

20 feature

⇓ Random Forest

Multiple D.T

⇓ Model Trained

10 feature, 800 Sample

→ During prediction, the outputs of all decision trees are aggregated (e.g., through averaging for regression tasks or voting for classification tasks) to produce the final prediction.

Random Forest → ML Algo

↓

Supervised Algo

↓

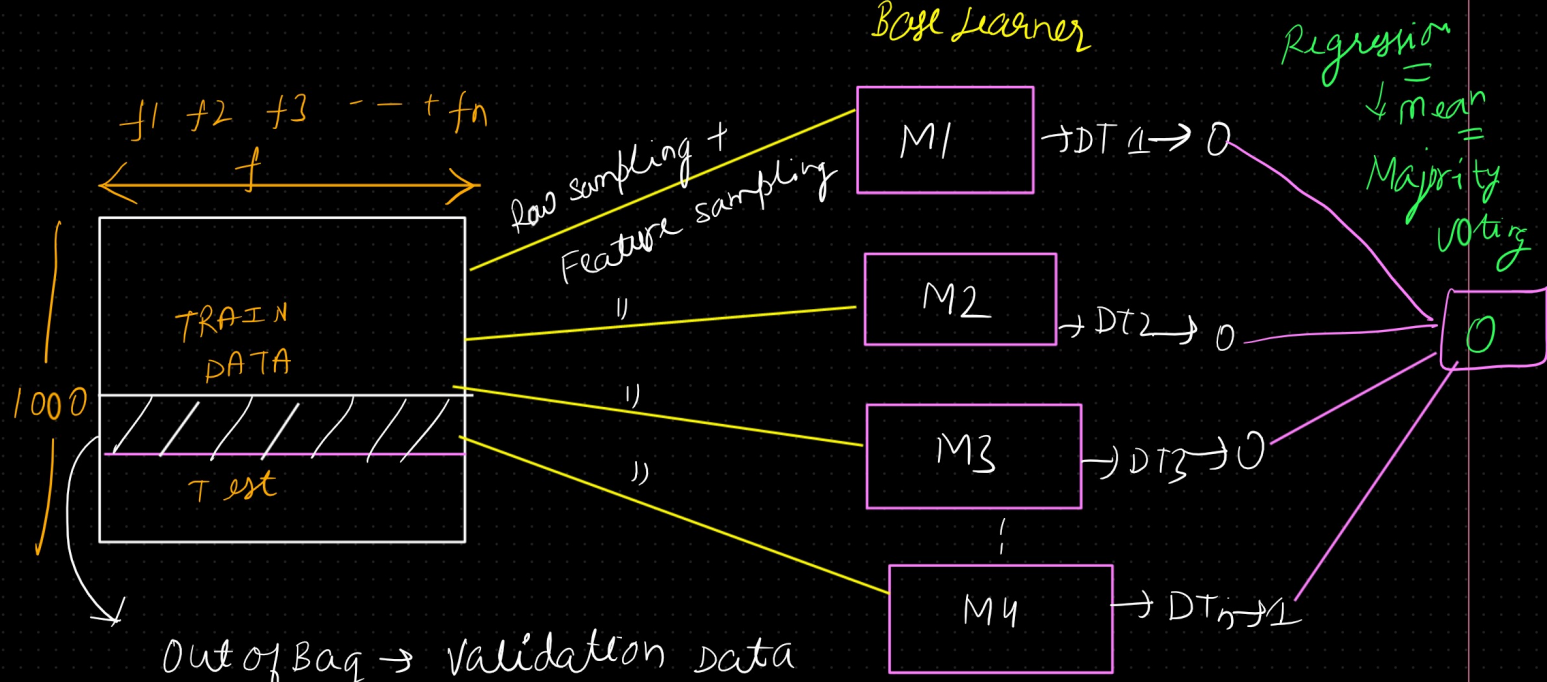
Classification / Regression

Voting

↘

Average (mean)

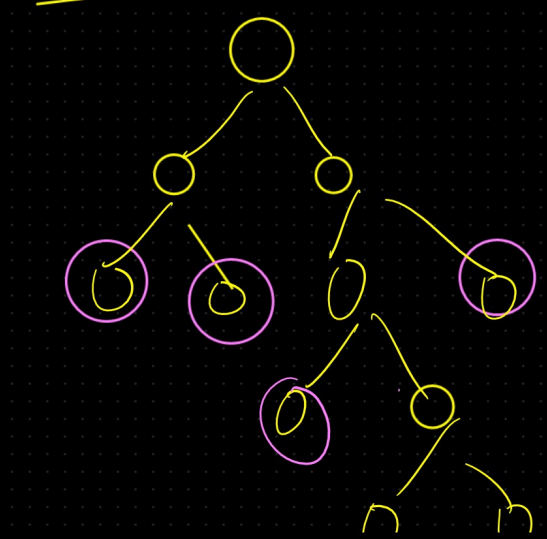
Random Forest Classifier And Regressor



Note:-

Classifier → Majority Voting

Regression → Average O/p of the model



Low Bias

↓

Overfitting → TRAIN ACC ↑↑

Test ACC ↓↓

↓

high variance

Generalized Model

What is a random forest in simple terms?

Random forest is an algorithm that generates a 'forest' of decision trees. It then takes these many decision trees and combines them to avoid overfitting and produce more accurate predictions.

What is the difference between decision trees and random forest?

The difference between decision trees and random forest is that decision trees consider all possible outcomes in the search for the best outcome based on the data provided; random forest generates random predictions from multiple decision trees and averages these out. As a result, decision trees may fall victim to overfitting, but random forest doesn't.

How does random forest work?

Random forest produces multiple decision trees, randomly choosing features to make decisions when splitting nodes to create each tree. It then takes these randomized observations from each tree and averages them out to build a final model

Out of bag (Score)

The out-of-bag (OOB) score, also known as out-of-bag error, is a method used in Random Forest and other ensemble learning techniques to estimate the model's performance without the need for a separate validation set. It measures the prediction error of each individual tree in the ensemble on the samples that were not included in its bootstrap training set.

Random Forest Classifier:
Random Forest Classifier is used for classification tasks, where the goal is to predict the class labels of instances based on input features. Here's how it works:

Data Preparation:

You start with a labeled dataset where each instance has features (independent variables) and corresponding class labels (dependent variables).
Random Sampling:

Random Forest randomly selects subsets of the training data with replacement (bootstrapping). Each subset is used to train a decision tree.
Random Feature Selection:

At each node of the decision tree, only a random subset of features is considered for finding the best split.
Decision Tree Construction:

Decision trees are grown independently to their maximum depth, without pruning, based on the bootstrapped samples and random feature subsets.
Voting:

During prediction, each decision tree in the forest independently predicts the class label of the instance.
The final prediction is determined by majority voting: the class label with the most votes across all trees is assigned to the instance.

Example (Random Forest Classifier):

Suppose you have a dataset containing information about various characteristics of fruits (e.g., color, diameter, weight) and their corresponding labels indicating whether they are "apple," "orange," or "banana." You want to build a model to predict the type of fruit based on these features.

Color	Diameter (cm)	Weight (g)	Label
Red	7	150	Apple
Orange	6	120	Orange
Yellow	8	175	Banana
Red	7.5	155	Apple
...

Random Forest Regressor:
Random Forest Regressor is used for regression tasks, where the goal is to predict continuous values rather than discrete class labels. Here's how it works:

Data Preparation:

You start with a labeled dataset where each instance has features (independent variables) and corresponding continuous target values (dependent variables).
Random Sampling:

Random Forest randomly selects subsets of the training data with replacement (bootstrapping). Each subset is used to train a decision tree.
Random Feature Selection:

At each node of the decision tree, only a random subset of features is considered for finding the best split.
Decision Tree Construction:

Decision trees are grown independently to their maximum depth, without pruning, based on the bootstrapped samples and random feature subsets.
Averaging:

During prediction, each decision tree in the forest independently predicts the target value of the instance.
The final prediction is obtained by averaging the predictions of all trees.

Example (Random Forest Regressor):

Suppose you have a dataset containing information about various characteristics of houses (e.g., size, number of bedrooms, location) and their corresponding sale prices. You want to build a model to predict the sale price of houses based on these features.

Size (sq. ft.)	Bedrooms	Location	Sale Price ($)
1500	3	Suburb	250,000
2000	4	City	400,000
1800	3	Rural	220,000
2200	4	Suburb	350,000
...