Ensemble techniques

In machine learning Ensemble techniques are broadly divided into two categories.

1. Bagging

Bagging is also known as bootstrap aggregation is the ensemble learning method that is commonly used to reduce variance within a noisy dataset.

Imp Note:- the random forest algorithm is considered an extension of the bagging method.

It utilizes both bagging & feature randomness to create an uncorrelated forest of decision trees.

2. Boosting

Boosting is an Ensemble technique which focus on improving the model performance by reducing the error made by previous model.

Note:- Boosting are divided into
Types of boosting :) AdaBoost.

2) Gradient boosting
3) XG Boost (or) Extreme gradient boosting.

Note :- Ensemble means Group of tree.

(1) Bagging

- Bagging is one of the Ensemble techniq -ue which construct multiple base learners / Estimators using "Boot strap Aggregation".

- Question :- What is Boot strap?
  A:- Boot strap refers to a resampling technique that involves generating multiple dataset through random sampling with replacement from a given dataset.

- In bagging, Bootstrap means ?:- row sampling with replacement.

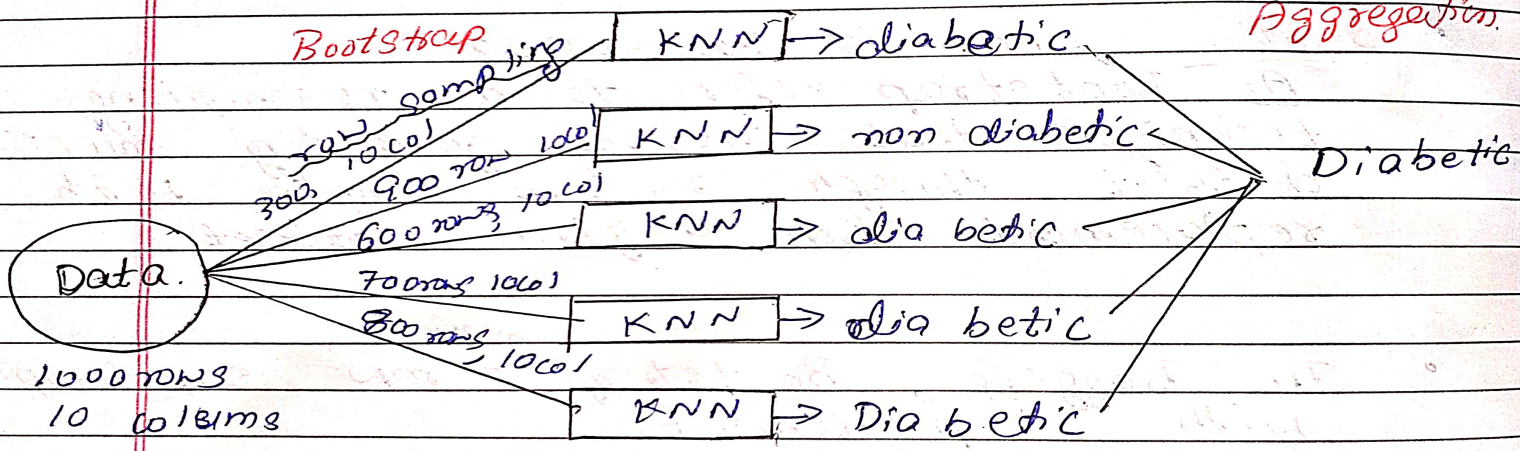
- To reduce the variance of multiple models we use bagging.

- Bagging is also known as bootstrap aggregation.

- It comes under Supervised learning
- It is used to solve both Regression and classification problem.

- Base learners, also known as weak learners refers to the individual models or algorithms used in ensemble learning methods

These base learners are combined to form a stronger and more accurate ensemble model. Each base learner by itself may not be highly accurate but when combined with other base learners, they contribute to the overall predictive performance of the ensemble.



Hyper parameter in Bagging

- Estimator :- Logistic (KNN) SVM
- no of Estimators :- no of base learners
- bagging is recommended technique to apply mainly on classification problems.

Ensemble :-

(means group of Trees)

(1) Random Forest (2) Bagging (3) Boosting

→ Ensemble

(1) Random Forest :-

- It comes under supervised Learning.
- It is used to solve both regression & classification problem.
- Random forest is an ensemble technique which builds multiple decision tree parallely using "Bootstrap Aggregation"

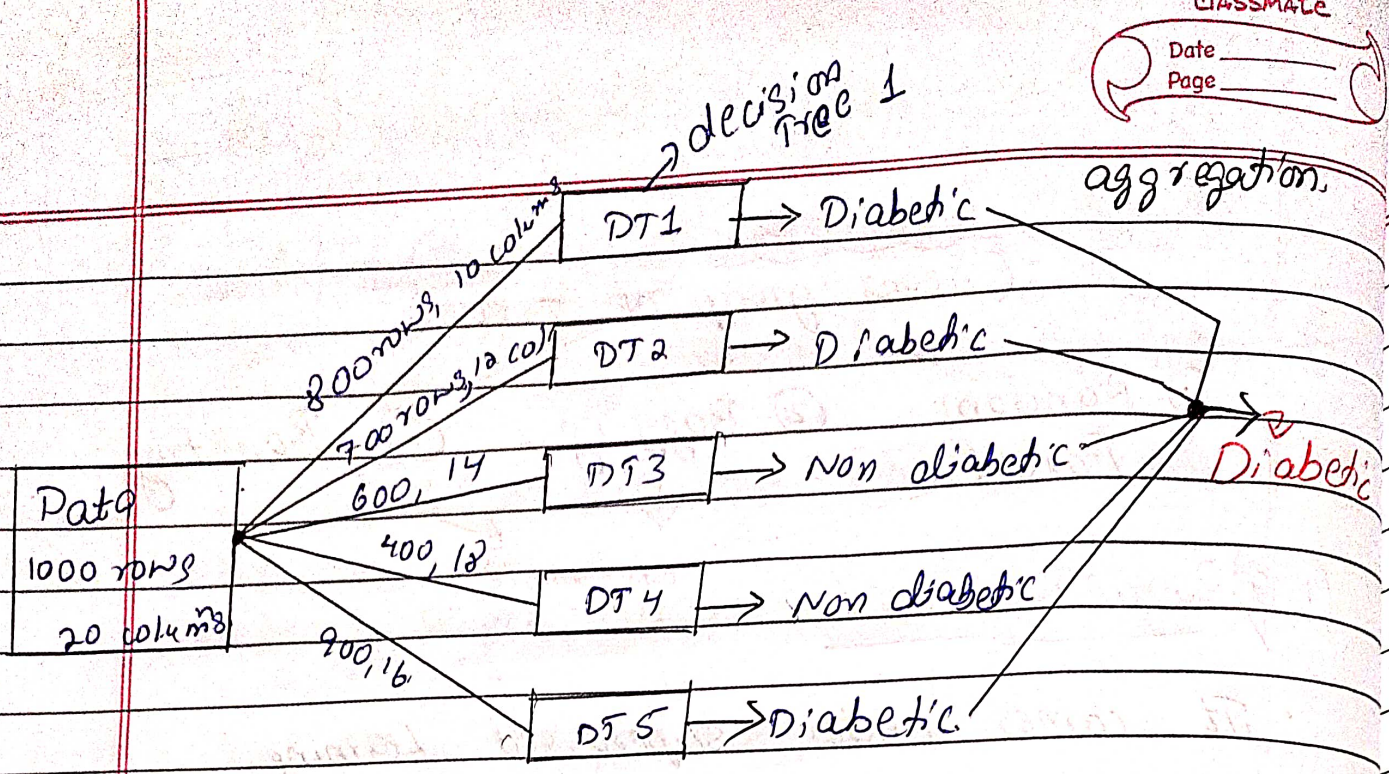
Bootstrap → row sampling + column sampling with replacement in random forest.

Eg:

1) 100 rows, 10 cols → 50 rows, 5 columns (sampling without replacement.)
→ 50 row, 5 columns

2) 1000 rows, 20 cols. → (with replacement) 50 rows, 10 columns (sampling with replacement.)
→ 70 rows, 12 columns.

Note:- Bootstrap + means usually sampling with replacement.



Aggregation → Majority voting → [Classification]

Aggregation → Average → [Regression]

hyperparameter Techniques

(1) **Grid search CV:-**

It will Evaluate model for all the possible combination and select best parameters

(2) **Randomized search CV:-**

It will Evaluate the model on specified combination which are selected randomly and gives best parameters.

hyperparameter tuning in Random Forest:

1) n_estimators :- no of decision trees.

2) max_features :- refers to the maximum number of features that are considered when making a split in a decision tree @ random forest model.

3) bootstrap :- [True, False]

True :- sampling with replacement
False :- sampling without replacement.

4) max depth
5) min sample split
6) min sample leaf.

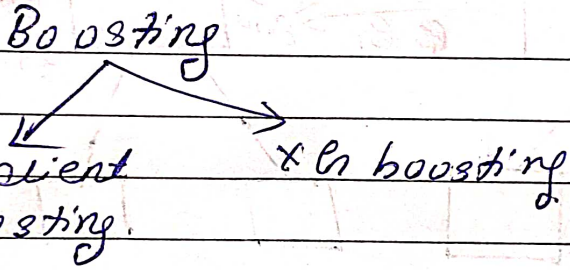
already explained in decision Tree.

Note:-

n_iter :- • It takes no of combinations that has to be selected.
• used in randomized hyperparameter optimization @ random search CV.

(2) Boosting :-

- It comes under supervised learning
- It can be used to solve Regression & classification
- Boosting is an Ensemble technique which focus on improving the model performance by reducing the Error made by previous model
- Boosting algorithm are very accurate and better models to make prediction

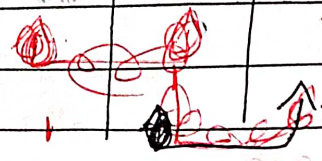


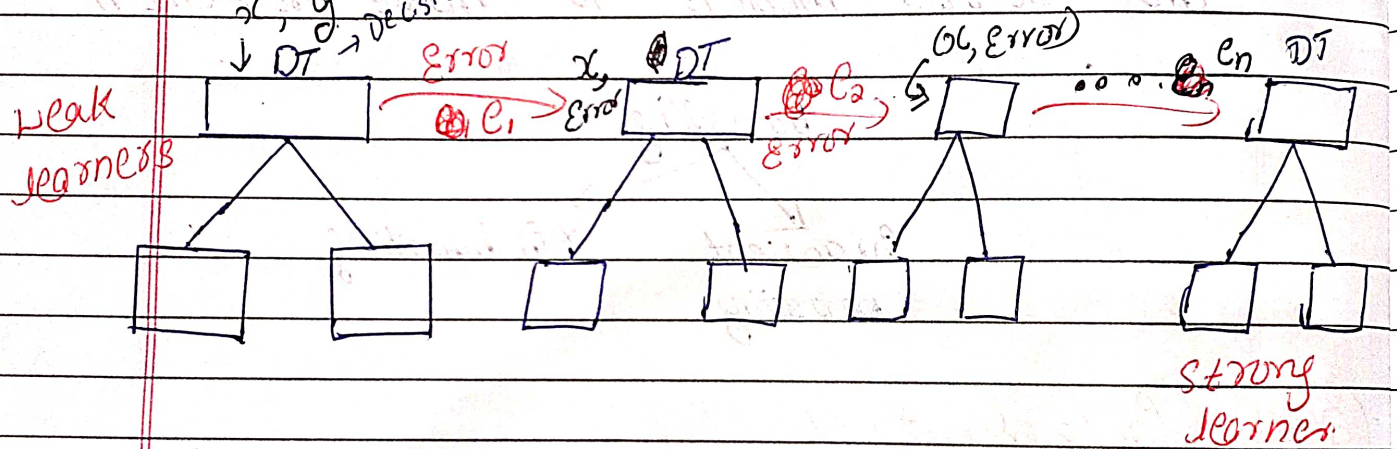
(2a) Gradient boosting :-

- It is the Ensemble technique which comes under Boosting.
- Gradient Boosting ~~struct~~ construct multiple weak learns sequentially which focus on improving the model performance by reducing the mistake made by the previous weak learner.

- Goal is to reduce the mistake made by the privious base learner to find the best possible strong model.

x	y	$\hat{y}$	Error	$\hat{y}$	Error	....
2	7	10	3			repeting
3	8	12	4			till
4	9	15	6			you
5	10	17	7			get
6	11	20	9			a
						less
						error





hyper parameter in Boosting Gradient

boosting mainly are :-

1) learning rate
2) n_estimator
3) max_depth
4) sub sample :- The fraction of samples to be used for fitting each base learner. It helps introduce randomness and reduce overfitting

Note :- We don't prefere using hyperparameters to gradient boosting rather we go for xgboosting

Date
Page

(2) (b) XGBoosting (or) Extreme Gradient Boosting

- XGBoost is an optimized gradient boosting machine learning algorithm that is widely used for regression and classification Task.
- It stands for "Extreme Gradient Boosting" and is known for its speed, performace and scalability

some important points

1. Gradient Boosting:- XGboost belongs to the family of Gradient boosting algorithms, which combine weak predictive models (typically decision trees) to create a stronger ensemble model.

It iteratively adds new models to correct the Errors made by previous models.

2. Boosting Iterations:- XGboost builds an Ensemble of decision trees iteratively. In each boosting iteration, a new decision tree is added to the Ensemble.

The tree is trained to minimize the overall loss of the Ensemble, using a techni-que called gradient boosting.

3. Regularized Learning objective:- XGboost uses a regularized learning objective

function that consists of a loss function and a regularization term.

The loss function measures the model's performance while regularization term controls the complexity of the model.

Regularization helps prevent overfitting and improves the model's ability to generalize to unseen data.

4) Feature Importance:-

Xgboost provides a measure of feature importance, which indicates the relative importance of each feature in the prediction task.

5) Regularization Techniques:-

Xgboost offers several techniques to control model complexity & prevent overfitting.

It includes L1 & L2 regularization (similar to ridge & lasso regression).

Trees pruning to remove unnecessary split. and a shrinkage parameter (also known as learning rate) to control the contribution of each tree to the ensemble.

6. Parallel processing :-

XGboost is designed to work faster by doing multiple things at once.

It can use multiple CPU core on a computer to train and make prediction more quickly.

This means it can process data in parallel, dividing the work among different cores, which speeds things up.

7. Handling missing values :-

XGBoost can handle missing values in the data. It designed in a such a way that it can handle missing values.

8. Cross validation :-

We use cross validation by importing the function from sklearn but xgboost is enabled with inbuilt CV function.

9. Regularization :-

This is considered to be as a dominant factor of the algorithm.

Regularization is a technique that is used to get rid of overfitting of the model.

10. Wide Range of Applications :-

xgboost has been successfully applied to various ML tasks, including regression, classification, raking, and recommendation systems. It has won

numerous data science competitions & is widely adopted in both industry and academia.

Overall, Xgboost is versatile algorithm that combines the strengths of gradient boosting with regularization techniques and parallel processing. It offers efficient and accurate solutions for a wide range of machine learning problems.

hyperparameters in Xgboosting

1) Gamma:- It controls how much a tree's branches can be further split.

Eg:- ~~The values~~ range ~~from 0 to 200~~, with higher values being more conservative.

- In simple terms, 'gamma' determines how much improvement in the loss function is required for a split to occur.
- higher gamma value will be create new split.
- It is used to reduce overfitting
- Gamma decides minuminimum reduction in loss to split further.

2) learning rate:- It determines how fast the model learns from the data.

- The learning rate in xgboost determine how much the model learns from each round of training.

→ higher learning rate means the model learns faster, but it may also risk overshooting (poor performance) the best solution.

→ A lower learning rate makes the model learn more slowly but potentially lead to a more accurate and precise outcome.

- It is important to find the right learning rate that allows the model to learn effectively without learning too quickly @ too slowly.

3) max depth :- depth of the tree.

4) n_estimator :- number of base learners.
It determine number of trees in the model

5) reg_alpha :-
6) reg_lambda :- → Regularization parameter used to reduce overfitting

→ In simpler term, higher the values of these parameters make the model simpler and less prone to overfitting.