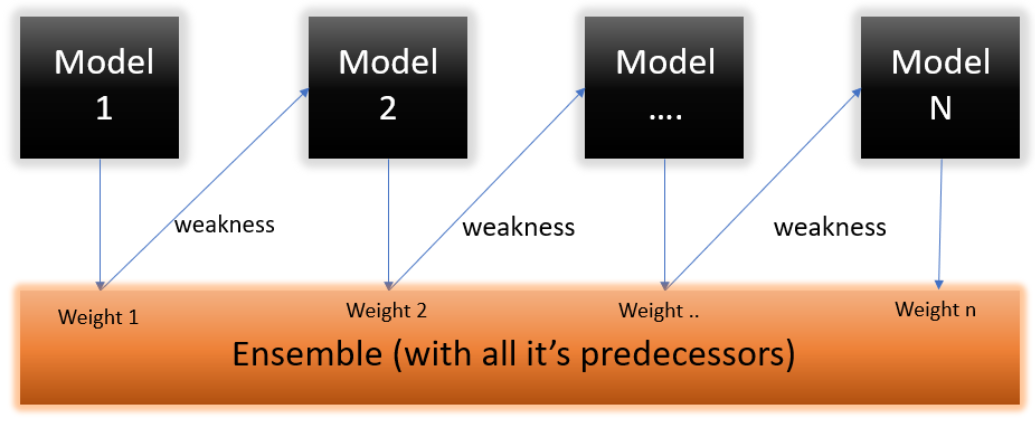
|  |  |
| --- | --- |
| SR | Title |
| Q1 | Explain Adaboost/ Any ML Algorithm? |
|  | * Supervised/ Unsupervised |
| * Parametric/ Non parametric/Assumptions |
| * Effect of outliers |
| * Effect of scaling |
| * Hyper parameter tunning |
| * Evaluation measures: r2/cnf\_matrix, ROC/AUC |
| Use above points to explain any algorithm if interviewer ask to explain | |

**Adaboost Algorithm**

Adaboost is a supervised machine learning algorithm and it is based on boosting ensemble technique. In boosting, all the weak learners are connected in series. Each weak learner learns from the mistakes of his predecessor. The most common weak learner or base model used in Adaboost is Decision Stump. Decision stump are nothing but decision tree with one split.

In Adaboost, we build a model and gives equal weights to all the data points. It then assigns higher weights to points that are misclassified. Now all the points which have higher weights are given more importance in the next model. It will keep training models until and unless a low error is received.

**Working of Adaboost**

Step1:

We build the first model and train on dataset for predictions. The first model also assigns equal weight to each record and calculates error.

Weight = 1/no of records/samples

Error, e = sum of weights of misclassified records.

Learning rate,

Step2:

Identify the record for which our base model has correct and wrong predictions and adjust their weights using following formula

Correct weight,

Incorrect weight,

Step3:

Normalized the weight and create range from normalized weights. Generate random number equals to no of records between 0 and 1. Based on this random number selects the sample/ record to create a new dataframe/ dataset.

Step4:

Use this newly created dataset and initial weights for second base model. Make predictions and identify correctly classified and misclassified records and update their weights. Follow the same procedure as in steps 1 to 3

Step5:

When error reduced to zero, stop and make final predictions. The final prediction is nothing but weighted sum of all the predictions made by all the models.

**Effect of Scaling on Adaboost Algorithm**

Algorithms Like Decision Trees, Random Forest, Adaboost, Gradient Boosting, etc. are not significantly affected by Feature Scaling Since the trees in these algorithms are constructed based on conditions and are not dependent on the Range of values. Algorithms like Linear Discriminant Analysis and Naive Bayes Also do not require feature scaling.

**Effect of Outlier on Adaboost Algorithm**

Outliers can be bad for boosting because boosting builds each tree on previous trees' residuals/errors. Outliers will have much larger residuals than non-outliers, so gradient boosting will focus a disproportionate amount of its attention on those points.

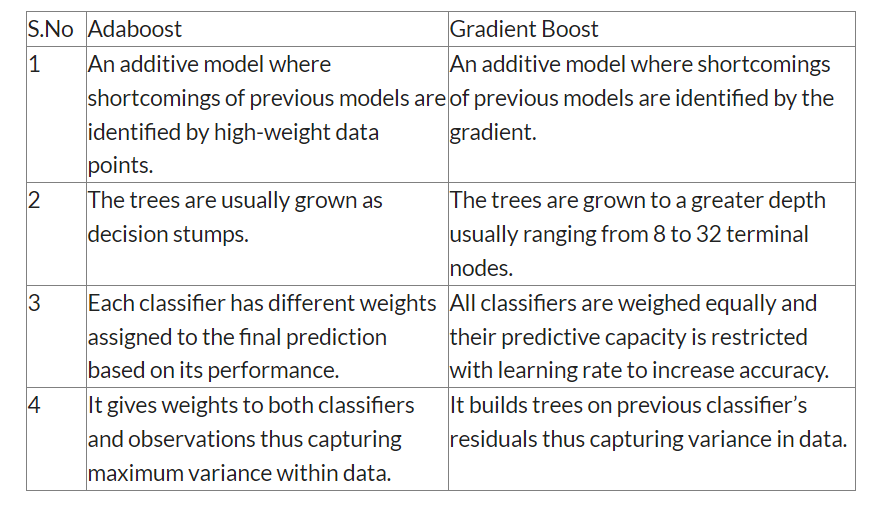
**Adaboost Hyper Parameters**

**Base Estimator:** By default, Decision tree with single split. Apart from DT, any other algorithm can also be used as base model. Usually, Adaboost with decision stump gives best results.

**No of Estimators:** By default, no of estimators are 50. But Adaboost is enough smart to quiet if error reduced to zero before building 50 base models.

**Learning Rate:** Weight applied to each classifier at each boosting iteration. A higher learning rate increases the contribution of each classifier

**Algorithm:** The SAMME.R algorithm typically converges faster than SAMME, achieving a lower test error with fewer boosting iterations.

**Adaboost Vs Gradient Boost**