Unit 2.6
Ensemble Learning

Disclaimer

The content is curated from online/offline resources and used for educational purpose only

Ensemble Learning

Suppose you are planning to Buy

Visit Showroom → Salesman Guide you → ?

Opinion from friends

Opinion from families

Customer review on websites

Final Decision

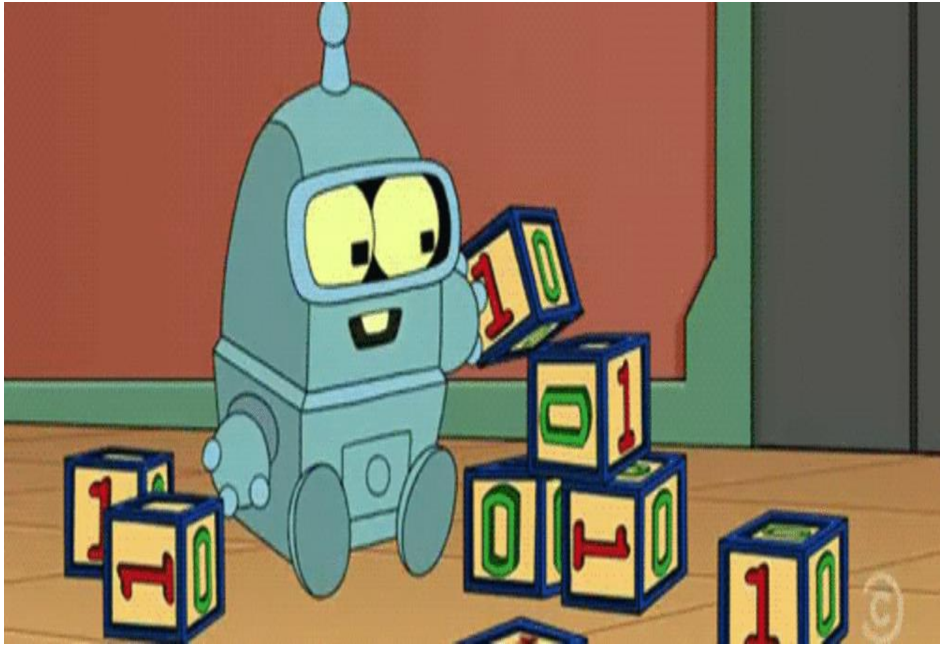
Learning Objectives

You will learn in this lesson:

- Introduction to Ensemble Machine Learning
- Types of Ensemble Machine Learning
    - Bagging
    - Boosting
    - Stacking
- How Gradient Boosting works?
- How Xgboost works?

Introduction of Ensemble Machine Learning

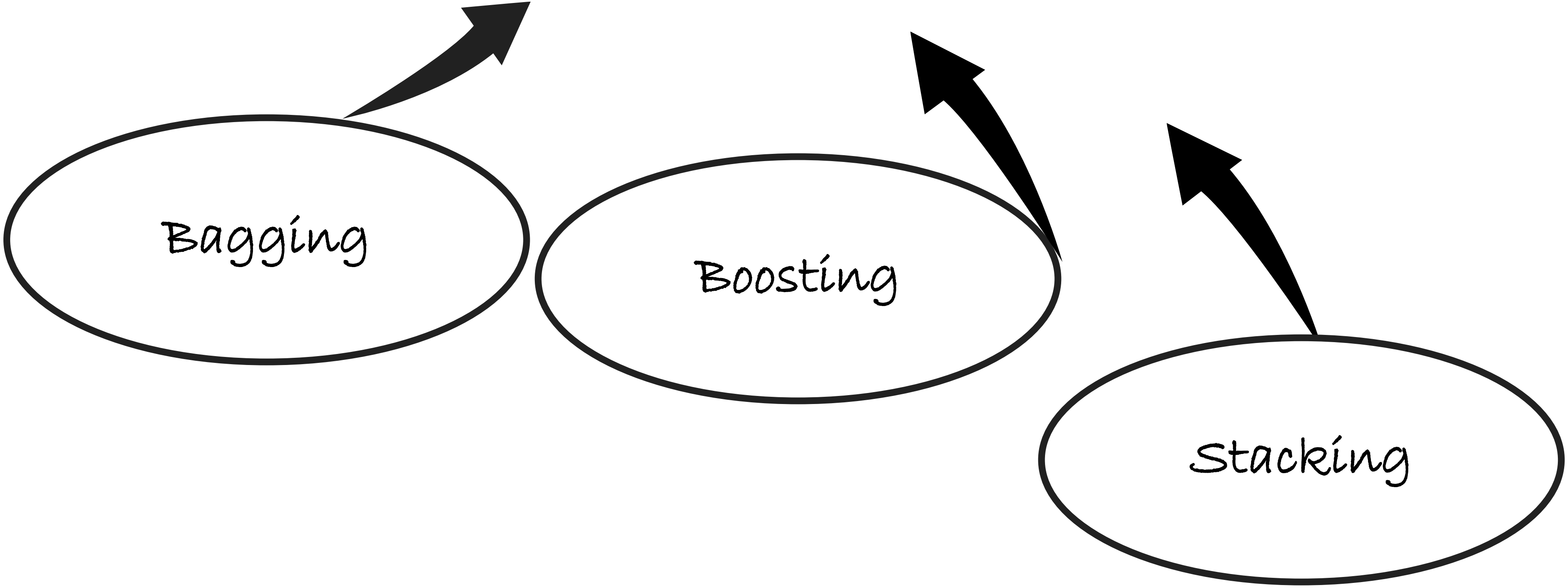
- Literary meaning of the word ‘ensemble’ is group.
- These methods involve a group of predictive models.
- Achieve better accuracy and model stability.
- Noise, variance and bias are the major sources of error.
- Ensemble learning is one way to tackle bias-variance trade-off.



Reference

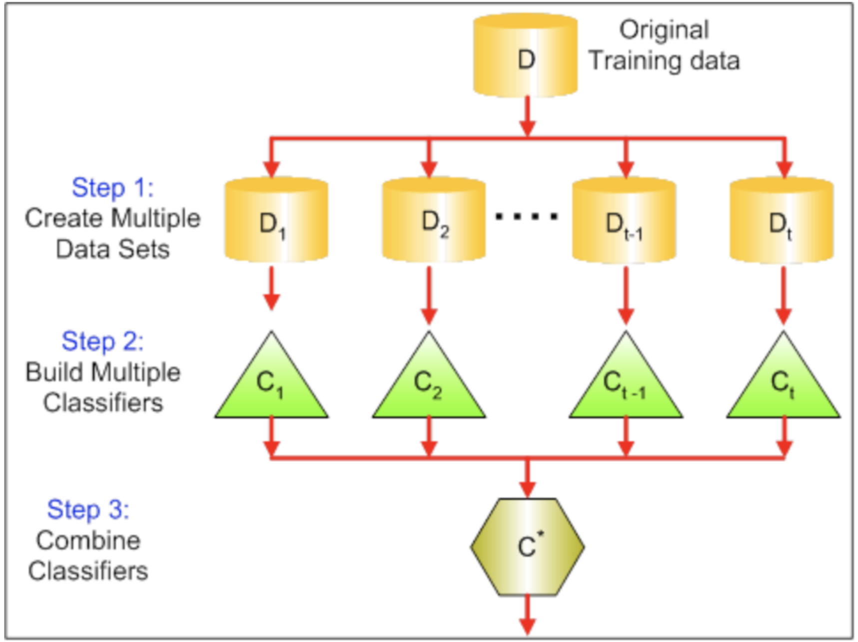
Types of Ensemble Learning

There are various ways to ensemble *weak* learners to come up with *strong* learners:



Bagging

- An ensemble technique used to reduce the variance
- Combines the result of multiple classifiers modeled on different sub-samples of the same data set.
- **Example- Random Forest**



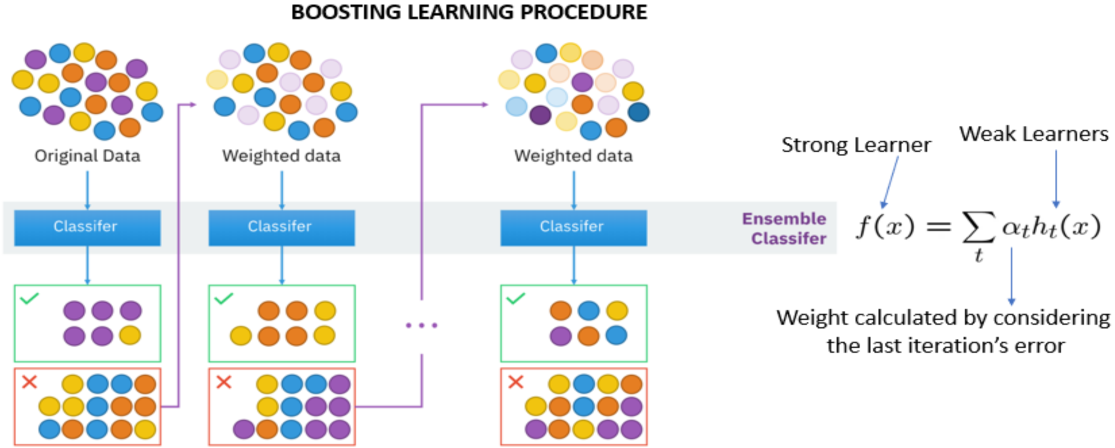
Reference

Boosting

- Boosting fit a sequence of weak learners.
- Include models that are only slightly better than random guessing
- More weight is given to examples that were misclassified by earlier rounds

Prominent Boosting techniques:

1. Gradient Boosting Machine
2. XGBoost
3. AdaBoost
4. LightGBM
5. CatBoost

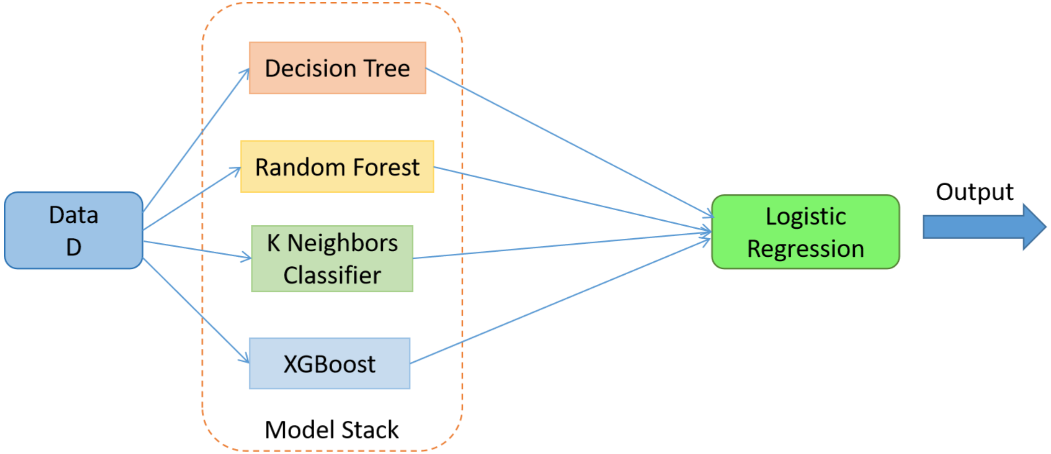


Reference

Stacking

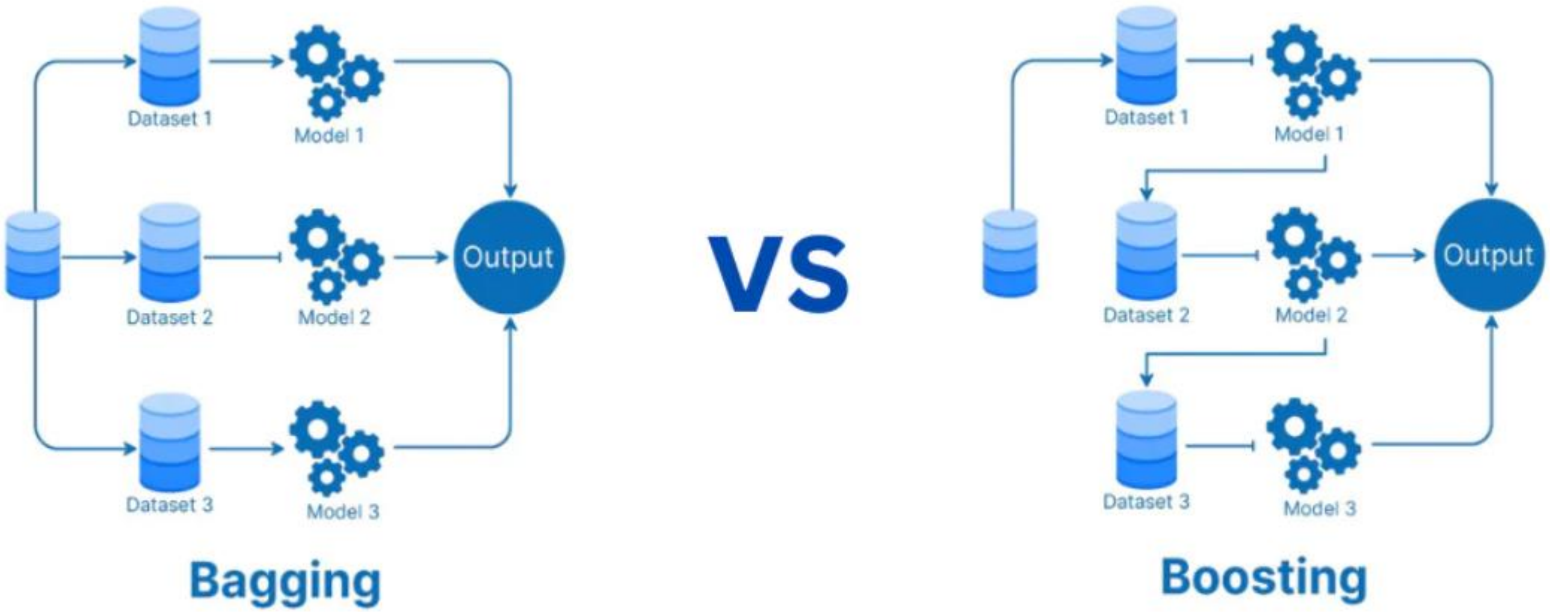
- **Stacking:** an ensemble technique.
- Uses a meta-learning algorithm to learn.
- Best combine the predictions from two or more base machine learning algorithms.

In stacking, a single model is used to learn how to best combine the predictions from the contributing models (e.g. instead of a sequence of models that correct the predictions of prior models).



Reference

Bagging vs. Boosting



[Reference]

Random Forests

- As in bagging, we build a number of decision trees on bootstrapped training samples.
- Each time a split in a tree is considered.
- A random sample of *m* predictors is chosen as split candidates from the full set of p predictors.
- Note that if $m = p$, then this is bagging.

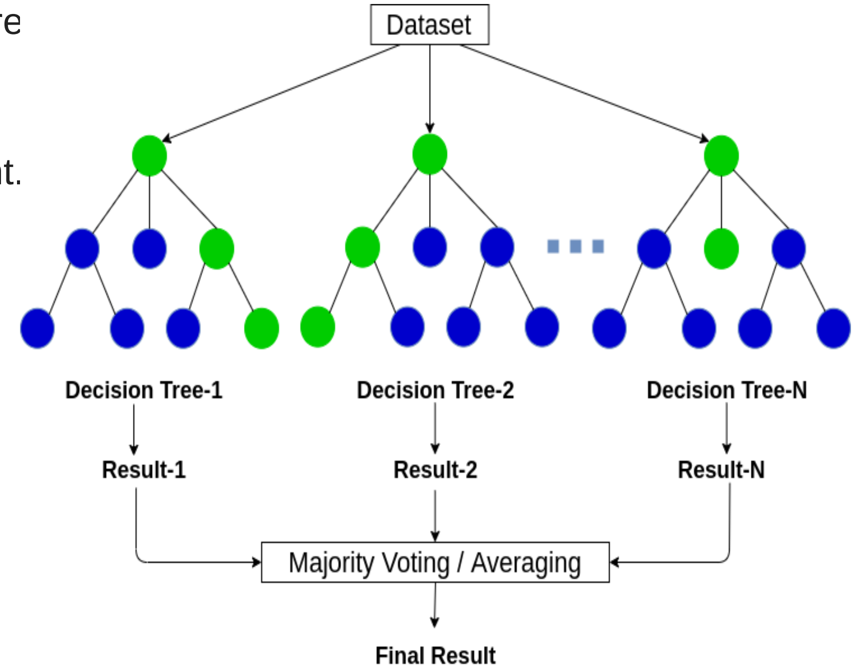
Reference

Random Forest Algorithm

- Chooses a random sample/random subset from the entire data set.
- Each model is generated from the samples (Bootstrap Samples) provided by the Original Data with replacement.
- This step of row sampling with replacement is called ***bootstrap***.
- Each model is trained independently, which generates results.
- To make a prediction at a new point *x* we do:

    For regression: average the results

    For classification: majority vote



Reference

Random Forests Tuning

Suggested recommendations:

- For classification, the default value for *m* is $\sqrt{p}$ and the minimum node size is one.
- For regression, the default value for m is *p/3* and the minimum node size is five.
- In practice the best values for these parameters will depend on the problem, and they should be treated as tuning parameters.
- Like with Bagging, we can use OOB(**Out-of-Bag**) therefore, RF can be fit in one sequence, with cross-validation being performed along the way.
- Once the OOB error stabilizes, the training can be terminated

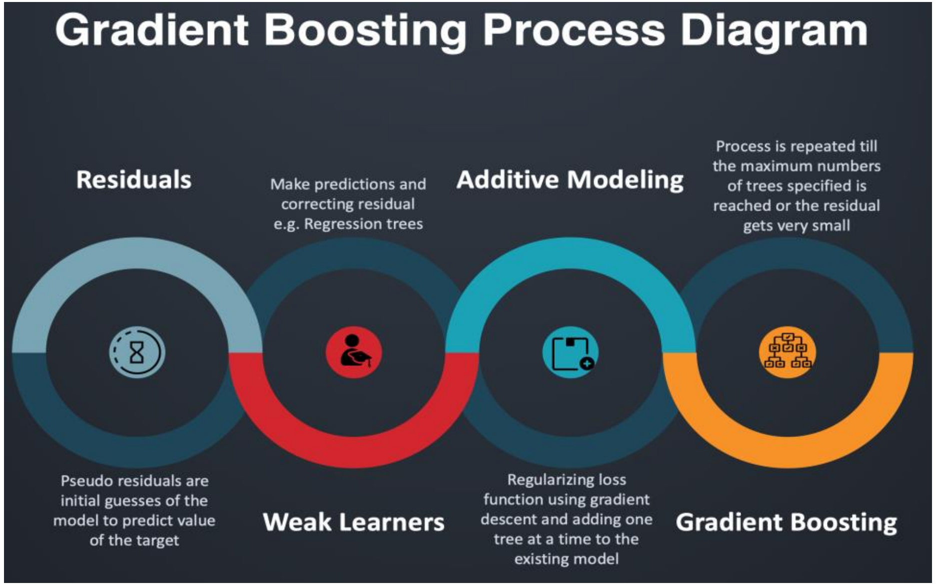
Random Forests Issues

- When the number of variables is large, but the fraction of relevant variables is small, random forests are likely to perform poorly when *m* is small , **Why?**
- Because At each split the chance can be small that the relevant variables will be selected
- For example, with 3 relevant and 100 not so relevant variables the probability of any of the relevant variables being selected at any split is ~0.25
- Random forests “cannot overfit” the data with respect to number of trees. **Why?**
- The number of trees, does not mean increase in the flexibility of the model

Lab 1: **Demonstrating Random Forest Algorithm on Credit Card Dataset**

Gradient Boosting Machine

- Boosting does not involve bootstrap sampling
- Trees are grown sequentially: each tree is grown using information from previously grown trees
- Like bagging, boosting involves combining a large number of decision trees, $f^1, \ldots, f^B$.



Reference

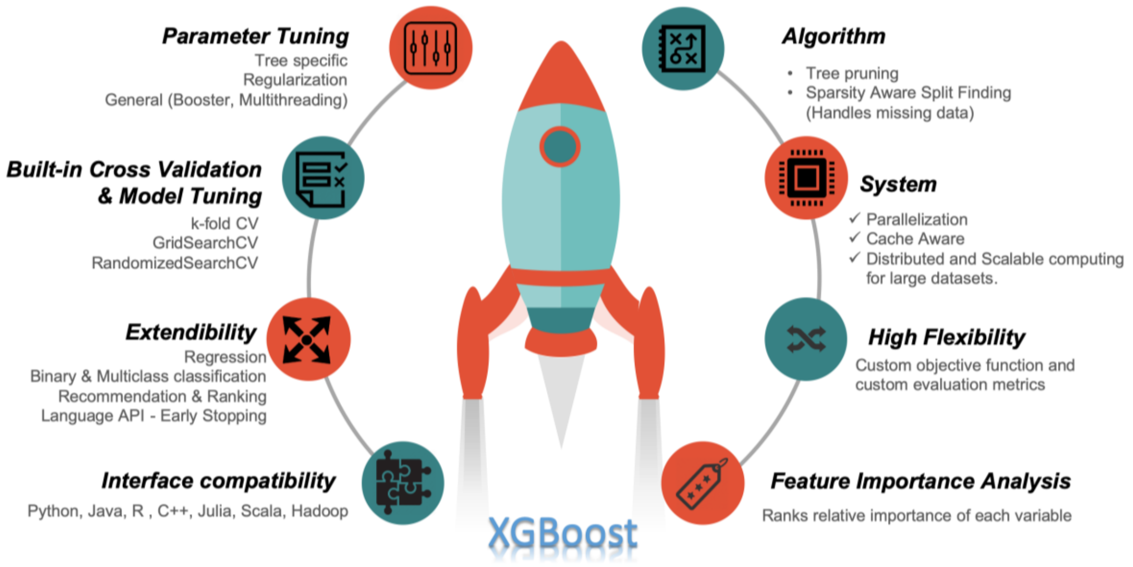
Sequential Fitting

- Given the current model,
- we fit a decision tree to the **residuals** from the model. Response variable now is the residuals and not Y
- We then add this new decision tree into the fitted function in order to update the residuals
- The learning rate has to be controlled

XGBoost

Extreme Gradient Boosting (XGBoost) is just an extension of gradient boosting with the following added advantages:

- **Regularization**
- **Parallel Processing**
- **High Flexibility**
- **Handling Missing Values**
- **Tree Pruning**
- **Built-in Cross-Validation**
- **Continue on Existing Model**



Reference

Lab 2: [Demonstrating XG Boost on Credit Card Dataset]

Summary

- Bagging is a technique for improving the accuracy of predictions made by machine learning models. Bagging works by constructing a number of different models, each of which is based on a randomly-selected subset of the training data.
- Boosting work by combining a number of weaker models into a stronger one.
- Random forest is an ML ensemble algorithm based on aggregation of several decision trees. It is accurate, efficient, and relatively quick to create.
- Random Forest overcomes the drawbacks of the decision tree algorithm by reducing the overfitting of the dataset and improving accuracy.
- XGBoost algorithm is an extended version of the gradient boosting algorithm. It is basically designed to enhance the performance and speed of a Machine Learning model.

Quiz

1. **Which of the following is true about Bagging?**

a) Bagging is a type Sequential ensemble learning.

b) The main aim of bagging is to reduce bias.

c) The bagging helps to reduce overfitting.

d) All

Answer) c

Quiz

2. Which of the following algorithm are not an example of ensemble learning algorithm?

a) Random Forest.

b) AdaBoost.

c) Gradient Boosting.

d) Decision Tree

Answer) d

Quiz

3. Which of the following is true about Boosting?

a) Boosting is a type Sequential ensemble learning.

b) The main aim of boosting is to reduce bias.

c) The boosting helps to reduce overfitting.

d) Both a and b

Answer) d

Quiz

4. Random Forest has __________ as base learning models?

a) Multiple Decision tree.

b) Meta learner.

c) Bagging.

d) None

Answer) a

Reference

- https://uc-r.github.io/gbm_regression
- https://www.scaler.com/topics/machine-learning/random-forest-algorithm/
- https://www.v7labs.com/blog/ensemble-learning
- https://www.upgrad.com/blog/random-forest-hyperparameter-tuning/
- https://scikit-learn.org/stable/modules/ensemble.html
- https://www.upgrad.com/blog/random-forest-hyperparameter-tuning/
- https://www.simplilearn.com/tutorials/machine-learning-tutorial/random-forest-algorithm

Thank you...!