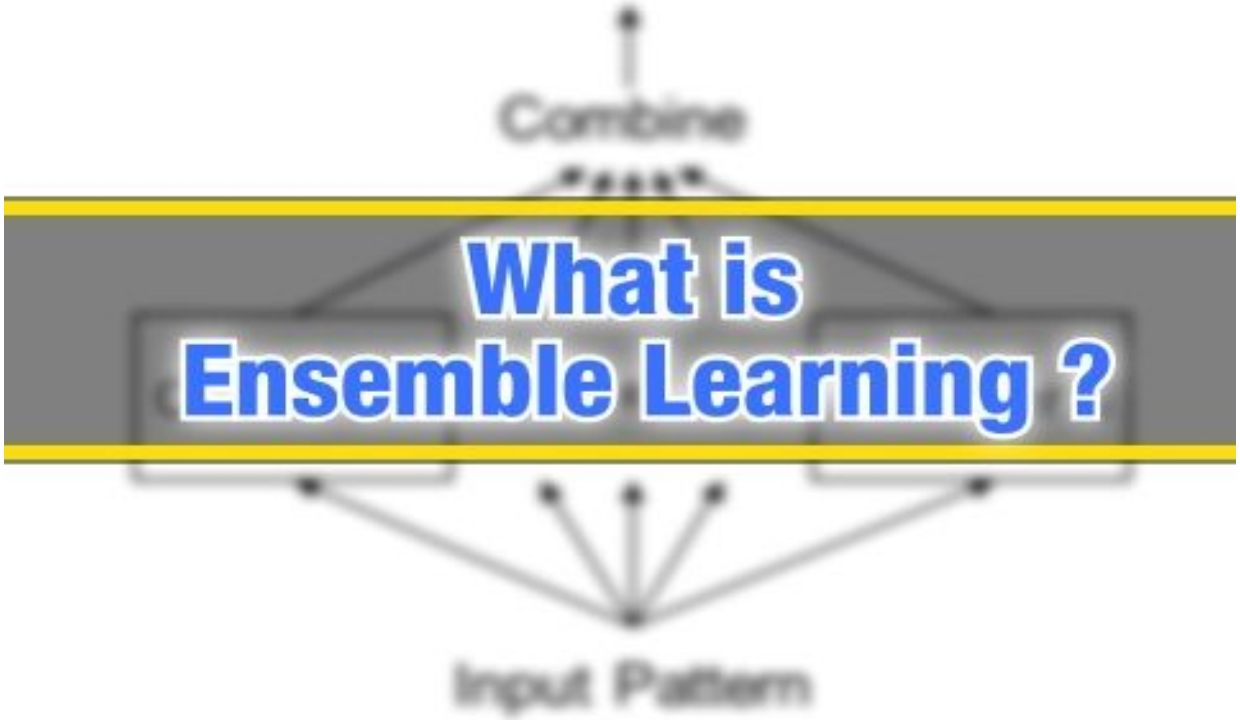
INTERNSHIPSTUDIO

Ensemble Learning Techniques



Agenda

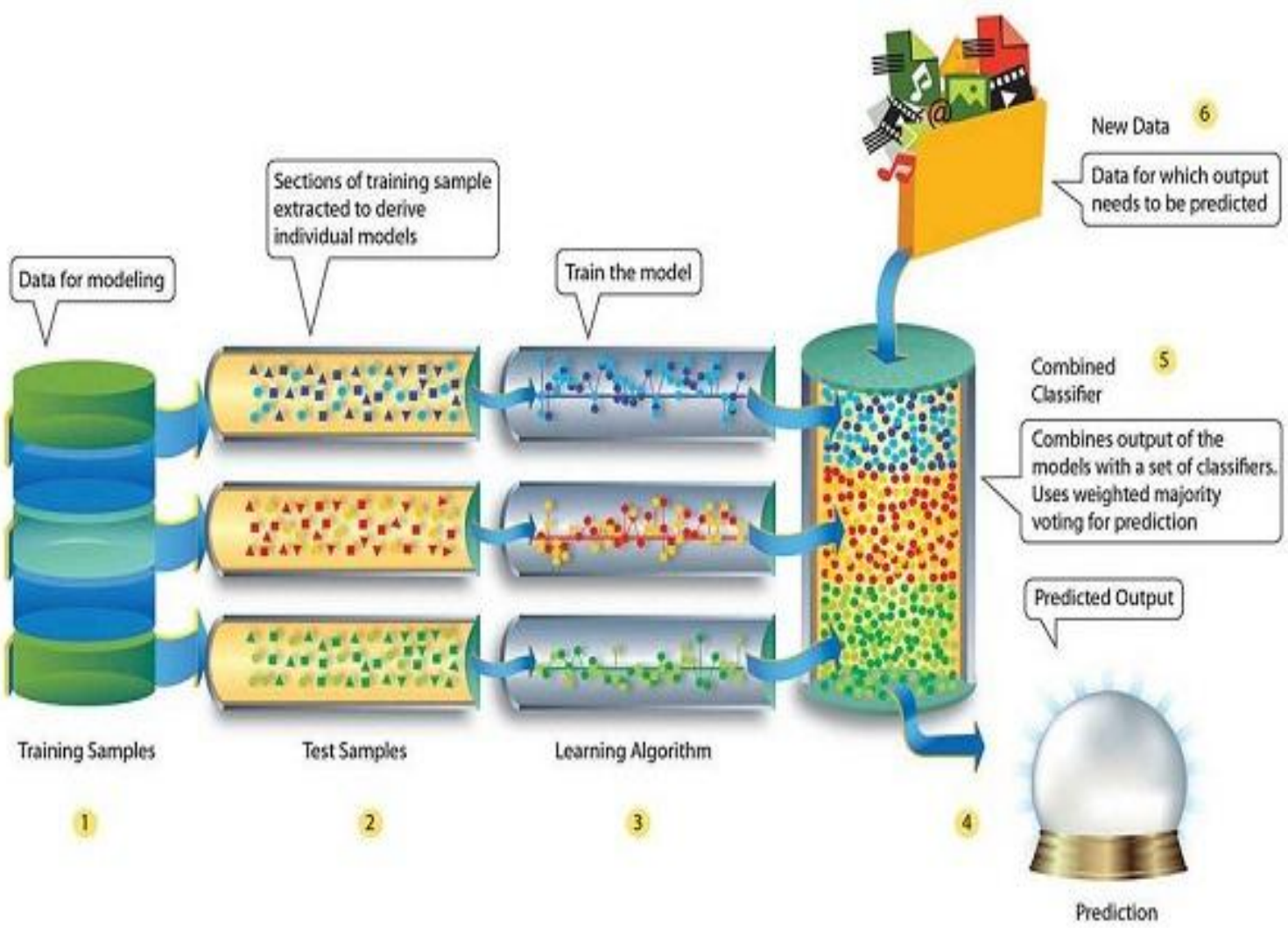
INTERNSHIPSTUDIO

- Ensemble Learning
- Boosting
- Gradient Boosting and XGBoost
- Overfitting/Underfitting
- How to address Overfitting/Underfitting

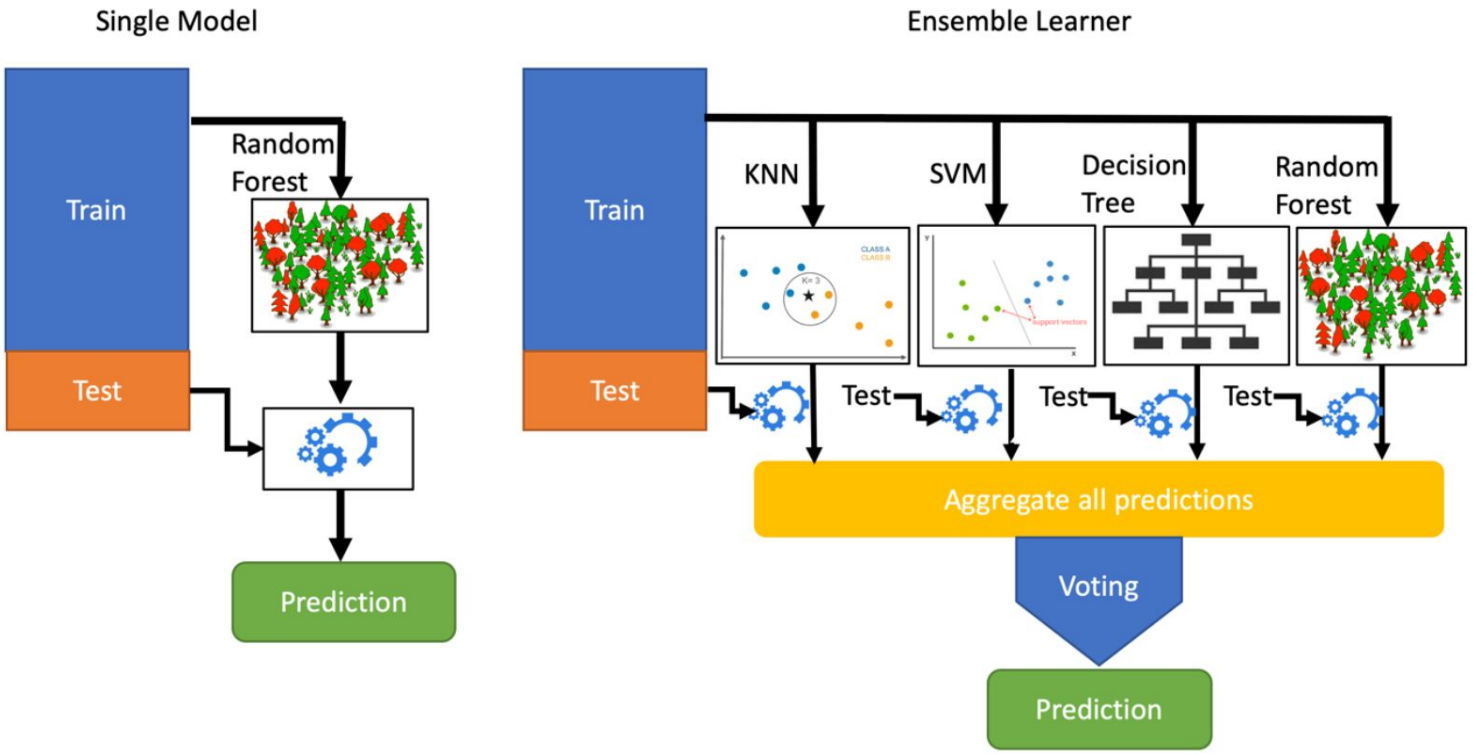
Ensemble Learning

- **Ensemble methods** is a machine learning technique that combines several base models in order to produce one optimal predictive model.
- The process of generating models from data is called learning or training and the learned model can be called as hypothesis or learner.
- This type of machine learning algorithm helps in improving the overall performance of the model.
- The learning algorithms which construct a set of classifiers are known as Ensemble methods.

Ensemble Learning



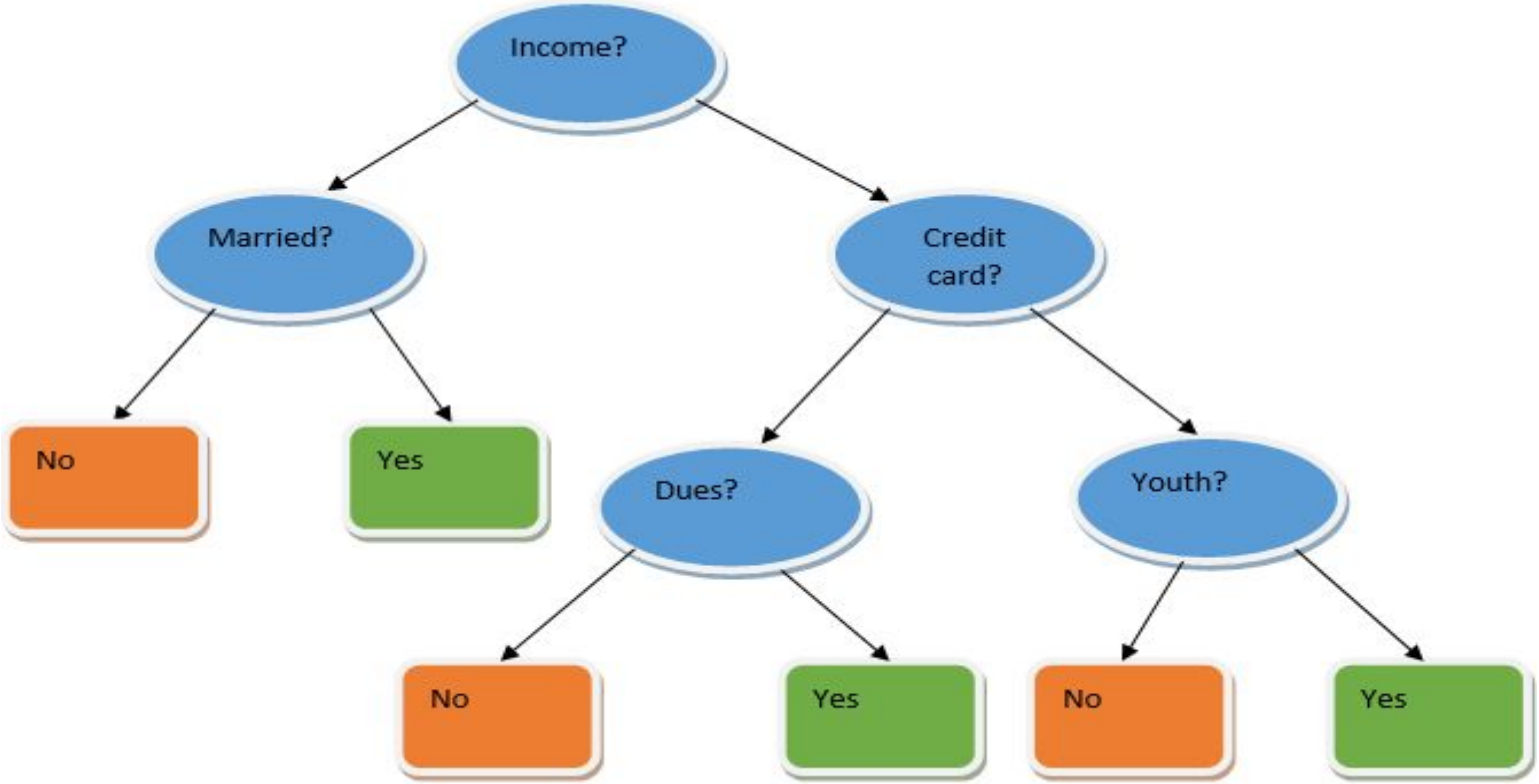
Single Model Prediction vs Ensemble Learner



Why Ensemble Methods?

INTERNSHIPSTUDIO

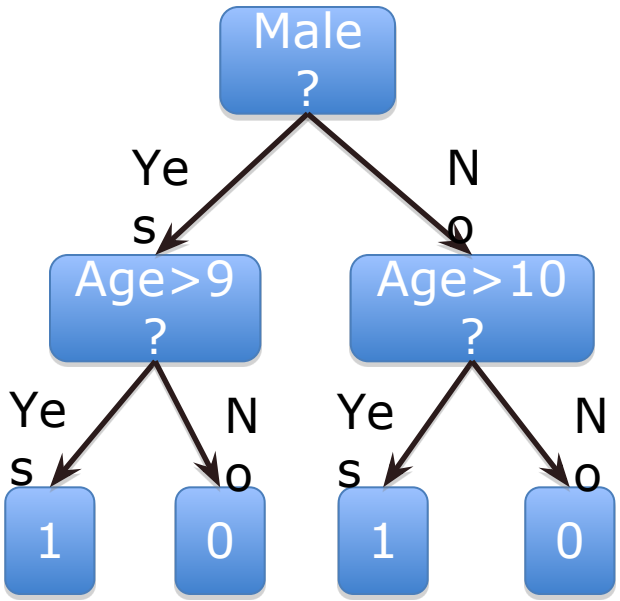
- A diverse set of models in comparison to single models are likely to make better decisions.
- A decision tree basically works on several rules and provides a predictive output, where the rules are the nodes and their decisions will be their children and the leaf nodes will constitute the ultimate decision. The example of a decision tree below about a bank loan decision.



One classifier is not enough!

- Performance
  - None of the classifiers is perfect
  - Complementary
    - Examples which are not correctly classified by one classifier may be correctly classified by the other classifiers
- Potential Improvements
  - Utilize the complementary property

An EXAMPLE



Name	Age	Male?	Height > 55”
Alice	14	0	1
Bob	10	1	1
Carol	13	0	1
Dave	8	1	0
Erin	11	0	0
Frank	9	1	1
Gena	8	0	0

$$x = \begin{bmatrix} age \\ 1_{[gender=male]} \end{bmatrix}$$

$$y = \begin{cases} 1 & height > 55" \\ 0 & height \leq 55" \end{cases}$$

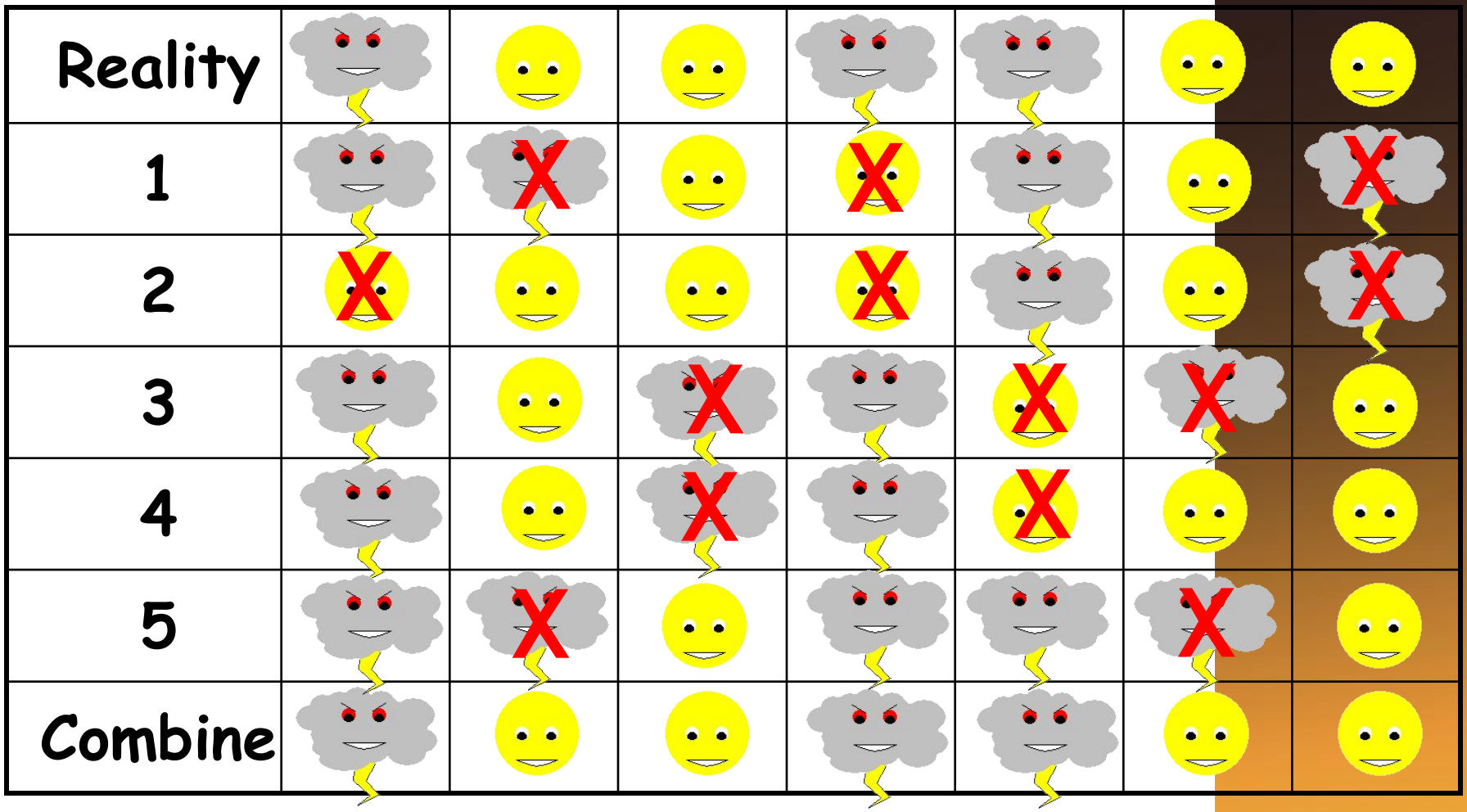
Ensembles of Classifiers

Combine the classifiers to improve the performance

Ensembles of Classifiers

- Two ways to combine the classification results from different classifiers to produce the final output
  - Unweighted voting
  - Weighted voting

Example: Weather Forecast



Type of Ensemble methods:

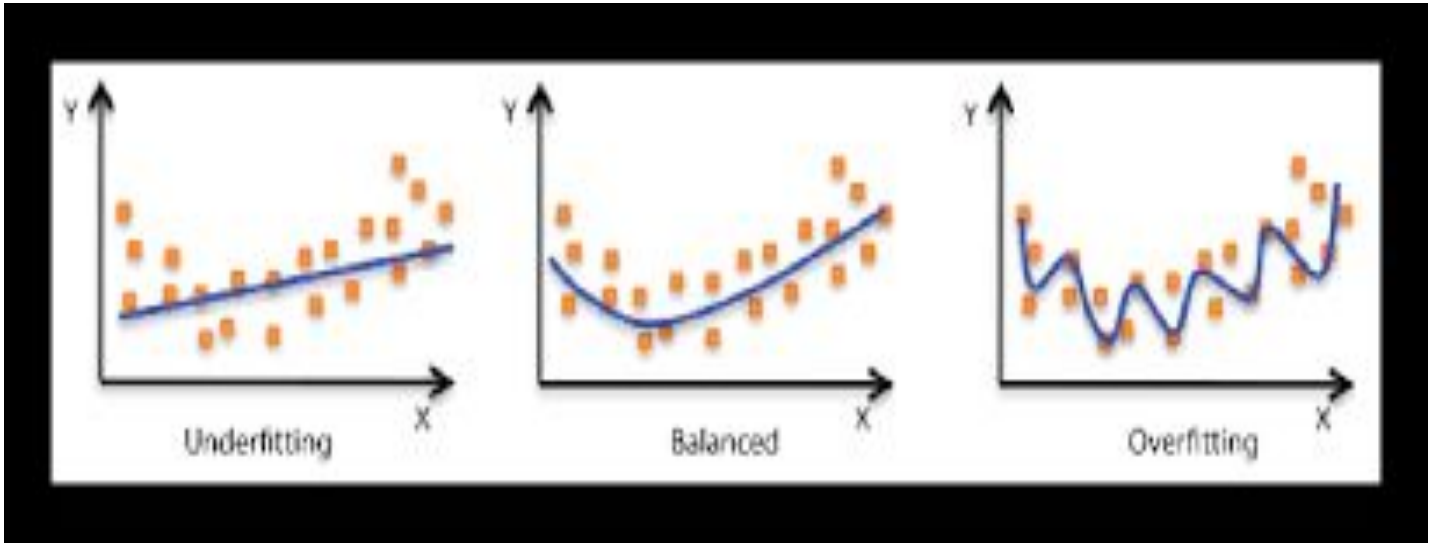
The three most popular methods for combining the predictions from different models are:

- **Bagging** Building multiple models (typically of the same type) from different subsamples of the training dataset.
- **Boosting**. Building multiple models (typically of the same type) each of which learns to fix the prediction errors of a prior model in the chain.
- Building multiple models (typically of differing types) and simple statistics (like calculating the mean) are used to combine predictions.

Bias and Variance

INTERNSHIPSTUDIO

- Bias is an error that occurs due to incorrect assumptions in our algorithm; a high bias indicates our model is too simple/underfit.
- Variance is the error that is caused due to sensitivity of the model to very small fluctuations in the data set; a high variance indicates our model is highly complex/overfit.
- An ideal ML model should have a proper balance between bias and variance.



Ensemble methods

- Ensemble methods that minimize variance
  - Bagging
  - Random Forests
- Ensemble methods that minimize bias
  - Functional Gradient Descent
  - Boosting
  - Ensemble Selection

INTERNSHIPSTUDIO

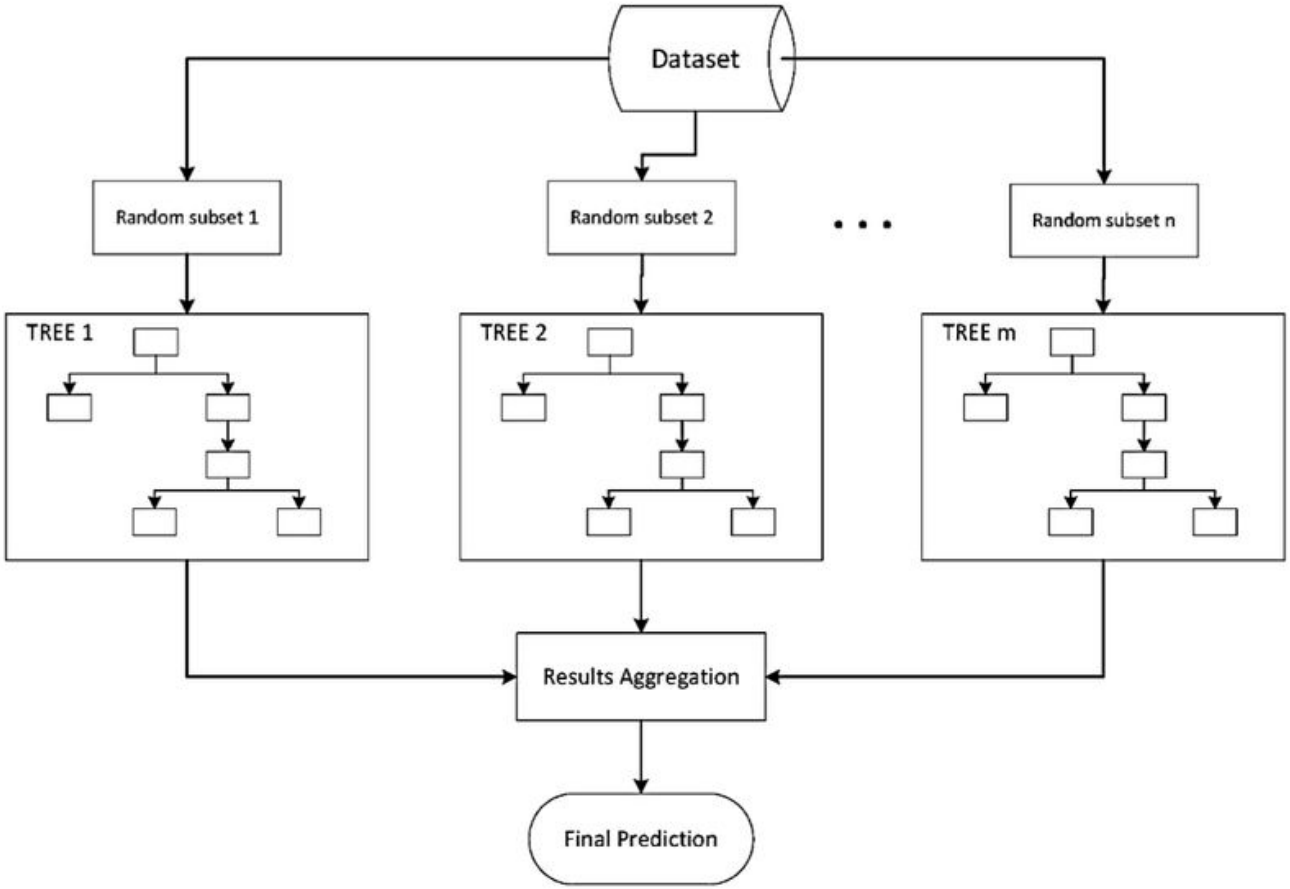
- Q.1 What is Ensemble Learning?
- Q.2 What is the need of ensemble learning in ml?
- Q.3 Why only one classifier is not enough in Machine Learning?
- Q.4 What are the types of Ensemble Methods?

Bagging

Bootstrap **AGG**regating or BAGGing gets its name because it combines Bootstrapping and Aggregation to form one ensemble model.

- Given a sample of data, multiple subsamples are pulled and a Decision Tree is formed on each of the subsamples.
- After that an algorithm is used to aggregate over the Decision Trees to form the most efficient predictor.
- Once we have a prediction from each model then use a model averaging technique to get the final prediction output.
- One of the famous techniques used in Bagging is Random Forest. In the Random forest, we use multiple decision trees.

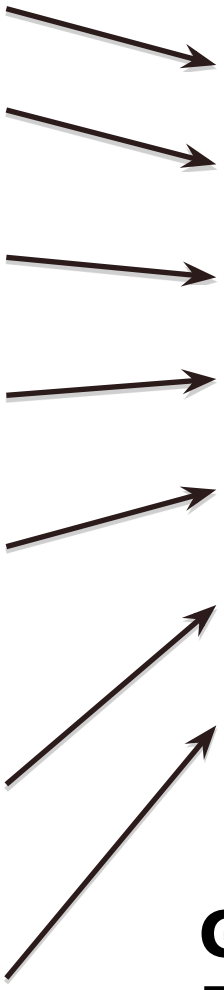
Bagging



Given a Dataset, subsamples are pulled and a Decision Tree is formed on each bootstrapped sample. The results of each tree are aggregated to yield the strongest, most accurate predictor.

Person	Age	Male?	Height > 55”
James	11	1	1
Jessica	14	0	1
Alice	14	0	1
Amy	12	0	1
Bob	10	1	1
Xavier	9	1	0
Cathy	9	0	1
Carol	13	0	1
Eugene	13	1	0
Rafael	12	1	1
Dave	8	1	0
Peter	9	1	0
Henry	13	1	0
Erin	11	0	0
Rose	7	0	0
Iain	8	1	1
Paulo	12	1	0
Margaret	10	0	1
Frank	9	1	1
Jill	13	0	0
Leon	10	1	0
Sarah	12	0	0
Gena	8	0	0
Patrick	5	1	1

⋮



Person	Age	Male?	Height > 55”
Alice	14	0	1
Bob	10	1	1
Carol	13	0	1
Dave	8	1	0
Erin	11	0	0
Frank	9	1	1
Gena	8	0	0

y

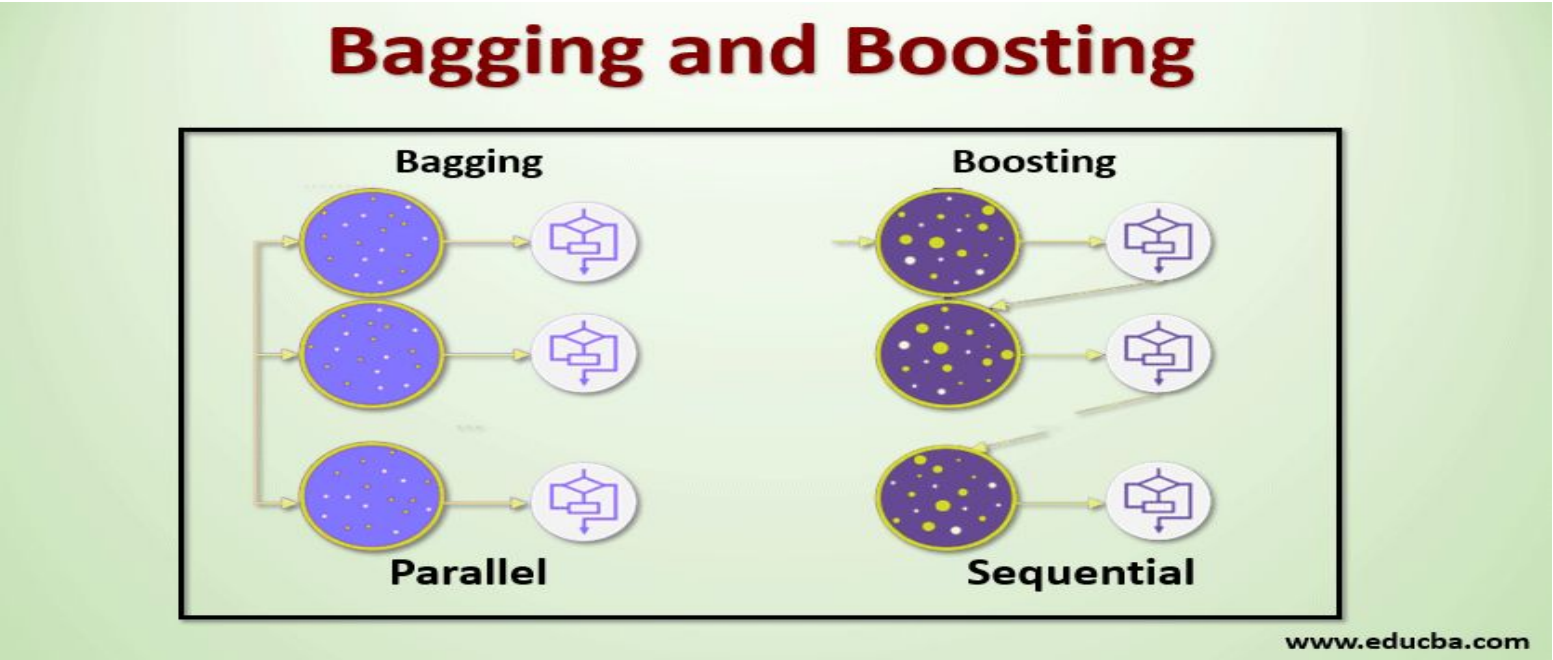
h(x)

Generalization Error:

$$L(h) = E_{(x,y)\sim P(x,y)}[f(h(x),y)]$$

Boosting

- The term ‘Boosting’ refers to a family of algorithms which converts weak learner to strong learners.
- The weak learner is the classifiers that are correct only up to a small extent with the actual classification, while the strong learners are the classifiers that are well correlated with the actual classification.
- To find weak rule, we apply base learning (ML) algorithms with a different distribution. Each time base learning algorithm is applied, it generates a new weak prediction rule. After many iterations, the boosting algorithm combines these weak rules into a single strong prediction rule.



Boosting

Choosing different distribution for each round

In boosting we take records from the dataset and pass it to base learners sequentially

- Suppose we have m number of records in the dataset. Then we pass a few records to base learner BL1 and train it and then we pass all the records from the dataset and see how the Base learner works.
- For all the records which are classified incorrectly by the base learner, we only take them and pass it to other base learner say BL2 and simultaneously we pass the incorrect records classified by BL2 to train BL3.
- This will go on unless and until we specify some specific number of base learner models we need.
- Finally, we combine the output from these base learners and create a strong learner, as a result, the prediction power of the model gets improved.

Top advantages and disadvantages

Advantages of Bagging

- Multiple weak learners can work better than a single strong learner.
- It provides stability and increases the accuracy of the ML algorithm that is used in classification and regression.
- It helps in reducing variance i.e. it avoids overfitting.

Disadvantages of Bagging

- It may result in high bias if it is not modelled properly and thus may result in underfitting.
- Since we must use multiple models, it becomes computationally expensive and may not be suitable in various use cases.

Advantages of Boosting

- It is one of the most successful techniques in solving the two-class classification problems.
- It is good at handling the missing data.

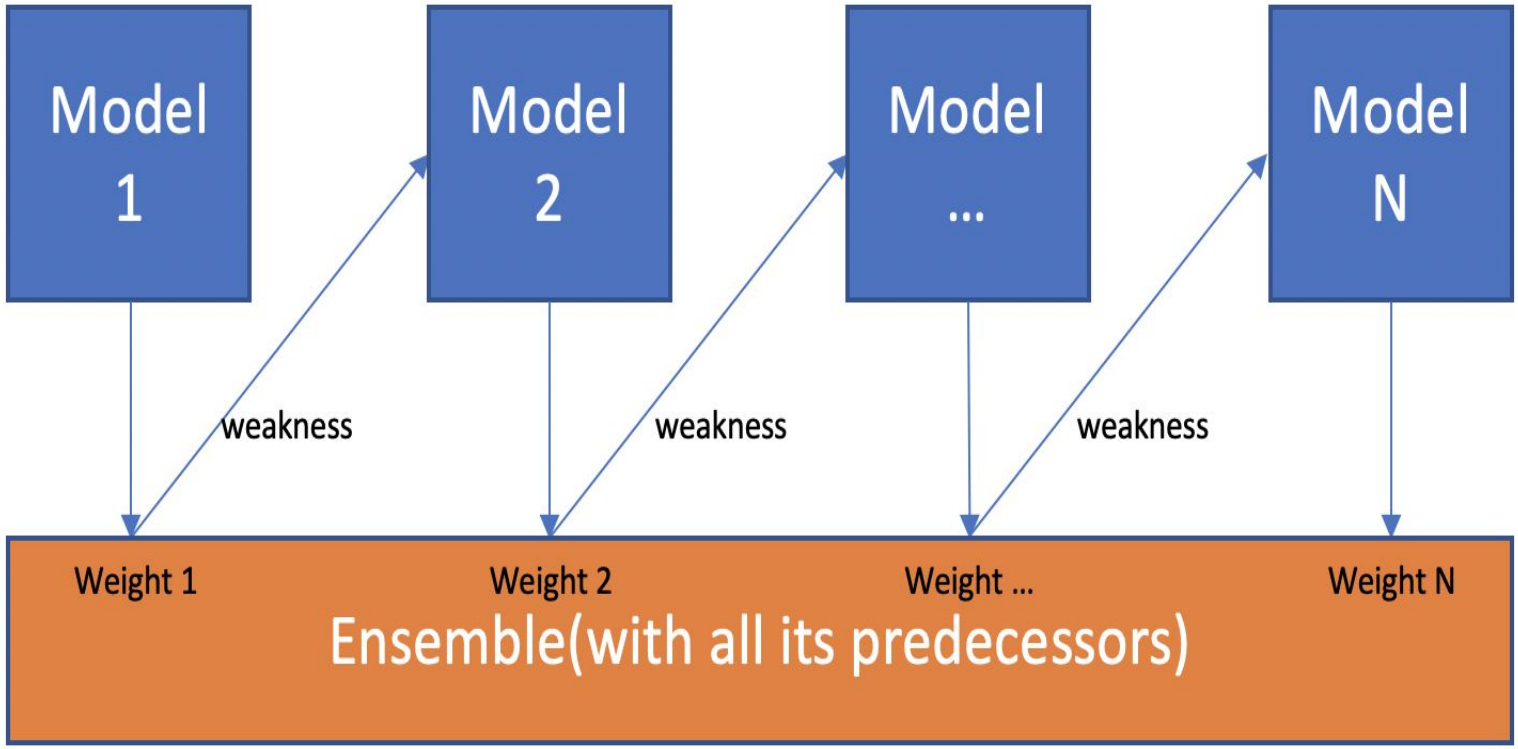
Disadvantages of Boosting

- Boosting is hard to implement in real-time due to the increased complexity of the algorithm.
- High flexibility of this techniques results in a multiple number of parameters than have a direct effect on the behaviour of the model.

Types of Boosting Algorithms

- Gradient Tree Boosting
- XGBoost

Model 1,2,..., N are individual models (e.g. decision tree)



INTERNSHIPSTUDIO

- Q.1 What do you mean by Bagging?
- Q.2 What do you mean by Boosting?
- Q.3 What is the goal of boosting?
- Q.4 What are different methods of Boosting?

Boosting Algorithm: Gradient Boosting

Gradient boosting is a technique for regression and classification problems. The prediction model produced in the form of an ensemble of weak prediction models.

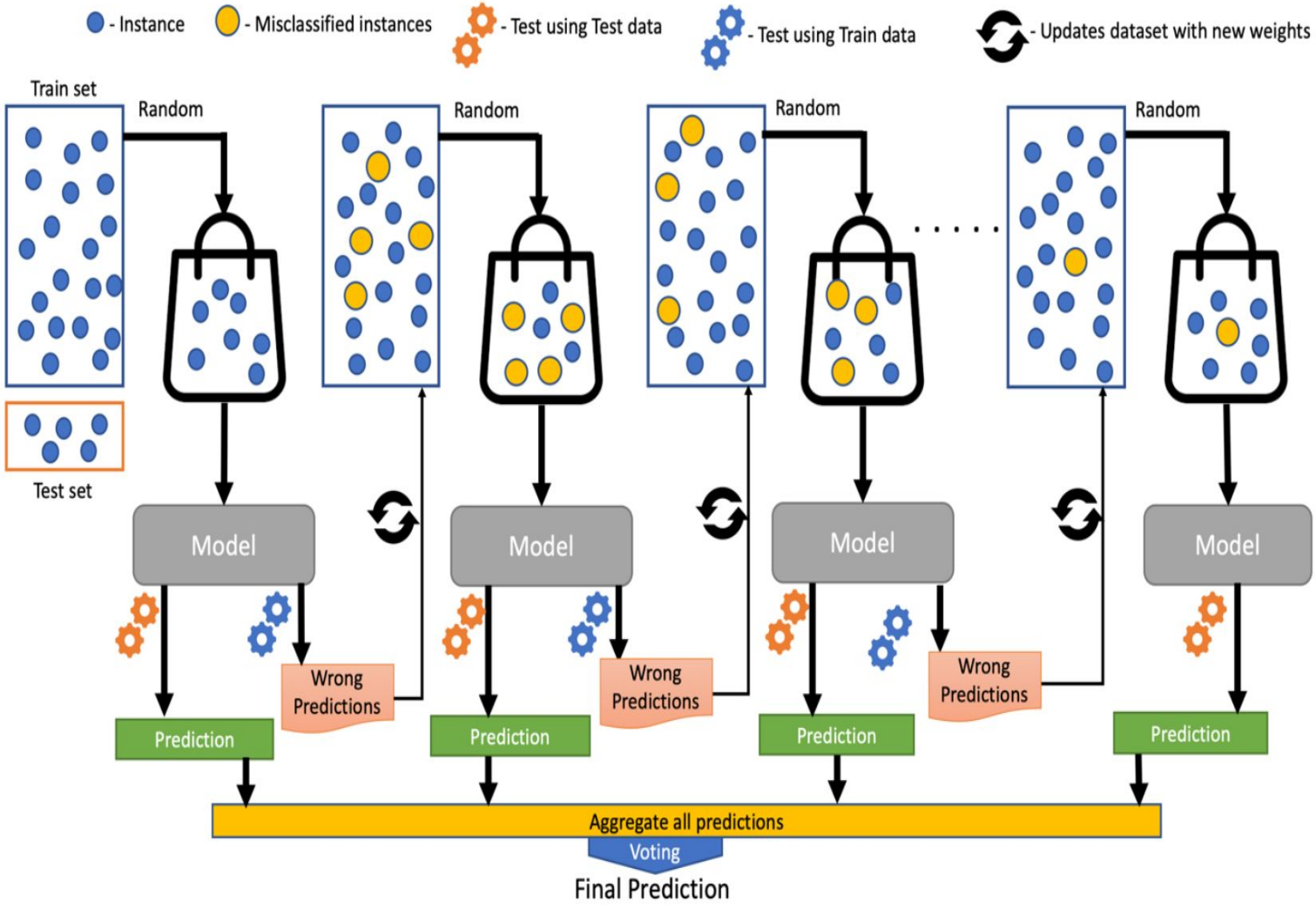
The accuracy of a predictive model can be boosted in two ways:

a. Either by using feature engineering or
b. By applying boosting algorithms.

There are many boosting algorithms like

- Gradient Boosting
- XGBoost
- AdaBoost

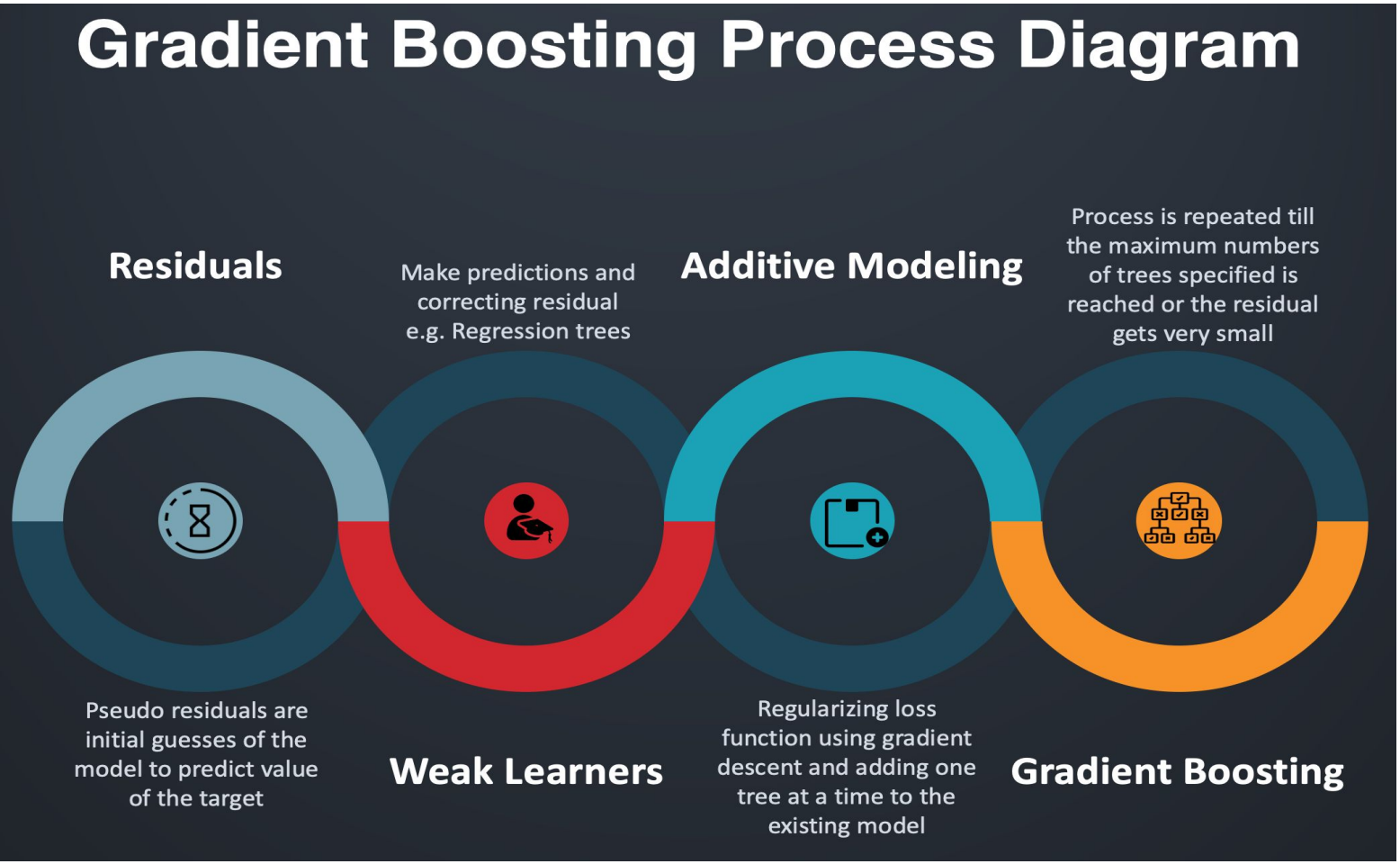
Internal working of boosting algorithm



Gradient Boosting

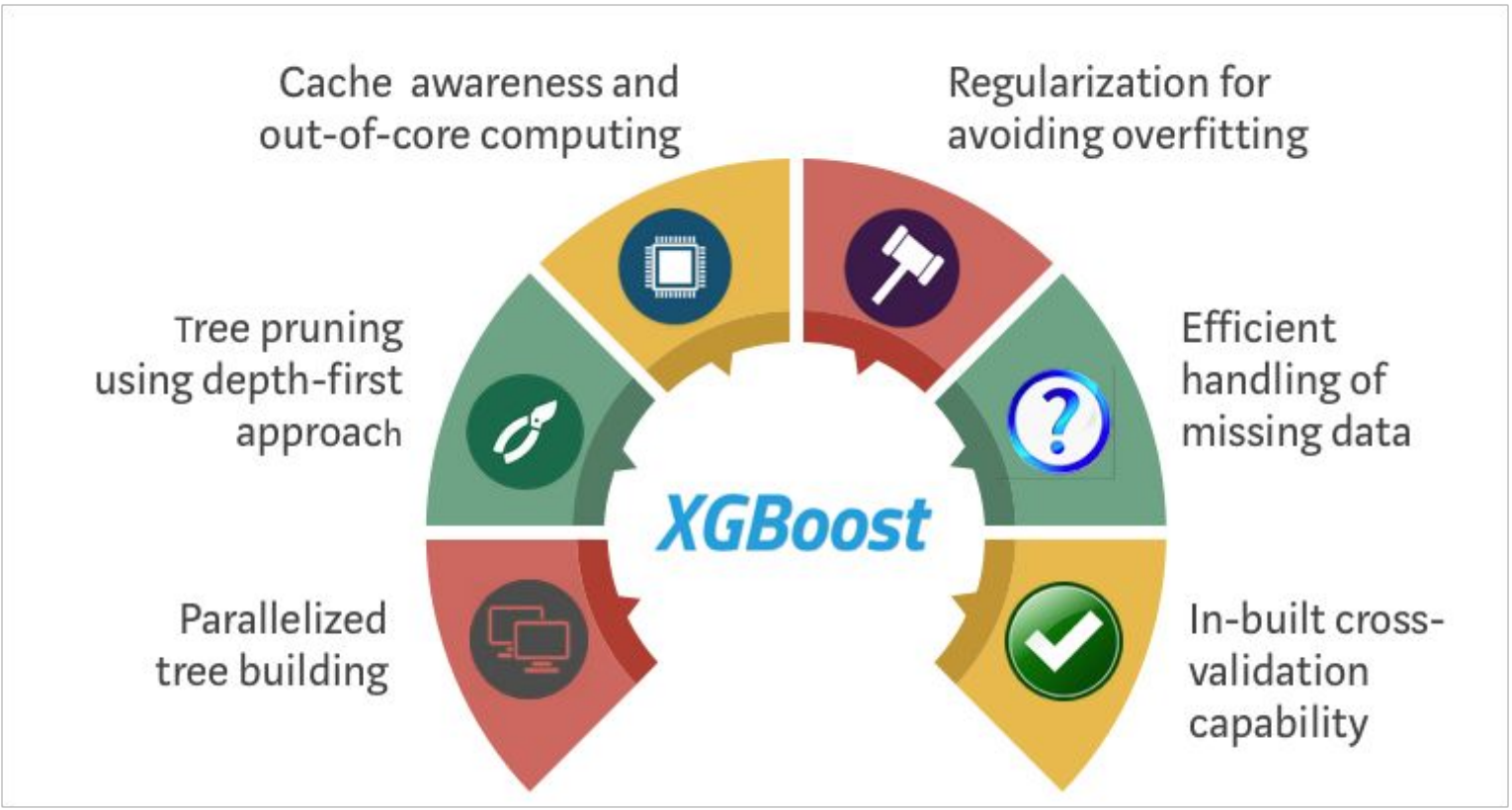
Gradient boosting Algorithm involves three elements:

- A loss function to be optimized.
- Weak learner to make predictions.
- An additive model to add weak learners to minimize the loss function.



Extreme Gradient Boosting (XGBoost)

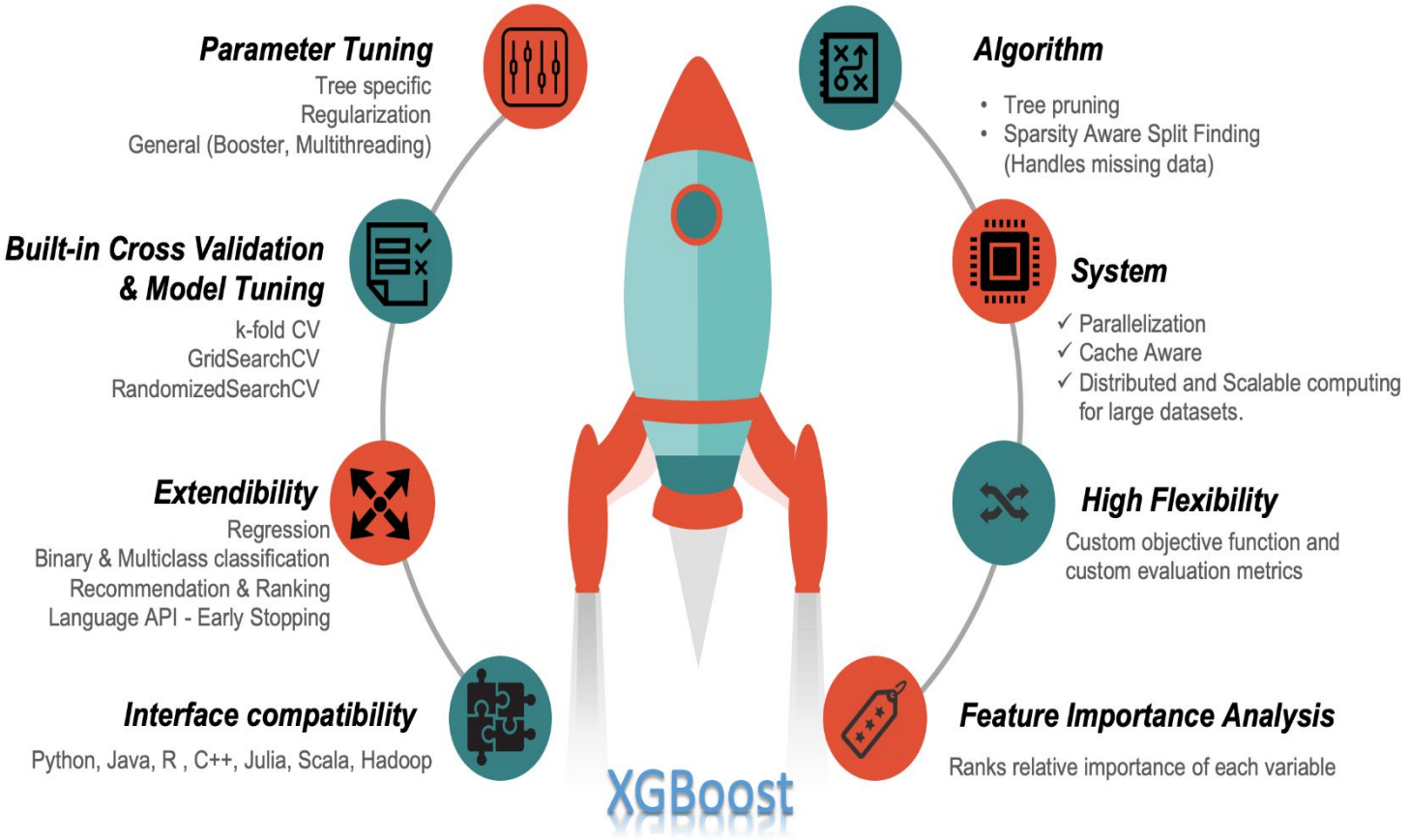
- XGBoost Algorithm is an implementation of gradient boosted decision trees, designed for speed and performance.
- Basically, it is a type of software library. It can be used for supervised learning tasks such as Regression, Classification, and Ranking.
- It is built on the principles of gradient boosting framework and designed to "push the extreme of the computation limits of machines to provide a scalable, portable and accurate library."



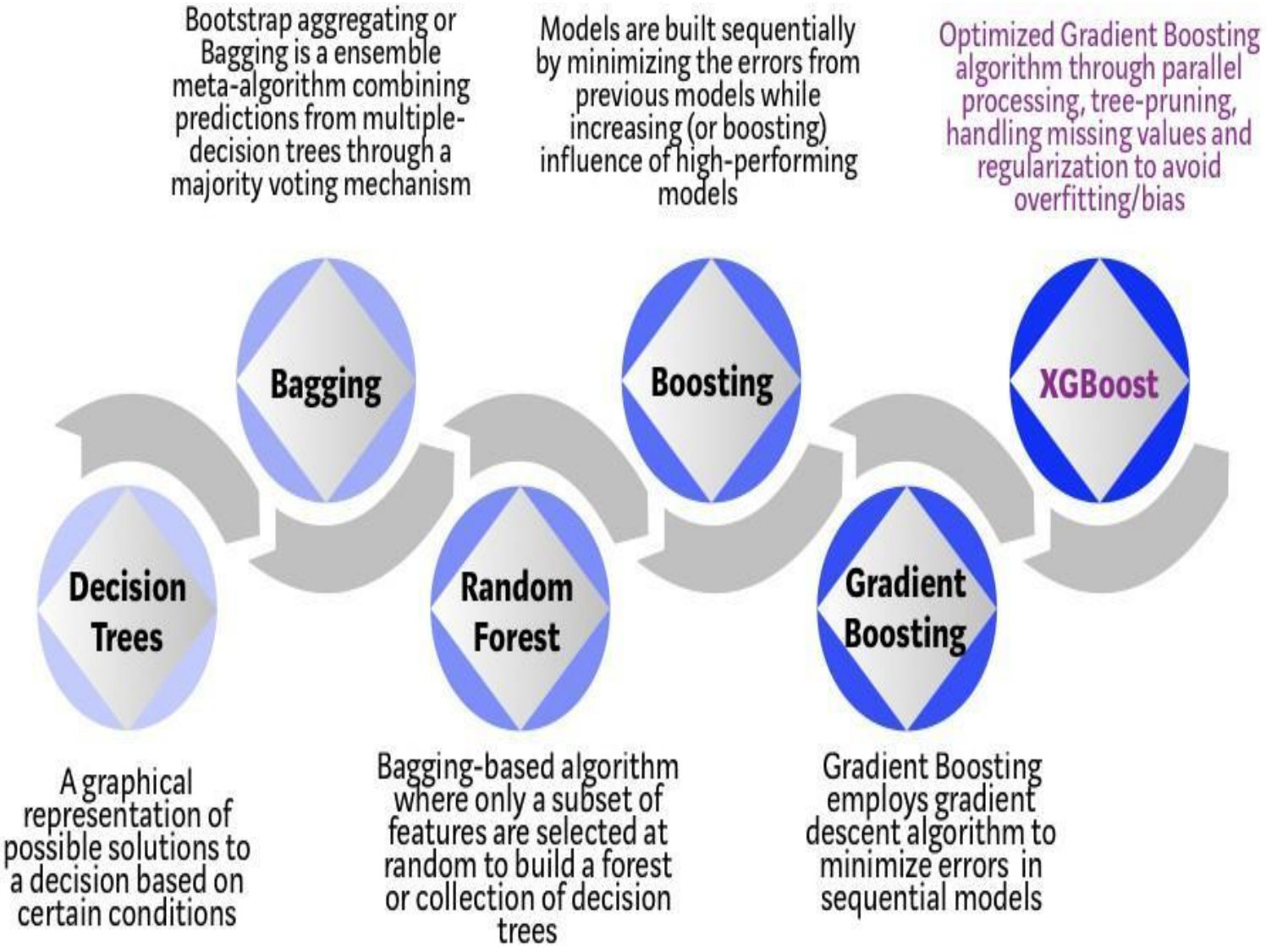
System Feature- **XGBoost**

For use of a range of computing environments this library provides:

- Parallelization of tree construction using all of your CPU cores during training.
- Distributed Computing for training very large models using a cluster of machines & Out-of-Core Computing for very large datasets that don't fit into memory.



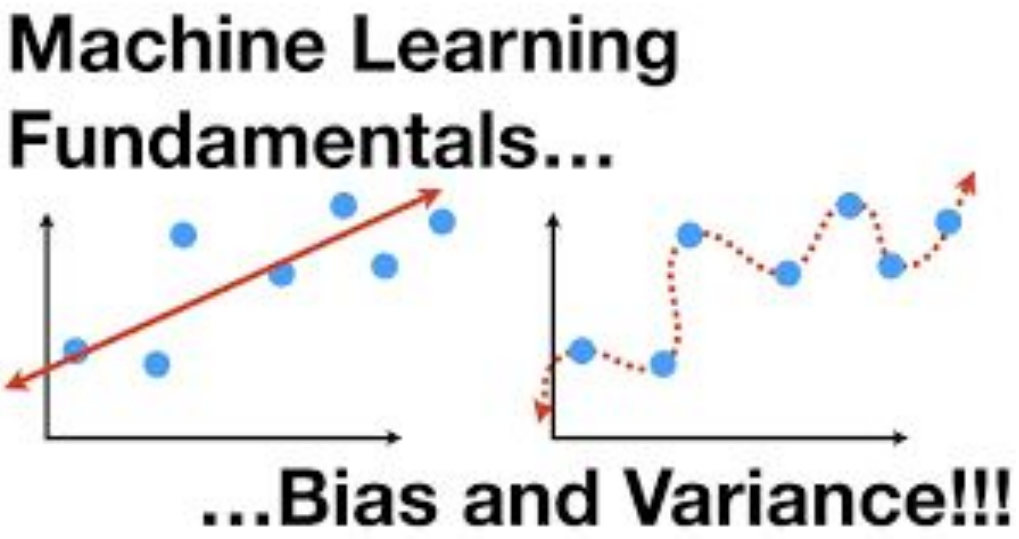
Comparison- XGBoosting



What is Bias?

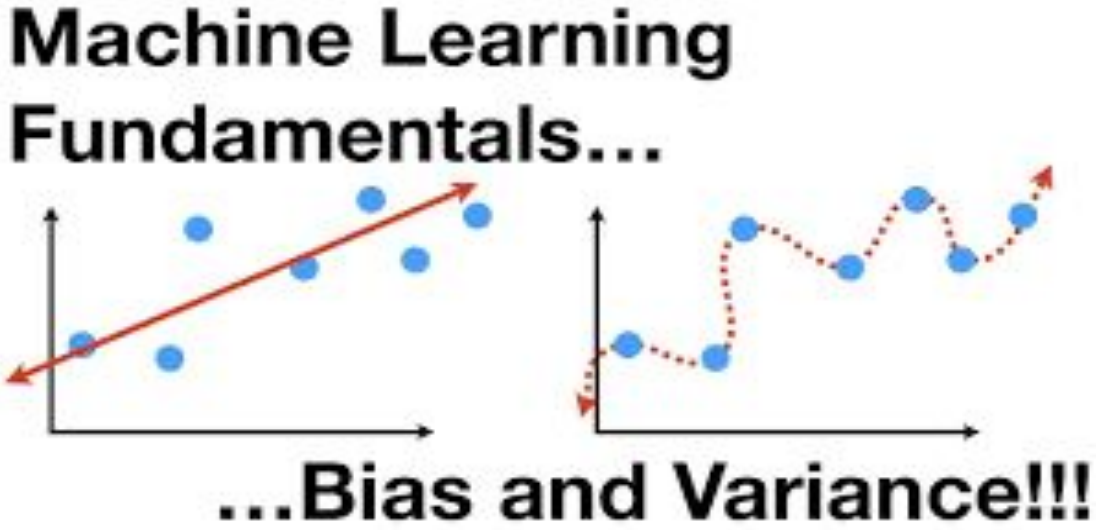
INTERNSHIPSTUDIO

- Bias is how far are the predicted values from the actual values. If the average predicted values are far off from the actual values then the bias is high.
- High bias causes algorithm to miss relevant relationship between input and output variable. When a model has a high bias then it implies that the model is too simple and does not capture the complexity of data thus underfitting the data.



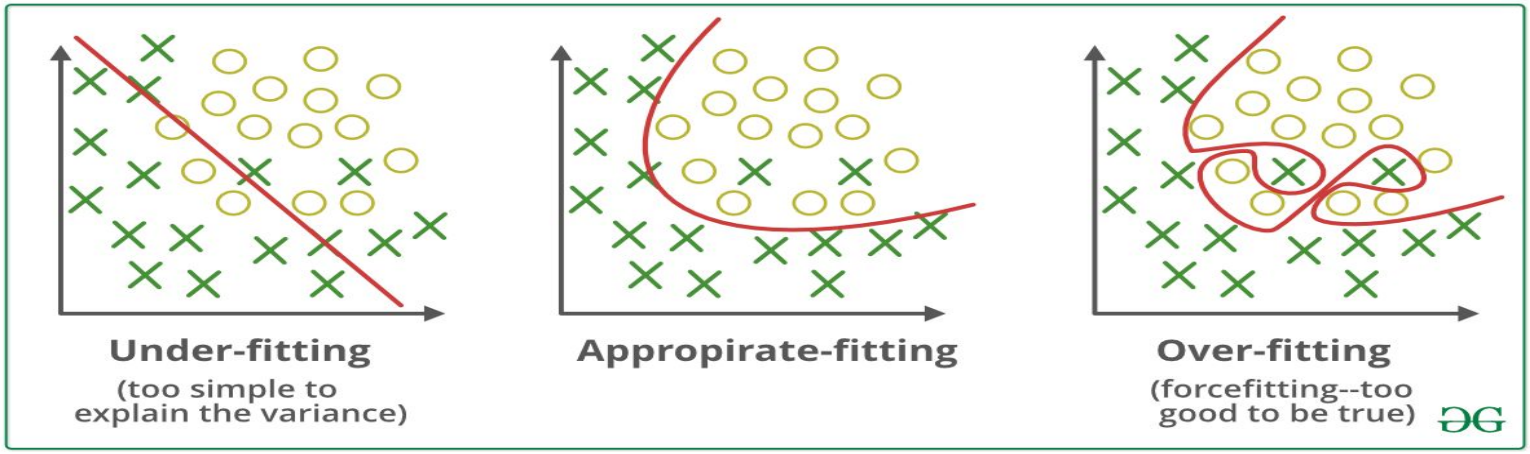
What is Variance ?

- Variance occurs when the model performs good on the trained dataset but does not do well on a dataset that it is not trained on, like a test dataset or validation dataset. Variance tells us how scattered are the predicted value from the actual value.
- High variance causes overfitting that implies that the algorithm models random noise present in the training data.



What is Underfitting?

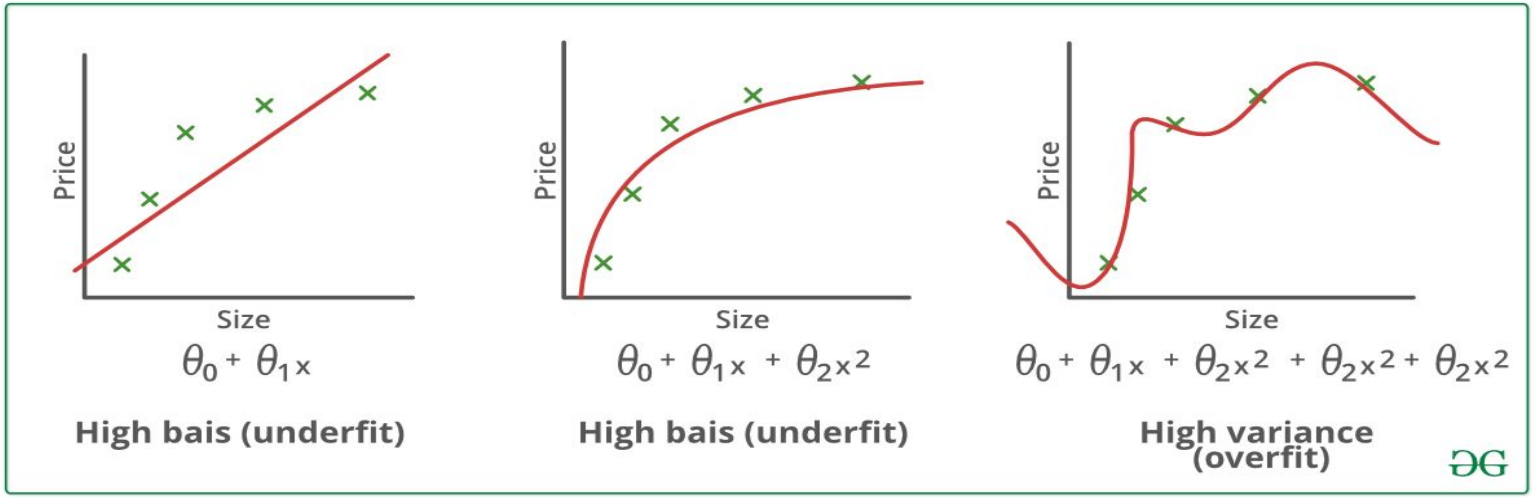
- A statistical model or a algorithm is said to have underfitting when it cannot capture the underlying trend of the data.
- Underfitting destroys the accuracy of our machine learning model.
- Its occurrence simply means that our model or the algorithm does not fit the data well enough.
- It usually happens when we have less data to build an accurate model and also when we try to build a linear model with a non-linear data.



Underfitting

- Underfitting can be avoided by using more data and also reducing the features by feature selection.

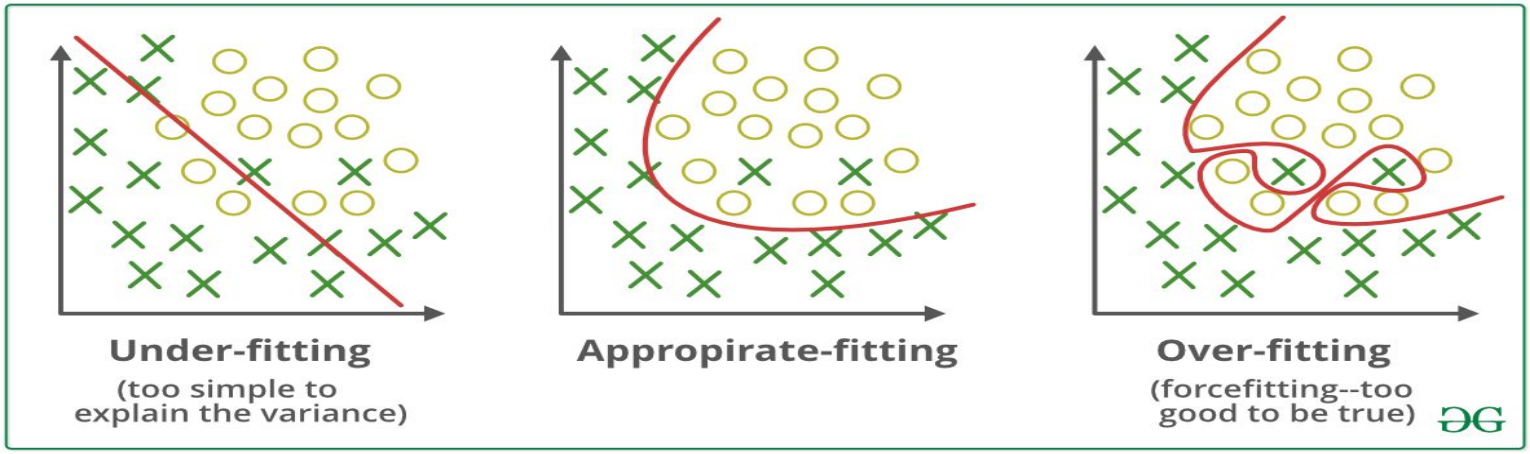
Underfitting – High bias and low variance



What is Overfitting?

INTERNSHIPSTUDIO

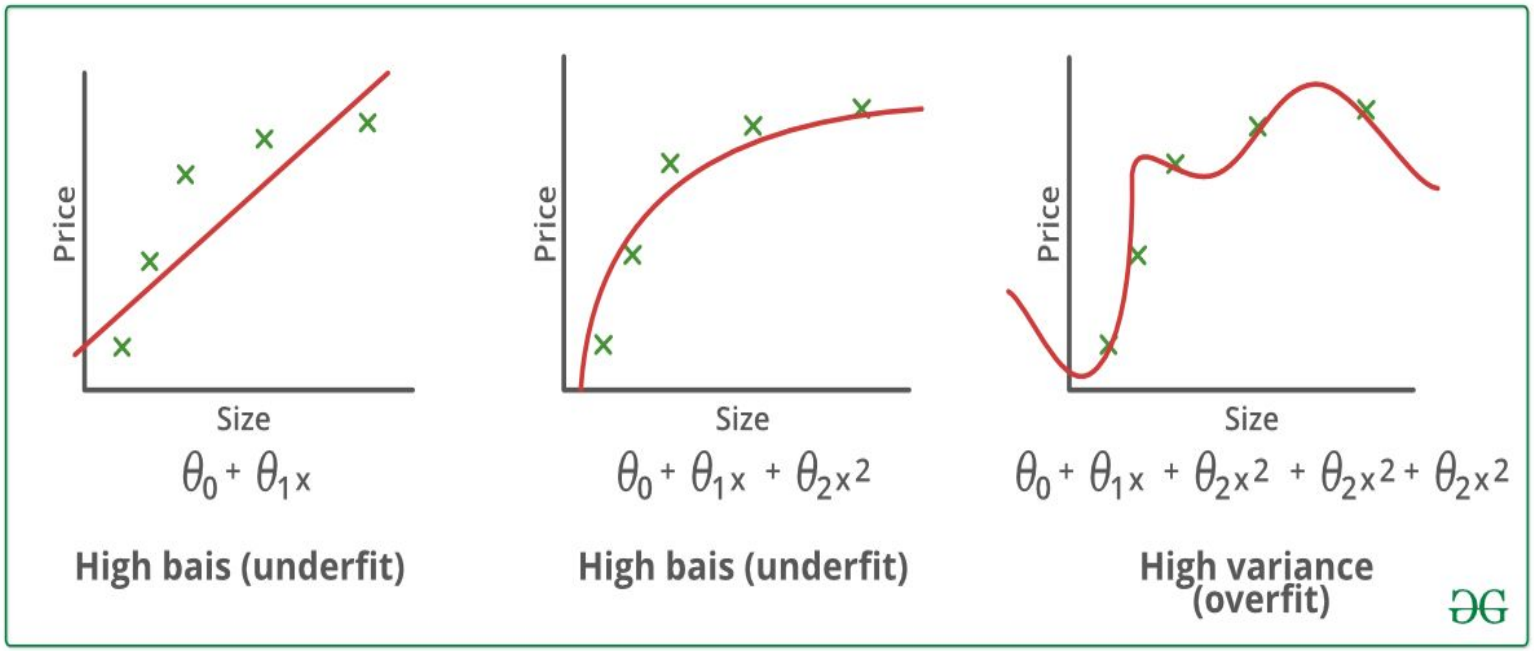
- Overfitting refers to a model that models the training data too well.
- Overfitting happens when a model 'learns' the detail and noise in the training data to the extent that it try to 'cheat' predictions on new data.
- This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model.



Overfitting

- When a model gets trained with so much of data, it starts learning from the noise and inaccurate data entries in our data set.

Overfitting – High variance and low bias



How to reduce Overfitting?

Techniques to reduce overfitting :

1. Increase training data.

2. Reduce model complexity.

3. Early stopping during the training phase (have an eye over the loss over the training period as soon as loss begins to increase stop training).

4.Use dropout for neural networks to tackle overfitting.

How to reduce Underfitting?

INTERNSHIPSTUDIO

Techniques to reduce underfitting

1. Increase model complexity

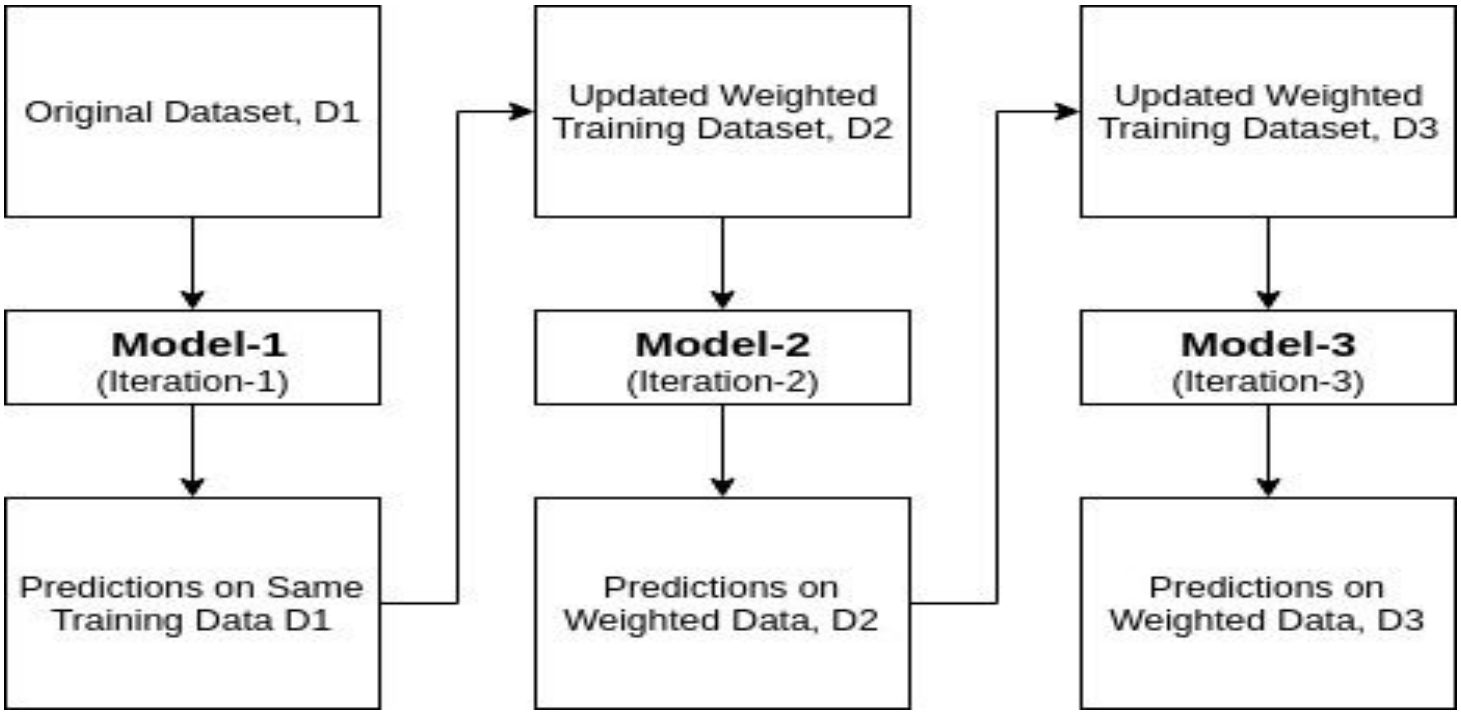
2. Increase number of features, performing feature engineering

3. Remove noise from the data.

4. Increase the number of epochs or increase the duration of training to get better results.

AdaBoost

- AdaBoost is short for Adaptive Boosting.
- It combines multiple classifiers to increase the accuracy of classifiers.
- AdaBoost is an iterative ensemble method.
- AdaBoost classifier builds a strong classifier by combining multiple poorly performing classifiers so that you will get high accuracy strong classifier.

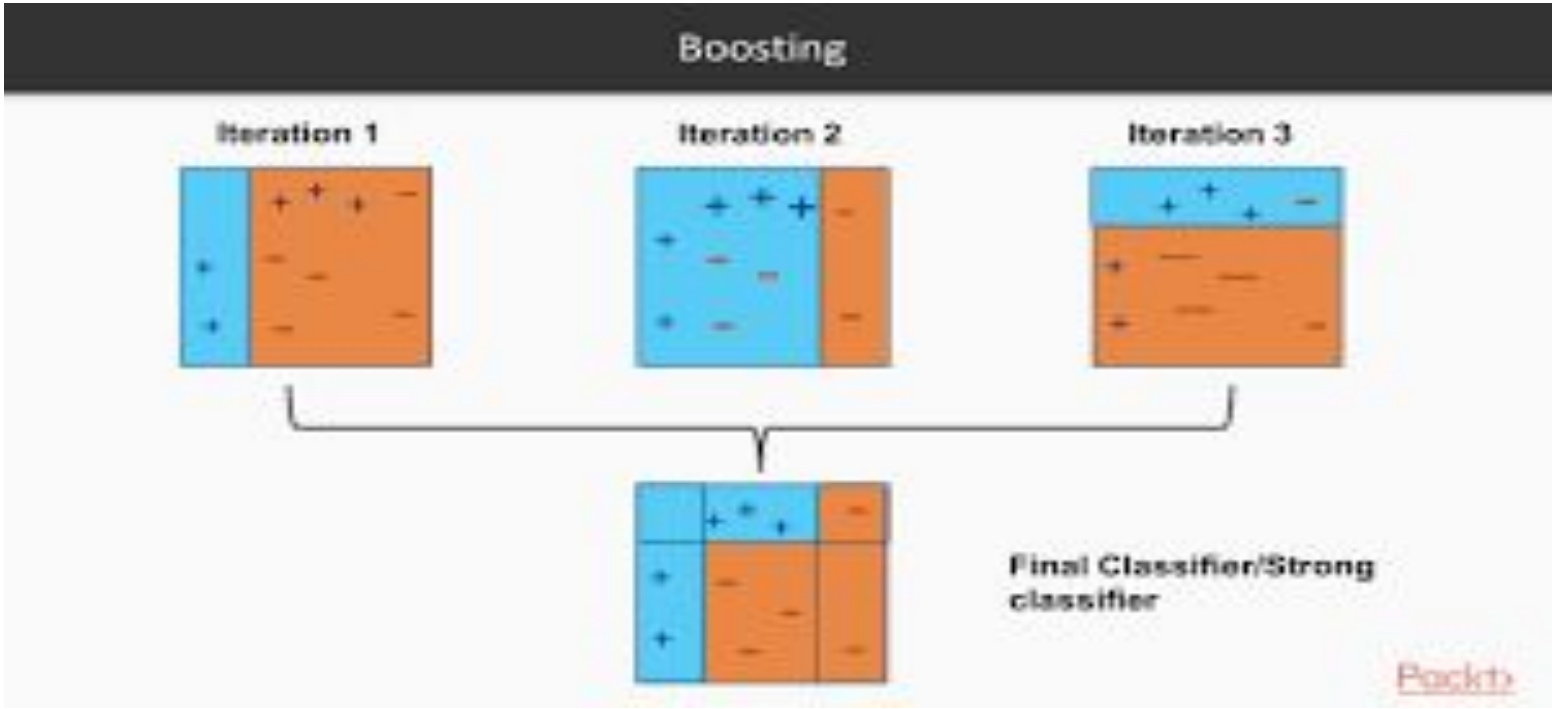


AdaBoosting

→ The weak learners in AdaBoost are decision trees with a single split, called decision stumps.

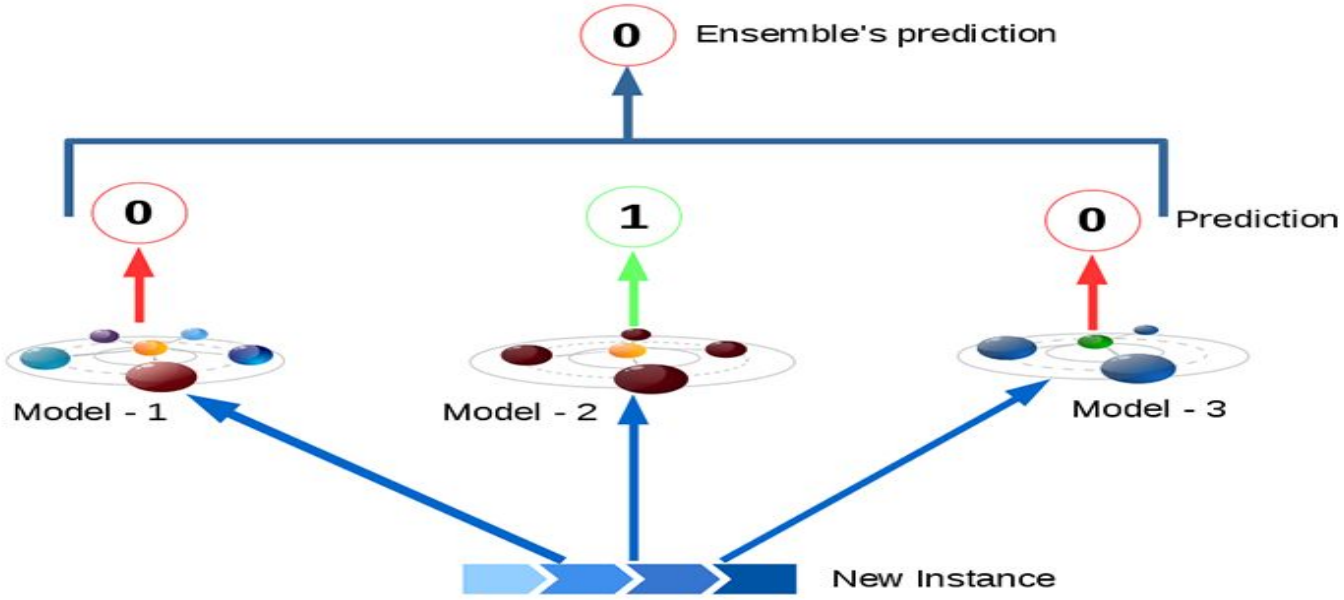
→ AdaBoost works by putting more weight on difficult to classify instances and less on those already handled well.

→ AdaBoost algorithms can be used for both classification and regression problem.



Voting

- Voting is one of the simplest ways of combining the predictions from multiple machine learning algorithms.
- It works by first creating two or more standalone models from your training dataset. A Voting Classifier can then be used to wrap your models and average the predictions of the sub-models when asked to make predictions for new data.
- You can create a voting ensemble model for classification using the VotingClassifier class.



INTERNSHIPSTUDIO

Q.1 What is Gradient Boosting?

Q.2 What is XGBoosting?

Q.3 What is Overfitting?

Q.4 What is Underfitting?

Q.5 How to reduce Overfitting?

Thank You