DECISION TREES AND ENSEMBLE LEARNING

ALGORITHM, IMPLEMENTATION, METRICS, APPLICATIONS

AGENDA

- Decision Tree algorithm
- Ensemble Learning

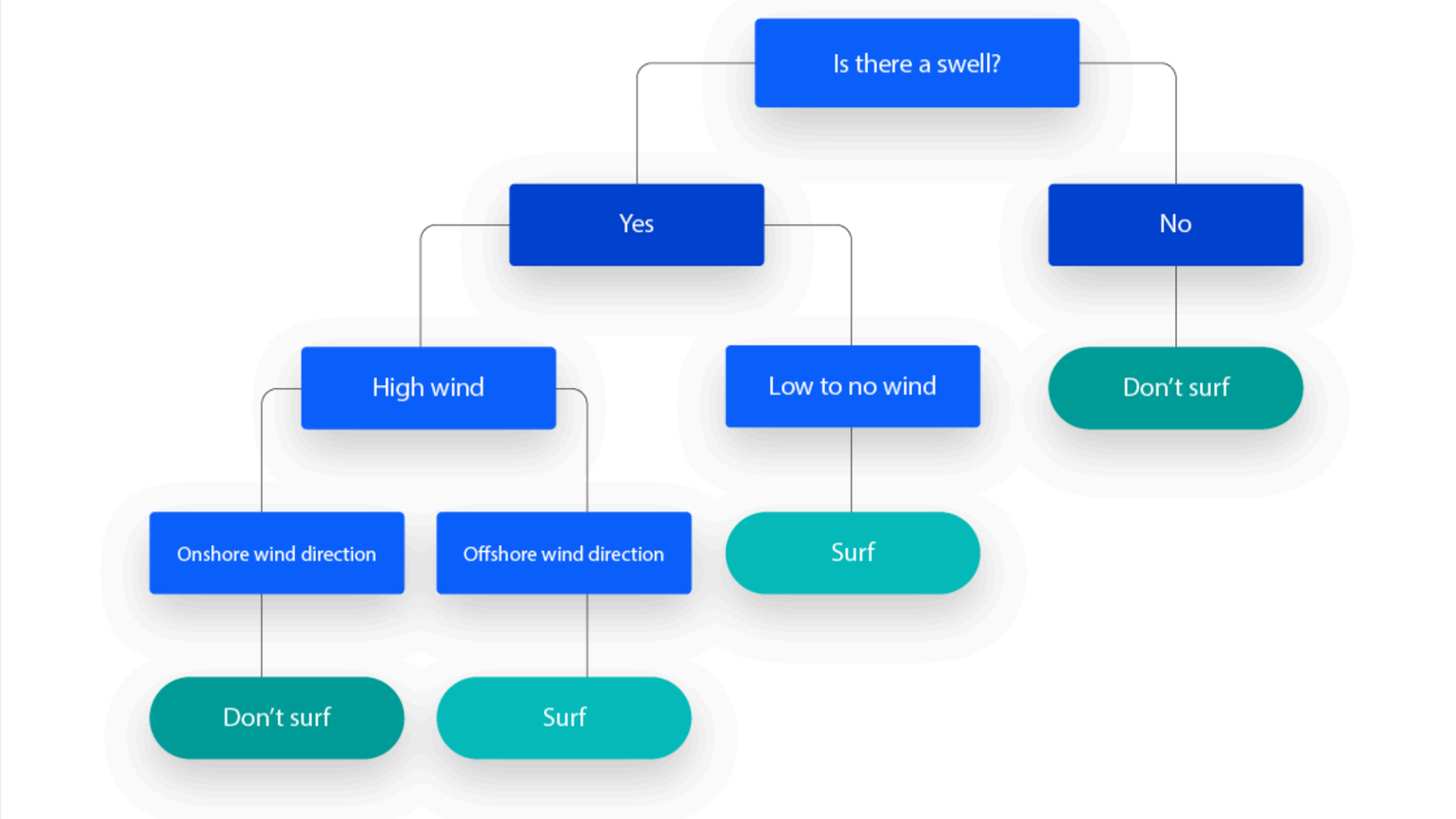
WHAT'S A DECISION TREE?

- A decision tree is a supervised learning algorithm used for classification and regression tasks. It has a tree structure consisting of a root node, branches, internal (decision) nodes, and leaf nodes.
- The root node starts the tree with no incoming branches, and its outgoing branches lead to internal nodes, which eventually connect to leaf nodes representing all possible outcomes in the dataset.

EXAMPLE

Example:

As an example, let's imagine that you were trying to assess whether or not you should go surf, you may use the following decision rules to make a choice:



DECISION TREE - ALGORITHM

- **Step 1** : Start with the entire dataset as the root.
- **Step 2** :Select the best attribute using a criterion (e.g., Gini index, Information gain) to split the data.
- **Step 3** : Split the dataset into subsets based on the chosen attribute's values.
- **Step 4** : Create a node for each subset.
- **Step 5** : Repeat the process for each node:
- **Step 6** : Terminate when all nodes are pure or a stopping criterion (e.g., maximum depth) is met.

ATTRIBUTE SELECTION MEASURES

- Selecting the best attribute for the nodes in a decision tree is crucial. This is done using the Attribute Selection Measure (ASM), which helps identify the optimal attribute for the root and sub-nodes.
- There are two popular techniques for ASM, which are:
  --> Information Gain
  --> Gini Index

INFORMATION GAIN

- Information gain is the measurement of changes in entropy after the segmentation of a dataset based on an attribute.
- It calculates how much information a feature provides us about a class.
- According to the value of information gain, we split the node and build the decision tree.
- It can be calculated using the below formula:

Information Gain= Entropy(S)- [(Weighted Avg) *Entropy(each feature)]

ENTROPY

- Entropy: Entropy is a metric to measure the impurity in a given attribute. It specifies randomness in data. Entropy can be calculated as:
- Entropy(s)= -P(yes)log2 P(yes)- P(no) log2 P(no)
- Where,

  S= Total number of samples

  P(yes)= probability of yes

  P(no)= probability of no

SAMPLE DATASET

Day	Weather	Temperatu	Humidity	Wind	Play?
Day 1	Sunny	Hot	High	Weak	No
Day 2	Sunny	Hot	High	Strong	No
Day 3	Cloudy	Hot	High	Weak	Yes
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	Cool	Normal	Weak	Yes
Day 6	Rain	Cool	Normal	Strong	No
Day 7	Cloudy	Cool	Normal	Strong	Yes
Day 8	Sunny	Mild	High	Weak	No
Day 9	Sunny	Cool	Normal	Weak	Yes
Day 10	Rain	Mild	Normal	Weak	Yes
Day 11	Sunny	Mild	Normal	Strong	Yes
Day 12	Cloudy	Mild	High	Strong	Yes
Day 13	Cloudy	Hot	Normal	Weak	Yes
Day 14	Rain	Mild	High	Strong	No

WORKING

Calculate IG of the weather

Step 1 : Entropy of entire dataset

$$S\{+9,-5\} = -\left(\frac{9}{14}\right)\log 2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log 2\left(\frac{5}{14}\right) = 0.94$$

Step 2 : Entropy of all the attributes:

$$\text{Entropy of Sunny } \{+2,-3\} = -\left(\frac{2}{5}\right)\log 2\left(\frac{2}{5}\right) - \left(\frac{3}{5}\right)\log 2\left(\frac{3}{5}\right) = 0.971$$

$$\text{Entropy of Cloudy } \{+4,0\} = -\left(\frac{4}{4}\right)\log 2\left(\frac{4}{4}\right) - \left(\frac{0}{4}\right)\log 2\left(\frac{0}{4}\right) = 0$$

$$\text{Entropy of Rain } \{+3,-2\} = -\left(\frac{3}{5}\right)\log 2\left(\frac{3}{5}\right) - \left(\frac{2}{5}\right)\log 2\left(\frac{2}{5}\right) = 0.97$$

$$\text{Information Gain} = \text{Entropy(whole data)} - \frac{5}{14}\text{Entropy(S)} - \frac{4}{14}\text{Entropy(C)} - \frac{5}{14}\text{Entropy(R)}$$

$$= 0.246$$

WORKING

Calculate IG of the temperature

Step 1 : Entropy of entire dataset

$$S\{+9,-5\} = -\left(\frac{9}{14}\right)\log 2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log 2\left(\frac{5}{14}\right) = 0.94$$

Step 2 : Entropy of all the attributes:

$$\text{Entropy of Hot } \{+2,-2\} = -\left(\frac{2}{4}\right)\log 2\left(\frac{2}{4}\right) - \left(\frac{2}{4}\right)\log 2\left(\frac{2}{4}\right) = 1$$

$$\text{Entropy of Mild } \{+4,-2\} = -\left(\frac{4}{6}\right)\log 2\left(\frac{4}{6}\right) - \left(\frac{2}{6}\right)\log 2\left(\frac{2}{6}\right) = 0.91$$

$$\text{Entropy of Cold } \{+3,-1\} = -\left(\frac{3}{4}\right)\log 2\left(\frac{3}{4}\right) - \left(\frac{1}{4}\right)\log 2\left(\frac{1}{4}\right) = 0.81$$

$$\text{Information Gain} = \text{Entropy(whole data)} - \frac{4}{14}\text{Entropy(H)} - \frac{6}{14}\text{Entropy(M)} - \frac{4}{14}\text{Entropy(C)}$$

$$= 0.029$$

WORKING

Calculate IG of the humidity

Step 1 : Entropy of entire dataset

$$S\{+9,-5\} = -\left(\frac{9}{14}\right)\log 2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log 2\left(\frac{5}{14}\right) = 0.94$$

Step 2 : Entropy of all the attributes:

$$\text{Entropy of Hot } \{+3,-4\} = -\left(\frac{3}{7}\right)\log 2\left(\frac{3}{7}\right) - \left(\frac{4}{7}\right)\log 2\left(\frac{4}{7}\right) = 0.98$$

$$\text{Entropy of Normal } \{+6,-1\} = -\left(\frac{6}{7}\right)\log 2\left(\frac{6}{7}\right) - \left(\frac{1}{7}\right)\log 2\left(\frac{1}{7}\right) = 0.59$$

$$\text{Information Gain} = \text{Entropy(whole data)} - \frac{7}{14}\text{Entropy(H)} - \frac{7}{14}\text{Entropy(N)}$$

$$= 0.15$$

WORKING

Calculate IG of the wind

Step 1 : Entropy of entire dataset

$$S\{+9,-5\} = -\left(\frac{9}{14}\right)\log 2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log 2\left(\frac{5}{14}\right) = 0.94$$

Step 2 : Entropy of all the attributes:

$$\text{Entropy of Strong } \{+3,-3\} = -\left(\frac{3}{6}\right)\log 2\left(\frac{3}{6}\right) - \left(\frac{3}{6}\right)\log 2\left(\frac{3}{6}\right) = 1$$

$$\text{Entropy of Weak } \{+6,-2\} = -\left(\frac{6}{8}\right)\log 2\left(\frac{6}{8}\right) - \left(\frac{2}{8}\right)\log 2\left(\frac{2}{8}\right) = 0.81$$

$$\text{Information Gain} = \text{Entropy(whole data)} - \frac{6}{14}\text{Entropy(S)} - \frac{8}{14}\text{Entropy(W)}$$

$$= 0.0478$$

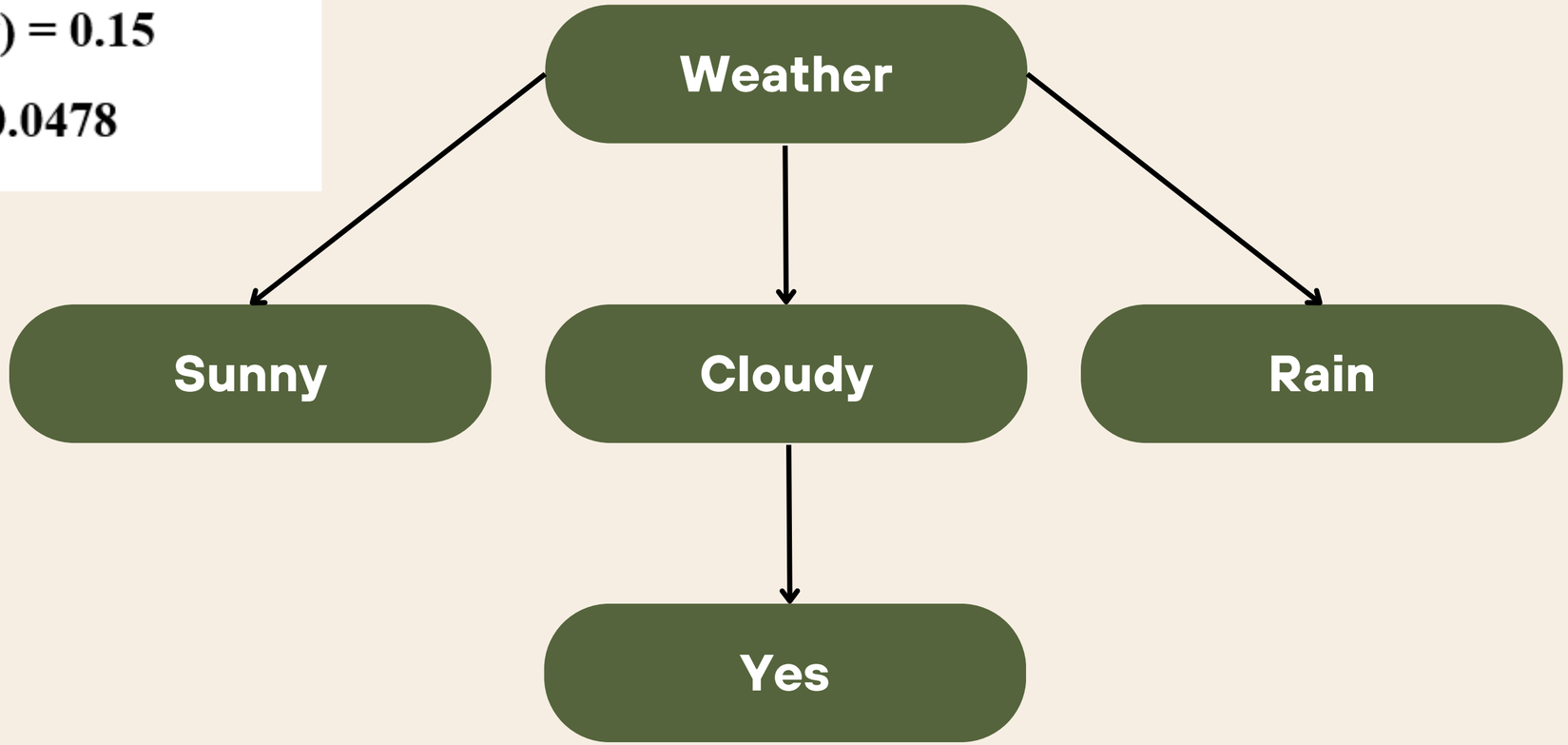
WORKING

Information Gain (S, Weather) = 0.246

Information Gain (S, Temperature) = 0.029

Information Gain (S, Humidity) = 0.15

Information Gain (S, Wind) = 0.0478

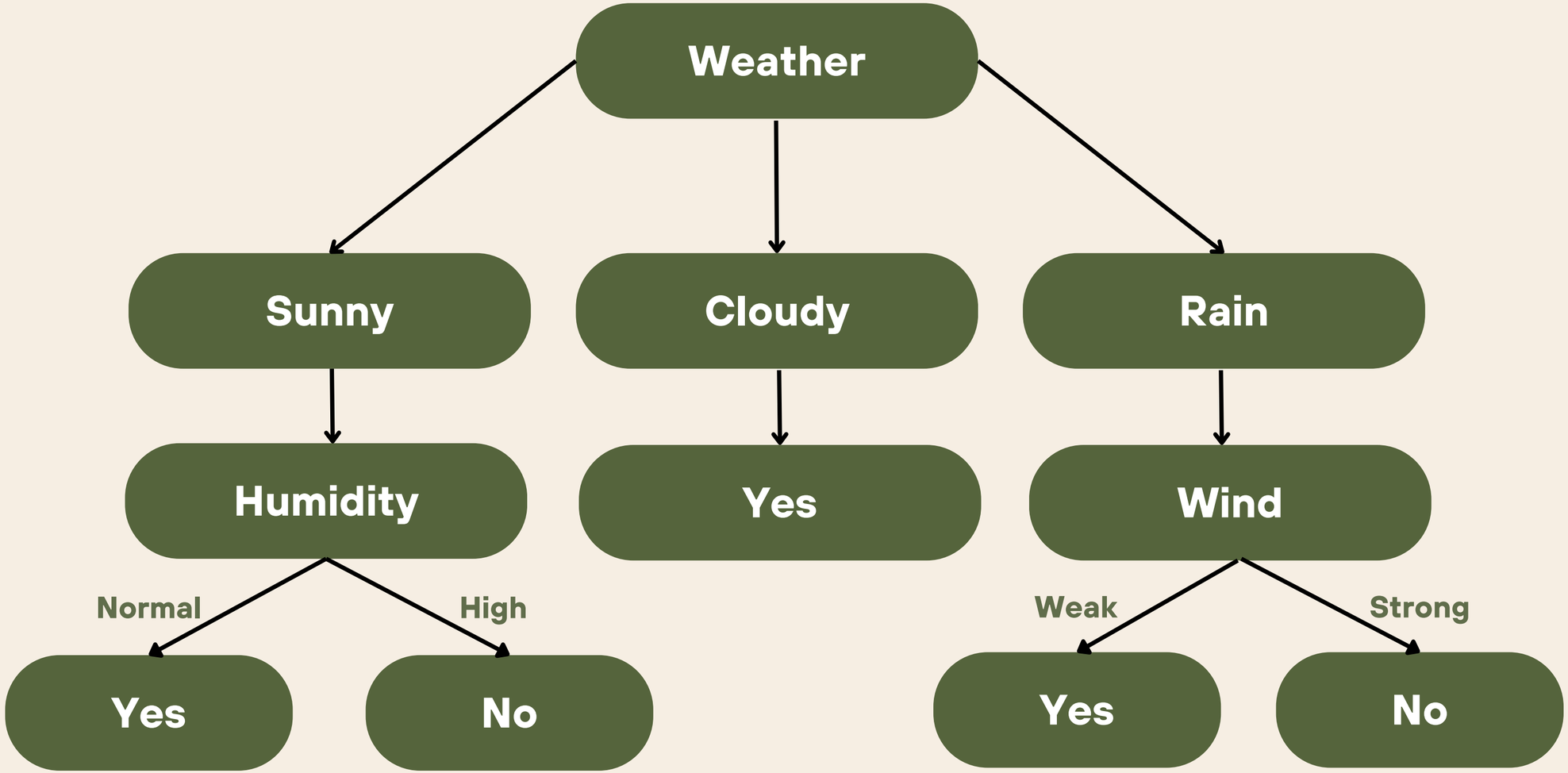


WORKING

Day	Weather	Temperatu	Humidity	Wind	Play?
Day 1	Sunny	Hot	High	Weak	No
Day 2	Sunny	Hot	High	Strong	No
Day 8	Sunny	Mild	High	Weak	No
Day 9	Sunny	Cool	Normal	Weak	Yes
Day 11	Sunny	Mild	Normal	Strong	Yes

Day	Weather	Temperatu	Humidity	Wind	Play?
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	Cool	Normal	Weak	Yes
Day 6	Rain	Cool	Normal	Strong	No
Day 10	Rain	Mild	Normal	Weak	Yes
Day 14	Rain	Mild	High	Strong	No

FINAL DECISION TREE



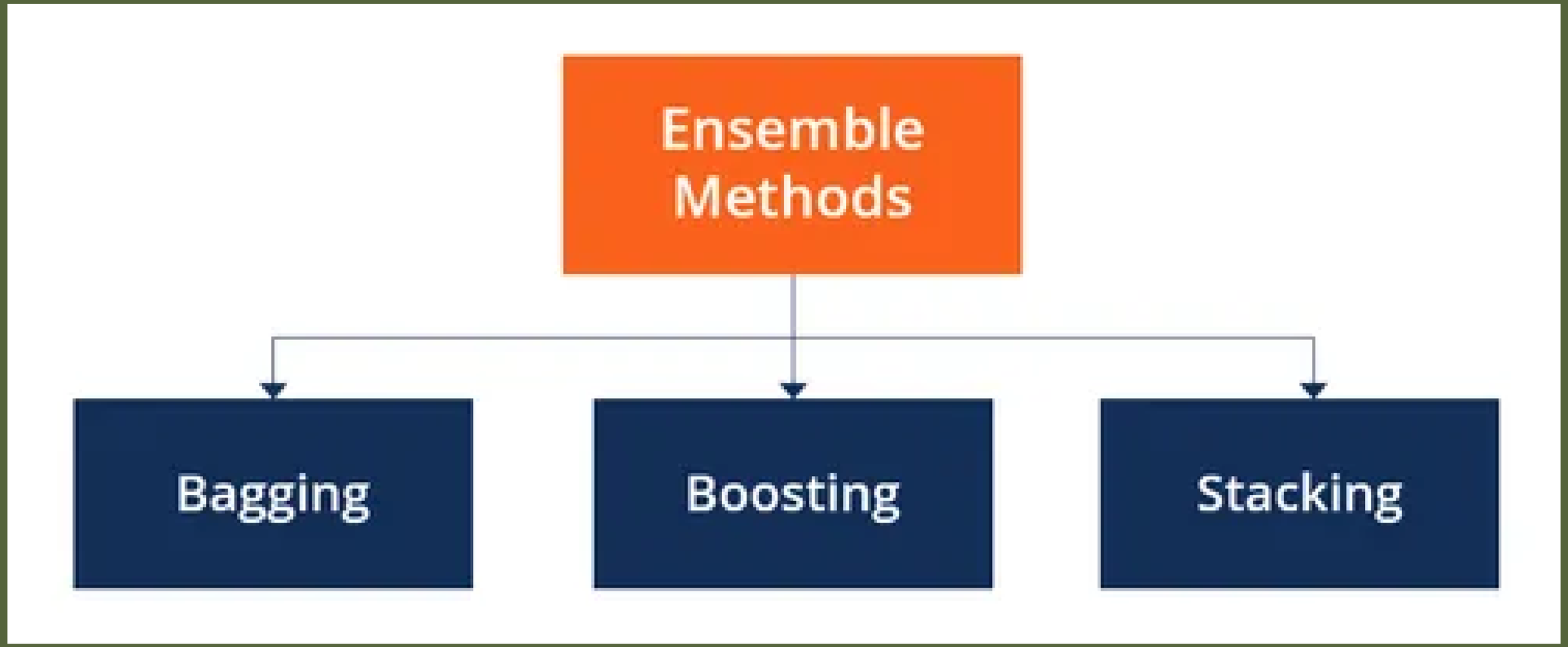
DEMO

ENSEMBLE LEARNING

ALGORITHM, IMPLEMENTATION,
METRICS, APPLICATIONS

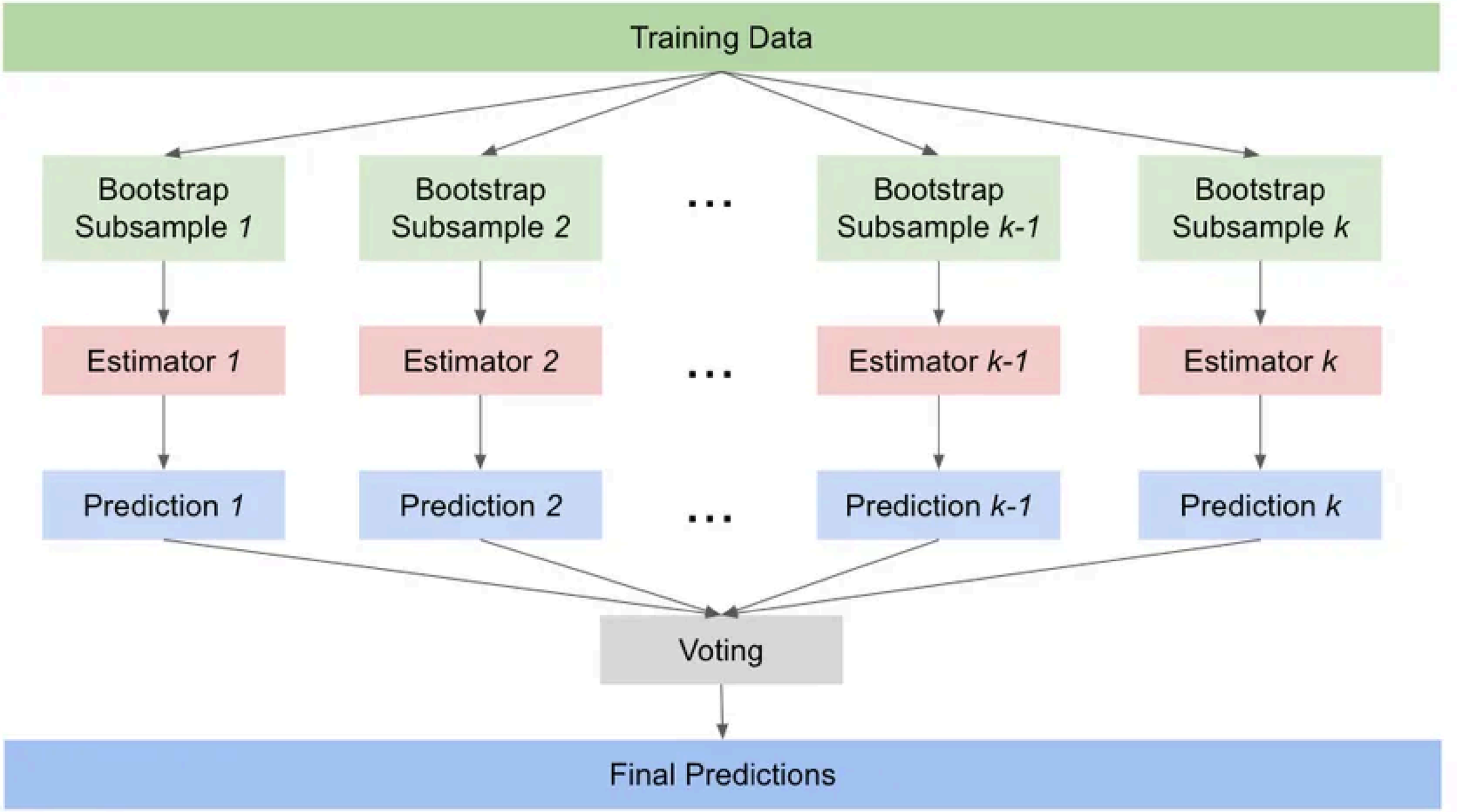
WHAT'S ENSEMBLE LEARNING?

- Ensemble learning is a machine learning technique that combines the predictions from multiple individual models to obtain a better predictive performance than any single model.
- Several individual base models (experts) are fitted to learn from the same data and produce an aggregation of output based on which a final decision is taken.


Ensemble Methods
Bagging
Boosting
Stacking

BAGGING

- Bootstrap Sampling: Divides the original training data into ‘N’ subsets and randomly selects a subset with replacement in some rows from other subsets. This step ensures that the base models are trained on diverse subsets of the data and there is no class imbalance.
- Base Model Training: For each bootstrapped sample, train a base model independently on that subset of data. These weak models are trained in parallel to increase computational efficiency and reduce time consumption.
- Prediction Aggregation: To make a prediction on testing data combine the predictions of all base models. For classification tasks, it can include majority voting or weighted majority while for regression, it involves averaging the predictions.
- Out-of-Bag (OOB) Evaluation: Some samples are excluded from the training subset of particular base models during the bootstrapping method. These “out-of-bag” samples can be used to estimate the model’s performance without the need for cross-validation.
- Final Prediction: After aggregating the predictions from all the base models, Bagging produces a final prediction for each instance.

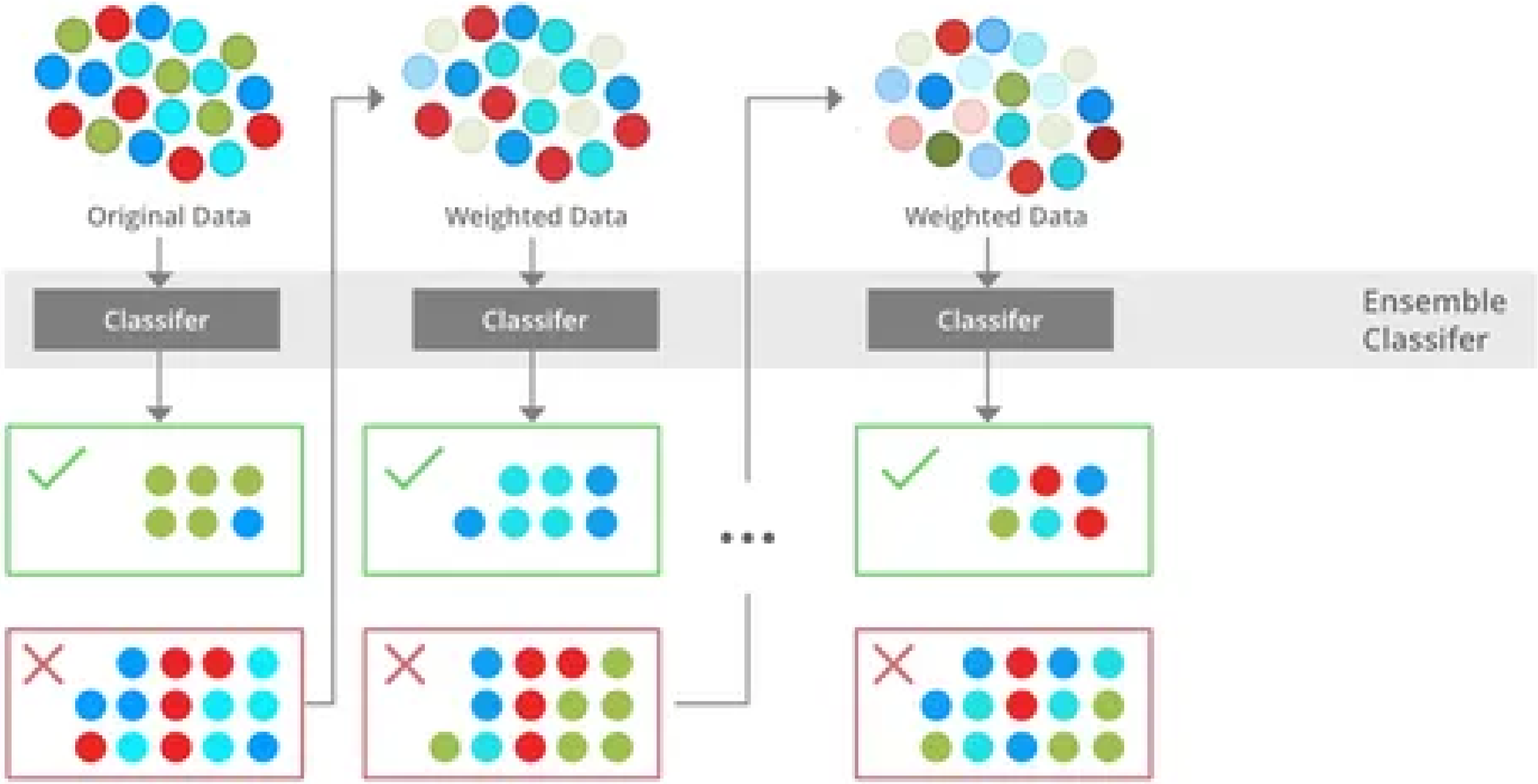

Training Data
Bootstrap Subsample 1
Bootstrap Subsample 2
...
Bootstrap Subsample k-1
Bootstrap Subsample k
Estimator 1
Estimator 2
...
Estimator k-1
Estimator k
Prediction 1
Prediction 2
...
Prediction k-1
Prediction k
Voting
Final Predictions

BOOSTING

- Boosting is an ensemble technique that combines multiple weak learners to create a strong learner.
- AdaBoost (Adaptive Boosting): AdaBoost assigns different weights to data points, focusing on challenging examples in each iteration. It combines weighted weak classifiers to make predictions.
- Gradient Boosting: Gradient Boosting, including algorithms like Gradient Boosting Machines (GBM), XGBoost, and LightGBM, optimizes a loss function by training a sequence of weak learners to minimize the residuals between predictions and actual values, producing strong predictive models.

BOOSTING

- *Initialise the dataset and assign equal weight to each of the data point.*
- *Provide this as input to the model and identify the wrongly classified data points.*
- *Increase the weight of the wrongly classified data points.*
- *if (got required results)*
- *Goto End*
- *else*
- *Repeat*
- *End*


Original Data
Weighted Data
Weighted Data
Classifer
Classifer
Classifer
Ensemble
Classifer

ADVANTAGES	DISADVANTAGES
Improved accuracy	Increased complexity
Robustness and stability	Computational costs
Reduction of overfitting	Maintenance
Versatility	Redundancy
Flexibility	High latency
Scalability	Sensitivity to model selection

APPLICATIONS

- *Natural language processing.*
- *Financial decision making*.
- *Image and speech recognition.*
- *Healthcare - prediction of diseases*
- *Path planning of autonomous vehicles.*
- *Sports analytics*

Boosting vs Bagging

Boosting	Bagging
In Boosting we combine predictions that belong to different types	Bagging is a method of combining the same type of prediction
The main aim of boosting is to decrease bias, not variance	The main aim of bagging is to decrease variance not bias
At every successive layer Models are weighted according to their performance.	All the models have the same weightage
New Models are influenced by the accuracy of previous Models	All the models are independent of each other

RANDOM FOREST CLASSIFIER

ALGORITHM, IMPLEMENTATION, METRICS, APPLICATIONS

WHAT'S A RANDOM FOREST?

- Random Forest algorithm is an ensemble learning algorithm, which is utilized for classification and regression tasks.
- It is formed by a large collection of decision trees.

ENSEMBLE LEARNING IN RANDOM FOREST

Vote

Vote

Vote

Vote

Vote Aggregation

Government

WORKING

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

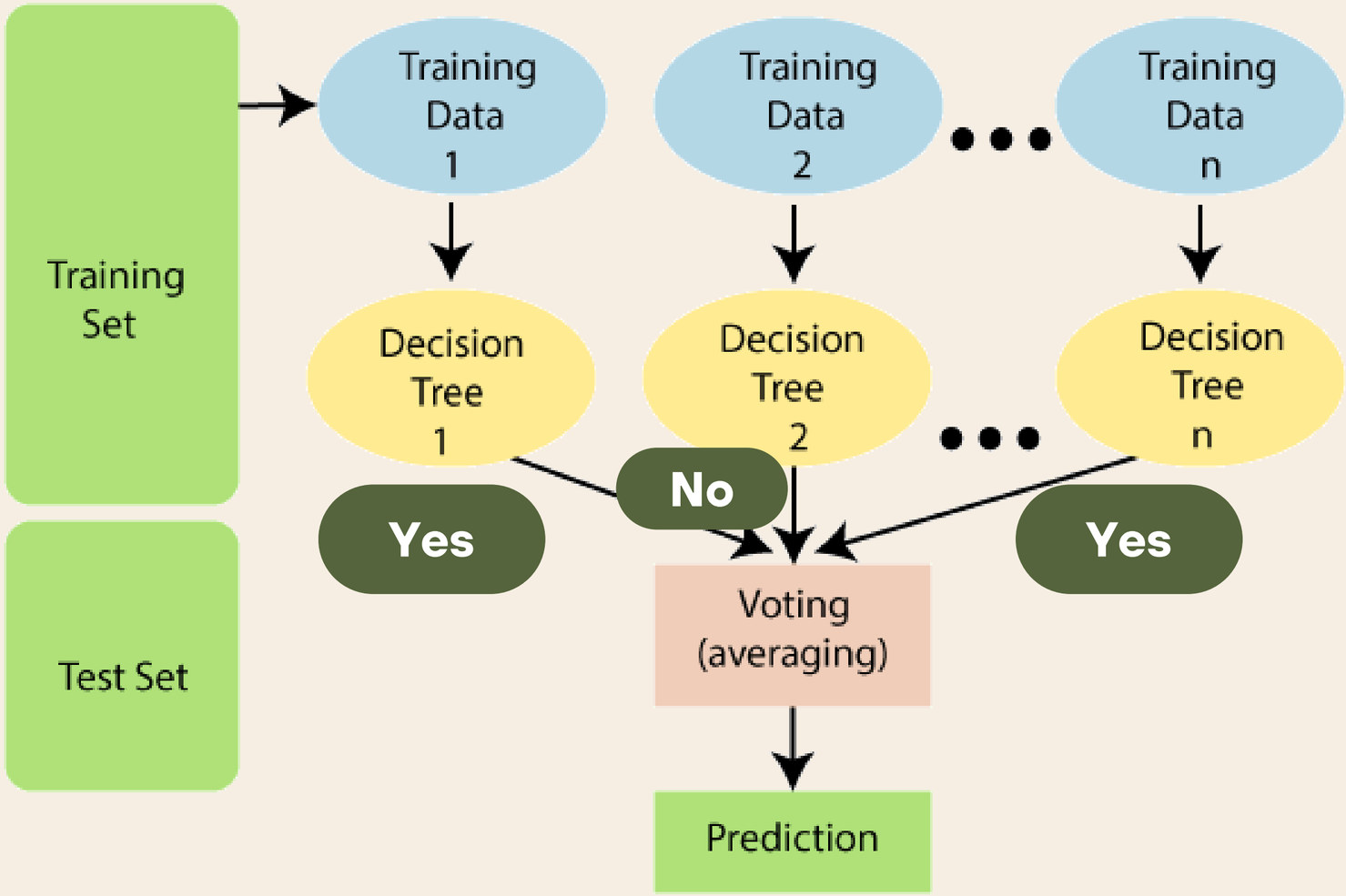
Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

WORKING

Day	Weather	Temperatu	Humidity	Wind	Play?
Day 1	Sunny	Hot	High	Weak	No
Day 2	Sunny	Hot	High	Strong	No
Day 8	Sunny	Mild	High	Weak	No
Day 9	Sunny	Cool	Normal	Weak	Yes
Day 11	Sunny	Mild	Normal	Strong	Yes

Day	Weather	Temperatu	Humidity	Wind	Play?
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	Cool	Normal	Weak	Yes
Day 6	Rain	Cool	Normal	Strong	No
Day 10	Rain	Mild	Normal	Weak	Yes
Day 14	Rain	Mild	High	Strong	No


Training Set
Test Set
Training Data 1
Training Data 2
Training Data n
Decision Tree 1
Decision Tree 2
Decision Tree n
Yes
No
Yes
Voting (averaging)
Prediction

ADVANTAGES

- It is capable of handling large datasets with high dimensionality.
- It enhances the accuracy of the model and prevents the overfitting issue.
- Normalising of data is not required as it uses a rule-based approach.

DISADVANTAGES

- It requires much computational power as well as resources as it builds numerous trees to combine their outputs.
- It also requires much time for training as it combines a lot of decision trees to determine the class.
- Due to the ensemble of decision trees, it also suffers interpretability and fails to determine the significance of each variable.

DEMO

THANK YOU

By
Mohit Sanjeev Mahajan (1BG21CS048)
Shreyas YM (1BG21CS126)