# Decision Trees and Random Forest

## Agenda

In this lesson we will cover the following concepts of classification with the help of a business use case

- Decision Tree
- Random Forest
- Bagging and Bootstrapping

## Decision Tree

Let us understand the basics of Decision Tree and its components with the help of a diagram.

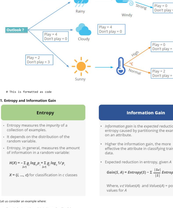
- Decision tree is a tree-like structure in which the internal node represents the test on an attribute.
- The topmost node in a decision tree is known as the root node. It learns to partition based on the attribute value.
- Each branch represents the outcome of the test and each leaf node represents the class label.
- A path from root to leaf represents the classification rules.

### Splitting

- If pure node is detected, it can be classified as leaf node
- If impure node is detected, the best attribute is selected for splitting is continued until a pure node is detected.

**Example:**

Rains a decision now to check if the match will be played or not based on climatic conditions.




This is formatted as code

### 1. Entropy and Information Gain

**Entropy**

- Entropy measures the impurity of a collection of examples.
- It depends on the distribution of the random variable.
- Entropy, in general, measures the amount of information in a random variable.

$$H(X) = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}$$

$X = \{x_1, x_2, ..., x_n\}$ is classification in n classes

**Information Gain**

- Information gain is the expected reduction in entropy caused by partitioning the examples on an attribute.
- Higher the information gain, the more effective the attribute in classifying training data.
- Expected reduction in entropy, given A

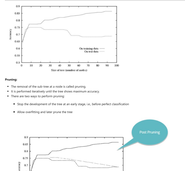
$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Where, $v \in Values(A)$ and $Values(A)$ = possible values for A

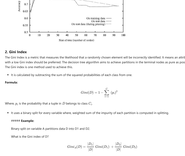
Let us consider an example where:

- Sample for humidity = {High, Normal} = {N, S, }
- Sample for wind = {Weak, Strong} = {W, S }

### 2. Calculate gain:




### 3. Select highest gain:




### 4. Inference:

- Humidity provides the best prediction for the target
- For each possible value of humidity, you can add a successor to the tree

### Overfitting and Pruning

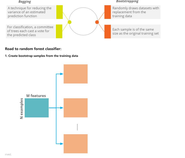
**Overfitting**

Overfitting refers to the unwanted behavior of a machine learning algorithm. It occurs when a model's performance

on the training dataset improves at the expense of poor performance on non-training data, such as a holdout test dataset or new data. In decision tree learning, overfitting can result from a decision tree that is too complex, considering all attributes and all training data. This tree will perform well on the training data but will perform poorly when applied to new data. It is because overfitting

**Pruning**

- The removal of the sub-tree at a node is called pruning.
- It is performed iteratively until the tree shows maximum accuracy.
- There are two ways to perform pruning:
  - Stop the development of the tree at an early stage, i.e. before perfect classification
  - Allow overfitting and later prune the tree




### 2. Gini Index

The Gini Index is a metric that measures the likelihood that a randomly chosen element will be incorrectly identified. It means an attribute with a lower Gini index should be preferred. The Gini index algorithm aims to achieve predictions for the terminal nodes as pure as possible. The Gini index is a cost function used to evaluate splits.

It is calculated by subtracting the sum of the squared probabilities of each class from one.

**Formula**

$$Gini(E) = 1 - \sum_{j=1}^{n} (p_j)^2$$

Where $p_j$ is the probability that a tuple in D belongs to class $C_j$

It uses a binary split for every variable where, weighted sum of the impurity of each partition is computed in splitting.

**Formula Example**

Binary split on variable A partitions data D into D1 and D2

What is the Gini index of D?

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

- In case of a discrete-valued attribute, the subset that gives the minimum gini index is selected as a splitting attribute. In the case of continuous-valued attribute, the strategy is to select each pair of adjacent values as a possible split point and point with smaller gini index chosen as the splitting point.

**Features**

- Gini index favors larger partitions.
- It works with categorical attributes.
- Gini index is perfectly unbiased when it is even.

### Random Forest

Random Forest is a supervised learning algorithm used in classification and regression problems. It builds decision trees for different samples and takes the majority for classification and the average for regression.

**Features**

- Runs efficiently on larger databases.
- Handles thousands of input variables without variable deletion.
- Provides an estimate of what variables are important in the classification.
- It gives a good estimate for the missing data and maintains accuracy when a large proportion of the data are missing.

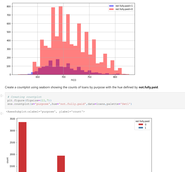
### Bagging and Bootstrapping

**Read to random forest classifier:**

**1. Create bootstrap samples from the training data**

**2. Each sample contributes to a decision tree classifier**

**3. Majority vote is taken by using the various trees generated from various bootstrap samples**


All Features
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote
Majority Vote