* Classify a pattern through a sequence of questions (20-question game); next question asked depends on the answer to the current question.
* This approach is particularly useful for non-metric data; questions can be asked in a “yes-no” or “true-false” style that do not require any notion of metric
* Sequence of questions is displayed in a directed decision tree
* Root node, links or branches, leaf or terminal nodes
* Classification of a pattern begins at the root node until we reach the leaf node; pattern is assigned the category of the leaf node
* Benefit of decision tee:
  + Interpretability: a tree can be expressed as a logical expression
  + Rapid classification: a sequence of simple queries
  + Higher accuracy & speed:

**CART algorithm** can be used for building both Classification and Regression Decision Trees. The impurity (or purity) measure used in building decision tree in CART is Gini Index. The decision tree built by CART algorithm is always a binary decision tree (each node will have only two child nodes).

**Classification Tree**: When decision or target variable is categorical, the decision tree is classification decision tree.  E.g. predicting whether customer will default or not (Binary Target variable). Or predicting food choices of the customers (nominal variable) using set of independent variable is an example of Classification Decision Tree.

**Regression Tree**: When the decision or target variable is continuous variable, the decision tree is called regression decision tree. e.g. predicting house prices using attributes of houses such as size of a house, type of house (independent, apartment etc) and others. The independent variables can continuous and categorical variables.

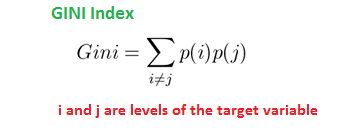
Decision trees are used for both classification and regression problems, this story we talk about classification

1. Decision tress often mimic the human level thinking so it’s so simple to understand the data and make some good interpretations.
2. Decision trees actually make you see the logic for the data to interpret (not like black box algorithms like SVM, NN, etc..)

A decision tree is a tree where each node represents a feature (attribute), each link (branch) represents a decision (rule) and each leaf represents an outcome (categorical or continues value). The whole idea is to create a tree like this for the entire data and process a single outcome at every leaf (or minimize the error in every leaf).

1. ID3 (Iterative Dichotomiser 3) → uses ***Entropy function***and [***Information gain***](https://en.wikipedia.org/wiki/Information_gain_in_decision_trees)as metrics.
2. CART (Classification and Regression Trees) → uses ***Gini Index(Classification)*** as metric.

* Many alternative measures to Information Gain. Information gain is a criterion used for split search but leads to overfitting.
* Most popular alternative: **Gini index**
  + used in e.g., in CART (Classification And Regression Trees)
  + impurity measure (instead of entropy)



**Gini Index:**

**for each branch in split:**

**Calculate percent branch represents #Used for weighting**

**for each class in branch:**

**Calculate probability of class in the given branch.**

**Square the class probability.**

**Sum the squared class probabilities.**

**Subtract the sum from 1. #This is the Ginin Index for branch**

**Weight each branch based on the baseline probability.**

**Sum the weighted gini index for each split.**

**GINI of a split**

GINI (s,t)             = GINI (t) – PL GINI (tL) – PR GINI (tR), where

s                              : split

t                              : node

GINI (t)                : Gini Index of input node t

PL                           : Proportion of observation in Left Node after split, s

GINI (tL)             : Gini of Left Node after split, s

PR                           : Proportion of observation in Right Node after split, s

GINI (tR)               : Gini of Right Node after split, s

**Maximum value of Gini Index** could be when all target values are equally distributed.

For Binary Target variable, Max Gini Index value

= 1 - (1/2)2 - (1/2)2  
= 1 - 2\*(1/2)2  
= 1- 2\*(1/4)  
= 1-0.5  
= 0.5

Similarly for Nominal variable with k level, the maximum value Gini Index is = 1–1/k

**Minimum value of Gini Index** will be 0 when all observations belong to one label.

**When To Stop Splitting?**

**If we continue to grow the tree fully until each leaf node corresponds to the lowest impurity, then the data have typically been overfit; in the limit, each leaf node has only one pattern!**

**• If splitting is stopped too early, error on training data is not sufficiently low and performance will suffer**

**• Validation and cross-validation**

* + **Continue splitting until error on validation set is minimum**
  + **Cross-validation relies on several independently chosen subsets**

**• Stop splitting when the best candidate split at a node reduces the impurity by less than the preset amount (threshold)**

**• How to set the threshold? Stop when a node has small no. of points or some fixed percentage of total training set (say 5%)**

**• Tradeoff between tree complexity or size vs. test set accuracy**

**Pruning**

Occasionally, stopping tree splitting suffers from the lack of sufficient look ahead

• A stopping condition may be met too early for overall optimal recognition accuracy

• Pruning is the inverse of splitting

• Grow the tree fully—until leaf nodes have minimum impurity.

Then all pairs of leaf nodes (with a common antecedent node) are considered for elimination.

• Any pair whose elimination yields a satisfactory (small) increase in impurity is eliminated, and the common antecedent node is declared as leaf node