# Decision Trees

Decision trees are a popular and intuitive machine learning algorithm used for both classification and regression tasks.

## Basic Concept

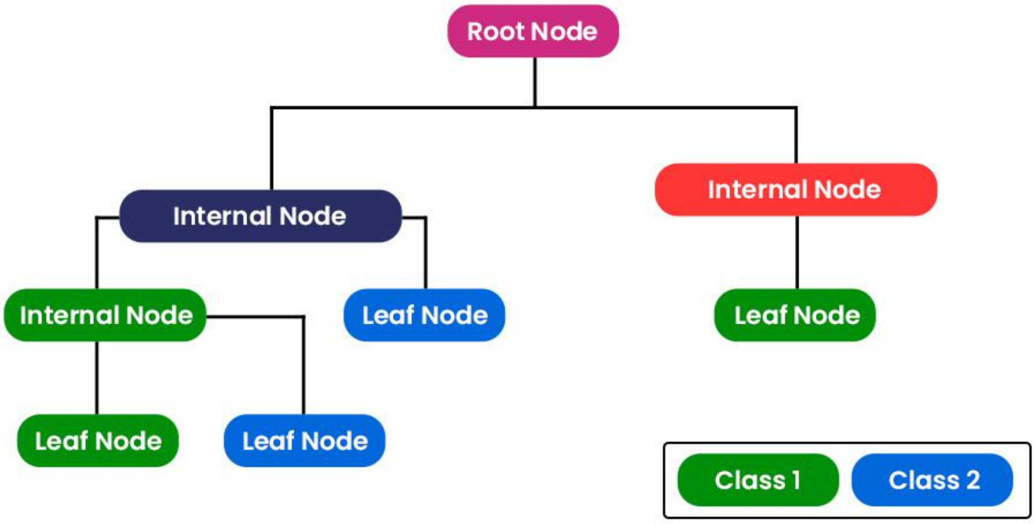
A decision tree is a flowchart-like structure where each internal node represents a "test" on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label or a numerical value (for regression).

## Structure

- Root Node: The topmost node in the tree.
- Internal Nodes: Nodes that test an attribute and branch.
- Branches: Outcomes of a test.
- Leaf Nodes: Final decisions or predictions.

## How it works

- The tree starts at the root node.
- At each internal node, it makes a decision based on an input feature.
- It follows the appropriate branch based on the decision.
- This process continues until it reaches a leaf node, which provides the final prediction.




# Advantages:

- Easy to understand and interpret
- Requires little data preparation
- Can handle both numerical and categorical data
- Can handle multi-output problems

# Disadvantages:

- Can create overly complex trees that don't generalize well
- Can be unstable (small variations in data can result in a very different tree)
- May create biased trees if some classes dominate

# The mathematics behind decision trees

## 1. Splitting Criteria:

The most important mathematical aspect of decision trees is how they decide to split nodes. This is typically done using one of two main metrics:

### a) Entropy and Information Gain (for classification):

Entropy:

$$H(S) = -\Sigma(p_i * log2(p_i))$$

Where p_i is the proportion of class i in the set S.

Information Gain:

$$IG(S, A) = H(S) - \Sigma((|S_v|/|S|) * H(S_v))$$

Where A is the attribute, S_v is the subset of S for which attribute A has value v.

### b) Gini Impurity (for classification):

$$Gini(S) = 1 - \Sigma(p_i^2)$$

Where p_i is the proportion of class i in the set S.

### c) Variance Reduction (for regression):

$$Variance = (1/n) * \Sigma(x_i - \mu)^2$$

Where x_i are individual values and μ is the mean.

## 2. Pruning:

Post-pruning often uses a cost-complexity metric:

$$R_\alpha(T) = R(T) + \alpha|T|$$

Where R(T) is the error rate of the tree T, |T| is the number of leaf nodes, and α is a complexity parameter.

## 3. Prediction (for regression):

For a leaf node, the prediction is typically the mean of the target values in that node:

$$y_{pred} = (1/n) * \Sigma y_i$$

Where $y_i$ are the target values in the leaf node.

## 4. Probability Estimation (for classification):

The probability of a class in a leaf node is the proportion of samples of that class in the node:

$$P(class_k) = (\text{number of samples of } class_k) / (\text{total samples in the node})$$

## 5. Feature Importance:

Often calculated as the total reduction of the criterion brought by that feature:

$$Importance(f) = \Sigma(w * (criterion_{before} - criterion_{after}))$$

Where w is the weighted number of samples reaching that node.

## 6. Stopping Criteria:

Mathematical conditions for stopping tree growth, such as:

- Maximum depth reached
- Minimum samples in a node < threshold
- Improvement in criterion < threshold