Decision Tree Algorithm

Shah Md. Arshad Rahman Ziban

1 What is a Decision Tree?

A Decision Tree is a popular supervised machine learning algorithm used for classification and regression problems. It models decisions in a tree-like structure, breaking down data into smaller subsets based on if-else conditions.

1.1 Why use a Decision Tree?

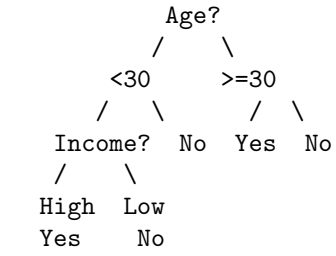
- Simple and easy to interpret
- Works with both categorical and numerical data
- Requires minimal data preprocessing

2 Structure of a Decision Tree

A Decision Tree consists of:

- **Root Node** → The first node, representing the entire dataset.
- **Decision Nodes** → Internal nodes where data splits happen.
- **Leaf Nodes** → The final output (a class label in classification, a value in regression).
- **Branches** → Arrows connecting nodes, representing decision outcomes.

Example:



3 How Does a Decision Tree Work?

The goal is to split the data into pure groups (where most or all labels in a group are the same).

3.1 Step-by-Step Process

1. Choose the best feature to split the data.
2. Split the data based on that feature.
3. Repeat for each child node until all data is classified or stopping conditions are met.

The key question is: How do we determine the best feature to split on?

4 Feature Selection (Splitting Criteria)

To decide which feature to split on, we use mathematical techniques:

4.1 Entropy and Information Gain (ID3 Algorithm)

Entropy measures the impurity in a dataset. The lower the entropy, the purer the data.

$$Entropy(S) = -\sum p_i \log_2(p_i) \tag{1}$$

where p_i is the probability of class i.
Information Gain tells us how much entropy is reduced after a split:

$$IG = Entropy(parent) - \sum \left(\frac{|child|}{|parent|} \times Entropy(child) \right) \tag{2}$$

Higher Information Gain indicates a better split.

4.2 Gini Index (CART Algorithm)

Another way to measure impurity is the Gini Index:

$$Gini = 1 - \sum p_i^2 \tag{3}$$

Lower Gini Index indicates a better split.

4.3 Variance Reduction (For Regression Trees)

Used in regression problems, where the goal is to minimize variance in output values.

$$Variance = \frac{1}{n} \sum (y_i - \bar{y})^2 \tag{4}$$