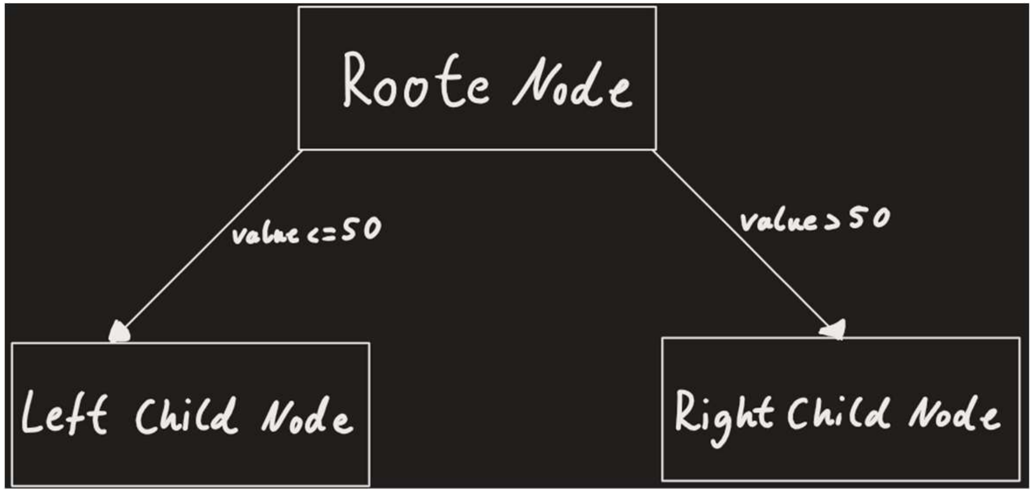
Decision Stumps

- A decision stump is a basic form of a decision tree, consisting of a single root node connected to two leaf nodes. It splits the dataset on one feature.

Example:

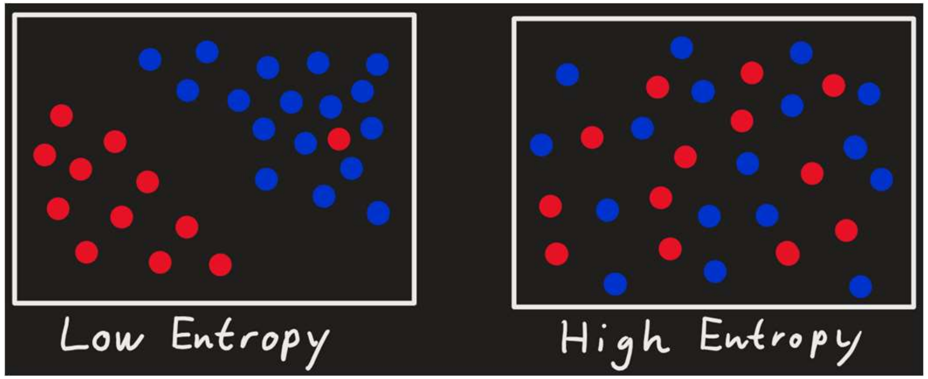
- Root Node: Splits data based on a numerical threshold.
- Left Child Node: Contains instances where the feature value is <= 50.
- Right Child Node: Contains instances where the feature value is > 50.



Entropy and Information Gain

Entropy:

- Entropy measures the impurity or disorder within a dataset. It helps to determine how a feature divides the data into groups that are homogeneous with respect to the target attribute.
- Formula: $H = -\sum_i p(c_i)\ log_2\ p(c_i)$, where $p(c_i)$ is the probability of class c_i.
- A high entropy value indicates a mixture of different classes, whereas a low entropy suggests a homogenous set.



Information Gain:

- Measures the reduction in entropy after a dataset is split on an attribute. It is crucial for determining the best attribute that yields the most informative split at each node in a tree.
- Formula: $\Delta H = H_p\ - [\frac{n_1}{n}H_1 + \frac{n_2}{n}H_2]$, where H_p is the entropy of the parent set before split, and H_1, H_2 are the entropies of the two sets after the split.
- A higher Information Gain value signifies a more effective attribute for splitting the data, leading to purer nodes.

Decision Stump Implementation

1. Data Preparation

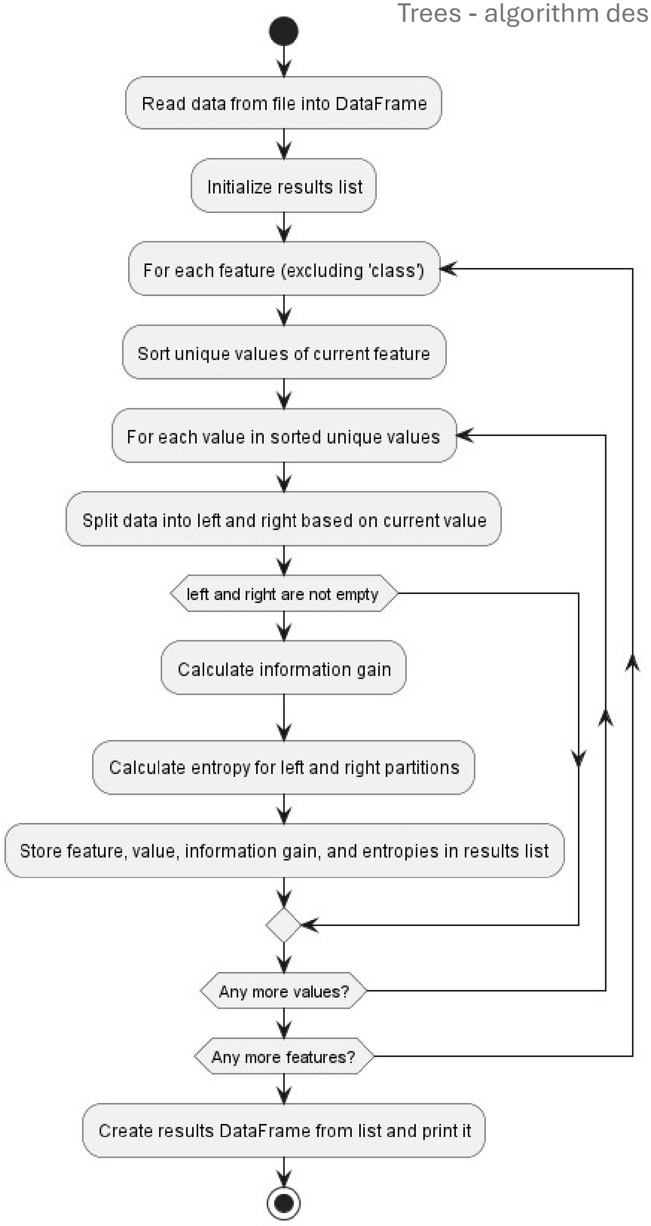
- Load data into DataFrame.
- Initialize an empty list for results.

2. Feature Analysis

- Iterate over each feature, excluding the target ('class').
- Sort values and split data to calculate information gain and entropy.

3. Result Compilation

- Store results (feature, value, information gain, entropy) in a DataFrame and print it.


Read data from file into DataFrame
Initialize results list
For each feature (excluding 'class')
Sort unique values of current feature
For each value in sorted unique values
Split data into left and right based on current value
left and right are not empty
Calculate information gain
Calculate entropy for left and right partitions
Store feature, value, information gain, and entropies in results list
Any more values?
Any more features?
Create results DataFrame from list and print it