Decision Trees

Decision Trees classify data by repeatedly splitting it into smaller and purer groups. The math behind these splits is based on entropy and information gain, which measure how mixed or pure a dataset is.

1. Entropy - Measuring Impurity

Entropy tells us how uncertain a dataset is.
If a group contains both classes equally (50-50), it is highly impure.

Formula

$$\boxed{H(S) = -\sum_{c} P(c) \log_2 P(c)}$$

where:
- S = dataset
- $P(c)$ = proportion of class c.

Intuition
- Entropy = 0 → perfectly pure (all same class)
- Entropy = 1 → perfectly mixed (50% - 50%)
- Entropy increases when classes mix more

Example:

If a node has 8 zeros and 2 ones:

$$P(0) = 0.8, \quad P(1) = 0.2$$

$$H = -[0.8 \log_2 (0.8) + 0.2 \log_2 (0.2)]$$

2. Splitting a Node - Weighted Entropy

When we split a dataset using a feature + threshold, it creates two subsets:

$$S_{left}, \; S_{right}$$

Each subset has its own entropy.
The ~~total~~ impurity after split:

$$\boxed{H_{split} = \frac{|S_L|}{|S|} H(S_L) + \frac{|S_R|}{|S|} H(S_R)}$$

This is a weighted average - bigger groups contribute more to impurity.

3. Information Gain – How Good a Split Is

A good split reduces impurity.

Information Gain (IG) quantifies how much entropy decreases after the Split :

$$\boxed{IG = H(S) - H_{split}}$$

We choose the split with the highest Information Gain.

Intuition:

- If children are purer → IG is high → good split
- If purity does not improve → IG is low → bad split.

---

4. Choosing the Best Split

For each feature X_j and threshold t:

1. Divide data into left / right:

$$X_j \le t, \quad X_j > t$$

2. Compute entropies:

$$H(S_L), H(S_R)$$

3. Compute Information Gain:

$$IG(X_j, t) = H(S) - H_{split}$$

4. Pick the feature-threshold pair giving maximum IG.

---

5. Stopping Conditions (When to Stop Splitting)

A node becomes a leaf when:

- All samples are same class (entropy = 0)
- Minimum samples reached
- No IG improvement possible
- Max depth reached

whatever class is majority in that node is the output:

$$\boxed{\text{Leaf prediction} = \arg\max_{c} P(c)}$$

6. Prediction Math (Traversing the Tree)

For a given input vector x:

- Start at root node
- Check condition

$$\boxed{x_{feature} \leq threshold?}$$

- Move left or right
- stop at a ~~Bo~~ leaf and output its class

This is simple rule-based ~~evalut~~ evaluation; ~~not go~~

Summary

- Entropy measures impurity.
- Information Gain measures how much a split improves purity.
- Decision Trees try every possible split and choose the one with the highest IG.
- Recursively splitting creates the full tree.
- Leaves predict by choosing the most common class.
