Decision Trees, Entropy, Gini Impurity, and Information Gain

---

1. Decision Trees – Big Picture

A **Decision Tree** is like a flowchart:

- Each **node** → a question/split about a feature.
- Each **branch** → the outcome of that question.
- Each **leaf** → a final decision (class or prediction).

The tree keeps **splitting data** into purer subsets until:

- All samples in a node belong to one class, or
- A stopping condition is reached (max depth, min samples, etc.).

The goal:

Choose splits that make the data in each branch as pure (homogeneous) as possible.

---

2. Geometric Intuition

Imagine a **2D feature space** (like Height vs. Weight).

- Decision Trees split this space into **rectangular regions** by drawing straight horizontal/vertical lines.
- Each split **reduces uncertainty** about the class labels.
- For example:
    1. First split: "Is Height > 170 cm?" → divides into tall vs. short groups.
    2. Second split (on the "short" group): "Is Weight < 60 kg?" → further refines.

The process is recursive — every new split tries to isolate points of the same class.

---

3. How Decision Trees Decide Where to Split

The algorithm looks at **all possible features** and **possible split points**.

For each possible split, it calculates a **score** that measures how "good" the split is.

That score comes from metrics like:

---

4. Entropy

Meaning:

Entropy measures **disorder/randomness** in a set.

- If all examples in a set are of the same class → Entropy = 0 (pure).
- If classes are perfectly mixed → Entropy is maximum.
- More knowledge -> less entropy

Formula:

$$Entropy(S) = -\sum_{i=1}^{n} p_i \log_2(p_i)$$

Where:

- p_i = proportion of class i in the set S
- The log is base 2 (measured in bits)

Example:

Dataset: 4 apples, 4 oranges

- $p_{apple} = 4/8 = 0.5$
- $p_{orange} = 0.5$

$$Entropy = -(0.5 \log_2 0.5 + 0.5 \log_2 0.5) = 1$$

Max entropy → fully mixed.

---

5. Gini Impurity

Meaning:

Probability of **randomly picking the wrong class** if you randomly label an item according to the class distribution.

Formula:

$$Gini(S) = 1 - \sum_{i=1}^{n} p_i^2$$

Example:

Same 4 apples, 4 oranges:

$$Gini = 1 - (0.5^2 + 0.5^2) = 0.5$$

Lower Gini → purer node.

Key Difference:

- **Entropy** uses logarithm → more sensitive to changes in probabilities.
- **Gini** is simpler computationally and often works just as well.

---

6. Information Gain

Purpose:

Measures **how much uncertainty is reduced** after a split.

Formula:

$$Information\ Gain = Entropy(Parent) - \sum_k \frac{|S_k|}{|S|} \cdot Entropy(S_k)$$

Where:

- S_k are subsets after the split
- Weighted average ensures bigger subsets count more.

Interpretation:

High Information Gain → split made subsets much purer.

---

7. Handling Numerical Data

- For continuous values (e.g., "Age"), we test splits like:
    - Age ≤ 30 vs Age > 30
- Try many thresholds → choose the one with best score (highest IG or lowest Gini).

---

8. Advantages

- Easy to interpret & visualize
- No feature scaling needed
- Handles numerical & categorical data
- Captures non-linear relationships

---

9. Disadvantages

- Prone to overfitting (fix with pruning, max depth)
- Unstable (small changes in data can change tree)
- Greedy splitting → may miss global optimum

---

10. CART

- **CART** = Classification and Regression Trees
- For classification → often uses Gini Impurity
- For regression → uses variance reduction (MSE)

---

11. Type of Errors in Context

While not specific to trees, misclassification errors still apply:

- **Type 1 Error (False Positive)**: Predict "Yes" when actual is "No"
- **Type 2 Error (False Negative)**: Predict "No" when actual is "Yes"

✅ Core Memory Hook:

- **Entropy** = disorder, log-based
- **Gini** = misclassification probability
- **Information Gain** = reduction in disorder
- Trees = keep splitting until pure or limit reached
