**Decision Tree Classification**

A **Decision Tree** is a **supervised machine learning algorithm** used for **classification (and regression)** that splits data into branches based on feature values, forming a **tree-like structure**.

It makes decisions **just like a flowchart**.

**Core Idea (Intuition)**

At each step, choose the **best feature** to split the data so that the resulting groups are **as pure as possible**.

* Root node → first split
* Internal nodes → decisions
* Leaf nodes → final class labels

**Tree Structure**

* Root Node
* (Best Feature)
* / \
* Decision Decision
* Node Node
* \ /
* Leaf Nodes
* (Class Output)

This is how the tree like structure looks

Yes

No

Not Qualified?

Qualified?

Some Test

Not Placed?

No

Placed?

Should go for higher education or not?

**How Decision Tree Classification Works (Step-by-Step)**

**🔹 Step 1: Select the Best Feature**

The algorithm evaluates all features and chooses the one that:

* Best separates the classes
* Creates the **purest child nodes**

**🔹 Step 2: Split the Dataset**

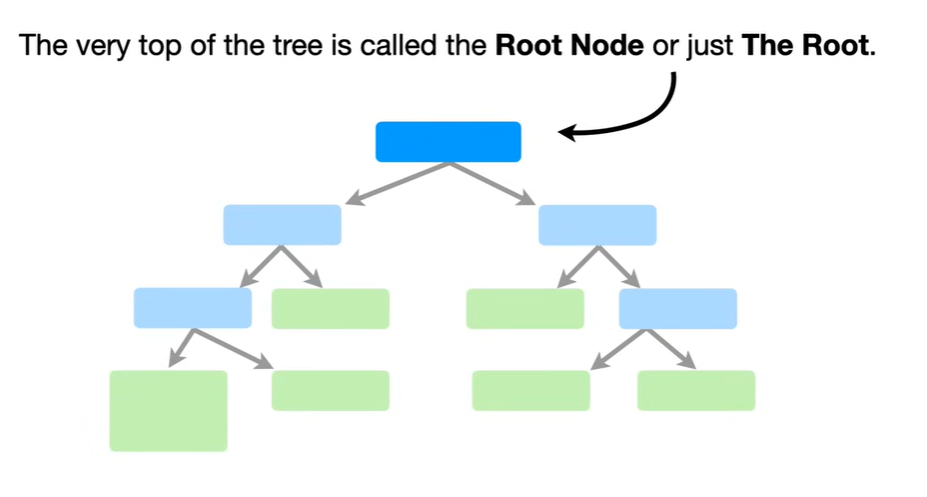
* Data is split based on feature values
* Each split forms a new branch

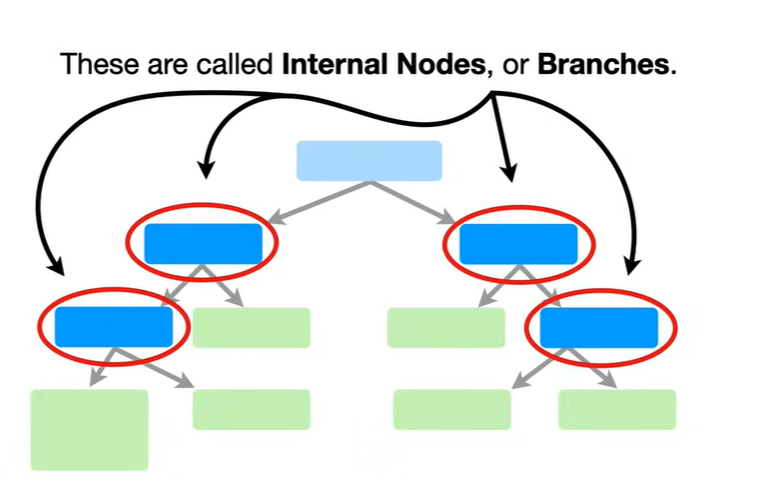
**🔹 Step 3: Repeat Recursively**

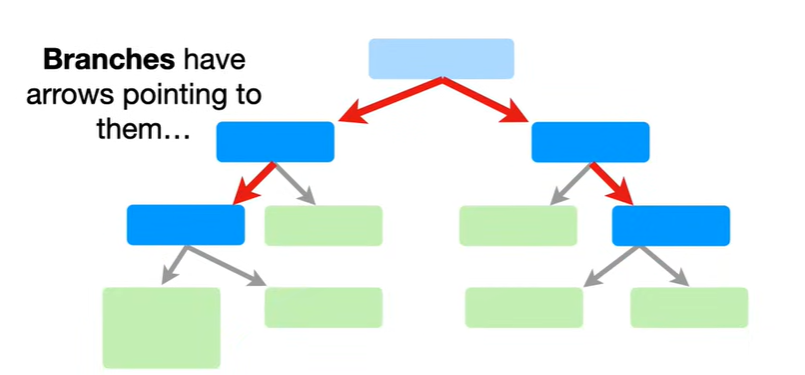
* Apply the same process to each child node
* Stop when:
  + All samples belong to one class, or
  + Max depth is reached, or
  + No further improvement is possible

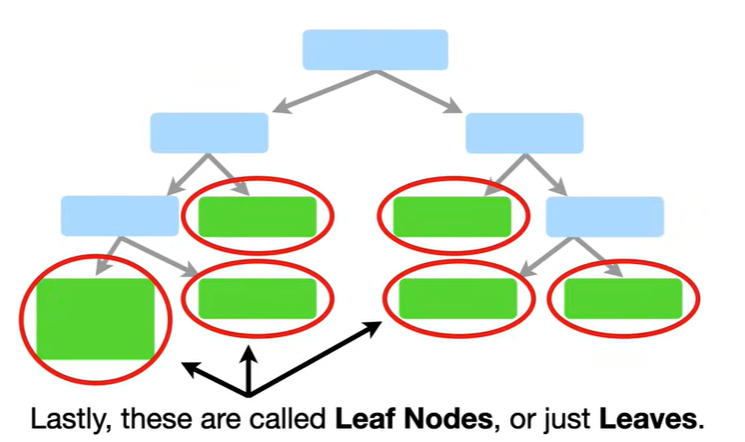
**🔹 Step 4: Assign Class to Leaf Node**

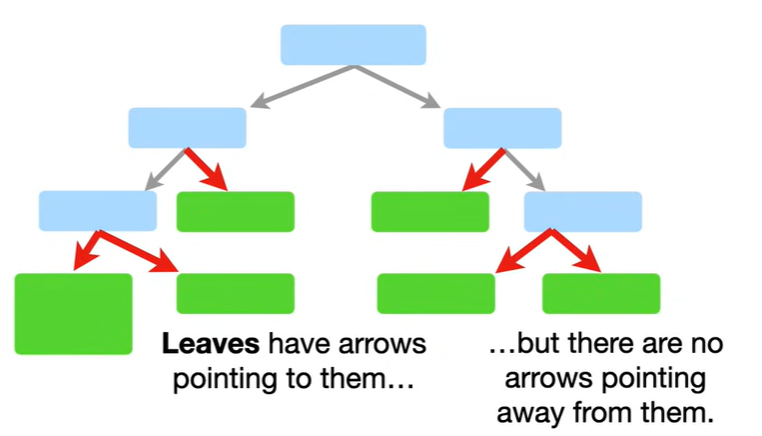
* Leaf node outputs the **majority class**

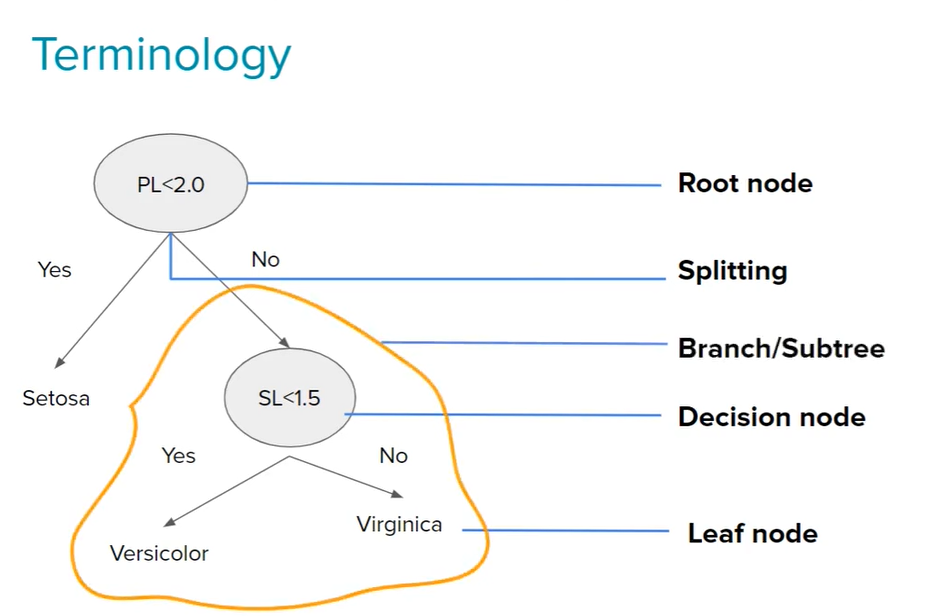












**5️⃣ How “Best Split” Is Chosen (Important)**

Decision trees use **impurity measures**:

**🔸 1. Gini Impurity (CART)**

Measures how often a randomly chosen element would be misclassified.

* Lower Gini = purer node

**🔸 2. Entropy & Information Gain (ID3, C4.5)**

* **Entropy** measures disorder
* **Information Gain** measures reduction in entropy after split

Choose feature with **maximum information gain**

**What is Entropy?**

Entropy is a measure of uncertainty or impurity in a dataset.  
It tells us how mixed the classes are in a subset of data.

1️⃣ Intuition

* If all samples in a group belong to one class, entropy = 0 → perfectly pure
* If samples are evenly split among classes, entropy = maximum → very impure

**Formula of Entropy**

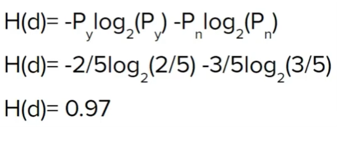
For a dataset with classes:

Where:

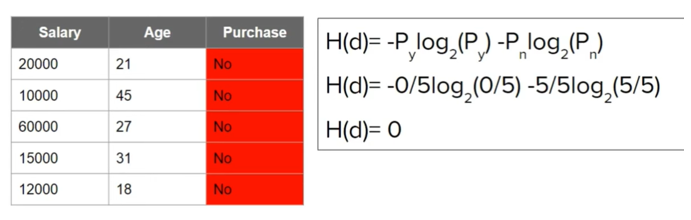
* = proportion of samples in class
* Sum over all classes

****

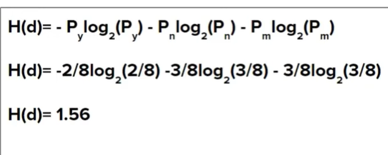
**Entropy of first dataset after applying the formula is 0.97**

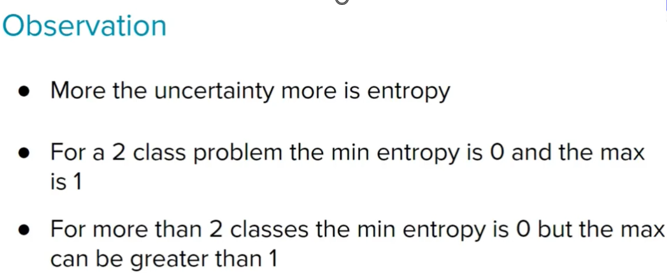
****

**Similarly, Entropy of second dataset after applying the formula is 0.72**

****

**Calculating entropy for a 3 class problem**



**How it works in decision trees**

* **High entropy**: A node with high entropy has a large mix of different classes. For example, a node with an equal number of "yes" and "no" outcomes has the highest possible entropy.
* **Low entropy**: A node with low entropy is close to being "pure," meaning it contains data points of predominantly one class. A node with only "yes" outcomes has zero entropy.
* **Splitting the data**: Decision trees use entropy to decide the best way to split the data at each step. They choose the feature and value that results in the biggest decrease in entropy (the highest [information gain](https://www.google.com/search?q=information+gain&rlz=1C1CHBF_enIN1086IN1086&oq=what+is+entropy+in+decision+tree&gs_lcrp=EgZjaHJvbWUqDAgAEAAYFBiHAhiABDIMCAAQABgUGIcCGIAEMgwIARAAGBQYhwIYgAQyCAgCEAAYFhgeMggIAxAAGBYYHjIICAQQABgWGB4yCAgFEAAYFhgeMggIBhAAGBYYHjIICAcQABgWGB4yCAgIEAAYFhgeMggICRAAGBYYHtIBCDg0NThqMGo3qAIAsAIA&sourceid=chrome&ie=UTF-8&mstk=AUtExfBnY-xO_QmWvg4-Yq1Q6cKz-TpAPLQ9xQ5tbtZLkP2bbpIyou-91tAuWPC8F0q_D3z75wDN0pOviiCm9wQXtaGpCxUpJXPo_nEtNEOpGow4MafuHEWmEevWlJUn-rD-VjCZZqB31P381kIULVukjZzCGgha-ZLcK8OBRTSx-pvbNjE&csui=3&ved=2ahUKEwj4hKyF8IGRAxX_IzQIHVTPG3UQgK4QegQIBRAD))
* **Reducing uncertainty**: By splitting the data, the tree aims to reduce the uncertainty (entropy) at each level. This process continues until a leaf node is reached, which contains a classification based on the data that has been filtered at each split.
* **Goal**: The objective is to create a tree that results in the lowest possible entropy at the final leaf nodes, making it easier to accurately classify new data points.

**Information Gain**

Decision Trees choose the feature that reduces entropy the most after splitting.

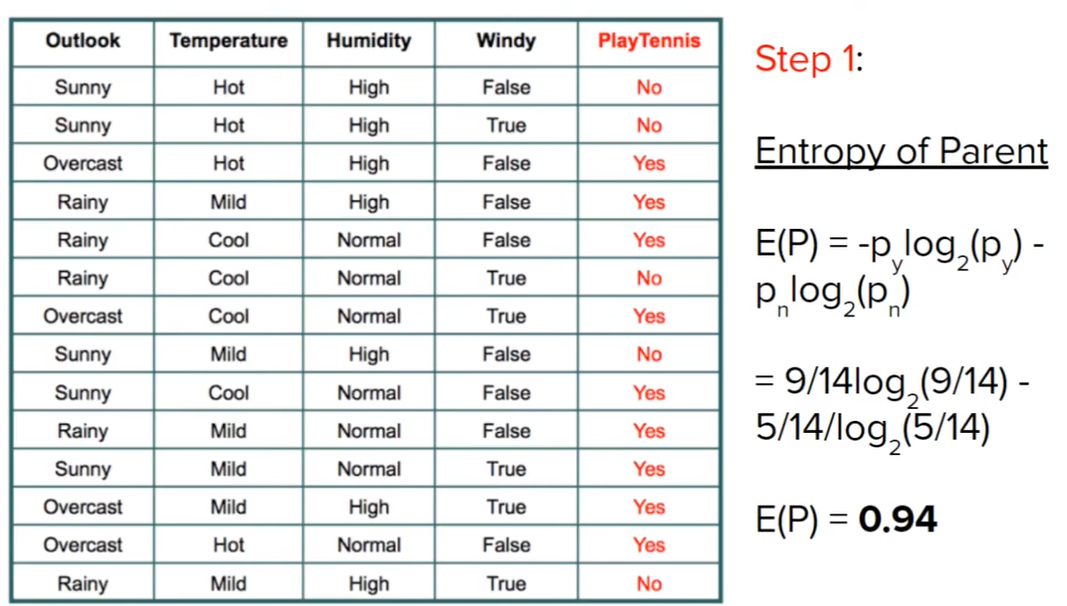
This reduction is called Information Gain (IG):

* Feature with highest IG → chosen for split
* Goal: make children as pure as possible

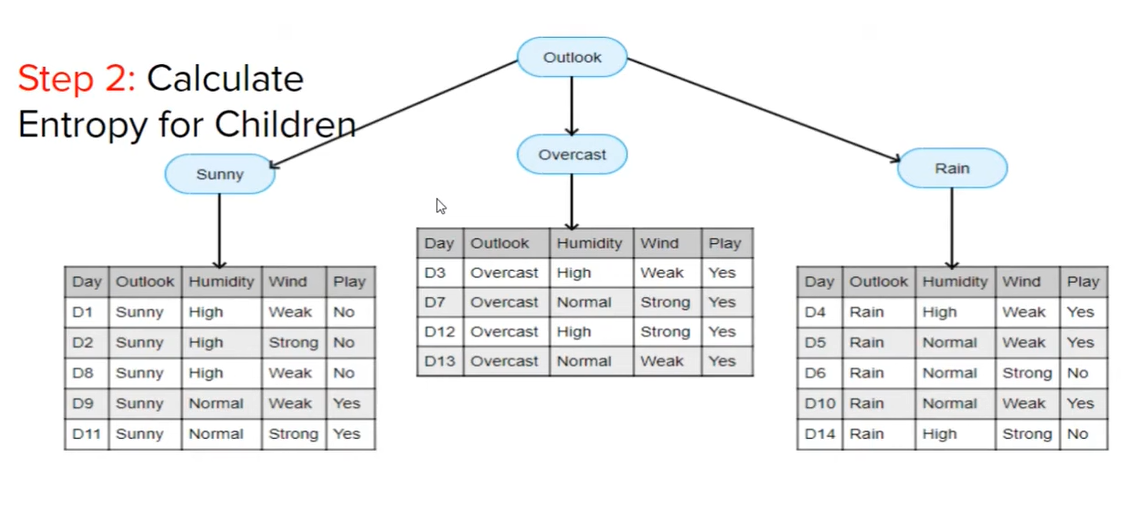


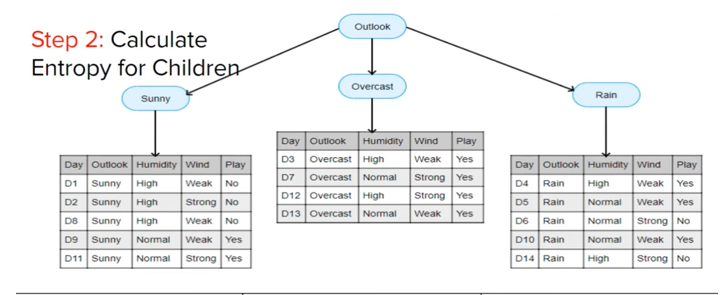
Here are the steps how splitting criteria works using information gain

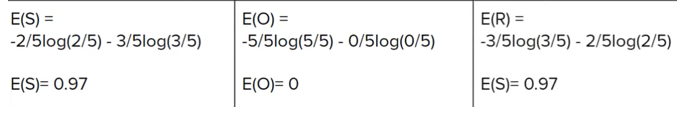
First we choose a feature, let say ‘Outlook’. We consider it as a parent feature we calculate its entropy

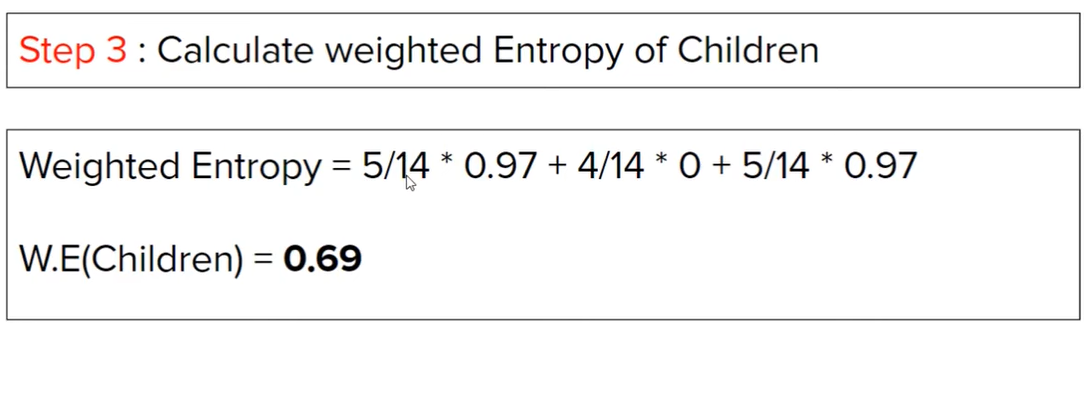


Then the ‘Outlook’ feature has 3 children – Sunny, Overcast and Rain. We calculate their entropy









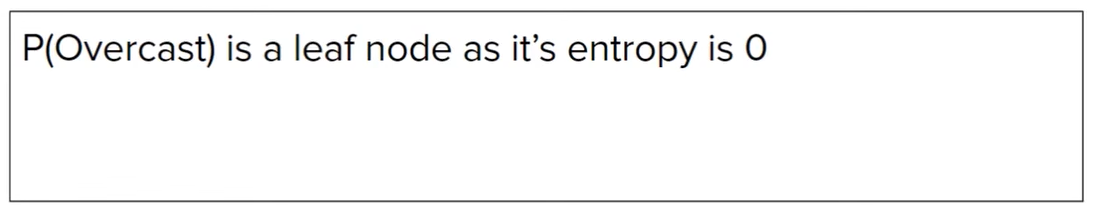
Weight is calculated as follows

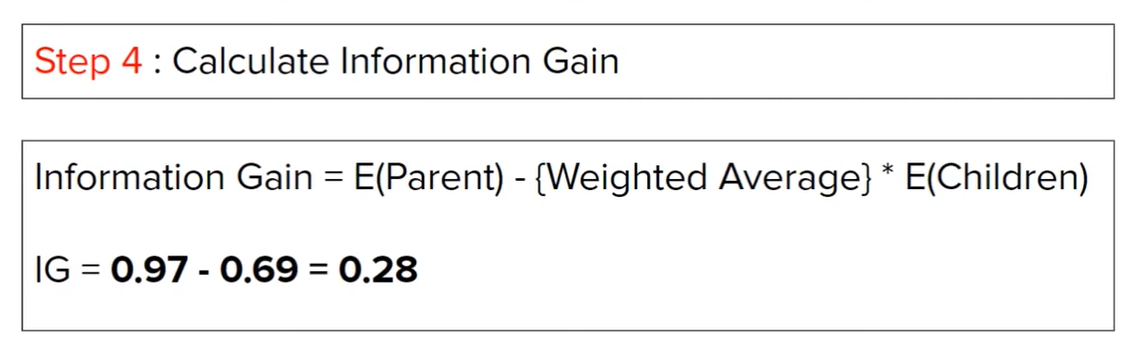
Total rows = 14

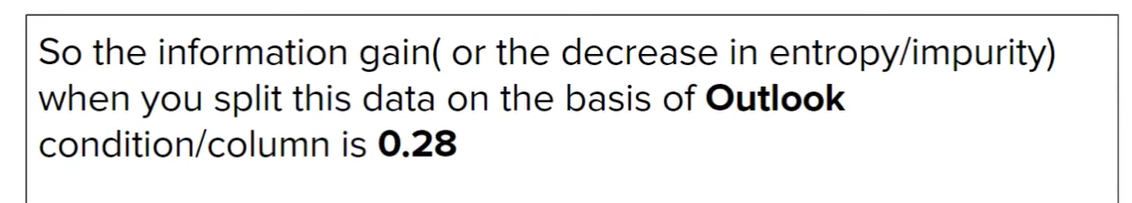
Sunny rows = 5

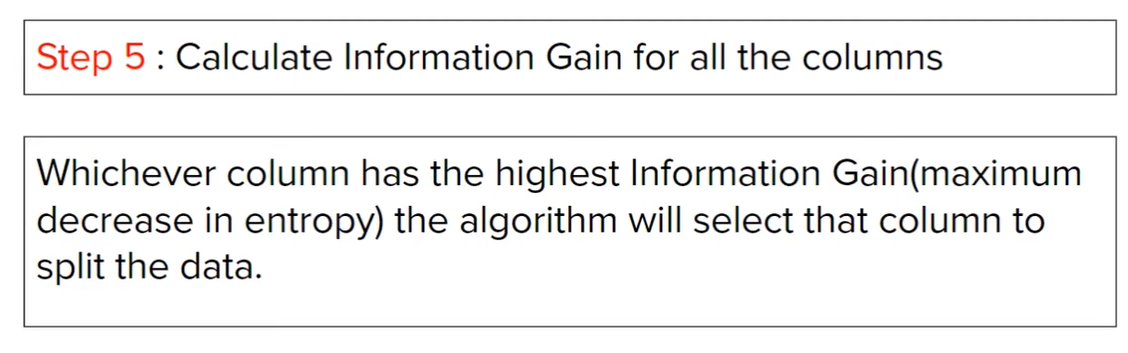
Overcast rows = 4

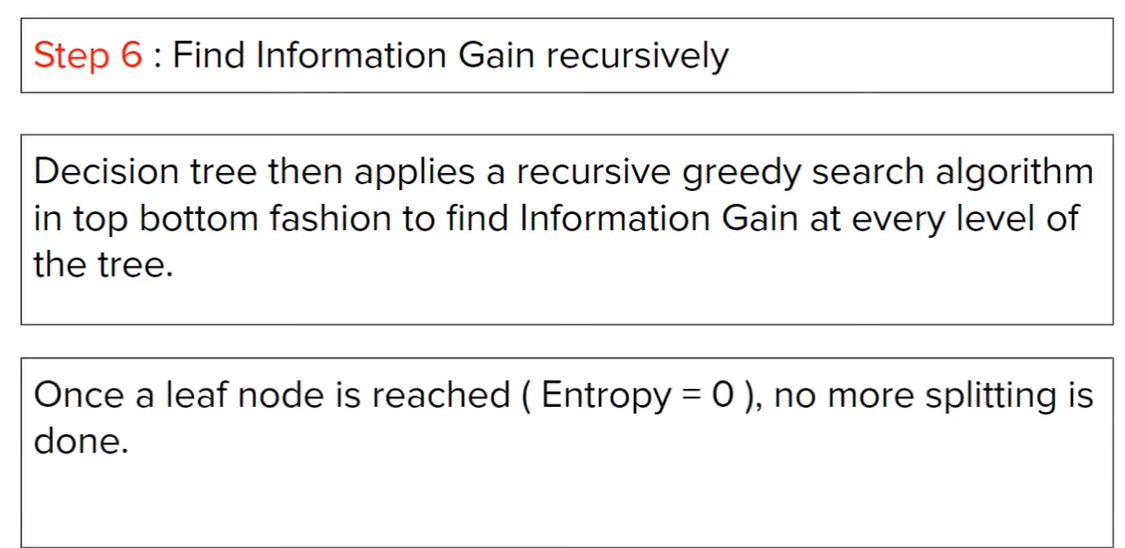
Rain rows = 5











**Gini Impurity**

**Gini Impurity** is another way to measure **how impure or mixed a node is** in a decision tree.

* Like entropy, it tells us **how “uncertain” the class labels are** in a subset.
* Used mainly in **CART (Classification and Regression Trees)**.

**1️⃣ Intuition**

* **Gini = 0** → all samples belong to one class → pure node
* **Gini = 0.5 (for 2 classes)** → node is evenly split → maximum impurit

**2️⃣ Formula**

For a node with classes:

Where:

* = proportion of samples in class

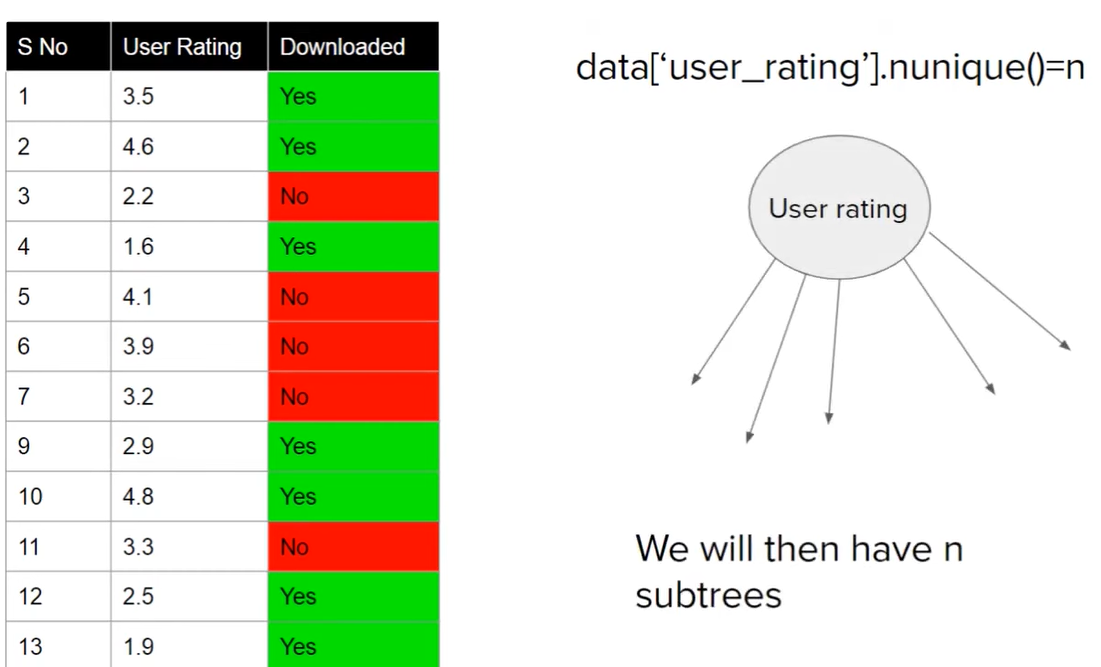


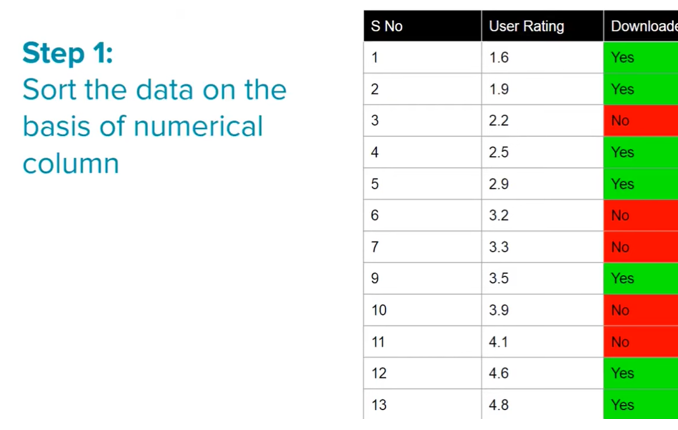


Gini is fast

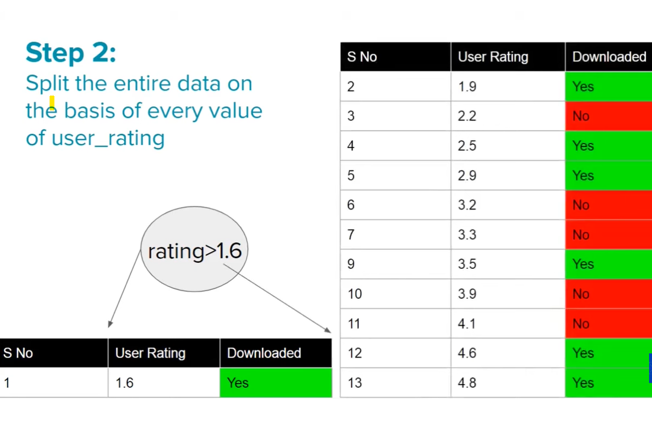
Entropy provides balanced trees

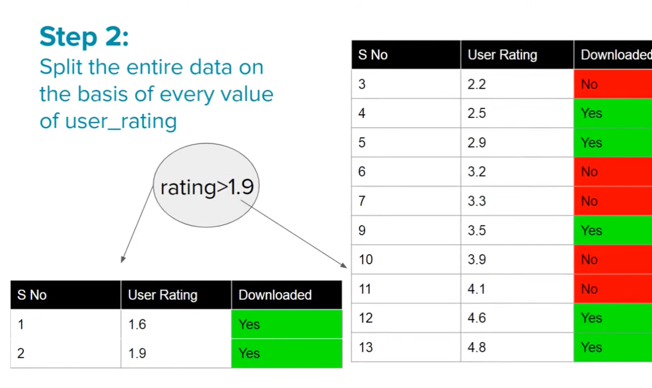
**How Decision Trees work with Numerical Data?**



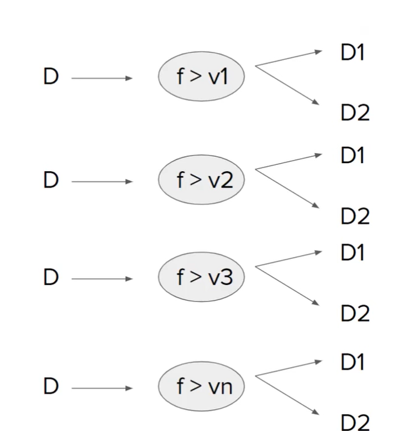
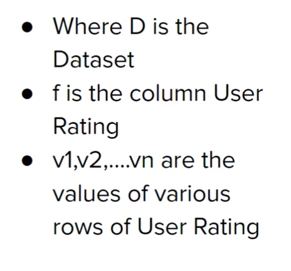


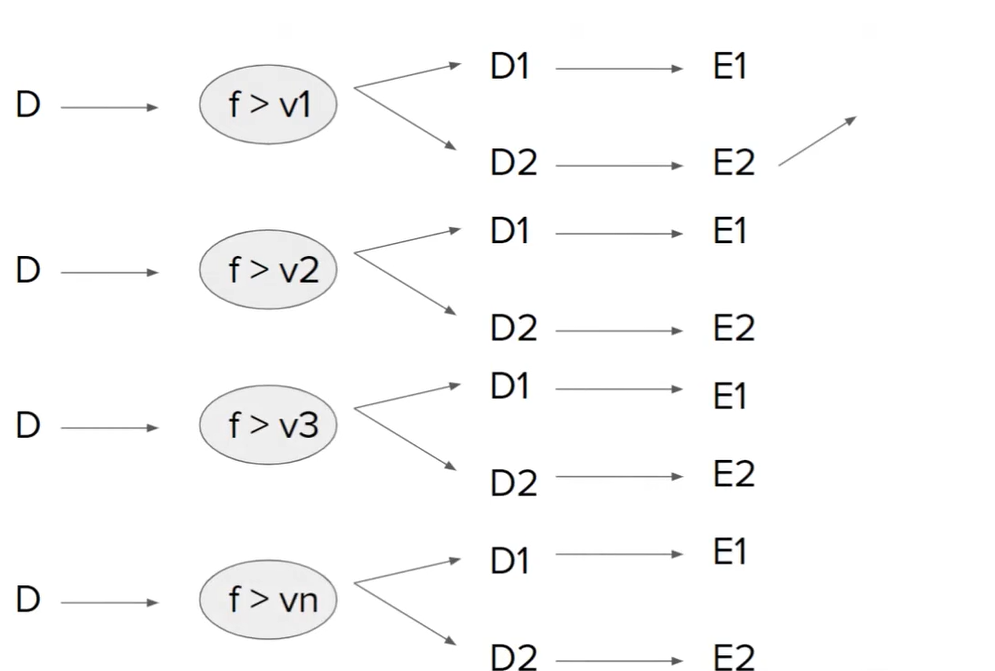
Lets say we take User Rating column and split the data in ascending order

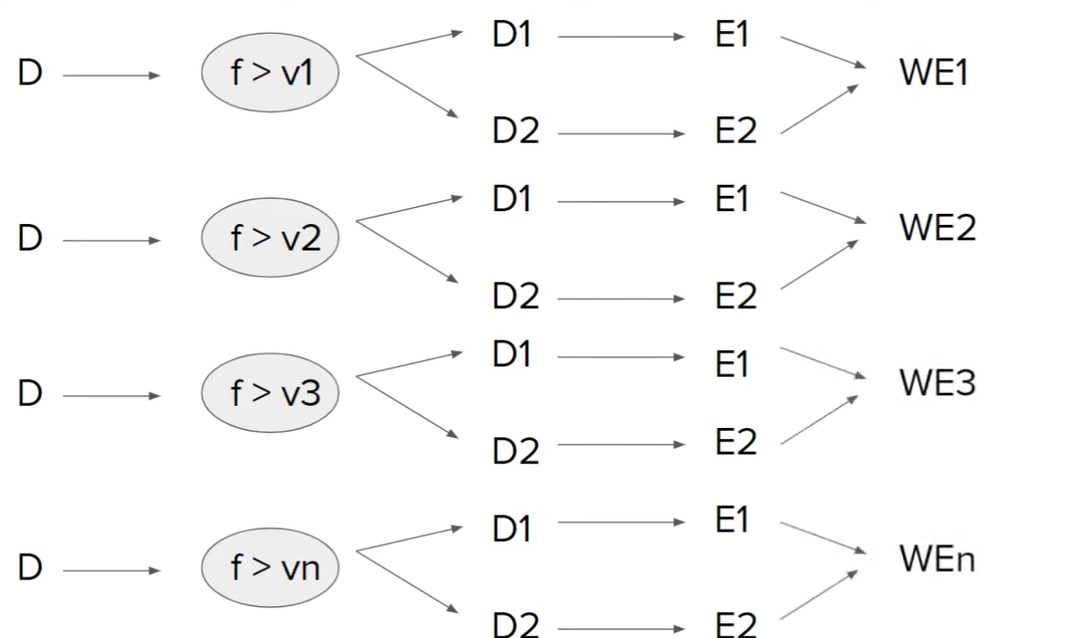




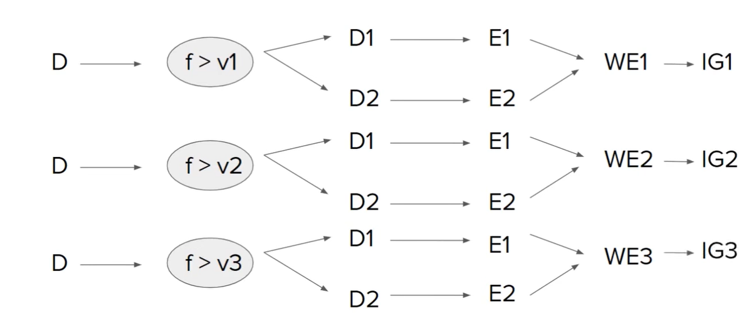


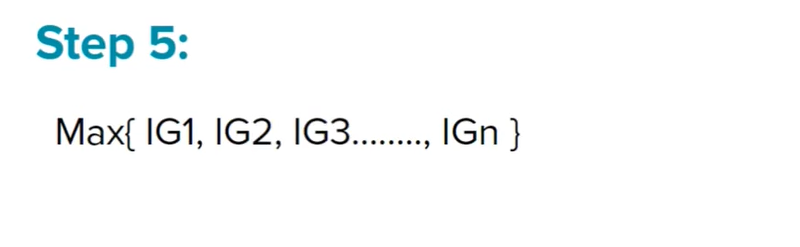




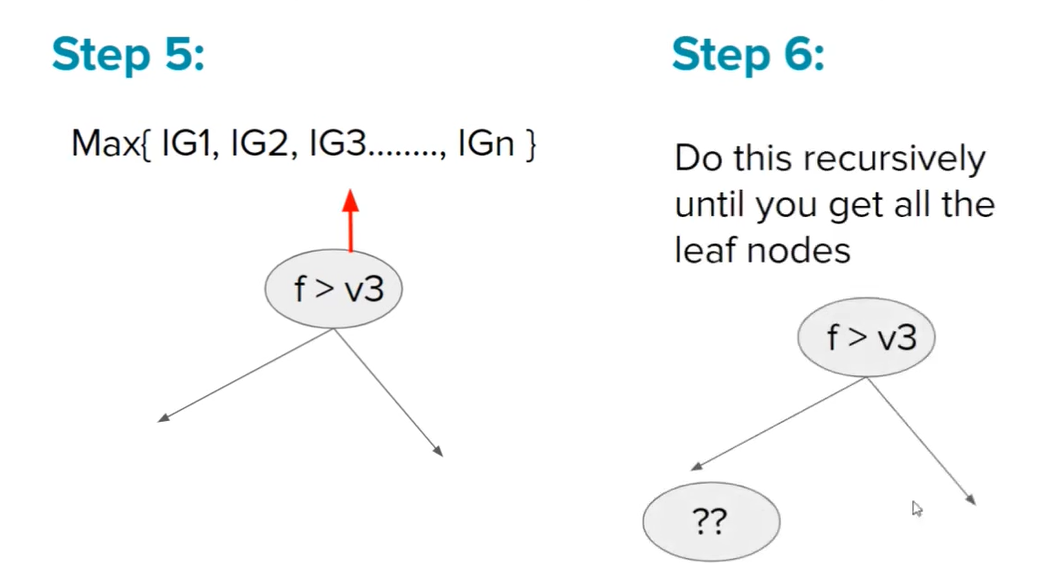
For each split entropy and weighted entropy is calculated



And then the information gain is calculated



Maximum information gain is picked and then at the point at which we get the maximum information gain, that point is picked as the best splitting criteria



**Pruning**

**Pruning** is the process of **removing parts of a decision tree that do not improve generalization** in order to **reduce overfitting**.

In simple words: “Cut the branches that don’t help the model perform better on unseen data.”

**Why Pruning Is Needed**

Decision trees **tend to overfit** when:

* The tree grows very deep

**Symptoms of overfitting:**

* Training accuracy → very high
* Testing accuracy → low

Pruning **simplifies the tree** and improves performance on unseen data.

**2️⃣ Types of Pruning**

**1️⃣ Pre-Pruning (Early Stopping)**

* Stop growing the tree **before it fully fits the data**
* Common methods:
  + Set **max\_depth** → maximum tree depth
  + Set **min\_samples\_split** → minimum samples to split a node
  + Set **min\_samples\_leaf** → minimum samples in a leaf node
  + Set **max\_leaf\_nodes** → limit number of leaves

**2️⃣ Post-Pruning (Cost-Complexity / Reduced Error Pruning)**

* Grow the full tree first → then remove unnecessary branches
* Techniques:
  + **Reduced error pruning:** Remove nodes if validation accuracy does not decrease
  + **Cost-complexity pruning (α-pruning in scikit-learn):** Removes branches that **increase complexity more than they improve accuracy**