

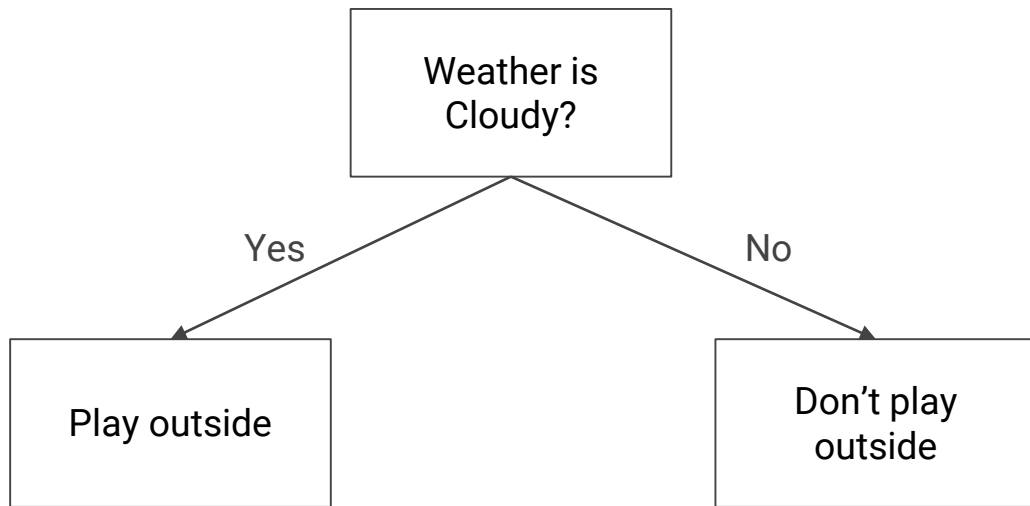
Decision Tree & Intro to Ensemble Techniques

Dipon Talukder

Assessment Developer (Data and AI),
Workera.ai

Research Assistant, Universal Machine.io
Adjunct Faculty, East Delta University

Decision Tree

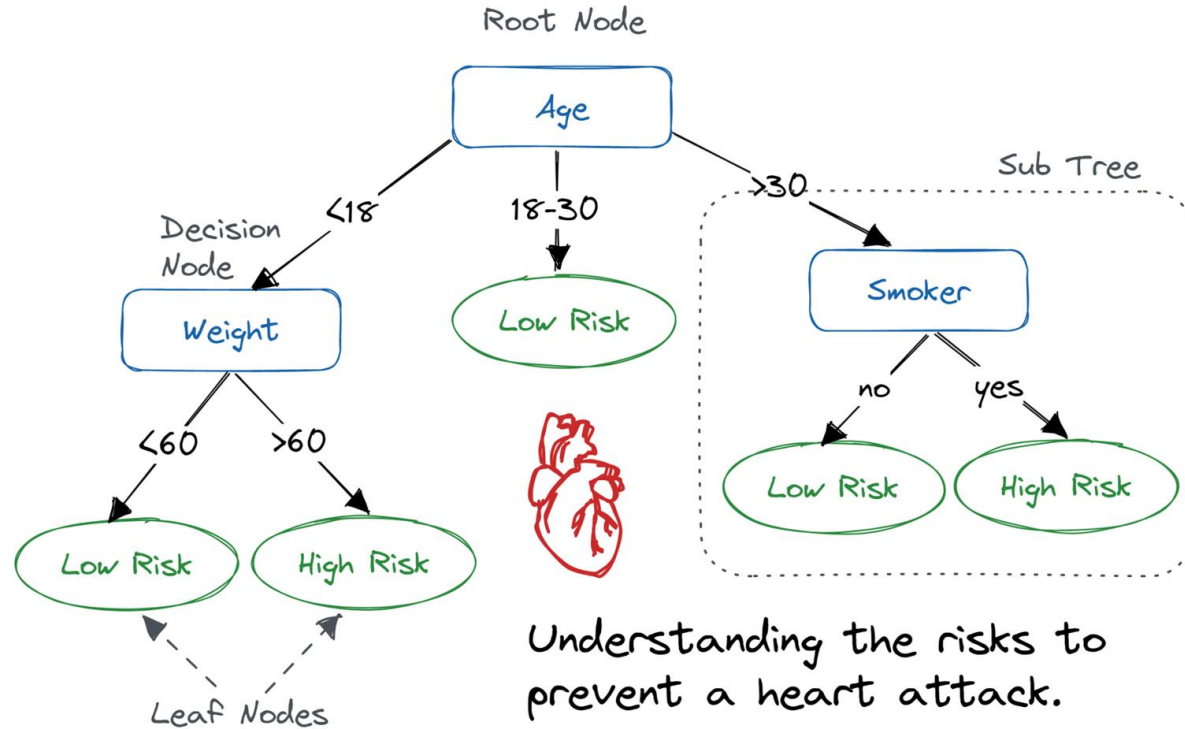


This is called a classification tree.

There is also a regression tree. That means DT can be used for regression tasks too.

But our today's class will focus on classification tree only.

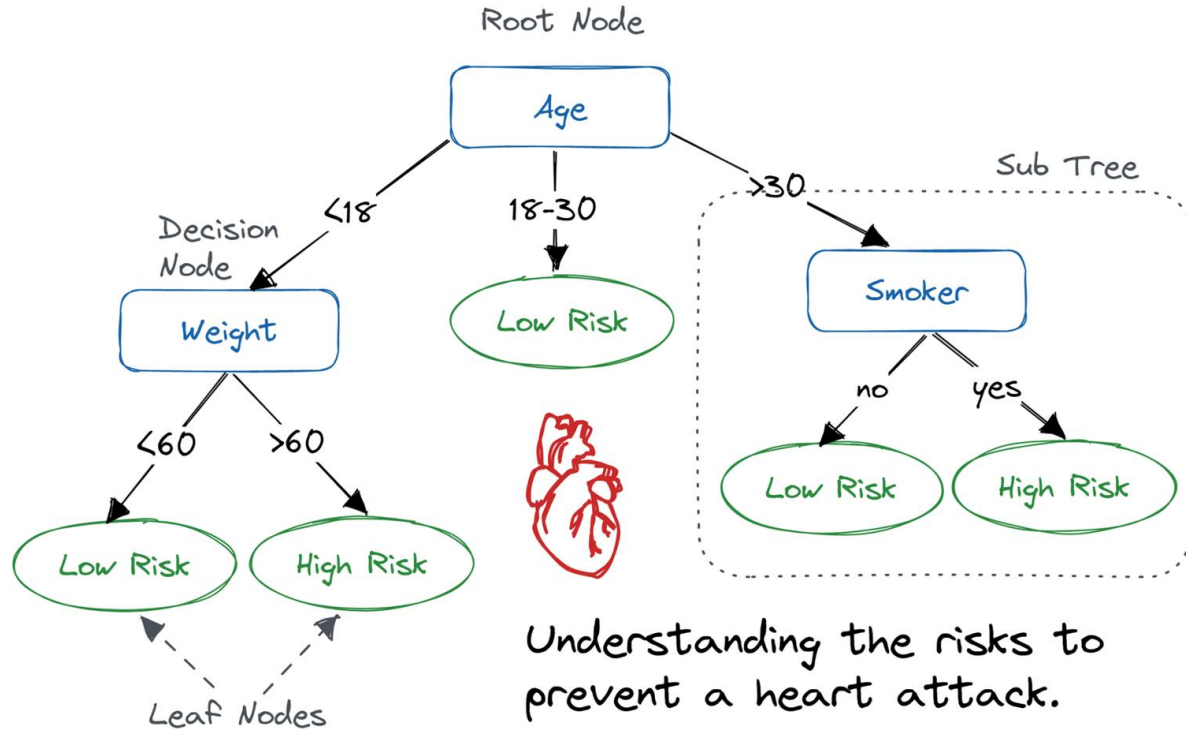
Decision Tree



It can work with both numerical(age, weight) and categorical (smoker) values.

Start from the top and just move down until the final node is reached.

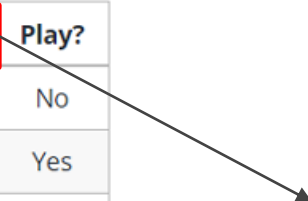
Tree Terminologies



- Root Node
- Decision Nodes / Internal Nodes
- Leaf Nodes
- Branch / Sub-Tree
- Parent and Child Node

Brief Intro to Building a Decision Tree

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Sunny	Mild	Normal	Strong	Yes
4	Cloudy	Mild	High	Strong	Yes
5	Rainy	Mild	High	Strong	No
6	Rainy	Cool	Normal	Strong	No
7	Rainy	Mild	High	Weak	Yes
8	Sunny	Hot	High	Strong	No
9	Cloudy	Hot	Normal	Weak	Yes
10	Rainy	Mild	High	Strong	No

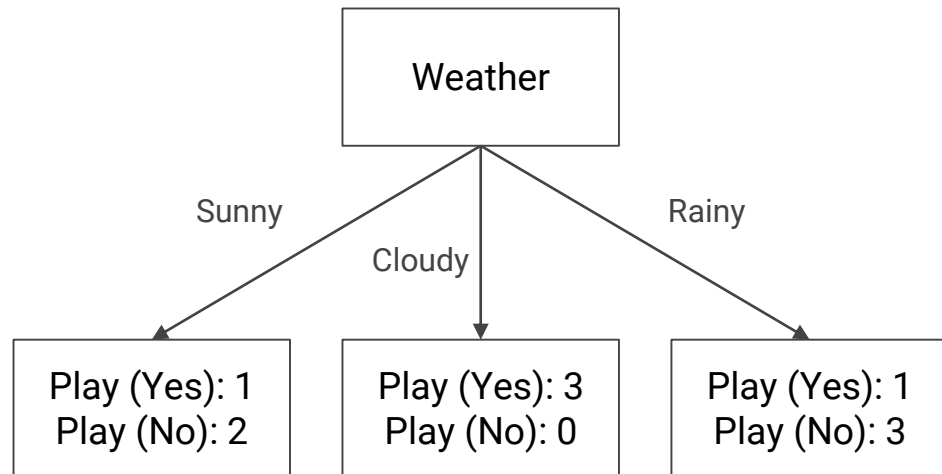


Root ?????
What question
should be asked
first?

Let's see which one predicts the 'Play'
best out of these four features.

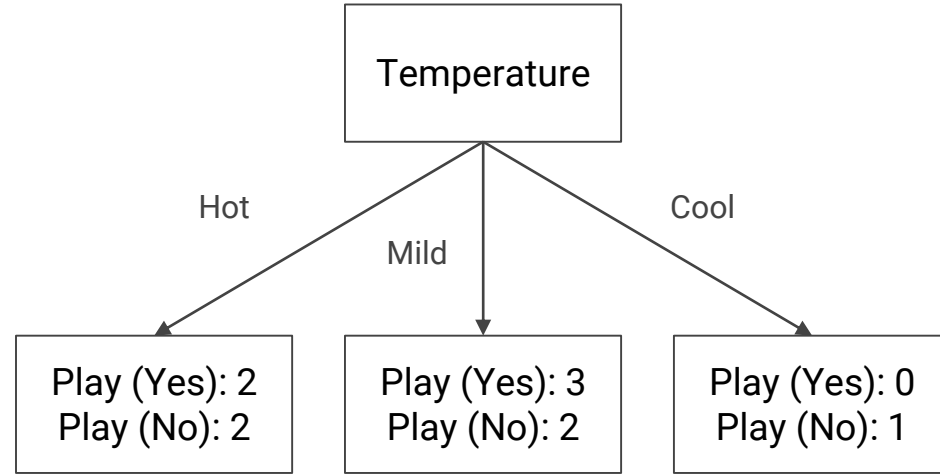
Brief Intro to Building a Decision Tree

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Sunny	Mild	Normal	Strong	Yes
4	Cloudy	Mild	High	Strong	Yes
5	Rainy	Mild	High	Strong	No
6	Rainy	Cool	Normal	Strong	No
7	Rainy	Mild	High	Weak	Yes
8	Sunny	Hot	High	Strong	No
9	Cloudy	Hot	Normal	Weak	Yes
10	Rainy	Mild	High	Strong	No



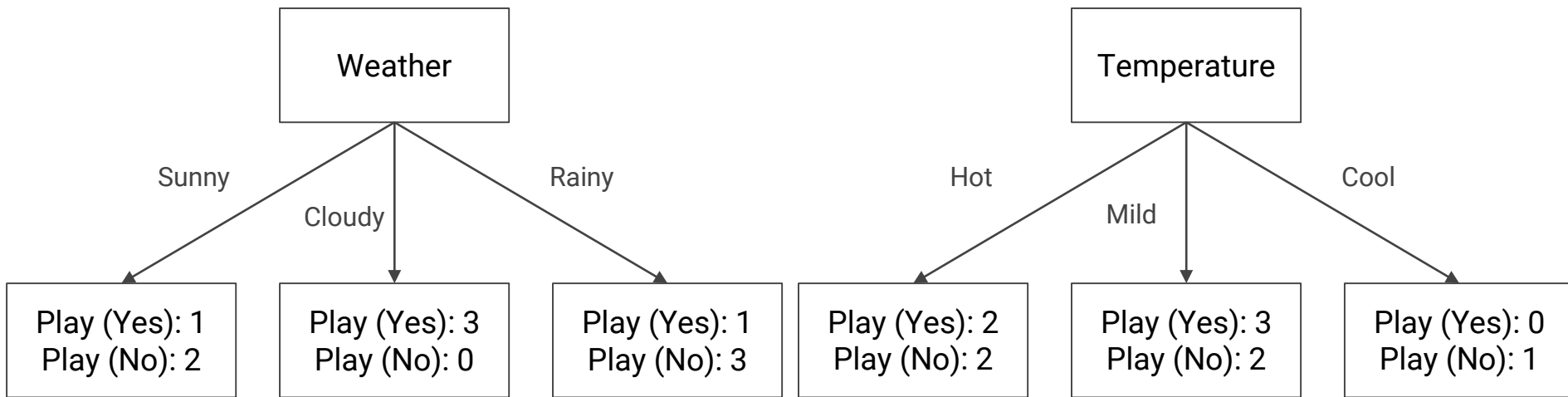
Brief Intro to Building a Decision Tree

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Sunny	Mild	Normal	Strong	Yes
4	Cloudy	Mild	High	Strong	Yes
5	Rainy	Mild	High	Strong	No
6	Rainy	Cool	Normal	Strong	No
7	Rainy	Mild	High	Weak	Yes
8	Sunny	Hot	High	Strong	No
9	Cloudy	Hot	Normal	Weak	Yes
10	Rainy	Mild	High	Strong	No



Same for Humidity and Wind...

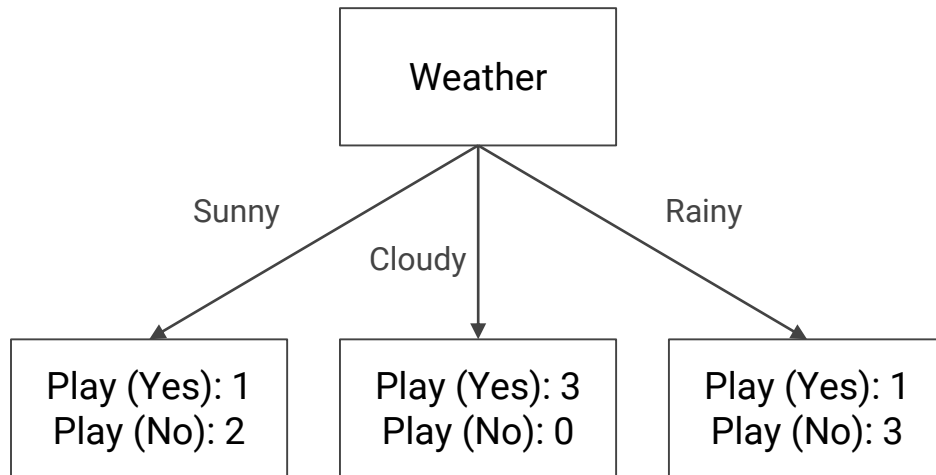
Comparing Two Features



These are called “Impurity”

- We need to quantify these “impurities”
- Most common methods are: **Gini Impurities**, Entropy, Information Gain

Gini Impurity



Gini Impurity of a Leaf = $1 - (\text{Probability of "Yes"})^2 - (\text{Prob of "No"})^2$

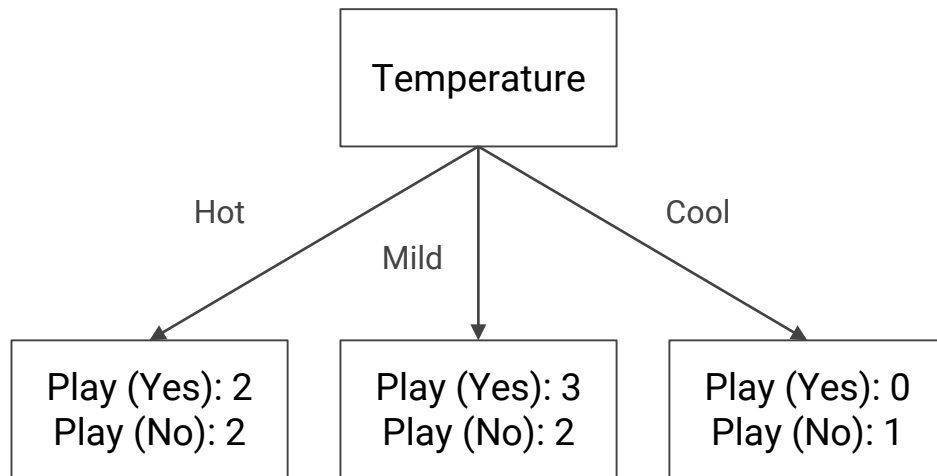
So, Gini Impurity (Sunny) = $1 - (\frac{1}{3})^2 - (\frac{1}{4})^2 = 0.82$

Gini Impurity (Cloudy) = $1 - (\frac{3}{3})^2 - (\frac{0}{3})^2 = 0$

Gini Impurity (Rainy) = $1 - (\frac{1}{4})^2 - (\frac{3}{4})^2 = 0.375$

Gini Impurity (Weather) = Weighted average of all Gini Impurities
= $(3 * 0.82 + 3 * 0 + 4 * 0.75) / 10 = 0.396$

Gini Impurity



Gini Impurity of a Leaf = $1 - (\text{Probability of "Yes"})^2 - (\text{Prob of "No"})^2$

So, Gini Impurity (Hot) = $1 - (2/4)^2 - (2/4)^2 = 0.5$

Gini Impurity (Mild) = $1 - (3/5)^2 - (2/5)^2 = 0.48$

Gini Impurity (Cool) = $1 - (0/1)^2 - (1/1)^2 = 0$

Gini Impurity (Temperature) = Weighted average of all Gini Impurities
= $(4 * 0.5 + 5 * 0.48 + 0 * 1) / 10 = 0.44$

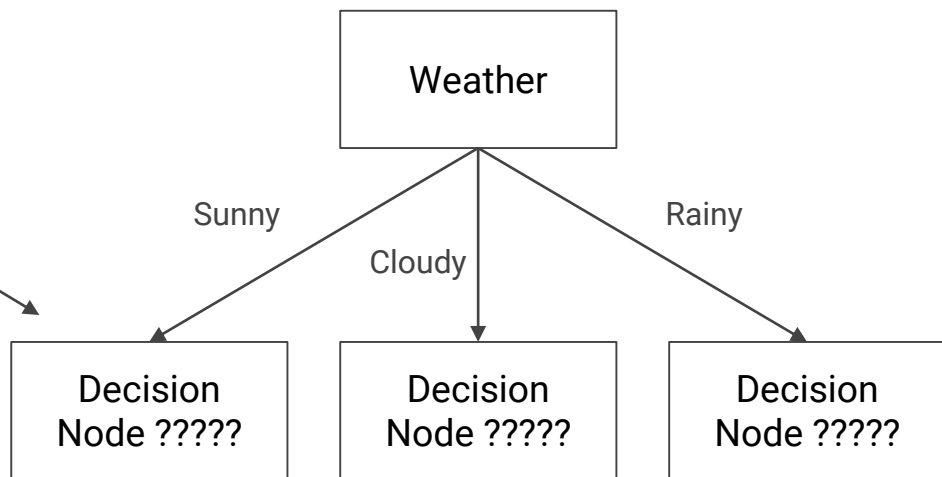
Gini Impurity

Gini Impurity (Weather) = 0.396 & Gini Impurity (Temperature) = 0.44

- Gini Impurity of (Weather) < Gini Impurity (Temperature)
- So “Weather” is preferred over “Temperature” as root
- Do the same for ALL of the features (and you will find that Impurity of Weather is the lowest)
- So we choose “Weather” as the root.

Gini Impurity

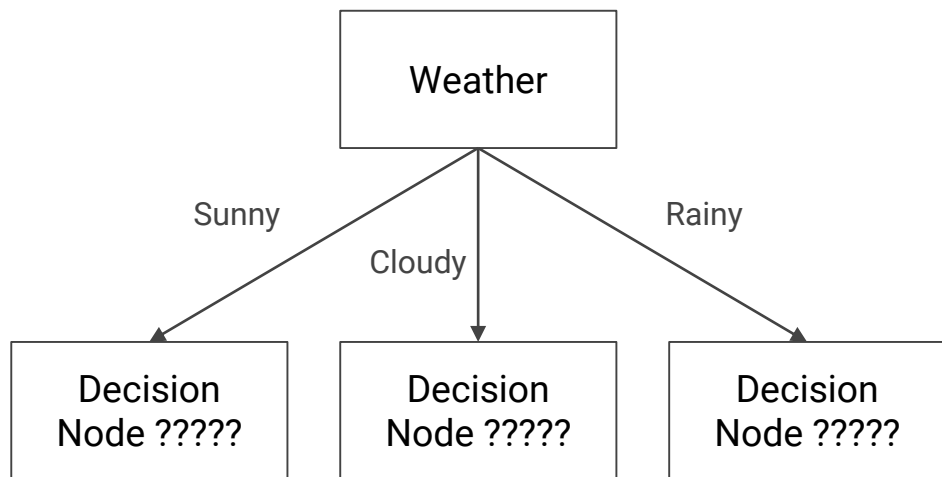
Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2					
3	Sunny	Mild	Normal	Strong	Yes
4					
5					
6					
7					
8	Sunny	Hot	High	Strong	No
9					
10					



To expand the branch under (Weather = "Sunny") calculate Gini Impurity with these 3 data points

Gini Impurity

Day	Weather	Temperature	Humidity	Wind	Play?
1					
2	Cloudy	Hot	High	Weak	Yes
3					
4	Cloudy	Mild	High	Strong	Yes
5					
6					
7					
8					
9	Cloudy	Hot	Normal	Weak	Yes
10					

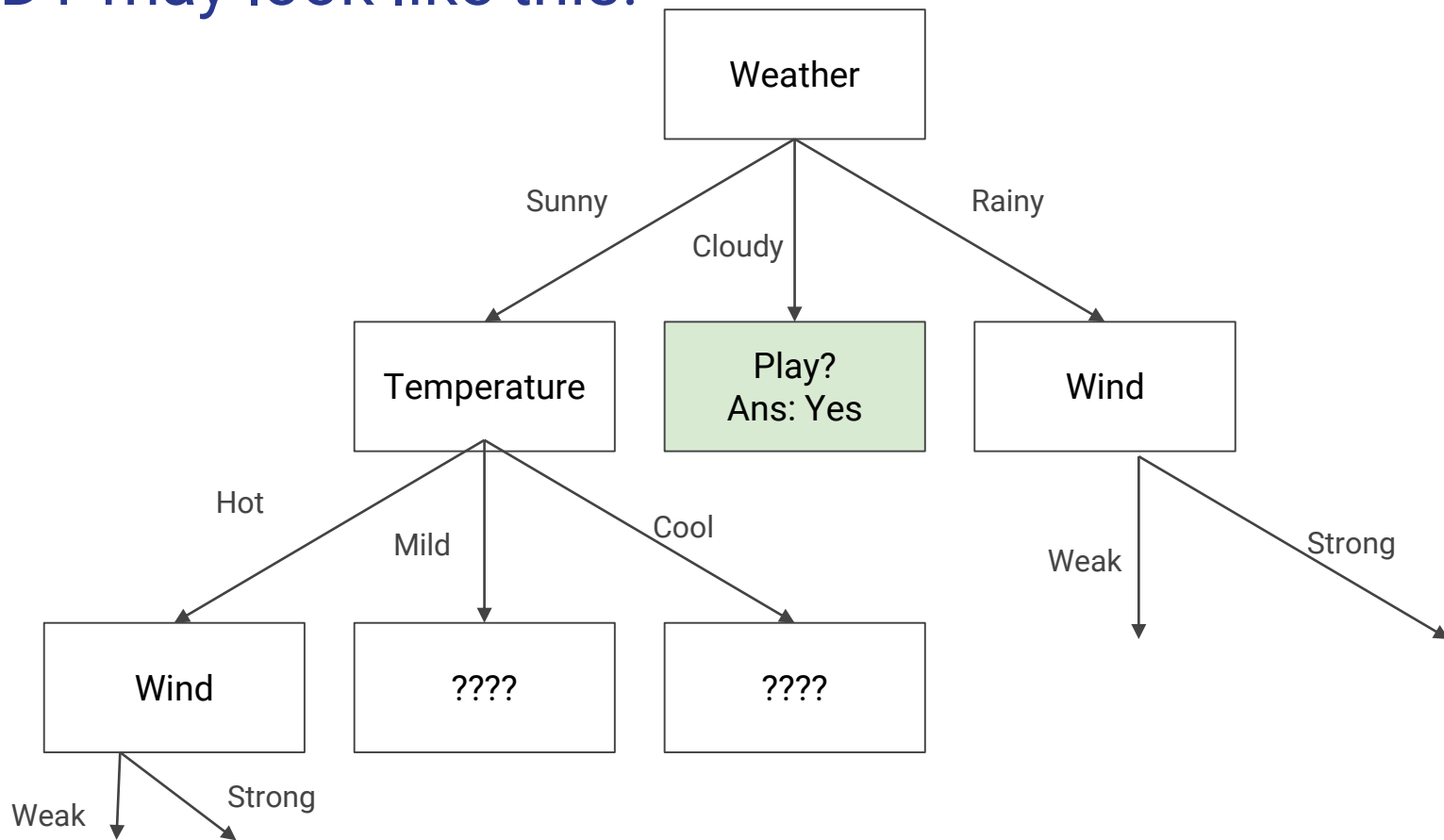


Did you notice that,

Gini Impurity of (Weather = Cloudy) is zero?

What is the meaning of that?

The DT may look like this!



Again, Decision Tree

- A **non-parametric supervised learning algorithm** for classification and regression tasks
- Decision trees typically **make binary splits**, meaning each node divides the data into two subsets based on a single feature or condition.
- Decision trees often assume that the features used for splitting nodes are independent. In practice, feature independence may not hold, but decision trees can **still perform well if features are correlated**.
- Decision trees are **prone to overfitting** when they capture noise in the data. **Pruning** and setting appropriate stopping criteria are used to address this assumption. **[MOST IMPORTANT]**
- Small datasets may lead to overfitting, and large datasets may result in overly complex trees. The sample size and tree depth should be balanced.
- Highly interpretable.

Pruning

- Pruning is another method that can help us avoid overfitting.
- It helps in improving the performance of the Decision tree by cutting the nodes or sub-nodes which are not significant.
- Additionally, it removes the branches which have very low importance.



What do we call a place
with lots and lots of trees?

Random Forest

Primary Issue with Decision Tree:

- One tree captures only one aspect of the dataset and depends on it.
- This results in overfitting (very good with train data) and less generalization ability of the model.
- Performs poorly with test data.

Random Forest:

- Take simplicity of the decision tree + add multiple aspect of the dataset

Random Forest

Step 1: Create a bootstrapped dataset.

- This new dataset may have **repetitive samples** from the original dataset.
- **Randomly selected** from original dataset
- Size same as the original dataset.

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Sunny	Mild	Normal	Strong	Yes
4	Cloudy	Mild	High	Strong	Yes



Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
4	Cloudy	Mild	High	Strong	Yes
3	Sunny	Mild	Normal	Strong	Yes
3	Sunny	Mild	Normal	Strong	Yes

Random Forest

Step 2: Create a decision tree using the bootstrapped dataset

- You are allowed to take only a subset of total features at each step.
(could be 2, could be 3)
- For example: We will randomly choose 2 features at each step.
 - First step: Between Temperature and Wind
 - Second step: Between Weather and Humidity

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Sunny	Mild	Normal	Strong	Yes
4	Cloudy	Mild	High	Strong	Yes

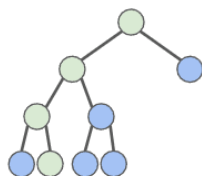


Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
4	Cloudy	Mild	High	Strong	Yes
3	Sunny	Mild	Normal	Strong	Yes
3	Sunny	Mild	Normal	Strong	Yes

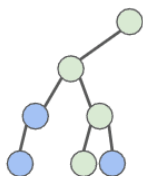
Random Forest

Step 3: Go back to Step 1 and repeat it at least 100 times and create 100 decision trees.

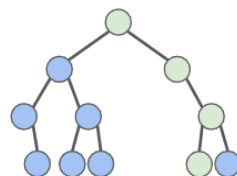
- Trees will have various shapes with various depth.
- Each tree will capture various aspect of the same dataset.



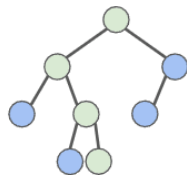
Tree 1: Cat



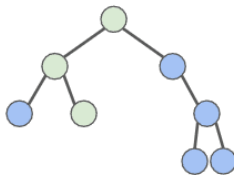
Tree 2: Dog



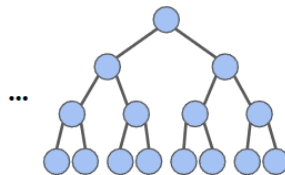
Tree 3: Cat



Tree 4: Cat



Tree 5: Cat



Tree n

Random Forest

Million Dollar Question: We have 100s of trees. How do we predict the class of a data?

VOTING Obviously!!!

Random Forest

Pass the new data through all the decision trees inside the forest.

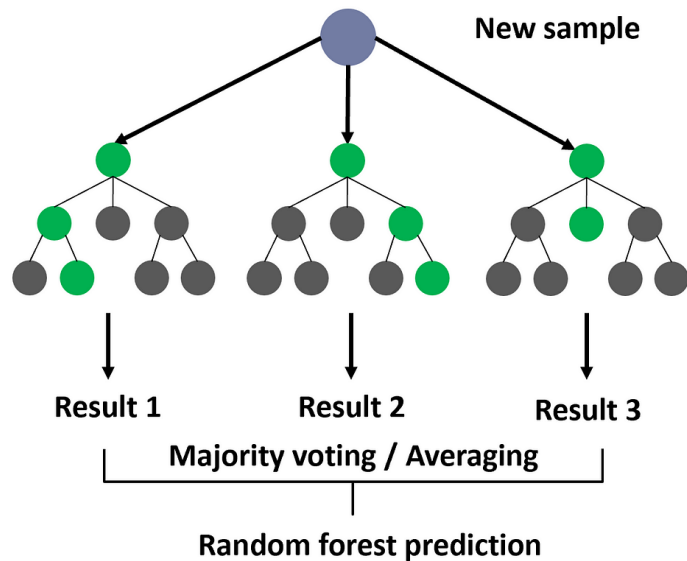
- Majority vote wins!

8	Sunny	Hot	High	Strong
---	-------	-----	------	--------

Decision Trees	
Play = Yes	Play = No
22	78

So predict as NO!

Random Forest



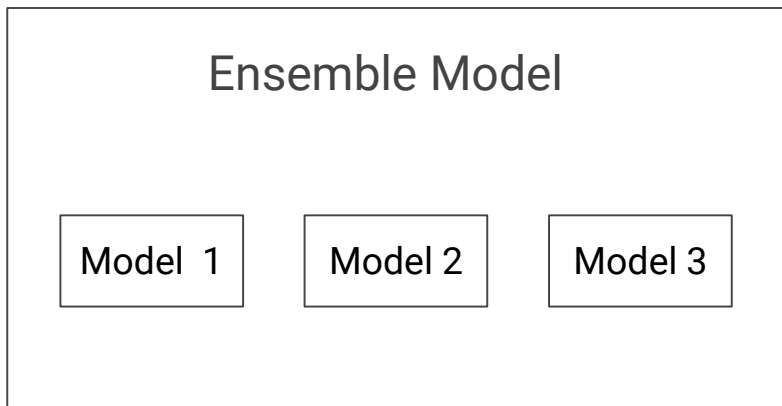
We play with/tune the hyperparameters to get the BEST
Random Forest



Ensemble Techniques

Ensemble Techniques

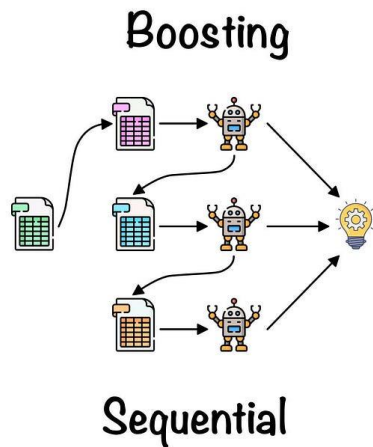
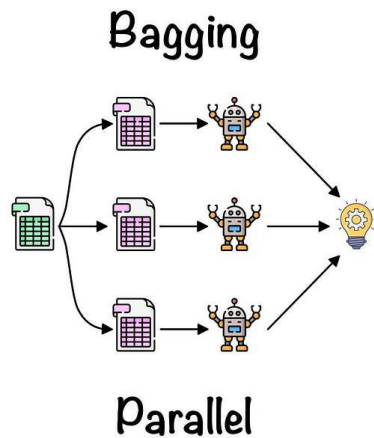
- Ensemble techniques in machine learning function much like seeking advice from multiple sources before making a significant decision
- Ensemble learning is a machine learning technique that enhances accuracy and resilience in forecasting by merging predictions from multiple models.
- It aims to mitigate errors or biases that may exist in individual models by leveraging the collective intelligence of the ensemble.



Ensemble Techniques (Only Trees)

Two Ensemble Learning Techniques:

1. Bagging
2. Boosting

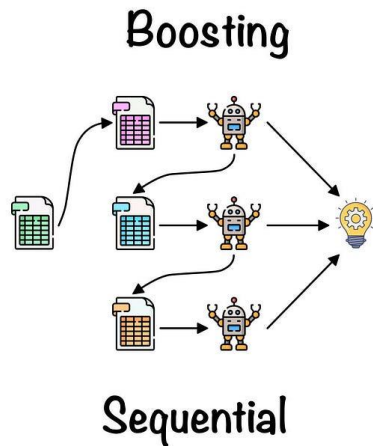
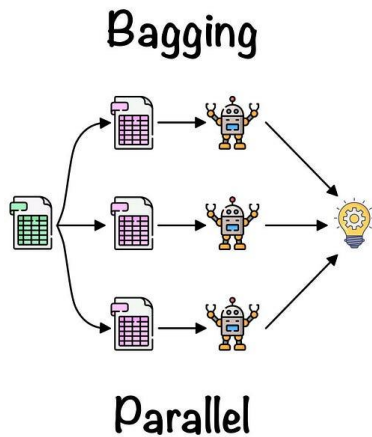


Random Forest is an ensemble learning technique! Guess which one?

Ensemble Techniques (Only Trees)

Two Ensemble Learning Techniques:

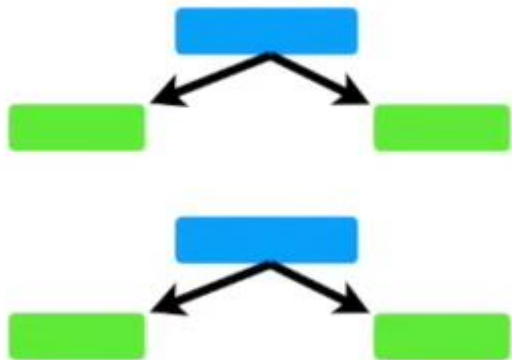
1. Bagging (Random Forest)
2. Boosting (Gradient Boost, AdaBoost, XGBoost)



Random Forest is an ensemble learning technique! Guess which one?

Adaboost

- In Random Forest, we create multiple full sized, randomly shaped trees.
- In Adaboost, it's just a node with two leaves.



- These trees are called Stumps.
- Adaboost -> Forest of stumps.
- Each stumps is also a “weak learner.”

Adaboost

- Adaboost combines a lot of weak learners to make classification.
- Stump outputs are “weighted.” Some stump has more influence than the others in the output. Unlike RF where each tree had equal rights.
- Each stump is made by taking the error of the previous stump.

Adaboost

Step 1: Assigning Weights

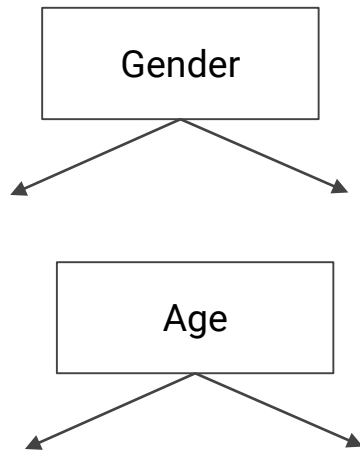
Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

$$w(x_i, y_i) = \frac{1}{N}, \quad i = 1, 2, \dots, n$$

Adaboost

Step 2: Choose weak learner using Gini Index for each feature.

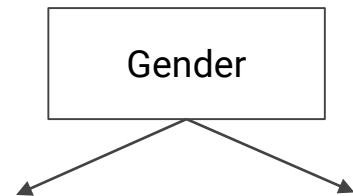
Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5



Adaboost

Step 3: Calculate the Influence/weight

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5



$$\frac{1}{2} \log \frac{1 - \text{Total Error}}{\text{Total Error}}$$

Adaboost

Step 3: Calculate the Influence/weight

- Error always between 0 and 1. 0 indicates perfect stump 1 indicates horrible.
- Consider only one out of five data is misclassified. In that case:

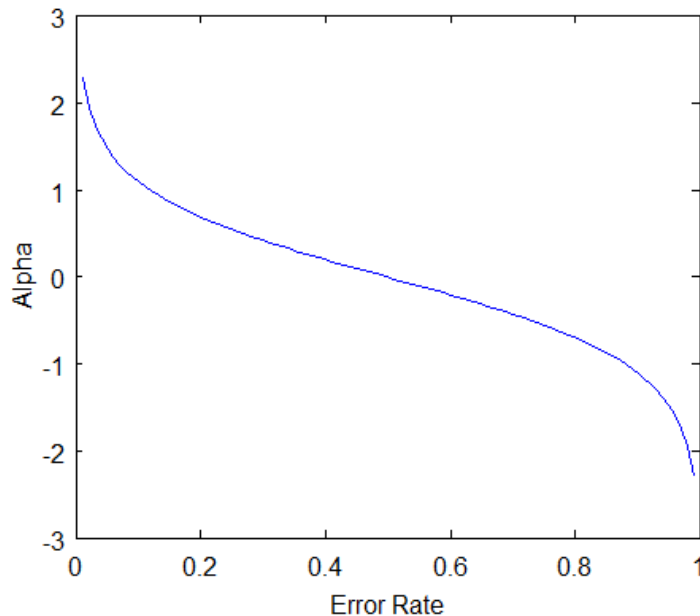
$$\text{Performance of the stump} = \frac{1}{2} \log_e \left(\frac{1 - \text{Total Error}}{\text{Total Error}} \right)$$

$$\alpha = \frac{1}{2} \log_e \left(\frac{1 - \frac{1}{5}}{\frac{1}{5}} \right)$$

$$\alpha = \frac{1}{2} \log_e \left(\frac{0.8}{0.2} \right)$$

$$\alpha = \frac{1}{2} \log_e(4) = \frac{1}{2} * (1.38)$$

$$\alpha = 0.69$$



Adaboost

Step 3: Update weights

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

*New sample weight = old weight * $e^{\pm \text{Amount of say } (\alpha)}$*

Adaboost

Step 3: Update weights

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

$$\text{New sample weight} = \text{old weight} * e^{\pm \text{Amount of say } (\alpha)}$$

- The amount of, say (alpha) will be negative when the sample is correctly classified.
- The amount of, say (alpha) will be positive when the sample is miss-classified.

Adaboost

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

Step 3: Update weights

$$\text{New sample weight} = \text{old weight} * e^{\pm \text{Amount of say } (\alpha)}$$

- There are four correctly classified samples and 1 wrong. Here, the sample weight of that datapoint is 1/5, and the amount of say/performance of the stump of Gender is 0.69.
- New weights for correctly classified samples are:

$$\text{New sample weight} = \frac{1}{5} * \exp(-0.69)$$

$$\text{New sample weight} = 0.2 * 0.502 = 0.1004$$

Adaboost

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

Step 3: Update weights

$$\text{New sample weight} = \text{old weight} * e^{\pm \text{Amount of say } (\alpha)}$$

- For wrongly classified samples, the updated weights will be:

$$\text{New sample weight} = \frac{1}{5} * \exp(0.69)$$

$$\text{New sample weight} = 0.2 * 1.994 = 0.3988$$

Adaboost

Step 3: Update weights

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

Row No.	Gender	Age	Income	Illness	Sample Weights	New Sample Weights
1	Male	41	40000	Yes	1/5	0.1004
2	Male	54	30000	No	1/5	0.1004
3	Female	42	25000	No	1/5	0.1004
4	Female	40	60000	Yes	1/5	0.3988
5	Male	46	50000	Yes	1/5	0.1004



Thank You!