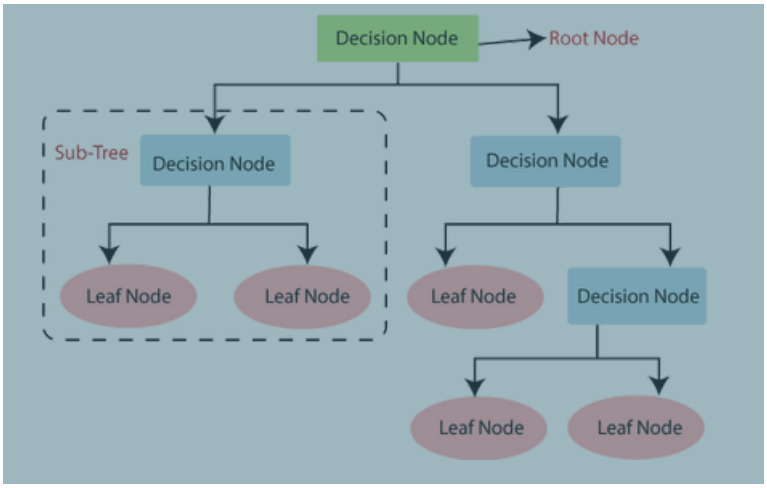
03

Classification: Decision Tree

Decision Tree

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems. It is a tree-structured classifier, where.

- Internal nodes represent the features of a dataset,
- Branches represent the decision rules,
- Each leaf node represents the outcome.



How to Construct a Decision Tree?

There are 2 popular tree building-algorithm out there: ID3 and CART (Classification and Regression Tree). The main difference between these two models is the cost function that they use. The Decision Tree algorithm intuition is as follows:

- For each attribute in the dataset, the Decision-Tree algorithm forms a node.
- For evaluating the task in hand, we start at the root node, and we work our way down the tree by following the corresponding node that meets our condition or decision.
- This process continues until a leaf node is reached. It contains the prediction or the outcome of the Decision Tree.

Attribute Selection

There are different attribute selection measures to identify the attribute which can be considered as the root node at each level. There are 2 popular attribute selection measures. They are as follows:

- Information gain in ID3 algorithm
- Gini index in CART algorithm

Entropy

Entropy, also called as Shannon Entropy is denoted by H(S) for a finite set S, is the measure of the amount of uncertainty or randomness in data.

$$H(S) = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}$$

Highest entropy is when there's no way of determining what the outcome. Consider a coin which has heads on both sides. Since we know beforehand that it'll always be heads, this event has no randomness, and its entropy is zero.

ID3 (Iterative Dichotomiser) Algorithm

The ID3 (Iterative Dichotomiser) Decision Tree algorithm uses entropy to calculate information gain.

- Entropy measures the impurity in the given dataset. In Physics and Mathematics, entropy is referred to as the randomness or uncertainty of a random variable X. In information theory, it refers to the impurity in a group of examples.
- Information gain is the decrease in entropy. Information gain computes the difference between entropy before split and average entropy after split of the dataset based on given attribute values.

Information Gain

Information Gain denoted by IG(S, A) for a set S is the effective change in entropy after deciding on a particular attribute A. It measures the relative change in entropy with respect to the independent variables.

$$IG(S,A) = H(S) - H(S,A) \qquad IG(S,A) = H(S) - \sum_{i=0}^{n} P(x) * H(x)$$

Where IG(S, A) is the information gain by applying feature A. H(S) is the Entropy of the entire set, while the second term calculates the Entropy after applying the feature A, where P(x) is the probability of event x.

Calculating Information Gain

In the example, we can see in total there are5 No's and 9 Yes's.

$$Entropy(S) = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}$$

$$Entropy(S) = -\left(\frac{9}{14}\right)\log_2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log_2\left(\frac{5}{14}\right)$$

$$= 0.940$$

Remember that the Entropy is 0 if all members belong to the same class (no uncertainty), and 1 when half of them belong to one class and other half belong to other class (high randomness/high uncertainty). Here it's 0.94 which means the distribution is fairly random. Now the next step is to choose the attribute that gives us highest possible Information Gain which we'll choose as the root node.

Gini Impurity

Gini impurity measures the frequency at which any element of the dataset will be mislabelled when it is randomly labeled.

$$GiniIndex = 1 - \sum_{j} p_j^2$$

for j = 1 to number of classes

The minimum value of the Gini Index is 0. This happens when the node is pure, this means that all the contained elements in the node are of one unique class. Moreover, it gets the maximum value when the probability of the two classes is the same.

$$Gini_{min} = 1 - (1^2) = 0$$

$$Gini_{max} = 1 - (0.5^2 + 0.5^2) = 0.5$$

Gini Index (for Outlook Feature)

Let's understand the CART algorithm with the help of golf playing decision dataset. Note that number of classes is 2 for 'No' = not playing golf and 'Yes' = playing golf. Outlook consists of 3 values (Sunny, Overcast, Rain)

Outlook	Yes	No	Number of Instance
Sunny	2	3	5
Overcast	4	0	4
Rain	3	2	5

Gini (Outlook = Sunny) =
1 – (2/5)2 – (3/5)2 = 1 – 0.16 – 0.36 = 0.48
Gini (Outlook = Overcast) =
1 – (4/4)2 – (0/4)2 = 0
Gini (Outlook = Rain) =
1 – (3/5)2– (2/5)2 = 1 – 0.36 – 0.16 = 0.48

Then, weighted sum of Gini indexes for outlook feature:
Gini (Outlook) =
(5/14) x 0.48 + (4/14) x 0 + (5/14) x 0.48 =
0.171 + 0 + 0.171 = 0.342

Gini Index (for Humidity Feature)

Humidity is a binary classfeatureand has 2 values (High and Normal)

Humidity	Yes	No	Number of Instance
High	3	4	7
Normal	6	1	7

Gini (Humidity = High) =
1 – (3/7)2 – (4/7)2 = 1 – 0.183 – 0.326 = 0.489
Gini (Humidity = Normal) =
1 – (6/7)2 – (1/7)2 = 1 – 0.734 – 0.02 = 0.244

Weighted sum for humidity feature will be calculated next
Gini (Humidity) = (7/14) x 0.489 + (7/14) x 0.244 = 0.367

Gini Index (for Temperature Feature)

Temperature	Yes	No	Number of Instance
Hot	2	2	4
Cool	3	1	4
Mild	4	2	6

Gini (Temp = Hot) = 1 – (2/4)2– (2/4)2 = 0.5
Gini (Temp = Cool) = 1 – (3/4)2 – (1/4)2= 1 – 0.5625 – 0.0625 = 0.375
Gini (Temp = Mild) = 1 – (4/6)2– (2/6)2 = 1 – 0.444 – 0.111 = 0.445

We'll calculate weighted sum of gini index for temperature feature
Gini (Temp) = (4/14) x 0.5 + (4/14) x 0.375 + (6/14) x 0.445 = 0.142 + 0.107 + 0.190 = 0.439

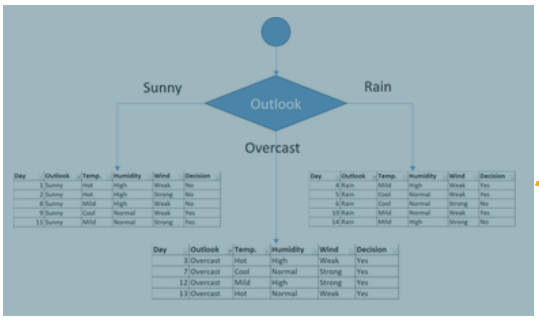
Gini Index (for Wind Feature)

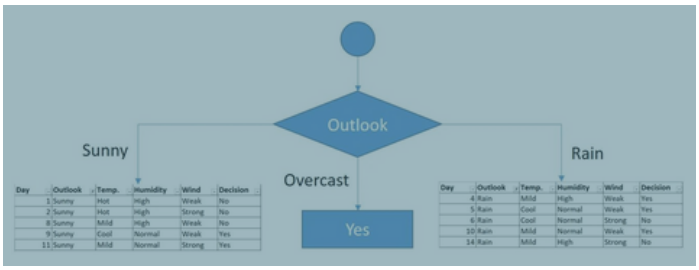
Wind	Yes	No	Number of Instance
Weak	6	2	8
Strong	3	3	6

Gini (Wind = Weak) =1 – (6/8)2 – (2/8)2 = 1 – 0.5625 –0.062=0.375
Gini (Wind = Strong) = 1 – (3/6)2 – (3/6)2 = 1–0.25–0.25= 0.5
Gini (Wind) = (8/14) x 0.375 + (6/14) x 0.5 = 0.428

Deciding Root Node

Gini (Outlook) = 0.342
Gini (Temperature) = 0.439
Gini (Humidity) = 0.367
Gini (Wind) = 0.428
We see Outlook has the lowest index and put it at the top of the tree.



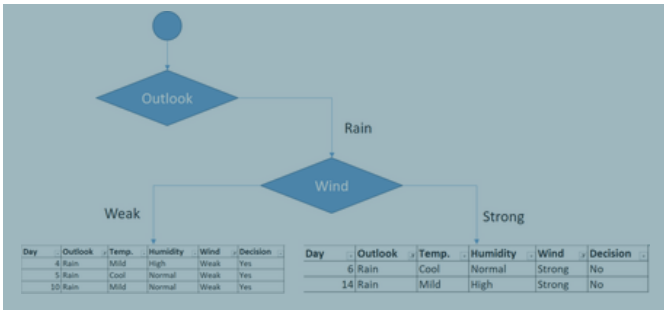


Decision for Rain Sub Dataset

Below is the dataset for Sunny Outlook

Day	Outlook	Temp.	Humidity	Wind	Decision
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
10	Rain	Mild	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

With the same calculation as before you'll get
Gini (Rain, Temperature) = 4.66
Gini (Rain, Humidity) = 4.66
Gini (Rain, Wind) = 0



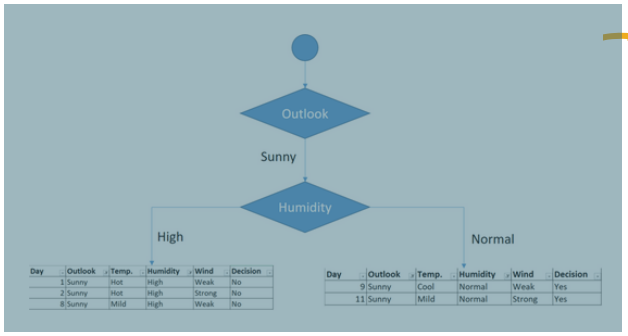
As seen, the decision is always yes when the wind is weak. On the other hand, the decision is always no if the wind is strong.

Decision for Sunny Sub Dataset

This one is for Sunny Outlook

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes

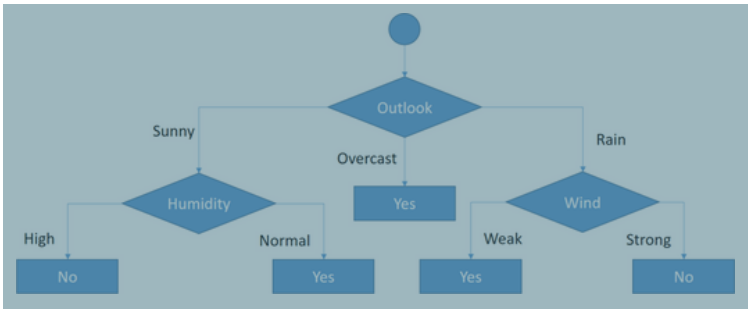
With the same calculation as before you'll get
Gini (Sunny, Temperature) = 0.2
Gini (Sunny, Humidity) = 0
Gini (Sunny, Wind) = 0.466



We choose humidity and the decision tree of the sunny outlook will become like this. As seen, the decision is always no for high humidity and yes for normal humidity. So, this branch is over.

Final Form of Decision Tree built by CART Algorithm

You might realize that we've created exactly the same tree as in the ID3 example. This does not mean that ID3 and CART algorithms always produce the same trees. This simple example fortunately generates the same tree. But it's not always the case.



Some of Decision Advantages

- It can capture nonlinear relationships: They can be used to classify non-linearly separable data.
- Easy to understand, interpret, visualize.
- Compared to other algorithms decision trees requires less effort for data preparation during pre-processing.
- A decision tree does not require normalization of data.
- A decision tree does not require caling of data as well.
- Missing values in the data also do NOT affect the process of building a decision tree to any consider able extent.
- A Decision tree modelis very intuitive and easy to explain to technical teams as well as stakeholders.

Some of Decision Tree Disadvantages

- A small change in the data can cause a large change in the structure of the decision tree causing **instability**.
- Adding a new data point can lead to re-generation of the overall tree and all nodes need to be recalculated and recreated.
- It can't be used in big data: If the size of data is too big, then one single tree may grow a lot of nodes which might result in complexity and leads to **overfitting**.

Picture Source

- https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm
- pexels.com