Building a Decision Tree using ID3 Algorithm (Step by Step example) :

Example to understand decision tree creation using ID3 algorithm :

Consider a piece of data collected over the course of 14 days where the features are **Outlook, Temperature, Humidity, Wind** and the **outcome variable is whether Golf was played on the day**. Now, our job is to build a **predictive model** which takes in above 4 parameters and predicts whether Golf will be played on the day. We'll build a decision tree to do that using **ID3 algorithm.**

Day	Outlook	Temperature	Humidity	Wind	Play Golf
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

ID3 Algorithm will perform following tasks recursively

1. Create root node for the tree
2. If all examples are positive, return leaf node 'positive'
3. Else if all examples are negative, return leaf node 'negative'
4. Calculate the entropy of current state H(S)
5. For each attribute, calculate the entropy with respect to the attribute 'x' denoted by H(S, x)
6. Select the attribute which has maximum value of IG(S, x)
7. Remove the attribute that offers highest IG from the set of attributes

8. Repeat until we run out of all attributes, or the decision tree has all leaf nodes.

Now we'll go ahead and grow the decision tree. The initial step is to calculate H(S), the Entropy of the current state. In the above example, we can see in total there are 5 No's and 9 Yes's.

To build a decision tree, we need to calculate two types of entropy using frequency tables as follows:

a) Entropy using the frequency table of one attribute:

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)
= 0.94

$$Entropy(S) = -\left(\frac{9}{14}\right)\log_2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log_2\left(\frac{5}{14}\right)$$
$$= 0.940$$

Remember that the **Entropy is 0 if all members belong to the same class, and 1 when half of them belong to one class and other half belong to other class that is perfect randomness**.
Here it's 0.94 which means the distribution is fairly random.

b) Entropy using the frequency table of two attributes:

Now the next step is to choose the attribute that gives us highest possible Information Gain which we'll choose as the root node. So check for each possible attribute.

$$E(T, X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)***E**(3,2) + **P**(Overcast)***E**(4,0) + **P**(Rainy)***E**(2,3)

= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

Similarly calculate entropy for other attributes i.e. Temp, Humidity, Windy

E(PlayGolf, Temp) =?

E(PlayGolf, Humidity) =?

E(PlayGolf, Windy) =?

c) Calculate Information Gain:

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

= 0.940 – 0.693 = 0.247

Similarly compute **Information gain for temp, humidity and Windy**. The computed results are shown below.

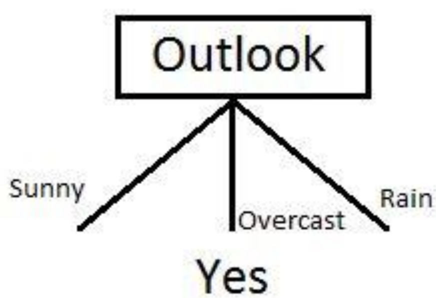
$$IG(S, Outlook) = 0.246$$
$$IG(S, Temperature) = 0.029$$
$$IG(S, Humidity) = 0.151$$
$$IG(S, Wind) = 0.048 \text{ (Previous example)}$$

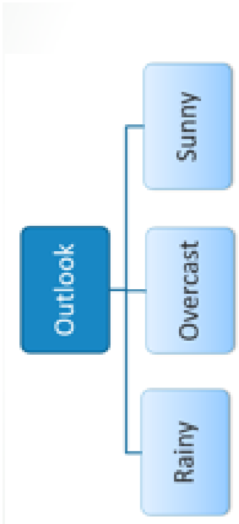
We can clearly see that IG(S, Outlook) has the highest information **gain of 0.246,** hence **we chose Outlook attribute as the root node**. At this point, the decision tree looks like.

d) Choose attribute with the largest information gain as the decision node, divide the dataset by its branches and repeat the same process on every branch.

Here we observe that whenever the outlook is **Overcast, Play Golf is always 'Yes'**, it's no coincidence by any chance, the simple tree resulted because of **the highest information gain is given by the attribute Outlook**.

★		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

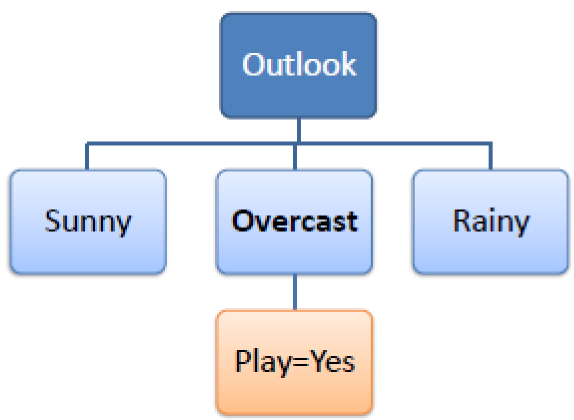




Outlook	Temp	Humidity	Windy	Play Golf
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Sunny	Mild	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Overcast	Cool	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

A branch with entropy of 0 is a leaf node.

Temp	Humidity	Windy	Play Golf
Hot	High	FALSE	Yes
Cool	Normal	TRUE	Yes
Mild	High	TRUE	Yes
Hot	Normal	FALSE	Yes



Now how do we proceed from this point? We can simply apply **recursion**, you might want to look at the algorithm steps described earlier.Now that we've used Outlook, we've got three of them remaining Humidity, Temperature, and Wind. And, we had three possible values of Outlook: Sunny, Overcast, Rain. Where the Overcast node already ended up having leaf node 'Yes', so we're left with two subtrees to compute: Sunny and Rain.

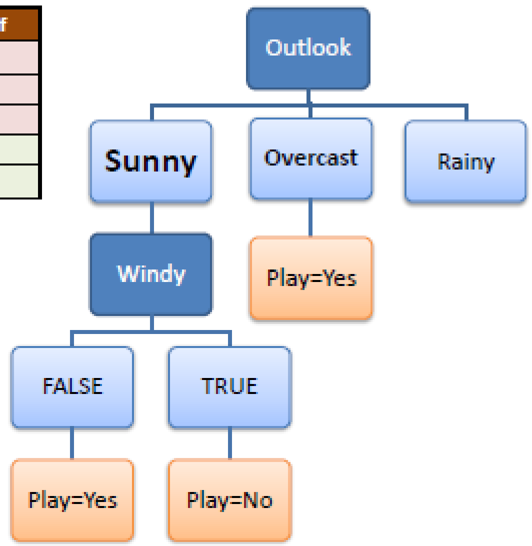
Next iterations:

So.e.g. compute entropy for sunny (as it is non-leaf node) and then Information gain with other attributes

$$H(S_{sunny}) = \left(\frac{3}{5}\right)\log_2\left(\frac{3}{5}\right) - \left(\frac{2}{5}\right)\log_2\left(\frac{2}{5}\right) = 0.96$$

A branch with entropy more than 0 needs further splitting.

Temp	Humidity	Windy	Play Golf
Mild	High	FALSE	Yes
Cool	Normal	FALSE	Yes
Mild	Normal	FALSE	Yes
Cool	Normal	TRUE	No
Mild	High	TRUE	No



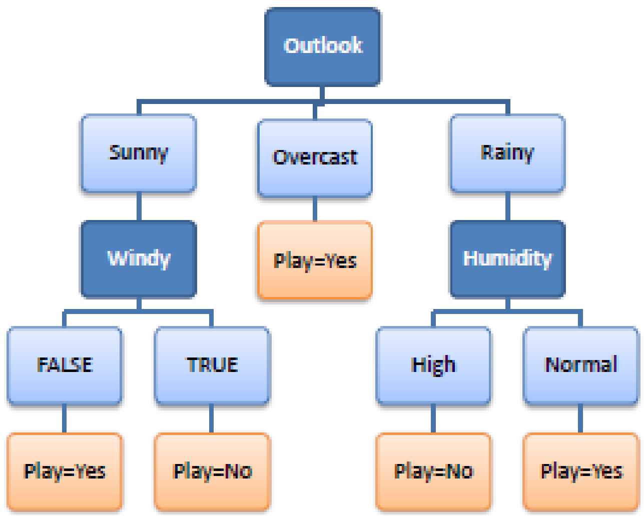
In the similar fashion, we compute the following values

$$IG(S_{sunny}, Humidity) = 0.96$$

$$IG(S_{sunny}, Temperature) = 0.57$$

$$IG(S_{sunny}, Wind) = 0.019$$

The final tree would look like this:



Decision Tree to Decision Rules

A decision tree can easily be transformed to a set of rules by mapping from the root node to the leaf nodes one by one.

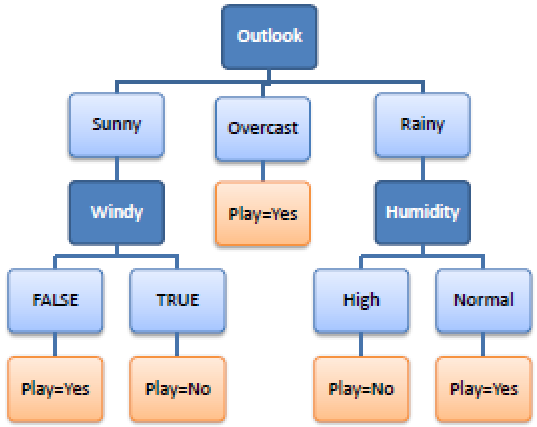
R$_1$: IF (Outlook=Sunny) AND (Windy=FALSE) THEN Play=Yes

R$_2$: IF (Outlook=Sunny) AND (Windy=TRUE) THEN Play=No

R$_3$: IF (Outlook=Overcast) THEN Play=Yes

R$_4$: IF (Outlook=Rainy) AND (Humidity=High) THEN Play=No

R$_5$: IF (Outlook=Rain) AND (Humidity=Normal) THEN Play=Yes



References:

1. Information Gain: https://en.wikipedia.org/wiki/Information_gain_in_decision_trees
2. Entropy: https://en.wikipedia.org/wiki/Entropy_(information_theory)
3. ID3: https://en.wikipedia.org/wiki/ID3_algorithm
4. Entropy YouTube video: https://www.youtube.com/watch?v=O__7lAqni7A
5. ID3 Example: https://www.cise.ufl.edu/~ddd/cap6635/Fall-97/Short-papers/2.htm