

Let us take an example of the last 10 days weather dataset with attributes outlook, temperature, wind, and humidity. The outcome variable will be playing cricket or not. We will use the ID3 algorithm to build the decision tree.

Day	Outlook	Temperature	Humidity	Wind	Play cricket
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Step1: The first step will be to create a root node.

Step2: If all results are yes, then the leaf node “yes” will be returned else the leaf node “no” will be returned.

Step3: Find out the Entropy of all observations and entropy with attribute “x” that is $E(S)$ and $E(S, x)$.

Step4: Find out the information gain and select the attribute with high information gain.

Step5: Repeat the above steps until all attributes are covered.

Calculation of Entropy:

Yes No

9 5

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

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

$$= 0.940$$

If entropy is zero, it means that all members belong to the same class and if entropy is one then it means that half of the tuples belong to one class and one of them belong to other class. 0.94 means fair distribution.

Find the information gain attribute which gives maximum information gain.

For Example “Wind”, it takes two values: Strong and Weak, therefore, $x = \{\text{Strong, Weak}\}$.

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

Find out $H(x)$, $P(x)$ for $x = \text{weak}$ and $x = \text{strong}$. $H(S)$ is already calculated above.

Weak= 8

Strong= 8

$$P(S_{weak}) = \frac{\text{Number of Weak}}{\text{Total}}$$

$$= \frac{8}{14}$$

$$P(S_{strong}) = \frac{\text{Number of Strong}}{\text{Total}}$$

$$= \frac{6}{14}$$

For “weak” wind, 6 of them say “Yes” to play cricket and 2 of them say “No”. So entropy will be:

$$Entropy(S_{weak}) = -\left(\frac{6}{8}\right) \log_2 \left(\frac{6}{8}\right) - \left(\frac{2}{8}\right) \log_2 \left(\frac{2}{8}\right)$$

$$= 0.811$$

For “strong” wind, 3 said “No” to play cricket and 3 said “Yes”.

$$\begin{aligned} \text{Entropy}(S_{\text{strong}}) &= -\left(\frac{3}{6}\right) \log_2 \left(\frac{3}{6}\right) - \left(\frac{3}{6}\right) \log_2 \left(\frac{3}{6}\right) \\ &= 1.000 \end{aligned}$$

This shows perfect randomness as half items belong to one class and the remaining half belong to others.

Calculate the information gain,

$$\begin{aligned} IG(S, \text{Wind}) &= H(S) - \sum_{i=0}^n P(x) * H(x) \\ IG(S, \text{Wind}) &= H(S) - P(S_{\text{weak}}) * H(S_{\text{weak}}) - P(S_{\text{strong}}) * H(S_{\text{strong}}) \\ &= 0.940 - \left(\frac{8}{14}\right) (0.811) - \left(\frac{6}{14}\right) (1.00) \\ &= 0.048 \end{aligned}$$

Similarly the information gain for other attributes is:

$$IG(S, \text{Outlook}) = 0.246$$

$$IG(S, \text{Temperature}) = 0.029$$

$$IG(S, \text{Humidity}) = 0.151$$

The attribute outlook has the **highest information gain** of 0.246, thus it is chosen as root. Overcast has 3 values: Sunny, Overcast and Rain. Overcast with play cricket is always “Yes”. So it ends up with a leaf node, “yes”. For the other values “Sunny” and “Rain”.

Table for Outlook as “Sunny” will be:

Temperature	Humidity	Wind	Golf
Hot	High	Weak	No
Hot	High	Strong	No
Mild	High	Weak	No
Cool	Normal	Weak	Yes
Mild	Normal	Strong	Yes

Entropy for “Outlook” “Sunny” is:

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

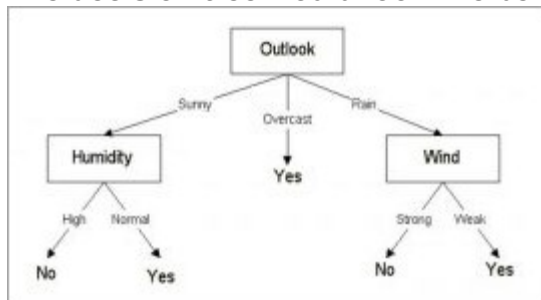
Information gain for attributes with respect to Sunny is:

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

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

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

The information gain for humidity is highest, therefore it is chosen as the next node. Similarly, Entropy is calculated for Rain. **Wind gives the highest information gain. The decision tree would look like below:**



Example 2

AGE	Income	STUDENT	CREDIT_RATING	BUYS_COMPUTER
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
	mediu			
>40	m	no	fair	yes
>40	low	yes	fair	yes
				.
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
	mediu			
<=30	m	no	fair	no
<=30	low	yes	fair	yes
	mediu			
>40	m	yes	fair	yes
	mediu			
<=30	m	yes	excellent	yes
	mediu			
31...40	m	no	excellent	yes
31...40	high	yes	fair	yes
	mediu			
>40	m	no	excellent	no

To calculate Entropy and Information Gain, we are computing the value of Probability for each of these 2 classes.

»Positive: For 'buys_computer=yes' probability will come out to be :

»Negative: For 'buys_computer=no' probability comes out to be :

Entropy in D: We now put calculate the Entropy by putting probability values in the formula stated above.

$$Info(D) = I(9,5) = - \frac{9}{14} \log_2\left(\frac{9}{14}\right) - \frac{5}{14} \log_2\left(\frac{5}{14}\right) = 0.940$$

We have already classified the values of Entropy, which are:

Entropy =0: Data is completely homogenous (pure)

Entropy =1: Data is divided into 50- 50 % (impure)

Our value of Entropy is **0.940**, which means our set is almost *impure*.

Let's delve deep, to find out the suitable attribute and calculate the Information Gain.

	Ye s No	
Age		
<=30	2	3
31...		
40	4	0
>40	3	2

What is information gain if we split on "Age"?

This data represents how many people falling into a specific age bracket, buy and do not buy the product.

For example, for people with Age 30 or less, 2 people buy (Yes) and 3 people do not buy (No) the product, the Info (D) is calculated for these 3 categories of people, that is represented in the last column.

The Info (D) for the age attribute is computed by the total of these 3 ranges of age values. Now, the question is what is the 'information gain' if we split on 'Age' attribute.

$$Info_{age}(D) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.694$$

The difference of the total Information value (0.940) and the information computed for age attribute (0.694) gives the 'information gain'.

Similarly, we calculate the 'information gain' for the rest of the attributes:

Information Gain (Age) =0.246

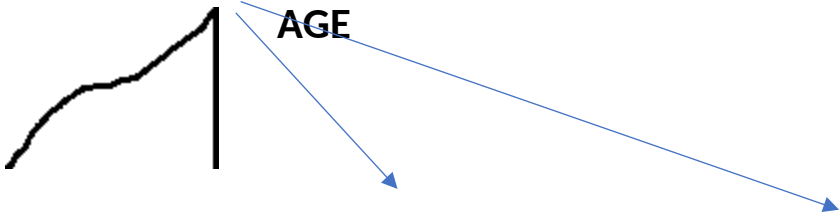
Information Gain (Income) =0.029

Information Gain (Student) = 0.151

Information Gain (credit_rating) =0.048

On comparing these values of gain for all the attributes, we find out that the 'information gain' for 'Age' is the highest. Thus, splitting at 'age' is a good decision.

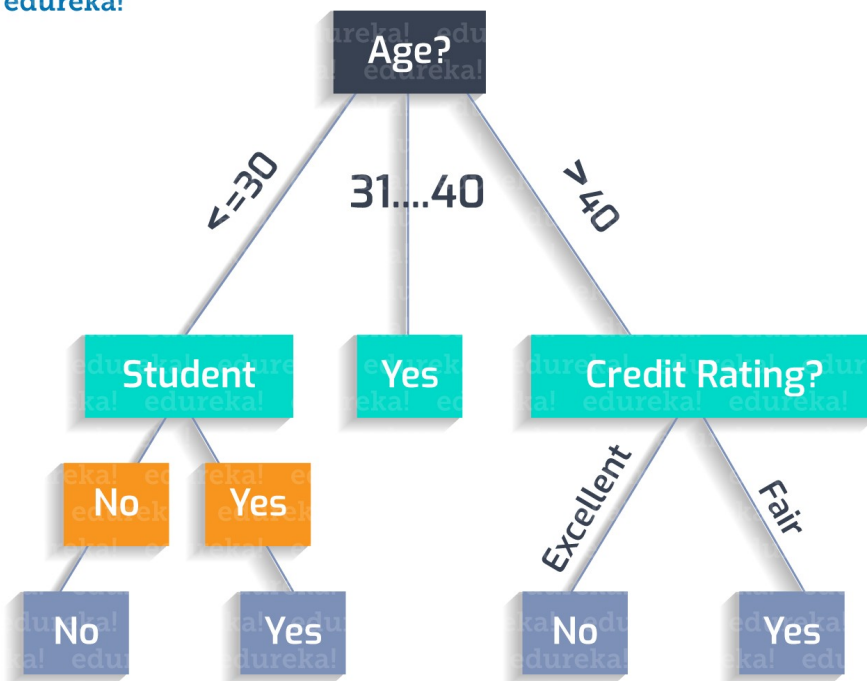
Similarly, at each split, we compare the information gain to find out whether that attribute should be chosen for split or not.



Income	STUDENT	CREDIT_RATING	BUY	Income	STUDENT	CREDIT_RATING	BUY	Age	Income	STUDENT	CREDIT_RATING	BUY
high	no	fair	no	high	no	fair	yes	Young	medium	no	fair	yes
high	no	excellent	no	low	yes	excellent	yes	Young	low	yes	fair	yes
medium	no	fair	no	medium	no	excellent	yes	Young	low	yes	excellent	no
low	yes	fair	yes	high	yes	fair	yes	Old	medium	yes	fair	yes
medium	yes	excellent	yes					Old	medium	no	excellent	no

Thus, the optimal tree created looks like :

edureka!



The classification rules for this tree can be jotted down as:

If a person's age is less than 30 and he is not a student, he will not buy the product.

$\text{Age}(<30) \wedge \text{student}(\text{no}) = \text{NO}$

If a person's age is less than 30 and he is a student, he will buy the product.

$\text{Age}(<30) \wedge \text{student}(\text{yes}) = \text{YES}$

If a person's age is between 31 and 40, he is most likely to buy.

$\text{Age}(31...40) = \text{YES}$

If a person's age is greater than 40 and has an excellent credit rating, he will not buy.

$\text{Age}(>40) \wedge \text{credit_rating}(\text{excellent}) = \text{NO}$

If a person's age is greater than 40, with a fair credit rating, he will probably buy.

$\text{Age}(>40) \wedge \text{credit_rating}(\text{fair}) = \text{Yes}$

Thus, we achieve the perfect Decision Tree!!

Example 3

Consider an example where we are building a decision tree to predict whether a loan given to a person would result in a write-off or not. Our entire population consists of 30 instances. 16 belong to the write-off class and the other 14 belong to the non-write-off class. We have two features, namely “Balance” that can take on two values -> “< 50K” or “>50K” and “Residence” that can take on three values -> “OWN”, “RENT” or “OTHER”. Build a DT , decide what attribute to split on first and what feature provides more information, or reduces more uncertainty about our target variable out of the two using the concepts of Entropy and Information Gain.