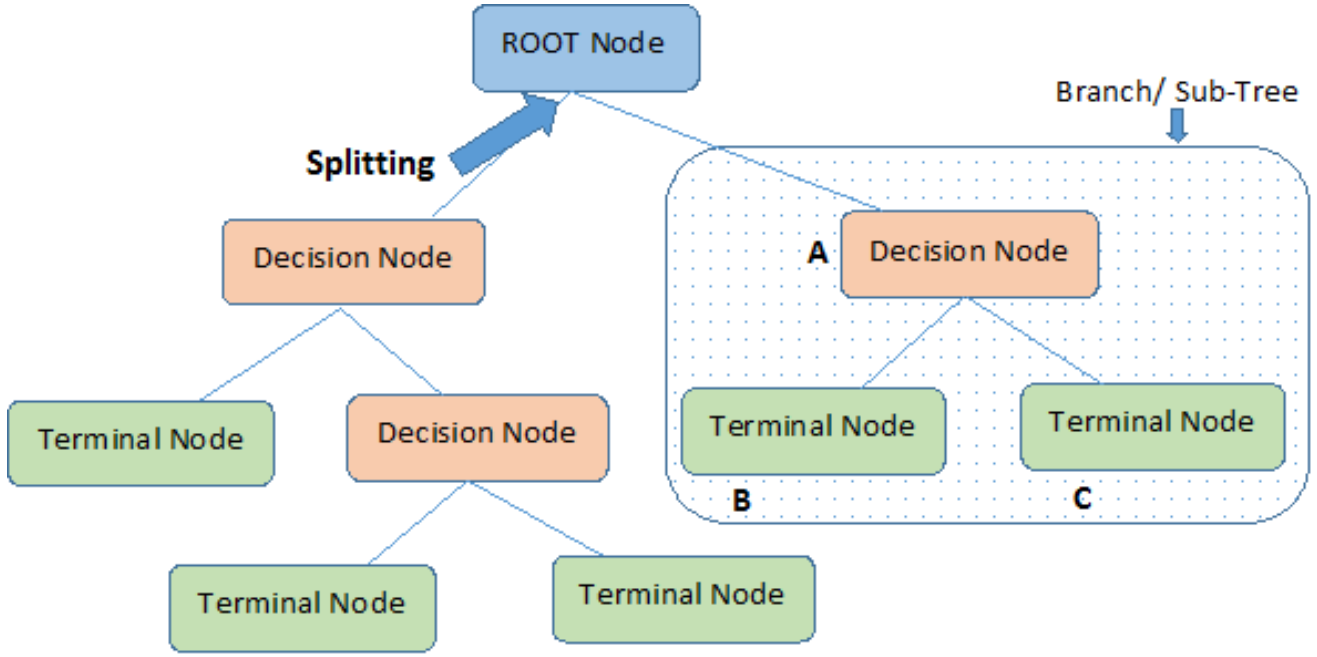
Decision Tree

Important Terminology related to Decision Trees

- **Root Node:** It represents entire population or sample and this further gets divided into two or more homogeneous sets.
- **Splitting:** It is a process of dividing a node into two or more sub-nodes.
- **Decision Node:** When a sub-node splits into further sub-nodes, then it is called decision node.
- **Leaf/ Terminal Node:** Nodes with no children (no further split) is called Leaf or Terminal node.
- **Pruning:** When we reduce the size of decision trees by removing nodes (opposite of Splitting), the process is called pruning.
- **Branch / Sub-Tree:** A sub section of decision tree is called branch or sub-tree.
- **Parent and Child Node:** A node, which is divided into sub-nodes is called parent node of sub-nodes where as sub-nodes are the child of parent node.



Note:- A is parent node of B and C.

Assumptions while creating Decision Tree

- Some of the assumptions we make while using Decision tree:
- At the beginning, the whole training set is considered as the **root.**
- Feature values are preferred to be categorical. If the values are continuous then they are discretized prior to building the model.
- Records are **distributed recursively** on the basis of attribute values.
- Order to placing attributes as root or internal node of the tree is done by using some statistical approach.

Algorithm used in decision trees:

ID3

Gini Index

Chi-Square

Reduction in Variance

we only talk about a few which are

- CART (Classification and Regression Trees) → uses ***Gini Index(Classification)*** as metric.
- ID3 (Iterative Dichotomiser 3) → uses ***Entropy function*** and ***Information gain*** as metrics.

Classification with using the ID3 algorithm.

- weather dataset(playing game Y or N based on weather condition).
- We have four X values (outlook,temp,humidity and windy) being categorical and one y value (play Y or N) also being categorical.
- so we need to learn the mapping (what machine learning always does) between X and y.
- This is a binary classification problem, lets build the tree using the ***ID3*** algorithm
- To create a tree, we need to have a root node first and we know that nodes are features/attributes(outlook,temp,humidity and windy),so which one do we need to pick first??

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	false	Don't Play
Sunny	Hot	High	true	Don't Play
Overcast	Hot	High	false	Play
Rain	Mild	High	false	Play
Rain	Cool	Normal	false	Play
Rain	Cool	Normal	true	Don't Play
Overcast	Cool	Normal	true	Play
Sunny	Mild	High	false	Don't Play
Sunny	Cool	Normal	false	Play
Rain	Mild	Normal	false	Play
Sunny	Mild	Normal	true	Play
Overcast	Mild	High	true	Play
Overcast	Hot	Normal	false	Play
Rain	Mild	High	true	Don't Play

so which one do we need to pick first??

Answer: determine the attribute that best classifies the training data; use this attribute at the root of the tree. Repeat this process at for each branch.

This means we are performing top-down, greedy search through the space of possible decision trees.

So how do we choose the best attribute?

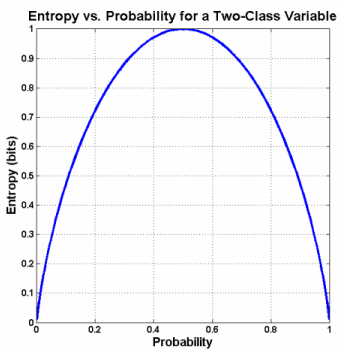
Answer: use the attribute with the highest ***information gain*** in ***ID3***

In order to define information gain precisely, we begin by defining a measure commonly used in information theory, called ***entropy*** *that characterizes the (im)purity of an arbitrary collection of examples."*

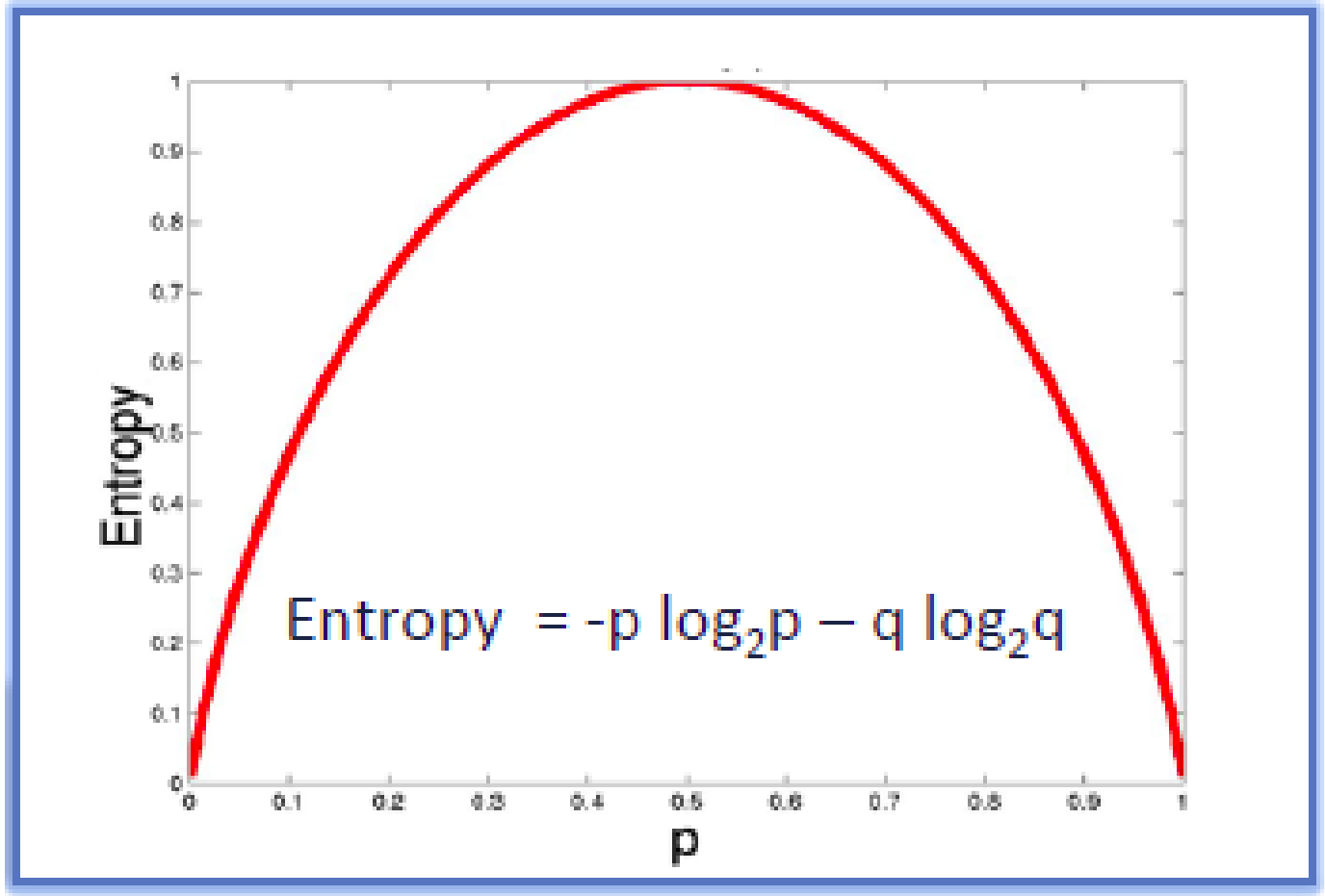
Entropy: Entropy is the measure of randomness of elements.
It is the measue of uncertainty in the given data set
It can also be called as measurement of purity
Mathematically it is defined as :

$$H = -\sum p(x) \log p(x)$$

- Entropy value lies in the range {0, $\log_2$ m} for m class scenario and {0,1} for 2 class scenario.
- Entropy is 0 when the sample is completely homogeneous.



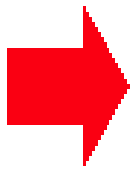
- ID3 algorithm uses entropy to calculate the homogeneity of a sample.
- If the sample is completely homogeneous the entropy is zero
- If the sample is an equally divided it has entropy of one.

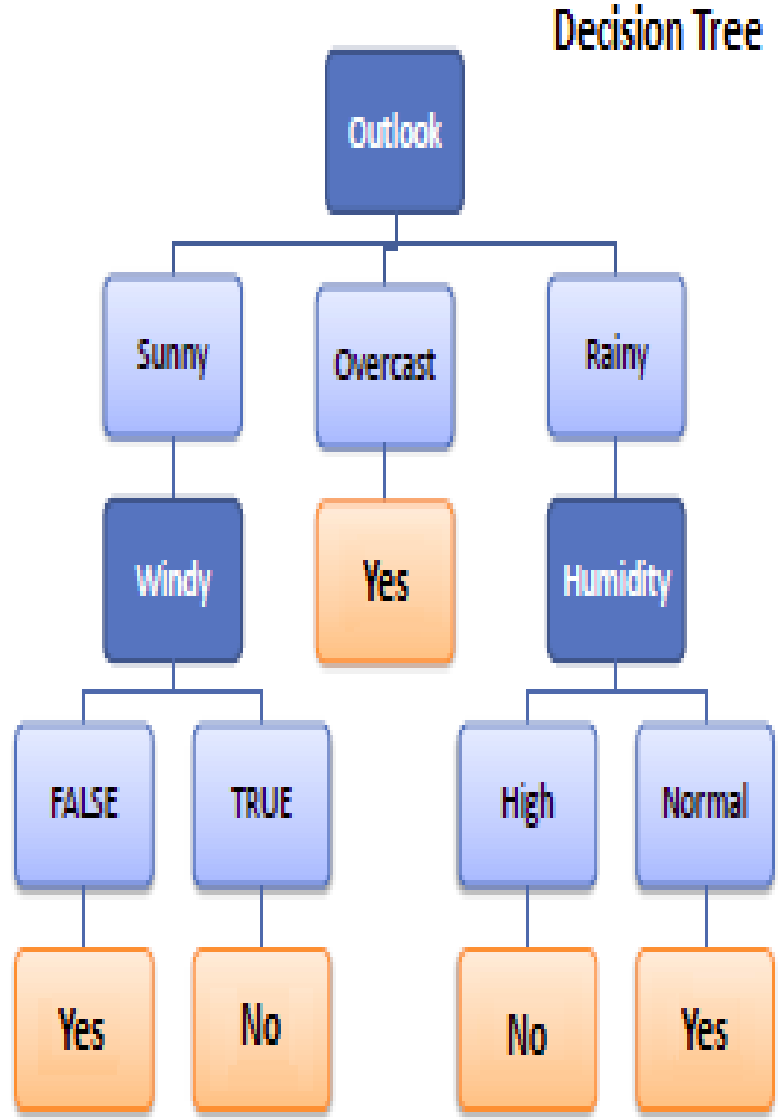


Entropy = $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

Predictors | Target

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No





Predictors				Target
Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

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

Steps to calculate information gain:

Step 1: calculate the overall entropy of the target.

- For every feature calculate the entropy and information gain

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:

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

Out of 14 instances, 9 are classified as yes, and 5 as no

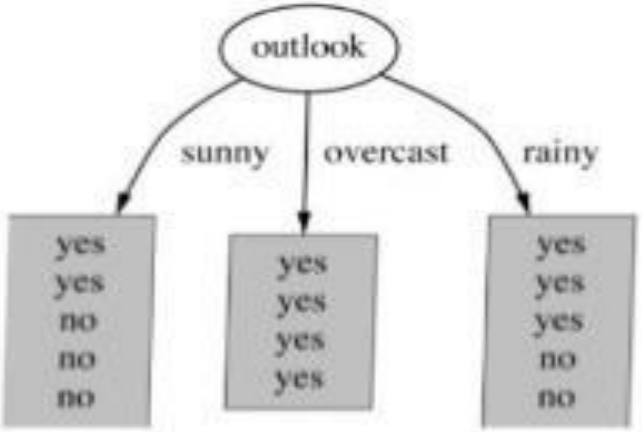
pyes = −(9/14)*log2(9/14) = 0.41

pno = −(5/14)*log2(5/14) = 0.53

H (S) = pyes + pno = 0.94

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

b) Entropy using the frequency table of two attributes:



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

E (Outlook=sunny) = $-\frac{2}{5}\log\left(\frac{2}{5}\right)-\frac{3}{5}\log\left(\frac{3}{5}\right)=0.971$

E (Outlook=overcast) = $-1\log(1)-0\log(0)=0$

E (Outlook=rainy) = $-\frac{3}{5}\log\left(\frac{3}{5}\right)-\frac{2}{5}\log\left(\frac{2}{5}\right)=0.971$

} H(S,Outlook)

Information Gain:

The information gain is based on the decrease in entropy after a data-set is split on an attribute.

Constructing a decision tree is all about finding attribute that returns the highest information gain (i.e., the most homogeneous branches).

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

E (Outlook=sunny) = $-\frac{2}{5}\log\left(\frac{2}{5}\right)-\frac{3}{5}\log\left(\frac{3}{5}\right)=0.971$

E (Outlook=overcast) = $-1\log(1)-0\log(0)=0$

E (Outlook=rainy) = $-\frac{3}{5}\log\left(\frac{3}{5}\right)-\frac{2}{5}\log\left(\frac{2}{5}\right)=0.971$

} H(S,Outlook)

Average Entropy information for Outlook

$$I(\text{Outlook}) = \frac{5}{14} * 0.971 + \frac{4}{14} * 0 + \frac{5}{14} * 0.971 = 0.693$$

Gain (Outlook) = E(S) – I (outlook) = 0.94-.693 = 0.247

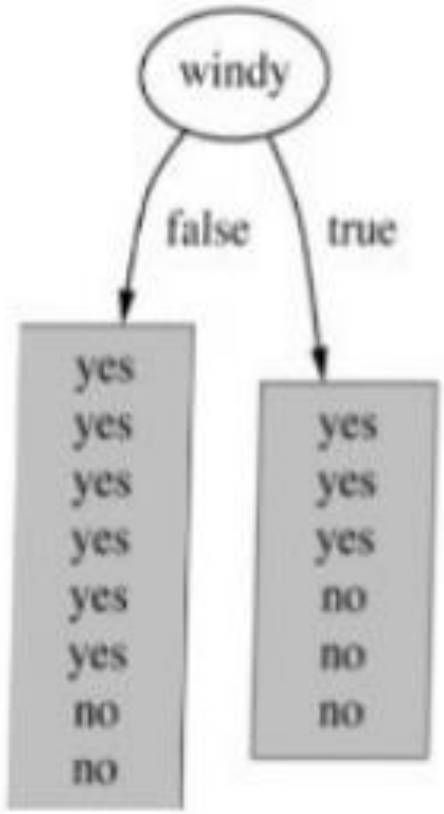
INFORMATION GAIN

Definition: Information gain (IG) measures how much "information" a feature gives us about the class.

Why it matters ?

- Decision Trees algorithm will always try to maximize Information gain.
- An attribute with highest Information gain will be tested/split first.

- Information gain = entropy(parent) – [weighted average] * entropy(children)



$$E\ (\text{Windy=false}) = -\frac{6}{8}\log\left(\frac{6}{8}\right) - \frac{2}{8}\log\left(\frac{2}{8}\right) = 0.811$$

$$E\ (\text{Windy=true}) = -\frac{3}{6}\log\left(\frac{3}{6}\right) - \frac{3}{6}\log\left(\frac{3}{6}\right) = 1$$

Average entropy information for Windy

$$I\ (\text{Windy}) = \frac{8}{14} * 0.811 + \frac{6}{14} * 1 = 0.892$$

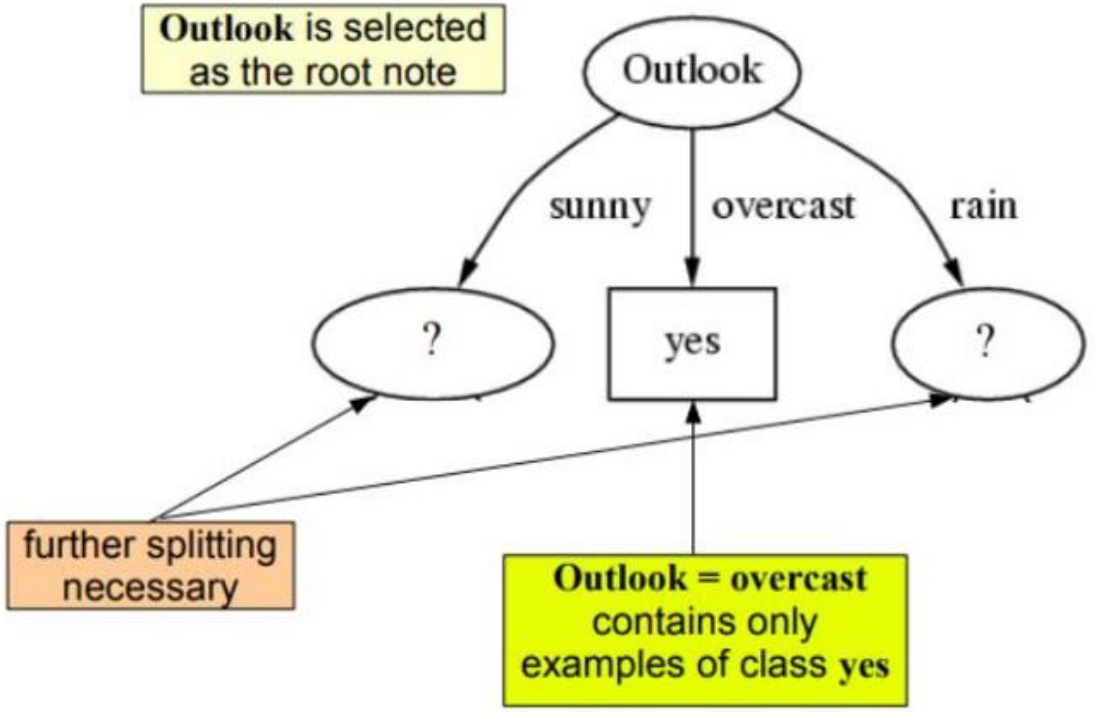
Gain (Windy) = E(S) – *I* (Windy) = 0.94-0.892=0.048

Similarity we can calculate for other two attributes(Humidity and Temp).

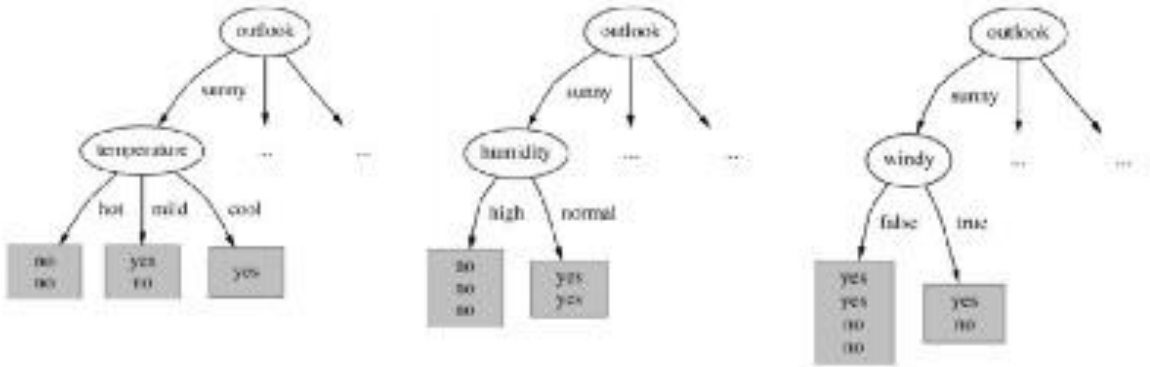
- Pick the highest gain attribute.

Outlook		Temperature	
Info:	0.693	Info:	0.911
Gain: 0.940-0.693	0.247	Gain: 0.940-0.911	0.029
Humidity		**Windy**	
Info:	0.788	Info:	0.892
Gain: 0.940-0.788	0.152	Gain: 0.940-0.892	0.048

So our root node is **Outlook.**

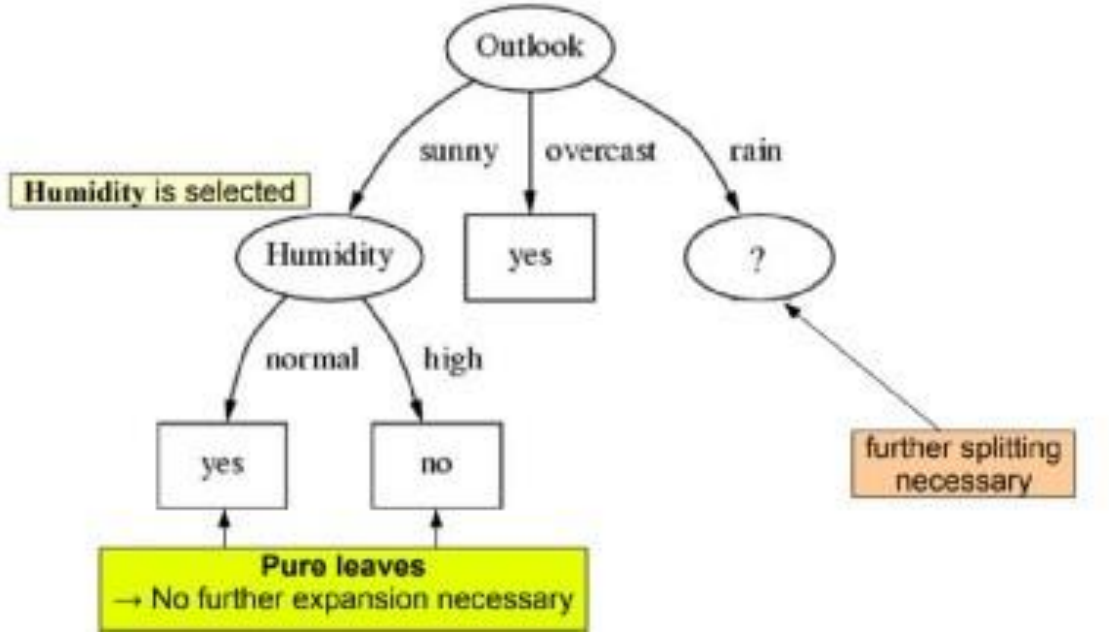


Repeat the same thing for sub-trees till we get the tree.

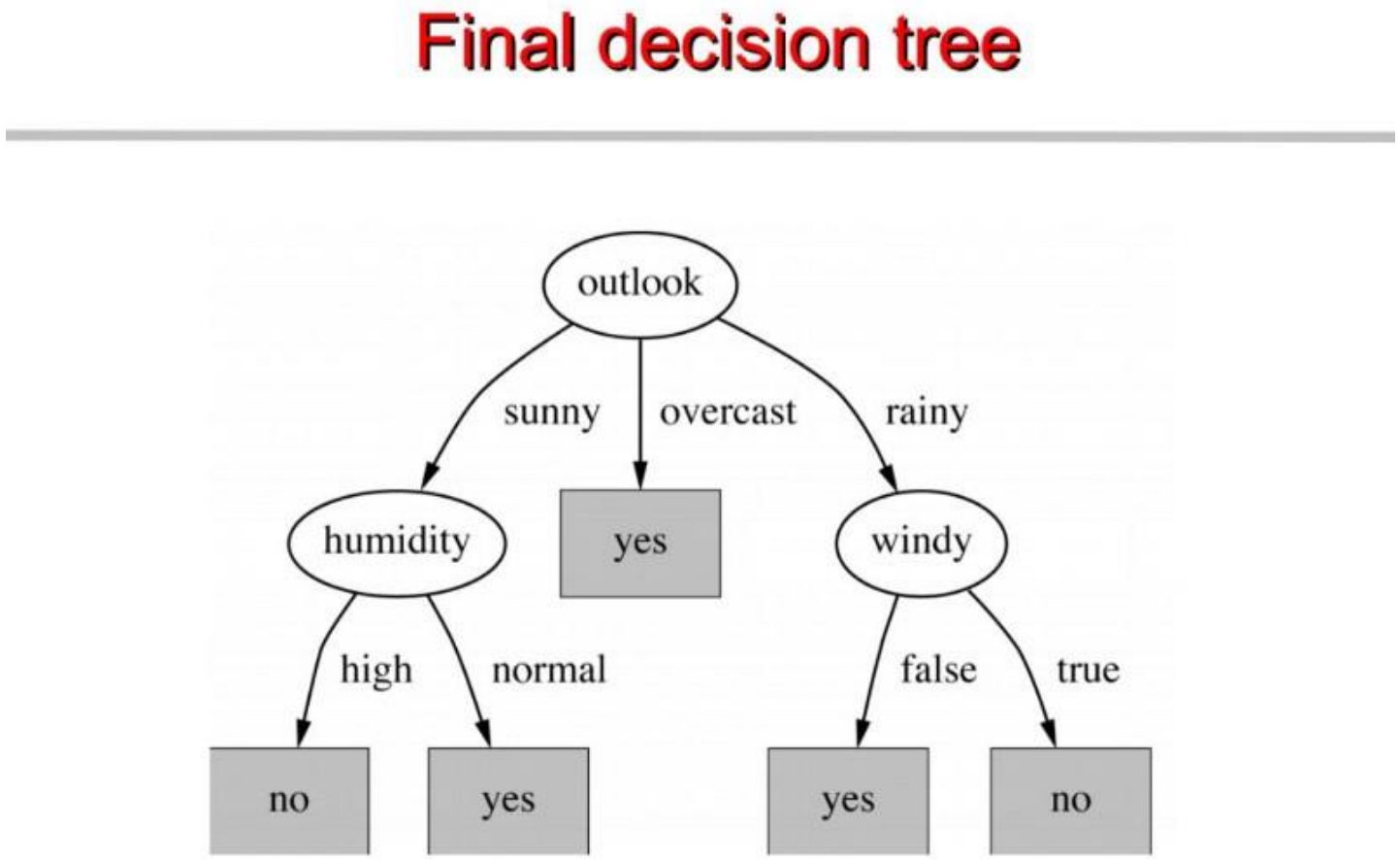


Gain(*Temperature*) = 0.571 bits
Gain(*Humidity*) = 0.971 bits
Gain(*Windy*) = 0.020 bits

Humidity is selected



Finally we get the tree something like this.



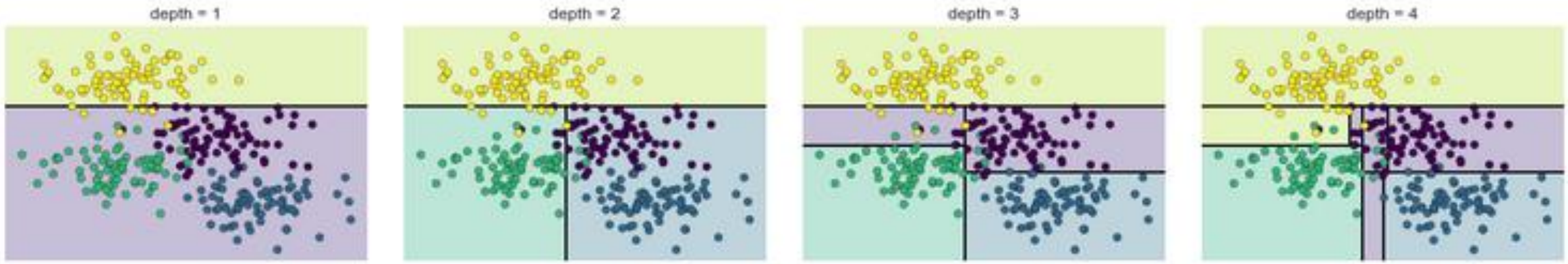
GINI INDEX

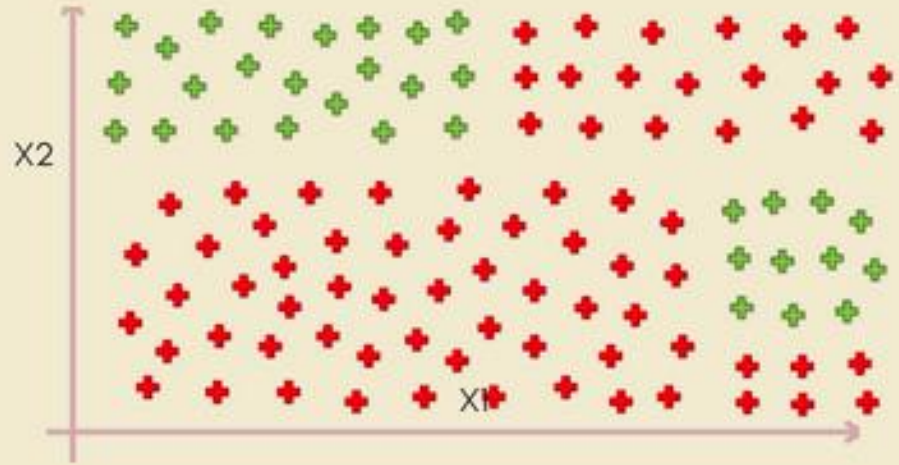
- Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified.
- It means an attribute with lower gini index should be preferred.

GINI Index

$$Gini = \sum_{i \neq j} p(i)p(j)$$

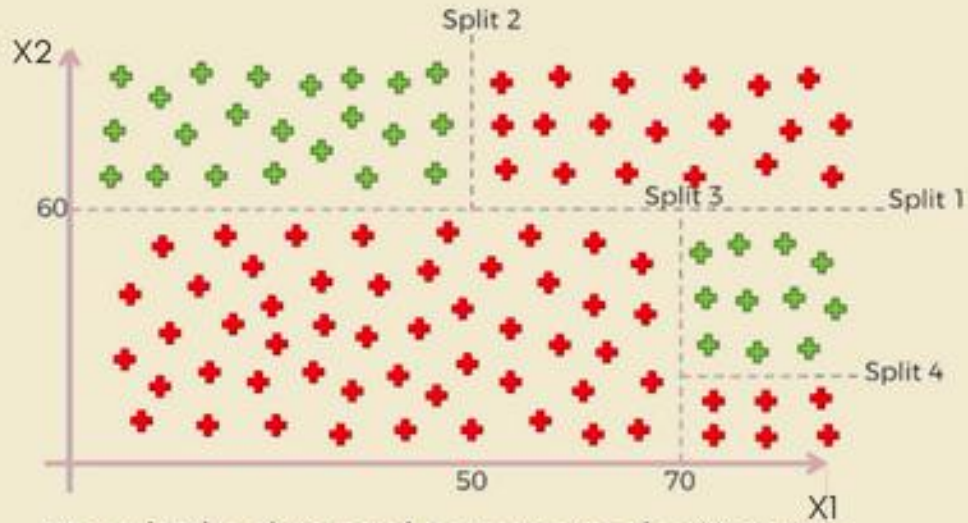
i and j are levels of the target variable


depth = 1
depth = 2
depth = 3
depth = 4

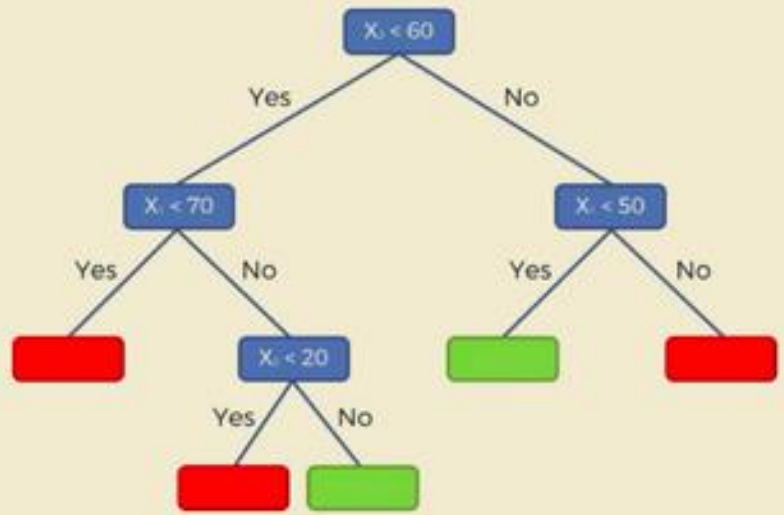


Here we've got an example with lots of points on our two dimensional scatter plot.
Now how does a decision tree work.

So what it is going to do is cut it up into slices in several iterations.



We split the data and construct a decision tree side by side which we will use later. This very task is achieved by using various algorithms. It builds a decision tree from a fixed set of examples and the resulting tree is used to classify future samples.



The resulting Tree (obtained by applying algorithms like CART, ID3) which will be later used to predict the outcomes