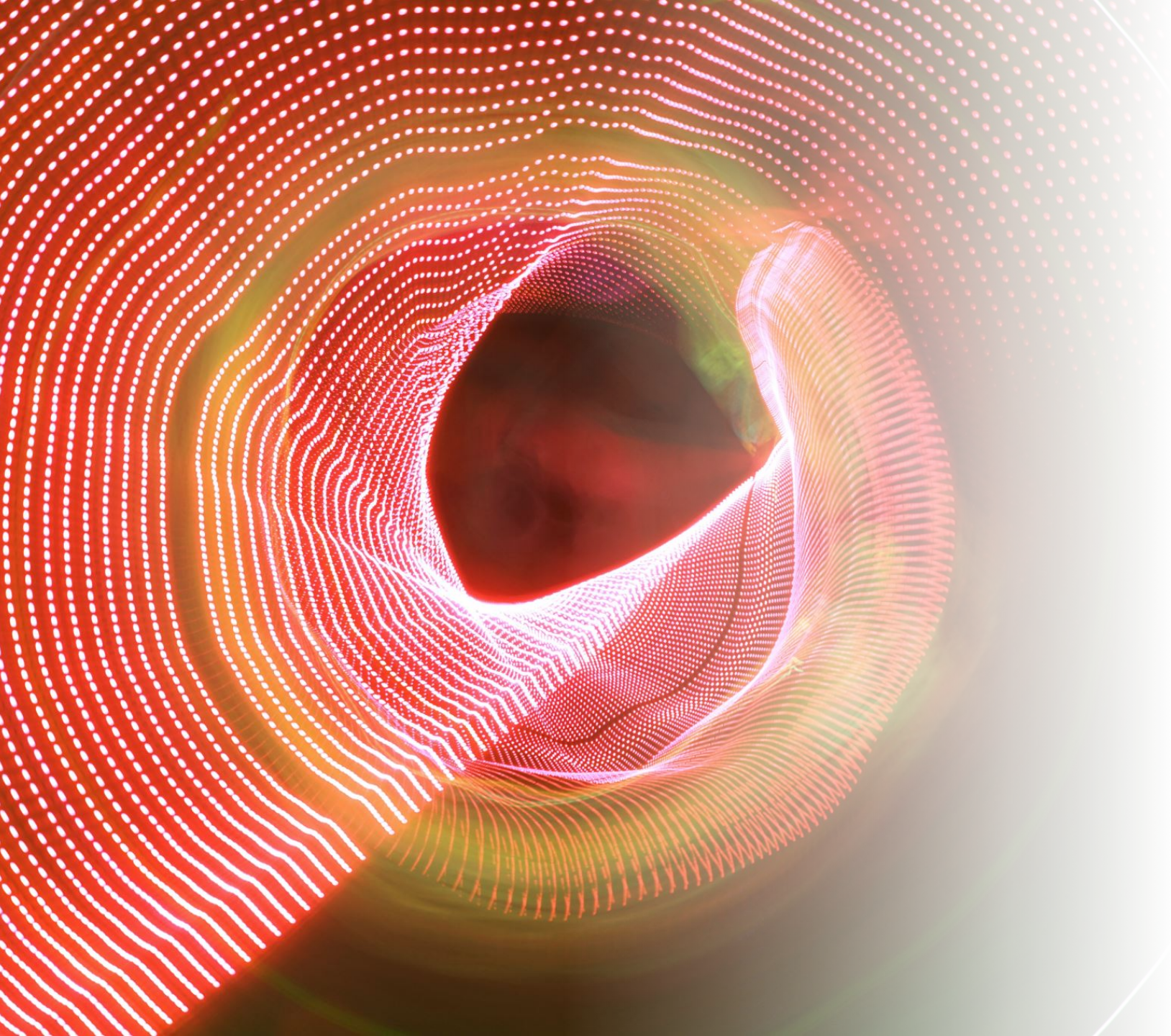


Supervised Learning

Decision Trees

Agenda

- What is CART
- What is ID3
- Decision Tree Algorithm
- Applications for Decision Tree
- Attribute Selection Measures
- What is Entropy
- What is Gini Index
- Advantages of Decision Tree
- Disadvantages of Decision Tree
- case studies and practical

What is CART

CART (Classification and Regression Tree) methodology, which is also known as *recursive partitioning*

What is a Decision Tree

Decision Tree Analysis is a general, predictive modelling tool that has applications spanning a number of different areas. In general, decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions. It is one of the most widely used and practical methods for supervised learning. Decision Trees are a non-parametric supervised learning method used for both classification and regression tasks. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

What is a Decision Tree

Decision Tree Analysis is a general, predictive modelling tool that has applications spanning a number of different areas. In general, decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions. It is one of the most widely used and practical methods for supervised learning. Decision Trees are a non-parametric supervised learning method used for both classification and regression tasks. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

The decision rules are generally in form of if-then-else statements. The deeper the tree, the more complex the rules and fitter the model.

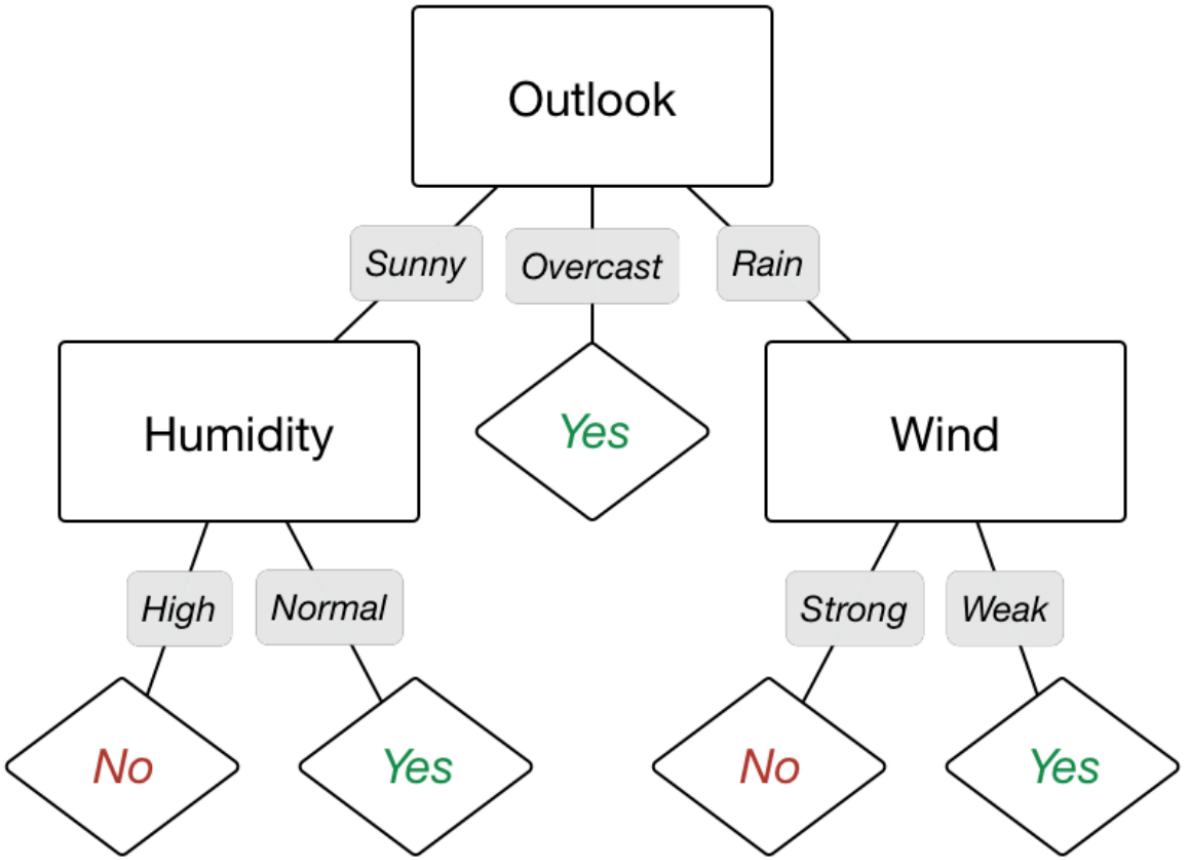
Decision Tree Example

Let's assume we want to play Tennis on a particular day – say Saturday – how will you decide whether to play or not. Let's say you go out and check if it's hot or cold, check the speed of the wind and humidity, how the weather is, i.e. is it sunny, cloudy, or rainy. You take all these factors into account to decide if you want to play or not.

Decision Tree Example

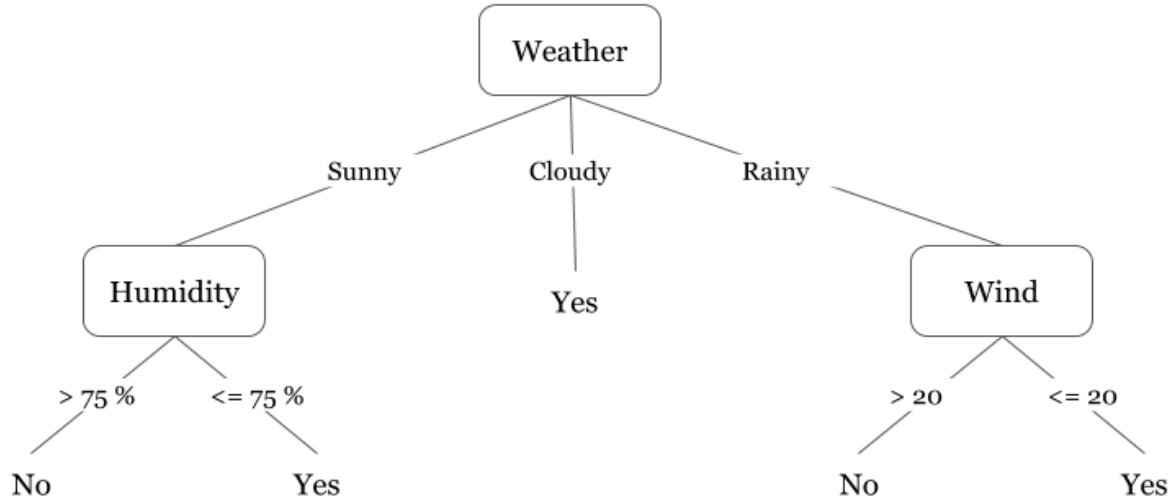
Outlook	Temperature	Humidity	Windy	PlayTennis
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Decision tree. A simple form of If Else



Decision tree. A simple form of If Else

We can see that each node represents an attribute or feature and the branch from each node represents the outcome of that node. Finally, its the leaves of the tree where the final decision is made. If features are continuous, internal nodes can test the value of a feature against a threshold



Applications for Decision Tree

Decision trees have a natural “if … then … else …” construction that makes it fit easily into a programmatic structure. They also are well suited to categorization problems where attributes or features are systematically checked to determine a final category. For example, a decision tree could be used effectively to determine the species of an animal.

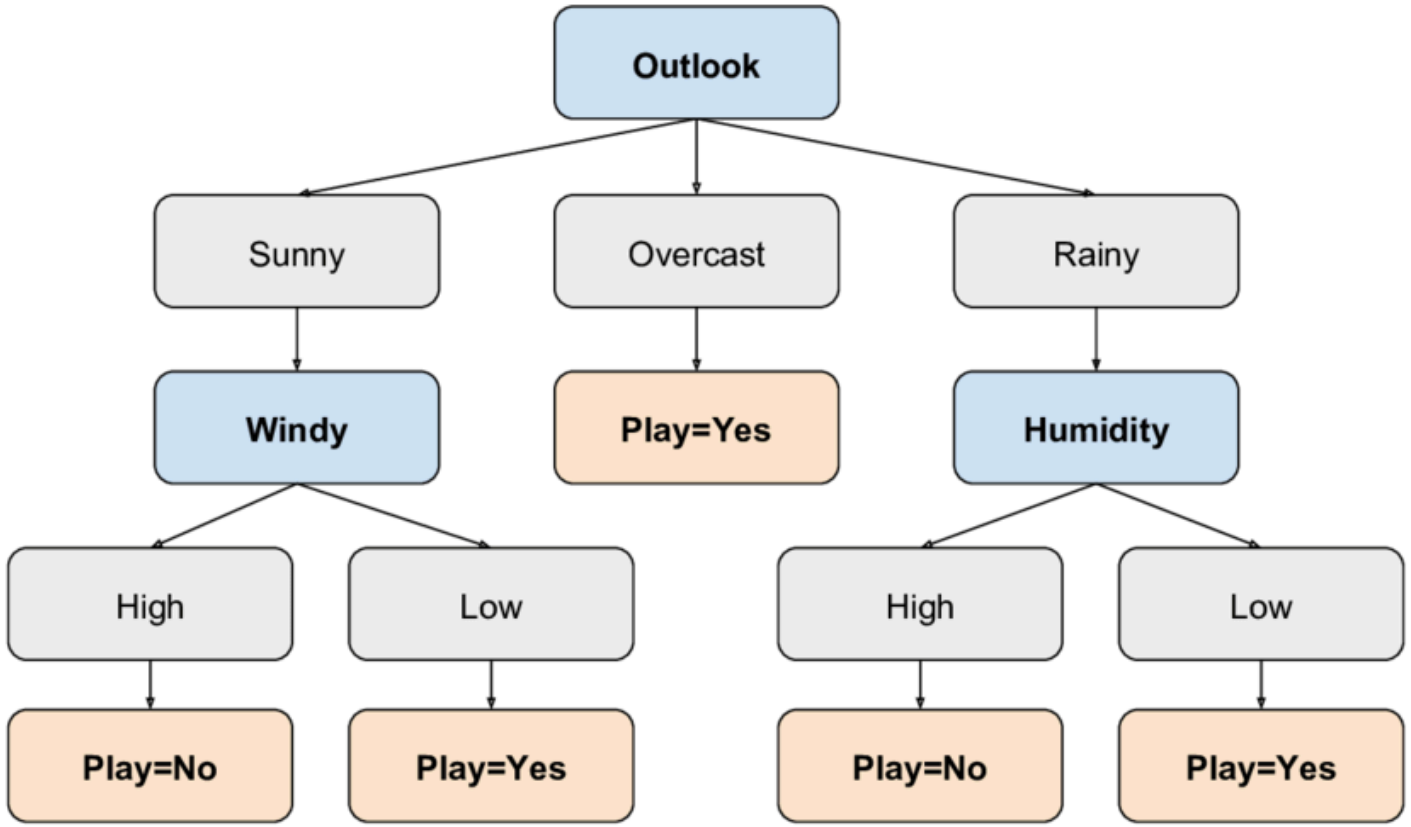
General algorithm for a decision tree

A general algorithm for a decision tree can be described as follows:

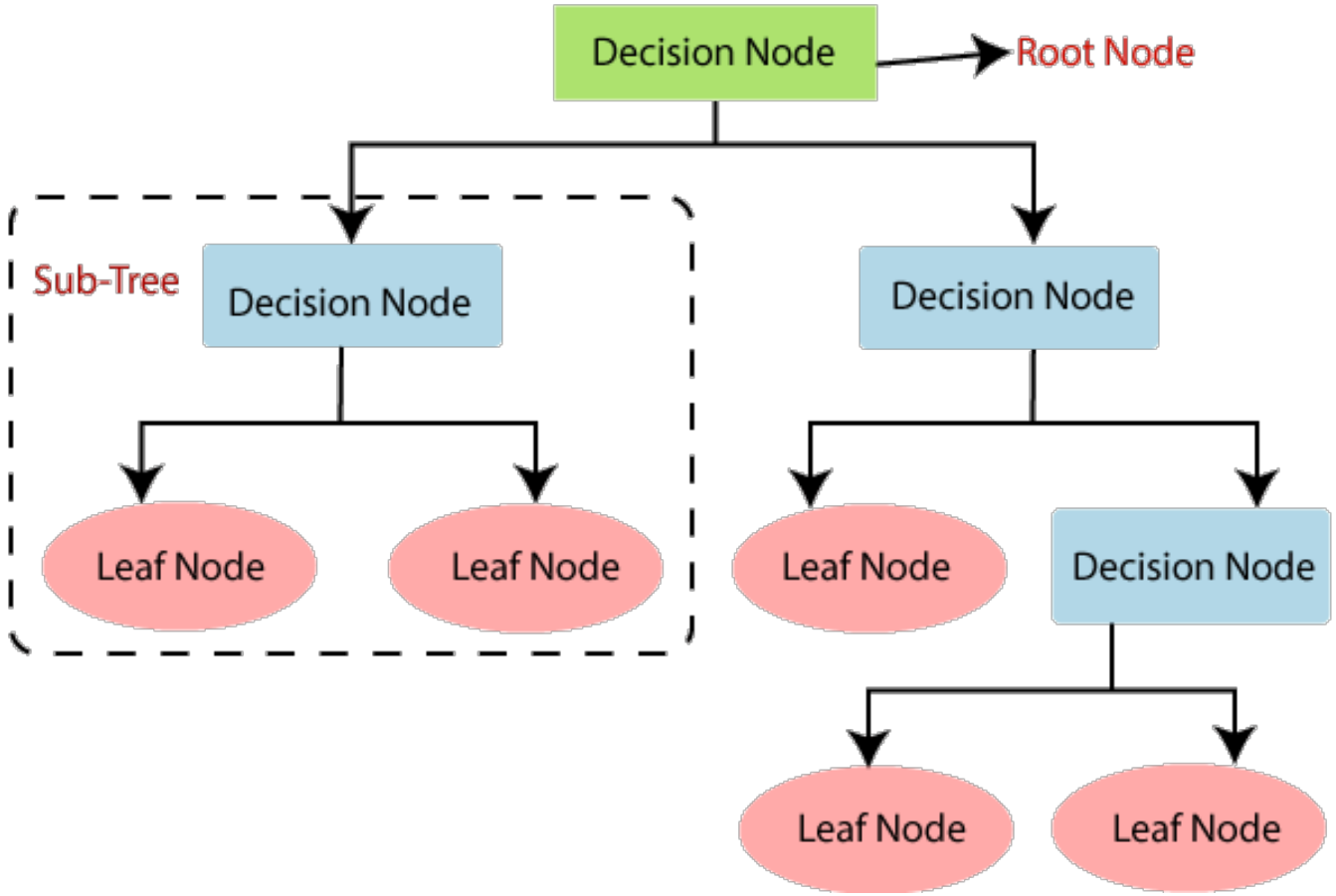
1. Pick the best attribute/feature. The best attribute is one which best splits or separates the data.
2. Ask the relevant question.
3. Follow the answer path.
4. Go to step 1 until you arrive to the answer.

The best split is one which separates two different labels into two sets.

Bigger Tree


Outlook
Sunny
Overcast
Rainy
Windy
Play=Yes
Humidity
High
Low
High
Low
Play=No
Play=Yes
Play=No
Play=Yes

Parts of Decision trees



Common terms used with 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 do not split is called Leaf or Terminal node.
- Pruning: When we remove sub-nodes of a decision node, this process is called pruning. You can say opposite process of splitting.
- Branch / Sub-Tree: A sub section of entire 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 whereas sub-nodes are the child of parent node.

Types of Decision Trees

- **Categorical Variable Decision Tree:** Decision Tree which has categorical target variable then it called as categorical variable decision tree. E.g.:- In above scenario of student problem, where the target variable was “Student will play cricket or not” i.e. YES or NO.

- **Continuous Variable Decision Tree:** Decision Tree has continuous target variable then it is called as Continuous Variable Decision Tree

Decision tree for classification

Decision tree learning is a method that uses inductive inference to approximate a target function, which will produce discrete values. It is widely used, robust to noisy data, and considered a practical method for learning disjunctive expressions.

Properties of Decision trees for classifications

Instances are represented by attribute-value pairs.

- Instances are described by a fixed set of attributes (e.g., temperature) and their values (e.g., hot).
- The easiest situation for decision tree learning occurs when each attribute takes on a small number of disjoint possible values (e.g., hot, mild, cold).
- Extensions to the basic algorithm allow handling real-valued attributes as well (e.g., a floating point temperature).

The target function has discrete output values.

A decision tree assigns a classification to each example.

- Simplest case exists when there are only two possible classes (Boolean classification).
- Decision tree methods can also be easily extended to learning functions with more than two possible output values.

Properties of Decision trees for classifications

- Disjunctive descriptions may be required.
  - Decision trees naturally represent disjunctive expressions.
- The training data may contain errors.
  - Decision tree learning methods are robust to errors - both errors in classifications of the training examples and errors in the attribute values that describe these examples.
- The training data may contain missing attribute values.
  - Decision tree methods can be used even when some training examples have unknown values (e.g., humidity is known for only a fraction of the examples).
- Learned functions are either represented by a decision tree or re-represented as sets of if-then rules to improve readability.

How Decision tree make its choices

A decision tree is constructed by looking for regularities in data.

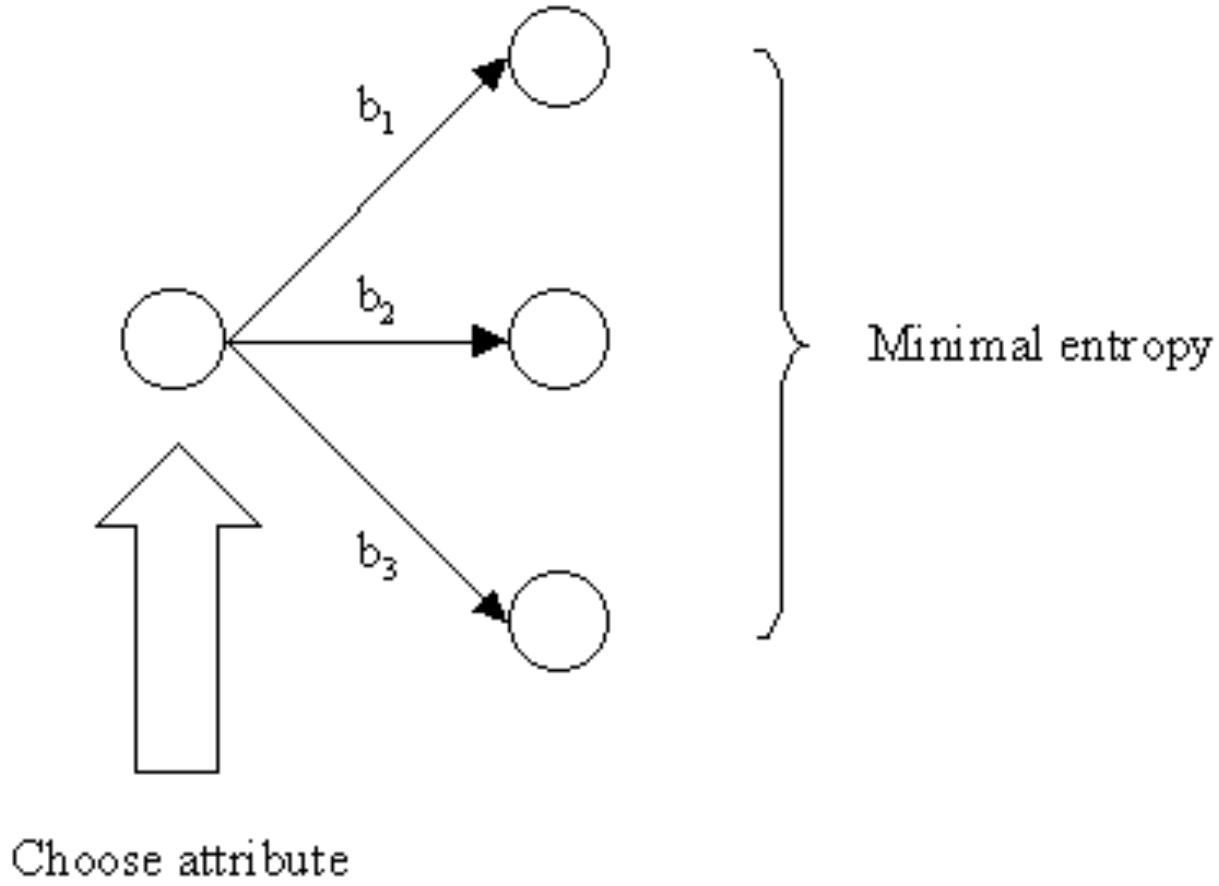
General Form

- Until each leaf node is populated by as homogeneous a sample set as possible:
- Select a leaf node with an inhomogeneous sample set.
- Replace that leaf node by a test node that divides the inhomogeneous sample set into minimally inhomogeneous subsets, according to an entropy calculation.

Specific Form

- Examine the attributes to add at the next level of the tree using an entropy calculation.
- Choose the attribute that minimizes the entropy.

How Decision tree make its choices



Algorithm used in decision trees

- ID3
- Gini Index
- Chi-Square
- Reduction in Variance

Special terms before you understand ID3

A **greedy algorithm** is any **algorithm** that follows the problem-solving heuristic of making the locally optimal choice at each stage with the intent of finding a global optimum.

A globally optimal solution is a feasible solution with an objective value that is as good or better than all other feasible solutions to the model. The ability to obtain a globally optimal solution is attributable to certain properties of linear models.

- In general, solvers return a local minimum (or optimum). The result might be a global minimum (or optimum), but this result is not guaranteed.
- A *local* minimum of a function is a point where the function value is smaller than at nearby points, but possibly greater than at a distant point.
- A *global* minimum is a point where the function value is smaller than at all other feasible points.

What is entropy : ID3

The core algorithm for building decision trees is called **ID3.** Developed by J. R. Quinlan, this algorithm employs a top-down, greedy search through the space of possible branches with no backtracking. ID3 uses ***Entropy*** and ***Information Gain*** to construct a decision tree.

What is entropy : ID3

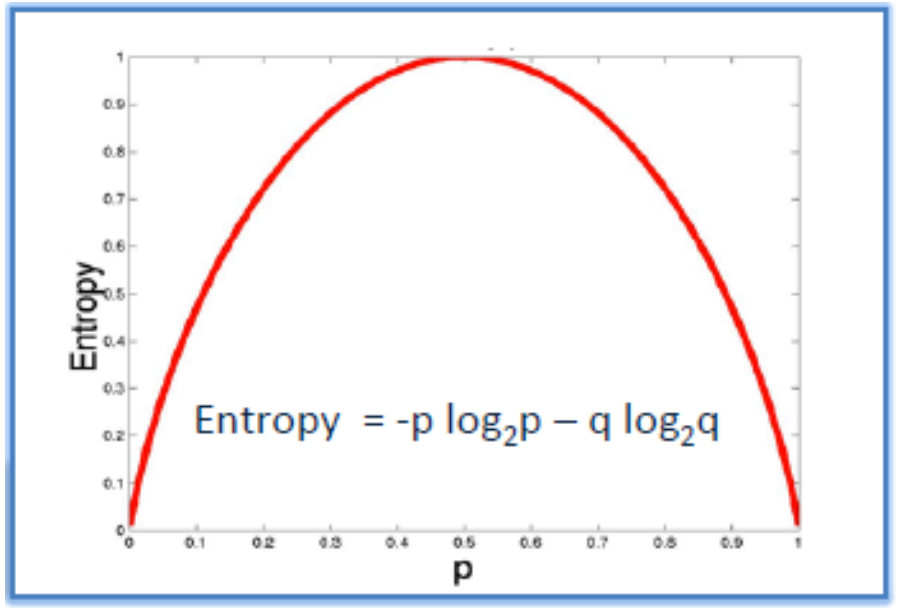
Entropy is the measure of disorder in a dataset

Information gain is the measure of the decrease in disorder by partitioning the parent dataset

High knowledge -------- low entropy

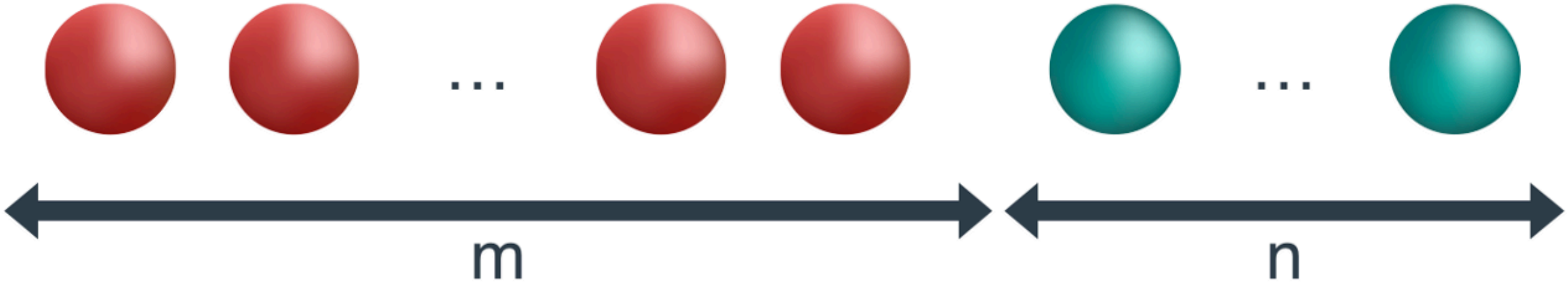
What is entropy : ID3

A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogeneous). ID3 algorithm uses entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is equally divided then it has entropy of one.



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

What is entropy : ID3



$$\text{Entropy} = \frac{-m}{m+n}\log_2\left(\frac{m}{m+n}\right) + \frac{-n}{m+n}\log_2\left(\frac{n}{m+n}\right)$$

What is entropy : ID3

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

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

a) Entropy using the frequency table of one attribute:

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

What is entropy : ID3

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

b) Entropy using the frequency table of two attributes:

		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)

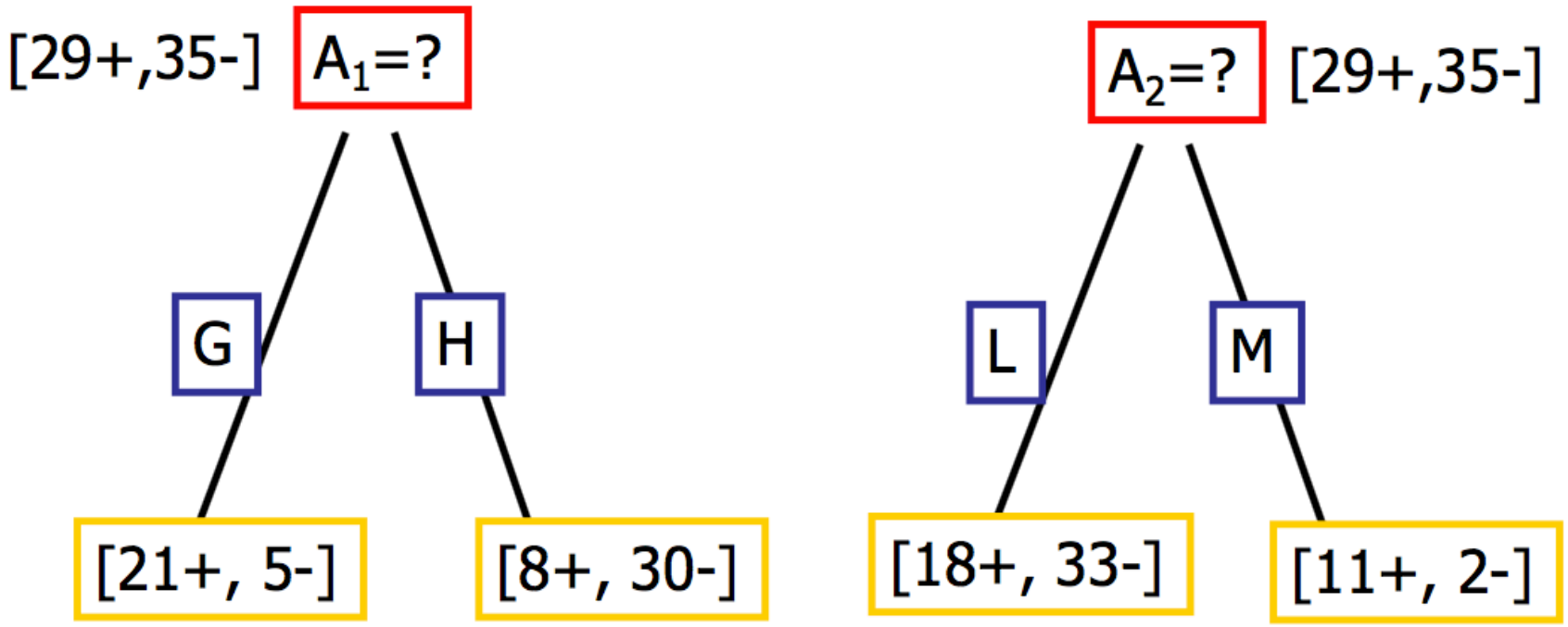
= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

What is entropy : 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).

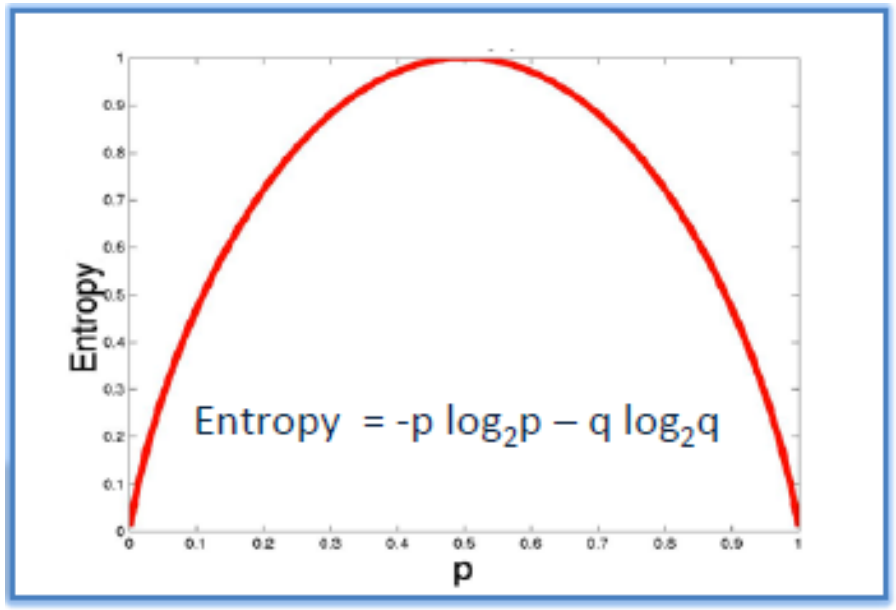
Example: Which attribute is best?



Example: Which attribute is best?

- S is a sample of training examples
- p_+ is the proportion of positive examples
- p_- is the proportion of negative examples
- Entropy measures the impurity of S

$$\text{Entropy}(S) = -p_+ \log_2 p_+ - p_- \log_2 p_-$$

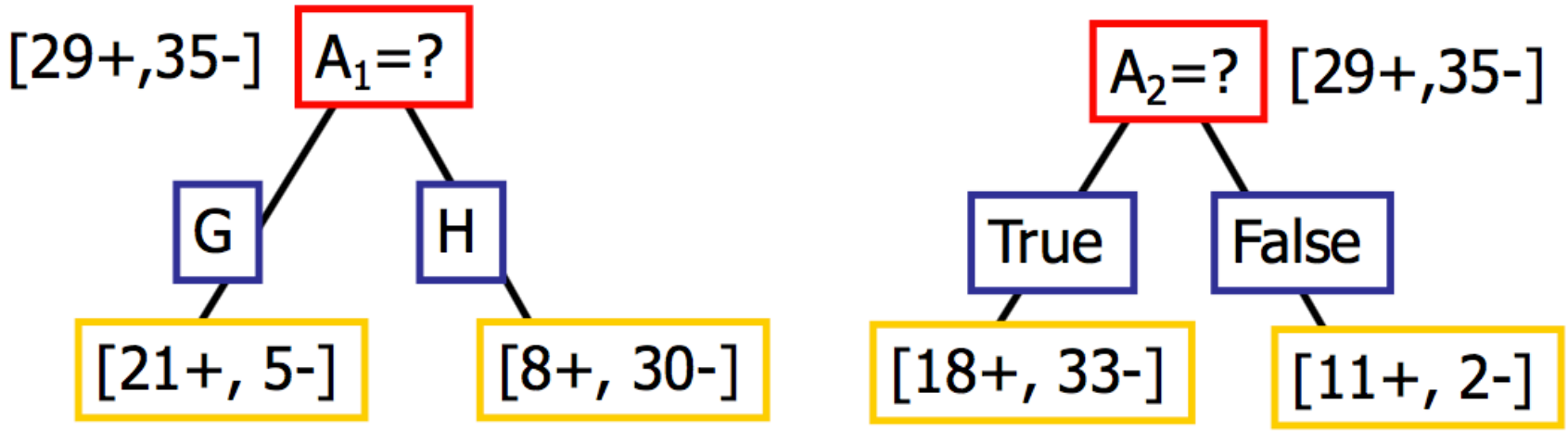


Example: Which attribute is best?

- Gain(S,A): expected reduction in entropy due to sorting S on attribute A

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in D_A} \frac{|S_v|}{|S|} Entropy(S_v)$$

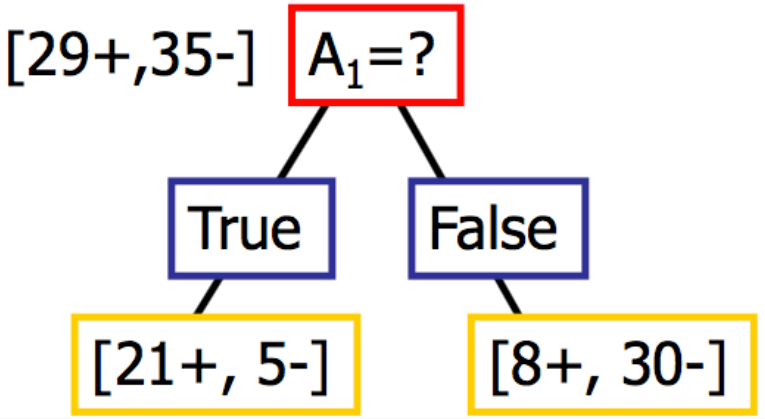
Entropy([29+,35-]) = -29/64 log2 29/64 – 35/64 log2 35/64
= 0.99

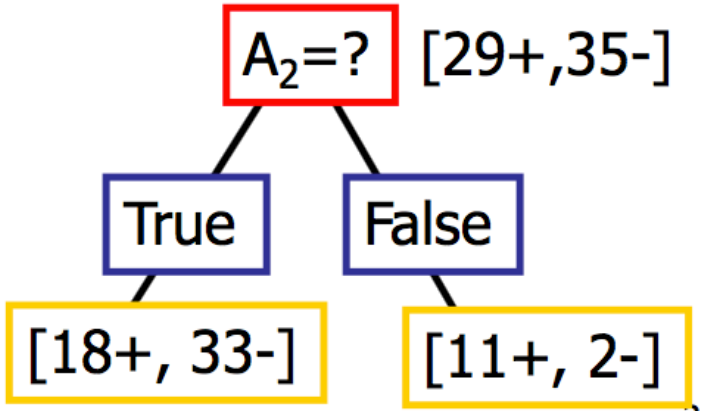


Example: Which attribute is best?

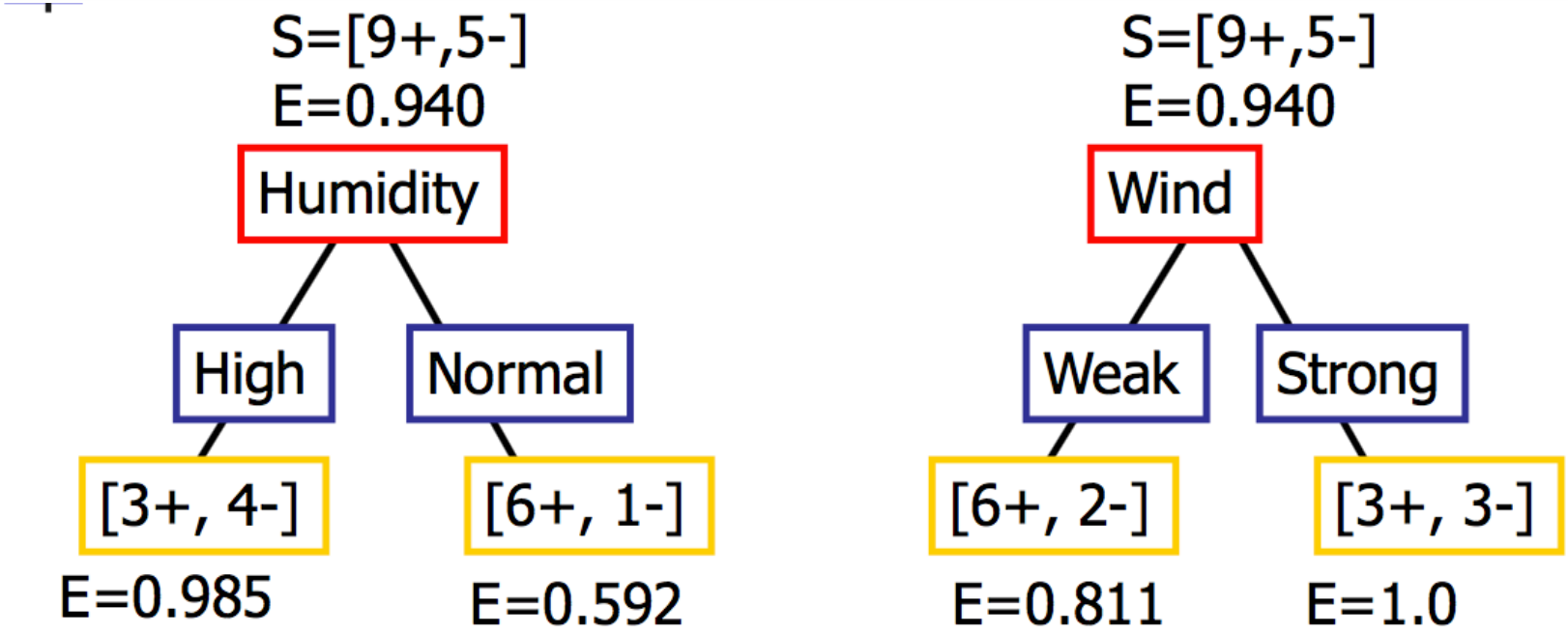
Entropy([21+,5-]) = 0.71
Entropy([8+,30-]) = 0.74
Gain(S,A_1)=Entropy(S)
-26/64*Entropy([21+,5-])
-38/64*Entropy([8+,30-])
=0.27

Entropy([18+,33-]) = 0.94
Entropy([11+,2-]) = 0.62
Gain(S,A2)=Entropy(S)
-51/64*Entropy([18+,33-])
-13/64*Entropy([11+,2-])
=0.12





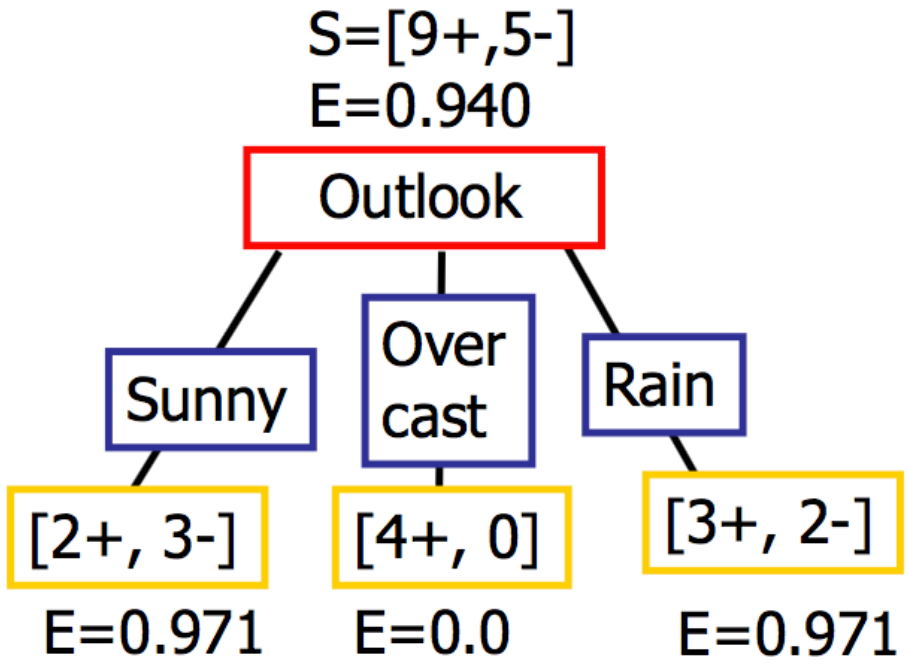
Example: Which attribute is best?



Gain(S,Humidity)
=0.940-(7/14)*0.985
– (7/14)*0.592
=0.151

Gain(S,Wind)
=0.940-(8/14)*0.811
– (6/14)*1.0
=0.048

Example: Which attribute is best?



Gain(S,Outlook)
=0.940-(5/14)*0.971
-(4/14)*0.0 – (5/14)*0.0971
=0.247

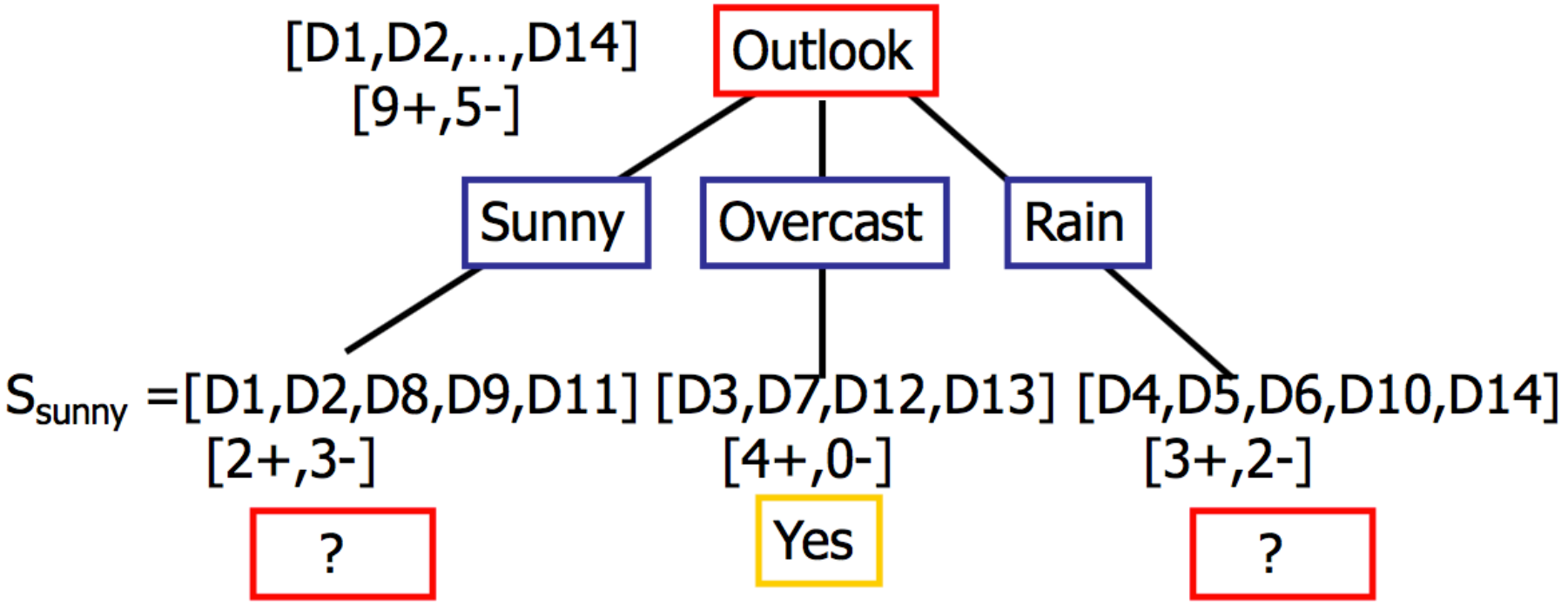
Example: Which attribute is best?

The information gain values for the 4 attributes are:

- Gain(S,Outlook) =0.247
- Gain(S,Humidity) =0.151
- Gain(S,Wind) =0.048
- Gain(S,Temperature) =0.029

where S denotes the collection of training examples

Example: Which attribute is best?



$Gain(S_{sunny}, Humidity)=0.970-(3/5)0.0 - 2/5(0.0) = 0.970$

$Gain(S_{sunny}, Temp.)=0.970-(2/5)0.0 -2/5(1.0)-(1/5)0.0 = 0.570$

$Gain(S_{sunny}, Wind)=0.970= -(2/5)1.0 - 3/5(0.918) = 0.019$

Advantages of Decision Tree

- 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 scaling of data as well.
- Missing values in the data also does NOT affect the process of building decision tree to any considerable extent.
- A Decision trees model is very intuitive and easy to explain to technical teams as well as stakeholders.

Disadvantages of Decision Tree

- Over fitting: Decision-tree learners can create over-complex trees that do not generalize the data well. This is called overfitting. Over fitting is one of the most practical difficulty for decision tree models. This problem gets solved by setting constraints on model parameters and pruning.
- Information gain in a decision tree with categorical variables gives a biased response for attributes with greater no. of categories.
- Generally, it gives low prediction accuracy for a dataset as compared to other machine learning algorithms.
- Calculations can become complex when there are many class label.

What is Gini Index

According to Wikipedia, the goal is to "measure how often a randomly chosen element from the set would be incorrectly labeled"

let's go back to the gumball examples. If we decided to arbitrarily label all 4 gumballs as red, how often would one of the gumballs be incorrectly labeled?

4 red and 0 blue:

A Gini score of 0 is the most pure score possible.

$$Gini\ Index = 1\text{-}(probability\_red^2 + probability\_blue^2) = 1\text{-}(1^2 + 0^2) = 0$$

The impurity measurement is 0 because we would never incorrectly label any of the 4 red gumballs here. If we arbitrarily chose to label all the balls 'blue', then our index would still be 0, because we would always incorrectly label the gumballs.

What is Gini Index

2 red and 2 blue:

$$Gini\ Index = 1\text{-}(probability\_red^2 + probability\_blue^2) = 1\text{-}(0.5^2 + 0.5^2) = 0.5$$

The impurity measurement is 0.5 because we would incorrectly label gumballs wrong about half the time. Because this index is used in binary target variables (0,1), a gini index of 0.5 is the least pure score possible. Half is one type and half is the other. Dividing gini scores by 0.5 can help intuitively understand what the score represents. 0.5/0.5 = 1, meaning the grouping is as impure as possible (in a group with just 2 outcomes).

What is Gini Index

3 red and 1 blue:

$$Gini\ Index = 1\text{-}(probability\_red^2 + probability\_blue^2) = 1\text{-}(0.75^2 + 0.25^2) = 0.375$$

The impurity measurement here is 0.375. If we divide this by 0.5 for more intuitive understanding we will get 0.75, which is the probability of incorrectly/correctly labeling.

What is Gini Index

Gini Index Intuition:

$$Gini = 1 - \sum_j p_j^2$$

The impurity measurement here is 0.375. If we divide this by 0.5 for more intuitive understanding we will get 0.75, which is the probability of incorrectly/correctly labeling.

Final Takeaways:

- Gini's maximum impurity is 0.5 and maximum purity is 0
- Entropy's maximum impurity is 1 and maximum purity is 0
- Different decision tree algorithms utilize different impurity metrics: CART uses Gini; ID3 and C4.5 use Entropy. This is worth looking into before you use decision trees /random forests in your model.

Practical case study