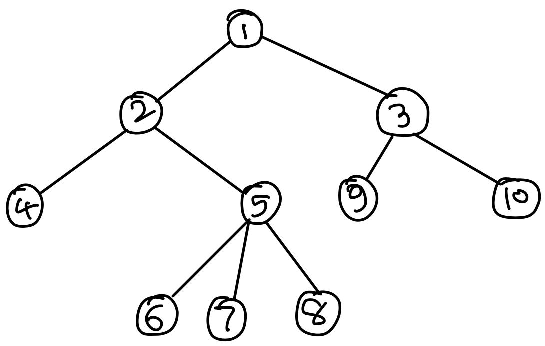
Decision trees

Intelligence can be captured in a set of if then else rules that provide branching for classification.

Trees as arrays:



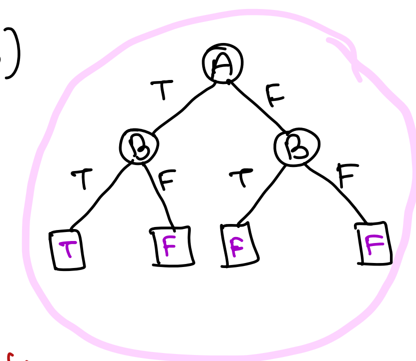
parents

0	1	2	3	4	5	6	7	8	9	10
x	x	1	1	2	2	5	5	5	3	3

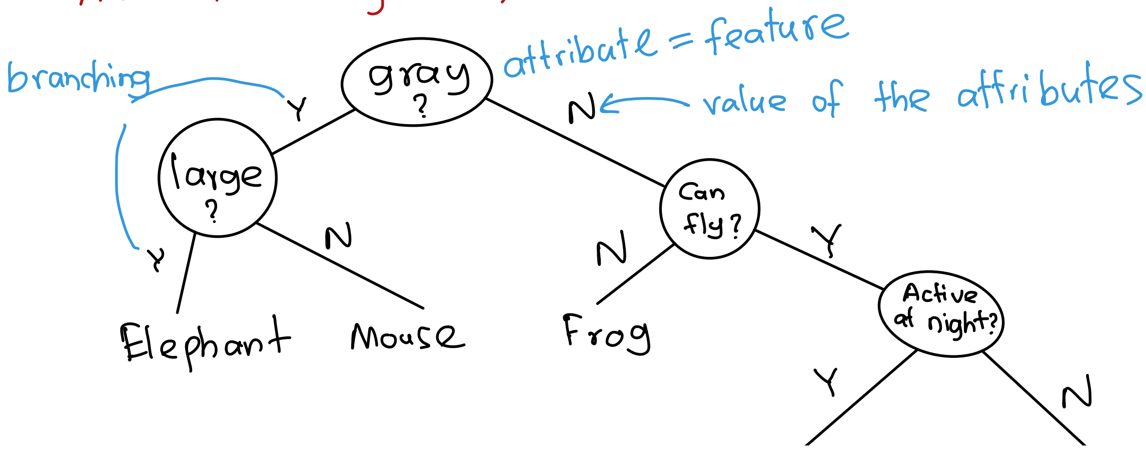
array

Logical propositions

$(A \& B) \mid (\neg A \& \neg B)$

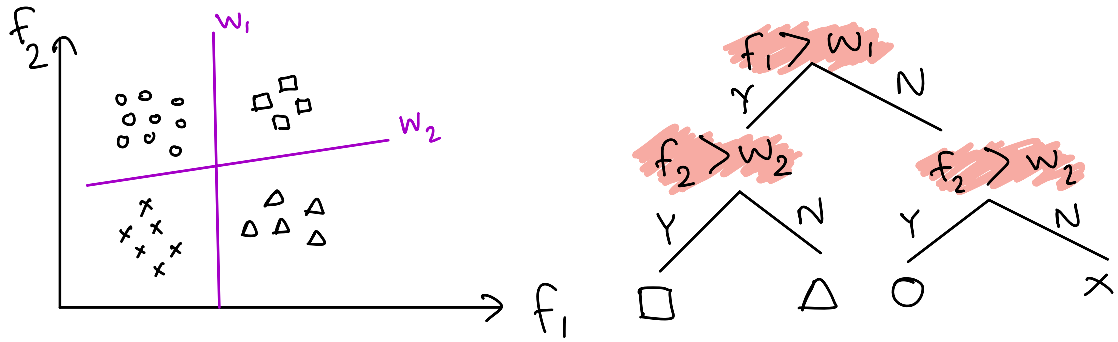


Animal Recognition



Decision trees and Classification

Owl Eagle



DTs
- nodes verify/evalulate attributes.
- branches that embody attribute values.
- leaves categorize/classify instances

Why DTs could be a good AI choice?

- output is discrete
- no large data is available
- data is noisy
- classes are disjoint

* How do I grow a tree?
, when I have many attributes

* How to select the best attribute to generate the most compact bronching?

Lets restrict things to binary.

S : Set of training samples

$S_\oplus$: Positive samples

$S_\ominus$: Negative samples

$-\log_2 2^{-1}$

$$P_\oplus = \frac{|S_\oplus|}{|S|} \qquad\qquad P_\ominus = \frac{|S_\ominus|}{|S|}$$

$$\text{Entropy}(S) = -P_\oplus \log_2 P_\oplus - P_\ominus \log_2 P_\ominus$$

We know (from information theory) that the optimal length code for a message with probability p is $-\log_2 p$ bits.

Entropy quantifies the (expected) number of bits to encode a class of randomly drawn samples. But to construct a tree, we need to know how much we gain, when we add a specific attribute. $S_v \subset S$

$$\underbrace{\text{Gain}(S, A)}_{\text{expected reduction in entrophy upon sorting on } A} = \text{Entropy}(S) - \sum_{v \in V_A} \frac{|S_v|}{|S|} \cdot \text{Entropy}(S_v)$$

all values of all attributes

Play Tennis

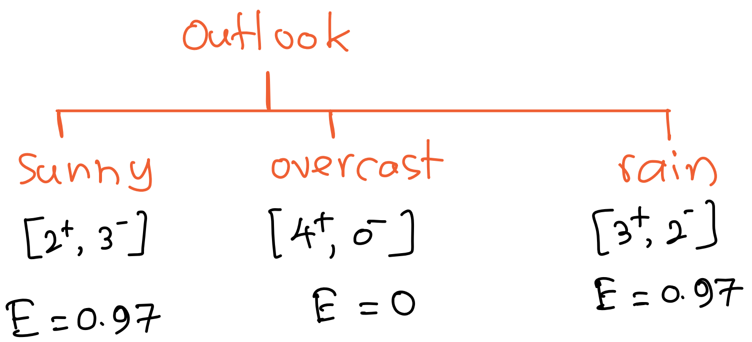
attributes | | | decision

Day	Outlook	Temp	...	Play
				Y
				N

$S = [9^+, 5^-]$ $\quad E(S) = 0.94$

$p^{\oplus} = \frac{9}{14} = 0.6429$

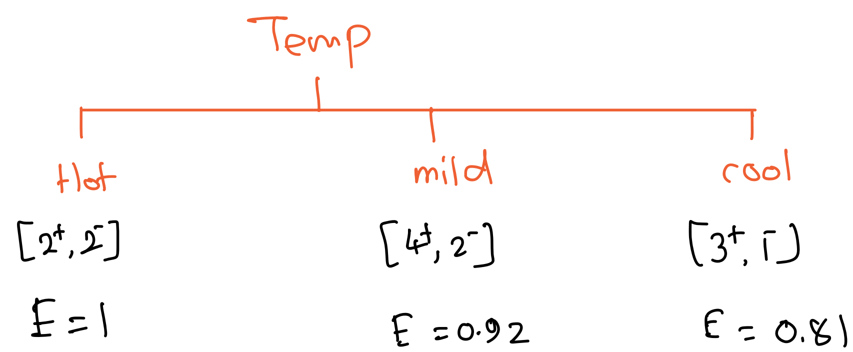
$p^{\ominus} = \frac{5}{14} = 0.3571$



Cant make decision with this

$$Gain(S, outlook) = 0.94 - \left(\frac{5}{14} \times 0.97 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.97\right) = 0.246$$

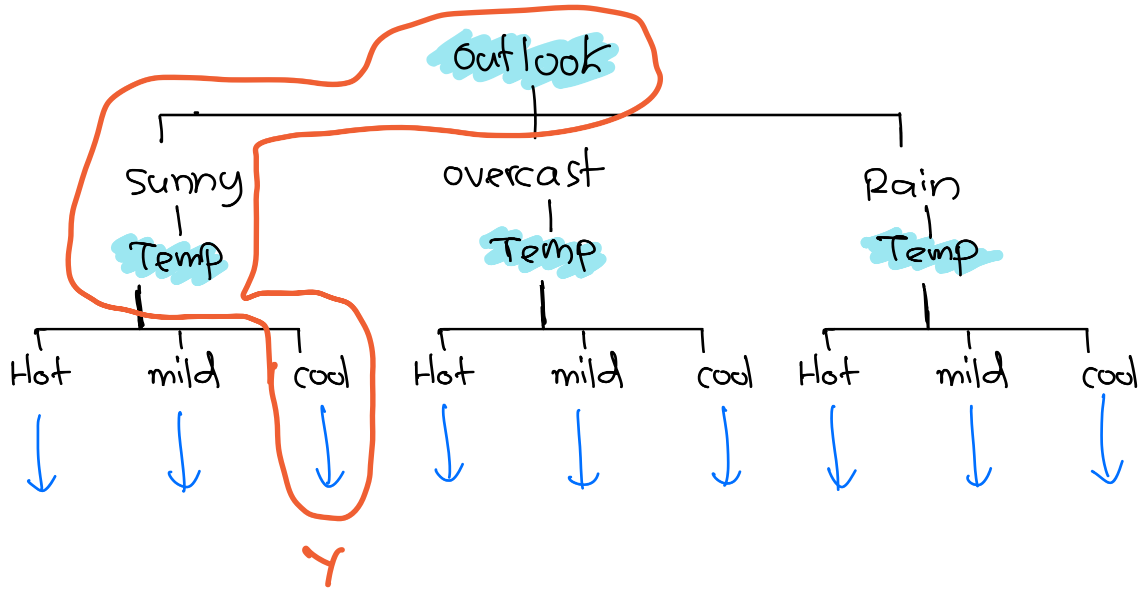
* calculate gain for other attributes



$$Gain(S, Temp) = 0.94 - \left(\frac{4}{14} \times 1 + \frac{6}{14} \times 0.92 + \frac{4}{16} \times 0.81\right) = 0.029$$

Gain(S, outlook) > Gain(S, Temp)

start with outlook at the top



Intelligence : Entropy and info gain

DT construction algorithms

* ID3 (Iterative Dichotomizer V3) what we did
* C45 and C5

Overfitting ? very large tree (eg: million trees)

We prefer short/small trees (compact trees)

Avoid overfitting !

1. Grow full tree ⟶ post pruning
2. Stop when branching not statistically significant

K fold cross validation