# Decesion Tree

Sahasra Ranjan

April 2020

A decesion tree is a map of the possible outcomes of a series of a related choices. It allows to weigh possible actions against one another based of various factors.

It uses a tree-like model of decesion. It typically starts with a single node, which branches into possible outcomes. Each of those outcomes leads to additional nodes, which branch off into other possiblities.

| Day | *Outlook* | *Temperature* | *Humidity* | *Wind* | *PlayTennis* |
|---|---|---|---|---|---|
| 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 |

Figure 1: Dataset for possiblity of a tennis match

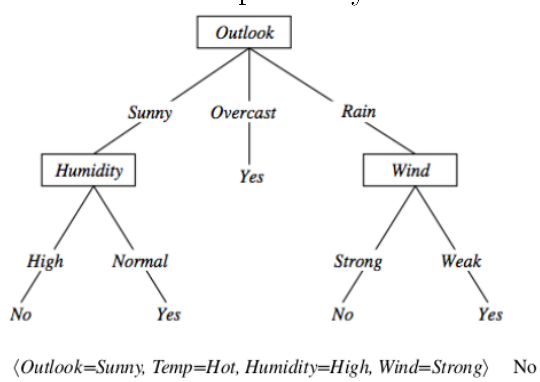



Figure 2: Decesion Tree for the same

## Advantages and Disadvantages of Decesion Trees

### Advantages:

- Performs classification without requiring much computation.
- Provides clear indication of important fields for prediction of classification.

**Disadvantages:**

- Less appropriate for predicting continious attributes.
- Computationally expensive to train

## Creating a Decesion Tree

For every node, we have to create subtrees with all the possiblities. and then further repeat for other features.

For eg., In the tennis match problem, for the first node let's check outlook, since having three possiblity (viz. Sunny, Overcast, Rainy), we created three subtrees and then further we keep asking for other features like Humidity & wind to get the final tree.

## Greedy Approach for creating decesion tree

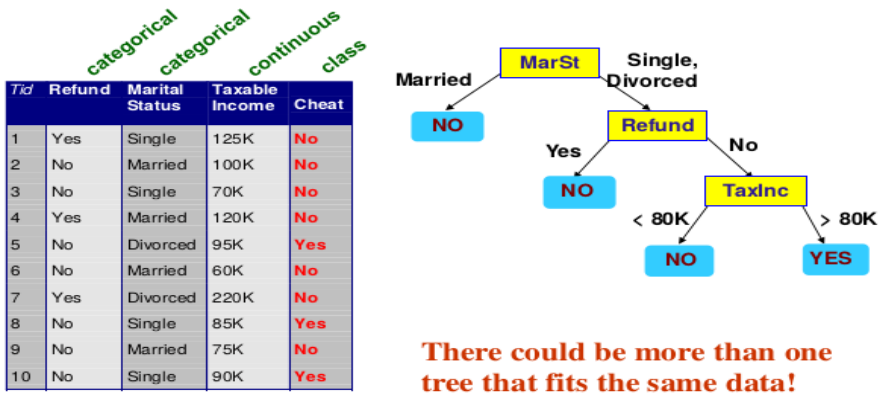
Greedy approach is implemented by making an optimal local choice at each node. By making these local optimal choices, we reach the approximate optimal solution globaly.

The algorithm can be summarized as:

1. At each stage (node), pick out the best feature as the test conditiion.
2. Now split the node into possibel outcomes (internal nodes)
3. Repeat the above steps till all the test conditions have been exhausted into leaf nodes.

## Continuous Features

There might be some feature which are not categorical, for these we need to create possibilities on the basis of aprropriate ranges. One such tree is shown below:




| Tid | Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|---|
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |

Figure 3: Decesion tree with continuous feature

## Entropy

In the most layman terms, Entropy is nothing but the **The measure of disorder**. Why is it important to study entropy for machine learning?

Entropy is a measure of disorder or uncertainity and the goal of machine learning models and Data Scientists in general is to reduce uncertainity.

The Mathematical formula for entropy is -

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

Where $p_i$ is the frequentist probability of an element/class $i$ in out data.

Let's say we have only two classes, a positive and a negative class. Out of 100 data, suppose that 70 belongs to $-$ve class and 30 to $+$ve. Then, P$+$ will be 0.3 and P$-$ will be 0.7. Entropy $E$ will be given by:

$$E = -\frac{3}{10} \times log_2 \frac{3}{10} - \frac{7}{10} \times log_2 \frac{7}{10} \approx \mathbf{0.88} \tag{2}$$

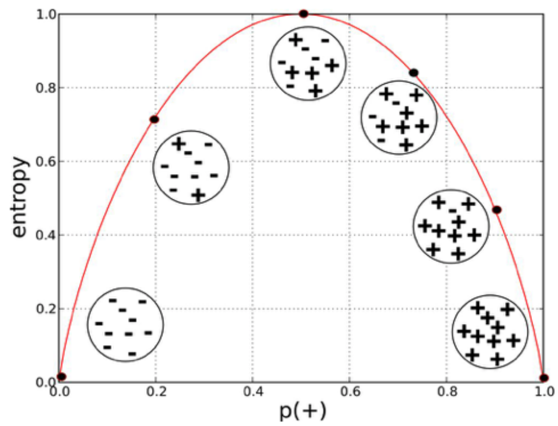



Figure 4: Entropy distribution frequentist probability

## Information Gain

Information gain is basically how much Entropy is removed after training a decesion tree.
**Higher information gain = more entropy removed.**
In technical terms, Information Gain from X on Y is defined as:

$$IG(Y, X) = E(Y) - E(Y|X) \tag{3}$$

Basic of inforation gain is well explained here: A Simple Explanation of Information Gain and Entropy