Decision Trees

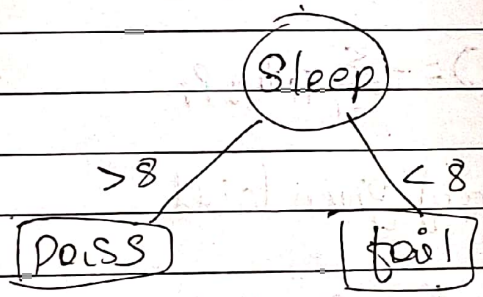
→ Decision trees basically work like how humans make decisions

→ Suppose take an example that we should go check whether we are going to pass the exam this weekend or not

→ For checking that let's take some factors or features into consideration

i) Sleep

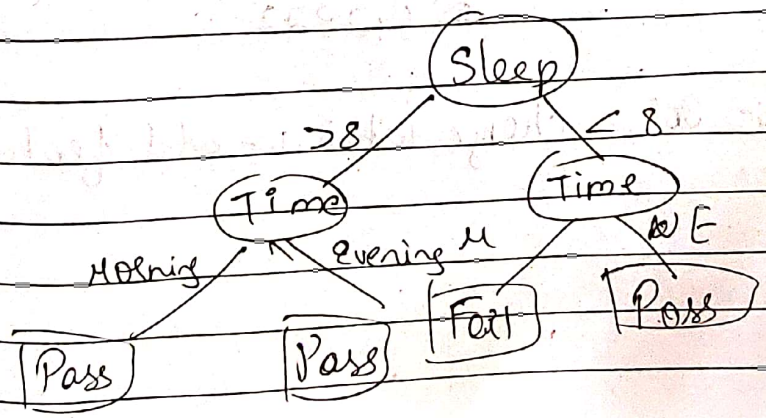
⇒ If sleep is more than 8 hours then pass or else fail



Let's add one more feature

ii) Exam time

⇒ If the exam is in evening then pass or else fail if we have less than 8hrs sleep



→ Building optimal tree is an NP-Complete, which means there is no computationally efficient way to determine an optimal tree. It always involves some sort of greedy algorithm

→ So building a tree with local optimal considerations will give a benifit of optimization procedure

→ Now how to select specific features.
Features are picked on a concept called Information gain.

→ For that 1st we define the entropy of tree.

$$\text{Entropy}(T) = -\sum_{i=1}^{2} p_i \log p_i$$

p_i denotes fraction of given label.

Suppose in our Exam there 20 students. Out of which 12 passed & 8 didnot pass. So the entropy would be

$$\text{Entropy}(T_{\text{original}}) = -\frac{12}{20}\log\frac{12}{20} - \frac{8}{20}\log\frac{8}{20}$$

$$\approx 0.2922$$

Now how does this value change when we add features

⇒ Let's Say we have added sleep factor

1st Case :- More than 8 hours sleep.

⇒ Here out of 20 Consider 10 Students

⇒ In these 10 Students 9 passed and 1 failed.

then $Entropy(split_{>8hours}) = \frac{-9}{10}\log\frac{9}{10} - \frac{1}{10}\log\frac{1}{10}$

$\simeq 0.1412$

2nd Case :- Less than 8hours

⇒ Here 7 students of 10 failed

⇒ only 3 passed

then $Entropy(split_{<8hours}) = \frac{-7}{10}\log\frac{7}{10} - \frac{3}{10}\log\frac{3}{10}$

$\simeq 0.265$

⇒ Now Information gain is Calculated as

$$Infogain(T,f) = Entropy - Entropy(T/f)$$

Here T is Original tree & f is Considering feature

$$= 0.2922 - \frac{10}{20} \times 0.265 - \frac{10}{20} \times 0.1412$$

$$\simeq 0.0891$$

⇒ we will be continuing this process until we get information gain as 0. and then we will Consider those features which are included and build the decision tree.

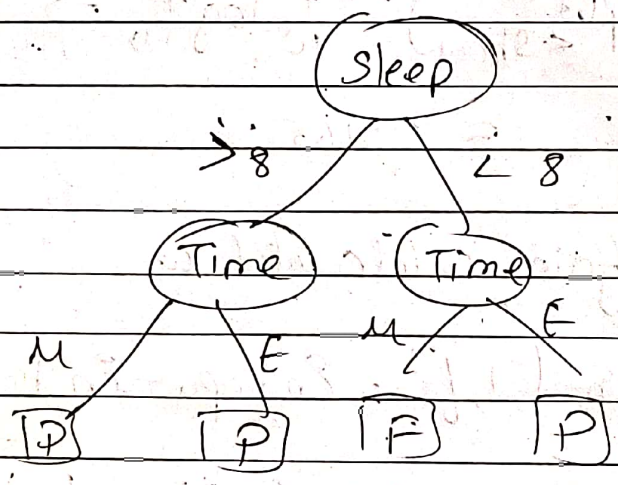
Pruning the Tree

→ The major issue with decision trees is that they learn more on training data and less on new data points

→ So to avoid this there is a procedure called reduced error pruning.

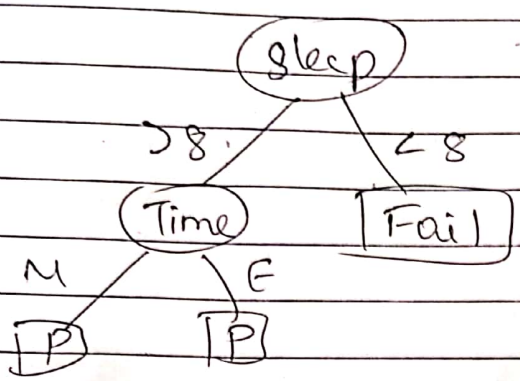
which means the tree iteratively replaces each node with its most popular label until it does not affect the accuracy of tree.

Suppose



Here the labels for less than 8 hours and ~~evening~~ morning are 9 Fail & 1 pass. where less than 8 hours and ~~morning~~ evening are 1 Fail and 3 pass. Here the occurence of Fail label is more. So replace this node with Fail which is called pruning

This should be done until it does not worsen the performance.



Pruned Tree