DECISION TREE

Decision Tree Classifier

Q What is a decision Tree?

Representation of Classification

The root of the tree which is the top most node is known as the root node. Branch is the connection between the nodes. Leaf is the end of the tree. Here in the tree, the root is the root node.

				← Target

Target attribute predicted by the decision tree using the features.

Why is a decision tree preferred as predictor?

Pure & Impure split:

All leaf is 0 or 1 means homogeneous

If some leaf nodes have both 0 and 1 then it is called impure split.

Similarly, we create a new feature if continuous which we get a pure split.

Q How do we calculate the purity of the split?

Q1 Entropy - The uncertainty in data.

Q2 Gini Impurity - How often we misclassify random element get misclassified.

Q3 How are the features selected? Information Gain - The reduction in entropy achieved after splitting a node.

Entropy:

Gini Impurity

$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$

Entropy on uncertainty is 0 & pure node

p_1 : Probability of the 1st class

p_2 : Probability of the 2nd class

For eg $p_1 = \frac{1}{2}$, $p_2 = \frac{1}{2}$

Entropy $H(X) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$

$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$

$= 1$

So, Entropy ranges from 0 to 1

For a pure split Entropy = 0

FOR: Pick up points Y or N in right Entropy → Impure split

But which feature to choose to split?

We use Information Gain:

$Gain(S, F_1) = H(S) - \underbrace{\sum_{v \in F_1} \frac{|S_v|}{|S|} H(S_v)}_{\text{Feature } F_1}$

$H(S)$ is the Entropy of the node.

S_v : Sample at the node

$|S|$: Total No of samples

F_1 : Feature

$H(S_v)$: Entropy of S_v

$H(S) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$

Entropy $H(X) = -\frac{p}{n}\log_2\frac{p}{n} - \frac{n}{n}\log_2\frac{n}{n}$

$= -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.94$

$H(S_{Weak}) = -\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8} = 0.811$

$H(S_{Strong}) = -\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6} = 1$

$H(S_{Weak}) = -\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6} = 1$

$H(S_{Strong}) = -\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8} = 0.811$

$G(S, Wind) = 0.94 - \left[\frac{8}{14} \times 0.811 + \frac{6}{14} \times 1\right] \approx 0.048$

$G \rightarrow$ No of samples at S_v

$|S| \rightarrow$ No of samples at root node

$H(S_v) \rightarrow$ Entropy at S_v

$G \rightarrow$ No of samples at S_v

$|S| \rightarrow$ No of samples at root node

$H(S_v) \rightarrow$ Entropy at S_v

We have only done for F_1

Say F_1 has a gain of 0.052

$Gain(S, F_1) > Gain(S, F_2) \Rightarrow$ we start with feature 1

The Information Gain of each feature is calculated in this manner.

Coming back to Gini Impurity:

Gini I always turn 0-0.5

Which feature to choose to split at the first node decided by Info Gain.

What is Info Gain in terms of Entropy:

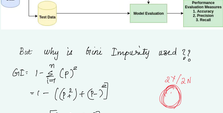
It is the reduction in Entropy achieved by splitting the dataset based on feature A.

High Info Gain ⇒ feature A splits the dataset into subsets that are more pure

$IG(S) = H(S) - H(S|A)$

Info Gain $HC(S/A)$ is low ⇒ Entropy on uncertainty is low ⇒ Much more purer split

The subset that gives the minimum Gini Index for that chosen is selected as splitting attribute



But why is Gini Impurity used?

$GI = 1 - \sum_{i=1}^{n} (p_i)^2$

$= 1 - \left[(\frac{1}{2})^2 + (\frac{1}{2})^2\right]$

$= 1 - \left[(\frac{1}{2})^2 + (\frac{1}{2})^2\right]$

$= 1 - \left(\frac{1}{2}\right) = 0.5$

When to use GI and Entropy?

GI used because computationally easier. Entropy has log factor is simple math in Gini.

Gini favors than Entropy.

used when large no. of features

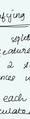
Decision Tree Regressor

f1 f_2 o/p (say a continuous feature)

Entire mean of o/p is calculated. Loss function used is not GI or Entropy. It uses Mean Squared Error or Mean Absolute Error.

$MSE = \frac{1}{2m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2$

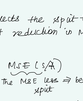
Based on say f1, it will try to compute a mean value for the o/p & then calculate MSE. Then it go ahead perform splitting



The splitting continues until the loss reduces

The mean value at the leaf node will be our output.

f1	o/p
c_1	20
c_2	22
c_3	26
	28
	30



Mean changes of each node & its does MSE

Clarifying:

When splitting a node in the tree using a feature 'A', the dataset S is divided into 2 subsets

Instances where threshold ≤ A & > A

For each split MSE of the 2 sets is calculated.

The decision tree selects the split that results in greatest reduction in MSE

Or MSE gain

i.e $\Delta MSE = MSE - \underbrace{MSE(S/A)}_{\text{the MSE low} \Rightarrow \text{better split}}$

The feature & threshold that maximise ΔMSE are chosen for the split

Hyperparameters:

① Post pruning

② Pre pruning

Post pruning: we look at the decision trees formed & decide to cut off the branch

In this case we can perform post pruning & cut off the rest of the branch

For prepruning we can set the max depth / max leaves & pre prune.