DECISION TREE

Assumption

Feature Independence: Decision trees assume that features used for splitting nodes are independent or have weak interdependencies. If there are strong correlations between features, it might lead to redundant splits and less accurate trees.

Information Gain/Gini Impurity: Many decision tree algorithms use information gain or Gini impurity as the criteria for selecting the best feature to split at each node. These criteria assume that reducing entropy or impurity leads to better splits and, eventually, more accurate predictions.

Binary data:Another assumption of decision trees is that the splits are binary. This means that each split in the decision tree separates the data into two subsets based on the values of a single input feature.

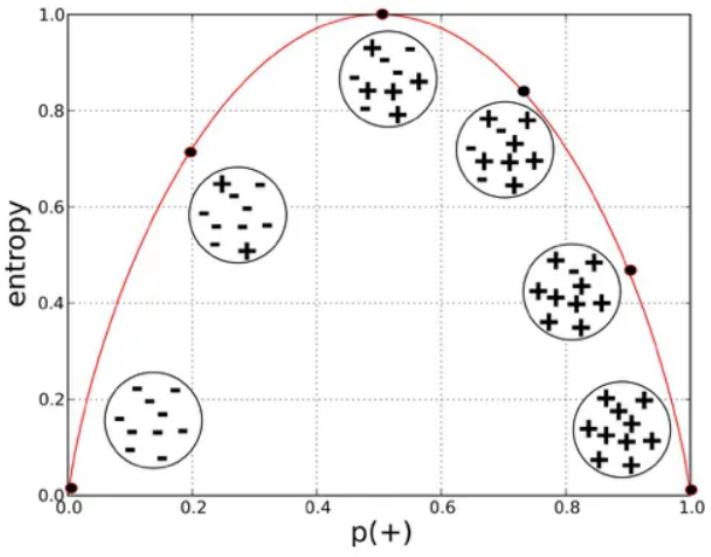
Principles

Entropy: Defined as the randomness or measuring the disorder of the information being processed in Machine Learning. Further, in other words, we can say that entropy is the machine learning metric that measures the unpredictability or impurity in the system.

Mathematically

$$L_{cross}(R) = -\sum_{c} \hat{p}_c \log_2 \hat{p}_c$$

\



We can see when p=0.5 E(S) is maximum hence we need more information about the system to classify.

Information gain:

It shows net change in entropy if a data is divided into two sub data.

$$Gain = E_{parent} - E_{children}$$

Working

Splitting

We select a Node and divide the dataset into child dataset with a splitting feature and a threshold.
Formally

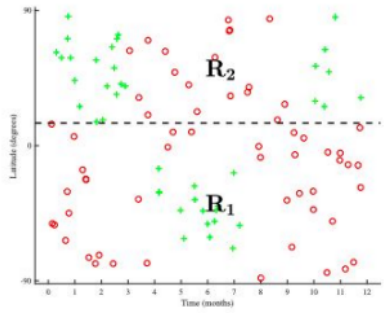
$$R_1 = \{X \mid X_j < t, X \in R_p\}$$
$$R_2 = \{X \mid X_j \geq t, X \in R_p\}$$

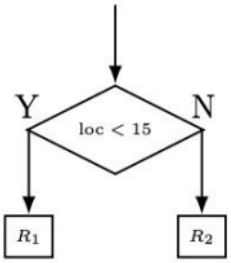
For this we select the best threshold and feature ,so we define a loss function and split which minimises that loss function is what we choose.
Loss function we choose here is Information gain. We try to maximise this Information gain.

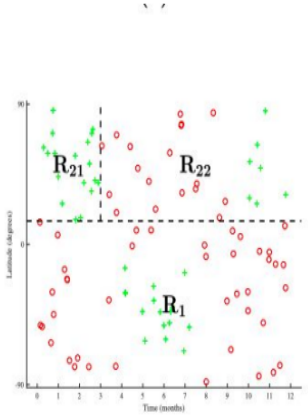
$$L(R_p) - \frac{|R_1|L(R_1) + |R_2|L(R_2)}{|R_1| + |R_2|}$$

We took the weighted mean of children.

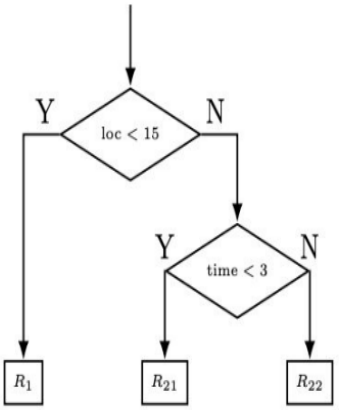
We split when we get max L(Rp) .
We build both left and right using this algorithm recursively.

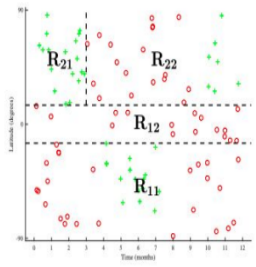


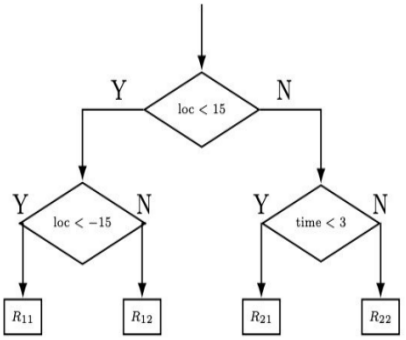




(b)







We have several conditions to stop making tree.

a) If we get a pure dataset with only one time of category in it we call it a leaf.
b) Min_split: If size of after split is less than this value we don't split and assign value by seeing max category in that node.
c) Max_depth: If we reach a limit in terms of height of tree we stop.
d) Min_gain: if Max(information gain) is less than this value after splitting we don’t split.

Predictions:

When we get a data with some feature value we predict by:
a)start from the root node, see its feature,condition of splitting and go to left ,right accordingly.
b)In the next node do the same process again until reaching a leaf.
c) if the leaf is pure we assign that label or the highest frequency category in the leaf

.