Entropy vs Gini Impurity: (Which to use When)

Whenever dataset small (1000, 2000 records) → Use Entropy

Whenever dataset large (1M, 100000, more) → Use Gini Impurity

~~Decsin~~

Decision Tree for Numerical Split:

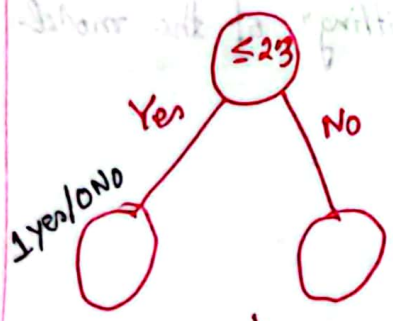
In category features we found the pure split and impure split by counting the target feature values (Yes/no). But what should we do with a continuous feature?

Example:

f1	O/P
2·3	Yes
3·6	Yes
4	No
5·2	No
6·7	Yes
7·8	No
9·0	Yes

→ Approach

1) Sort the feature values
2) Set threshold value
3) Take the minimum value as threshold and split (~~2·3~~) (≤ 2·3)
4) Then check purity with entropy
5) Then do further split based on another feature
6) Then calculate total Information gain
7) (2–6) will be continued for next values (3·6, 4, 5·2 ...)



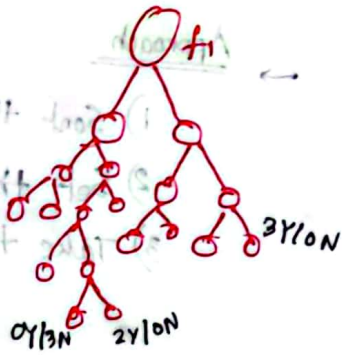
→ Then calculate entropy to check purity

So, we will get information split from every f1 data values, the greater the gain value, that value we will consider.

Problem: When we have a large dataset of millions of records, Time complexity is extremely high to make decision tree for each continous value and check information gain.

Post Pruning and Pre Pruning Decision Trees:

In decision Tree We keep splitting all the nodes till we get all the left node means till the level where every node is a pure split



But there is a big problem in it. When we split our training data like this, it usually leads to "Over fitting" of the model.

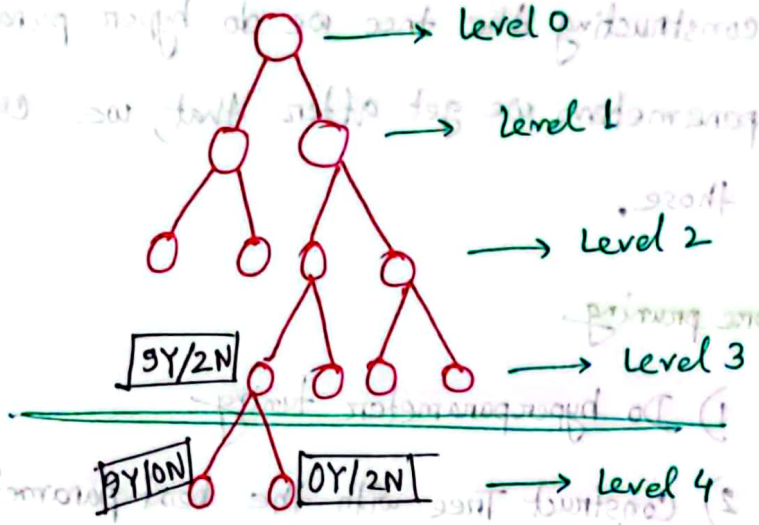
Train data Acc ↑↑

Test data Acc ↓↓

To increase the accuracy of Test ~~am~~ data, we use the two techniques

→ Post Pruning
→ Pre Pruning (Reduce Overfitting)

① Post Pruning



We can see that, in level 3, one node has 9 Yes and 2 No. So the probability we we are getting hen Yes is very high because of the high ratio so, we don't need to further split to find pure split as we alsready found the greater probability.

"Steps in post pruning":

1) Construct Decision Tree
2) Use maxDepth, feature numbers, paramaters of scikit learn to prune the tree (To cut the branches)

In post pruning we take the hyper parameter tuning MaxDepth.

② Prepruning:

In this case we first use hyper parameter tune in the model and it already says how much depth the tree should, what should be the parameters, sample size, features etc.! So before constructing the tree we do hyper parameter tuning and the parameters we get after that, we construct the tree using those.

Steps for pre pruning

1) Do hyperparameter tuning
2) Construct Tree with the best parameters

Which one to use when?

→ For "small dataset", use post Pruning

→ For "large Dataset", Use. Pre Pruning