

ID3 Algorithm

1. Start with the entire dataset
2. Check if the dataset is pure
 - If all instances in the dataset have the same class label, return that class as the result (leaf node).
3. Check if the dataset is empty.
 - If empty, return the most frequent class from parent node (for handling missing data)
4. Check if no more features to split
 - If no, return the most frequent class in the dataset (leaf node)
5. Calculate Information gain.

IG =

$$IG(D, A) = Entropy(D) - \sum_{v \in \text{Values}(A)} \left(\frac{|D_v|}{|D|} \times Entropy(D_v) \right)$$

where:

D - dataset

A - feature.

 D_v - Subset of dataset where feature A has value v. $\text{Values}(A)$ unique values of feature A.

6. Choose the feature with the highest Information Gain.
 - feature with highest information gain is selected as the splitting criterion for the current node.
7. Split the dataset based on the selected feature
 - Create a branch for each unique value of the selected feature and partition into subsets.

8. Re recursively apply ID3 to each subset
→ apply to each subset until one of the
stopping conditions are met (pure, dataset empty,
no feature left)

9.

Prune the tree

→ It can be pruned by removing branches
that don't add value.

10/3/24