Decision Tree

Decision tree is one of the predictive modelling approaches used in statistics, data mining and machine learning.

Decision Trees are a non-parametric **supervised learning** method used for both **classification**and **regression**tasks.

Tree models where the target variable can take a discrete set of values are called **classification trees**.

Decision trees where the target variable can take continuous values (typically real numbers) are called **regression trees**.

**Approach to make decision tree**

**Information Gain:**

Information gain is used to decide which feature to split on at each step in building the tree. Simplicity is best, so we want to keep our tree small.

A commonly used measure of purity is called information. For each node of the tree, the information value **measures how much information a feature gives us about the class. The split with the highest information gain will be taken as the first split and the process will continue until all children nodes are pure, or until the information gain is 0.**

**Gini Impurity:**

**Pure**

Pure means, in a selected sample of dataset all data belongs to same class (PURE).

**Impure**

Impure means, data is mixture of different classes**.**

**Definition of Gini Impurity**

Gini Impurity is a measurement of the likelihood of an incorrect classification of a new instance of a random variable, if that new instance were randomly classified according to the distribution of class labels from the data set.

If our dataset is Pure then likelihood of incorrect classification is 0. If our sample is mixture of different classes then likelihood of incorrect classification will be high.

Steps for Making decision tree

* Get list of rows (dataset) which are taken into consideration for making decision tree (recursively at each nodes).
* Calculate uncertanity of our dataset or Gini impurity or how much our data is mixed up etc.
* Generate list of all question which needs to be asked at that node.
* Partition rows into True rows and False rows based on each question asked.
* Calculate information gain based on gini impurity and partition of data from previous step.
* Update highest information gain based on each question asked.
* Update best question based on information gain (higher information gain).
* Divide the node on best question. Repeat again from step 1 again until we get pure node (leaf nodes).

**Advantage of Decision Tree:**

* Easy to use and understand.
* Can handle both categorical and numerical data.
* Resistant to outliers, hence require little data preprocessing.

**Disadvantage of Decision Tree**

* Prone to overfitting.
* Require some kind of measurement as to how well they are doing.
* Need to be careful with parameter tuning.
* Can create biased learned trees if some classes dominate.

**How to avoid overfitting the Decision tree model**

Overfitting is one of the major problem for every model in machine learning. If model is overfitted it will poorly generalized to new samples. To avoid decision tree from overfitting **we remove the branches that make use of features having low importance.**This method is called as **Pruning or post-pruning.**This way we will reduce the complexity of tree, and hence imroves predictive accuracy by the reduction of overfitting.

**Cross Validation:**  Cross-validation is a [resampling](https://en.wikipedia.org/wiki/Resampling_(statistics)) method that uses different portions of the data to test and train a model on different iterations. It is mainly used in settings where the goal is prediction, and one wants to estimate how [accurately](https://en.wikipedia.org/wiki/Accuracy) a [predictive model](https://en.wikipedia.org/wiki/Predictive_modelling) will perform in practice. In a prediction problem, a model is usually given a dataset of known data on which training is run (training dataset), and a dataset of unknown data (or first seen data) against which the model is tested (called the [validation dataset](https://en.wikipedia.org/wiki/Validation_set) or testing set). The goal of cross-validation is to test the model's ability to predict new data that was not used in estimating it, in order to flag problems like [overfitting](https://en.wikipedia.org/wiki/Overfitting) or [selection bias](https://en.wikipedia.org/wiki/Selection_bias)

**How to avoid overfitting the Decision tree model**

Overfitting is one of the major problem for every model in machine learning. If model is overfitted it will poorly generalized to new samples. To avoid decision tree from overfitting **we remove the branches that make use of features having low importance.**This method is called as **Pruning or post-pruning.**This way we will reduce the complexity of tree, and hence imroves predictive accuracy by the reduction of overfitting.

Pruning should reduce the size of a learning tree without reducing predictive accuracy as measured by a [cross-validation](https://en.wikipedia.org/wiki/Cross-validation_(statistics)) set. There are 2 major Pruning techniques.

* Minimum Error: The tree is pruned back to the point where the cross-validated error is a minimum.
* Smallest Tree: The tree is pruned back slightly further than the minimum error. Technically the pruning creates a decision tree with cross-validation error within 1 standard error of the minimum error.

## **Early Stop or Pre-pruning**

An alternative method to prevent overfitting is to try and stop the tree-building process early, before it produces leaves with very small samples. This heuristic is known as early stopping but is also sometimes known as pre-pruning decision trees.

At each stage of splitting the tree, we check the cross-validation error. If the error does not decrease significantly enough then we stop. Early stopping may underfit by stopping too early. The current split may be of little benefit, but having made it, subsequent splits more significantly reduce the error.