# Decision Trees

* It is a supervised multi-class classifier
* It can be used for both classification (if we have to predict categorical variable) and regression (if we have to predict real number). Widely used in classification.
* In this technique we split the population into two or more homogeneous sets based on most significant splitter using input variables:
* This is also called a rule based engine. Each branch is a rule. If age > 20 **AND** gender = male **OR** stays = city **THEN** he will churn.
* **Gini Index**: measures the quality of split
* Measures the amount of randomness in the data.
* Minimizes the misclassification, Tend to find the largest class.
* Gini index for a node is sum of squares of probability of the class.
* **Entropy**: amount of uncertainty in the data
* **Measure of information gain, how much information is explained by each variable**
* Finds the group of classes that make up less than 50% of the data.
* Entropy for exploratory analysis.
* **Pruning:** to avoid over fitting. We allow the tree to grow completely first and then prune according to the cross-validation results.

# Random forest

* It is a supervised multi-class classifier. A black box model.
* It can be used for both classification and regression. Widely used in classification.
* Implements ensemble: method for combining weak learners in an attempt to produce strong learners. (eg: 1 ant cannot kill an elephant, but 1000 ants can). Here week learners are decision trees (multiple decision trees are built).
* In Random Forest, we grow multiple trees as opposed to a single tree as in decision tree model. Each tree will give an output. The majority of that output is considered as classification, Average in regression.
* In DT, we built model using **whole dataset**, considering all features. Where as in RF, we build model using fraction of total number of rows is selected random and a particular feature are selected at a random to train on a dataset.

# Gradient Boosting:

* Here we build a tree by giving equal weights to all the variables. There will be some error in the model/tree.
* Now that error is learnt or rectified by changing the parameters to boost the performance. Now even this tree will have some error.
* This process is performed multiple times until we have a negligible error. This process is called bootstrapping.

# Extended Gradient Boosting:

* Implements regularization techniques L1 and L2 (is a technique used in an attempt to solve the overfitting problem in statistical models)
* Built in cross-validation, Handles missing values, Automatic tree pruning

**Model performance: for all the models discussed above**

* We will perform cross-validation, split the data into train and test datasets (70-30). Build a model on training dataset and check the training accuracy.
* Now apply the model on test dataset (using predict function). Check test accuracy.
* If the train and test accuracy is not satisfactory, then change the model parameters and build new models.
* Continue this step until u get a decent train and test accuracies.(By doing this you will not over fit the model on train dataset)
* Also check precision, recall, F1 score and Support.
* Finally we can check explained variance value (best is 1) and mean absolute error (best is 0)

**NOTE: This should be performed for all the models. K- Fold validation can also be performed.**