Decision Tree

Using the decision algorithm, we start at the tree root and split the data on the feature that results in the **largest information gain (IG).**In an iterative process, we can then repeat this splitting procedure at each child node until the leaves are pure. **We typically prune the tree by setting a limit for the maximal depth of the tree to prevent overfitting.**

**Maximizing information gain – getting the most bang for the buck**

Our objective function is to maximize the information gain at each split, which we define as follows:



f is the feature to perform the split, Dp and Dj are dataset of the parent and jth chile node, and Nj is the number of samples in the jth child node. As we can see, the information gain is simply the difference between the impurity of the parent node and the sum of the child node impurities – the lower the impurity of the child nodes, the larger the information gain. However, for simplicity and to reduce the combinatorial search space, most libraries implement binary decsiton trees. This means that each parent node is split into two child nodes, Dleft and Dright:



The three impurity measures or splitting criteria that are commonly used in binary decision decision tree are Gini index (Ig),entropy (Ih) and the classification error (Ie).

1. Entroy :



Here ,  is the proportion of the samples that belongs to class c for a particular node t. The entropy is therefore 0 if all samples at a node belong to the same class, and the entropy is maximal if we have a unfiormal class distribution.

For example, in a binary class setting, the entropy is 0 if p (i=1|t ) = 1 or p (i=0|t ) = 0 . If the classes are distributed uniformly with p(i=1|t ) = 0.5 and p (i=0|t )= 0.5, the entropy is 1. Therefore, we can say that the entropy criterion attempts to maximize the mutual information  
in the tree.

1. Gini index: can be understood as criterion to minimize the probability of misclassification:

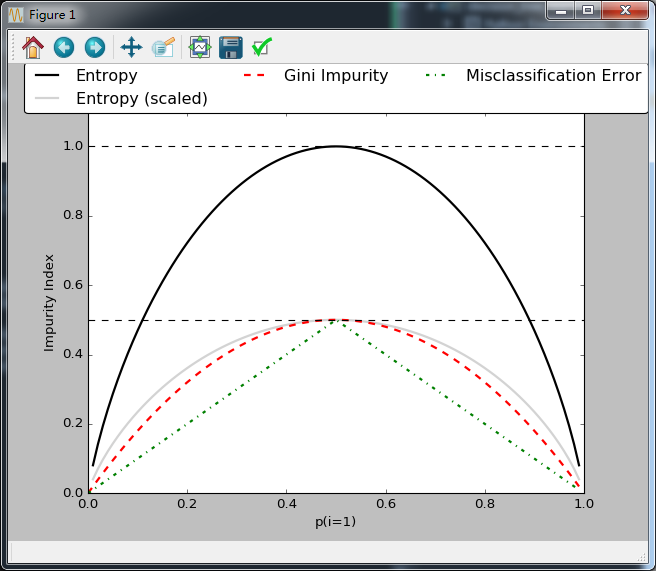


Similar to entropy, the Gini index is maximal if the classes are perfectly mixed.

1. Classification error:



This is a useful criterion for pruning but not recommended for growing a decision three, since it is less sensitive to changes in the class probabilities of the nodes.



**Building a decision tree**

The deeper the decision tree, the more complex the decision boundary becomes, which can easily result in overfitting.

