Decision Tree

The purpose of this exercise is to implement a decision tree growing algorithm and apply it to the data set of US congressional voting records. The task is to learn to predict party affiliation (Democrat/Republican) from the votes on a number of issues. The data is divided into the training set and the test set.

The algorithm is as follows:

- For current node \( N \)
  - If all training examples classified correctly then
    * Stop
  - Otherwise
    * Compute the splitting criterion for each attribute
    * Split the current node according to the attribute which maximizes the splitting criterion
    * Recursively apply to each newly created child node \( N_i \)

As a node-splitting criterion you can use information gain, that is the difference in class entropy between the current node and the weighted sum of entropies of the proposed child nodes:

\[
IG(N|A) = H(N) - \sum_{i \in A} \frac{|N_i|}{|N|} H(N_i) \tag{1}
\]

where \( N \) is the current node, \(|N|\) is its size, \( i \) is one of the possible values of attribute \( A \), and \( N_i \) is a child node such that all the training examples it contains have value \( i \) for attribute \( A \). The entropy of a node is the entropy of the probability distribution of the class labels of the examples the node contains.

After growing the tree on the training set, apply it to the test set and report the error rate. Try pruning the tree size at different depth levels (that is remove nodes below that level). Again report error rates on the test set using those pruned trees.

Please send your solutions by Wednesday Jan 22 to psr-tutorial@lsv.uni-saarland.de

Important: Please use PDF as a document format. If you need to compress files, use ZIP or GZIP.