

AN OVERVIEW OF THE CART METHODOLOGY

Introduction

CART® is a robust data-mining and data-analysis tool that automatically searches for important patterns and relationships and quickly uncovers hidden structure even in highly complex data. This discovered knowledge is then used to generate accurate and reliable predictive models for applications such as profiling customers, targeting direct mailings, detecting telecommunications and credit-card fraud, and managing credit risk.

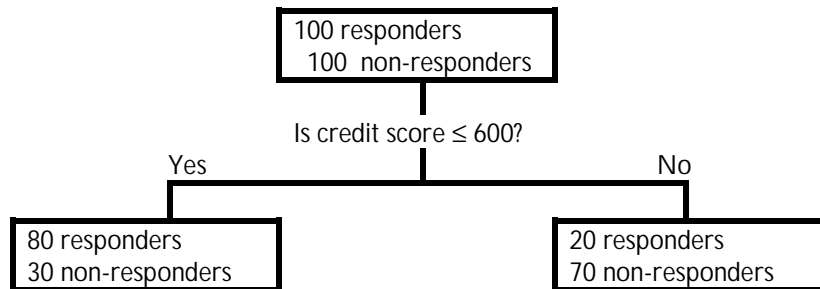
CART uses an intuitive Windows-based interface, making it accessible to both technical and non-technical users. Underlying the “easy” interface, however, is a mature theoretical foundation that distinguishes CART from other methodologies and other decision-tree tools.

The CART methodology is technically known as binary recursive partitioning. The process is binary because parent nodes are always split into exactly two child nodes and recursive because the process can be repeated by treating each child node as a parent. The key elements of a CART analysis are a set of rules for:

- splitting each node in a tree,
- deciding when a tree is complete, and
- assigning each terminal node to a class outcome (or predicted value for regression).

Splitting Rules

To split a node into two child nodes, CART always asks questions that have a “yes” or “no” answer. For example, the question “Is credit score ≤ 600 ?” splits the tree’s root, or parent, node into two branches with “yes” cases going to the left child node and “no” cases to the right.



How do we come up with candidate splitting rules? CART’s method is to look at all possible splits for all variables included in the analysis. For example, consider a data set with 215 cases and 19 variables. CART considers up to 215 times 19 splits for a total of 4085 possible splits. Any problem will have a finite number of candidate splits and CART will conduct a brute force search through them all.

Choosing a Split

CART's next activity is to rank order each splitting rule on the basis of a quality-of-split criterion. The default criterion used in CART is the GINI rule, essentially a measure of how well the splitting rule separates the classes contained in the parent node. Five alternative criteria are also available for classification trees and two criteria for regression trees. In addition, to deal more effectively with select data patterns, CART also offers splits on linear combination of continuous predictor variables.

Class Assignment

Once a best split is found, CART repeats the search process for each child node, continuing recursively until further splitting is impossible or stopped. Splitting is impossible if only one case remains in a particular node or if all the cases in that node are exact copies of each other (on predictor variables). CART also allows splitting to be stopped for several other reasons, including that a node has too few cases. (The default for this lower limit is 10 cases, but may be set higher or lower to suit a particular analysis).

Once a terminal node is found we must decide how to classify all cases falling within it. One simple criterion is the plurality rule: the group with the greatest representation determines the class assignment (*e.g.*, in the sample decision tree above, the case assignment for the left child node is responders and non-responders for the right child node.). CART goes a step further: because each node has the potential for being a terminal node, a class assignment is made for every node whether it is terminal or not. The rules of class assignment can be modified from simple plurality to account for the costs of making a mistake in classification and to adjust for over- or under-sampling from certain classes.

A common technique among the first generation of tree classifiers was to continue splitting nodes (growing the tree) until some goodness-of-split criterion failed to be met. When the quality of a particular split fell below a certain threshold, the tree was not grown further along that branch. When all branches from the root reached terminal nodes, the tree was considered complete. While this technique is still embodied in several commercial programs, including CHAID™ and KnowledgeSEEKER™, it often yields erroneous results. CART uses a completely different technique.

Pruning Trees

Instead of attempting to decide whether a given node is terminal or not, CART proceeds by growing trees until it is not possible to grow them any further. Once CART has generated a "maximal tree," it examines smaller trees obtained by pruning away branches of the maximal tree. Unlike other methods, CART does not stop in the middle of the tree-growing process, because there might still be important information to be discovered by drilling down several more levels.

Testing

Once the maximal tree is grown and a set of sub-trees are derived from it, CART determines the best tree by testing for error rates or costs. With sufficient data, the simplest method is to divide the sample into learning and test sub-samples. The learning sample is used to grow an overly-large tree. The test sample is then used to estimate the rate at which cases are misclassified (possibly

adjusted by misclassification costs). The misclassification error rate is calculated for the largest tree and also for every sub-tree. The best sub-tree is the one with the lowest or near-lowest cost, which may be a relatively small tree.

Some studies will not have sufficient data to allow a good-sized separate test sample. The tree-growing methodology is data intensive, requiring many more cases than classical regression. When data are in short supply, CART employs the computer-intensive technique of cross validation.

Cross Validation

Cross validation is used if data are insufficient for a separate test sample. In such cases, CART grows a maximal tree on the entire learning sample. This is the tree that will be pruned back. CART then proceeds by dividing the learning sample into 10 roughly-equal parts, each containing a similar distribution for the dependent variable. CART takes the first 9 parts of the data, constructs the largest possible tree, and uses the remaining 1/10 of the data to obtain initial estimates of the error rate of selected sub-trees. The same process is then repeated (growing the largest possible tree) on another 9/10 of the data while using a different 1/10 part as the test sample. The process continues until each part of the data has been held in reserve one time as a test sample. The results of the 10 mini-test samples are then combined to form error rates for trees of each possible size; these error rates are applied to the tree based on the entire learning sample.

The upshot of this complex process is a set of fairly reliable estimates of the independent predictive accuracy of the tree. This means that we can know how well any tree will perform on completely fresh data—even if we do not have an independent test sample. Because the conventional methods of assessing tree accuracy can be wildly optimistic, cross validation is the method CART normally uses to obtain objective measures for smaller data sets.

Conclusion

CART uses a combination of exhaustive searches and computer-intensive testing techniques to reveal important patterns and relationships hidden in data. It can be applied to virtually any data set and can proceed with little or no guidance from the user. Thus, if you have a data set and have no idea how to proceed with its analysis, you can simply hand it over to CART and let it do the work. If this sounds too good to be true, the natural question is: does CART really deliver useful results that you can trust?

The surprising answer is a resounding yes. When automatic CART analyses are compared with stepwise logistic regressions or discriminant analyses, CART typically performs about 10% to 15% better on the learning sample. CART's performance on test samples is even more important. Because CART does not suffer from the statistical deficiencies that plague conventional stepwise techniques, CART will typically be far more accurate on new data. Further, when automatic CART analyses are compared with the best parametric models of sophisticated teams of statisticians, CART is still competitive. CART can often generate models in an hour or two that are only slightly worse in predictive accuracy than models that may take specialists several days to develop.

