Case Study: Visualization for Decision Tree Analysis in Data Mining

Todd Barlow Padraic Neville
SAS Institute Inc.
todd.barlow@sas.com *padraic.neville@sas.com*

Abstract

Decision trees are one of the most popular methods of data mining. Decision trees partition large amounts of data into smaller segments by applying a series of rules. Creating and evaluating decision trees benefits greatly from visualization of the trees and diagnostic measures of their effectiveness. This paper describes an application, EMTree Results Viewer, that supports decision tree analysis through the visualization of model results and diagnosis. The functionality of the application and the visualization techniques are revealed through an example of churn analysis in the telecommunications industry.

1. Introduction

Data mining uses a set of analytical methods for discovering previously unknown relationships in data. The relationships are often complex. Visualization facilitates understanding of complex models that arise in data mining. This paper presents visualization ideas to help users understand a type of predictive model called decision trees. The ideas have been implemented in the EMTree Results Viewer.

Decision trees are one of the most popular types of predictive models. A decision tree is created by partitioning a large data set into subsets, and then partitioning each of the subsets, until the subsets cannot be partitioned further. In keeping with the tree metaphor, the original data set is the *root node*, the subsets are *nodes*, and the unpartitioned subsets are *leaves*. *Branches* from a node are the subsets created by partitioning a node. The purpose of building a decision tree is to partition a large heterogeneous group of things (usually people) into smaller, homogeneous groups. By creating homogeneous groups, the analyst can predict with greater certainty how individuals in each group will behave. The final groups, shown as leaves in the tree, are defined by a sequence of partitioning rules.

Typically a partitioning rule uses a single variable when assigning a case to a branch. Anybody can quickly comprehend a single rule and judge whether it is sensible. Unfortunately, judging the sensibility of a sequence of simple rules is complicated, and a large tree with lots of partitions is difficult to comprehend. EMTree is designed to help analysts build and understand complex decision trees by visualizing the partitioning of cases, which helps the analyst comprehend the predictions of a model, and by visualizing model diagnostics, which helps the analyst assess the reliability of a model.

Figure 1 shows screenshots of the application with 4 of the 14 linked views available to the user. In Figure 1, window A contains the assessment plot that shows the subtrees available for analysis. Selecting a point in the plot updates the other views to show details about the selected subtree. Window B contains a compact view of the tree. It shows the tree topology and some information about the quality of the model. Window C contains the traditional tree view and some information about the model. Window D contains a list of variables in the model. Selecting a node in windows B or C highlights the node or row in the other windows. Selecting a variable in D highlights the nodes using that variable in B or C. Each of these views and their relationships will be discussed in more detail later.

2. Visualization of decision trees

In this paper, we demonstrate the visualization tools in the application through an example of *churn data analysis*, i.e., the analysis by a telecommunications company of their service subscribers to determine which subscribers should receive special offers to discourage them from switching their service to another company. Using historical usage and cancellation data, an analyst creates a decision tree that predicts the probability that a subscriber would cancel their service. The analyst then inspects the model to judge whether it makes sense, then applies the model to new validation data to judge whether the model is reliable.

Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)
1522-4048/01 $17.00 © 2001 IEEE

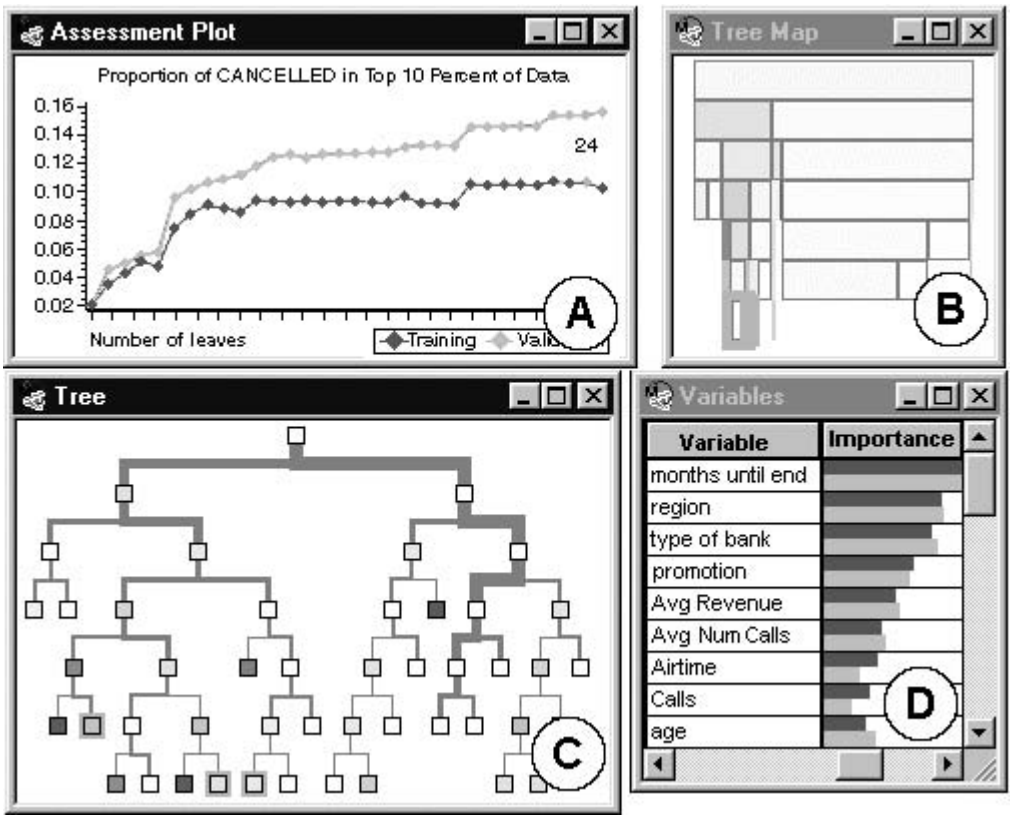

Figure 1. Four of the linked views in the decision tree application.

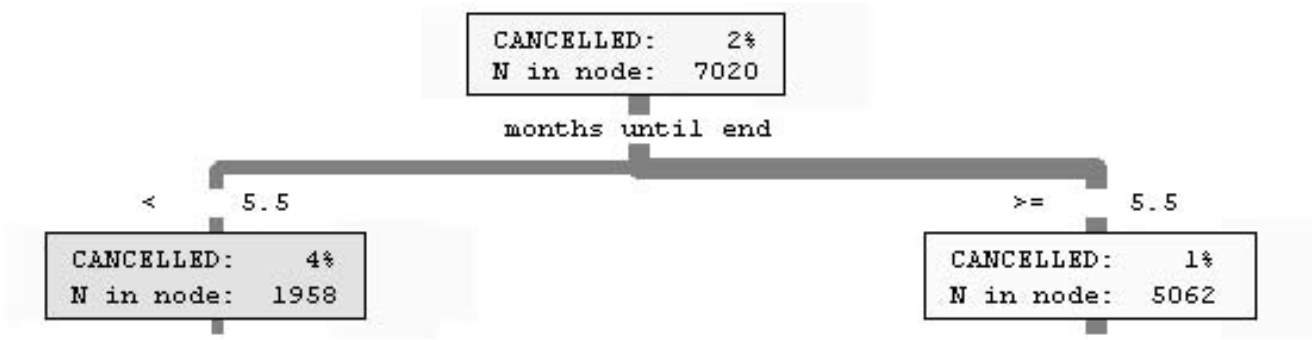

Figure 2. Root node with splitting rule and branch statistics

Figure 2 shows a view of the root node in the tree. The text in the node shows that there are 7,020 people in the training data set. Two percent of those people cancelled their service. The tree creates the first two branches by partitioning people based on the number of months remaining in the service obligation. Cancellations occur at a higher rate among people with less than 5.5 months remaining. The higher rate is indicated both by the text and by the intensity of the purple color. Choosing to color the nodes by the main statistic of interest allows the user to turn off the text and view more of the tree. Windows B and C in Figure 1 show the 24-leaf tree selected by the application. The node colors indicate that there are several leaves with concentrations of cancellations high enough to be of potential interest.

While identifying the leaves with high concentrations of the target group, the analyst also has to determine if the leaves are large enough to be useful in categorizing people. Small leaves are less interesting than large leaves because it is more cost effective and statistically reliable to target a few large groups than many small groups. In the tree, branch line thickness is proportional to the

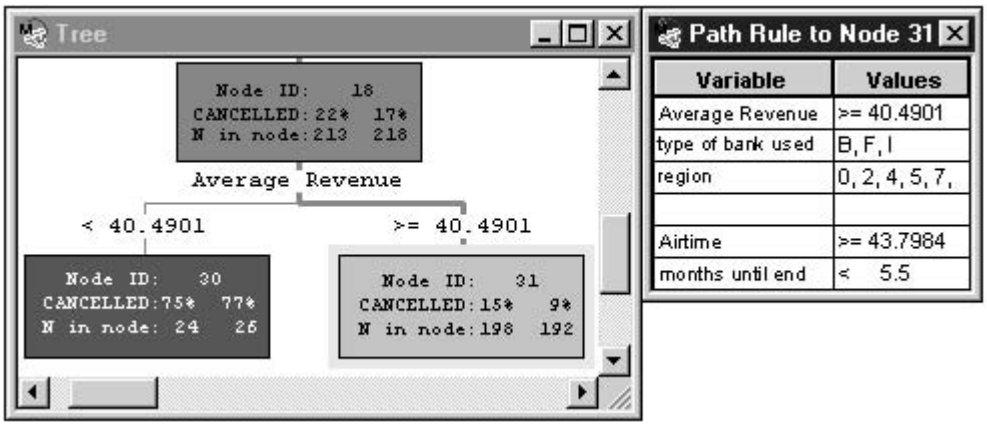

Figure 3. Tree nodes with text and rules defining selected leaf

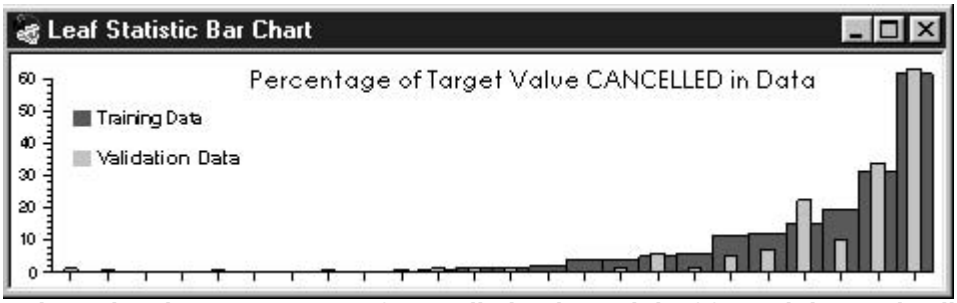

Figure 4. Bar chart showing percentage of cancellation in each leaf for training and validation data.

square root of the number of cases in each branch. The square root transformation helps distinguish differences between nodes with fewer people. This transformation is one of several available to the user when setting branch line width. The user can interactively choose which transformation works best.

Window B in Figure 1 shows a compact representation of the tree based on methods of displaying clusters [1] and hierarchies [2]. This view maintains the top-to-bottom, left-to-right orientation of the traditional tree view while representing node size through node width. The root node is at the top of the tree. It contains all the people in the data set. All other nodes contain a percentage of those people. Their width is proportional to the number of people in the node. The red lines represent nodes that contain so few people that they cannot be drawn accurately in the space allocated to the view. This view was designed to be used as a navigation tool for larger trees. The user could display the traditional tree with text in the node and navigate around the tree using the compact view. Selecting a node in the compact view moves the same node in the traditional view to the center of the window. When the traditional tree contains text, selecting nodes in the compact view is a way of navigating through the tree without having to manually scroll the traditional view's window.

The compact view also clearly displays node color and size, the two primary clues for finding subsets of interest. The selection of leaves that are large enough and have a high enough concentration of the target group is a subjective process. Business rules and statistical rules guide the judgment of the analyst. The interaction between the tree and compact tree facilitates this process. Those leaves of a useful size with interesting concentrations of canceling subscribers are selected in both views.

The analyst now might want to characterize the people represented in the selected leaves, and determine whether the isolation of this group makes sense. In this example, we can simply list the values of the variables required for a person to be in a selected leaf. These rules can be displayed as part of the tree or in a separate window. Figure 3 shows both methods of displaying rules. The rules window updates to show the rules for the selected node.

If the rules make sense, the analyst needs to determine if the model is reliable. To test the model, the rules are applied to a validation data set of 7020 cases withheld from the training process. The bar chart in Figure 4 is one

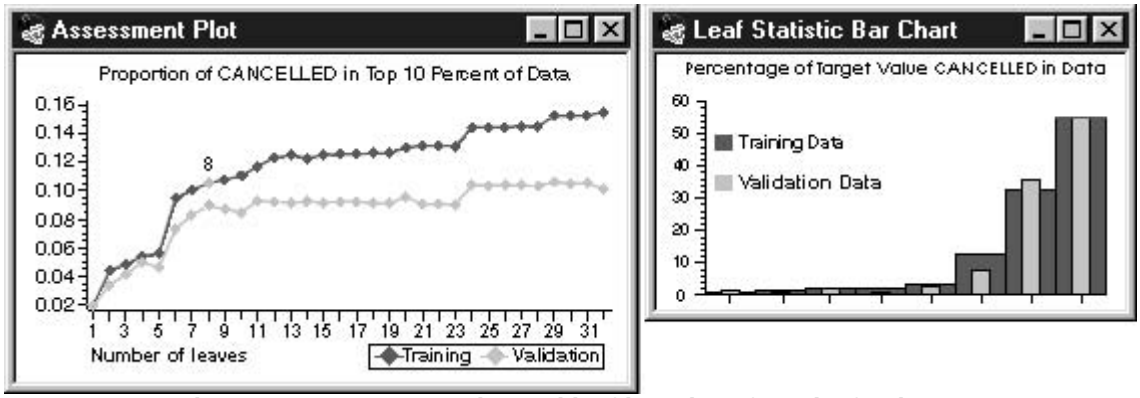

Figure 5. Assessment plot and leaf bar chart for 8-leaf subtree

of four views that compare the validation data with the training data. The height of a bar represents the proportion of cancellations in a leaf. The dark, wide bars represent the training data. The light, narrow bars represent the validation data. The leaves are arranged in ascending order of the concentration of people in the training data who cancelled. The model uses the training data to make predictions. If the model were reliable, the bars representing the validation data would increase in a manner similar to the training bars. Figure 4 shows several leaves in which the validation bar height does not match the training bar height, thereby indicating some unreliability.

A likely explanation of the poor prediction is that the model overfit the training data, possibly because nodes deep in the tree have too few observations on which to base reliable predictions. Breiman [3] and Quinlan [4] recommend pruning large trees to create smaller trees that predict more reliably. To facilitate this, the application automatically generates a series of subtrees with an increasing number of leaves. Figure 5 shows an assessment plot of subtree effectiveness verses the number of leaves. In this example, effectiveness is measured as the proportion of cancellations among the 10% of customers the tree predicts to be most likely to cancel. A user may interactively change this measure. The two curves show effectiveness on training and validation data. The separation of the curves, with the validation curve below the training curve, suggests that model reliability drops off for subtrees with more than 6 leaves. The flatness of the validation curve between 8 and 23 leaves shows that there is little improvement in predictive accuracy as the subtrees become more complex. There is a slight improvement for the subtree with 24 leaves but the curve is flat for the remaining subtree. Based on these curves, the analyst selects the subtree with 8 leaves. Selecting the point in the plot updates the other views. Figure 5 also shows the bars for a tree with only eight leaves. The validation bars increase monotonically which is consistent with reliable predictions.

3. Conclusions and Future Work

The creation and evaluation of decision trees is an iterative process. It requires visualizations of trees at varying levels of detail, diagnostic plots, and a variety of tables. Our experience and our interaction with other data analysts shows that the analysis process involves frequent switching between views, and modification of views. We have described an application, EMTree Results Viewer, designed to support these activities. EMTree Results Viewer comprises a set of linked views. These views offer a variety of ways of visualizing relevant information. They also interact so that the presentation of the information supports the understanding of relationships among the views. Interviews with users and feedback from early adopters suggest that this approach to decision tree analysis will be successful.

NOTE: The EMTree Results Viewer discussed in this paper incorporates major contributions in design and development from Jennifer Clegg, Lina Pratt, John Schroedl, and Pei-Yi Tan.

4. References

[1] B.K. Kleiner and J.A. Hartigan, "Representing Points in Many Dimensions by Trees and Castles", *Journal of the American Statistical Association*, American Statistical Association, Alexandria, VA, June 1981, 260-272.
[2] P. Dykstra, Xdu, US Army Research Laboratory, 1991. Available at http://sd.wareonearth.com/~phil/xdu/
[3] L. Breiman, J. H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees*, Wadworth, Belmont, CA, 1984.
[4] R.J. Quinlan, *Programs for Machine Learning*, Morgan Kaufmann, San Mateo, CA, 1993.