Article from

Predictive Analytics and Futurism

December 2016
Issue 14

Use Tree-based Algorithm for Predictive Modeling in Insurance

By Dihui Lai, Bingfeng Lu

Artificial intelligence (AI) has captured the attention of a broad audience recently. A creative use of an AI algorithm with "big data" could potentially bring revolutionary benefits to many industries. Deep learning, for example, has brought great success in areas like auto-drive, voice/face recognition and the ancient game Go.

Among all the AI algorithms, the decision tree has been widely used for supervised learning and it shows great capability in solving classification and regression problems. The inherent structure of a tree algorithm makes it good for addressing rule-based problems, determining similarities among the objects and classifying groups. In this article, we will overview some popular tree-based models and understand how such models may be applied to the insurance industry.

WHAT IS A TREE-BASED MODEL?

The essences of a tree model, as one may have expected, are roots, branches and leaves. A tree model always starts with a root and grows into branches. Unlike a real tree, a model tree usually only has two splits on each branch. This is also known as a binary tree. The binary structure is powerful yet easy to describe mathematically. A binary tree keeps growing through a series of yes/no questions until the leaves are reached.

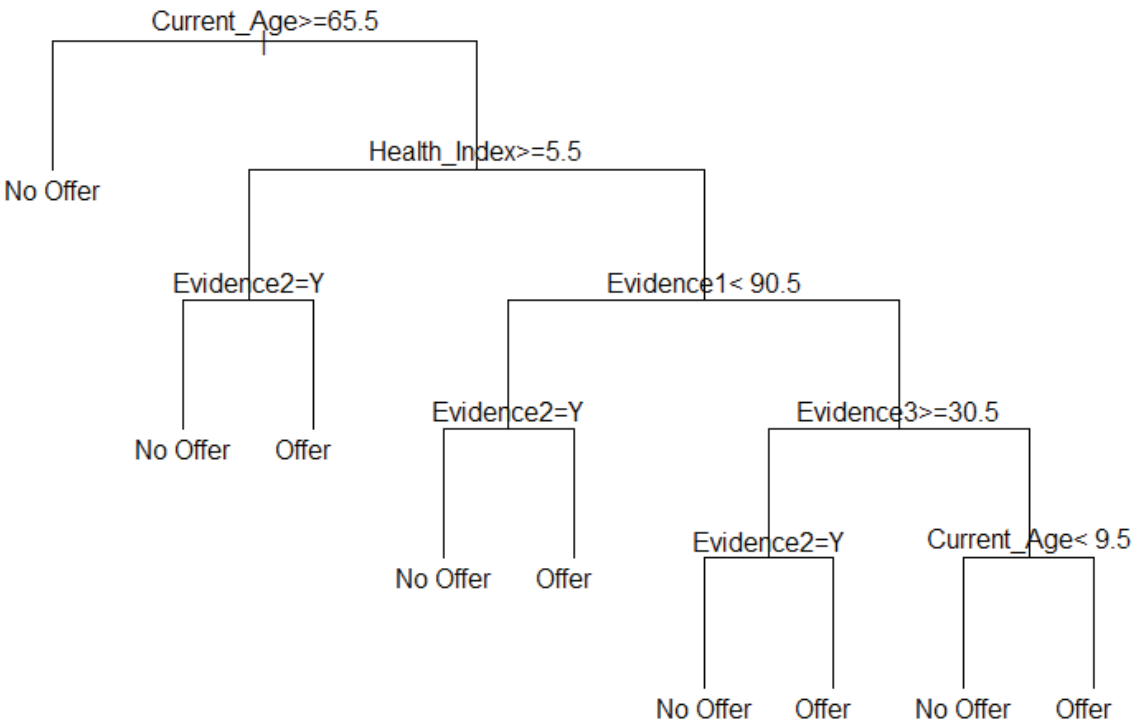
A tree-based model may contain multiple trees and form a tree ensemble. The techniques that are used for growing a tree ensemble include bagging, random forest, gradient boosting, etc.

CLASSIFICATION AND REGRESSION TREE (CART)

The best way to understand a tree algorithm is to begin with the classification and regression tree (CART) model. Not surprisingly, the model is commonly used to solve classification and regression problems. To grow a CART tree, an algorithm automatically figures out at each branch the split that minimizes the overall impurities in the child nodes. The trees keep branching until the leaves are reached. Growing a tree without proper bounding will potentially lead to over-fitting. A technique called pruning is always used to prevent over-fitting.

To understand the power of a CART model, let us look at an upsell problem that a health insurance company is trying to

FIGURE 1



solve. In the scenario, the company tries to offer a certain new insurance product to their existing customers. To determine if an offer can be made, the underwriters have to look through a series of rules and assess the health risks of their customers. For the purpose of demonstration, we built a CART model based on existing customers with decisions made by underwriters and show that the model reproduces the underwriting decision with great accuracy.

Here we only consider a subset of all the rules in the underwriting manual, including only five variables (i.e., age, health index and three other evidence variables). The target variable is the underwriting decision, which can be either "offer" or "no offer." Without understanding the medical knowledge behind the underwriting rules, the CART model (Figure 1) successfully reproduces the underwriting decisions with an accuracy greater than 99.9 percent.

Looking into the splits that the tree model learned, we find them matching the underwriting rules very well. For example, the top splits correctly state the fact that this product only targets people younger than 66. Besides the age restriction, the model also correctly learned more complex rules that are combinations of age, health index and a series of other evidences.

Figure 1: *A CART tree for predicting the underwriting decision of a health product, based on age, health index and three other evidence variables. For each split, the model will go to the left branch if the label is true and to the right branch otherwise. For a certain input, the model tree keeps making decisions upon splits until a leaf, either "Offer" or "No Offer," is reached.*

As a comparison, we also built a logistic regression model for the same data set. The regression model has a slightly worse prediction accuracy of 95 percent. It is not surprising as the underwriting decision is made by answering a series of yes or no rules, which fits into the inherent structure of a tree model better than a regression model.

THE LIMIT OF CART MODEL

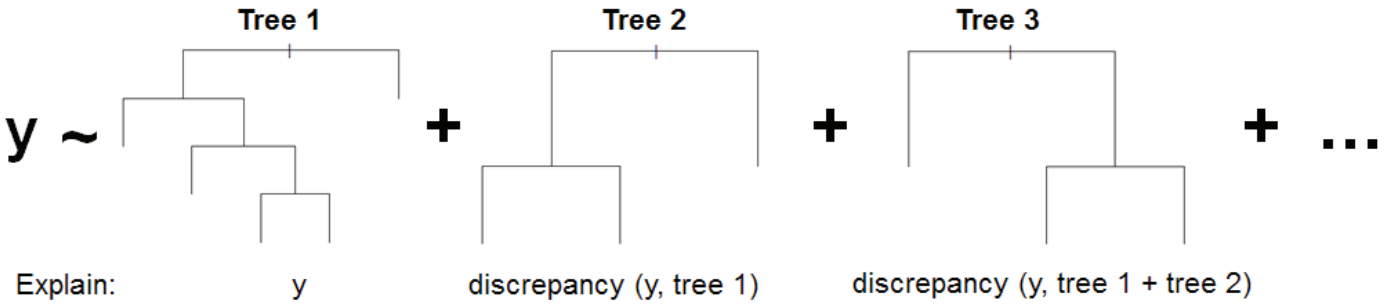
Apparently, the example above is oversimplified. A real underwriting manual considers far more than just five factors. Moreover, there could be hundreds of factors that determine a person's health situation and a lot of them may not even be known to us. For example, the contributing factors to cancer could be aging, tobacco, sun exposure, radiation exposure, chemicals, viruses, bacteria, etc. However, there is still a 10 percent to 30 percent chance[1] that a person developed cancer due to "bad luck" (the unknown). Moreover, the occurrence of an event might result from complex intermingles of various factors that do not fit into a yes/no structure.

In facing complex problems, the performance of a CART model is not always satisfactory. More sophisticated approaches are often needed to tackle a real world problem.

TREE ENSEMBLE

Random Forest: The occurrence of an event might be due to numerous factors in a complex way. If one single tree cannot

FIGURE 2



handle the complexity, would a number of them do? The answer is often yes. In a random forest algorithm, all trees are grown in parallel to predict the same target, but each tree is only provided with a subset of all information available. It may sound a little counter-intuitive that the algorithm is trying to build a better model with less information. In fact, although each single tree grown this way is a weaker predictor by itself, the final decision that is made through a voting mechanism from the tree crowd normally ends up better.[2]

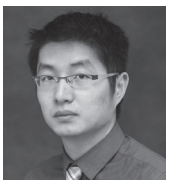
Boosted Tree is another popular way of growing a tree ensemble. Unlike Random Forest where all trees are grown to predict the same target, boosting algorithms approach the target sequentially. Specifically, the algorithm starts by growing a simple base tree that tries to make a good approximation of the target function. It is fine if the base tree cannot make accurate predictions as a subsequent tree will grow to make improvement on top of the existing one. The variation that cannot be explained by the base tree will be the target of the second tree (Figure 2). If discrepancy between the trees' prediction and the target remains, a third tree will grow. The process continues iteratively until a converging point is reached. Loosely speaking, the algorithm grows trees one-by-one and each tree is grown to correct the error that results from its precedents.

Boosted Tree often has a great performance accuracy[3] as it can capture complex nonlinear patterns effectively. However, an intuitive interpretation of the trees grown from boosting algorithms is usually hard because the trees (except the base tree) are not actually predicting the target directly.

Figure 2: *A schematic draw for Boosted Tree Algorithm. A base tree (Tree 1) is grown targeting on the objective function. A following tree (Tree 2, Tree 3) is then grown to explain the discrepancy between the target and the existing trees. The process continues until a converging point is reached.*

WHICH ALGORITHM TO CHOOSE?

With all the tree and forest descriptions, it is probably a good time to make a decision on which algorithm to pick. Should one use a single tree, a forest of trees, or other AIs? As we have demonstrated in our CART model, a tree algorithm is generally good at reproducing a system that is designed upon discrete rules (e.g., underwriting decisions), especially if the rules follow a binary structure. The tree algorithm naturally puts data into blocks and is therefore ideal for business problems like segmentations, categorization, etc. When you wish to solve problems that consist mainly of continuous changes (e.g., mortality risks, claim incident), algorithms like GLM (generalized linear model), or survival analysis might be better choices. Neural network based algorithms like deep learning are normally good at solving problems that involve image processing, handwriting recognition, face recognition, etc., which are not often confronted in insurance. ■



Dihui Lai, Ph.D., is a data scientist at RGA Reinsurance Company in Chesterfield, Mo. He can be reached at *dlai@rgare.com.*

Bingfeng Lu is an assistant data scientist at RGA Reinsurance Company in Chesterfield, Mo. He can be reached at *blu@rgare.com.*

REFERENCES

1. *Song Wu, Scott Powers, Wei Zhu and Yusuf A. Hannun, "Substantial contribution of extrinsic risk factors to cancer development," Nature, 529, no. 7584 (2016).*
2. *Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani, "An Introduction to Statistical Learning with Applications in R", Springer, (2015), pgs. 319–20.*
3. *Trevor Hastie, Robert Tibshirani and Jerome Friedman, "The Elements of Statistical Learning Data Mining, Inference, and Prediction," Springer, (2013), pgs. 589–91*