Computational Statistics and Data Analysis 54 (2010) 1197–1205

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

The Bayesian Additive Classification Tree applied to credit risk modelling

Junni L. Zhang [a,*], Wolfgang K. Härdle [b]

[a] *Department of Business Statistics and Econometrics, Guanghua School of Management, Peking University, Beijing 100871, PR China*

[b] *Center for Applied Statistics and Economics, Wirtschaftswissenschaftliche Fakultät, Humboldt-Universität zu Berlin, Spandauer Straβe 1, 10178, Berlin, Germany*

A R T I C L E   I N F O

Article history:
Received 2 January 2009
Received in revised form 14 October 2009
Accepted 26 November 2009
Available online 18 December 2009

Keywords:
Classification and regression tree
Financial ratio
Misclassification rate
Accuracy ratio

A B S T R A C T

We propose a new nonlinear classification method based on a Bayesian "sum-of-trees" model, the Bayesian Additive Classification Tree (BACT), which extends the Bayesian Additive Regression Tree (BART) method into the classification context. Like BART, the BACT is a Bayesian nonparametric additive model specified by a prior and a likelihood in which the additive components are trees, and it is fitted by an iterative MCMC algorithm. Each of the trees learns a different part of the underlying function relating the dependent variable to the input variables, but the sum of the trees offers a flexible and robust model. Through several benchmark examples, we show that the BACT shows excellent performance. We apply the BACT technique to classify whether firms would be insolvent. This practical example is very important for banks to construct their risk profile and operate successfully. We use the German Creditreform database and classify the solvency status of German firms based on financial statement information. We show that the BACT is a serious competitor to the logit model, CART, the Support Vector Machine, random forest and gradient boosting.

1. Introduction

Classification techniques have been popularly used in many fields. Standard classification tools include linear and quadratic discriminant analysis and the logistic model. The support vector machine (SVM) (Vapnik, 1995, 1997) has recently emerged as an important nonlinear classification tool. It maps the input space nonlinearly into a high dimensional feature space, and tries to find linear separating hyperplanes for the classes in the feature space, penalizing the distances of misclassified cases to the hyperplanes. The SVM has been widely and successfully applied to classification problems in many domains and often show excellent performance compared to other classification methods.

Decision trees compose an important category of nonlinear classification methods. Ever since the introduction of the classification and regression tree (CART) by Breiman et al. (1984), it has attracted strong interest from researchers and practitioners. Fig. 1 shows an example of a classification tree, where the root node (t_1) contains all training observations, and the training data are recursively partitioned by values of the input variables (x's) until reaching the leaf (terminal) nodes (t_3, t_4, t_6 and t_7) where the classification decision (for y) is made for all observations contained therein. For regression problems in which the dependent variable is continuous, a predicted value for the dependent variable would be assigned for all observations contained in each leaf node.

Traditional search methods for CART models use locally greedy algorithms to find the partitions. The Bayesian approaches for CART models (Chipman et al., 1998; Denison et al., 1998; Wu et al., 2007) specify a formal prior distribution for trees and other parameters and use Markov Chain Monte Carlo methods to sample them from the posterior distribution.

* Corresponding author.
E-mail addresses: zjn@gsm.pku.edu.cn (J.L. Zhang), haerdle@wiwi.hu-berlin.de (W.K. Härdle).

0167-9473/$ – see front matter © 2010 Published by Elsevier B.V.
doi:10.1016/j.csda.2009.11.022

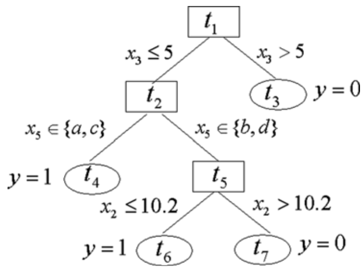

Fig. 1. Example of a classification tree.

The boosted tree model (Freund and Schapire, 1997; Friedman, 2001, 2002) and the random forest model (Breiman, 2001) combine a set of trees to improve model performance. Boosting fits a tree in each step to explain residuals not fitted by previous trees, and the final model is associated with the sum of all trees. Random forest fits a tree at each step with randomly sampled data and predictors, and then average predictions across the trees. Chipman et al. (in press) proposed the Bayesian Additive Regression Tree (BART), a Bayesian model combining a set of trees, in which the mean of a continuous dependent variable is approximated by a sum of trees. This "sum-of-trees" model is defined by a prior and a likelihood, and fitted by iterative MCMC algorithm. Each individual tree explains a different portion of the underlying mean function, but the sum of these trees turns out to be a flexible and adaptive model. Chipman et al. (in press) showed that BART is a serious competitor to LASSO (Efron et al., 2004), gradient boosting (Friedman, 2001), random forests, and neural networks with one layer of hidden units. We will extend BART into the classification context, and therefore term the resulting classification technique as the Bayesian Additive Classification Tree (BACT).

To investigate the differences among the logit model, SVM, CART and BACT, we plot in Figs. 2 and 3 the contours of these models trained to classify the solvency status of German firms using the German Creditreform database based on only two variables — the ratio of operating income to total assets ($x3$ in the figures) and the ratio of accounts payable to total sales ($x24$ in the figures). Details of this application will be discussed in Section 4. The contours for the logit model are linear, thus making it inflexible for complex applications. The SVM finds flexible smooth curves in the input space (linear hyperplanes in the feature space) that can separate the classes. The CART is based on a single tree which recursively partitions the observations by the input variables, and hence the contours are piecewise linear. The BACT is based on the sum of many trees, so the contours are not constrained to be piecewise linear as in CART; although these contours are not as smooth as in SVM, they are quite flexible in explaining complex structure.

The rest of this paper is organized as follows. Section 2 will describe the BACT in detail. Section 3 will use several benchmark examples from the UCI Machine Learning Repository to compare the performance of the BACT with the logit model, the SVM, gradient boosting and random forests. Section 4 will discuss our application to classification of solvency status of Germany firms using the German Creditreform database. Section 5 then concludes.

2. The Bayesian Additive Classification Tree (BACT)

2.1. The model

Consider a binary classification problem in which a dependent variable $Y \in \{1, 0\}$ needs to be predicted based on a set of input variables $\boldsymbol{x} = (x_1, \ldots, x_p)^\top$. The majority of classification models assume that there is a latent continuous variable Y^* that determines the value of Y as follows

$$\begin{cases} Y = 1 & \text{if } Y^* \ge 0 \\ Y = 0 & \text{if } Y^* < 0 \end{cases} \tag{1}$$

In the context of generalized linear models (GLM), the relationship of Y^* and $\boldsymbol{x}$ is

$$Y^* = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \varepsilon,$$

where the distribution of ε determines the link function, e.g. logit or probit. The generalized additive models (GAM, Hastie and Tibshirani, 1990) replace each linear term in the GLM by a more generalized functional form and relate Y^* to $\boldsymbol{x}$ by

$$Y^* = \beta_0 + f_1(x_1) + \cdots + f_p(x_p) + \varepsilon,$$

where each f_j is an unspecified smooth function.

Following the idea of the BART in Chipman et al. (in press), we assume that Y^* is related to $\boldsymbol{x}$ through an additive model, where each additive component is a tree based on all input variables (rather than a flexible function based on a single input variable as in GAM). In order to formally introduce the model, we first introduce some notation. Let m denote the number of trees to be used. For $j = 1, \ldots, m$, let T_j denote the jth tree with a set of partition rules based on the input variables, and let L_j denote the number of leaf nodes in T_j; for $l = 1, \ldots, L_j$, let μ_{jl} denote the (continuous) predicted value associated with