MBACT - Multiclass Bayesian Additive Classification Trees

B. Kindo *, H. Wang, and E. Peña *
Department of Statistics,
University of South Carolina,
Columbia SC 29208

May 22, 2014

Abstract

In this article, we propose Multiclass Bayesian Additive Classification Trees (MBACT) as a nonparametric procedure to deal with multiclass classification problems. MBACT is a multiclass extension of BART: Bayesian Additive Regression Trees [Chipman et al., 2010]. In a range of data generating schemes and real data applications, MBACT is shown to have good predictive performance, competitive to existing procedures, and in particular it outperforms most procedures when the relationship between the response and predictors is nonlinear.

1 Introduction

Multiclass classification deals with classifying or discriminating units into either one of $K$ classes; Here $K \in \mathbb{Z}^+ = \{1, 2, \ldots\}$ with $K \geq 3$. Multiclass classification problems are encountered in multiple disciplines. For instance, in a bio-medical application, [Khan et al., 2001] used artificial neural networks to classify small round blue cell tumors into four tumor sub-types:

*Research partially supported by NSF Grant DMS1106435 and NIH Grants R01CA154731, RR17698, and 1P30GM103336-01A1.
Ewing family of tumors (EWS), rhabdomyosarcoma (RMS), neuroblastoma (NB), and Burkitt lymphoma (BL). Financial bonds issued by governments (local or national), credit institutions and companies often receive a credit rating which classifies the creditworthiness of the issuing organization into ordered multinomial gradations. Standard and Poor's rating agency, for example, currently uses the gradations AAA, AA, A, BBB, BB, B, and so forth to discriminate the credit risks associated with financial obligations, with the class AAA indicating the highest level of creditworthiness. Insurance companies classify their policyholders into several classes. A simple classification may include the classes Ultra-preferred, Preferred, and Standard. In application of Atmospheric and Oceanic systems, multiclass support vector machines was incorporated into Earth Observing System Models in Lee et al. 2004 to classify the radiance profiles of clouds as clear, liquid clouds, or ice clouds. Other applications in text recognition, spectral imaging, chemistry, and forensic science are studied in Li et al. 2004, Rennie and Rifkin 2001, Fauvel et al. 2006, Evett and Spiehler 1987, Vergara et al. 2012.

Several methods for binary classification, multiclass classification and the usual continuous response regression have been proposed. We focus on methods that use the idea of ensemble learning. A short list of such methods includes Boosting, Bagging, Random Forests, and BART in Freund et al. 1996, Breiman 1996, 2001, Chipman et al. 2010, respectively. A common approach in the literature of multiclass classification is the reduction of the multiclass problem to several binary classification problems. This is attributed to the existence of a plethora of good binary classification methods and algorithms. The simplest of such reductions is One-Against-All Dietterich and Bakiri 1995a, which transforms a $K > 2$ multiclass classification problem into $K$ class $k$ versus classes $\{1, \ldots, K\} \setminus \{k\}$ binary classification problems for $k = 1, \ldots, K$. Another widely used technique is the fitting of binary classification models for all pairwise combinations of $\{1, \ldots, K\}$, resulting in $K(K - 1)/2$ binary models. This type of reduction is termed as pairwise coupling with details provided in Hastie and Tibshirani 1998, Wu et al. 2004. Other reduction techniques include “error-correcting output codes” Dietterich and Bakiri 1995b, Schapire and Singer 1998, and “All versus All” reduction Hsu and Lin 2002, Allwein et al. 2001.

In this article, we extend the Bayesian Additive Regression Trees (BART) proposed in Chipman et al. 2010, to tackle multiclass classification problems. BART is a Bayesian sum of regression trees model for implementing nonparametric and nonlinear regression and binary classification. Across dif-
ferent data generating schemes, BART enjoys good predictive performance and robustness. The binary classification extension of BART was used in [Zhang and Härdle, 2010] to classify credit risks of German financial institutions. A work most related to this article [Agarwal et al., 2013] utilized BART for the purposes of satellite image classification. Their multiclass classification methodology combined binary BART, and one-versus-all technique of transforming a multiclass problem to a series of binary classification problems.

We utilize a Bayesian probit model formulation (cf. [Albert and Chib, 1993]) in conjunction with the idea of ensemble of trees as in [Chipman et al., 2010], to propose MBACT: Multiclass Bayesian Additive Classification Trees for unordered and ordered multinomial response regression. MBACT gives a way to estimate posterior class probabilities in nonparametric and nonlinear manner. In a comprehensive simulation study with various data generating schemes, MBACT is shown to be robust, and we argue that MBACT is a serious contender in its predictive performance to widely used multiclass classification methods. Its ability to retrieve the signal when a nonlinear relationship exists between the predictors and response is noteworthy. Moreover, MBACT allows us to measure the relative importance of each predictor hence, incorporating an important tool for variable selection within the model. This article is organized as follows. The introduction proceeds with a brief review of some existing multiclass classification procedures; Sections 2 and 3 deal with setting up the priors and posterior updating schemes of MBACT for unordered and ordered responses respectively; in Section 4, we outline a general framework for a multiclass classification problem and model evaluation which take into account the cost of misclassification; Sections 5 and 6 involve comparison of unordered and ordered MBACT to other procedures through simulation studies; Section 7 applies MBACT to real data sets; and concluding remarks are given in Section 8.

1.1 Review of Some Existing Methods

In this subsection, we outline a brief review of existing multiclass classification methods. The methods described here are either widely used non-linear multiclass classification methods, or that they are known to have very good predictive ability. Classification and Regression Trees (CART) [Breiman et al., 1984], works by creating recursive binary partitions of the predictor space that take the form \( \{ x_j > c(x_j) \} \) versus \( \{ x_j \leq c(x_j) \} \), where \( x_j \) is
the \( j \)th predictor, and \( c(x_j) \) a constant that lies within the range of \( x_j \). The partitioning of the space is followed by fitting simple functions (e.g., constant function) within each resulting partition. For each recursive split, the selection of a predictor and splitting value is done so that a pre-specified loss is minimized (i.e., maximum information gain is obtained). CART is considered one of the earliest nonlinear and nonparametric classification and regression methods that handles both binary and multiclass classification problems.

KNN, k-Nearest Neighbors [Fix and Hodges Jr, 1952, Cover and Hart, 1967], is one of the oldest nonparametric classification techniques based on the intuition that, as described by Thomas Cover in [Cover, 20, March 1982], “things that look alike must be alike.” For each unit in the test data set, KNN picks the most represented class of the \( k \) closest/recent training predictors as the predicted class, where \( k \) is an integer and the closeness of a point is measured in terms of Euclidean distance. The use and properties of KNN in multiclass classification have been studied in [Cover and Hart, 1967, Bay, 1998, Athitsos and Sclaroff, 2005]. Quadratic Discriminant Analysis (QDA) is based on the assumptions that the prior probability that a unit belongs to class \( k \) is \( \pi_k \) with \( \sum_{k=1}^{K} \pi_k = 1 \), and that, given class \( y = k \), the vector of predictors \( x \) follows a multivariate normal distribution, \( x|y = k \sim N(\mu_k, \Sigma_k) \) for \( k = 1, \ldots, K \). Applying Bayes Theorem, the posterior probability of belonging to class \( k \) is obtained by

\[
P(y = k|x) = \frac{f(x|y = k) \pi_k}{\sum_{i=1}^{K} f(x|y = i) \pi_i}.
\]

Linear Discriminant Analysis (LDA) is a special case of QDA obtained by setting \( \Sigma_k = \Sigma \) for all \( k = 1, \ldots, K \). Details on LDA and QDA are found in [Duda et al., 2012, Hastie et al., 2001]. In binary response classification, Support Vector Machines (SVM) [Cortes and Vapnik, 1995, Vapnik, 1999] finds an optimal hyperplane in the predictor space that seeks to separate one response class from the other. When the data are separable, a unique hyperplane is obtained when the distance between the closest two points (support points) in either class is maximized. When the data are not completely separable, some perturbation is induced so that some points are allowed to be on the wrong side of the hyperplane. Nonlinear hyperplanes are induced in the predictor space by projecting linear hyperplanes obtained in an extended predictor space with the extension done via basis expansions which include the use of polynomial, spline, wavelet, and reproducing kernel Hilbert space
expansions among others. Extensive study of SVM is found in [Vapnik, 1999, Hastie et al., 2001]. The multiclass extension of SVM is often implemented by reduction of the multiclass problem into a sequence of binary classification problems. [Hsu and Lin, 2002, Lee et al., 2004, Zhang, 2004] have studied the theoretical and application aspects of multi-category support vector machines which does not reduce the problem into binary classification problems. Random Forest, [Breiman, 2001, 2000], is considered among the best multiclass classifiers. Random Forest is an ensemble learning method in which each classifier is a classification tree generated in the following manner.

1. A bootstrap sample of the training data is selected to fit a CART like model without pruning. The partitions of the training data are termed as “bootstrap” and “out of bootstrap” samples.

2. At each node, a subset of the predictors is randomly selected and the best predictor within that specific subset is chosen as a variable to split the tree node.

Each learner (tree) votes on the class membership of a test unit, and the majority vote is taken as the Random Forest prediction. Recently [Biau et al., 2008, Biau, 2012] have studied the consistency of Random Forest. However, the statistical and mathematical underpinnings as to why Random Forest has very good predictive ability are yet to be extensively studied.

2 Multiclass Bayesian Additive Classification Trees for Unordered Response

Suppose we have a training data set \((y_i, x_i)\) for \(i = 1, \ldots, n\), where \(y_i \in \mathcal{Y} = \{1, \ldots, K\}\) and \(x_i \in \mathcal{X} = \mathbb{R}^p\) denote the observed class and predictors for the \(i^{th}\) individual or unit, respectively. We make the assumption that \((y_i, x_i), i = 1, \ldots, n\) are observations from independent and identical random variables \((Y_i, X_i), i = 1, \ldots, n\) that are governed by the probability distribution \(P(Y, X)\). Our goal is to classify a new test unit with a vector of predictors \(x \in \mathcal{X}\) into one of \(K\) classes.

The observed response \(y \in \mathcal{Y}\) can be considered to arise from an unobserved vector of latent variables \(z \in \mathbb{R}^{K-1}\) [Albert and Chib, 1993, Geweke et al., 1994]. If none of the entries of the vector \(z\) are greater than zero, the class \(y = 1\) is observed. On the other hand, if the \(k^{th}\) element is the largest
positive entry of $z$, the class $y = k + 1$ is observed, for $k = 1, \ldots, K - 1$. Suppose that for the observed training data $\{(y_i, x_i), i = 1, \ldots, n\}$, there are corresponding latent vectors $\{z_i = (z_{i1}, \ldots, z_{iK-1})^T, i = 1, \ldots, n\}$ with $z_{ik}$ defined as

$$z_{ik} = G_k(x_i, \Theta_k) + \epsilon_{ik}, \quad \epsilon_{ik}, \sim N(0, 1)$$

for $k = 1, \ldots, K - 1$, and $i = 1, \ldots, n$ where

$$G_k(x, \Theta_k) = \sum_{j=1}^{n_T} g_{kj}(x, \theta_{kj})$$

is a sum of $n_T$ regression trees with tree parameters $\Theta_k = (\theta_{k1}, \ldots, \theta_{kn_T})$. Note that $g_{kj}(x, \theta_{kj})$ is the $j^{th}$ tree in the sum $G_k(x, \Theta_k)$ that subdivides the predictor space into $b$ partitions (or terminal nodes) assigning terminal node $l$ to the parameter $\theta_{kjl}$, $l \in \{1, \ldots, b\}$. The partitioning of the predictor space is done by recursive two way (i.e., binary) splits of the form $\{x < s\}$ versus $\{x \geq s\}$ where $x$ is one of the predictors, and $s$ is a value in the range of $x$. Details on the tree generating scheme is provided in Section 2.1.1. Multiclass Bayesian Additive Classification Trees is, thus, a multiclass probit model summarized as

$$y_i = \begin{cases} 1, & \text{if } z_{ik} \leq 0 \text{ for all } k \\ k + 1, & \text{if } (z_{ik} = \max_l z_{il}) \wedge (z_{ik} > 0) \end{cases}$$

$$z_i | x_i \sim N(G(x_i, \Theta), \Sigma)$$

where $\Theta = (\Theta_1, \ldots, \Theta_{K-1})$, is the collection of all terminal node parameters of $(K - 1) \times 1$ vector of sum of classification trees,

$$G(x_i, \Theta) = (G_1(x_i, \Theta_1), \ldots, G_{K-1}(x_i, \Theta_{K-1}))^T.$$  

We take a simplifying assumption that the covariance matrix $\Sigma$ in (3) to be the $K - 1$ identity matrix $I$ indicating that each sum of classification trees is independent of the other. With this simplifying assumption in place, the parameters of this model are

$$\{ (g_{kj}, \theta_{kj}) , k = 1, \ldots, K - 1, j = 1, \ldots, n_T \}.$$  

Observe that the trees are also considered as parameters and a Bayesian proceeds with specification of prior distributions on all the parameters of the model.
2.1 Prior Specification

By taking $\Sigma = I$, we impose an independence assumption on the prior that

$\{(g_{kj}, \theta_{kj}), j = 1, \ldots, n_T\} \perp \{(g_{qj}, \theta_{qj}), j = 1, \ldots, n_T\}$ for $k \neq q$. \hfill (5)

With further assumption that given a single tree in a sum of trees, the terminal node parameters are independent, for fixed $k$, we take priors of the form

$$p((g_{k1}, \theta_{k1}), \ldots, (g_{kn_T}, \theta_{kn_T})) = \prod_{j=1}^{n_T} p(g_{kj}, \theta_{kj})$$

$$= \prod_{j=1}^{n_T} p(\theta_{kj}|g_{kj}) p(g_{kj})$$

where the prior on terminal node parameters $\theta_{kj} = (\theta_{jk1}, \ldots, \theta_{jkb})$ given the $j^{th}$ tree in the $k^{th}$ sum of trees is taken as $p(\theta_{kj}|g_{kj}) = \prod_{l=1}^{p} p(\theta_{kjl}|g_{kj})$. The prior specification for a single tree, $p(g_{kj})$, is described in detail in Section 2.1.1.

2.1.1 The $g_{kj}$ prior

As in [Chipman et al., 1998, 2010], the prior on a single tree $g_{kj}$ is specified in terms of a “tree-generating stochastic process.” The tree generation begins with a trivial initial node followed by splitting of the form $\{x < s\}$ versus $\{x \geq s\}$ where $x$ is randomly selected from $x_1, x_2, \ldots, x_p$. Each of the $p$ predictors have an equal probability of being selected as the splitting predictor $x$ indicating that this prior specification places equal level of importance to each predictor, analogous to the usual assignment of uniform prior on parameters. The level of importance placed on the predictors changes as the prior gets updated in the posterior updating process. Let $d_{\eta}$ be the depth (the number of parent nodes) of a terminal node $\eta$. The probability that the terminal node is split is given by

$$\frac{\alpha}{(1 + d_{\eta})^\beta}, \quad \alpha \in (0, 1), \beta \in [0, \infty). \hfill (7)$$

A lower value of $\alpha$, and a higher value of $\beta$ result in a tree with small number of terminal nodes. In other words, influence of individual trees in the sum can be controlled by carefully choosing $\alpha$ and $\beta$. The values $\alpha = 0.95$, and $\beta = 2$
are taken as default values, but they can also be chosen via cross validation. Given a splitting predictor, the splitting value $s$ is taken to be a random sample from discrete uniform distribution of the set of observed values of the selected predictor provided that such a value does not result in an empty partition. Note that this prior specification is impartial to importance of the predictors. The observed response vector will be utilized while updating the prior as described in Section 2.2.

2.1.2 The $\theta_{kji} | g_{kj}$ Prior

Given a tree $g_{kj}$ with $b$ terminal nodes, the prior distribution on the terminal node parameters is taken to be

$$\theta_{kji} | g_{kj} \overset{iid}{\sim} N(\mu_k, \tau_k^2)$$

Note that $z_{ik} - G_1(x_i, \Theta_1) \sim N(0,1)$ and highly likely to take values in the interval $(-3, 3)$. We can, thus, select $\mu_k$ and $\tau_k$ in such a way that the cumulative effect of the prior on all $\theta_{kji} | g_{kj}$s in the $k^{th}$ sum of trees assigns high probability to the interval $(-3, 3)$. This requirement can be captured by choosing $\mu_k = 0$ and $\tau_k = \frac{3.0}{r \sqrt{n_T}}$ where $r > 0$ and $n_T$ the number of trees in the $k^{th}$ sum of trees. The hyper-parameters $r$, and $n_T$ play the role of adjusting the level of shrinkage on the contribution of each individual tree. Default values $r = 2$, and $n_T = 200$ are recommended. Again, these can be chosen via cross validation, thus, including the parameters discussed in Section 2.1, the set of tuning parameters of this model are $\alpha$, $\beta$, $r$, and $n_T$.

2.2 Posterior Computation for Unordered Multiclass Bayesian Additive Classification Trees

The assumption in (5) that each of the $K-1$ sum of trees are independent of one another, allows us to write the posterior

$$p((g_{kj}, \theta_{kj}); k = 1, \ldots, K-1, j = 1, \ldots, n_T | y, x)$$

in the form

$$\prod_{k=1}^{K-1} p((g_{k1}, \theta_{k1}), \ldots, (g_{kn_T}, \theta_{kn_T}) | y, x).$$
Since direct computation of joint posterior distribution is a difficult undertaking, we make use of the MCMC algorithm based on the idea of data augmentation for posterior sampling proposed in [Tanner and Wong, 1987]. Specifically, the Gibbs sampling algorithm outlined below is implemented. For each \( k \), \( n_T \) successive draws of \((g_{kj}, \theta_{kj})\) are sampled from the conditional
\[
(g_{kj}, \theta_{kj}) \mid \{(g_{kj'}, \theta_{kj'}) ; j' \neq j, j' = 1, \ldots, n_T\}, z, y
\]
followed by a draw from
\[
z \mid \{(g_{kj}, \theta_{kj}), k = 1, \ldots, K - 1, j = 1, \ldots, n_T\}, y.
\]
The draw in (11) is from truncated multivariate normal distribution which can be done through a sequence of univariate normal sampling. The sampling scheme for the \((l + 1)\)th iteration is done via either one of the following two bullets depending on whether the multiclass response \( y = 1 \), or \( y \neq 1 \).

- When \( y_i = 1 \) for all \( k \), \( z^{(l+1)}_{ik} \) given \( \{z^{(l)}_{ik'}, k' \neq k\} \), \( G^{(l)} \), \( \Theta^{(l)} \), and \( y_i = 1 \), can be sampled from \( N\left(G^{(l)}_k (x_i, \Theta^{(l)}), 1\right) \) truncated from above by 0. Here, \( G^{(l)} \) is vector of \( K - 1 \) sum of trees in the \( l \)th iteration.

- When \( y_i \neq 1 \), for \( k = 1, \ldots, K - 1 \), the following two steps are used.
  
  i. Sample \( z^{(l+1)}_{ik} \) given \( \{z^{(l)}_{ik'}, k' \neq k\} \), \( G^{(l)} \), \( \Theta^{(l)} \), and \( y_i = k + 1 \) from \( N\left(G^{(l)}_k (x_i, \Theta^{(l)}), 1\right) \) truncated from below by 0, followed by
  
  ii. draws of \( z^{(l+1)}_{iq} \) given \( z^{(l+1)}_{ik}, \{z^{(l)}_{ik'}, k' \neq k, k' \neq q\} \), \( G^{(l)} \), \( \Theta^{(l)} \), and \( y_i = k + 1 \) from \( N\left(G^{(l)}_k (x_i, \Theta^{(l)}), 1\right) \) truncated from above by \( z^{(l+1)}_{ik} \) for each \( q = 1, \ldots, K - 1, q \neq k \).

Turning our attention to (10), we proceed by rewriting (1) as
\[
z_{ik} = \left[ \sum_{l \neq j} g_{kl}(x_i, \theta_{kl}) \right] = g_{kj}(x_i, \theta_{kj}) + \epsilon_{ik}
\]
Now, the left hand side of (12) can be considered as a form of residuals \( R_{ik(j)} \) associated with the \( j \)th tree in the sum of trees. This quantity is used as the
response vector of a single tree model with \( g_{kj} \) taken as an initial value of the tree structure. The tree structure can then be updated by undertaking one of the actions GROW, PRUNE, CHANGE, or SWAP with the Metropolis-Hastings algorithm described in [Chipman et al., 1998]. However, we resort to taking the actions GROW, or PRUNE only as described in [Pratola et al., 2012] for the sake of computational efficiency. Aside from the computational efficiency, the proposal by [Pratola et al., 2012] is not inferior to that of [Chipman et al., 1998] since each individual tree tends to be shallow in our model setup, and that we have a sum of trees in contrast to the single tree model of [Chipman et al., 1998].

Once the tree is updated, the vector \( \theta_{kj} \) representing the terminal node parameters is sampled from \( \theta_{kj} | g_{kj}, R_k(j) \) which is again a draw from normal distribution. After sufficient burn-in, with a posterior sample of size \( S \), \( p_k(x_i) \) can be calculated using

\[
p_1(x_i) = \frac{1}{S} \sum_{s=1}^{S} \prod_{k} \Phi \left( - \sum_{j=1}^{n_p} g_{kj}(x_i, \theta^{(s)}_{kj}) \right)
\]

\[
p_{k+1}(x_i) = \frac{1}{S} \sum_{s=1}^{S} \prod_{k} \Phi \left( z_{ik}^{(s)} - \sum_{j=1}^{n_p} g_{kj}^{(s)}(x_i, \theta^{(s)}_{kj}) \right) I(z_{ik} > 0, z_{ik} > z_{il}, l \neq k)
\]

for \( k = 1, \ldots, K - 1 \). A test data feature \( x \) is run down all the trees in each sum of trees and in all posterior iterations after burn-in to calculate \( p_k(x) \) for \( k = 1, \ldots, K \).

### 3 Multiclass Bayesian Additive Classification Trees for Ordered Response

Sometimes a natural ordering is exhibited by the response variable. For example, a college student may be classified into one of the ordered gradations: Freshman, Sophomore, Junior, or Senior. A corporate bond can be classified into one of the ordered gradations: Junk grade, Low grade, Medium grade, or High grade. When such ordering in the response vector exists, define \( \eta_{ik} = \sum_{j=1}^{k} p_j(x_i) \) for \( k = 1, \ldots, K - 1 \) with \( \eta_{ik} \) modeled as

\[
\eta_{ik} = \Phi (\gamma_k - G(x_i, \Theta)) \ i = 1, \ldots n, k = 1, \ldots, K - 1,
\]
where \( p_j(x_i) \) is as defined in the preceding section, \( \Phi(\cdot) \) the standard normal CDF, and \( G(x_i, \Theta) \) a sum of classification trees,

\[
G(x_i, \Theta) = g_1(x_i, \theta_1) + \ldots + g_{n_T}(x_i, \theta_{n_T}).
\]  

(15)

The parameter \( \Theta \) is the collection of all tree parameters \( \theta_1, \ldots, \theta_{n_T} \) where \( \theta_j = (\theta_{j1}, \ldots, \theta_{jb}) \) is the vector of terminal node parameters of the \( j \)th tree in the sum. The model in (14) can be motivated by assuming that the values taken by the response \( y \) arise from an unobserved latent variable \( z \). Specifically, \( y_i = k \) if \( z_i \in (\gamma_{k-1}, \gamma_k] \) for \( k = 1, \ldots, K \). Here, we take \( \gamma_0 = -\infty \), \( \gamma_K = +\infty \), and \( \gamma_1 = 0 \) for the sake of identifiability. Thus, the parameters of ordered multiclass Bayesian additive classification trees are the bin/threshold parameters \( \gamma = (\gamma_0, \ldots, \gamma_K) \), the tree structures \( G \equiv (g_1, \ldots, g_{n_T}) \), and the terminal node parameters corresponding to all the trees in the sum \( \Theta = (\theta_1, \ldots, \theta_{n_T}) \). The prior specifications on these parameters are a uniform non-informative prior on the threshold parameters \( \gamma \); and priors of the form specified in Sections 2.1.1 and 2.1.2 for tree priors, and the corresponding terminal node parameter priors, respectively.

### 3.1 Posterior Computation for Ordered Multiclass Bayesian Additive Classification Trees

We now seek to find the posterior distribution of the parameters which are governed by the model in (14). The joint posterior distribution of all the parameters in the model, \( p(G, \Theta, \gamma|x, y) \), up to a proportionality constant is given by

\[
p(G, \Theta) p(\gamma) \prod_{i=1}^{n} \sum_{k=1}^{K} I(y_i = k) [\Phi(\gamma_k - G(x_i, \Theta)) - \Phi(\gamma_{k-1} - G(x_i, \Theta))]
\]

(16)

where \( p(G, \Theta) \) is the joint prior of the tree structure and the terminal node parameters, and \( p(\gamma) \) the prior on the bin parameters \( \gamma \). Again, since the computation of the joint posterior (16) is difficult, we implement a Gibbs sampling algorithm. Note that the joint distribution \( p(G, \Theta, \gamma, z|x, y) \) after placing a non-informative prior on \( \gamma \) is proportional to

\[
\prod_{i=1}^{n} \left[ \phi(z_i - G(x_i, \Theta)) \left\{ \sum_{k=1}^{K} I(y_i = k) I(\gamma_{k-1} < z_i \leq \gamma_k) \right\} \right].
\]

(17)
Thus, the conditional posterior distributions of $z_1, \ldots, z_n$ given $G, \Theta, \gamma, x, y$ are

$$z_i | G, \Theta, \gamma, y_i = k \sim N(G(x_i, \Theta), 1) I(\gamma_{k-1} < z_i \leq \gamma_k)$$

for $k = 1, \ldots, K, \quad (18)$

and the conditional density of $\gamma_k$ given $\{\gamma_j, j \neq k\}, G, \Theta, \gamma, x, y$ is proportional to

$$\prod_{i=1}^{n} [I\{y_i = k\}I(\gamma_{k-1} < z_i \leq \gamma_k) + I\{y_i = k + 1\}I(\gamma_k < z_i \leq \gamma_{k+1})]. \quad (19)$$

The Gibbs sampler is implemented by cycling through draws from (19), (18), and $n_T$ successive draws of $(g_j, \theta_j) | \{(g_l, \theta_l), l \neq j\}, z, y, x$ enumerated below.

i. Observe from (19) that $\gamma_k$ given $\{\gamma_j, j \neq k\}, G, \Theta, \gamma, x, y$ can be sampled from a uniform distribution with the upper bound $b = \min(\min_i \{z_i : y_i = k + 1\}, \gamma_{k+1})$, and lower bound $a = \max(\max_i \{z_i : y_i = k\}, \gamma_{k-1})$.

ii. Draws of $z_i, i = 1, \ldots, n$, from truncated normal distribution given in (18).

iii. The $n_T$ successive draws of

$$(g_j, \theta_j) | \{(g_l, \theta_l), l \neq j\}, z, y, x \quad (20)$$

are done by rewriting (14) as

$$\gamma_k - z = \sum_{l=1}^{n_T} g_j(x, \theta_j),$$

and further rearranging to obtain

$$\gamma_k - z - \sum_{l \neq j} g_j(x, \theta_j) = g_j(x, \theta_j). \quad (21)$$

Now, the left hand side of (21) can be considered as a residual $R_{(j)}$ associated with the $j^{th}$ tree in the sum of trees. Hence, we can sample from (20) as described in Section 2.2.
4 Mathematical Formulation and Model Evaluation

In this section, we outline a mathematical formulation to a multiclass classification problem. Suppose we have the data structure described in the first paragraph of Section 2. A multiclass classification problem deals with developing a classifier or function $\rho (\mathbf{x}) = \rho (\mathbf{x}; \mathbf{Y}, \mathbf{X}) : \mathcal{X} \to \mathcal{Y}$. Suppose that $y \in \mathcal{Y}$ is the unobserved true class of a test unit that has features $\mathbf{x}$. If we know $p_k (\mathbf{x}) = P (y = k|\mathbf{x})$, the probability that it belongs to class $k$ given its features, for $k = 1, \ldots, K$, or have a “good” estimator $\hat{p}_k (\mathbf{x})$ of $p_k (\mathbf{x})$, we can utilize these probabilities to make “cost-efficient” class prediction. Our multiclass classification method involves estimation of posterior class probabilities using Bayesian additive classification trees followed by “cost-efficient” class prediction.

The remaining part of this section delves into the description of “cost-efficient” class prediction. Suppose that if the prediction or action $\rho (\mathbf{x}) = a \in \mathcal{Y}$ is taken when, in fact, the true class is $y = l$, a pre-specified cost $C_{la} \geq 0$ is incurred. That is, there is a cost function $C (y, a)$ that assigns a pre-specified loss to every combination of action $a \in \mathcal{Y}$ and true class $y \in \mathcal{Y}$: $C (y, a) : \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$. The pre-specified cost combinations are described in Table 1. The idea behind “cost-efficient” decision is to utilize the class probability estimates so that the overall average cost is minimized. Table 1 can equivalently be expressed via the loss function

$$L (y, a) = \sum_{l=1}^{K} \sum_{m=1}^{K} C_{lm} I (y = l, a = m). \quad (22)$$

where $I (.)$ is the usual indicator function. If the misclassification costs are all equal, and the correct classification costs are all 0 (i.e., $C_{lm} = C > 0$ for $l \neq m$).
and \( C_{lm} = 0; l, m \in \{1, \ldots, K\} \), the loss function simplifies to
\[
L(y, a) = C I (y \neq a).
\] (23)

In order to arrive at the classification (prediction) of the test unit, we seek to minimize the average loss associated with the decision \( \rho(x) \). That is, we minimize \( E_{(Y,X)} L(Y, \rho(X)) \) which we can rewrite as
\[
E_{(Y,X)} L(Y, \rho(X)) = E_X E_{Y|X} L(Y, \rho(X))
\]
\[
= E_X \sum_{l=1}^{K} \sum_{m=1}^{K} C_{lm} E (I(\rho(X) = m)I(Y = l)|X = x)
\]
\[
= E_X \sum_{l=1}^{K} \sum_{m=1}^{K} C_{lm} I(\rho(x) = m)P(Y = l|x).
\]

The optimal action, in this case, is the class that minimizes the inner sum \( \sum_{m=1}^{K} C_{lm} I(\rho(x) = m)P(Y = l|x) \). It is easily seen that
\[
\arg \min_{m} \sum_{l=1}^{K} C_{lm} I(\rho(x) = m)P(Y = l|x) = \arg \min_{l} \sum_{l=1}^{K} C_{lm} \hat{p}_l(x).
\]

Hence, with a “good” estimator \( \hat{p}_l(x) \) of \( p_l(x) \) obtained by learning on the training data set, we predict the test unit to belong to the class
\[
\hat{y} = \arg \min_{m} \sum_{l=1}^{K} C_{lm} \hat{p}_l(x).
\] (24)

5 Simulation Studies and Variable Selection for Unordered MBACT

5.1 Simulated Data 1

In this simulation study, we generate a 3-class multiclass problem with 15 predictors, all of which are important. We utilize the nonlinear function \( f : [0,1]^5 \to \mathcal{R} \) used in [Friedman, 1991] which takes arguments that are
independent and identical standard uniform random variables. The function is given by
\[ f(x_1, \ldots, x_5) = 10 \sin (\pi x_1 x_2) + 20(x_3 - \frac{1}{2})^2 + 10x_4 + 5x_5. \] (25)

Define \( y_1 = f(x_1, \ldots, x_5) \), \( y_2 = f(x_6, \ldots, x_{10}) \), \( y_3 = f(x_{11}, \ldots, x_{15}) \), and the multiclass response \( y = \arg \max_i \{y_i, i = 1, 2, 3\} \). Fifty training and test data set pairs each with 100 observations are generated and the procedures Unordered MBACT, Support Vector Machines (SVM) with linear and radial kernels, Random Forest(RF), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Multinomial Logistic regression, Classification and Regression Trees (CART), Neural Networks, and K-Nearest Neighbors are fit. For each competing procedure, we selected the tuning parameters via 10 fold cross-validation in the training data. Note that we have used the default settings for MBACT in all our comparisons giving the other methodologies a slight advantage if they have tuning parameters. When making a comparison between the methodologies, we use the out of sample average error rate
\[
\frac{1}{m} \sum_{i=1}^{m} I(\omega_i \neq \hat{\omega}_i),
\] (26)
where \( \omega_i \) is the actual class and \( \hat{\omega}_i \) the predicted class of the \( i \)th observation in a given test data set. Note that this makes use of the loss function defined in (23) with \( C = 1 \). A set of tuning parameters that results in the lowest out-of-sample average error rate is selected. The tuning parameters are described in Table 2.

In this data generating scheme, Unordered-MBACT yields the lowest misclassification rate (26), and standard error in out-of-sample prediction. The average training and test error rates along with their respective standard errors in parentheses displayed in Table 3. Box plots of the fifty replication test misclassification rates, for each methodology, are displayed in Figure 1.

The code for the implementation of our procedure which is adapted from BayesTree package in R implementing BART [Chipman et al., 2010], and the C++ code of PBART [Pratola et al., 2012]. It is then modified to suit our multiclass classification setting. It can be found on the website http://www.stat.sc.edu/~kindo/research/ with the simulation study in this section used as an illustration.
Table 2: The columns of this table display the list of procedures in our comparison, the R Package utilized, and the tuning parameters for each procedure optimized using cross-validation. The abbreviations in the first column stand for the procedures mentioned in the first paragraph of Section 5.1.

| Methodology          | R Package   | Tuning parameter(s) |
|----------------------|-------------|---------------------|
| RF                   | randomForest| mtry                |
| CART                 | rpart       | no tuning parameters|
| SVM Linear           | kernlab     | C                   |
| SVM Radial           | kernlab     | C and σ              |
| QDA                  | MASS        | no tuning parameters|
| LDA                  | MASS        | no tuning parameters|
| NNET                 | nnet        | size and decay      |
| Multinomial Logistic | nnet        | decay               |
| KNN                  | caret       | k                   |

Figure 1: Box plot of Error Rates as defined in (26) for Simulated Data set I of each procedure in the horizontal axis.

One of the aspects that a practitioner considers in choosing among competing statistical methodologies is computational efficiency. From 50 independent replications of the simulation example in this section, we obtain the minimum, lower quartile, median, upper quartile, and maximum computation time in seconds as 7.658, 8.379, 10.464, 13.146, and 16.430, respectively. This is done using a personal laptop computer with the following specifi-
Table 3: Average error rates as defined in (26) and standard errors (in parentheses) based on 50 different training and test data set combinations for each procedure under comparison. The abbreviations in the first column are and their meanings are listed in the first paragraph of this section.

| Procedure            | Training Error Rate | Test Error Rate | Rank |
|----------------------|---------------------|-----------------|------|
| RF                   | 0.0000 (0.0000)     | 0.3592 (0.0074) | 7    |
| CART                 | 0.3144 (0.0092)     | 0.5168 (0.0089) | 10   |
| SVM Linear           | 0.1236 (0.0060)     | 0.3434 (0.0085) | 5    |
| SVM Radial           | 0.0360 (0.0063)     | 0.3450 (0.0089) | 6    |
| QDA                  | 0.0238 (0.0023)     | 0.4626 (0.0092) | 9    |
| LDA                  | 0.1680 (0.0062)     | 0.3244 (0.0071) | 3    |
| NNET                 | 0.0130 (0.0039)     | 0.3372 (0.0075) | 4    |
| Multinomial Logistic | 0.1486 (0.0062)     | 0.3144 (0.0073) | 2    |
| KNN                  | 0.2878 (0.0062)     | 0.4120 (0.0089) | 8    |
| U-MBACT              | 0.0494 (0.0029)     | 0.3096 (0.0067) | 1    |
(i.e., noise) predictors. Figures 3 summarizes the robustness of MBACT, RF, and SVM-Radial when additional noise predictors are included in the predictor space. The test error rate (26) over 50 replications is on the vertical axis while the number of unimportant predictors included in the predictor space is on the horizontal axis. The solid line represents the median test error rate while the dashed lines represent the 5th and 95th percentiles.

Figure 2: Variable Selection Plot. The vertical axis represents the proportion of times a predictor is used in forming or updating a tree, and the horizontal axis an index of the predictors.

5.2 Waveform Recognition Simulation Study

A simulation study is conducted using the popular three class waveform recognition problem of [Breiman et al., 1984]. Waveform recognition problem is considered as a difficult multinomial classification problem used as benchmark artificial data in many papers (see [Gama et al., 2003, Hastie and Tibshirani, 1996, Keerthi et al., 2005]) dealing with multiclass classification problem. Two of the three wave form functions \( h_1(i) = \max(6 - |i - 11|, 0) \), \( h_2(i) = h_1(i - 4) \), and \( h_3(i) = h_1(i + 4) \) are randomly weighted to generate 21 predictors. Specifically, the predictors generated by a contaminated random combination of waveform functions \( h_1(.) \) and \( h_2(.) \) are taken to be the
Figure 3: Comparison of Robustness to Unimportant Predictors of Random Forest, radial kernel Support Vector Machines, and U-MBACT. The vertical axis represents test error rate as defined in (26) while the horizontal axis represents the number of unimportant predictors included in the predictor space. The solid line in each plot is the median over 50 independent replications while the dashed lines represent the 5th and 95th percentiles.

features of class label 1. Similarly, waveform functions $h_1(.)$ and $h_3(.)$ are used for class label 2, and waveform functions $h_2(.)$ and $h_3(.)$ for class label 3.

\[
x_i = u h_1(i) + (1 - u)h_2(i) + \epsilon_i, \ i = 1, \ldots, 21 \text{ for class 1} \\
x_i = u h_1(i) + (1 - u)h_3(i) + \epsilon_i, \ i = 1, \ldots, 21 \text{ for class 2} \\
x_i = u h_2(i) + (1 - u)h_3(i) + \epsilon_i, \ i = 1, \ldots, 21 \text{ for class 3}
\]

where $u$ is a uniform(0,1) random variable and $\epsilon_i \sim N(0,1)$. Table 4 summarizes the average error rates (26) with the corresponding standard errors in parentheses based on 50 simulations each with a training data set of size 300 and test data set of size 500. Among the procedures we performed, MBACT gives the lowest average misclassification rate. Although it is not significantly lower than that of RF and SVMRadial, it is among the best for this data generating scheme. A lower average error rate of $0.157(0.005)$ on test data set is obtained in [Hastie and Tibshirani, 1996] using penalized Discriminant Analysis by Gaussian Mixtures with 10 simulations, each with a training data set of size 300 and test data set of size 500.
| Procedure   | Training Error Rate | Test Error Rate | Rank |
|-------------|---------------------|----------------|------|
| RF          | 0.0000 (0.0000)     | 0.1609 (0.0178) | 3    |
| CART        | 0.2047 (0.0295)     | 0.3123 (0.0246) | 10   |
| SVMLinear   | 0.0820 (0.0194)     | 0.1725 (0.0235) | 5    |
| SVMRadial   | 0.0644 (0.0268)     | 0.1605 (0.0168) | 2    |
| LDA         | 0.1303 (0.0201)     | 0.1967 (0.0240) | 7    |
| QDA         | 0.0424 (0.0134)     | 0.2102 (0.0205) | 9    |
| NNET        | 0.0122 (0.0215)     | 0.1999 (0.0312) | 8    |
| Multinom    | 0.0946 (0.0205)     | 0.1658 (0.0192) | 4    |
| KNN         | 0.1344 (0.0235)     | 0.1748 (0.0194) | 6    |
| MBACT       | 0.0617 (0.0141)     | 0.1603 (0.0167) | 1    |

Table 4: Waveform Recognition Results. Average error rates as defined in [20] and standard errors (in parentheses) based on 50 different training and test data set combinations for each procedure under comparison. The abbreviations in the first column are and their meanings are listed in the first paragraph of Section 5.1.

6 Simulation Study for Ordered MBACT

6.1 Nonlinear Ordered Simulated Data

In this section, we simulate a 5-class data set in which the response is ordered, and nonlinearly related to the predictors. Only twenty of the fifty predictors that are independent and identically distributed standard normal random variables, \(x_1, \ldots, x_{50}\), are related to the multiclass response. The simulation setup is as follows: Define random variables \(X_1^*, X_2^*, \) and \(Y\) as

\[
X_1^* = \max \{ \min \{x_1, x_2, x_3\}, \min \{x_4, x_5, x_6\}, \min \{x_7, x_8, x_9\} \}, \\
X_2^* = \min \{ \max \{x_{11}, x_{12}, x_{13}\}, \max \{x_{14}, x_{15}, x_{16}\}, \max \{x_{17}, x_{18}, x_{19}\} \}, \\
Y = aX_1^* + bX_2^* + \sin(cx_{10}) + \cos(dx_{20}),
\]

where \(a, b, c, \) and \(d\) are real numbers. \(Y\) is then discretized using monotone class boundary points in order to obtain ordered multiclass responses. The boundary points that separate class labels are calculated using the \(Y\) values in the training data in such a way that the number of observations belonging to each of the classes are nearly equal. For the simulation study in this section, we used data generating parameters: \(a = 0.25, b = 0.30, c = 0.20, d = 0.10,\) and 500 instances in the training as well as the test data sets.
The predictive performance of each procedure is evaluated using the Mean Absolute Deviation (MAD) criterion defined as

$$\frac{1}{m} \sum_{j=1}^{m} |\hat{\omega}_j - \omega_j|,$$  \hspace{1cm} (28)

where $\hat{\omega}_j$ and $\omega_j$ are the predicted and actual ordered classes of the $j^{th}$ observation in a given test data set, respectively. This makes use of the loss function defined in (22) with $C_{lm} = |l - m|$. The idea is that the cost of making a prediction error is higher for those predictions farther from the truth. The test error rates and standard errors (in parentheses) in Table 5 are based on 50 replications of training-test pairs. As we see from the results of this simulation study and real data applications of Section 7.2, ordered MBACT has a good predictive performance as measured by MAD when the response is ordinal.

| Methodology | Test MAD     | Training MAD  | Rank |
|-------------|--------------|---------------|------|
| RF          | 0.7664 (0.0471) | 0.7688 (0.0464) | 2    |
| CART        | 0.8543 (0.0903) | 0.8564 (0.0894) | 8    |
| SVM.L       | 0.8268 (0.0489) | 0.8287 (0.0484) | 6    |
| SVM.R       | 0.8464 (0.0392) | 0.8484 (0.0400) | 7    |
| NNET        | 0.7704 (0.0638) | 0.7718 (0.0637) | 3    |
| Multinom    | 0.7977 (0.0416) | 0.7997 (0.0406) | 5    |
| KNN         | 1.1975 (0.0604) | 1.1968 (0.0615) | 9    |
| QDA         | 1.1982 (0.0631) | 1.1975 (0.0636) | 10   |
| LDA         | 0.7864 (0.0429) | 0.7891 (0.0410) | 4    |
| O.MBACT     | **0.6506 (0.0391)** | 0.6529 (0.0368) | 1    |

Table 5: Average MADs (Mean Absolute Deviations) as defined in (28) with standard errors in parenthesis for Nonlinear Ordered Simulated Data

7 Real Data

7.1 Real Data Application for U-MBACT

In this section, we apply our procedure, U-MBACT, to some real data sets and compare its performance with other procedures. We use Forensic Glass
Classification, and Seeds data sets which are available for download from University of California at Irvine (UCI) machine learning data repository [Bache and Lichman, 2013]. In the forensic glass classification data, 9 features of 214 glass samples are measured and each unit is classified as either of the seven glass types: building windows float processed, building windows non-float processed, vehicle windows float processed, vehicle windows non-float processed, containers, tableware, or headlamps. This is a 6-class classification problem since none of the units are classified as vehicle windows non-float processed. In our analysis we split the data so that 50 units are in the test data and the remaining 164 in the training. The results in Table 6 show the average classification error rate and standard errors for the different classification methods over 50 random repetitions of training and test splits. QDA could not be implemented in this data set since one of the classes has very small representation. Unordered MBACT and RF are the two top performing procedures in terms of having the lowest misclassification rate.

In the Seeds data set [Charytanowicz et al., 2010], seven characteristics of X-ray images of wheat kernels are used to predict whether the kernel is for a wheat that belongs to Kama, Rosa, or Canadian wheat classification. The seven attributes are area, perimeter, compactness, length of kernel, width of kernel, asymmetry coefficient, and length of kernel groove. The data set consists of 210 measurements. Table 7 displays the training and test error rates with their respective standard errors over 50 replications with the data split such that there are 90 observations in the test data set. In this data set, the parametric procedures with the exception of neural networks perform better than the nonparametric procedures.

### 7.2 Real Data Application with Ordered Response

In this section, we compare the predictive performance of ordered MBACT, and other competitive methodologies on modified real data sets. The data sets are modified so that the response vector is ordinal. That is, the data sets in their original form had continuous numerical responses, but we discretize the continuous responses using bins/buckets by creating monotone class boundaries as described in [Chu and Ghahramani, 2005].

Four data sets used for comparing O-MBACT with some existing classification methodologies are Ailerons, Triazines, Wisconsin breast cancer, and Abalone from UCI machine learning repository [Bache and Lichman, 2013] with brief description as follows. Ailerons data set is used to predict
| Procedure     | Training Error Rate | Test Error Rate | Rank |
|---------------|---------------------|-----------------|------|
| RF            | 0.0000 (0.0000)     | 0.2884 (0.0066) | 2    |
| CART          | 0.2484 (0.0040)     | 0.3216 (0.0077) | 4    |
| SVMLinear     | 0.2313 (0.0050)     | 0.3652 (0.0072) | 7    |
| SVMRadial     | 0.0857 (0.0072)     | 0.3048 (0.0074) | 3    |
| LDA           | 0.3265 (0.0032)     | 0.3844 (0.0078) | 9    |
| NNET          | 0.2277 (0.0057)     | 0.3272 (0.0078) | 5    |
| Multinomial   | 0.2770 (0.0031)     | 0.3692 (0.0072) | 8    |
| KNN           | 0.2860 (0.0069)     | 0.3592 (0.0080) | 6    |
| U-MBACT       | 0.1938 (0.0024)     | **0.2784 (0.0072)** | 1    |

Table 6: Forensic Glass Classification Results. Average error rates as defined in [26] and standard errors (in parentheses) for each procedure under comparison. The abbreviations in the first column are and their meanings are listed in the first paragraph of Section 5.1.

| Procedure     | Training Error Rate | Test Error Rate | Rank |
|---------------|---------------------|-----------------|------|
| RF            | 0.0000 (0.0000)     | 0.07756 (0.00328) | 7    |
| CART          | 0.06183 (0.00226)   | 0.10400 (0.00299) | 10   |
| SVMLinear     | 0.02233 (0.00233)   | 0.05933 (0.00332) | 4    |
| SVMRadial     | 0.02817 (0.00304)   | 0.08089 (0.00339) | 8    |
| QDA           | 0.02850 (0.00141)   | **0.03756 (0.00215)** | 1    |
| LDA           | 0.03233 (0.00150)   | 0.05222 (0.00275) | 2    |
| NNET          | 0.00517 (0.00203)   | 0.05978 (0.00410) | 5    |
| Multinomial   | 0.01567 (0.00194)   | 0.05556 (0.00380) | 3    |
| KNN           | 0.07683 (0.00245)   | 0.09956 (0.00325) | 9    |
| U-MBACT       | 0.04067 (0.00179)   | 0.07400 (0.00267) | 6    |

Table 7: Seeds Data set Results. Average error rates as defined in [26] and standard errors (in parentheses) for each procedure under comparison. The abbreviations in the first column are and their meanings are listed in the first paragraph of Section 5.1.

the control when flying the F16 aircraft based on attributes that describe the status of the aircraft. Triazines data set was used to study the quantitative structure-activity relationship of 2,3-diamino-6,6-dimethyl-5-phenyl-dihydrotriazine derivatives and their inhibition of dihydrofolate reductase (see [Hirst et al., 1994] for details). The data set Wisconsin breast cancer
has 32 predictors and a response which is either time to recurrence if a breast cancer patient returns due to a recurrence of the disease, or time to being free of cancer if the disease does not recur. Abalone data set relates the age of abalone to the attributes of a mollusk such as gender, weight, shell weight, and so forth. An accurate measure of the age involves observing the number of rings which is done after cutting the shell through the cone. A summary of the data sets is described in Table 8.

| Data set            | Ailerons | Triazines | Wisconsin | Abalone |
|---------------------|----------|-----------|------------|---------|
| Predictors(Numeric, Nominal) | 40(40, 0) | 60(60,0) | 32(32,0) | 8(7,1)  |
| Training Instances  | 3577     | 100       | 130        | 1000    |
| Test Instances      | 3577     | 86        | 64         | 3177    |
| Number of Replications Done | 10       | 50        | 50         | 25      |

Table 8: Description of Ordered Real Data Sets.

The response vector for each data set is discretized in order to obtain a 5-class ordered classification problem. The results in Table 9 are the average mean absolute deviations (MADs) (28) and standard errors on the test data sets over the number of replications mentioned in the last row of Table 8.

| Methodology | Wisconsin | Abalone | Triazines | Ailerons |
|-------------|-----------|---------|-----------|----------|
| RF          | 1.3741 (0.1047) | 0.2490 (0.0056) | 0.6759 (0.0559) | 0.4237 (0.0070) |
| CART        | 1.5729 (0.2757) | 0.2707 (0.0101) | 0.7608 (0.0879) | 0.6395 (0.0248) |
| SVMLinear   | 1.4265 (0.1078) | 0.2575 (0.0121) | 0.7401 (0.0780) | 0.4307 (0.0071) |
| SVMRadial   | 1.3731 (0.1669) | 0.2637 (0.0079) | 0.7097 (0.0574) | 0.8852 (0.0203) |
| NNET        | 1.4911 (0.2199) | **0.2323 (0.0075)** | 0.7645 (0.0812) | 0.4783 (0.1097) |
| Multinom    | 1.4098 (0.1059) | 0.2388 (0.0058) | 0.7553 (0.0710) | 0.4117 (0.0053) |
| KNN         | 1.3731 (0.1243) | 0.2613 (0.0067) | 0.7630 (0.0764) | 1.2224 (0.0170) |
| O-MBACT     | **1.1527 (0.0963)** | 0.2574 (0.0075) | **0.6519 (0.0599)** | **0.4042 (0.0036)** |
| U-MBACT     | 1.3440 (0.1137) | 0.2468 (0.0073) | 0.7076 (0.0617) | 0.6202 (0.0190) |

Table 9: Results of Ordered Real Data Sets. This table shows MADs defined in (28), and standard errors (in parenthesis) on test data sets over the number of replications mentioned in the last row of Table 8.

8 Conclusions

In this article, we have introduced Bayesian classification methodologies U-MBACT: Unordered Multiclass Bayesian Additive Classification Trees and O-MBACT: Ordered Multiclass Bayesian Additive Classification Trees as
multiclass classification (i.e., the number of classes $K \geq 3$) extension of BART: Bayesian Additive Regression Trees, and binary BART [Chipman et al., 2010] [Zhang and Härdle, 2010]. While U-MBACT can be used as a general purpose multiclass classifier, O-MBACT is suitable for modeling response vectors which exhibit inherent ordering.

Our methodologies are nonparametric and perform particularly well when the relationship between the predictors and the response is nonlinear. Across a range of simulation settings and real data illustrations, U-MBACT and O-MBACT are shown as having good predictive performances comparable to existing powerful classification methodologies and procedures. Aside from the competitive predictive performances exhibited by U-MBACT and O-MBACT, other desirable features of our procedures include retrieval of variable importance measures, and robustness to the existence of unimportant predictors. Through simulation studies and real data applications, we have demonstrated that U-MBACT and O-MBACT are important additions to multiclass classification toolbox.

References

Reshu Agarwal, Pritam Ranjan, and Hugh Chipman. A new bayesian ensemble of trees classifier for identifying multi-class labels in satellite images. 2013.

James H Albert and Siddhartha Chib. Bayesian analysis of binary and poly-chotomous response data. *Journal of the American Statistical Association*, 88(422):669–679, 1993.

Erin L Allwein, Robert E Schapire, and Yoram Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *The Journal of Machine Learning Research*, 1:113–141, 2001.

Vassilis Athitsos and Stan Sclaroff. Boosting nearest neighbor classifiers for multiclass recognition. In *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, pages 45–45. IEEE, 2005.

K. Bache and M. Lichman. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml
Stephen D Bay. Combining nearest neighbor classifiers through multiple feature subsets. In *Proceedings of the Fifteenth International Conference on Machine Learning*, volume 3, pages 37–45. Citeseer, 1998.

Gérard Biau. Analysis of a random forests model. *The Journal of Machine Learning Research*, 9(8888):1063–1095, 2012.

Gérard Biau, Luc Devroye, and Gábor Lugosi. Consistency of random forests and other averaging classifiers. *The Journal of Machine Learning Research*, 9:2015–2033, 2008.

L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Leo Breiman. Some infinity theory for predictor ensembles. Technical report, Technical Report 579, Statistics Dept. UCB, 2000.

Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. Chapman & Hall/CRC, 1984.

Małgorzata Charytanowicz, Jerzy Niewczas, Piotr Kulczycki, Piotr A Kowalski, Szymon Lukasik, and Sławomir Żak. Complete gradient clustering algorithm for features analysis of x-ray images. In *Information Technologies in Biomedicine*, pages 15–24. Springer, 2010.

H.A. Chipman, E.I. George, and R.E. McCulloch. Bart: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298, 2010.

Hugh A Chipman, Edward I George, and Robert E McCulloch. Bayesian cart model search. *Journal of the American Statistical Association*, 93 (443):935–948, 1998.

Wei Chu and Zoubin Ghahramani. Gaussian processes for ordinal regression. In *Journal of Machine Learning Research*, pages 1019–1041, 2005.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967.
Thomas M. Cover. Historical comments on nearest neighbor pattern classification. *Citation Classics*, 13, 20, March 1982.

T.G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *arXiv preprint cs/9501101*, 1995a.

Thomas G. Dietterich and Ghulum Bakiri. Solving multiclass learning problems via error-correcting output codes. *arXiv preprint cs/9501101*, 1995b.

Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. Wiley-interscience, 2012.

Ian W Evett and EJ Spiehler. Rule induction in forensic science. *KBS in Government, Online Publications*, pages 107–118, 1987.

Mathieu Fauvel, Jocelyn Chanussot, and Jon Atli Benediktsson. Evaluation of kernels for multiclass classification of hyperspectral remote sensing data. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 2, pages II–II. IEEE, 2006.

Evelyn Fix and Joseph L Hodges Jr. Discriminatory analysis-nonparametric discrimination: Small sample performance. Technical report, DTIC Document, 1952.

Y. Freund, R.E. Schapire, et al. Experiments with a new boosting algorithm. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*, pages 148–156. MORGAN KAUFMANN PUBLISHERS, INC., 1996.

Jerome H Friedman. Multivariate adaptive regression splines. *The Annals of Statistics*, pages 1–67, 1991.

João Gama, Ricardo Rocha, and Pedro Medas. Accurate decision trees for mining high-speed data streams. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 523–528. ACM, 2003.

John Geweke, Michael Keane, and David Runkle. Alternative computational approaches to inference in the multinomial probit model. *The Review of Economics and Statistics*, pages 609–632, 1994.
T. Hastie and R. Tibshirani. Classification by pairwise coupling. *The annals of statistics*, 26(2):451–471, 1998.

Trevor Hastie and Robert Tibshirani. Discriminant analysis by gaussian mixtures. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 155–176, 1996.

Trevor Hastie, Robert Tibshirani, and J Jerome H Friedman. *The elements of statistical learning*, volume 1. Springer New York, 2001.

Jonathan D Hirst, Ross D King, and Michael JE Sternberg. Quantitative structure-activity relationships by neural networks and inductive logic programming. ii. the inhibition of dihydrofolate reductase by triazines. *Journal of Computer-Aided Molecular Design*, 8(4):421–432, 1994.

Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2): 415–425, 2002.

S Sathiya Keerthi, KB Duan, SK Shevade, and Aun Neow Poo. A fast dual algorithm for kernel logistic regression. *Machine Learning*, 61(1-3): 151–165, 2005.

J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu, C. Peterson, et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature medicine*, 7(6):673–679, 2001.

Yoonkyung Lee, Grace Wahba, and Steven A Ackerman. Cloud classification of satellite radiance data by multicategory support vector machines. *Journal of Atmospheric and Oceanic Technology*, 21(2):159–169, 2004.

Tao Li, Chengliang Zhang, and Mitsunori Oghara. A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. *Bioinformatics*, 20(15):2429–2437, 2004.

MT Pratola, H Chipman, J Gattiker, D Higdon, R McCulloch, and W Rust. Parallel bayesian additive regression trees. *http://arxiv.org/abs/1309.1906*, 2012.
Jason DM Rennie and Ryan Rifkin. Improving multiclass text classification with the support vector machine. 2001.

Robert E Schapire and Yoram Singer. Improved boosting algorithms using confidence-rated predictions. In Proceedings of the Eleventh Annual Conference on Computational Learning Theory, pages 80–91. ACM, 1998.

Martin A Tanner and Wing Hung Wong. The calculation of posterior distributions by data augmentation. Journal of the American Statistical Association, 82(398):528–540, 1987.

Vladimir Vapnik. The nature of statistical learning theory. Springer, 1999.

Alexander Vergara, Shankar Vembu, Tuba Ayhan, Margaret A Ryan, Margie L Homer, and Ramón Huerta. Chemical gas sensor drift compensation using classifier ensembles. Sensors and Actuators B: Chemical, 2012.

Ting-Fan Wu, Chih-Jen Lin, and Ruby C. Weng. Probability estimates for multi-class classification by pairwise coupling. J. Mach. Learn. Res., 5:975–1005, dec 2004. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1005332.1016791.

Junni L Zhang and Wolfgang K Härdle. The bayesian additive classification tree applied to credit risk modelling. Computational Statistics & Data Analysis, 54(5):1197–1205, 2010.

Tong Zhang. Statistical analysis of some multi-category large margin classification methods. The Journal of Machine Learning Research, 5:1225–1251, 2004.