Statistics and Its Interface Volume 2 (2009) 361–368

Boosting on the functional ANOVA decomposition

Yongdai Kim*, Yuwon Kim, Jinseog Kim,
Sangin Lee, and Sunghoon Kwon

A boosting algorithm on the functional ANOVA decomposition, called ANOVA boosting, is proposed. The main idea of ANOVA boosting is to estimate each component in the functional ANOVA decomposition by combining many base (weak) learners. A regularization procedure based on the L_1 penalty is proposed to give a componentwise sparse solution and an efficient computing algorithm is developed. Simulated as well as bench mark data sets are analyzed to compare ANOVA boosting and standard boosting. ANOVA boosting improves prediction accuracy as well as interpretability by estimating the components directly and providing componentwisely sparser models.

Keywords and phrases: Functional ANOVA decomposition, Boosting, Variable selection.

1. INTRODUCTION

Given an output $y \in \mathcal{Y}$ and its corresponding input $\mathbf{x} = (x^{(1)}, \ldots, x^{(p)}) \in \mathcal{X}$, suppose we are interested in a functional relationship $f : \mathcal{X} \to \mathcal{Y}$ between $\mathbf{x}$ and y. When the dimension of the input (i.e. p) is high, one has a lot of difficulties in estimating and interpreting f. One of the most important learning methods for high dimensional data is a boosting method, which constructs a strong learner by combining many base (weak) learners. The boosting method has shown great success in statistics and machine learning areas for their significant improvement in prediction accuracy. Since [1] introduced the first boosting algorithm – AdaBoost, various extensions have been proposed by [2] and [3].

In this paper, we develop a way of using a boosting algorithm to estimate the components in the functional ANOVA decomposition, which is given as

$$f(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x}) + \cdots + f_K(\mathbf{x}) \tag{1}$$

where the components $f_k(\mathbf{x})$ depend only on low dimensional elements of an input vector $\mathbf{x}$. The main idea of the proposed boosting algorithm is to estimate each component f_j, $j = 1, \ldots, K$ by combining base learners. We call the proposed boosting method "ANOVA boosting." First, we propose sets of base learners for the components in the functional ANOVA decomposition to make the model identifiable. In particular, we use stumps (decision trees with only two terminal nodes) as base learners for main effect terms and their tensor products as base learners for interaction effect terms. Second, we develop a regularization procedure which gives a componentwisely sparse solution. Finally, we implement an efficient computational algorithm.

An advantage of ANOVA boosting over standard boosting methods is that ANOVA boosting can estimate and identify important components and their influence to the output simultaneously. In contrast, as [3] explained, standard boosting methods estimate only the highest order interaction components, and so estimating lower order components requires additional post-processing procedures. See, also, [4]. This advantage of ANOVA boosting makes it possible to select (or delete) relevant (or irrelevant) input variables. When the dimension of input is high, the final estimated model of a standard boosting method includes many noisy components and we need to identify which components are real signals and which are noises. Since ANOVA boosting can estimate each component simultaneously, we can easily develop a method which can identify signal and noisy components in the estimated model. For this purpose, we develop a componentwise sparse regularization procedure called the *componentwisely adaptive L_1 penalty*, which is motivated by the adaptive lasso by [5].

There are several modified boosting algorithms which give sparser solutions than standard boosting. [6] developed a similar boosting algorithm for the generalized additive model, and [7] proposed a boosting method called sparser boosting which yields a sparser solution than standard boosting. ANOVA boosting can estimate higher order interaction terms while the algorithm of [6] can estimate only main effect terms. Also, ANOVA boosting also gives a componentwisely sparser solution in contrast to the sparser boosting of [7] which only gives a sparser solution in terms of base learners. That is, important components can be selected by ANOVA boosting but not by sparser boosting.

ANOVA boosting has several advantages over the kernel based method for the functional ANOVA decomposition. [8] used the kernel machine for the functional ANOVA decomposition to improve the interpretability, and their idea has

*Corresponding author.

been studied and extended by [9], [10] and [11]. However, the kernel machine has a problem with categorical inputs since the Gram matrix can be singular and so the algorithm fails to converge. Also, when the dimension of the input is high, the computational cost for inverting the Gram matrix is expensive. In contrast, categorical inputs can be processed easily and computation is simpler since no matrix inversion is required in ANOVA boosting.

The paper is organized as follows. Section 2 presents various ingredients of ANOVA boosting such as model, the choice of base learners and regularization procedure. In Section 3, a computational algorithm is presented. Simulated as well as real datasets are analyzed in Section 4. Concluding remarks follow in Section 5.

2. ANOVA BOOSTING

2.1 Model

Let $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ be n input-output pairs of a training dataset where $\mathbf{x}_i \in \mathcal{X} \subset R^p$ and $y_i \in \mathcal{Y}$, which are assumed to be a random sample from a probability measure P on $\mathcal{X} \times \mathcal{Y}$. Let $\mathbf{x}_i = (x_i^{(1)}, \ldots, x_i^{(p)})$ where $x_i^{(j)} \in \mathcal{X}_j \subset R$ and $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_p$. Let $\mathcal{F}$ be a given set of functions on R^p and let $l : \mathcal{Y} \times R \to R$ be a loss function. The objective of statistical learning is to find a function $f^* \in \mathcal{F}$ which minimizes $\mathrm{E}_P(l(Y, f(\mathbf{X})))$ among $f \in \mathcal{F}$.

The functional ANOVA decomposition of f is

$$f(\mathbf{x}) = \beta_0 + \sum_{j=1}^{p} f_j(x^{(j)}) + \sum_{j<k} f_{jk}(x^{(j)}, x^{(k)}) + \cdots$$

where β_0 is a constant, f_j are the main effect components, f_{jk} are second order interaction components and so on. For simplicity, we consider the model truncated up to the second order interaction components for f. That is, $\mathcal{F}$ consists of functions having the form

$$f(\mathbf{x}) = \beta_0 + \sum_{j=1}^{p} f_j(x^{(j)}) + \sum_{j<k} f_{jk}(x^{(j)}, x^{(k)}).$$

Given predefined probability measures μ_j on $\mathcal{X}_j$, let $\mathcal{F}_j$ be the set of functions f_j in $L_2(\mu_j)$ satisfying

$$\int_{\mathcal{X}_j} f_j(x^{(j)})\mu_j(dx^{(j)}) = 0 \quad \text{for } f_j \in \mathcal{F}_j, \tag{2}$$

and let $\mathcal{F}_{jk}$ be the set of functions f_{jk} in $L_2(\mu_j \times \mu_k)$ satisfying

$$\begin{aligned} &\int_{\mathcal{X}_j} f_{jk}(x^{(j)}, x^{(k)})\mu_j(dx^{(j)}) = 0, \\ &\int_{\mathcal{X}_k} f_{jk}(x^{(j)}, x^{(k)})\mu_k(dx^{(k)}) = 0. \end{aligned} \tag{3}$$

Then, we can write

$$\mathcal{F} = \{1\} \oplus \left[\bigoplus_{j=1}^{p} \mathcal{F}_j\right] \oplus \left[\bigoplus_{j<k} \mathcal{F}_{jk}\right]$$

where all subspaces $\{1\}$, $\mathcal{F}_j$, $\mathcal{F}_{jk}$, $j = 1, \ldots, p$, $j < k$ are orthogonal on $L_2(\mu)$ where $\mu = \prod_{j=1}^{p} \mu_j$, and hence all components are identifiable.

2.2 Choice of base learners

The basic idea of ANOVA boosting is to estimate each component (i.e. f_js and f_{jk}s) by a linear combination of base learners. For this, we have to choose sets of base learners $\mathcal{G}_j$ and $\mathcal{G}_{jk}$ for the components f_j and f_{jk}, respectively.

For $\mathcal{G}_j$, we use the set of decision trees with only two terminal nodes split by the variable $x^{(j)}$. For the side condition, we enforce

$$\int_{\mathcal{X}_j} g_j(x^{(j)})\mu_j(dx^{(j)}) = 0 \tag{4}$$

for $g_j \in \mathcal{G}_j$, and hence the resulting f_j satisfies (2). For a continuous input variable, let $g_j(x^{(j)}) = \theta_L I(x^{(j)} \le s) + \theta_R I(x^{(j)} > s)$. To satisfy the side condition (4), we should have

$$\mu_j(x^{(j)} \le s)\theta_L + \mu_j(x^{(j)} > s)\theta_R = 0. \tag{5}$$

That is, we can choose the split value s freely, but the predictive values θ_L and θ_R should be selected to satisfy (5). Categorical inputs can be treated similarly.

For $\mathcal{G}_{jk}$, we use the tensor products of the base learners in $\mathcal{G}_j$ and $\mathcal{G}_k$. That is, we let $\mathcal{G}_{jk} = \mathcal{G}_j \otimes \mathcal{G}_k$. That is, for any $g_{jk} \in \mathcal{G}_{jk}$, there exist $g_j \in \mathcal{G}_j$ and $g_k \in \mathcal{G}_k$ such that $g_{jk}(x^{(j)}, x^{(k)}) = g_j(x^{(j)})g_k(x^{(k)})$. With $\mathcal{G}_{jk}$, the resulting f_{jk} automatically satisfies the identifiability condition (3). Note that g_{jk} have the form of

$$\begin{aligned} g_{jk}(x^{(j)}, x^{(k)}) = {} & \theta_{LL} I(x^{(j)} \le s_j, x^{(k)} \le s_k) \\ & + \theta_{LR} I(x^{(j)} \le s_j, x^{(k)} > s_k) \\ & + \theta_{RL} I(x^{(j)} > s_j, x^{(k)} \le s_k) \\ & + \theta_{RR} I(x^{(j)} > s_j, x^{(k)} > s_k) \end{aligned}$$

with the identifiability condition

$$\begin{aligned} &\mu_j(x^{(j)} \le s_j)\theta_{LL} + \mu_j(x^{(j)} > s_j)\theta_{RL} = 0, \\ &\mu_j(x^{(j)} \le s_j)\theta_{LR} + \mu_j(x^{(j)} > s_j)\theta_{RR} = 0, \\ &\mu_k(x^{(k)} \le s_k)\theta_{LL} + \mu_k(x^{(k)} > s_k)\theta_{LR} = 0, \\ &\mu_k(x^{(k)} \le s_k)\theta_{RL} + \mu_k(x^{(k)} > s_k)\theta_{RR} = 0. \end{aligned} \tag{6}$$

It is easy to see that one of θ_{LL}, θ_{LR}, θ_{RL} and θ_{RR} uniquely defines the other three values. In this view, we may say that the degree of freedom of g_{jk} is the same as that of g_j and g_k.

For the choice of μ_j, the most natural one is P_j, the marginal probability measure of $x^{(j)}$, which are unknown. We estimate $P_j(x^{(j)} \le s)$ by their empirical counterparts $\sum_{i=1}^n I(x_i^{(j)} \le s)/n$.

2.3 Regularization

In ANOVA boosting, the final model has the form

$$f(\mathbf{x}) = \beta_0 + f_0(\mathbf{x}) \tag{7}$$

where

$$f_0(\mathbf{x}) = \sum_{j=1}^{p} \sum_{g \in \mathcal{G}_j} \beta_g g(x^{(j)}) + \sum_{j<k} \sum_{g \in \mathcal{G}_{jk}} \beta_g g(x^{(j)}, x^{(k)}). \tag{8}$$

First, we need to control the norm of base learners to make βs estimable. For this, we let

$$\sup_{x^{(j)} \in \mathcal{X}_j} |g(x^{(j)})| \le 1 \quad \text{for all } g \in \mathcal{G}_j,$$

and

$$\sup_{(x^{(j)}, x^{(k)}) \in \mathcal{X}_j \times \mathcal{X}_k} |g(x^{(j)}, x^{(k)})| \le 1 \quad \text{for all } g \in \mathcal{G}_{jk},$$

for all j, k.

Second, we need a regularization procedure for βs to avoid overfitting and ensure componentwise sparsity. For this, we propose to use the componentwisely adaptive L_1 constraint given as follows. Let $\beta_g^{(0)}$ be the initial estimates obtained by a standard boosting method, and let

$$w_j = \left(\sum_{g \in \mathcal{G}_j} |\beta_g^{(0)}| \right)^{\gamma} \quad \text{and} \quad w_{jk} = \left(\sum_{g \in \mathcal{G}_{jk}} |\beta_g^{(0)}| \right)^{\gamma} \tag{9}$$

for some $\gamma \ge 0$. Then, the componentwisely adaptive L_1 constraint is defined by

$$\sum_{j=1}^{p} \frac{\sum_{g \in \mathcal{G}_j} |\beta_g|}{w_j} + \sum_{j<k} \frac{\sum_{g \in \mathcal{G}_{jk}} |\beta_g|}{w_{jk}} \le \lambda \tag{10}$$

where γ and λ are regularization parameters which can be selected by using test samples or cross-validation. The proposed constraint (10) is motivated by the adaptive Lasso by [5]. Finally, we propose to estimate the βs by minimizing the empirical risk $C_n(\beta_0, f_0) = \sum_{i=1}^n l(y_i, f(\mathbf{x}_i))$ with the constraint (10).

Remark. It would be possible to use different regularization parameters for the components. That is, we let $\frac{\sum_{g \in \mathcal{G}_j} |\beta_g|}{w_j} \le \lambda_j$ and $\frac{\sum_{g \in \mathcal{G}_{jk}} |\beta_g|}{w_{jk}} \le \lambda_{jk}$. This is useful when we have prior information about the importance of the components. For example, to incorporate the prior information that the main effect components are more important than higher order interaction component, we let $\lambda_{jk} \le \lambda_j$. The algorithm developed in the next section can be modified easily for this case.

3. COMPUTATIONAL ALGORITHM

Given g in $\mathcal{G} = \cup_j \mathcal{G}_j \bigcup \cup_{j<k} \mathcal{G}_{jk}$, let $h_g(\mathbf{x}) = \lambda w_g g(\mathbf{x})$ where $w_g = w_k$ if $g \in \mathcal{G}_k$ and $w_g = w_{jk}$ if $g \in \mathcal{G}_{jk}$. Then, we can rewrite (8) by

$$f_0(\mathbf{x}) = \sum_{j=1}^{p} \sum_{g \in \mathcal{G}_j} \theta_g h_g(x^{(j)}) + \sum_{j<k} \sum_{g \in \mathcal{G}_{jk}} \theta_g h_g(x^{(j)}, x^{(k)}) \tag{11}$$

and the constraint (10) becomes $\sum_{g \in \mathcal{G}} |\theta_g| \le 1$ where $\theta_g = \beta_g/(\lambda w_g)$. Hence, for fixed β_0, we can use of the MarginBoost.L_1 algorithm of [12]. However, there is a room to improve the MarginBoost.L_1 algorithm. The final estimated model from the algorithm may be less sparse than it should be. This is because the MarginBoost.L_1 algorithm keeps adding base learners to update the model. Hence, when unnecessary base learners are added in the early stage of iteration, they are never deleted from the estimated model. This may not be a serious problem for prediction accuracy, but it affects largely to the sparsity of the estimated model. For resolving this problem, we employ a deletion step after each iteration. In the deletion step, some base learners in the model are deleted. By doing so, we improve the convergence speed and ensure the sparsity of the final estimated model.

The idea of the deletion step is as follows. After m iterations, there are at most m many base learners whose coefficients are not zero. Then, we move the non-zero coefficients to the their gradient direction until either a non-zero coefficient becomes zero or the optimization criterion is satisfied. To explaining more details, given a current estimated model f_0, let $\mathcal{G}^+ = \{g \in \mathcal{G} : \theta_g > 0\}$. That is, $f_0(\mathbf{x}) = \sum_{g \in \mathcal{G}^+} \theta_g h_g(\mathbf{x})$. Since the set of base learners is negation closed (i.e. if $g \in \mathcal{G}$, then $-g \in \mathcal{G}$) we assume that all the non-zero coefficients θ_g are positive and $\sum_{g \in \mathcal{G}^+} \theta_g \le 1$. Let $\nabla_g = \partial C_n(\beta_0, f_0)/\partial \theta_g$ for $g \in \mathcal{G}^+$, and let $\nabla_g^* = \nabla_g - \sum_{g \in \mathcal{G}^+} \nabla_g / \#\mathcal{G}^+$ where $\#\mathcal{G}^+$ is the cardinality of $\mathcal{G}^+$. Consider new coefficients $\theta_g(v) = \theta_g - v\nabla_g^*$ for some $v \ge 0$. Since $\sum_{g \in \mathcal{G}^+} \nabla_g^* = 0$, we have $\theta_g(v) \ge 0$ for all $g \in \mathcal{G}^+$ and $\sum_{g \in \mathcal{G}^+} \theta_g(v) \le 1$ as long as $0 \le v \le \eta$ where $\eta = \min\{\theta_g/\nabla_g^* : \nabla_g^* > 0\}$. We update θ_g by $\theta_g(\hat{v})$ where $\hat{v} = \operatorname{argmin}_{v \in [0,\eta]} C_n(\beta_0, f_{0v})$ where $f_{0v}(\mathbf{x}) = \sum_{g \in \mathcal{G}^+} \theta_g(v) h_g(\mathbf{x})$. When $\hat{v} = \eta$, at least one of θ_g, $g \in \mathcal{G}^+$ becomes 0 and hence the corresponding base learner is deleted from the estimated model. Note that the deletion step always reduces the empirical risk, and hence the algorithm also converges to the global optimum as the MarginBoost.L_1 algorithm does under regularity conditions.

The MarginBoost.L_1 algorithm and deletion step, which we call the *ANOVA boosting* algorithm, is presented in Fig. 1. Figure 2 compares the convergence speeds of the ANOVA boosting and MarginBoost.L_1 algorithms with a simulated data set from the model 1 in Section 4.1. It is clear that the ANOVA boosting algorithm converges much faster than the MarginBoost.L_1 algorithm. The training error measured by the empirical risk (the average loss over

1. Let β_0 and f_0 be the initial estimates from a standard boosting algorithm.
2. Let $\lambda_j = |f_j|^\gamma$ and $\lambda_{jk} = |f_{jk}|^\gamma$ where
$$|f_j| = \sum_{g \in \mathcal{G}_j} |\beta_g| \text{ and } |f_{jk}| = \sum_{g \in \mathcal{G}_{jk}} |\beta_g|.$$
3. Repeat until convergence
   - Addition step: MarginBoost.L_1 algorithm
     (a) Find $\hat{g}$ in $\mathcal{G}$ which minimizes $\sum_{i=1}^n h_g(\mathbf{x}_i) z_i$ where
     $$z_i = \left. \frac{\partial l(y_i, a)}{\partial a} \right|_{a=f(\mathbf{x}_i)}.$$
     (b) Find $\hat{\alpha}$ by
     $$\hat{\alpha} = \text{argmin}_{\alpha \in [0,1]} C_n(\beta_0, (1-\alpha) f_0 + \alpha h_{\hat{g}}).$$
     (c) Update $f_0 = (1-\hat{\alpha}) f_0 + \hat{\alpha} h_{\hat{g}}$.
   - Deletion step
     (a) Let $f_0(\mathbf{x}) = \sum_{g \in \mathcal{G}^+} \theta_g h_g(\mathbf{x})$ where $\mathcal{G}^+ = \{g : \theta_g > 0\}$.
     (b) Let $\nabla_g = \sum_{i=1}^n h_g(\mathbf{x}_i) z_i$ for $g \in \mathcal{G}^+$ and let $\nabla_g^* = \nabla_g - \sum_{g \in \mathcal{G}^+} \nabla_g / \#\mathcal{G}^+$
     (c) Find $\hat{v}$ by
     $$\hat{v} = \text{argmin}_{v \in [0,\eta]} C_n(\beta_0, f_{0v})$$
     where $f_{0v}(\mathbf{x}) = \sum_{g \in \mathcal{G}^+} (\theta_g - v \nabla_g^*) h_g(\mathbf{x})$ and $\eta = \min\{\theta_g / \nabla_g^* : \nabla_g^* > 0\}$.
     (d) Update $f_0 = f_{0\hat{v}}$.
   - Update β_0
     (a) Update $\beta_0 = \text{argmin}_{\gamma \in R} C(\gamma, f_0)$.

Figure 1. The ANOVA boosting algorithm.

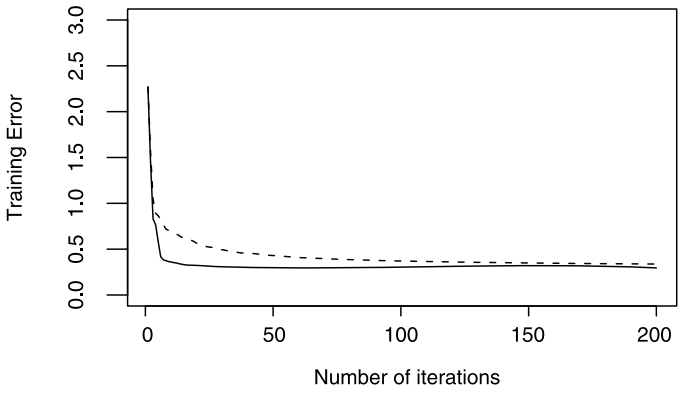

Figure 2. Training error (empirical risk) curves on the number of iterations for the MarginBoost.L_1 (dashed line) and ANOVA boosting algorithms (solid line).

the training samples) achieves its minimum after around 25 iterations of the ANOVA boosting algorithm while the training error keeps decreasing even after 200 iterations of the MarginBoost.L_1 algorithm.

The ANOVA boosting algorithm always converges since the empirical risk $C_n(\beta_0, f_0)$ always decreases after each iteration. The ANOVA boosting algorithm differs from standard boosting algorithms such as AdaBoost [1] and gradient boosting [3] which need a stopping rule to avoid overfitting. This is an another advantage of the ANOVA boosting algorithm.

4. EXPERIMENTS

We compare empirical performance of ANOVA boosting with a standard boosting method in terms of prediction accuracy and variable selectivity. For a standard boosting method, we use the MarginBoost.L_1 of Mason et al. (2000). For variable selectivity, we compute the relative frequencies of components selected. The regularization parameters γ and λ are selected by 5-fold cross validation. We search the optimal value of γ only on $\{0, 0.5, 1\}$ to save computing time.

4.1 Simulation

We consider the following four models for simulation. The first two models are regression problems and the last two models are logistic regression.

Model 1: The input vector $\mathbf{x}$ is generated from a 10 dimensional uniform distribution on $[0,1]^{10}$. For given $\mathbf{x}$, y is generated from the model $y = f(x) + \epsilon$, where

$$f(x) = 5g_1(x^{(1)}) + 3g_2(x^{(2)}) + 4g_3(x^{(3)}) + 6g_4(x^{(4)})$$

Table 1. Estimates of the error rate and sparsity (standard errors) in 100 simulations

	Method	MIS-rate	NNZ
Model 1	Boosting	1.2881 (0.0210)	9.94 (0.0239)
	ANOVA boosting	1.1155 (0.0146)	4.58 (0.0768)
Model 2	Boosting	0.1908 (0.0007)	49.81 (0.0466)
	ANOVA boosting	0.1641 (0.0015)	11.15 (0.2556)
Model 3	Boosting	0.2397 (0.0010)	9.78 (0.0628)
	ANOVA boosting	0.2253 (0.0011)	7.42 (0.1646)
Model 4	Boosting	0.1781 (0.0012)	12.61 (0.2755)
	ANOVA boosting	0.1606 (0.0007)	3.92 (0.1468)

and ϵ is a normal variate with mean 0 variance σ^2 which is selected to give the signal to noise ratio 3:1. Here,

$$g_1(t) = t; \qquad g_2(t) = (2t-1)^2; \qquad g_3(t) = \frac{\sin(2\pi t)}{2-\sin(2\pi t)}$$

$$g_4(t) = 0.1\sin(2\pi t) + 0.2\cos(2\pi t) + 0.3\sin^2(2\pi t) + 0.4\cos^3(2\pi t) + 0.5\sin^3(2\pi t).$$

This model is used by [9]. The model has only main effect components, and $x_5, \ldots, x_{10}$ are noisy input variables. We apply the boosting algorithms with the square error loss.

Model 2: Model 2 is the same as Model 1 except

$$f(x) = g_1(x^{(1)}) + g_2(x^{(2)}) + g_3(x^{(3)}) + g_4(x^{(4)}) + g_1(x^{(3)}x^{(4)}) + g_2\left(\frac{x^{(1)}+x^{(3)}}{2}\right) + g_3(x^{(1)}x^{(2)}).$$

That is, there are three interaction terms in the true model.

Model 3: The input vector $\mathbf{x}$ is generated from 10 dimensional multivariate norm distribution with mean 0 and variance matrix Σ, the off-diagonals of which are 0.2 and the diagonals are 1. For given $\mathbf{x}$, y is generated from the Bernoulli distribution with $\Pr(Y=1|\mathbf{x}) = \exp(f(\mathbf{x}))/(1+\exp(f(\mathbf{x})))$ where

$$f(\mathbf{x}) = \frac{4}{3}x_1 + \pi\sin(\pi x_2) + \frac{1}{10}x_3^5 + 3e^{-x_4^2/2} - 1.5.$$

The model has only main effect components, and $x_5, \ldots, x_{10}$ are noisy input variables. We apply the boosting algorithms with the negative log-likelihood loss.

Model 4: The input vector $\mathbf{x}$ is generated from 5 dimensional multivariate norm distribution with mean 0 and variance matrix Σ, the off-diagonals of which are 0.2 and the diagonals are 1. For given $\mathbf{x}$, y is generated from the Bernoulli distribution with $\Pr(Y=1|\mathbf{x}) = \exp(f(\mathbf{x}))/(1+\exp(f(\mathbf{x})))$ where

$$f(\mathbf{x}) = 2x_1 + \pi\sin(\pi x_1) + x_2 - 2x_2^3 + 4\exp(-2|x_1 - x_2|).$$

The model has 2 main effect components and one second order interaction component, and x_3, x_4, x_5 are noisy input variables.

Table 1 compares the prediction accuracy and sparsity. Sparsity is measured by the number of non-zero components. We simulate 100 data sets of size 250. The error rate is evaluated on 10,000 testing points. In the table, the MIS-rate is the average misclassification error rate on the test samples and the NNZ is the average number of non-zero components. From Table 1, we can see ANOVA boosting is more accurate and selects less components than the standard boosting. That is, ANOVA Boosting has superior prediction power as well as interpretability compared to the standard boosting. Better performance of ANOVA boosting is expected since the true models are sparse.

Table 2 shows the relative frequency of each variable appearing in the 100 estimated models, which shows that ANOVA boosting successively deletes many noisy components compared to the standard boosting.

4.2 Analysis of real data sets

We analyze the four real data sets which are available on the UCI machine learning repository. The description of the four data sets is presented in Table 3. In the table, Type represents if the data set is either a regression problem (R) or a classification problem (C). N.obs is the number of observations, Cont. means continuous type inputs and Categ. represents categorical inputs.

The main effect model as well as the second order interaction model are fitted. Table 4 summarizes the prediction accuracy as well as the sparsity of ANOVA boosting and the standard boosting on the six data sets. The error rates are calculated by the 10-fold cross-validation.

The results show that ANOVA boosting is consistently more accurate than the standard boosting in most cases (one exception for "Bupa" and main effect model). Also, ANOVA Boosting produces more sparse models than the standard boosting. In particular, for the data set "Sonar" with the second order interaction model, the ANOVA boosting model consists of only 25.7 components while the standard boosting model has 111 components (i.e. 75% reduction).

Table 2. The relative frequencies of appearance of components in the models chosen in 100 runs

Model 1	Method	X_1	X_2	X_3	X_4	Others
	Boosting	1.00	1.00	1.00	0.96	
	ANOVA Boosting	1.00	1.00	0.90	0.93	0.59
Model 2	Method	X_1	X_2	X_1X_2	Others	
	Boosting	1.00	1.00	0.80		
	ANOVA Boosting	1.00	1.00	0.41	0.12	
Model 3	Method	X_1	X_2	X_3	X_4	Others
	Boosting	1.00	1.00	1.00	0.99	
	ANOVA Boosting	1.00	1.00	1.00	1.00	0.10
Model 4	Method	$X_1 \sim X_4$	X_1X_2	X_1X_3	X_3X_4	Others
	Boosting	1.00	1.00	1.00	0.69	
	ANOVA Boosting	1.00	1.00	0.78	0.11	0.05

Table 3. Description of the six data sets

			Inputs	
Name	Type	N.obs	Cont.	Categ.
Bupa	C	345	6	0
Breast	C	286	3	6
Sonar	C	210	60	0
Housing	R	506	12	1

4.3 Illustration on the data set "Breast"

We investigate more about the components selected in the breast cancer data set. This data set includes 201 instances of one class (no-recurrence-events) and 85 instances of another class (recurrence-events). The instances are described by 9 attributes – $X1$: age, $X2$: menopause (lt40, he40, premeno), $X3$: tumor-size, $X4$: invasion node, $X5$: node-caps (yes or no), $X6$: degree of malignance, $X7$:breast location (left or right), $X8$: breast quad (left-up, left-low, right-up, right-low, central), $X9$: irradiated (yes or no).

Since the second order interaction model is better in prediction accuracy than the main effect model in Table 4, we present the results from the second order interaction model. Figure 3 gives the L_1 norms of the 12 selected components out of 45 candidate components. Among these, Fig. 4 shows the estimated functional forms of the first 6 components having the largest L_1 norms. There are three main effects and three second order interaction components. The risk of the recurrence of breast tumor increases as the deg-mailg, invnodes and tumor-size increase. Also, the three interaction components show that the location of the cancer are interacted with the status of menopause and age. These suggest that different treatments would be applied according to the age of a patient, status of menopause and location of the cancer.

Table 4. Estimates of the accuracies and number of non-zero components (standard errors) in the four data sets

Data	Model	Method	MIS-rate	NNZ
Bupa	Main effect	Boosting	0.2868 (0.0247)	6.0 (0.0000)
		ANOVA Boosting	0.2926 (0.0237)	6.0 (0.0000)
	Second order	Boosting	0.3362 (0.0223)	20.3 (0.3000)
		ANOVA Boosting	0.3187 (0.0175)	12.7 (1.0333)
Sonar	Main effect	Boosting	0.1583 (0.0235)	40.8 (1.7048)
		ANOVA Boosting	0.1529 (0.0193)	24.7 (1.0005)
	Second order	Boosting	0.1631 (0.0225)	111 (10.8443)
		ANOVA Boosting	0.1575 (0.0225)	25.7 (1.6401)
Breast	Main effect	Boosting	0.2494 (0.0130)	7.1 (0.5467)
		ANOVA Boosting	0.2449 (0.0094)	4.8 (0.5537)
	Second order	Boosting	0.2462 (0.0186)	21.2 (2.5638)
		ANOVA Boosting	0.2421 (0.0156)	14.7 (3.0112)
Housing	Main effect	Boosting	15.8973 (1.8412)	11.8 (0.1334)
		ANOVA Boosting	14.6608 (1.7641)	9.4 (0.4760)
	Second order	Boosting	14.6569 (1.8556)	39.1 (0.6904)
		ANOVA Boosting	13.4980 (1.4603)	23.3 (0.8171)

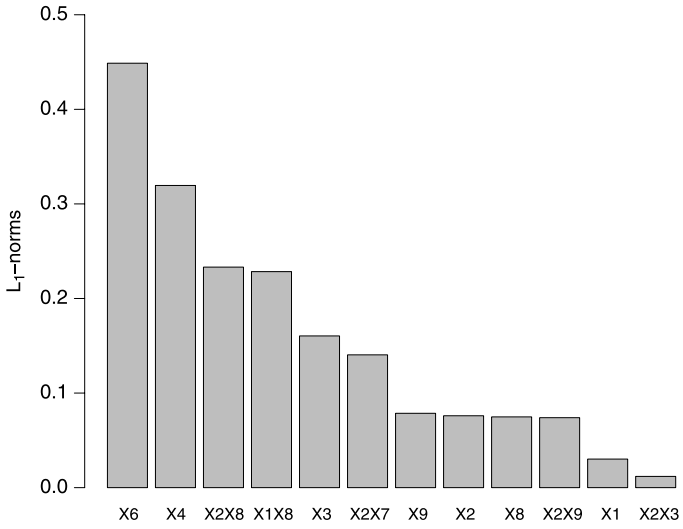

Figure 3. L_1 norms of the 12 selected components.

5. CONCLUDING REMARKS

By simulations and analysis of real data sets, we have illustrated that ANOVA boosting improves the interpretability of the standard boosting significantly by estimating the components directly and providing componentwisely sparser models without sacrificing prediction accuracy. Also, the newly proposed computational algorithm converges faster and can be applied to high dimensional data.

The final estimated components of ANOVA boosting are not smooth. This is because decision trees are used as base learners. If one wants smooth estimates, one can use smooth base learners such as the radial basis functions and smooth splines. As long as we have base learners for main effect components, base learners for higher order interactions can be constructed via the tensor product operation. See [13] for this approach. However, there is an advantage of using decision trees as base learners. ANOVA boosting is expected

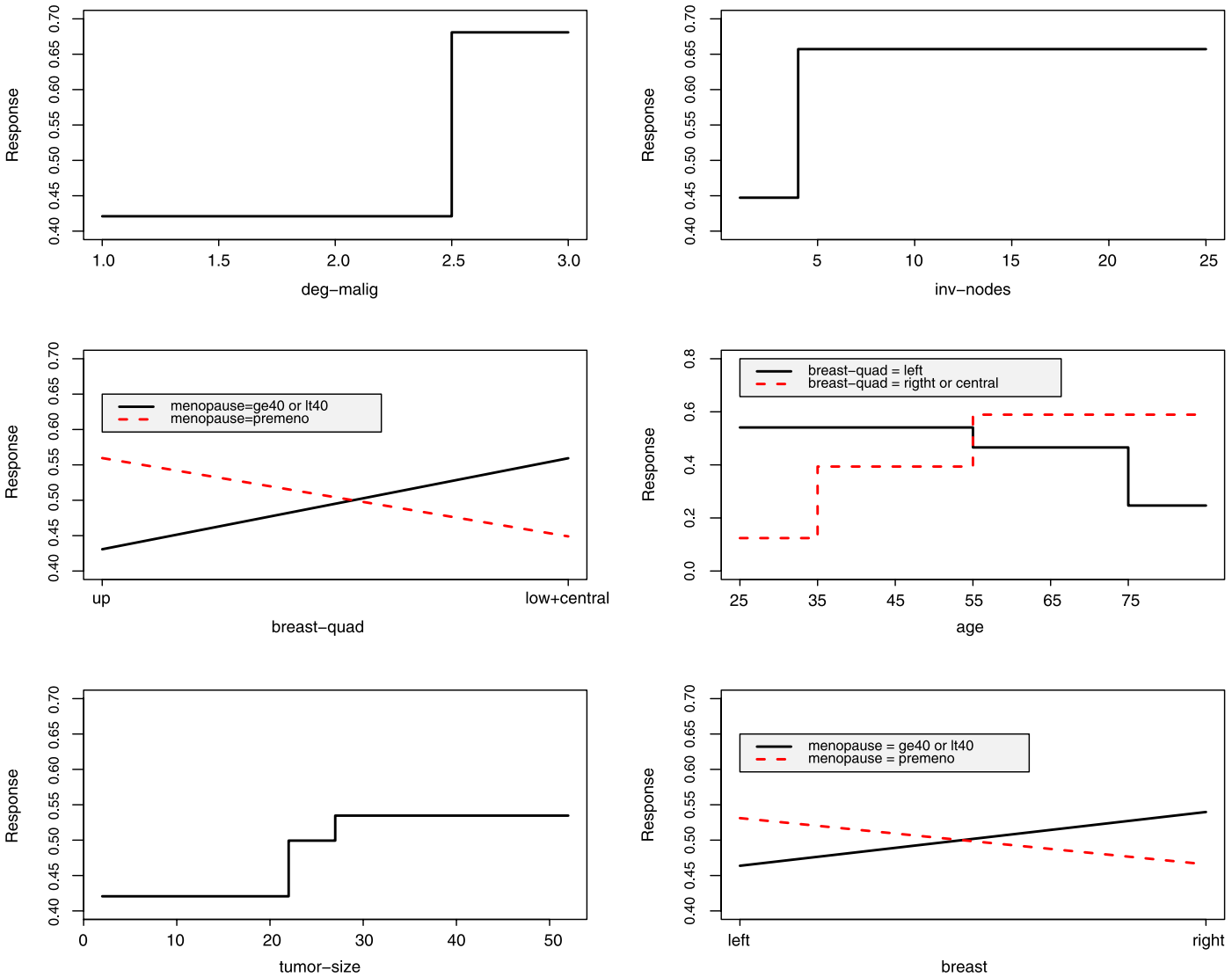

Figure 4. Estimated functional forms of the 6 components having the largest L_1 norm for the Breast Cancer data.

to be robust to input noise since decision trees are so. This is because decision trees are invariant to a monotone transformation of an input. So, in practice, we can use ANOVA boosting without preprocessing input variables.

ACKNOWLEDGEMENTS

This work was supported by the Korea Science and Engineering Foundation (KOSEF) grant funded by the Korea government (MEST) R01-2007-000-20045-0 and the Engineering Research Center of Excellence Program of Korea Ministry of Education, Science and Technology (MEST)/Korea Science and Engineering Foundation (KOSEF), grant number R11-2008-007-01002-0.

Received 26 May 2009

REFERENCES

[1] Freund, Y. and Schapire, R. (1997). *Journal of Computer and System Sciences* **55** 119–139. MR1473055

[2] Schapire, R. and Singer, Y. (1999). *Machine Learning* **37** 297–336.

[3] Friedman, J. H. (2001). *Annals of Statistics* **29** 1189–1232. MR1873328

[4] Friedman, J. H. and Popescu, B. E. (2005). Predictive learning via rule ensembles. Technical report, Stanford University.

[5] Zou, H. (2006). *Journal of the American Statistical Association* **101** 1418–1429. MR2279469

[6] Tutz, G. and Binder, H. (2006) *Biometrics* **62** 961–971. MR2297666

[7] Buhlmann, P. and Yu, B. (2006). *Journal of machine learning research* **7** 1001–1024. MR2274395

[8] Gunn, S. R. and Kandola, J. S. (2002). *Machine Learning* **48** 137–163.

[9] Lin, Y. and Zhang, H. (2003). Component Selection and Smoothing in Smoothing Spline Analysis of Variance Models. Technical Report 1072, Department of Statistics, University of Wisconsin-Madison.

[10] Lee, Y., Kim, Y., Lee, S., and Koo, J.-Y. (2006). *Biometrika* **93** 555–571. MR2261442

[11] Zhang, Z. and Lin, Y. (2006). *Statistica Sinica* **16** 1021–1041. MR2281313

[12] Mason, L., Baxter, J., Bartlett, P. L., and Frean, M. (2000). Functional gradient techniques for combining hypotheses. In Smola, A. J., Bartlett, P. L., Scholkopf, B., and Schuurmans, D. (eds.), Advances in Large Margin Classifiers, pp. 221–246. MIT press, Cambridge. MR1820960

[13] Zhang, H., Wahba, G., Lin, Y., Voelker, M., Ferris, M., Klein, R., and Klein, B. (2004). *Journal of the American Statistical Association* **99** 659–672. MR2090901

Yongdai Kim
Department of Statistics
Seoul National University
Korea
E-mail address: ydkim0903@gmail.com

Yuwon Kim
NHN Corp.
Korea
E-mail address: gary@stats.snu.ac.kr

Jinseog Kim
Department of Statistics
Dongguk University
Korea
E-mail address: jskim@stats.snu.ac.kr

Sangin Lee
Department of Statistics
Seoul National University
Korea
E-mail address: lsi44@statcom.snu.ac.kr

Sunghoon Kwon
Department of Statistics
Seoul National University
Korea
E-mail address: shkwon0522@gmail.com