BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

A boosting method for maximizing the partial area under the ROC curve

Osamu Komori*[1] and Shinto Eguchi[1,2]

Abstract

Background: The receiver operating characteristic (ROC) curve is a fundamental tool to assess the discriminant performance for not only a single marker but also a score function combining multiple markers. The area under the ROC curve (AUC) for a score function measures the intrinsic ability for the score function to discriminate between the controls and cases. Recently, the partial AUC (pAUC) has been paid more attention than the AUC, because a suitable range of the false positive rate can be focused according to various clinical situations. However, existing pAUC-based methods only handle a few markers and do not take nonlinear combination of markers into consideration.

Results: We have developed a new statistical method that focuses on the pAUC based on a boosting technique. The markers are combined componentially for maximizing the pAUC in the boosting algorithm using natural cubic splines or decision stumps (single-level decision trees), according to the values of markers (continuous or discrete). We show that the resulting score plots are useful for understanding how each marker is associated with the outcome variable. We compare the performance of the proposed boosting method with those of other existing methods, and demonstrate the utility using real data sets. As a result, we have much better discrimination performances in the sense of the pAUC in both simulation studies and real data analysis.

Conclusions: The proposed method addresses how to combine the markers after a pAUC-based filtering procedure in high dimensional setting. Hence, it provides a consistent way of analyzing data based on the pAUC from maker selection to marker combination for discrimination problems. The method can capture not only linear but also nonlinear association between the outcome variable and the markers, about which the nonlinearity is known to be necessary in general for the maximization of the pAUC. The method also puts importance on the accuracy of classification performance as well as interpretability of the association, by offering simple and smooth resultant score plots for each marker.

Background

The receiver operating characteristic (ROC) curve has been widely used in various scientific fields, in situations where the evaluation of discrimination performance is of great concern for the researchers. The area under the ROC curve (AUC) is the most popular metric because it has a simple probabilistic interpretation [1] and consists of two important rates used to assess classification performance: the true positive rate (TPR) and the false positive rate (FPR). The former is a probability of an affected subject being correctly judged as positive; the latter is that of an unaffected subject being improperly judged as positive. These two rates are shown to be more adequate to evaluate the classification accuracy than the odds ratio or relative risk [2]. However, the AUC has been severely criticized for inconsistency arising between statistical significance and the corresponding clinical significance when the usefulness of a new marker is evaluated [3]. Recently, Pencina et al. [4] propose a criterion termed integrated discriminant improvement and show the advantage over the AUC in the assessment of a new marker. In this context, the partial AUC (pAUC) is paid more attention than the AUC, especially in clinical settings where a low FPR or a high TPR is required [5-7].

Dodd and Pepe [8] propose a regression modeling framework based on the pAUC, and apply this framework

* Correspondence: komori@ism.ac.jp
[1] Prediction and Knowledge Discovery Research Center, The Institute of Statistical Mathematics, Midori-cho, Tachikawa, Tokyo 190-8562, Japan
[2] The Institute of Statistical Mathematics and Department of Statistical Science, The Graduate University for Advanced Studies Midori-cho, Tachikawa, Tokyo 190-8562, Japan
Full list of author information is available at the end of the article

to investigation of a relationship between a test result and the patient characteristics. Cai and Dodd [9] make some modifications to improve the efficiency of the estimation for parameters, and provide graphical tools for the model checking. In regard to classification problems, Pepe and Thompson [10] propose a method for deriving a linear combination of two markers that optimizes the AUC as well as the pAUC. However, as recognized by Pepe et al. [11], more general approaches are required when the number of markers is large. Moreover, the nonlinear combination of markers is necessary to maximize the AUC as well as the pAUC even in a simple setting such that normality is assumed to the distribution of markers [12]. However, the existing methods [10,13,14] only deal with the *linear* combination of markers.

In this paper, we propose a new statistical method designed to maximize the pAUC, as an extension of AUCBoost [12], using a boosting technique and the approximate pAUC. The approximation-based method makes it possible to *nonlinearly* combine more than two markers, based on basis functions of natural cubic splines as well as decision stumps. The resultant score plots for each marker enable us to observe how the markers are associated with the outcome variable in a visually apparent way. Hence, our boosting method attaches importance not only to the classification performance but also to the interpretation of the results, which is essential in clinical and medical fields.

This paper is organized as follows. In the Methods section, we present a new boosting method for the maximization of the pAUC after giving a brief review of the pAUC and the approximate pAUC. Then, we show a relationship between the pAUC and the approximate pAUC in Theorem 1, which justifies the use of the approximate pAUC in the boosting algorithm. In the Results and Discussion section, we compare the proposed method with other existing ones such as SDF [10], AdaBoost [15], LogitBoost [16] and GAMBoost [17]. In addition, we demonstrate the utility of the proposed method using real data sets; one of them is breast cancer data, in which we use both clinical and genomic data. In the last section, we summarize and make concluding remarks.

Methods

pAUC and approximate pAUC

Partial area under the ROC curve

Let y denote a class label for cases (y = 1) and controls (y = 0), and $\boldsymbol{x}$ be a vector of p markers as $\boldsymbol{x} = (x_1, x_2, ..., x_p)$. Given a score function $F(\boldsymbol{x})$ and a threshold c, we judge the subject as positive if $F(\boldsymbol{x}) \geq c$, and as negative if $F(\boldsymbol{x}) < c$. The corresponding false positive rate (FPR) and true positive rate (TPR) are given as

$$\mathrm{FPR}(c) = \int \mathrm{H}(F(\boldsymbol{x}) - c)\, g_0(\boldsymbol{x}) dx,$$
$$\mathrm{TPR}(c) = \int \mathrm{H}(F(\boldsymbol{x}) - c)\, g_1(\boldsymbol{x}) dx,$$

where H is the Heaviside function: $\mathrm{H}(z) = 1$ if $z \geq 0$ and 0 otherwise, and $g_0(\boldsymbol{x})$ and $g_1(\boldsymbol{x})$ are probability density functions given class 0 and class 1, respectively. Note that FPR and TPR are also dependent on the score function F. However, for the sake of simplicity, we abbreviate it when the abbreviation does not cause ambiguity.

Then, the ROC curve is defined as a plot of TPR against FPR when the threshold c moves on a real number line:

$$\mathrm{ROC}(F) = \{(\mathrm{FPR}(c), \mathrm{TPR}(c)) | c \in \mathrm{R}\},$$

and the area under the ROC curve (AUC) is given as

$$\mathrm{AUC}(F) = \int_{\infty}^{-\infty} \mathrm{TPR}(c)\, d\mathrm{FPR}(c).$$

In this setting, we consider a part of the AUC by limiting the value of FPR between α_1 and α_2, with corresponding thresholds c_1 and c_2, respectively:

$$\begin{aligned} \alpha_1 &= \int \mathrm{H}(F(\boldsymbol{x}) - c_1)\, g_0(\boldsymbol{x}) d\boldsymbol{x}, \\ \alpha_2 &= \int \mathrm{H}(F(\boldsymbol{x}) - c_2)\, g_0(\boldsymbol{x}) d\boldsymbol{x}, \end{aligned} \qquad (1)$$

where $0 \leq \alpha_1 < \alpha_2 \leq 1$ ($c_2 < c_1$). In this paper, we set the values to be 0 and 0.1, respectively. However, it is also worth considering to take $\alpha_1 > 0$ and choose $\alpha_2 - \alpha_1$ to be small enough, so that we essentially maximize TPR for the fixed range of FPR. Then, the pAUC can be divided into a fan-shaped part and a rectangular part:

$$\begin{aligned} & \mathrm{pAUC}(F, \alpha_1, \alpha_2) \\ = & \int_{c_1}^{c_2} \mathrm{TPR}(c)\, d\mathrm{FPR}(c) \\ = & \int_{c_1}^{c_2} \int_{c_2 \leq F(\boldsymbol{x}) \leq c_1} \mathrm{H}(F(\boldsymbol{x}) - c)\, g_1(\boldsymbol{x})\, d\boldsymbol{x} d\mathrm{FPR}(c) \\ & + \mathrm{TPR}(c_1)(\alpha_2 - \alpha_1). \end{aligned}$$

Its probabilistic interpretation is offered by Dodd [18] and Pepe [19] as

$$\begin{aligned} & \mathrm{pAUC}(F, \alpha_1, \alpha_2) \\ = & P(F(\boldsymbol{X}_1) \geq F(\boldsymbol{X}_0), c_2 \leq F(\boldsymbol{X}_0) \leq c_1) \\ = & (\alpha_2 - \alpha_1) P(F(\boldsymbol{X}_1) \geq F(\boldsymbol{X}_0) | c_2 \leq F(\boldsymbol{X}_0) \leq c_1). \end{aligned}$$

Given samples from class 0 $\{x_{0i} : i = 1, 2, \ldots, n_0\}$ and class $1\{x_{1j} : j = 1, 2, \ldots, n_1\}$, the empirical form is expressed as

$$\overline{\text{pAUC}}(F, \bar{\alpha}_1, \bar{\alpha}_2) = \frac{1}{n_0 n_1} \sum_{i \in I} \sum_{j=1}^{n_1} \text{H}\left(F(x_{1j}) - F(x_{0i})\right), \tag{2}$$

where $\bar{\alpha}_1$ and $\bar{\alpha}_2$ are empirical values that are the closest to α_1 and α_1 respectively; $I = \left\{i \middle| \bar{c}_2 \le F(\boldsymbol{x}_{0i}) \le \bar{c}_1\right\}$, where $\bar{c}_1$ and $\bar{c}_2$ are thresholds determined by $\bar{\alpha}_1$ and $\bar{\alpha}_2$.

Approximate pAUC

As seen in Equation (2), the empirical pAUC is non-differentiable. Eguchi and Copas [20] use the standard normal distribution function to approximate the AUC, and applies an algorithms in order to maximize the AUC. Ma and Huang [13] and Wang et al. [14] employ the similar approximation to the AUC by a sigmoid function for multiple marker combination. Since there is no essential difference between the two approximations, we use the standard normal distribution for the approximation of the pAUC:

$$\begin{aligned}&\text{pAUC}_\sigma(F, \alpha_1, \alpha_2)\\ &= \int_{c_1}^{c_2} \int_{c_{2 \le F(\boldsymbol{x}) \le c_1}} \text{H}_\sigma\left(F(\boldsymbol{x}) - c\right) g_1(\boldsymbol{x})\, d\boldsymbol{x} d\text{FPR}(c)\\ &+ \text{TPR}(c_1)(\alpha_2 - \alpha_1),\end{aligned}$$

where α_1 and α_2 are defined in Equation (1), and $\text{H}_\sigma(z)$ is an approximation of H(z) by the standard normal distribution function, that is, $\text{H}_\sigma(z) = \Phi(z/\sigma)$. Similarly, the corresponding empirical pAUC is defined as

$$\begin{aligned}&\overline{\text{pAUC}}_\sigma(F, \bar{\alpha}_1, \bar{\alpha}_2)\\ &= \frac{1}{n_0 n_1} \sum_{i \in I} \Bigg\{ \sum_{j \in J_{\text{fan}}} \text{H}_\sigma\left(F(\boldsymbol{x}_{1j}) - F(\boldsymbol{x}_{0i})\right)\\ &+ \sum_{j \in J_{\text{rec}}} \text{H}\left(F(x_{1j}) - F(x_{0i})\right)\Bigg\},\end{aligned}$$

where $J_{\text{fan}} = \left\{j \middle| \bar{c}_2 \le F(\boldsymbol{x}_{1j}) \le \bar{c}_1\right\}$ and $J_{\text{rec}} = \left\{j \middle| \bar{c}_1 < F(\boldsymbol{x}_{1j})\right\}$. A smaller scale parameter σ implies a better approximation of H(z).

pAUCBoost with natural cubic splines

Boosting

Boosting is one of the most popular method for classification in machine learning community. The main concept is that the score function F is constructed based on various simple functions, termed weak classifiers. There exist many boosting methods according to the objective functions [15-17,21,22]. The seminal and important one is AdaBoost, whose objective function is the exponential loss and its algorithm with the iteration number T is as follows.

1. Start with a score function $F_0(x_i) = 0$, $i = 1, 2, \ldots, n$, where $n = n_0 + n_1$.
2. For $t = 1, \ldots, T$
   (a) Calculate the weights $w_t(i)$

$$w_t(i) = \frac{1}{n} \exp\left\{-F_{t-1}(\boldsymbol{x}_i)(2y_i - 1)\right\}$$

   (b) For $\text{err}_t(f) = \sum_{i=1}^n w_t(i) \text{I}\left(2y_i - 1 \ne f(\boldsymbol{x}_i)\right) \Big/ \sum_{i=1}^n w_t(i)$, find the best weak classifier f_t

$$f_t = \underset{f \in \mathcal{F}_{Ada}}{\arg\min}\, \text{err}_t(f), \tag{3}$$

   where $\mathcal{F}_{Ada}$ is a set of weak classifiers taking values 1 or -1, and I(·) is the indicator function.
   (c) Calculate the coefficient β_t

$$\beta_t = \frac{1}{2} \log \frac{1 - \text{err}_t(f_t)}{\text{err}_t(f_t)}. \tag{4}$$

   (d) Update the score function as

$$F_t(\boldsymbol{x}) = F_{t-1}(\boldsymbol{x}) + \beta_t f_t(\boldsymbol{x})$$

3. Finally, output a final score function as

$$F(\boldsymbol{x}) = \sum_{t=1}^T \beta_t f_t(\boldsymbol{x}).$$

Based on this iterative procedure, we propose the pAUCBoost algorithm after defining the object function.

Objective function

We construct a score function $F(\boldsymbol{x})$ in an additive model for the maximization of the pAUC:

$$F(\boldsymbol{x}) = \sum_{k=1}^p F_k(x_k), \tag{5}$$

where $F_k(x_k)$ is the k-th component of $F(\boldsymbol{x})$, and the plot of $F_k(x_k)$ against x_k is called a score plot that describes the association between x_k and an outcome variable. The subset of weak classifiers for x_k is given as

$$\mathcal{F}_k = \left\{ f_{k,l}(x_k) = N_{k,l}(x_k)/Z_{k,l} \,|\, l = 1, \ldots m_k \right\},$$

where $N_{k,\,l}(x_k)$ is a basis function of x_k for representing a natural cubic spline with m_k knots, and $Z_{k,\,l}$ is a standardization factor that makes the heights of $N_{k,\,l}$'s uniform. Thus, $F_k(x_k)$ in Equation (5) has the following expression.

$$F_k(x_k) = \sum_l \beta_l f_{k,l}(x_k),$$

where β_l's are coefficients that are calculated in the pAUCBoost algorithm. Then, the set of weak classifiers that we use in pAUCBoost is defined as

$$\mathcal{F} = \bigcup_{k=1}^{p} \mathcal{F}_k.$$

In this setting, the objective function we propose is given as

$$\begin{aligned} &\overline{\text{pAUC}}_{\sigma,\lambda}(F, \bar{\alpha}_1, \bar{\alpha}_2) \\ &= \overline{\text{pAUC}}_{\sigma}(F, \bar{\alpha}_1, \bar{\alpha}_2) - \lambda \sum_{k=1}^{p} \int \left\{ F_k''(x_k) \right\}^2 dx_k, \end{aligned} \tag{6}$$

where $F_k''(x_k)$ is the second derivative of $F_k(x_k)$ and λ is a smoothing parameter that controls the smoothness of $F(x)$. It is rewritten as

$$\overline{\text{pAUC}}_{\sigma,\lambda}(F, \bar{\alpha}_1, \bar{\alpha}_2) = \overline{\text{pAUC}}_{1,\lambda\sigma^2}(F/\sigma, \bar{\alpha}_1, \bar{\alpha}_2) \tag{7}$$

Therefore, we have

$$\max_{\sigma,\lambda,F} \overline{\text{pAUC}}_{\sigma,\lambda}(F, \bar{\alpha}_1, \bar{\alpha}_2) = \max_{\lambda,F} \overline{\text{pAUC}}_{1,\lambda}(F, \bar{\alpha}_1, \bar{\alpha}_2). \tag{8}$$

We remark that the scale parameter σ in the definition of $\overline{\text{pAUC}}_{\sigma,\lambda}$ in Equation (6) can be fixed to 1 because of Equation (8). Hence, we redefine the objective function as

$$\overline{\text{pAUC}}_{\lambda}(F, \bar{\alpha}_1, \bar{\alpha}_2) \equiv \overline{\text{pAUC}}_{1,\lambda}(F, \bar{\alpha}_1, \bar{\alpha}_2)$$

without loss of generality.

The maximum value that is attained by a set of (F_1, F_2, ..., F_p) can take the larger value by replacing the functions with p sets of natural cubic splines. This can be proved in the same way as the result of generalized additive models [23], because the penalty term is the same. Hence, we find that the maximizer of the pAUCBoost objective function is the natural cubic spline.

pAUCBoost algorithm

Using weak classifiers f's $\in \mathcal{F}$, we construct a score function F for the maximization of the pAUC. Note that the coefficient β cannot be determined independently of the weak classifier, so we denote it as $\beta(f)$ in the following algorithm.

```
1.Start with a score function $F_0(\boldsymbol{x}) = 0$ and set every
  coefficient $\beta_0(f)$ to be 1 or -1, so that the candidates of
  the initial score function have positive or negative
  derivatives.
2. For $t = 1, ..., T$
   (a) For all $f$'s $\in \mathcal{F}$, calculate the values of thresholds $\bar{c}_1$ and $\bar{c}_2$ of $F_{t-1} + \beta_{t-1}(f)f$.
   (b) Update $\beta_{t-1}(f)$ to $\beta_t(f)$ with a one-step Newton-Raphson iteration.
   (c) Find the best weak classifier $f_t$
       $f_t = \arg\max_{f \in \mathcal{F}} \overline{\text{pAUC}}_{\lambda}(F_{t-1} + \beta_t(f)f, \bar{\alpha}_1, \bar{\alpha}_1)$   (9)
   (d) Update the score function as
       $F_t(\boldsymbol{x}) = F_{t-1}(\boldsymbol{x}) + \beta_t(f_t)f_t(\boldsymbol{x})$   (10)
3. Finally, output a final score function as
       $F(\boldsymbol{x}) = \sum_{t=1}^{T} \beta_t(f_t)f_t(\boldsymbol{x}).$
```

The dependency of the $\overline{\text{pAUC}}_{\lambda}(F_{t-1} + \beta_t(f)f, \bar{\alpha}_1, \bar{\alpha}_2)$ on thresholds $\bar{c}_1$ and $\bar{c}_2$ makes it necessary to pick up the best pair $(\beta_t(f_t), f_t)$ at the same time in step 2.(c). This process is quite different from that of AdaBoost, in which β_t and f_t are determined independently in Equations (3) and (4). Because of the dependency and the difficulty of getting the exact solution of $\beta_t(f_t)$, the one-step Newton-Raphson calculation is conducted in the boosting process. The one-step Newton-Raphson update is also employed in LogitBoost [16] and GAMBoost [17]. The details of the pAUCBoost algorithm are given in additional file 1: Details of the pAUCBoost algorithm.

Tuning procedure

We conduct K-fold cross validation to determine the smoothing parameter λ and the iteration number T. We divide the whole data into K subsets, and calculate the following objective function.

$$\mathrm{pAUC}_{\mathrm{CV}}(\lambda, T) = \frac{1}{K}\sum_{i=1}^{K} \overline{\mathrm{pAUC}}_{\lambda}^{(i)}\left(F^{(-i)}, \bar{\alpha}_1, \bar{\alpha}_2\right),$$

where $F^{(-i)}$ denotes a score function that is generated by the data without i-th subset, and $\overline{\mathrm{pAUC}}_{\lambda}^{(i)}$ is $\overline{\mathrm{pAUC}}_{\lambda}$ calculated by the i-th subset only. The optimal parameters are obtained at the maximum value of $\mathrm{pAUC}_{\mathrm{cv}}(\lambda, T)$ in a set of grid points (λ, T). In the case where the values of the $\mathrm{pAUC}_{\mathrm{cv}}(\lambda, T)$ are unstable, we calculate the $\mathrm{pAUC}_{\mathrm{cv}}(\lambda, T)$ 10 times and take the average to determine the optimal parameters. In our subsequent discussion, we set $K = 10$ and explicitly demonstrate the procedure in the section regarding real data analysis.

Relationship between pAUC and approximate pAUC

We investigate the relationship between the pAUC and the approximate pAUC, which gives a theoretical justification of the use of the approximate pAUC in the pAUCBoost algorithm.

Theorem 1. *For a pair of fixed* α_1 *and* α_2, *let*

$$\Psi(\gamma) = \mathrm{pAUC}_{\sigma}\left(F + \gamma m(\Lambda), \alpha_1, \alpha_2\right),$$

where γ is a scalar, $\Lambda(\boldsymbol{x}) = g_1(\boldsymbol{x})/g_0(\boldsymbol{x})$ and m is a strictly increasing function. Then, $\Psi(\gamma)$ is a strictly increasing function of γ, and we have

$$\begin{aligned}\sup_{F} \mathrm{pAUC}_{\sigma}(F, \alpha_1, \alpha_2) &= \lim_{\gamma \to \infty} \Psi(\gamma) \\ &= \mathrm{pAUC}(\Lambda, \alpha_1, \alpha_2).\end{aligned}$$

See additional file 2: Proof of Theorem 1 and Corollary 1 for the details. Note that Theorem 1 holds for the approximate pAUC by a sigmoid function, so it also gives the justification of the AUC-based methods of Ma and Huang [13] and Wang et al. [14], as a special case where $\alpha_1 = 0$ and $\alpha_2 = 1$. As proved in Eguchi and Copas [20] and Mcintosh and Pepe [24], the likelihood ratio $\Lambda(\boldsymbol{x})$ is the optimal score function that maximizes the AUC as well as the pAUC. In general, the Bayes risk consistency has been well discussed under an assumption of convexity for a variety of loss functions [25]. Theorem 1 suggests a weak version of the Bayes risk consistency for the nonconvex function in the limiting sense.

We also have a following corollary from Theorem 1.

Corollary 1. *For any score function F, let*

$$F_{\gamma\eta}(\boldsymbol{x}) = F(\boldsymbol{x}) + \gamma\,\eta(\boldsymbol{x}),$$

where η *is a score function, and* γ *is a scalar. For a fixed FPR of* $F_{\gamma\eta}$, *the TPR of* $F_{\gamma\eta}$*becomes a increasing function of* γ *if and only if* $\eta = m(\Lambda)$, *where m is a strictly increasing function.*

See additional file 2: Proof of Theorem 1 and Corollary 1 for the details. Note that the corollary holds for any FPR in the range of (0,1). Hence, we find that the score function that moves every and all TPR's upward from the original positions, is nothing but the optimal score function derived from likelihood. This fact is not derived from the Neyman-Pearson fundamental lemma [26], from which $m(\Lambda)$ is proved to maximize the AUC as well as pAUC. This corollary characterizes another property of the optimal score function $m(\Lambda)$.

Results and Discussion

Simulation studies

We compare the performance of pAUCBoost with that of the smooth distribution-free (SDF) method proposed by [10] in a two-dimensional setting, and with those of other existing boosting methods: AdaBoost, LogitBoost and GAMBoost in a higher-dimensional setting. The simulation setting is similar to that of [27]. Suppose that there are four types of sample distributions for each class, $y = 0$ or $y = 1$, as shown in Figure 1. The first panel shows an ideal situation, where we see very little overlap between the two class-conditional distributions. The second situation is of practical interest for disease screening, where FPR must be restricted to be as small as possible, in a case where invasive or costly diagnostic treatments will follow. A small portion of samples from class 1 (cases) is clearly distinguishable from the bulk of samples from class 0 (controls). On the other hand, in the third situation, cases are completely within the range of controls, and therefore not useful for disease screening. The fourth situation is similar to the second one, but some of the samples from cases deviate from controls clearly on both side of the distribution, rather than only on one side. This situation could be worth consideration in a case where high TPR is required with very low FPR in the same way as in the second situation.

In the simulation study, we apply pAUCBoost with $\bar{\alpha}_1 = 0$ and $\bar{\alpha}_2 = 0.1$. The training data set contains 50 controls and 50 cases, and the accuracy of the performance is evaluated based on 100 repetitions using test data sets of size 1000 (500 for each class).

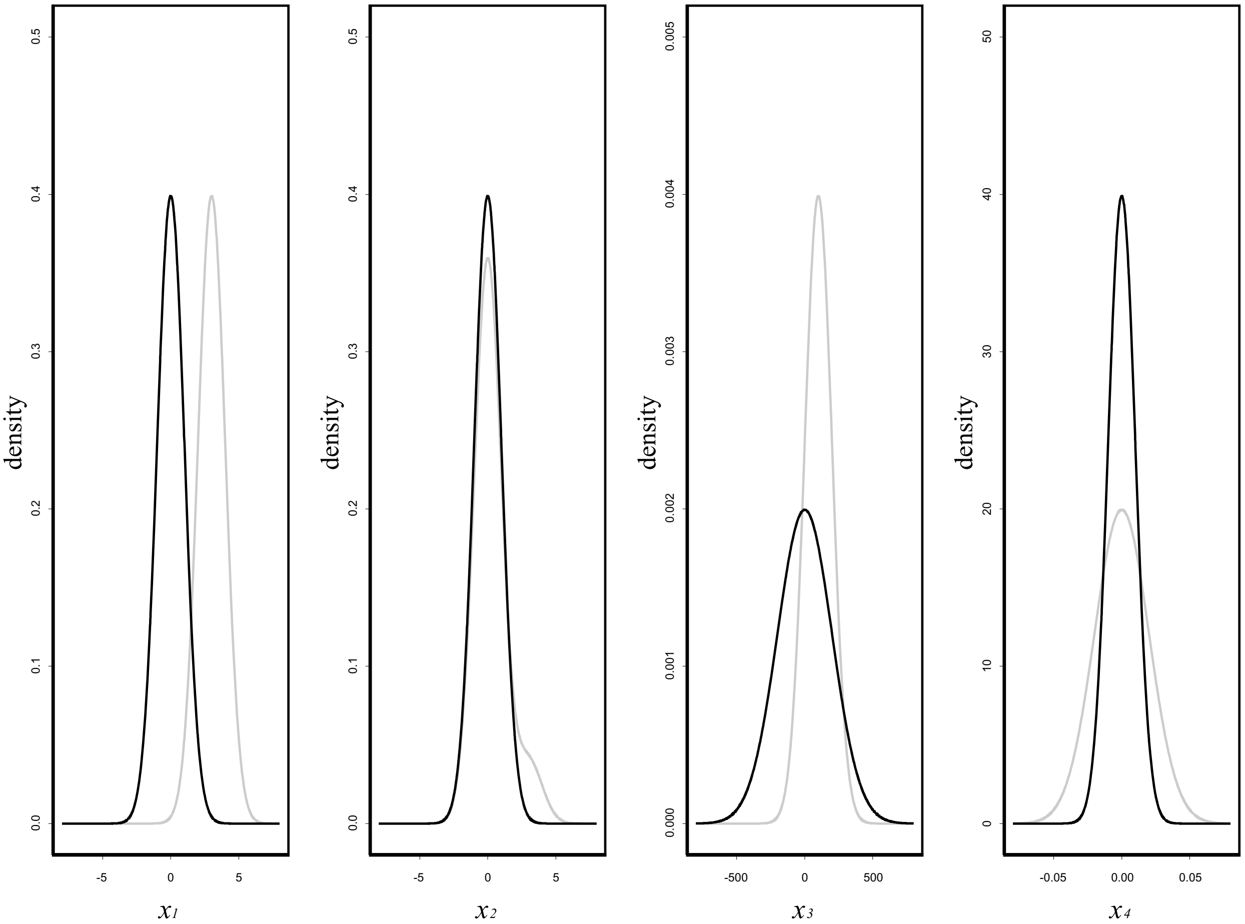


Figure 1 Illustration of simulation setting. Illustration of four different types of sample distributions for class 0 (black) and class 1 (gray).

Comparison with SDF

We consider the second situation, where we assume normality distributions such as $X_{20} \sim \mathcal{N}(0,1)$ and $X_{21} \sim \pi\mathcal{N}(0,1) + (1-\pi)\mathcal{N}(3,1)$ with mixing proportion $\pi = 0.9$, and the last situation: $X_{40} \sim \mathcal{N}(0, 1/100), X_{41} \sim \mathcal{N}(0, 4/100)$. That is, the conditional probability density function of a class label y given $\boldsymbol{x}$ is given by

$$p(y|\boldsymbol{x}) = \frac{y(\Lambda_1(\boldsymbol{x})-1)+1}{\Lambda_1(\boldsymbol{x})+1}, \tag{11}$$

where $\Lambda_1(\boldsymbol{x})$ is the likelihood ratio:

$$\begin{aligned} \Lambda_1(\boldsymbol{x}) &= \frac{\{\pi\phi(x_2)+(1-\pi)\phi(x_2-3)\}\phi(5x_4)}{2\phi(x_2)\phi(10x_4)} \\ &= \frac{1}{20}\left\{9+\exp\left(3x_2-\frac{9}{2}\right)\right\}\exp\left(\frac{75}{2}x_4^2\right) \end{aligned} \tag{12}$$

and $\varphi(z)$ is the standard normal density function. The resultant mean value (and the 95 percent confidence interval) of the pAUC based on pAUCBoost turns out to be 0.017 (0.012, 0.020), and the value of SDF to be 0.011 (0.005, 0.017). This large difference is because SDF assumes linearity of the score function of $F(\boldsymbol{x})$ as

$$F(\boldsymbol{x}) = \gamma_2 x_2 + \gamma_4 x_4,$$

and the coefficient of x_4 is estimated by SDF to be around 0 as shown in Figure 2 (a), under the condition that λ_2 is fixed to 1. On the other hand, pAUCBoost considers the nonlinearity of $F(\boldsymbol{x})$ as

$$F(\boldsymbol{x}) = F_2(x_2) + F_4(x_4),$$

as shown in Figure 2(b). The left panel shows the score plot of x_2, and the right one shows that of x_4. The pAUCBoost clearly captures the nonlinearity of F_4 (x_4), where one of the optimal score function in this setting is derived from Equation (12) as

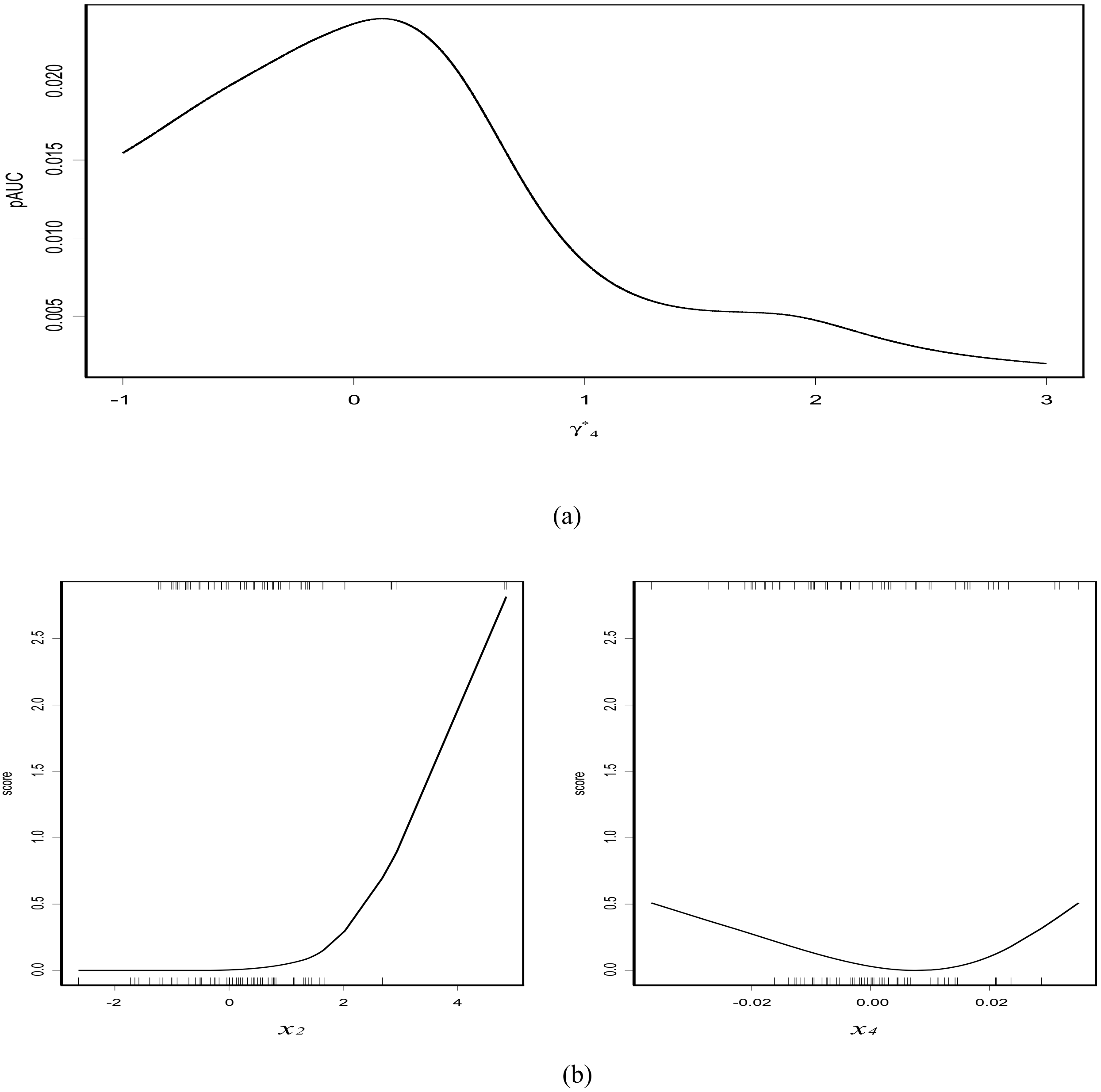


Figure 2 Comparison of SDF method and pAUCBoost method. (a) Illustration of the estimated value of pAUC by SDF method, where $\gamma_4^* = \gamma_4$ if $-1 \leq \gamma_4 \leq 1$ and 2-1/γ_4 otherwise; (b) the resultant score plots by pAUCBoost. The rug plot along the bottom of each graph describes the observations from class 0; the rug plot along the top of each graph describes those from class 0.

$$m_1(x) = \log\left\{9 + \exp\left(3x_2 - \frac{9}{2}\right)\right\} + \frac{75}{2}x_4^2.$$

Note that the ROC curve is invariant to a monotone transformation of the score function.

Although a nonlinear transformation could be applied to the data in advance, it is not practical to examine all marginal distributions and decide the appropriate transformations in general situations. Hence, it is better to take the nonlinearity into consideration in the method itself in this way.

We have also confirmed that the performance of pAUCBoost is compatible with that of SDF, in a setting when linearity of the score function is reasonable. We have averages of 0.013 (0.007, 0.017) and 0.013 (0.011, 0.015 pAUCBoost and the SDF method, respectively, under the situation that x_4 is also distributed as $X_{40} \sim \mathcal{N}(0,1)$ and $X_{41} \sim \pi\mathcal{N}(0,1) + (1-\pi)\mathcal{N}(3,1)$.

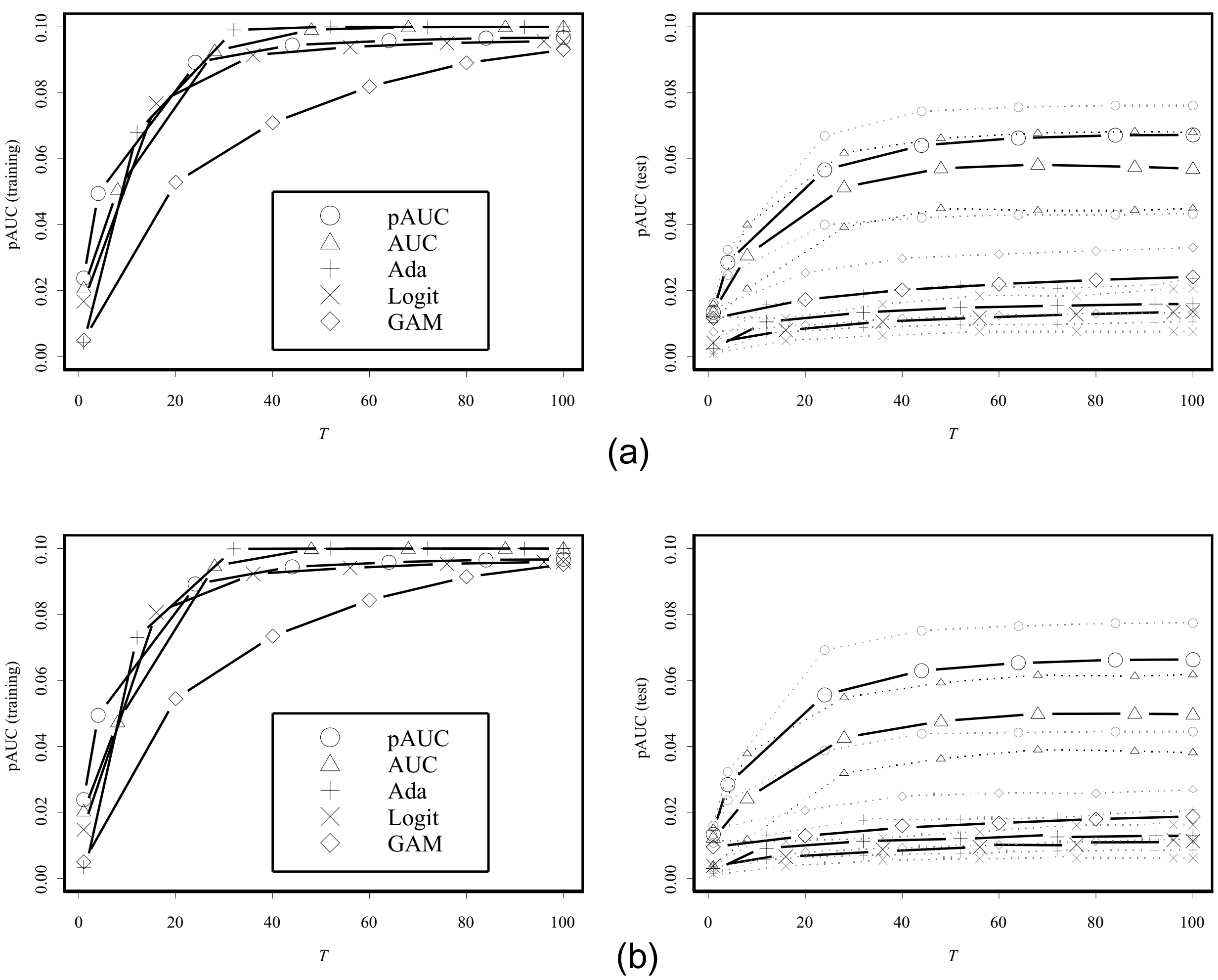


Figure 3 Results of simulation study based on the values of the pAUC. (a) The results of the pAUC with FPR between 0 and 0.1 for training data (left panel) and test data (right panel) with only informative genes. The gray dashed lines indicate the 95% confidence bands. (b) the results of the pAUC with noninformative genes added.

That is, the conditional probability density function of y given x is given same way as Equation (11):

$$p\left(y|\boldsymbol{x}\right)=\frac{y\left(\Lambda_2\left(\boldsymbol{x}\right)-1\right)+1}{\Lambda_2\left(\boldsymbol{x}\right)+1},$$

where

$$\begin{aligned}&\Lambda_2\left(\boldsymbol{x}\right)\\&=\frac{\left\{\pi\phi\left(x_2\right)+\left(1-\pi\right)\phi\left(x_2-3\right)\right\}}{\phi\left(x_2\right)}\\&\times\frac{\left\{\pi\phi\left(x_4\right)+\left(1-\pi\right)\phi\left(x_4-3\right)\right\}}{\phi\left(x_4\right)}\\&=\frac{1}{10}\left\{9+\exp\left(3x_2\frac{9}{2}\right)\right\}\frac{1}{10}\left\{9+\exp\left(3x_4-\frac{9}{2}\right)\right\}.\end{aligned}$$

The one of the optimal score function is given by

$$\begin{aligned}m_2\left(\boldsymbol{x}\right)&=\log\left\{9+\exp\left(3\boldsymbol{x}_2-\frac{9}{2}\right)\right\}\\&+\log\left\{9+\exp\left(3x_4-\frac{9}{2}\right)\right\}.\end{aligned}$$

It is interesting to note that almost the same results are obtained by these quite different statistical methods. SDF uses the estimated values of pAUC to derive a score function; on the other hand, pAUCBoost directly uses the empirical value of the approximate pAUC in the algorithm.

Comparison with other boosting methods

We focus on only the most practical situation in disease screening such as the second situation in Figure 1. Pepe et al. [27] show the utility of the use of the pAUC, in selec-

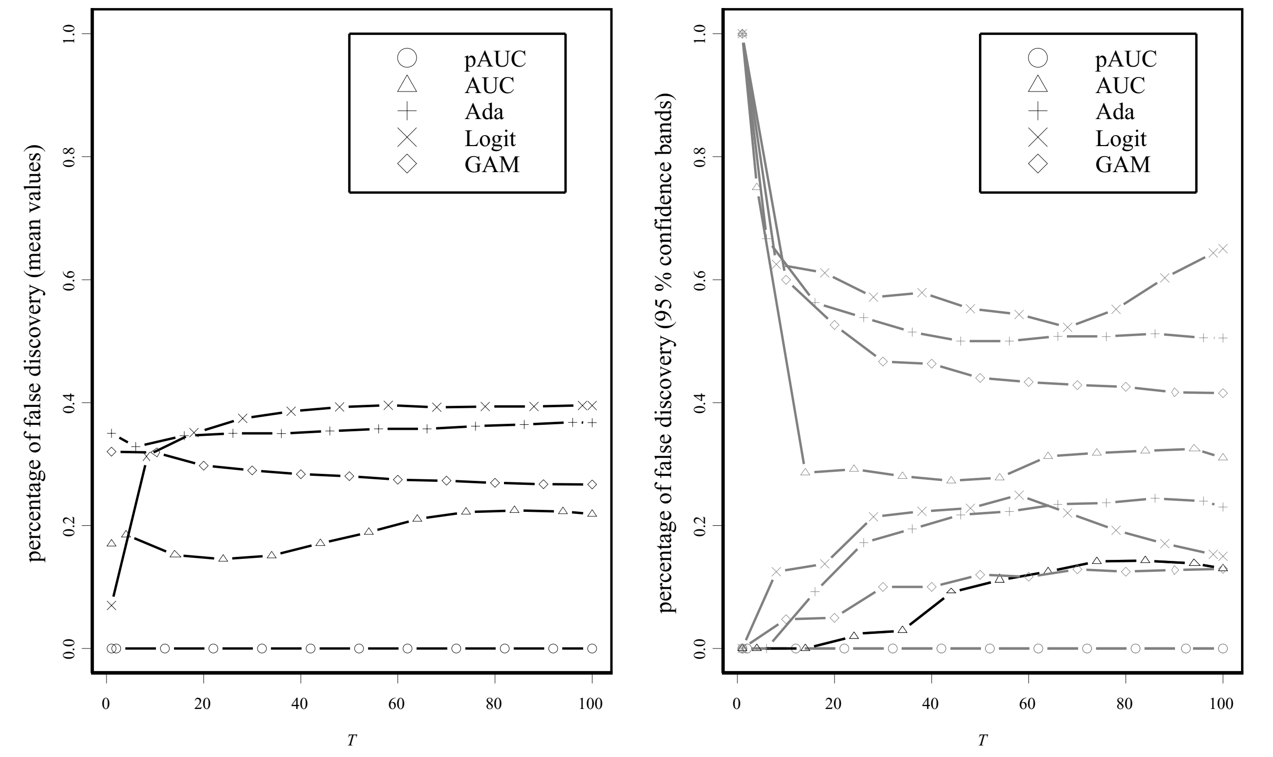


Figure 4 Results of simulation study based on the marker selection. The mean values of percentage of false discovery (left panel) and 95% confidence bands (right panel) for each boosting method. The horizontal axis denotes the iteration number of *T*. The lower sides of the 95% confidence bands of AUCBoost are shown by the heavy black line to emphasize the difference from those of pAUCBoost.

tion of potential genes that are useful for discrimination between normal and cancer tissues. The point is that the value of pAUC reflects the overlap of two distributions of controls and cases, so that we can select genes that are suitable for the purpose of further investigation. For example, some overexpressed genes encourage us to investigate the corresponding protein products. However, the task of how to combine the selected genes for better discrimination is still pending.

Suppose we select 50 genes by a filtering procedure, which are closely correlated each other, such that 50-dimensional gene vectors for class 0 and class 1 are distributed as $\boldsymbol{X}_0 \sim N(\mathbf{0}, \Sigma_0)$ and $\boldsymbol{X}_1 \sim N(\mathbf{0}, \Sigma_1)$, respectively. The covariance matrices are designed as $\boldsymbol{\Sigma}_0 = 0.95 \times \boldsymbol{W}_0 + 0.05 \times \boldsymbol{I}$ and $\boldsymbol{\Sigma}_1 = 0.95 \times \boldsymbol{W}_1 + 0.05 \times \boldsymbol{I}$, where $\boldsymbol{W}_0$ and $\boldsymbol{W}_1$ are 50 × 50 matrices that are sampled from Wishart distribution with the identity matrix and 10 degrees of freedom at every repetition of the simulation. The identity matrix $\boldsymbol{I}$ is added for avoiding the singularity of the covariance matrices. These matrices are normalized to have 1's on the diagonal part in the similar way to the simulation setting of Zhao and Yu [28], and the range of the correlations turns out to be about between *0.8* and *-0.8*. Then, we randomly replace 10 percent of samples from class 1 with those that are distributed as $\mathcal{N}(3,1)$ for each gene, so that each gene is informative in the sense of the pAUC as shown in the second situation of Figure 1.

Figure 3(a) shows plots of the average of the pAUC against iteration number *T* for five boosting methods. For all the boosting methods, the values of the pAUC based on the training data almost reach the upper bound values 0.1 after a number of iterations. However, the values based on the test data show clear differences. The pAUC-Boost properly detects the small difference of the two distributions illustrated in the second panel in Figure 1, and shows the best performance. On the other hand, AdaBoost, LogitBoost and GAMBoost cannot distinguish the two groups at all.

Next for illustration of the gene selection of pAUC-Boost, we added some noninformative genes to the 50 genes above, i.e., genes that are assumed to be normally distributed with the same mean and the same covariance matrix: $\boldsymbol{X}_0^{\text{noise}}, \boldsymbol{X}_1^{\text{noise}} \sim \mathcal{N}(\mathbf{0}, \Sigma)$, where is generated in the same way as above. The results in the left panel of Figure 3(b) are the almost the same as those in Figure

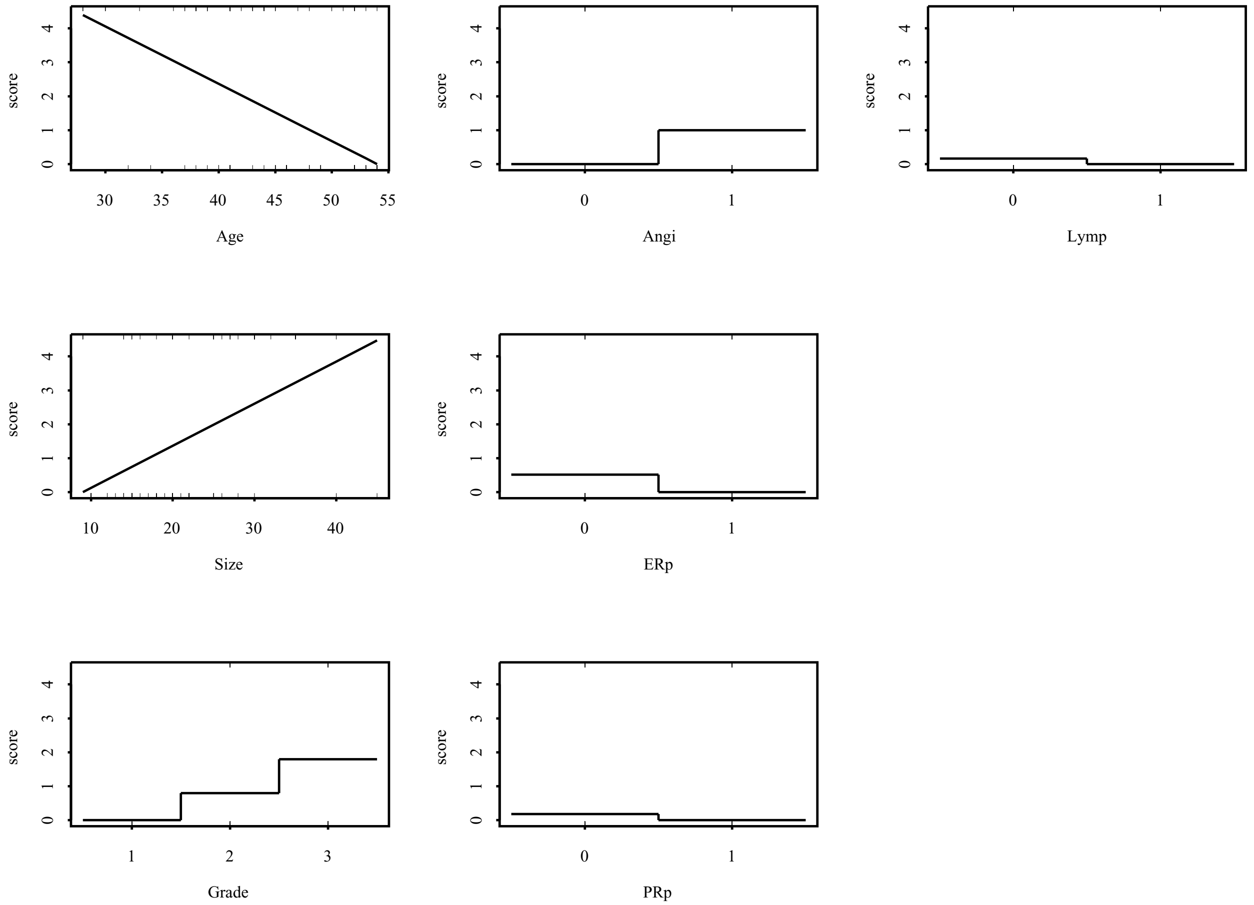


Figure 5 Score plots of clinical markers in breast cancer data. Score plots of clinical markers that describe the association between the markers and the outcome variable. The rug plot at the bottoms of each score plot shows the observations from patients with good prognosis; the rug plot for patients with distant metastases is described at the top of each score plot.

3(a). However, we can find a clear difference between the right panels. The performances of all methods except for pAUCBoost go down on a relatively large scale. We can observe that the mean values of the pAUC by pAUCBoost are above the upper sides of the 95% confidence bands of those by AUCBoost after around $T = 20$. This is mainly because of "false discovery", or selection of noninformative genes by chance. Figure 4 shows the resistance of pAUCBoost to false discovery. The mean values of percentage of false discovery (the number of selected noninformative genes over the number of selected genes) are plotted in the left panel; the 95 percent confidence bands (gray lines) are plotted in the right panel, against the iteration number T, respectively. We see that the boosting methods other than pAUCBoost select noninformative genes from the early stage of boosting procedure. The difference of performance of pAUCBoost from the others is 95% significant after around $T = 15$ as shown in the right panel. The upper side bands of the 95% confidence bands reached 1 at the very beginning of the iteration for AUCBoost, AdaBoost, LogitBoost and GAMBoost. The boosting methods other than pAUCBoost clearly suffer from false discovery. pAUCBoost seems to have an advantage because it focuses on the essential part of the sample distribution in the sense of the pAUC.

Mainly, there are two types of weak classifiers: smoothing splines and decision stumps. Buhlmann and Yu [21] proposed to use smoothing splines in the L_2 Boost algorithm, and Tutz and Binder [17] used B-splines in GAMBoost. However, the way of fitting the weak classifiers in pAUCBoost is different from those methods. Our algorithm updates a score function with a basis function of a natural cubic spline for one marker in Equations (9) and (10). On the other hand, their algorithms update a score function with *a set* of basis functions for one marker. Hence, our resultant score functions have tendency to have simpler forms (See the illustrations of score plots in the next section), which also leads to simple interpretation of the association between the markers and the outcome variable. Note that there exists a trade-off between

Table 1: The top 30 genes ranked by the probability of gene selection, and the values of the pAUC and AUC.

No	gene name	*Pg*(100)	pAUC	AUC
1	Contig41613_RC	0.728	0.036	0.666
2	NM_006931	0.728	0.035	0.678
3	Contig40831_RC	0.706	0.037	0.672
4	Contig55574_RC	0.639	0.035	0.654
5	AB023173	0.636	0.034	0.684
6	Contig63649_RC	0.626	0.034	0.749
7	NM_018964	0.586	0.034	0.660
8	AL137615	0.571	0.033	0.655
9	NM_006201	0.541	0.032	0.664
10	NM_001710	0.520	0.032	0.638
11	AA555029_RC	0.519	0.032	0.708
12	NM_020386	0.490	0.030	0.699
13	Contig7558_RC	0.488	0.032	0.659
14	Contig51464_RC	0.482	0.030	0.668
15	NM_014246	0.474	0.032	0.613
16	NM_007359	0.463	0.032	0.696
17	NM_006148	0.450	0.029	0.661
18	NM_004163	0.442	0.029	0.729
19	Contig37562_RC	0.423	0.031	0.630
20	Contig55377_RC	0.416	0.029	0.726
21	Contig47405_RC	0.404	0.029	0.718
22	NM_012261	0.393	0.029	0.721
23	NM_014400	0.379	0.028	0.681
24	Contig44409	0.368	0.029	0.692
25	AL080059	0.364	0.027	0.801
26	Contig60864 RC	0.358	0.029	0.637
27	NM_003748	0.353	0.025	0.793
28	AL080110	0.349	0.026	0.652
29	AL122101	0.343	0.028	0.708
30	NM_018120	0.336	0.026	0.671

the simplicity and the number of markers necessary for good performance of discrimination. However, the simplicity depends on the number of basis functions used for the selected markers, so the more complicated association can be expressed by increasing the number of the basis functions.

In AdaBoost and LogitBoost, decision stumps are used as weak classifiers [29,30]. The advantage of using decision stumps is that we can apply the boosting methods independently of the scale of the marker values. Hence, the decision stump-based method is resistant to outliers, which often occur in real data. However, it easily suffers from false discovery, as clearly shown in simulation studies. This causes poor performance in a setting where non-informative genes are mixed with informative ones. We have also confirmed that pAUCBoost with decision stumps for weak classifiers shows worse performance than that of pAUCBoost with natural cubic splines. Hence, we have to be much careful about which weak classifiers to be employed. It depends on the types of markers or the purpose of the analysis we are engaged in.

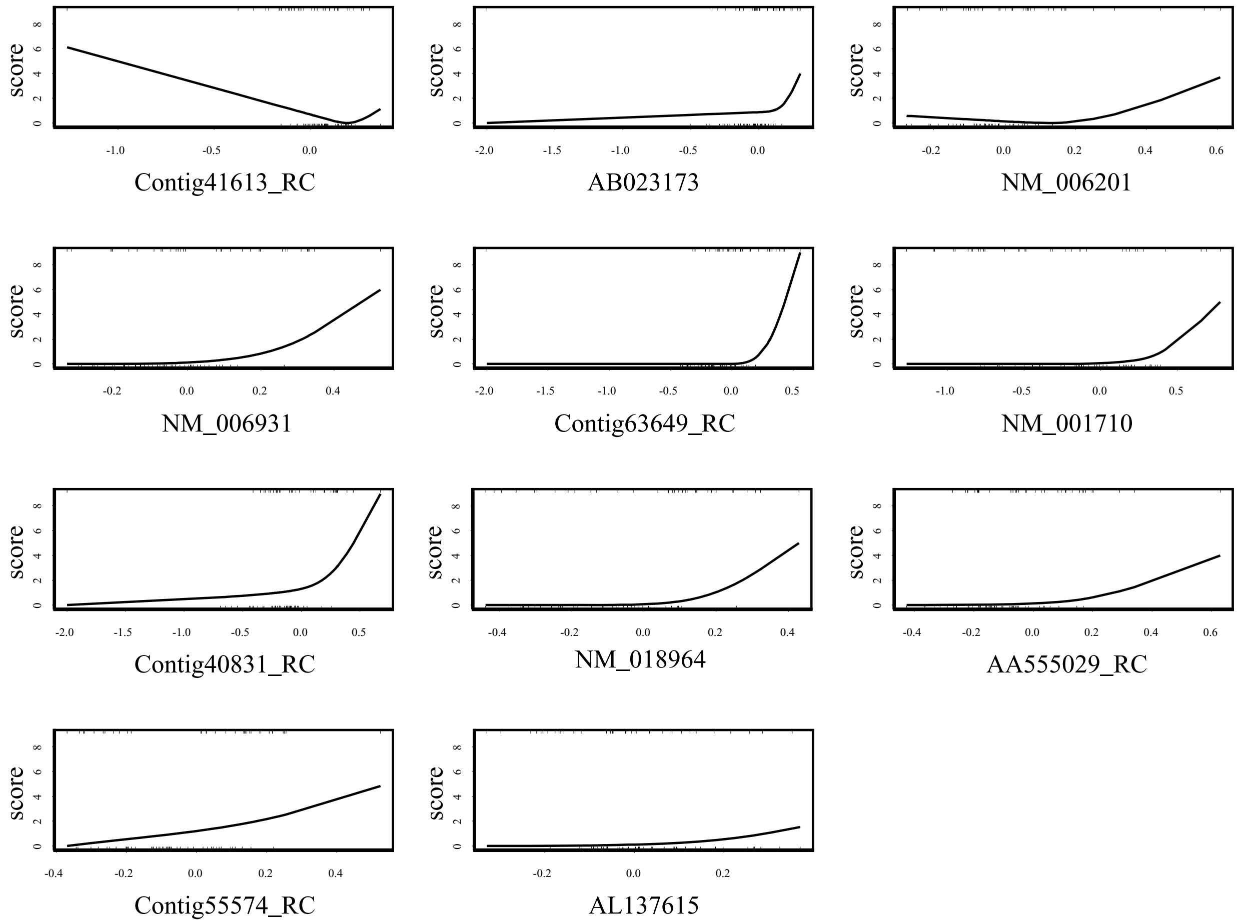
Application of pAUCBoost to real data

Breast cancer data

The breast cancer data of van't Veer et al. [31] contains not only gene expression profiles but also clinical markers such as Age, age of patients; Size, diameter of breast cancer; Grade, tumour grade; Angi, existence or nonexistence of angioinvasion; ERp, ER expression; $\mathrm{PR_p}$, PR expression; and Lymp, existence or nonexistence of lymphocytic infiltrate. First, we apply AUCBoost to these clinical markers and investigate their utility. The weak classifiers we use are natural cubic splines for continuous markers (Age and Size), and decision stumps to discrete or categorical markers. Second, we apply pAUCBoost with $\bar{\alpha}_1 = 0$ and $\bar{\alpha}_2 = 0.1$ to the gene expression data after a pAUC-based filtering procedure proposed by Pepe et al. [27]. The training data set and the test data set are the same as those in [31], that is, 44 patients with good prognosis and 34 patients with distant metastases for training data, and 7 and 12 patients for test data, respectively.

Figure 5 shows the results of the score plot generated by AUCBoost with $\lambda = 0.01$ and $T = 20$, which were determined by a 10-fold cross validation. The Age and Size showed almost linear association with the prognosis, and a tendency to develop metastases increased as the value of Grade. The patients with negative ER and negative PR were estimated to have high risk of metastases, which are consistent with the result of van't Veer et al. [31]. The order of description of the score plots is in accordance with that of markers selected in the AUCBoost algorithm. Hence, Age has the largest contribution to the value of the AUC. The order is from the upper left panel to the lower right panel, so the second important marker is Size and the last one is Lymp. We have found that the values of the AUC for training and test data are 0.846 and 0.964, respectively. These results are comparable to those of van't Veer et al. [31] that were derived from the gene expression data: 0.882 and 0.869, respectively. This means that clinical markers themselves also have the ability to discriminate to some extent the patients with good prognosis from those with metastases.

Next, we analyze the gene expression data as follows. The informative genes were selected, in the same way as [31], from the total of 25000 genes according to the crite-



Contig41613_RC

AB023173

NM_006201

NM_006931

Contig63649_RC

NM_001710

Contig40831_RC

NM_018964

AA555029_RC

Contig55574_RC

AL137615

Figure 6 Score plots of gene expressions in breast cancer data. Score plots of the selected 11 genes that describe the association between the genes and the outcome variable. The rug plot at the bottoms of each score plot shows the observations from patients with good prognosis; the rug plot for patients with distant metastases is described at the top of each score plot.

ria that the genes are two-fold regulated and that the p-values are less than 0.01 in more than 3 patients. Then, the approximately 5000 filtered genes were ordered based on their values of the pAUC with $\bar{\alpha}_1 = 0$ and $\bar{\alpha}_2 = 0.1$. In order to assess the variability of the top genes, we used the probability of gene selection proposed by Pepe et al. [27], that is

$$P_g(k) = P(\text{gene } g \text{ ranked in the top } k), \qquad (13)$$

where k was set to 100 in this analysis and this probability was calculated by 1000 bootstrap resampling replications. Table 1 shows the results of the top 30 genes ranked by P_g (100), along with the values of pAUC and AUC calculated from the original data. We picked up significant genes with P_g (100) > 0.5, and applied pAUCBoost to the 11 genes. The score plots in Figure 6 describe the nonlinear association between gene expressions and the prognosis. Among the 11 genes, Contig41613_RC showed a nonlinear and nonmonotonic association. That is, the gene expression of the patients with metastases had large variance as shown by the rug plot, compared with that of patients with good prognosis, which had a tendency to take small absolute values and concentrate around the origin. The nonlinearity of the associations can be captured by pAUCBoost in this way. The values of tuning parameter λ and T were determined to be 10^{-6} and 65 by 10-cross validation, as described in the left panel in Figure 7. The value of A is very small, and it seems to be ignorable. However, since the value of A has an implicit role to control the accuracy of approximation of the pAUC as seen Equations (7) and (8), it should not be set to 0. The right panel in Figure 7 shows the pAUC for training (solid) and test (dashed) data, as a function of T with $\lambda = 10^{-6}$. We saw that both of the values for training and test data are more than 3 times larger than those of van't Veer et al. [31]: 0.025 and 0.0084, respectively. Finally, we confirmed that the nonlinearity of score func-

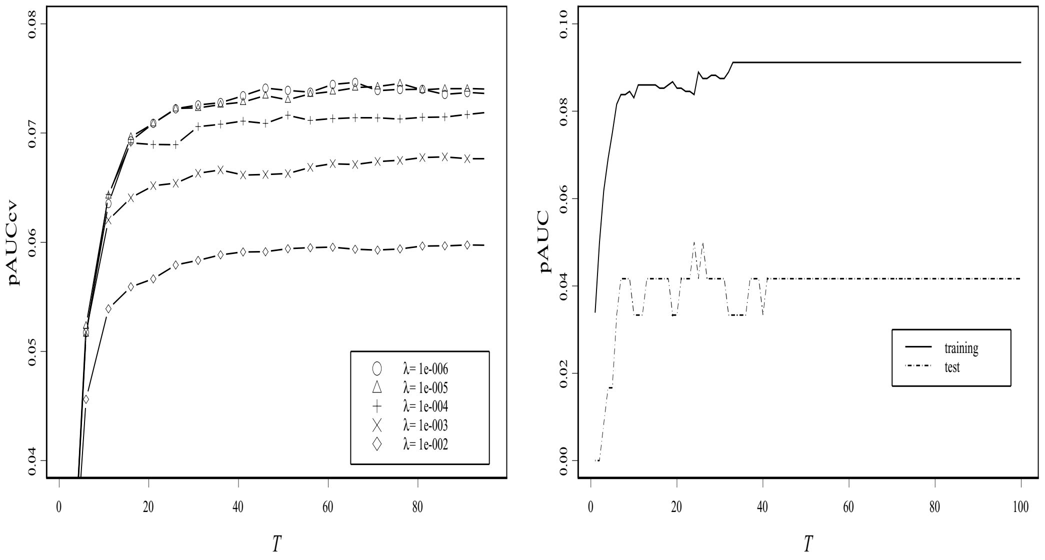


Figure 7 Results of the pAUC. The results of 10-fold cross validation with different values of smoothing parameter λ and iteration number T (left panel); the results of the values of pAUC for training data (solid) and test data (dashed) by pAUCBoost, as a function of T with $\lambda = 10^{-6}$ (right panel).

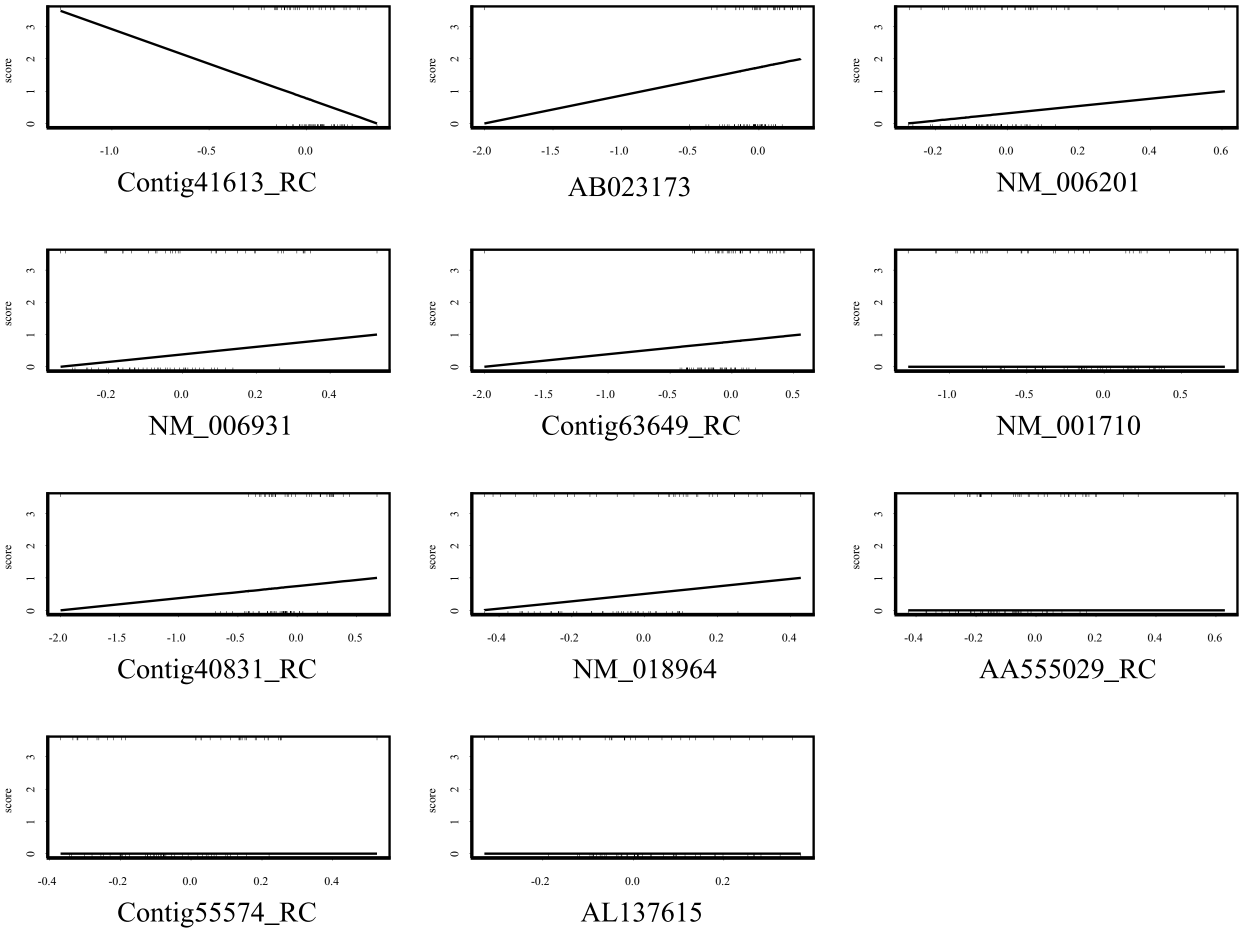


Figure 8 Linear score plots of gene expressions in breast cancer data. Score plots of the selected 11 genes generated by pAUCBoost using only linear weak classifiers.

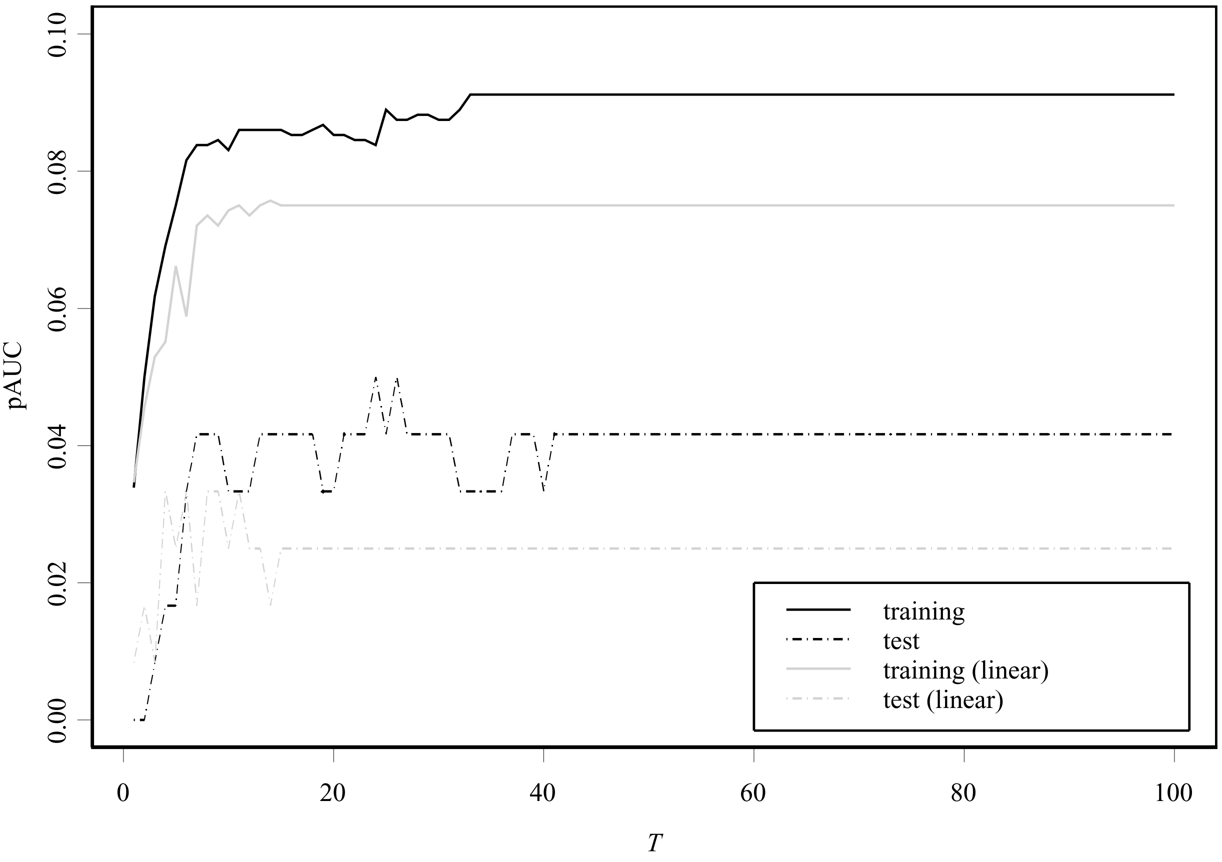


Figure 9 Comparison between linear and nonlinear score functions. Comparison based on the values of the pAUC between linear and nolinear score functions generated by pAUCBoost.

tion F as shown in Figure 6 played an important role for the classification performance. See the score plots that are generated by only linear basis functions of natural cubic splines in Figure 8, and resultant values of the pAUC in Figure 9. Both the values of the pAUC based on training data and those on test data changed for the worse. The results of other boosting methods, and the results for less stringent bounds on the values of $\bar{\alpha}_2$ are presented in additional file 3: Supplementary results of breast cancer data analysis.

Ovarian cancer data

This dataset was analyzed by Pepe et al. [27] for illustration of their pAUC-based filtering procedure in Equation (13). It consists of 1536 genes spotted on the glass arrays, and is available from the website of a textbook by Pepe [19]. It includes 23 controls with normal ovarian tissues and 30 cases with ovarian cancers. We divided the whole data into training data and test data in the ratio of 2 to 1. That is, the first 15 controls and 20 cases in the original data are used for training data; the others are for test data. Using the training data only, we ranked the genes according to the value of the pAUC with $\bar{\alpha}_1 = 0$ and $\bar{\alpha}_2 = 0.1$, and assessed the variability as in the breast cancer analysis above. Then, we picked up 20 genes that satisfy $P_g(100) > 0.9$ in the same way as Pepe et al. [27]. We found that there are 12 common genes to theirs, including g1483 that ranked best in their analysis. For these 20 filtered genes, we applied pAUCBoost and had the resultant score plots in Figure 10. As seen is Figure 10, the pAUCBoost selected 11 genes and attained the maximum value of the pAUC (0.1). Finally, we assessed the classification performance based on the independent test data, and had a high value of the pAUC (0.08). The classification is relatively easy, so the results of other boosting methods also reached the same values of the pAUC based on the independent test data.

Leukemia data

The third data we analyzed is leukemia data [32]. It contains 38 training samples and 34 test samples with 7129 genes for acute myeloid leukemia (AML) and acute lym-

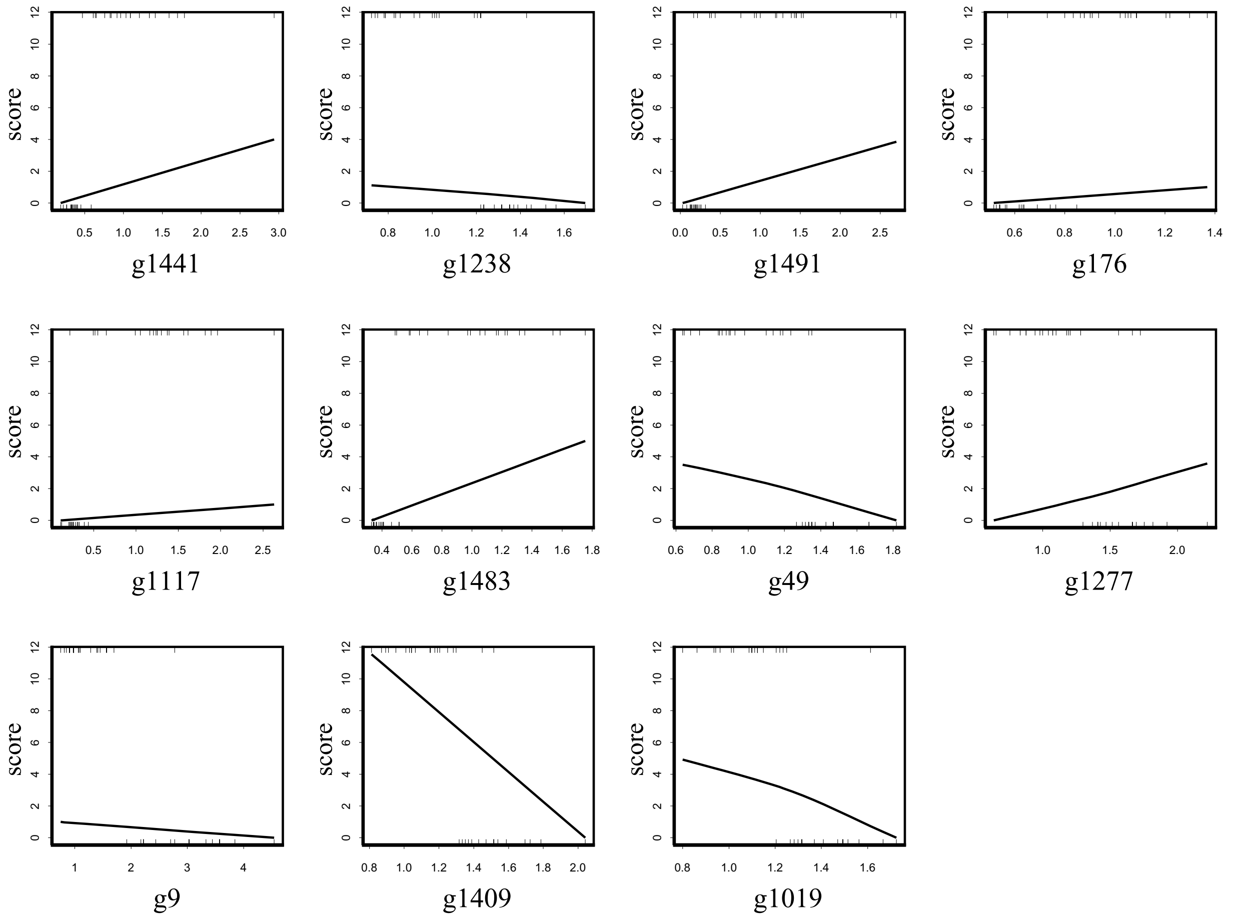


Figure 10 Score plots of gene expressions in ovarian cancer data. Score plots of the selected 11 genes by pAUCBoost based on ovarian cancer data. The rug plot at the bottoms of each score plot shows the observations from normal controls; the rug plot for ovarian cancer cases is described at the top of each score plot.

phoblastic leukemia (ALL). We repeated the same procedure as the previous two analyses above using the 8 filtered genes that satisfy $P_g(100) > 0.9$. We achieved the perfect classification performance regarding both training and test data sets, and had the score plots in Figure 11. The results of other boosting methods produced similar but a little worse values of the pAUC. That is, the values are more than 0.08 but less than 0.1 on the basis of test data.

Conclusions

We have developed the pAUCBoost algorithm to maximize the pAUC based on the approximate pAUC in the additive model. The use of the approximate pAUC is justified by showing a relationship with the pAUC in Theorem 1.

A resultant score function is decomposed componentwise into functions that are useful for understanding the associations between each marker and the outcome variable, as shown in real data analysis. Natural cubic splines that give the maximum of the pAUCBoost objective function are used for markers taking continuous values. In addition, using decision stumps for markers that take discrete or categorical values the proposed method enables us to treat various kinds of markers together.

We have also provided a consistent way to analyze gene expression data in the sense of the pAUC, as shown in the analysis of the breast cancer data, ovarian cancer data and leukemia data. The pAUC is shown to be useful by Pepe et al. [27] for selection of informative genes, some of which are overexpressed or underexpressed in cancer tissues. However, how to combine the selected genes and how to discriminate the cancer tissues from normal tissues, have not been addressed. We nonlinearly combined the genes ranked by the pAUC in order to produce a score function, by which the classification of controls and cases is done. Interestingly, we have found 4 genes in common with the 70 genes of van't Veer et al. [31]: Contig63649_RC, AA555029_RC, Contig40831_RC,

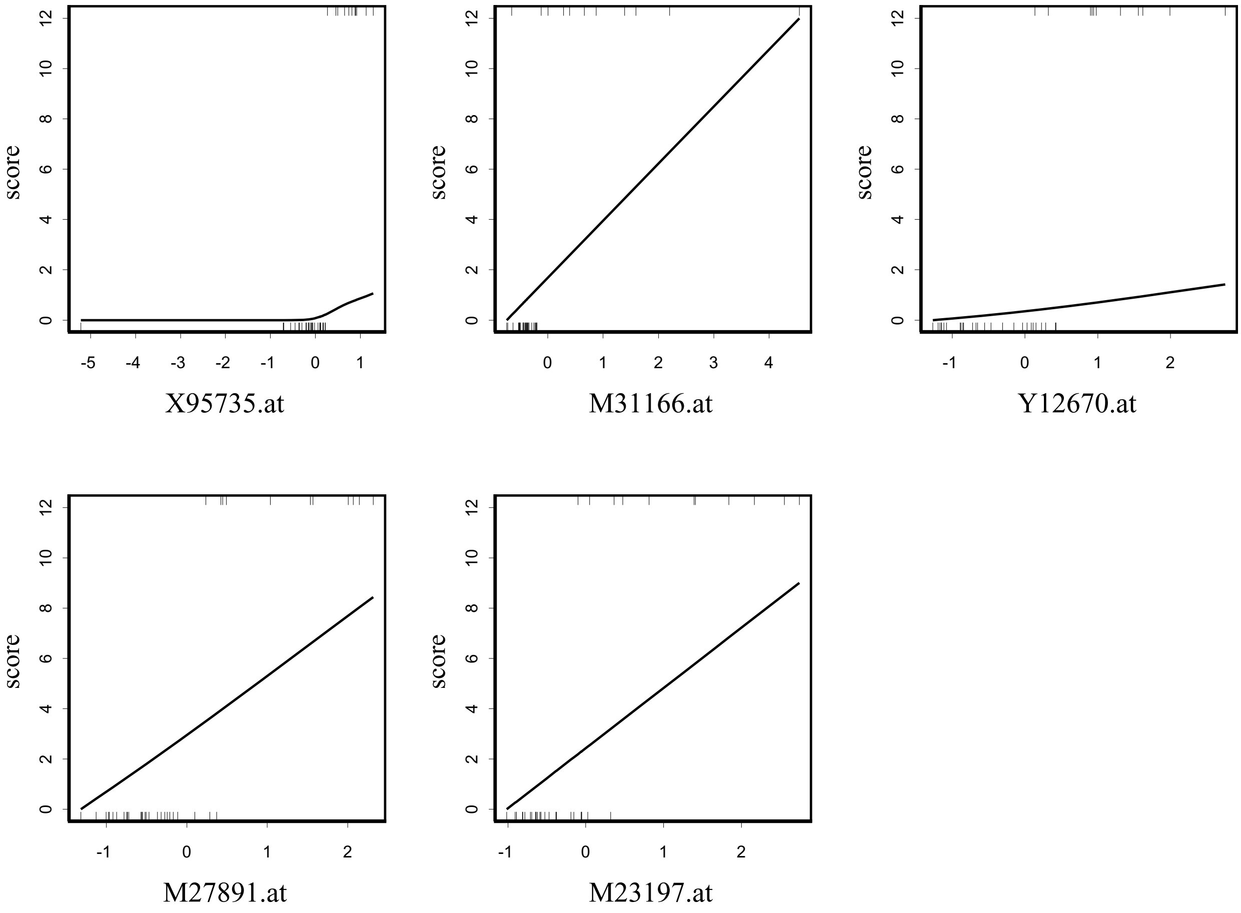


Figure 11 Score plots of gene expressions in leukemia data. Score plots of the selected 5 genes by pAUCBoost based on leukemia data. The rug plot at the bottoms of each score plot shows the observations from ALL; the rug plot for AML is described at the top of each score plot.

NM_L006931. 6 genes among the selected 11 genes are related to protein coding. We also applied pAUCBoost to the 70 genes for comparison with the result from the 11 genes. We found that it yielded a poor result, especially about the value of pAUC for test data. Hence, pAUCBoost with FPR restricted to be small should be applied to the genes or markers that have gone through a pAUC-based filtering procedure beforehand. In the usual analysis setting, in which markers do not have especially high values of the pAUC, AUCBoost is preferable because of the stable performance due to the comprehensive information it can take into the algorithm.

Additional material

Additional file 1 Details of the pAUCBoost algorithm. gives the details of the pAUCBoost algorithm.

Additional file 2 Proof of Theorem 1 and Corollary 1. contains the details of the proof of Theorem 1 and Corollary 1.

Additional file 3 Supplementary results of breast cancer data analysis. describes the supplementary results of breast cancer data, where the range of FPR is more relaxed.

Authors' contributions

OK carried out the simulation study and the real data analysis. OK and SE are responsible for the algorithm of proposed method, the proof of Theorem 1 and Corollary 1 in Additional File 2. They drafted the manuscript and approved the final manuscript.

Acknowledgements

The authors would like to express acknowledgement to Professor John Copas who kindly gave us some useful comments and suggestions to this paper. We also note that this study is supported by the Program for Promotion of Fundamental Studies in Health Sciences of the National Institute of Biomedical Innovation (NIBIO).

Author Details

[1]Prediction and Knowledge Discovery Research Center, The Institute of Statistical Mathematics, Midori-cho, Tachikawa, Tokyo 190-8562, Japan and [2]The Institute of Statistical Mathematics and Department of Statistical Science, The Graduate University for Advanced Studies Midori-cho, Tachikawa, Tokyo 190-8562, Japan

Received: 24 February 2010 Accepted: 10 June 2010
Published: 10 June 2010

References

1. Bamber D: **The area above the ordinal dominance graph and the area below the receiver operating characteristic graph.** *Journal of Mathematical Psychology* 1975, **12**:387-415.
2. Pepe MS, Janes H, Longton G, Leisenring W, Newcomb P: **Limitation of the odds ratio in gauging the performance of a diagnostic, prognostic, or screening marker.** *American Journal of Epidemiology* 2004, **159**:882-890.
3. Cook NR: **Use and misuse of the receiver operating characteristic curve in risk prediction.** *Circulation* 2007, **115**:928-935.
4. Pencina MJ, D'Agostino RB Sr, D'Agostino RB Jr, Vasan RS: **Evaluating the added predictive ability of a new marker: From area under the ROC curve to reclassification and beyond.** *Statistics in Medicine* 2008, **27**:157-172.
5. Baker SG: **The central role of receiver operating characteristic (ROC) curves in evaluating tests for the early detection of cancer.** *Journal of the National Cancer Institute* 2003, **95**:511-515.
6. Qi Y, Joseph ZB, Seetharaman JK: **Evaluation of different biological data and computational classification methods for use in protein interaction prediction.** *Proteins: Structure, Function, and Bioinformatics* 2006, **63**:490-500.
7. Hadjiiski L, Sahiner B, Chan HP, Petrick N, Helvie M: **Classification of malignant and benign masses based on hybrid ART2LDA approach.** *IEEE Transactions on Medical Imaging* 1999, **18**:1178-1187.
8. Dodd LE, Pepe MS: **Partial AUC estimation and regression.** *Biometrics* 2003, **59**:614-623.
9. Cai T, Dodd LE: **Regression analysis for the partial area under the ROC curve.** *Statistica Sinica* 2008, **18**:817-836.
10. Pepe MS, Thompson ML: **Combining diagnostic test results to increase accuracy.** *Biostatistics* 2000, **1**:123-140.
11. Pepe MS, Cai T, Longton G: **Combining predictors for classification using the area under the Receiver Operating Characteristic curve.** *Biometrics* 2006, **62**:221-229.
12. Komori O: **A boosting method for maximization of the area under the ROC curve.** *Annals of the Institute of Statistical Mathematics* 2009.
13. Ma S, Huang J: **Regularized ROC method for disease classification and biomarker selection with microarray data.** *Bioinformatics* 2005, **21**:4356-4362.
14. Wang Z, Chang YI, Ying Z, Zhu L, Yang Y: **A parsimonious threshold-independent pretein feature selection method through the area under receiver operating characteristic curve.** *Bioinformatics* 2007, **23**:2788-1794.
15. Freund Y, Schapire RE: **A decision-theoretic generalization of on-line learning and an application to boosting.** *Journal of computer and system sciences* 1997, **55**:119-139.
16. Friedman J, Hastie T, Tibshirani R: **Additive logistic regression: a statistical view of boosting.** *The Annals of Statistics* 2000, **28**:337-407.
17. Tutz G, Binder H: **Generalized Additive modeling with implicit variable selection by likelihood-based boosting.** *Biometrics* 2006, **62**:961-971.
18. Dodd LE, (Ed): **Regression methods for areas and partial areas under the ROC curve.** University of Washington: Ph.D. thesis; 2001.
19. Pepe MS, (Ed): **The Statistical Evaluation of Medical Tests for Classification and prediction.** New York: Oxford University Press; 2003.
20. Eguchi S, Copas J: **A class of logistic-type discriminant functions.** *Biometrika* 2002, **89**:1-22.
21. Bühlmann P, Yu B: **Boosting with the L_2 loss: regression and classification.** *Journal of the American Statistical Association* 2003, **98**:324-339.
22. Murata N, Takenouchi T, Kanamori T, Eguchi S: **Information geometry of *U*-boost and Bregman divergence.** *Neural Computation* 2004, **16**:1437-1481.
23. Hastie T, Tibshirani R, (Eds): **Generalized Additive Models.** Chapman & Hall; 1990.
24. McIntosh MW, Pepe MS: **Combining several screening tests: Optimality of the risk score.** *Biometrics* 2002, **58**:657-664.
25. Lugosi BG, Vayatis N: **On the Bayes-risk consistency of regularized boosting methods.** *The Annals of Statistics* 2004, **32**:30-55.
26. Neyman J, Pearson ES: **On the problem of the most efficient tests of statistical hypotheses.** *Philosophical Transaction of the Royal Society of London. Series A* 1933, **231**:289-337.
27. Pepe MS, Longton G, Anderson GL, Schummer M: **Selecting differentially expressed genes from microarray experiments.** *Biometrics* 2003, **59**:133-142.
28. Zhao P, Yu B: **Stagewise Lasso.** *Journal of Machine Learning Research* 2007, **8**:2701-2726.
29. Ben-Dor A, Bruhn L, Friedman N, Nachman I, Schummer M, Yakhini Z: **Tissue classification with gene expression profiles.** *Journal of Computation Biology* 2000, **7**:559-583.
30. Dettling M, Bühlmann P: **Boosting for tumor classification with gene expression data.** *Bioinformatics* 2003, **19**:1061-1069.
31. van't Veer LJ, Dai H, van de Vijver MJ, He YD, Hart AAM, Mao M, Peterse HL, van der Kooy K, Marton MJ, Witteveen AT, Schreiber GJ, Kerkhoven RM, Roberts C, Linsley PS, Bernards R, Friend SH: **Gene expression profiling predicts clinical outcome of breast cancer.** *Nature* 2002, **415**:530-536.
32. Golub TT, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA, Bloomfield CD, Lander ES: **Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring.** *Science* 1999, **286**:531-537.

doi: 10.1186/1471-2105-11-314
Cite this article as: Komori and Eguchi, A boosting method for maximizing the partial area under the ROC curve *BMC Bioinformatics* 2010, **11**:314