Mach Learn (2008) 72: 247–262
DOI 10.1007/s10994-008-5070-x

A critical analysis of variants of the AUC

Stijn Vanderlooy · Eyke Hüllermeier

Received: 22 June 2008 / Revised: 22 June 2008 / Accepted: 23 June 2008 / Published online: 15 July 2008

Abstract The area under the ROC curve, or AUC, has been widely used to assess the ranking performance of binary scoring classifiers. Given a sample, the metric considers the ordering of positive and negative instances, i.e., the sign of the corresponding score differences. From a model evaluation and selection point of view, it may appear unreasonable to ignore the absolute value of these differences. For this reason, several variants of the AUC metric that take score differences into account have recently been proposed. In this paper, we present a unified framework for these metrics and provide a formal analysis. We conjecture that, despite their intuitive appeal, actually none of the variants is effective, at least with regard to model evaluation and selection. An extensive empirical analysis corroborates this conjecture. Our findings also shed light on recent research dealing with the construction of AUC-optimizing classifiers.

Keywords ROC analysis · Area under the ROC curve · Ranking performance · Bias-variance analysis · AUC variants · AUC maximization

1 Introduction

In recent years, metrics for evaluating the performance of classifiers have received increasing attention in machine learning research. A common weakness of most metrics is that they are not robust to changes in the cost and class distributions governing the application domain (Provost et al. 1998). The area under the ROC (Receiver Operating Characteristics) curve, or simply AUC, has been advocated to be a robust scalar summary of the performance of a binary scoring classifier (Provost and Fawcett 2001; Ling et al. 2003). The metric assesses

Editors: Walter Daelemans, Bart Goethals, Katharina Morik.

S. Vanderlooy
MICC, Department of Computer Science, Maastricht University, Maastricht, The Netherlands
e-mail: s.vanderlooy@micc.unimaas.nl

E. Hüllermeier (✉)
Department of Mathematics and Computer Science, Marburg University, Marburg, Germany
e-mail: eyke@mathematik.uni-marburg.de

the ranking performance of a classifier without committing to a fixed threshold, which is required for mapping scores to binary class predictions.

The AUC is defined such that it only considers the sign of the differences between scores of pairs of positive and negative instances, while it ignores the absolute value of these score differences. In other words, it only depends on the ordering of the scores but not on the "margin" between them. Consequently, it can happen that a small change in scores leads to a considerable change in AUC value. Such an effect is especially apparent when the number of instances used to calculate the AUC is small. On the other hand, two classifiers can have the same AUC value, even though one of them is a "better separator" in the sense that it increases the difference between scores of positive and negative instances, respectively. It has been argued that this insensitivity toward score differences is disadvantageous for model evaluation and selection. For this reason, three variants of the AUC metric that take the score differences into account have recently been proposed (Ferri et al. 2005; Wu et al. 2007; Calders and Jaroszewicz 2007), along with first experimental results.

In this paper, we compare the conventional AUC and its three variants, both formally and empirically, within a unified framework. Our formal analysis leads us to conjecture that actually none of the variants should be able to outperform the conventional AUC with regard to model selection. This conjecture is then verified empirically on the basis of experiments with synthetic data and real benchmark data. Even though we do not invalidate previous experiments, our empirical study is arguably more extensive, especially since it considers different types of model selection scenarios. Finally, our contribution also sheds light on recent research dealing with the construction of classifiers that (approximately) optimize the AUC directly, rather than accuracy or another performance metric.

The remainder of the paper is organized as follows. In Sect. 2, we present the unified framework and explain the AUC metrics. In Sect. 3, we analyze these metrics in a formal way. An extensive experimental verification of our conjecture is provided in Sect. 4, and implications of our contribution to classifier construction are given in Sect. 5. Finally, Sect. 6 concludes the paper.

2 AUC and its variants

In this section, we first briefly introduce notation and recall the definition of the AUC metric. Afterward, we introduce a generalization and show that the three variants of the AUC can be represented as special cases thereof. The three variants are denoted, respectively, by scored AUC, soft AUC, and probabilistic AUC.

2.1 AUC

Consider an instance space $\mathcal{X}$ and let the sample space $\mathcal{X} \times \{-1, +1\}$ be endowed with a probability measure $\mathbb{P}$; thus, $\mathbb{P}(\boldsymbol{x}, c)$ denotes the probability to observe instance $\boldsymbol{x}$ with class label c. An instance $\boldsymbol{x}$ with class label $+1$ (-1) is called a positive (negative) instance. We refer to a scoring classifier f as an $\mathcal{X} \to [0, 1]$ mapping and $f(\boldsymbol{x})$ is interpreted as the probability or, more generally, as a degree of confidence that the class label of $\boldsymbol{x}$ is $+1$.

The AUC of a classifier f is equivalent to the probability that $f(\boldsymbol{x}) > f(\boldsymbol{y})$ given that $\boldsymbol{x}$ is a positive instance and $\boldsymbol{y}$ is a negative instance, both randomly drawn from the sample space. Empirically, the AUC has to be estimated on the basis of a sample (validation set) $S = \{(\boldsymbol{x}_i, c_i)\}_{i=1}^n \subseteq (\mathcal{X} \times \{-1, +1\})^n$. The estimate is given by the fraction of pairs $(\boldsymbol{x}_i, \boldsymbol{x}_j)$, with $\boldsymbol{x}_i$ a positive and $\boldsymbol{x}_j$ a negative instance such that $f(\boldsymbol{x}_i) > f(\boldsymbol{x}_j)$. So, we simply count

the number of correctly ordered pairs of instances with different class label. In case of a tie in the scores, $f(\boldsymbol{x}_i) = f(\boldsymbol{x}_j)$, the instance pair is counted with $1/2$ instead of assigning a full count of 1. It has been shown that this estimate of the AUC is unbiased and corresponds to the Wilcoxon-Mann-Whitney statistic (Mann and Whitney 1947; Hanley and McNeil 1982; Bradley 1997).

Obviously, an optimal classifier (i.e., a perfect ranker) has an AUC value of 1 while a value of 0.5 is obtained for a random classifier. An advantage of the AUC over other performance metrics is that it is invariant to changes in cost and class distributions since no threshold is fixed and applied on the scores. Also, even though the computational cost seems to grow quadratically in the number of instances, the metric can be computed in $\mathcal{O}(|S| \log |S|)$ by sorting the instances and keeping track of the ranks of the positives (Hand and Till 2001).

2.2 Generalization of AUC metrics

Given a sample S, consider the following generalization of the above estimate:

$$\text{gAUC}(f, S) = \frac{1}{|P| \cdot |N|} \sum_{\boldsymbol{x}_i \in P} \sum_{\boldsymbol{x}_j \in N} w\left(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j)\right), \tag{1}$$

where $P \subseteq S$ and $N \subseteq S$ denote, respectively, the subsets of positive and negative examples in S. The function $w(\cdot)$, that we shall call the *modifier function*, is a $[-1, 1] \to [0, 1]$ mapping and defines how to account for differences between the scores of positive and negative instances. For the conventional AUC metric, $w(\cdot)$ is given by (see Fig. 1a)

$$w(t) = w^*(t) = \begin{cases} 1 & \text{if } t > 0, \\ 0.5 & \text{if } t = 0, \\ 0 & \text{if } t < 0. \end{cases} \tag{2}$$

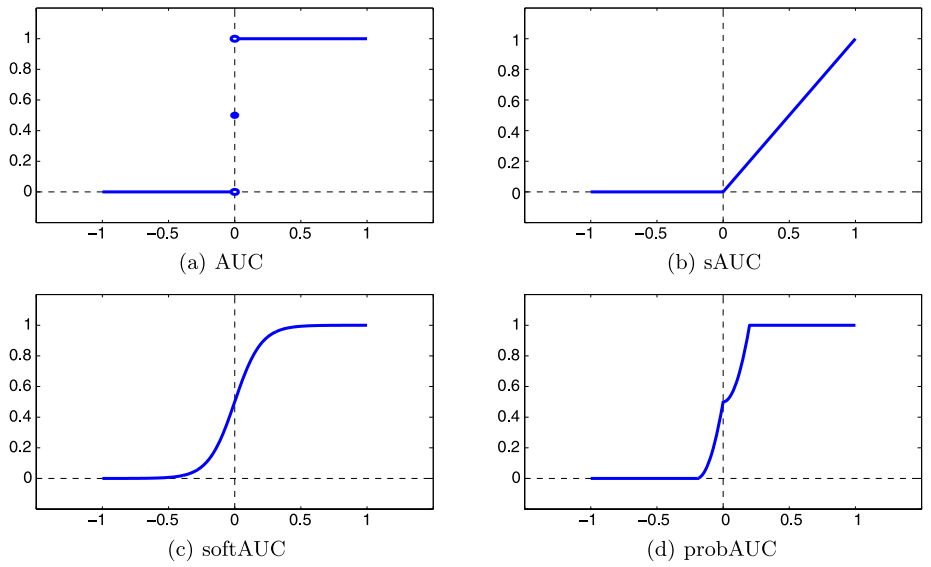

Fig. 1 Modifier functions that are used by the different AUC metrics

Henceforth, we refer to pairs $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in P \times N$ as PN-pairs and the differences in scores $f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j)$ are denoted *score margins*. A PN-pair is said to be correctly ordered or concordant when $f(\boldsymbol{x}_i) > f(\boldsymbol{x}_j)$; when the score margin is negative, it is discordant. Hence, with this terminology, we can say that the estimate of the AUC is calculated by counting the number of concordant PN-pairs and PN-pairs with zero score margin.

2.3 sAUC

The value of the AUC is invariant with respect to the score margin as long as its sign remains unchanged. However, as mentioned in the introduction, it appears intuitively reasonable to prefer larger score margins to smaller ones. Moreover, it was argued in Wu et al. (2007) that the concordance of a PN-pair is in a sense less reliable when the score margin is small. In order to make the evaluation of a classifier more robust, the authors proposed a variant of the AUC metric (called scored AUC) that takes the absolute value of the score margins into account.

The key idea is to count a PN-pair only in case the score margin exceeds a threshold $\tau \in [0, 1]$. It follows that the conventional AUC is recovered for $\tau = 0$, while the modified AUC is a decreasing function of τ. A value of 0 is obtained for $\tau = 1$.[1] The effect of different thresholds can be visualized by plotting each τ against the modified AUC. As a result, a piecewise linear curve is obtained in which the drops occur at the τ's that equal the score margin of one of the PN-pairs. The area under this curve aggregates the robustness (or sensitivity) of the conventional AUC over all possible thresholds, hereby preventing the user from committing to a single τ. The area is called the *scored* AUC (sAUC). It is straightforward to show that the sAUC equals (1) with the following modifier function (see Fig. 1b):

$$w(t) = \begin{cases} t & \text{if } t > 0, \\ 0 & \text{if } t \leq 0. \end{cases} \tag{3}$$

Thus, the contribution of a PN-pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ to the evaluation of a classifier is the score margin $t = f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j)$ if this score margin is positive, and 0 otherwise. A simple decomposition of the metric shows that it can be computed in linear time.

2.4 softAUC

We have seen that the scored AUC punishes classifiers that produce small score margins, as these are considered as uncertain, and therefore may contribute to the AUC just by chance. A second variant of the AUC metric, called softAUC, has originally been proposed as a differentiable approximation of AUC amenable to learning algorithms requiring a continuous objective function. For example, it can be used by a gradient descent routine to find a hyperplane that approximately maximizes the AUC (Calders and Jaroszewicz 2007). Nonetheless, the softAUC also fits the purpose of this paper and it can be represented as a special case of (1) using a sigmoidal modifier function (see Fig. 1c):

$$w(t) = \frac{1}{1 + \exp(-\beta t)}, \tag{4}$$

[1] We note that the effect of a particular τ can also be seen as smoothing the ROC curve along the parts where the score margins are not large enough.

with $\beta \in]0, \infty[$. We note that a large β implies that the sigmoid approximates the step function, i.e., softAUC converges to conventional AUC for $\beta \to \infty$.

The sigmoid function automatically smoothes out the region around zero score margin. Its computational cost is quadratic, but accurate approximations that are computable in linear time can be used as an alternative (Herschtal and Raskutti 2004; Calders and Jaroszewicz 2007).

2.5 probAUC

A third and last variant is the probabilistic AUC (probAUC). The key idea of this metric has been introduced in Ferri et al. (2005), but it has not been elaborated further at the present time. Yet, it is a more rigorous realization of the idea also underlying sAUC, namely that a score is considered as a "noisy" observation of a true score.

Given an estimated score $f(\boldsymbol{x}_i)$, the true score of the instance is modeled as a random variable uniformly distributed in $[f(\boldsymbol{x}_i) - h, f(\boldsymbol{x}_i) + h]$. Then, given a positive instance with a score in $[a - h, a + h]$ and a negative instance with a score in $[b - h, b + h]$, the probability that this PN-pair is concordant is[2]

$$\int_{a-h}^{a+h} \int_{b-h}^{x} (2h)^{-2}\, dy\, dx.$$

The above probability only depends on $t = a - b$ and is given by (see Fig. 1d)

$$w(t) = \begin{cases} 1 & \text{if } t \geq h, \\ \max(0, \frac{t}{2h}) + \frac{1}{2}(1 - \frac{|t|}{2h})^2 & \text{if } -h < t < h, \\ 0 & \text{if } t \leq -h. \end{cases} \tag{5}$$

We denote by probAUC the generalization (1) with (5) as a modifier function.

It is clear that the width $2h$ of the interval defines the level of smoothing, and probAUC converges to the conventional AUC for $h \to 0$. Instead of assuming a uniform distribution, other distributions such as a truncated Gaussian or a triangular can of course be considered. The computational cost using a uniform distribution grows quadratically in the number of instances.

3 Formal analysis of the AUC metrics

In this section, we present our formal analysis of the four aforementioned AUC metrics. The resulting conjecture is also illuminated and verified using experiments with synthetic data. We start this section with some potential problems and critical issues concerning the AUC variants.

3.1 Potential problems and critical issues

From the modifier functions shown in Fig. 1, it is obvious that sAUC is the most extreme modification of the conventional AUC. On the other hand, it is also the variant that has been

[2]Here, we ignore that the boundary cases $a > 1 - h$ and $a < h$ (and the same cases for b) do actually need special treatment.

investigated, with regard to model evaluation and selection, most thoroughly by means of experimental studies (Wu et al. 2007).

Independent of this argument, we would like to point out two potential disadvantages of sAUC. The first one is a kind of asymmetry of its modifier function (3), which considers concordant PN-pairs with small score margin as potentially discordant pairs, but not the other way round. That is, PN-pairs whose score margin is negative and small in absolute value (the negative instance in the pair has a slightly higher score than the positive instance) are not considered as potentially concordant pairs. The second problem is that, by aggregating over all possible margins, sAUC implicitly assumes that the classifiers to be evaluated produce scores in the same range. As a consequence, a classifier that produces scores close to the extreme values 0 and 1 is likely to be preferred over a classifier that produces less extreme values, even if the latter makes fewer ranking mistakes. As an illustration, consider the following two classifiers and their scores on a validation set consisting of four positive and three negative instances:

$$f_1 : 0.7+\ 0.7+\ 0.7+\ 0.7+\ 0.3-\ 0.3-\ 0.3-,$$
$$f_2 : 1.0+\ 1.0+\ 1.0+\ 0.0+\ 1.0-\ 0.0-\ 0.0-.$$

The classifier f_1 is a perfect ranker and therefore has maximal AUC. Yet, it has a low sAUC value (0.4 to be precise). Classifier f_2 has lower AUC since it gives a score of 0 and 1 to a positive and negative instance, respectively. Its sAUC value is however 0.5.[3] Thus, in this example, model selection based on sAUC would clearly lead to a questionable and, assuming that the sample is representative of the population, even incorrect choice. We note that the other two variants, softAUC and probAUC, overcome the first problem and strongly alleviate the second problem.

Finally, we remark that, in our opinion, the idea of considering a score $f(\boldsymbol{x}_i)$ as a kind of uncertain measurement (random variable) lacks a convincing theoretical justification. In fact, when the classifier is fixed (as it is the case in model evaluation), then the scores produced for instances $\boldsymbol{x}_i$ are determined in a deterministic way. Of course, changing the instances, i.e., evaluating a classifier on a different validation set, will also change the scores. Here, however, the randomness is introduced by the selection of the $\boldsymbol{x}_i$ and not by their scoring. Consequently, statistical properties of sampling are transferred to properties of scoring. In particular, assuming that the validation set is a representative sample, the obtained set of scores $f(\boldsymbol{x}_i)$ is also a representative sample of the scores produced by f. This point immediately leads us to the next subsection where we present a bias and variance analysis of the metrics.

3.2 Bias and variance of estimation

Recall that the goal in model selection is to select, among a set of candidates, a single model whose *true* AUC is highest. From a statistical point of view, the empirical AUC of a model f on a validation set S is clearly a good estimator, especially since it is an *unbiased* estimate of the true AUC value.

As opposed to this, it is obvious that $\mathrm{sAUC}(f, S) \leq \mathrm{AUC}(f, S)$ for all classifiers f, and often the inequality will be strict. Therefore, sAUC produces a biased estimate. More

[3] In fact, for f_2, the $\tau \mapsto \mathrm{AUC}(\tau)$ function is simply a horizontal line at the height of the conventional AUC value.

interestingly, since less obviously, even the symmetric modifier function used by softAUC will usually produce a biased estimate. To see this, we note that the expected value of a modified AUC metric is given by the expected value of $w(T)$, where T is the score margin for a randomly chosen PN-pair $(\boldsymbol{x}, \boldsymbol{y})$. Denote the cumulative distribution function (CDF) of T by $G(\cdot)$, i.e., $G(t) = \mathbb{P}(T \leq t)$, and let $g(\cdot)$ be the corresponding probability distribution function (PDF). The expected value is then given by

$$\mathbb{E}(w(T)) = \int_{-1}^{1} w(t)\, dG(t) = \int_{-1}^{1} w(t) g(t)\, dt.$$

Now, for any reasonable classifier f, one should expect that the PDF $g(\cdot)$ is monotone increasing, which means that higher score margins are not less probable than lower margins. More specifically, suppose that $g(t) \geq g(-t)$ for all $t \geq 0$. Using the property $w(-t) \leq 1 - w(t)$, which can easily be shown to hold for all three AUC variants, the difference between $\mathbb{E}(w(T))$ and the expected value of the conventional AUC can be bounded as follows:

$$\begin{aligned}
\mathbb{E}(w(T)) - \mathbb{E}(w^*(T)) &= \int_{-1}^{1} (w(t) - w^*(t)) g(t)\, dt \\
&= \int_{0}^{1} w(-t) g(-t) + (w(t) - 1) g(t)\, dt \\
&\leq \int_{0}^{1} (1 - w(t)) g(-t) - (1 - w(t)) g(t)\, dt \\
&= \int_{0}^{1} \underbrace{(1 - w(t))}_{\geq 0} \underbrace{(g(-t) - g(t))}_{\leq 0}\, dt \\
&\leq 0.
\end{aligned}$$

Thus, the true AUC is underestimated by the all three variants of the AUC.

Of course, biased estimation is not disadvantageous per se. First, with regard to model selection, a bias does actually not have any influence as long as it is constant, i.e., independent of the model. However, this is not guaranteed in our context since the PDF $g(\cdot)$ depends on the classifier f, and therefore is model-specific. Second, it is known in statistics that, by biasing an estimation, it is sometimes possible to reduce variance and, thereby, to obtain more precise estimations (Friedman 1997). Indeed, since $w(-t) \leq 1 - w(t)$ in conjunction with $0 \leq w(t) \leq 1$ also implies $w(-t)^2 \leq 1 - w(t)^2$, we can show in the same way as above that

$$\mathbb{E}(w(T)^2) - \mathbb{E}(w^*(T)^2) \leq 0,$$

and consequently comparing the variances gives

$$\mathbb{V}(w(T)) - \mathbb{V}(w^*(T)) = \underbrace{(\mathbb{E}(w(T)^2) - \mathbb{E}(w^*(T)^2))}_{\leq 0} - \underbrace{(\mathbb{E}^2(w(T)) - \mathbb{E}^2(w^*(T)))}_{\leq 0}.$$

This means that the change in variance can go into both directions. Examples for these two cases in terms of suitable PDFs $g(\cdot)$ can easily be constructed.

In summary, we conclude that the AUC variants produce estimates of the true AUC with a non-constant bias, which is a disadvantage. On the other hand, they may potentially reduce variance, which would be an advantage. First, however, this is not guaranteed since

they may as well increase variance. Second, this potential advantage will only be relevant for relatively small data sets, as the variance of the estimate decreases with sample size and, hence, becomes less important. All things considered, we do not see strong reasons to believe that any of the variants should outperform the conventional AUC in model evaluation and selection. On the contrary, we conjecture that the conventional AUC will show superior performance in this regard, a conjecture that we shall corroborate by means of several experimental studies.

3.3 Simulation studies with synthetic data

In this subsection, we describe the experimental setup and results from two simulation studies with synthetic data. For these studies, we have parametrized the softAUC with $\beta = 3$ and $\beta = 10$ in (4) to see the effect of parameter change. The results for probAUC are omitted for ease of presentation since probAUC behaves very similarly to softAUC (indeed, as can be seen in Fig. 1, the corresponding modifier functions have strong resemblance).

In a first experiment, we have simulated a classifier producing scores according to an exponential PDF truncated to [0, 1]:

$$\mathbb{P}(f(\boldsymbol{x}_i) \mid c) = \frac{\alpha}{1 - \exp(-\alpha)} \cdot \exp\left(-\alpha\, d(\boldsymbol{x}_i)\right),$$

where $d(\boldsymbol{x}_i) = 1 - f(\boldsymbol{x}_i)$ if $c = +1$ and $d(\boldsymbol{x}_i) = f(\boldsymbol{x}_i)$ if $c = -1$. The value of α determines the "strength" of the associated classifier: higher values decrease the probability of an incorrectly ordered PN-pair. In total, we generated 50 scores for positive instances and 50 scores for negative instances, and afterward computed the four AUC metrics on this sample. This experiment was repeated 5000 times to approximate expected values by averages. In Fig. 2, we show the bias and variance of the obtained values of the AUC metrics as a function of the parameter $\alpha = 1, 2, \ldots, 10$. In agreement with our theoretical results, we find that the AUC variants are indeed underestimates of the conventional AUC. As expected, the bias reduces when the strength of the classifier is increased since the generated scores lie close to the boundaries 0 and 1. It is also clear that, for sAUC, not only the bias but also the variance remains high, even for large α. We also see that a larger β for softAUC gives better results, verifying our conjecture that conventional AUC is still the best performance metric (recall

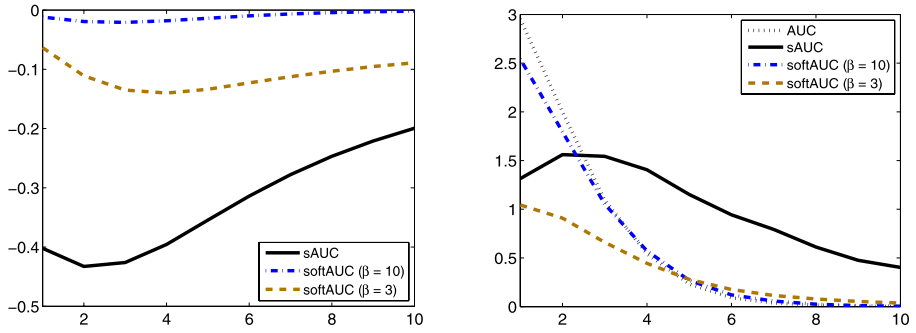

Fig. 2 Results of each AUC metric obtained in the first simulation experiment: (*left*) the bias, and (*right*) the variance with scale y-axis equal to 10^{-3}. Bias and variance of sAUC remains high. Also, for softAUC, a steep modifier function gives the best results

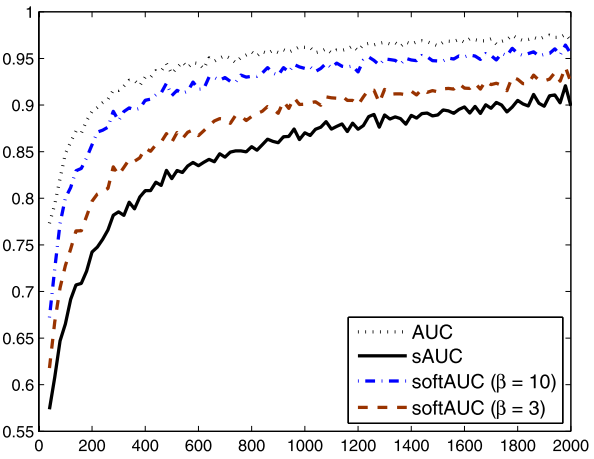

Fig. 3 Average fraction that each AUC metric selects the best model in the second simulation experiment. The sAUC metric performs worst throughout the setup

that softAUC converges to conventional AUC for $\beta \to \infty$). A similar claim can be made for probAUC.

In a second simulation experiment, we mimic a model selection scenario as follows. A data set is randomly generated with each of its two features in the interval $[-1/\sqrt{2}, +1/\sqrt{2}]$. In this way, the perpendicular distance of a test instance to the linear model lies in the interval [0, 1] and therefore can be used as a score without further modification. A linear model defined by a random weight vector and passing through the origin is used to label these instances. Two other suboptimal models are included in the model selection by adding Gaussian noise to the weight vector. Moreover, we randomly switch 10% of the labels of the positives and the negatives to make the selection harder. Afterward, we compute the AUC metrics for each of the three models. This experiment is repeated for 1000 times and the number of instances ranges from 40 to 2000. In Fig. 3, we show the average number of times that each AUC metric selects the best model. It is clear that sAUC performs extremely poor throughout the complete setup, while the other variants can be considered as competitive to conventional AUC.

The above experiments confirm what could be expected from our theoretical conjecture: none of the proposed variants is more effective than the conventional AUC. In the next section we provide even stronger evidence for this conjecture by means of an extensive set of experiments on real benchmark data.

4 Experimental analysis of the AUC metrics

In this section, we present a number of experiments on 16 binary benchmark data sets from the UCI repository (Asuncion and Newman 2007). These data sets and their most important characteristics are given in Table 1. We note that the data sets contain at most 1000 instances since, as already mentioned earlier, a positive effect of the modified versions of AUC, if any, is to be expected only for small data sets. For all experiments we used the `WEKA` machine learning software (Witten and Frank 2005).

We assess the performance of AUC, sAUC, softAUC, and probAUC as model selection criteria for two different settings. In the first setting, we replicated the experimental setup of Wu et al. (2007) where sAUC was introduced. Using this setup, the authors were able to show that this variant can indeed outperform AUC, albeit with very small differences. Our

Table 1 The sixteen UCI data sets: (1) reference number, (2) name, (3) number of instances, (4) number of features, and (5) percentage of the majority class

#	name	size	dim	% maj class	#	name	size	dim	% maj class
1	breast cancer	286	9	70.28	9	monks1	556	6	50.00
2	credit rating	690	15	55.51	10	monks2	604	6	65.72
3	german credit	1000	20	69.40	11	monks3	554	6	55.41
4	heart statlog	270	13	59.50	12	pima	768	9	65.10
5	horse colic	368	22	63.04	13	sonar	208	61	53.36
6	house votes	435	17	38.62	14	spect	267	23	58.80
7	ionosphere	351	35	35.90	15	tic-tac-toe	958	10	65.34
8	liver	345	7	42.03	16	breast wisc	699	9	65.52

results are largely in agreement with these experiments. However, by analyzing the results in more detail, we also conclude that they must be considered with reservation, due to the special characteristics of the setting. Therefore, we perform a second study using a setting with different and, from a model selection point of view, arguably more realistic characteristics. The parameters for softAUC and probAUC are $\beta = 10$ and $h = 0.1$, respectively. We determined these parameters such that the modifier functions behave steeply around the region of zero score margin, although not too steep to remain distinguishable from the conventional AUC.

4.1 Setting 1

A data set is partitioned into two equal-sized parts using a stratified split. One half is used as a training set and the other half is partitioned (again stratified) into 20% validation set and 80% test set. Ten different classifiers are trained with the same learning algorithm (J48 unpruned and with Laplace correction, naïve Bayes with kernel density estimation, and logistic regression) by randomly removing three features before training. The best classifier is then selected according to each of the four AUC metrics using the validation set. Finally, the performance of each selected model is assessed by comparing its AUC on the test set with that of the true best classifier. Henceforth, we call the difference between these two empirical AUCs the regret (which means that the regret of the best classifier is 0). Repeating this procedure 2000 times, we report the average regret per data set, learning algorithm, and AUC metric.

The results are given in Table 2 and they clearly show that no AUC metric is able to outperform any of the other metrics, regardless of the learning algorithm. In fact, the differences between the regrets are very small throughout. Moreover, from the regrets alone, it is impossible to conclude whether the differences between the metrics are due to small variations across the multiple runs or large differences in a few runs that represent a situation in which one of the metrics is clearly favoured. In Fig. 4, we therefore present the win-loss-equal statistics for each combination of two AUC metrics, as gathered over the 2000 runs for logistic regression (similar results were obtained for the other learning algorithms). These statistics are encoded as a horizontal bar chart for each data set, where the length of the bar to the left (right) of the baseline represents the fraction of wins (losses) for each metric combination. As can be seen, softAUC and probAUC often perform en par with AUC and thus often select the same model. Regarding the comparison of AUC and sAUC, the results are rather diverse and do not provide a clear picture. To explain these results, we have observed that most

Table 2 The average regret in the first setting for each data set, learning algorithm, and AUC metric. The first column shows the results for AUC, the second for sAUC, the third for softAUC, and the fourth column shows the results for probAUC

Data set	J48				NB				Logistic			
1	.0580	.0774	.0759	.0745	.0364	.0237	.0345	.0371	.0458	.0488	.0455	.0459
2	.0089	.0084	.0130	.0133	.0074	.0096	.0095	.0086	.0100	.0136	.0107	.0103
3	.0273	.0269	.0279	.0273	.0116	.0100	.0113	.0116	.0144	.0128	.0140	.0145
4	.0322	.0318	.0337	.0347	.0202	.0150	.0204	.0211	.0219	.0200	.0221	.0224
5	.0187	.0139	.0183	.0201	.0099	.0093	.0095	.0098	.0218	.0251	.0256	.0259
6	.0033	.0021	.0045	.0055	.0035	.0041	.0040	.0045	.0083	.0072	.0106	.0114
7	.0088	.0168	.0154	.0153	.0040	.0035	.0038	.0038	.0406	.0416	.0411	.0410
8	.0238	.2014	.1898	.1890	.0462	.0432	.0469	.0471	.0472	.0479	.0462	.0461
9	.0138	.0086	.0103	.0126	.0154	.0157	.0155	.0152	.0161	.0167	.0154	.0158
10	.0358	.1496	.1381	.1350	.0379	.0446	.0393	.0394	.0378	.0437	.0405	.0403
11	.0019	.0034	.0031	.0035	.0041	.0042	.0040	.0042	.0034	.0032	.0033	.0033
12	.0198	.0547	.0507	.0514	.0140	.0114	.0117	.0134	.0159	.0074	.0118	.0150
13	.0209	.0235	.0235	.0229	.0106	.0104	.0103	.0100	.0284	.0293	.0293	.0290
14	.0378	.0417	.0416	.0408	.0144	.0119	.0139	.0151	.0252	.0248	.0249	.0247
15	.0198	.0173	.0197	.0199	.0190	.0251	.0186	.0191	.0186	.0102	.0168	.0179
16	.0037	.0040	.0041	.0042	.0030	.0028	.0030	.0032	.0021	.0020	.0023	.0023

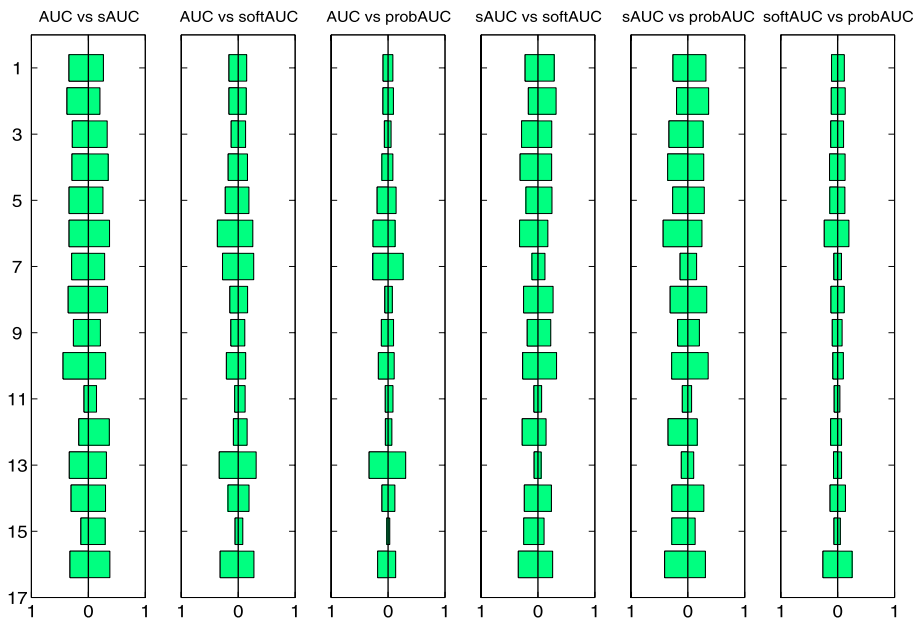

Fig. 4 Win-Loss-Equal statistics for the AUC metrics in the first setting. The length of the *left* (*right*) bar of each combination of two metrics represents the fraction of wins (losses), and the fraction of equals is given by one minus the total length of the bars

of the time a small number of $k \ll 10$ candidate classifiers have a similar validation AUC value while the rest is significantly worse. This finding is not surprising given the setup of the experiments, namely, a classifier achieves a good performance only if it is trained on the important features. Given this, the small differences in the average regrets can be attributed to the different bias of the metrics. More specifically, all metrics select one of the top-k classifiers with a high probability. Yet, while AUC chooses the top-1 model with probability 1, and softAUC and probAUC are likely to select the same model due to their relatively small bias, the larger bias of sAUC causes it to make a selection in a more or less random way. This, however, is actually not a disadvantage: Due to their almost equal performance, each of the top-k classifiers has more or less the same chance to achieve the best AUC on the test set, which explains that sAUC is indeed competitive.

Besides, it is important to note that the experimental setup only compares classifiers of the same type. Our discussion in Sect. 3.1 and the example given there suggest that this is again favourable for sAUC. Indeed, we have argued that sAUC is expected to have problems when it comes to comparing classifiers producing scores with different distribution or in a different range. In practice, this is often the case, for example when having to decide between classifiers from different learning algorithms such as a decision tree and a naïve Bayes classifier. And even when remaining within the same model class, changing the parameter setting(s) can have a large influence on the distribution of scores. Next, we analyze an example of that kind.

4.2 Setting 2

For the second setting, we construct a training set, validation set, and test set as in the first setting. Two different classifiers are then learned by applying J48 with and without Laplace correction. Laplace correction adds a pseudo-count of 1 to the class frequencies in the leaves, and therefore, shifts the scores away from the extreme values 0 and 1 more toward the middle. Trees with Laplace correction in the leaves have been shown to systematically outperform trees without Laplace correction (Ferri et al. 2003). The candidate models are decision trees without pruning since they have been shown to produce the best AUC values (Provost and Domingos 2003).

In this experimental setting, it is expected that sAUC will select more often the unpruned tree without Laplace correction which, on average, has lower test AUC value. We do not expect large differences between softAUC and probAUC, although they should select better (worse) models than sAUC (AUC) does.

The average regrets of the metrics are reported in Table 3. The results confirm our expectations. The conventional AUC has lowest average regret in all but one data set. Also, there is no clear distinction between softAUC and probAUC, and most of the time both metrics choose the same models as AUC does. The sAUC metric performs significantly worse in all data sets since it is misled by the extreme scores produced by the unpruned trees without Laplace correction. The win-loss-equal statistics depicted in Fig. 5 give additional details. Clearly, all metrics except for sAUC often choose the same (best) model.

5 Implications for AUC-optimizing classifiers

In the previous section, we focused on using the AUC for model selection. The metric is typically not used as an optimization criterion in the learning phase of a classifier. Often, learning is aimed to optimize the error rate, cross-entropy, or mean squared error. Optimizing

Table 3 The average regret in the second setting for each data set and AUC metric. The first column shows the results for AUC, the second for sAUC, the third for softAUC, and the fourth column shows the results for probAUC

Data set	J48 with and without pruning			
1	.0132	.0264	.0143	.0144
2	.0018	.0559	.0018	.0016
3	.0022	.0712	.0044	.0033
4	.0077	.0577	.0081	.0081
5	.0072	.0227	.0076	.0073
6	.0018	.0093	.0039	.0022
7	.0064	.0344	.0073	.0065
8	.0081	.0201	.0092	.0092
9	.0025	.0066	.0029	.0026
10	.0062	.0188	.0075	.0070
11	.0011	.0031	.0017	.0012
12	.0032	.0308	.0031	.0035
13	.0113	.0430	.0121	.0121
14	.0086	.0408	.0100	.0096
15	.0013	.0397	.0026	.0019
16	.0036	.0292	.0025	.0035

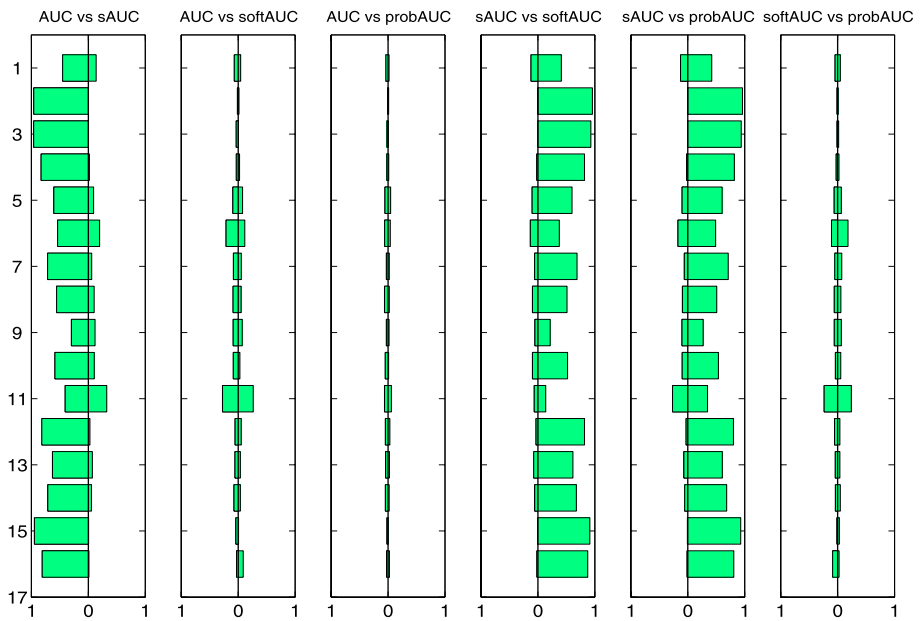

Fig. 5 Win-Loss-Equal statistics for the AUC metrics in the second setting. The length of the *left* (*right*) bar of each combination of two metrics represents the fraction of wins (losses), and the fraction of equals is given by one minus the total length of the bars

with respect to these metrics does however not guarantee a model with maximum AUC (Yan et al. 2003; Cortes and Mohri 2003; Tax and Veenman 2005; Caruana and Niculescu-Mizil 2006). In this section we briefly review two main research directions in learning AUC-optimizing classifiers, and we show how our results shed new light on this research and its results so far.

The first research direction is the formulation of an objective function and constraints for its optimization. Directly optimizing AUC is however hard since it is non-differentiable and many solutions may exist. For these reasons, a regularized SVM-like convex optimization problem has been proposed (Rakotomamonjy 2004; Brefeld and Scheffer 2005; Tax et al. 2006). Despite the high computation time of this AUC-SVM, experimental results do not show consistent increases in test AUC. A first reason has recently been given by Steck (2007), where it was shown that minimizing the hinge loss is an accurate approximation to maximizing the AUC. This implies that an SVM already has high AUC. A second reason can be given from the results in this paper. Without going into too much detail, a slack variable $\xi_{ij} = 1$ only when $f(\boldsymbol{x}_i) = f(\boldsymbol{x}_j)$ and $\xi_{ij} > 1$ when the PN-pair is incorrectly ordered. The larger the absolute value of the score margin of discordant pairs, the higher the slack. Thus, the mapping from PN-pairs to slacks is similar to the sAUC modifier function (except for a shifting and reflection, as it serves as a penalty function). As we have seen, however, optimizing sAUC is clearly different from optimizing AUC. We therefore presume that, for the same reason, the approach may fail as an AUC-maximizer.

The second research direction is based on gradient descent routines. To make such routines applicable, the AUC needs to be approximated using a function that is continuous and differentiable. A popular choice is a steep sigmoid function. Therefore, the proposed algorithms learn classifiers that optimize the softAUC, and indeed it has been verified that these classifiers have higher test AUCs than those obtained by AUC-SVMs (Herschtal and Raskutti 2004; Calders and Jaroszewicz 2007). Also, it has been observed that significantly better test AUCs are obtained by increasing β in (4) (Calders and Jaroszewicz 2007). Our results give a well-founded explanation for these findings. Of course, a disadvantage of the gradient descent routines is that they are restricted to learning hyperplanes in input space. It would therefore be an interesting direction of future research to extend AUC-SVMs to incorporate proper modifier functions, although improvements are likely to be small.

6 Conclusions

As it may appear unreasonable for a ranking performance metric to ignore the absolute value of the score margin of PN-pairs, several authors have argued for modified AUC metrics. These variants are less favorable toward models that generate small score margins and instead prefer models producing large margins, as small margins are considered as uncertain, and therefore may contribute to the AUC just by chance.

In this paper, we provided a critical analysis of this approach. We presented a general model that allows for a unified treatment of the conventional metric and its variants. A formal analysis addressed the bias and variance of the estimates of the true AUC value. The results of this analysis are supported by strong empirical evidence, which leads us to conjecture that none of the variants are as effective as the AUC itself.

Based on the findings of this work, we draw the following conclusions. First, the AUC variants are all biased and their variance can go in either direction. The net effect on the quality of the estimations is thus not clear and, hereby, there is no solid theoretical foundation for the variants. Second, our empirical results have shown that the conventional AUC

cannot be outperformed systematically, not in an ideal setting according to the theoretical analysis, and not in real model selection scenarios. The variants with a modifier function that closely resembles the step function perform best. In this respect we can also see that softAUC and probAUC, with properly chosen parameters, are accurate approximations of the conventional AUC metric. Third, we may conclude that AUC-optimizing model learning works best with a symmetric modifier function that sharply smoothes out the region around zero score margin. For approaches based on gradient descent, this may, however, be problematic since the gradient in this region might become numerically unstable. Therefore, an interesting direction of future work is to enhance AUC-SVMs to incorporate the softAUC metric.

We end with a simple, though interesting observation that we have not found elsewhere in the literature. Ignoring ties, the empirical AUC can be seen as the maximum likelihood estimation of a binomial distribution with success parameter equal to the true AUC value (the probability that a randomly chosen PN-pair is concordant). Computing the Bayes' estimator might therefore seem to be a reasonable alternative since it has regularization build in. The uniform distribution (or a more flexible beta distribution) can be used as prior, the binomial as likelihood, and the posterior is then again a beta distribution. However, the corresponding Bayes' estimator simply results in a Laplace correction of the maximum likelihood estimator, and therefore does not change the ordering of the classifiers in a model selection scenario.

Acknowledgements We would like to thank Johannes Fürnkranz for useful comments on an early draft of this paper. Stijn Vanderlooy is supported by the Dutch Organization for Scientific Research (NWO), ToKeN programme, viz. the IPOL project, grant nr.: 634.000.435.

References

Asuncion, A., & Newman, D. (2007). UCI machine learning repository.

Bradley, A. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, *30*(7), 1145–1159.

Brefeld, U., & Scheffer, T. (2005). AUC maximizing support vector learning. In Ferri, C., Lachiche, N., Macskassy, S., & Rakotomamonjy, A. (Eds.), *Proceedings of the 2nd workshop on ROC analysis in machine learning (ROCML 2005)*. Bonn, Germany, August 11, 2005.

Calders, T., & Jaroszewicz, S. (2007). Efficient AUC optimization for classification. In J. Kok, J. Koronacki, R. L. de Mántaras, S. Matwin, D. Mladenic, & A. Skowron (Eds.), *Proceedings of the 11th European conference on principles and practice of knowledge discovery in databases (PKDD 2007)* (pp. 42–53). Warsaw, Poland, September 17–21, 2007. Berlin: Springer.

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In W. Cohen & A. Moore (Eds.), *Proceedings of the 23rd international conference on machine learning (ICML 2006)* (pp. 161–168). Pittsburgh, PA, USA, June 25–29, 2006. New York: Assoc. Comput. Mach.

Cortes, C., & Mohri, M. (2003). AUC optimization vs. error rate minimization. In S. Thrun, L. Saul, & B. Schölkopf (Eds.), *Advances in neural information processing systems 16 (NIPS 2003)*. Vancouver, BC, Canada, December 8–13, 2003. Cambridge: MIT Press.

Ferri, C., Flach, P., & Hernández-Orallo, J. (2003). Improving the AUC of probabilistic estimation trees. In N. Lavrac, D. Gamberger, L. Todorovski, & H. Blockeel (Eds.), *Proceedings of the 14th European conference on machine learning (ECML 2003)* (pp. 121–132). Cavtat-Dubrovnik, Croatia, September 22–26, 2003. Berlin: Springer.

Ferri, C., Flach, P., Hernández-Orallo, J., & Senad, A. (2005). Modifying ROC curves to incorporate predicted probabilities. In C. Ferri, N. Lachiche, S. Macskassy, & A. Rakotomamonjy (Eds.), *Proceedings of the 2nd workshop on ROC analysis in machine learning (ROCML 2005)*. Bonn, Germany, August 11, 2005.

Friedman, J. (1997). On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, *1*, 55–77.

Hand, D., & Till, R. (2001). A simple generalisation of the area under the ROC curve for multiple class classification problems. *Machine Learning*, *45*, 171–186.

Hanley, J., & McNeil, B. (1982). The meaning and use of the area under a receiver operator characteristic ROC curve. *Radiology*, *143*(1), 29–36.

Herschtal, A., & Raskutti, B. (2004). Optimising area under the ROC curve using gradient descent. In C. Brodley (Ed.), *Proceedings of the 21st international conference on machine learning (ICML 2004)*. Banff, Alberta, Canada, July 4–8, 2004. New York: Assoc. Comput. Mach.

Ling, C., Huang, J., & Zhang, H. (2003). AUC: a statistically consistent and more discriminating measure than accuracy. In G. Gottlob & T. Walsh (Eds.), *Proceedings of the 18th international joint conference on artificial intelligence (IJCAI 2003)* (pp. 519–526). Acapulco, Mexico, August 9–15, 2003. Menlo Park: AAAI Press.

Mann, H., & Whitney, D. (1947). On a test whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, *18*(1), 50–60.

Provost, F., & Domingos, P. (2003). Tree-induction fir probability based ranking. *Machine Learning*, *52*(3), 199–215.

Provost, F., & Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning*, *42*(3), 203–231.

Provost, F., Fawcett, T., & Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. In J. Shavlik (Ed.), *Proceedings of the 15th international conference on machine learning (ICML 1998)* (pp. 43–48). Madison, WI, USA, July 24–27, 1998. San Mateo: Morgan Kaufmann.

Rakotomamonjy, A. (2004). Optimizing area under ROC curve with SVMs. In J. Hernández-Orallo, C. Ferri, N. Lachiche, & P. Flach (Eds.), *Proceedings of the 1st workshop on ROC analysis and artificial intelligence (ROCAI 2004)* (pp. 71–80). Valencia, Spain, August 22, 2004.

Steck, H. (2007). Hinge rank loss and the area under the ROC curve. In J. Kok, J. Koronacki, R. L. de Mántaras, S. Matwin, D. Mladenic, & A. Skowron (Eds.), *Proceedings of the 18th European conference on machine learning (ECML 2007)* (pp. 347–358). Warsaw, Poland, September 17–21, 2007. Berlin: Springer.

Tax, D., & Veenman, C. (2005). Tuning the hyperparameter of an AUC-optimized classifier. In K. Verbeeck, K. Tuyls, A. Nowe, B. Manderick, & B. Kuijpers (Eds.), *Proceedings of the 17th Belgium-Netherlands conference on artificial intelligence (BNAIC 2005)* (pp. 224–231). Brussels, Belgium, October 17–18, 2005. Brussels: Royal Flemish Academy of Belgium for Science and Arts.

Tax, D., Duin, R., & Arzhaeva, Y. (2006). Linear model combining by optimizing the area under the ROC curve. In Y. Tang, P. Wang, G. Lorette, D. Yeung, & H. Yan (Eds.), *Proceedings of the 18th international conference on pattern recognition (ICPR 2006)* (pp. 119–122). Hong Kong, China, August 20–24, 2006. Los Alamitos: IEEE Comput. Soc.

Witten, I., & Frank, E. (2005). *Data mining: practical machine learning tools and techniques* (2nd ed.). San Mateo: Morgan Kaufmann.

Wu, S., Flach, P., & Ferri, C. (2007). An improved model selection heuristic for AUC. In J. Kok, J. Koronacki, R. L. de Mántaras, S. Matwin, D. Mladenic, & A. Skowron (Eds.), *Proceedings of the 18th European conference on machine learning (ECML 2007)* (pp. 478–489). Warsaw, Poland, September 17–21, 2007. Berlin: Springer.

Yan, L., Dodier, R., Mozer, M., & Wolniewicz, R. (2003). Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic. In T. Fawcett & N. Mishra (Eds.), *Proceedings of the 20th international conference on machine learning (ICML 2003)* (pp. 848–855). Washington, DC, USA, August 21–24, 2003. Menlo Park: AAAI Press.