Approximate Reduction from AUC Maximization to 1-norm Soft Margin Optimization

Suehiro, Daiki
Department of Informatics, Kyushu University

Hatano, Kohei
Department of Informatics, Kyushu University

Takimoto, Eiji
Department of Informatics, Kyushu University

出版情報：Algorithmic learning theory : 22nd International Conference, ALT 2011, Espoo, Finland, October 5-7, 2011 : proceedings, pp.324-337, 2011. Springer Berlin Heidelberg
バージョン：
権利関係：

Approximate Reduction from AUC Maximization to 1-norm Soft Margin Optimization

Daiki Suehiro, Kohei Hatano, and Eiji Takimoto

Department of Informatics, Kyushu University
{daiki.suehiro, hatano, eiji}@inf.kyushu-u.ac.jp

Abstract. Finding linear classifiers that maximize AUC scores is important in ranking research. This is naturally formulated as a 1-norm hard/soft margin optimization problem over pn pairs of p positive and n negative instances. However, directly solving the optimization problems is impractical since the problem size (pn) is quadratically larger than the given sample size ($p + n$). In this paper, we give (approximate) reductions from the problems to hard/soft margin optimization problems of linear size. First, for the hard margin case, we show that the problem is reduced to a hard margin optimization problem over $p + n$ instances in which the bias constant term is to be optimized. Then, for the soft margin case, we show that the problem is approximately reduced to a soft margin optimization problem over $p+n$ instances for which the resulting linear classifier is guaranteed to have a certain margin over pairs.

1 Introduction

Learning to rank has been one of the most active areas of research in machine learning and information retrieval in the past decade, due to increasing demands in, for example, recommendation tasks and financial risk analysis [5, 13, 8, 4, 21, 6, 19, 2, 14]. Among the problems related to learning to rank, the bipartite ranking is a fundamental problem, which involves learning to obtain rankings over positive and negative instances. More precisely, for a given sample consisting of positive and negative instances, the goal of the bipartite ranking problem is to find a real-valued function h, which is referred to as a ranking function, with the following property: For a randomly chosen test pair of positive instance $\boldsymbol{x}^+$ and negative instance $\boldsymbol{x}^-$, the ranking function h maps $\boldsymbol{x}^+$ to a higher value than $\boldsymbol{x}^-$ with high probability. Thus, a natural measure for evaluating the goodness of ranking function h is the probability that $h(\boldsymbol{x}^+) > h(\boldsymbol{x}^-)$, which we call the AUC of h.

The bipartite ranking problem can be reduced to the binary classification problem over a new instance space, consisting of all pairs $(\boldsymbol{x}^+, \boldsymbol{x}^-)$ of positive and negative instances. More precisely, the problem of maximizing the AUC is equivalent to finding a binary classifier f of the form of $f(\boldsymbol{x}^+, \boldsymbol{x}^-) = h(\boldsymbol{x}^+) - h(\boldsymbol{x}^-)$ so that the probability that $f(\boldsymbol{x}^+, \boldsymbol{x}^-) > 0$ is maximized for a randomly

chosen instance pair. Several studies including RankSVMs [13, 4] have taken this approach with linear classifiers as the ranking functions. RankSVMs are justified by generalization bounds [21, 2] which say that a large margin over pairs of positive and negative instances in the sample implies a high AUC score under the standard assumption that instances are drawn i.i.d. under the underlying distribution.

The reduction approach, however, has a drawback that the sample constructed through the reduction is of size pn when the original sample consists of p positive and n negative instances. This is a quadratic blowup in size.

In this paper, we formulate AUC maximization as 1-norm hard/soft margin optimization problems[1] over pn pairs of p positive and n negative instances. We show some reduction schemes to 1-norm hard (or soft) margin optimization over $p+n$ instances which approximate the original problem over pairs. First, for the hard margin case where the resulting linear classifer is supposed to classfiy all pairs correctly by some positive margin, we show that the original problem over pairs is equivalent to the 1-norm hard margin problem over $p+n$ instances with the bias term.

Second, for the soft margin case, in which the resulting classsfier is allowed to misclassify a number of pairs, we show reduction methods to 1-norm soft margin optimization over instances that are guaranteed to have a certain margin over pairs of instance. When we solve the original problem over pairs, it can be shown that for any ε s.t. $0 < \varepsilon < 1$, the solution has a margin of least $\rho^* \geq \gamma^*$ over at least $(1-\varepsilon)pn$ pairs, where ρ^* and γ^* are optimal solutions of the primal and dual problems of the original problem. Note that the optimal solutions ρ^* and γ^* depend on ε respectively. On the other hand, for an appropriate parameter setting, one of our reduction methods guarantees that the resulting classifier has a margof at least γ^* for $(1-\sqrt{\varepsilon})^2 pn$ pairs. Note that, this guarantee might be rather weak, since the guaranteed margin γ^* is lower than the optimal margin ρ^* in general. However, if $\rho^* \approx \gamma^*$, say, when pairs are close to be linearly separable, our theoretical guarantee becomes sharper. Also, theoretically guaranteed reduction methods from AUC maximization to classification are quite meaningful since typical methods lack such properties.

We should note that our theoretical guarantee itself is not new. SoftRankBoost [15] is proved to have the same guarantee. But our reduction methods and SoftRankBoost are totally different. SoftRankBoost is designed using the smooth boosting framework [7, 23, 11, 12, 3]. On the other hand, our methods are built from an optimization theoretic perspective and provide a much clearer understanding for underlying optimization problems. In addition, our methods motivate practical heuristics to further improve AUCs.

In experiments using artificial and real data, the practical heuristics derived from the analysis achieve AUCs that are almost as high as the original soft

[1] In this paper we refer to 1-norm soft margin optimization as a soft margin optimization with 1-norm of the weight vector regularized. Note that sometimes the soft margin optimization of SVMs with 1-norm of slack valuables optimized is also called 1-norm soft margin optimization.

margin formulation over pairs while keeping the sample size linear. In addition, our methods also outperform previous methods including RankBoost [8] and SoftRankBoost.

There have been a number of studies in this field. Brefeld and Scheffer [4] and Fung et al. [10] proposed reduction methods from RankSVMs or 2-norm soft margin optimization over pairs to 2-norm soft margin optimization over instances. Raykar et al. investigated similar problems in the logisitic regression framework [18]. These reduction methods, however, do not have theoretical guarantees similar to ours. Further, these researches consider soft margin optimization problems where 2-norm of the weight vector is regularized. On the other hand, in our soft margin optimization, 1-norm of the weight vector is regularized. So, the resulting weight vector tends to be sparse, which is useful for feature selection. Freund et al proposed RankBoost [8], which is an efficient implementation of AdaBoost [9] over pairs of positive and negative instances and runs in linear time for a given sample size. Rudin and Schapire further demonstrated that under certain assumptions, AdaBoost is equivalent to RankBoost [21]. Since AdaBoost is shown to have at least half of the maximum margin asymptotically for the 1 norm hard margin optimization(see, e.g., [16, 17]), RankBoost and AdaBoost also have large margins over pairs. Rudin also proposed the P-Norm Push, which maximizes a criterion that assigns higher weights to rankings among top instances [20].

2 Preliminaries

Let $\mathcal{X}^+$ and $\mathcal{X}^-$ be the sets of positive instances and negative instances, respectively. Let $\mathcal{X} = \mathcal{X}^+ \cup \mathcal{X}^-$ be the instance space. A distribution D over $\mathcal{X}$ is said to be nontrivial if D has non-zero probability over both positive and negative instances. Given a non-trivial distribution D, we denote D^+ and D^- as the marginal distribution of D over positive and negative instances, respectively. A ranking function h is any function from $\mathcal{X}$ to $[-1, +1]$. The AUC of hypothesis h with respect to a non-trivial distribution D over $\mathcal{X}$ is given as

$$AUC_D(h) = \Pr_{\boldsymbol{x}^+, \boldsymbol{x}^- \sim D}\{h(\boldsymbol{x}^+) > h(\boldsymbol{x}^-) \mid \boldsymbol{x}^+ \in \mathcal{X}^+, \boldsymbol{x}^- \in \mathcal{X}^-\},$$

where each $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$ is drawn independently with respect to D.

Let S be a set of $m(= p + n)$ instances drawn i.i.d. with respect to D, which includes p positive instances and n negative instances, respectively. Let $S^+ = \{\boldsymbol{x}_1^+, \ldots, \boldsymbol{x}_p^+\}$ and $S^- = \{\boldsymbol{x}_1^-, \ldots, \boldsymbol{x}_n^-\}$, be the subsets of positive and negative instances respectively.

Given $\rho > 0$, we define

$$AUC_{S,\rho}(h) = \frac{\sum_{i=1}^{p} \sum_{j=1}^{n} I(h(\boldsymbol{x}_i^+) - h(\boldsymbol{x}_j^-) \geq \rho)}{pn},$$

where $I(\cdot)$ is the indicator function. The following theorem was presented by Rudin and Schapire.

Theorem 1 (Rudin and Schapire [21]) Let $\mathcal{F}$ be a set of ranking functions. Then, for any $\varepsilon > 0$, $\rho > 0$, for any $h \in \mathcal{F}$, the following holds

$$AUC_D(h) \geq AUC_{S,\rho}(h) - \varepsilon \tag{1}$$

with a probability of at least $1 - 2\mathcal{N}\left(\mathcal{F}, \frac{\rho}{4}\right) \exp\left\{-\frac{m\varepsilon^2 E^2}{8}\right\}$, where E is the expectation of $I(\boldsymbol{x}_i^+ \in \mathcal{X}^+, \boldsymbol{x}_j^- \in \mathcal{X}^-)$ when $\boldsymbol{x}_i^+$ and $\boldsymbol{x}_j^-$ are drawn independently from D, and $\mathcal{N}(\mathcal{F}, \varepsilon)$ is the covering number of $\mathcal{F}$, which is defined as the minimum number of balls of radius ε needed to cover $\mathcal{F}$ using L_∞ norm.

Here, note that the covering number is smaller if ρ is larger. So, a robust approach to learn a hypothesis with high AUC is to enlarge $AUC_{S,\rho}(h)$ for some large ρ.

2.1 1-norm soft margin over pairs of positive and negative instances

In this paper, we assume a finite set $\mathcal{H} = \{h_1, h_2, \ldots, h_N\}$ of ranking functions, which are functions from $\mathcal{X}$ to $[-1, +1]$. Our hypothesis class $\mathcal{F}$ is the set of convex combination of ranking functions in $\mathcal{H}$, i.e.,

$$\mathcal{F} = \left\{ f \;\middle|\; f(\boldsymbol{x}) = \sum_{k=1}^{N} \alpha_k h_k(\boldsymbol{x}), h_k \in \mathcal{H},\ \sum_{k=1}^{N} \alpha_k = 1, \alpha_k \geq 0 \right\}.$$

Now, our goal is to find a linear combination of ranking functions $f \in \mathcal{F}$ that has a large margin ρ over pairs of instances in S^+ and S^-.

More formally, we formulate our problem as optimizing the soft margin over pairs of positive and negative instances. For convenience, for any $q \geq 1$, let $\mathcal{P}_q$ be the q-dimensional probability simplex, i.e., $\mathcal{P}_q = \{\boldsymbol{p} \in [0,1]^q \mid \sum_i p_i = 1\}$. Then, for positive and negative sets of instances S^+ and S^-, the set $\mathcal{H}$ of ranking functions, and any fixed $\nu \in \{1, \ldots, pn\}$, the 1-norm soft margin optimization problem is given as follows:

$$\begin{aligned}
(\rho^*, \boldsymbol{\alpha}^*, \boldsymbol{\xi}^*) = \max_{\rho, \boldsymbol{\alpha}, \boldsymbol{\xi}} \ & \rho - \frac{1}{\nu} \sum_{i=1}^{p} \sum_{j=1}^{n} \xi_{ij} \\
\text{sub.to} \ & \\
& \sum_k \alpha_k (h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}_j^-))/2 \geq \rho - \xi_{ij} \ (i = 1, \ldots, p, j = 1, \ldots, n), \\
& \boldsymbol{\alpha} \in \mathcal{P}_N, \\
& \xi_{ij} \geq 0 \ (i = 1, \ldots, p, j = 1, \ldots, n).
\end{aligned} \tag{2}$$

In this problem, the goal is to maximize the margin ρ of the linear combination $\boldsymbol{\alpha}$ of ranking functions w.r.t. instances as well as to minimize the sum of "losses" ξ_{ij}, the quantity by which the target margin ρ is violated. Here $\nu \in \{1, \ldots, pn\}$ controls the tradeoff between the two objectives.

Then, using Lagrangian multipliers, the dual problem is given as

$$
\begin{aligned}
(\gamma^*, \boldsymbol{d}^*) = & \min_{\gamma, \boldsymbol{d}} \gamma \\
& \text{sub.to} \\
& \sum_{i,j} d_{ij}(h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}_j^-))/2 \leq \gamma \quad (k = 1, \ldots, N), \\
& 0 \leq d_{ij} \leq \frac{1}{\nu} \ (i = 1, \ldots, p, j = 1, \ldots, n), \\
& \boldsymbol{d} \in \mathcal{P}_{pn}.
\end{aligned} \tag{3}
$$

Since the problem is a linear program, by duality, we have $\rho^* - \frac{1}{\nu}\sum_{i,j}\xi_{ij}^* = \gamma^*$.

Furthermore, by using KKT conditions, it can be shown that (see, e.g., [22, 24]), the optimal solution guarantees the number of pairs $(\boldsymbol{x}_i^+, \boldsymbol{x}_j^-)$ for which $\sum_k \alpha_k(h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}_j^-))/2 \leq \rho^*$ is at most ν. In other words, setting $f = \sum_{k=1}\alpha_k h_k$, we have that $AUC_{S,\rho^*}(f)$ is at least $1 - \nu/pn$. Thus, solving 1-norm soft margin optimization pairs is a quite natural approach for improving the lower bound of $AUC_D(f)$.

3 1-norm hard margin optimization over pairs

In this section, we show the equivalence between two hard margin optimization problems, the 1-norm hard margin problem over pairs and the 1-norm hard margin problem with bias. The hard margin optimization problem is a special case of the soft margin problem in that the resulting classifier or ranking function is supposed to predict all the instances or pairs correctly with a positive margin.

The first problem we consider is the 1-norm hard margin optimization over pairs of positive and negative instances.

$$
\begin{aligned}
& \max_{\rho, \boldsymbol{\alpha} \in \mathcal{P}_N} \rho \\
& \text{sub.to} \\
& \sum_{k=1}^{N} \alpha_k(h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}_j^-))/2 \geq \rho \ (i = 1, \ldots, p, j = 1, \ldots, n).
\end{aligned} \tag{4}
$$

The second hard margin problem is the 1-norm hard margin optimization with bias.

$$\max_{\rho,\boldsymbol{\alpha}\in\mathcal{P}_N,b} \rho \tag{5}$$
$$\text{sub.to}$$
$$\sum_{k=1}^{N} \alpha_k h_k(\boldsymbol{x}_i^+) + b \geq \rho \ (i = 1, \ldots, p),$$
$$\sum_{k=1}^{N} \alpha_k h_k(\boldsymbol{x}_j^-)) + b \leq -\rho \ (j = 1, \ldots, n).$$

In the following, we show that both of these problems are equivalent, in the sense that we can construct an optimal solution of one problem from an optimal solution of the other problem.

Theorem 2 Let $(\rho_b, \boldsymbol{\alpha}_b, b_b)$ be an optimal solution of the 1-norm hard margin optimization with bias (5). Then, $(\rho_b, \boldsymbol{\alpha}_b)$ is also an optimal solution of the 1-norm hard margin optimization over pairs (4).

Proof. Let $(\rho_p, \boldsymbol{\alpha}_p)$ be an optimal solution of the 1-norm hard margin optimization over pairs. Clearly, $(\rho_b, \boldsymbol{\alpha}_b, b_b)$ is a feasible solution of the 1-norm hard margin optimization over pairs. So, $\rho_b \leq \rho_p$. Next, we show that the opposite is true. Let $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$ be positive and negative examples for which the margin of $\boldsymbol{\alpha}_p$ is minimized. Note that for the pair $(\boldsymbol{x}^+, \boldsymbol{x}^-)$ the constraint holds with equality. Let

$$b_p = -\frac{\sum_k \alpha_{p,k}(h_k(\boldsymbol{x}^+) + h_k(\boldsymbol{x}^-))}{2}.$$

Then, $(\rho_p, \boldsymbol{\alpha}_p, b_p)$ is a feasible solution of the 1-norm hard margin optimization with bias. For any positive instance $\boldsymbol{x}_i^+$, observe that

$$\begin{aligned}
\sum_{k=1}^{N} \alpha_{p,k} h_k(\boldsymbol{x}_i^+) + b_p &= \sum_{k=1}^{N} \alpha_{p,k} \frac{h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}^-)}{2} + \sum_{k=1}^{N} \alpha_{p,k} \frac{h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}^+)}{2} \\
&\geq \rho_p + \sum_{k=1}^{N} \alpha_{p,k} \frac{h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}^-) - (h_k(\boldsymbol{x}^+) - h_k(\boldsymbol{x}^-))}{2} \\
&\geq \rho_p + \rho_p - \rho_p = \rho_p.
\end{aligned}$$

A similar inequality holds for negative instances as well. Thus, we have $\rho_p \leq \rho_b$. □

4 Reduction methods from 1-norm soft margin optimization over pairs

In this section, we propose reduction methods from the 1-norm soft margin optimization over pairs to that over instances.

4.1 Our method

We would like to approximate the dual problem of the 1-norm soft margin optimization over pairs (3). The dual problem is concerned with finding a distribution over pn pairs of positive and negative instances satisfying the linear constraints. Our key idea is to replace the distribution d_{ij} with a product distribution $d_i^+ d_j^-$, where $\boldsymbol{d}^+$, $\boldsymbol{d}^-$ are distributions over positive and negative instances, respectively.

Letting $d_{ij} = d_i^+ d_j^-$, observe that

$$\begin{aligned}\sum_{i,j} d_{ij} \frac{h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}_j^-)}{2} &= \sum_{i,j} d_i^+ d_j^- \frac{h_k(\boldsymbol{x}_i^+) - h_k(\boldsymbol{x}_j^-)}{2} \\ &= \frac{\sum_i d_i^+ h_k(\boldsymbol{x}_i^+) \sum_j d_j^-}{2} - \frac{\sum_j d_j^- h_k(\boldsymbol{x}_j^-) \sum_i d_i^+}{2} \\ &= \sum_i d_i^+ h_k(\boldsymbol{x}_i^+)/2 - \sum_j d_j^- h_k(\boldsymbol{x}_j^-)/2.\end{aligned}$$

Then, we obtain the following problem.

$$\begin{aligned}&\min_{\boldsymbol{d},\gamma} \gamma \\ &\text{sub.to} \\ &\sum_{i=1}^{p} d_i^+ h_k(\boldsymbol{x}_i^+)/2 - \sum_{j=1}^{n} d_j^- h_k(\boldsymbol{x}_j^-)/2 \le \gamma \quad (k = 1, \ldots, N), \\ &\boldsymbol{d}^+ \in \mathcal{P}_p, \boldsymbol{d}^- \in \mathcal{P}_n, \\ &0 \le d_i^+ d_j^- \le \frac{1}{\nu} \ (i = 1, \ldots, p, j = 1, \ldots, n).\end{aligned} \tag{6}$$

Since we restrict distributions to be products of two distributions, the optimal solution yields a feasible solution of the original problem (2). This problem has $p + n + 1$ variables, whereas the original problem has $pn + 1$ variables. So this problem would be easier to solve. But, unfortunately, this problem is not convex since the constraints $d_i^+ d_j^- \le 1/\nu$ $(i = 1, \ldots, p, j = 1, \ldots, n)$ are not convex.

Later herein, we propose a method by which to find a local minimum of this non-convex problem (6). First, however, we show a restricted the problem, the solution of which has a certain amount of margin over pairs. In order to avoid non-convex constraints, we fix ν^+ and ν^- such that $\nu = \nu^+ \nu^-$ and enforce $d_i^+ \le 1/\nu^+$ and $d_j^- \le 1/\nu^-$. Equivalently, we fix $\nu^- = \nu^+/\nu$. As a result, we obtain the following problem.

$$
\begin{aligned}
\hat{\gamma}(\nu^+) = & \min_{\boldsymbol{d}^+, \boldsymbol{d}^-, \gamma} \gamma \\
& \text{sub.to} \\
& \sum_i d_i^+ h_k(\boldsymbol{x}_i^+)/2 - \sum_j d_j^- h_k(\boldsymbol{x}_j^-)/2 \leq \gamma \quad (k = 1, \ldots, N), \\
& \boldsymbol{d}^+ \in \mathcal{P}_p, \boldsymbol{d}^- \in \mathcal{P}_n, \\
& d_i^+ \leq 1/\nu^+, \\
& d_j^- \leq 1/\nu^- = \nu^+/\nu.
\end{aligned} \tag{7}
$$

Note that if we optimize ν^+, we obtain the minimum of problem (6), that is, $\min_{\nu^+} \hat{\gamma}(\nu^+) = \gamma^*$. Remember, however, that problem (6) is not convex w.r.t. ν^+ (see Fig. 1 for an example). Therefore, it is not straightforward to obtain the optimum.

On the other hand, for any fixed choice of ν^+ and ν^-, we can guarantee that the solution of problem (7) has a certain margin for several pairs.

Theorem 3 *Given ν^+ and ν^-, the solution of problem (7) has a margin of at least γ^* for at least $pn - \nu^+ n - \nu^- p + \nu^+ \nu^-$ pairs.*

Proof. Using Lagrangian multipliers, it can be shown that the dual problem of (7) is as follows:

$$
\begin{aligned}
(\hat{\rho}, \hat{\boldsymbol{\alpha}}, \hat{b}, \hat{\boldsymbol{\xi}}^+, \hat{\boldsymbol{\xi}}^-) = & \arg \max_{\boldsymbol{\alpha} \in \mathcal{P}_N, b, \boldsymbol{\xi}^+, \boldsymbol{\xi}^-} \rho - \frac{1}{2\nu^+} \sum_{i=1}^{p} \xi_i^+ - \frac{1}{2\nu^-} \sum_{j=1}^{n} \xi_j^- \\
& \text{sub.to} \\
& \sum_{k=1}^{N} \alpha_k (h_k(\boldsymbol{x}_i^+) + b \geq \rho - \xi_i^+ \ (i = 1, \ldots, p), \\
& -\sum_{k=1}^{N} \alpha_k h_k(\boldsymbol{x}_j^-) - b \geq \rho - \xi_j^- \ (j = 1, \ldots, n), \\
& \boldsymbol{\xi}^+, \boldsymbol{\xi}^- \geq \mathbf{0}.
\end{aligned} \tag{8}
$$

By using the KKT conditions, $\hat{\xi}_i^+ (\hat{d}_i^+ - 1/\nu^+) = 0$. Therefore, if $\hat{\xi}_i^+ > 0$ then $\hat{d}_i^+ = 1/\nu^+$. Similarly, if $\hat{\xi}_j^- > 0$ then $\hat{d}_j^- = 1/\nu^-$. Note that there are at most ν^+ instances such that $\hat{d}_i^+ = 1/\nu^+$. This implies that there are at most ν^+ instances whose corresponding $\hat{\xi}_i^+ > 0$. Again, similarly, there are at most ν^- instances with $\hat{\xi}_j^- > 0$. There are therefore, for at least $(p - \nu^+)(n - \nu^-)$ pairs, the margin of which is at least $\hat{\rho}$. Finally, by duality, $\hat{\rho} - (1/\nu^+) \sum_i \hat{\xi}_i^+ - (1/\nu^-) \sum_j \hat{\xi}_j^- = \hat{\gamma}$. Combined with the fact that $\hat{\gamma} \geq \gamma^*$, we have $\hat{\rho} \geq \gamma^*$, which completes the proof. □

We note that problem (8) in the proof is the primal form of the dual problem (7). In particular, for the choice that $\nu = \varepsilon pn$, $\nu^+ = \sqrt{\varepsilon}p$ and $\nu^- = \sqrt{\varepsilon}n$, we obtain the following corollary.

Corollary 4 For $\nu = \varepsilon pn$, $\nu^+ = \sqrt{\varepsilon}p$ and $\nu^- = \sqrt{\varepsilon}n$, a solution of problem (7) has a margin of at least γ^* for $(1-\sqrt{\varepsilon})^2 pn$ pairs.

Here, the lower bound ν/n of ν^+ is given so that the upper bound of d_j^- is at least $1/n$. Note that, the area under the tangent line of $d_i^+ = 1/\nu^+$ at $\nu^* = \nu_c^+$ is always included in the area $d_i^+ \leq 1/\nu^+$. Thus, any feasible solution of problem (9) is also a feasible solution of problem (6).

4.2 Practical Heuristics

Now we propose a practical method to find a local minimum of problem (6). Recall that in problem (6), we have non-convex constraints $d_i^+ \leq 1/\nu^+$ when we regard ν^+ as a variable. In order to avoid non-convex constraints, we consider a tangent line of $1/\nu^+$ at some point $\nu^+ = \nu_c^+$. More precisely, we consider the following problem.

$$
\begin{aligned}
(\tilde{\gamma}, \tilde{\boldsymbol{d}}^+, \tilde{\boldsymbol{d}}^-, \tilde{\nu}^+) = \arg \min_{\gamma, \boldsymbol{d}^+, \boldsymbol{d}^-, \nu^+} \ & \gamma \qquad (9)\\
\text{sub.to} \ & \\
& \sum_{i=1}^{p} d_i^+ h_k(\boldsymbol{x}_i^+)/2 - \sum_{j=1}^{n} d_j^- h_k(\boldsymbol{x}_j^-)/2 \leq \gamma \ (k = 1, \ldots, N),\\
& \boldsymbol{d}^+ \in \mathcal{P}_p, \boldsymbol{d}^- \in \mathcal{P}_n,\\
& d_i^+ \leq -\frac{1}{(\nu_c^+)^2}\nu^+ + \frac{2}{\nu_c^+} \quad (i = 1, \ldots, p),\\
& d_j^- \leq \frac{\nu^+}{\nu} \quad (j = 1, \ldots, n),\\
& \frac{\nu}{n} \leq \nu^+ \leq -\frac{(\nu_c^+)^2}{p} + 2\nu_c^+.
\end{aligned}
$$

Here the lower bound ν/n of ν^+ is added so that the upper bound of d_j^- is at least $1/n$. Also, the upper bound of ν^+ is given so that the upper bound of d_i^+ is at least $1/p$. Note that, the region under the tangent line of $d_i^+ = 1/\nu^+$ at $\nu^* = \nu_c^+$ is always contained in the region $d_i^+ \leq 1/\nu^+$. Thus, any feasible solution of problem (9) is also a feasible solution of problem (6).

Now we are ready to describe our heuristics:

1. Given some ν_c^+, solve problem (9) and get a solution $(\tilde{\gamma}, \tilde{\boldsymbol{d}}^+, \tilde{\boldsymbol{d}}^-, \tilde{\nu}^+)$.
2. Given $\nu^+ = \tilde{\nu}^+$, solve problem (7) and get a solution $(\hat{\gamma}, \hat{\boldsymbol{d}}^+, \hat{\boldsymbol{d}}^-)$.

Observe that the solution $(\tilde{\gamma}, \tilde{\boldsymbol{d}}^+, \tilde{\boldsymbol{d}}^-)$ of problem (9) is a feasible solution of problem (7) given $\nu^+ = \tilde{\nu}^+$. Thus, we have $\hat{\gamma} \leq \tilde{\gamma}$. Furthermore, if we set

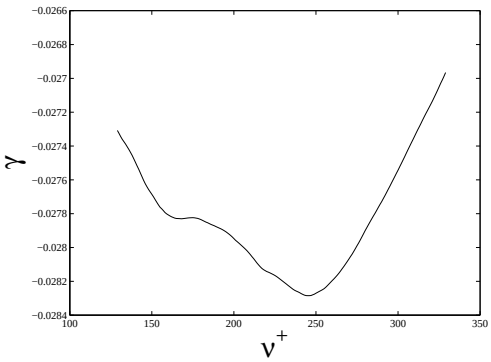

Fig. 1. Illustration of the function $\hat{\gamma}(\nu^+)$ for an artificial data set.

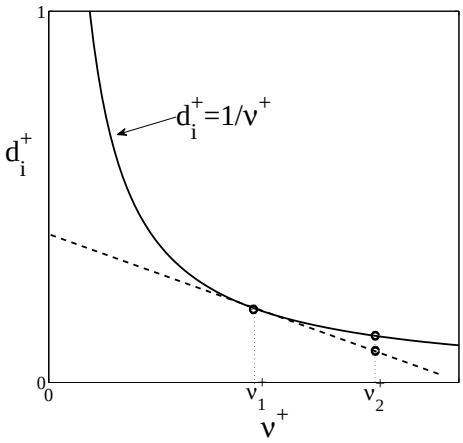

Fig. 2. Illustration of the heuristics. Here $\nu_1^+ = \nu_c^+$ and ν_2^+ is the solution of problem (9) given ν_c^+.

$\nu_c^+ = \hat{\nu}^+$, the solution $(\hat{\gamma}, \hat{\boldsymbol{d}}^+, \hat{\boldsymbol{d}}^-, \hat{\nu}^+)$ is a feasible solution of problem (9), so that the minimum $\tilde{\gamma}'$ of problem (9) satisfies $\tilde{\gamma}' \leq \hat{\gamma}$. Therefore, by repeating this procedure, we can obtain a monotonically decreasing sequence of γ, which will converge to a local minimum of problem (6). In an algorithmic perspective, the second step that solves problem (7) seems redundant. However, we add the second step for numerical stability since problem (7) has simpler constraints. Fig. 2 illustrates the heuristics.

5 Experiments

In this section, we present preliminary experimental results. The data sets include artificial data sets, and real data sets from the UCI Machine Learning Repository and Reuters.

5.1 Artificial Data

For the first experiment, we used artificial data sets with r-of-k threshold functions as target functions. An r-of-k threshold function f over N Boolean variables is associated with some set A of k Boolean variables and f outputs $+1$ if at least r of the k variables in A are positive and f outputs -1, otherwise. Assume that the instance space is $\{+1, -1\}^N$. In other words, the r-of-k threshold function f is represented as follows

$$f(\boldsymbol{x}) = \mathrm{sign}(\sum_{x \in A} x + k - 2r + 1).$$

For $N = 100$, $k = 30$, and $r = 1, 8, 15$, we fix r-of-k threshold functions which determine labels. Then for each set of parameters, we generate $m = 1000$ random instances so that ratios of positive and negative instances are $5 : 5$, $7 : 3$, and $9 : 1$ respectively. Finally, we add random noise into labels by changing the label of each instance with probabilities of 5%, 10%, and 15%. As hypotheses, we use N Boolean variables themselves and the constant hypothesis which always outputs $+1$.

We compare RankBoost [8], SoftRankBoost [15], 1-norm soft margin over pairs (LP-Pair), and our method. For RankBoost, we set the number of iterations to be $T = 500$,5000, and 10000, respectively. For the other methods, we set the parameter $\nu = \varepsilon pn$, where $\varepsilon \in \{0.05, 0.1, 0.15, 0.2, 0.25, 0.3\}$. We evaluate each method by 5-fold cross validation. As shown in Table 1, our method recorded higher AUCs than the other algorithm for almost the data sets. In addition, in 2, our method achieves especially high AUCs, which are greater than or equal to those of LP-Pair.

5.2 UCI Data

For the next experiment, we use data sets "hypothyroid", "ionosphere", "kr-vs-kp", "sick-euthroid", "spambase" from the UCI Machine Learning Repository[1]. The parameters of each algorithm are the same as in Section 5.1. As shown in Table 3, our method archives high AUCs for all data sets.

5.3 Reuters Data

Reuters data sets are data of Reuters news (Reuters-21758[2]), which are 10710 articles labeld by topics. We choose 5 major topics and consider 5 binary classification problems whose objective is to classify if a given article blongs to the

[2] http://www.daviddlewis.com/resources/testcollections/reuters21578

Table 1. AUCs for artificial data sets.

data		RankBoost			SoftRankBoost	LP-Pair	our method
r	niose	500	5000	10000			
1	5(%)	0.9313	0.9384	0.9378	0.7275	0.9745	**0.9818**
8		0.9251	0.9239	0.9239	0.9325	0.9564	**0.9596**
15		0.9264	0.9262	0.9262	0.9401	0.952	**0.9545**
1	10(%)	0.8777	0.8979	0.8979	0.7391	0.9125	**0.994**
8		0.8857	0.8853	0.8853	0.9043	0.9136	**0.9173**
15		0.8727	0.8727	0.8727	0.869	0.9043	**0.9007**
1	15(%)	0.8102	0.8389	0.8391	0.7442	0.8322	**1.0**
8		0.8371	0.8372	0.8372	**0.8793**	0.8608	0.8643
15		0.8377	0.8337	0.8337	0.856	**0.857**	0.8525

Table 2. AUCs for artificial data sets with random noises 5%, 10%, and 15%.

data		RankBoost			SoftRankBoost	LP-Pair	our method
$p:n$	r	500	5000	10000			
7:3	1	0.9177	0.9182	0.9179	0.7661	0.9472	**0.9624**
	8	0.9018	0.9015	0.9015	**0.9318**	0.9292	0.9308
	15	0.8959	0.8956	0.8956	**0.9353**	0.9294	0.9271
9:1	1	0.7695	0.7742	0.7738	0.7735	0.7924	**0.9431**
	8	0.7736	0.7736	0.7736	0.7718	**0.7818**	0.7648
	15	0.7247	0.7247	0.7247	**0.8266**	0.7426	0.7320

Table 3. AUCs for UCI data sets, when N, p, and n stand for the dimension, the number of positive and negative instances of each data sets, respectively.

data				RankBoost			SoftRankBoost	LP-Pair	our method
	N	p	n	1000	5000	10000			
hypothyroid	43	151	3012	0.9488	0.9468	0.9468	0.96	0.9511	**1.0**
ionosphere	34	225	126	0.9327	0.9253	0.9253	**0.9917**	0.9768	0.9865
kr-vs-kp	73	1669	1527	0.8712	0.8721	0.8721	0.9085	**1.0**	0.9276
sick-euthroid	43	293	2870	0.7727	0.8706	0.8706	0.7847	**1.0**	**1.0**
spambase	57	1813	2788	0.8721	0.7735	0.735	0.9359	**1.0**	**1.0**

topic. We prepare 30838 base classifiers which are decision stumps associated with words. More precisely, each base classifier anwers 1 if the given article contains the associated word and answers 0, otherwise. The results are summarized in Table 4. For Reuters date sets, our method shows better performance than RankBoost, but SoftRankBoost shows better AUCs for some topics (3 out of 5).

Table 4. AUCs for Reuters data sets, where p, and n stand for the number of positive and negative instances included in each of data sets.

data			RankBoost			SoftRankBoost	our method
topics	p	n	1000	5000	10000		
acq	2327	8383	0.9296	0.9347	0.9347	0.9363	**0.9388**
crude	592	10118	0.9133	0.9188	0.9203	**0.9944**	0.9329
earn	3802	6908	0.9567	0.9568	0.9566	**0.9952**	0.9652
money-fx	743	9967	0.9375	0.9335	0.9318	**0.9608**	0.9479
trade	529	10181	0.9290	0.9301	0.9291	0.9281	**0.9450**

5.4 Computation Time

Finally, we examine the computation time of LP-Pair and our method. We use a machine with four Intel Xeon 5570 2.93-GHz cores and a memory of 32 GByte. We use the artificial data that are used in Section 5.1, $N = 100$, $k = 10$, $r = 3$. The sizes of the data sets are $m = 100, 500, 1000, 1500$, respectively. The ratio of positive and negative instances is $5 : 5$, and we add random noise of 5%. We set $\varepsilon = 0.2$ for both LP-Pair and our method and evaluate each execution time by 5-fold cross validation. As is shown in Table 5, clearly our method is clearly faster than LP-Pair.

6 Conclusion and Future Work

In this paper, we have formulated AUC maximization as hard/soft margin optimization problems over pairs of positive and negative instances. In the hard

Table 5. Computation time(sec.).

m	LP-Pair	our method
100	0.102	0.11
500	24.51	0.514
1000	256.78	0.86
1500	1353	1.76

margin case, we showed that the original problem over pairs is equivalent to the 1-norm soft margin problem over $p + n$ instances with the bias term. In the soft margin case, we proposed a reduction method for the 1-norm soft margin optimization problem over instances, which is generally non-convex. Our reduction method is guaranteed to obtain a certain amount of margin over pairs of instances. Moreover, we have proposed heuristics that obtains more appropriate parameters. We have tested this method for artificial and real data. In comparison with other methods, our method achieved high AUCs in the experiments.

In the future, we intend to examine our methods for additional data sets including very large data sets. In addition, we would like to investigate why our method and SoftRankBoost sometimes achieve higher AUCs than the 1-norm soft margin over pairs.

7 Acknowledgements

This work was conducted in part, while the second author was visiting UC Santa Cruz. The authors would like to Manfred Warmuth for the stimulating discussion regarding the initial results. We would also like to Masayuki Takeda for his financial support. Finally, we would like to anonymous reviewers for helpful comments. This work was partly supported by JSPS KAKENHI 23700178 and KAKENHI (B) 23300003.

References

1. A. Asuncion and D. J. Newman. UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences, `http://mlearn.ics.uci.edu/MLRepository.html`, 2007.
2. N. Balcan, N. Bansal, A. Beygelzimer, D. Coppersmith, J. Langford, and G. B. Sorkin. Robust reductions from ranking to classification. In *Proceedings of the 20th Annual Conference on Learning Theory (COLT 2007)*, pages 604–619, 2007.
3. J. K. Bradley and R. Shapire. Filterboost: Regression and classification on large datasets. In *Advances in Neural Information Processing Systems 20*, pages 185–192, 2008.
4. U. Brefeld and T. Scheffer. Auc maximizing support vector learning. In *Proceedings of the ICML Workshop on ROC Analysis in Machine Learning*, 2005.
5. W. W. Cohen, R. E. Schapire, and Y. Singer. Learning to order things. *Journal of Artificial Intelliegence Research*, 10:243–279, 1999.
6. C. Cortes and M. Mohri. AUC optimization vs. error rate minimization. In *Adances in Neural Information Processing Systems 16*, 2004.
7. C. Domingo and O. Watanabe. MadaBoost: A modification of AdaBoost. In *Proceedings of 13th Annual Conference on Computational Learning Theory (COLT 2000)*, pages 180–189, 2000.
8. Y. Freund, R. Iyer, R. E. Shapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.
9. Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

10. G. Fung, R. Rosales, and B. Krishnapuram. Learning rankings via convex hull separation. In *Advances in Neural Information Processing Systems 18 (NIPS 2005)*, 2005.
11. D. Gavinsky. Optimally-smooth adaptive boosting and application to agnostic learning. *Journal of Machine Learning Research*, 2003.
12. K. Hatano. Smooth boosting using an information-based criterion. In *Proceedings of the 17 th International Conference on Algorithmic Learning Theory (ALT 2006)*, pages 304–319, 2006.
13. T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002.
14. P. M. Long and R. A. Servedio. Boosting the area under the roc curve. In *Advances in Neural Information Processing Systems 20*, 2008.
15. J. Moribe, K. Hatano, E. Takimoto, and M. Takeda. Smooth boosting for margin-based ranking. In *Proceedings of the 19th International Conference on Algorithmic Learning Theory (ALT 2008)*, pages 227–239, 2008.
16. G. Rätsch. *Robust Boosting via Convex Optimization: Theory and Applications*. PhD thesis, University of Potsdam, 2001.
17. G. Rätsch and M. K. Warmuth. Efficient margin maximizing with boosting. *Journal of Machine Learning Research*, 6:2131–2152, 2005.
18. V. C. Raykar, R. Duraiswami, and B. Krishnapuram. A fast algorithm for learning a ranking function from large-scale data sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1158–1170, 2008.
19. C. Rudin. Ranking with a p-norm push. In *Proceedings of 19th Annual Conference on Learning Theory*, pages 589–604, 2006.
20. C. Rudin. The P-Norm Push: A simple convex ranking algorithm that concentrates at the top of the list. *Journal of Machine Learning Research*, 10:2233–2271, 2009.
21. C. Rudin and R. E. Schapire. Margin-based ranking and an equivalence between AdaBoost and RankBoost. *Journal of Machine Learning Research*, 10:2193–2232, 2009.
22. B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett. New support vector algorithms. *Neural Computation*, 12(5):1207–1245, 2000.
23. R. A. Servedio. Smooth boosting and learning with malicious noise. *Journal of Machine Learning Research*, 4:633–648, 2003.
24. M. Warmuth, K. Glocer, and G. Rätsch. Boosting algorithms for maximizing the soft margin. In *Advances in Neural Information Processing Systems 20*, pages 1585–1592, 2008.