Robust Reductions from Ranking to Classification

Anonymous Author(s)
Affiliation
Address
City, State/Province, Postal Code, Country
email

Abstract

We reduce ranking, as measured by "Area Under the Reciever Operating Characteristic Curve" (AUC), to Binary classification. The core theorem shows that a binary classification regret of r on a certain induced binary problem implies an AUC regret of at most $4r$. This is a large improvement over naive approaches such as ordering according to regressed scores which has a regret transform of $r \to nr$ where n is the number of elements.

1 Introduction

Problems. In the bipartite ranking problem, we are given a set of unlabeled instances belonging to two classes (0 and 1), and the goal is to rank all instances from class 0 before any instance from class 1. Area under the ROC curve (AUC) is a measure of successful ranking where the loss is greater for mistakes at the beginning or the end of an ordering. This satisfies the intuition that an unwanted item placed at the top of a recommendation list, has higher asssociated loss than when placed in the middle. A handy shorthand for understanding AUC, is that it is one minus the normalized bubble sort distance between the predicted ordering and the true ordering.

The classification problem is simply predicting whether a label is 0 or 1 with success measured according to the probability of a misprediction.

These problems appear quite different. For example, the classification loss function is defined on a per-example basis while AUC is defined for sets of examples.

A Fundamental Question. A natural fundamental question to ask is "Are these problems truly different?" If the answer is yes, it implies that we require fundamentally different algorithms to optimize these different loss functions. If the answer is "no", then we can reuse existing algorithms and techniques to optimize AUC. An answer of "no" also suggests that other ranking losses such as those used by search companies may be solved directly with reuse of existing technology.

Question Complexities. Several basic observations help define this problem.

1. The question can not be answered satisfactorily under any assumptions about the way the world produces data. Even a minimal IID sample assumption is not met by a number of real-world applications where AUC is important. This observation implies most forms of analysis are not suitable.

   Machine learning reductions analysis *can* meet this challenge. At a high level, learning reductions analysis bounds the realized AUC performance in terms of the realized classification performance. Since the analysis is relative, no assumptions about the real-world process need be made.

   A consequence of this observation is that our analysis must cope with arbitrary high order dependencies between the labels of examples.

2. A natural approach to solving ranking is to order examples according to some "score" or estimated conditional class probability. There does indeed exist a general reduction from conditional class probability estimation to (hard) 0/1 classification [LZ05]. But, as discussed in [LZ05] this provides no satisfying solution for AUC. The fundamental difficulty is exhibited by test sets with one "1" and many "0"s. For these datasets, a classification error on the "1" with perfect prediction for the "0"s may result in a dramatic AUC loss. This observation implies that all solutions which order according to some predicted score dependent on a single element have a regret transform from r binary regret to AUC regret of nr where n is the number of elements ranked.

   In addition, this observation necessitates conditioning on the ratio of "0"s to "1"s in the IID stability analysis of AUC [SHD05].

Our Result We show that a pairwise classifier with a regret of r on a certain induced problem implies a regret of at most $4r$ with respect to area under the ROC curve for *all* (nonIID!) distributions over elements. The theorem is a large improvement over the current state-of-the-art which has a dependence on nr. For comparison, this proves a functionally tighter relationship from ranking to binary classification than has been proven for regression to binary classification ($r \rightarrow \sqrt{r}$) [LZ05] or multiclass to binary classification ($r \rightarrow 4\sqrt{r}$) [LB05].

Organization

Section 2 formalizes the setting. Section 3 presents the regret analysis with respect to the AUC. Section 5 shows that a common NP-hard algorithm for combining preferences [CSS99] is actually suboptimal compared to the mechanism we use in section 3. This happy observation implies we can comfortably use a relatively tractable $O(n^2)$ algorithm for inducing the binary problem and combining binary preferences to produce an ordering.

Relation with Previous Work

There is diverse previous work on ranking.

Several papers have proved generalization [SHD05] or large deviation bounds [CLV05] for ranking. These results (essentially) analyze the learnability of ranking directly by estimating the rate of convergence of empirical estimates of ranking loss to expected ranking loss. In contrast, the reductions analysis here shows that good classification performance implies good ranking performance, opening the possibility or reusing existing methods.

Cortes and Mohri [CM04] give a statistical analysis of the relationship between the AUC and the 0/1 error rate *on the same classification problem*, treating the two as different loss functions. They give expressions for the expected value and the standard deviation of the AUC over all classifications with a fixed number of errors under the assumption that all such classifications are equiprobable.

Boosting approaches to ranking [FIS+03, RCM+05] combine "weak rankers" to produce an overall final ranking. These algorithms train based on pairs (as the observation above shows is necessary for robustness) to produce a scoring function which is used to order things. The results here have two implications: (1) we show how to turn any classification algorithm into a weak ranker of the required type (2) we suggest a new mechanism for rank boosting: reduce the problem to binary classification and apply adaboost.

In "Learning to Order Things" [CSS99], the authors use pairwise predictors on an explicit classifier set which are combined via an exponential weighting scheme with an online-learning guarantee: the resulting ranker is not much worse than the best ranker in the explicit classifier set. This paper discusses the use (or approximate use) of an NP-hard algorithm for combining pairwise preference information. Here, we show that a much more tractable algorithm works well. In fact, that the NP-hard algorithm may sometimes perform worse! See section 5 for details.

2 Formal Setup

We first define the problems we are considering and then their derived quantities.

Algorithm 1 AUC-TRAIN (labeled set $S \in X^n$, binary learning algorithm A)

1. Set $n^0 = |\{(x,y) \in S : y = 0\}|$.
2. Let
$$S' = \{\langle (x_1, x_2), I(y_1 > y_2)\rangle : (x_1, y_1), (x_2, y_2) \in S \text{ and } y_1 \neq y_2\}$$
3. return $c = A(S')$.

Algorithm 2 DEGREE (Unlabeled set Q, importance-weighted pairwise classifier c)

1. For $x \in Q$, let $\deg(x) = |\{x' : c(x, x') = 1, x' \in Q\}|$.
2. Sort Q in the descending order of $\deg(x)$, breaking ties randomly.

A *binary classification problem* is defined by a distribution B over $X \times \{0,1\}$, where X is a feature space and $\{0,1\}$ is the binary prediction space. The goal is to find a classifier $c : X \to \{0,1\}$ minimizing the *classification loss*,

$$e(c, B) = \mathbf{Pr}_{(x,y)\sim B}[c(x) \neq y].$$

Let $o : X \times X \to \{0,1\}$ be an ordering function that given as input any two instances in X outputs 1 if it agrees with the ordering of its arguments; otherwise it outputs 0. The *AUC loss* of an ordering o on a set $S \in (X \times \{0,1\})^*$ is defined as

$$l_{\text{AUC}}(o, S) = \frac{\sum_{i,j} I(y_i > y_j) o(x_i, x_j)}{\sum_{i<j} I(y_i \neq y_j)}.$$

(Indices i and j in the summations range from 1 to n, with $i \neq j$.)

An *AUC problem* is defined by a distribution D over $(X \times \{0,1\})^*$. The goal is to find an ordering $o : X \times X \to \{0,1\}$ minimizing the expected AUC loss on D,

$$l(o, D) = \mathbf{E}_{S\sim D} l(o, S).$$

Note that D may encode arbitrary dependencies between examples.

Regret We prove a transformation bound on *regret* rather than loss. Regret (generally) is how well we could have done in comparison to how well we did. It separates errors from unremovable noise in the problem, thus the bounds apply nontrivially even on problems with large inherent noise.

Formally, the *classification regret* of classifier c on distribution B on binary examples is defined as

$$r(c, B) = e(c, B) - \min_{c^*} e(c^*, B).$$

Similarly, the *AUC regret* of ordering function o on distribution D over $(X \times \{0,1\})^*$ is given by

$$r_{\text{AUC}}(o, D) = l(o, D) - \min_{o^*} l(o^*, D).$$

Our goal is to design a ranking algorithm for which a small binary regret incurred by the selector cannot imply a large ranking regret.

3 Ordering by the Number of Wins: Regret Transform

The reduction consists of two components. The training part, AUC-TRAIN (Algorithm 1), transforms mixed pairs of examples into binary data. (A pair $(x_1, y_1), (x_2, y_2)$ is *mixed* if $y_1 \neq y_2$.)

For any process D generating datasets $S \in (X \times \{0,1\})^*$, we can define an induced distribution on binary examples in $(X \times X) \times \{0,1\}$ by first drawing S from D, and then applying AUC-TRAIN to S. We denote this induced distribution by AUC-TRAIN(D).

The test portion, DEGREE (Algorithm 2), uses the pairwise classifier c learned in Algorithm 1 to run a tournament on the test set, and then creates an ordering according to the number of wins in the tournament, breaking ties randomly.

Theorem 1. *For all joint distributions D, for all pairwise classifiers c,*

$$r_{\text{AUC}}(\textsc{Degree}(D,c),D) \leq 4r(c,\textsc{Auc-Train}(D)).$$

Note the quantification in the above theorem: it applies to *all* settings where algorithms 1 and 2 are used, in particular to D with arbitrary dependences between examples.

Proof: Given an unlabeled test set $x^n \in X^n$, the joint distribution D induces a conditional distribution $D(Y_1,\dots,Y_n \mid x^n)$ over the set of label sequences $\{0,1\}^n$. We prove the theorem for any fixed x^n, and then take the expectation over the draw of x^n at the end. In the remainder of the proof $Q(y^n) = D(y^n|x^n)$ is the conditional distribution over y^n given x^n. Similarly, we replace x_i with i where it is unambiguous.

The first step is to rewrite the regrets in terms of a sum over pairwise regrets. A pairwise loss is defined by:

$$l_Q(i,j) = \mathbf{E}_{y^n\sim Q(Y^n)} \frac{I(y_i > y_j)}{\sum_{i<j} I(y_i \neq y_j)}.$$

If $l_Q(i,j) < l_Q(j,i)$, the *regret* $r_Q(i,j)$ of ordering i before j is 0; otherwise, $r_Q(i,j) = l_Q(i,j) - l_Q(j,i)$.

We can assume without loss of generality that the ordering minimizing the AUC loss (thus having zero AUC regret) is $x_1x_2\dots x_n$. All regret-zero pairwise predictions must be consistent with the ordering; i.e., $r_Q(i,j) = 0$ for all $i < j$.

The AUC regret of o on Q can thus be decomposed as a sum of pairwise regrets:

$$\begin{aligned}
r_{\text{AUC}}(o,Q) &= l(o,Q) - \min_{o^*} l(o^*,Q) = \mathbf{E}_{y^n\sim Q} l(o,S) - \min_{o^*} \mathbf{E}_{y^n\sim Q} l(o^*,S) \\
&= \mathbf{E}_{y^n\sim Q} \frac{\sum_{i,j} I(y_i > y_j) o(i,j)}{\sum_{i<j} I(y_i \neq y_j)} - \min_{o^*} \mathbf{E}_{y^n\sim Q} \frac{\sum_{i,j} I(y_i > y_j) o^*(i,j)}{\sum_{i<j} I(y_i \neq y_j)} \\
&= \max_{o^*} \mathbf{E}_{y^n\sim Q} \frac{\sum_{i,j} I(y_i > y_j) o(i,j) - I(y_i > y_j) o^*(i,j)}{\sum_{i<j} I(y_i \neq y_j)} \\
&= \sum_{i<j:o(j,i)=1} r_Q(j,i) = \sum_{k=1}^{n-1} |\{i \leq k < j : o(j,i) = 1\}| r_Q(k+1,k)
\end{aligned}$$

The last inequality follows from the repeated use of Lemma 3.1 which says that we can decompose the pairwise regret for any pair $i < j$ as:

$$r_Q(j,i) = \sum_{k=i}^{j-1} r_Q(k+1,k).$$

The classification regret can also be written in terms of pairwise regrets:

$$\begin{aligned}
r(c,\textsc{Auc-Train}(Q)) &= e(c,\textsc{Auc-Train}(Q)) - \min_{c^*} e(c^*,\textsc{Auc-Train}(Q)) \\
&= \max_{c^*} \mathbf{E}_{y^n\sim Q} \left[\frac{\sum_{i,j} I(y_i > y_j) c(i,j) - I(y_i > y_j) c^*(i,j)}{\sum_{i<j} I(y_i \neq y_j)} \right] \\
&= \sum_{i<j:c(j,i)=1} r_Q(j,i) = \sum_{k=1}^{n-1} |\{i \leq k < j : c(j,i) = 1\}| r_Q(k+1,k).
\end{aligned}$$

The proof is done if we can show that the coefficient on $r_Q(k+1,k)$ are within a factor of 4 for each k. For any particular k, we have the set the classifier c induces a tournament on the set of n elements. This is precisely the setting of theorem 2 where the partition is into the first k elements and the second $n-k$ elements.

∎

We prove the lemma used in the proof above.

Lemma 3.1. *For any i, j, and k in x^n,*

$$r_Q(i,j) + r_Q(j,k) = r_Q(i,k).$$

Proof: Let d_{ijk} be a short-hand for the restriction of $D(Y_1, \ldots, Y_n \mid x^n)$ to $\{Y_i, Y_j, Y_k\}$. A simple algebraic manipulation verifies the claim.

$$\begin{aligned}
&r_Q(i,j) + r_Q(j,k) \\
&\qquad = d_{ijk}(100) + d_{ijk}(101) - d_{ijk}(010) - d_{ijk}(011) \\
&\qquad + d_{ijk}(010) + d_{ijk}(110) - d_{ijk}(001) - d_{ijk}(101) \\
&\qquad = d_{ijk}(100) + d_{ijk}(110) - d_{ijk}(001) - d_{ijk}(011) = r_Q(i,k),
\end{aligned}$$

Notice that all label assignments above have exactly two mixed pairs, so the factor of $1/2$ is cancelled. ∎

4 Theorem

In this section we establish a basic result relating the two regret functions. Because the proof is somewhat involved, it will be presented in this self-contained section, with its own notation.

Consider a bipartition of n nodes $1, \ldots, n$ into a nonempty set L of "losers" and a nonempty set W of "winners". Let T_0 be a tournament on these nodes, with the property that "W dominates L": every node $j \in W$ beats every node $i \in L$.

For conciseness, define the function $\mathbf{1}(a,b)$ to be 1 if $a > b$, $\frac{1}{2}$ if $a = b$, and 0 if $a < b$. Our cost function is then

$$c_s^B(T_0, T_f) = \sum_{i \in L} \sum_{j \in W} \mathbf{1}\left(\mathrm{outd}^{T_f}(i), \mathrm{outd}^{T_f}(j)\right).$$

Also, given the two tournaments T_0 and T_f, let $r(i,j) = 0$ if the direction of edge (i,j) agrees in the two, and 1 if it disagrees. The adversary's cost function is then

$$c_s^B(T_0, T_f) = \sum_{i \in L} \sum_{j \in W} r(i,j).$$

Theorem 2. *For every n, every bipartition of $\{1, \ldots, n\}$ into nonempty sets W and L, every tournament T_0 in which every $j \in W$ dominates every $i \in L$, and every tournament T_f,*

$$\frac{c_s^B(T_0,T)}{c_s^A(T_0,T)} = \frac{\sum_{i \in L} \sum_{j \in W} \mathbf{1}\left(\mathrm{outd}^{T_f}(i), \mathrm{outd}^{T_f}(j)\right)}{\sum_{i \in L} \sum_{j \in W} r(i,j)} \leq 4. \tag{1}$$

We believe that in fact $c_s^B / c_s^A \leq 2$, but so far we can only prove the weaker bound. The proof comprises the remainder of this section.

We think of maximizing the ratio (1) over the space described by the theorem's hypotheses, and showing that the maximum is at most 4. The numerator of (1) depends only on T_f. If we simply transform T_0 into T_f by flipping the edges that disagree, the denominator is the number of edge reversals *between* L *and* W. Note that the denominator is unchanged if we replace T_0 with the tournament $T_0{}'$ which agrees with T_f on $L \times L$ and on $W \times W$, and (like T_0) has W dominating L. Thus, we may equivalently restrict the maximization to tournaments T_0 and T_f which agree on $L \times L$ and $W \times W$. For such a pair of tournaments, each edge reversal $r(i,j)$ contributing 1 to the denominator has the effect of increasing the degree of $i \in L$ by 1, and decreasing the degree of $j \in W$ by 1.

We may thus focus on the tournaments' degree spectra rather than on the tournaments themselves, as we now explain. Let D_0 be the degree spectrum of T_0, *i.e.*, D_0 is the sequence $\{d_0(1), \ldots, d_0(n)\}$ where $d_0(i) = \mathrm{outd}^{T_0}(i)$. Likewise, let $D_f = \{d_f(1), \ldots, d_f(n)\}$ be the degree spectrum of T_f. The numerator of (1) is determined by D_0 and the partition (L,W): it is

$$\sum_{i \in L} \sum_{j \in W} \mathbf{1}\left(d_f(i), d_f(j)\right). \tag{2}$$

The denominator is determined by the two degree spectra and the partition; it is

$$\frac{1}{2}\left[\sum_{i\in L}(d_f(i)-d_0(i))+\sum_{j\in W}(d_0(j)-d_f(j))\right]. \tag{3}$$

The theorem treats the maximum of the ratio of (2) to (3) where the degree sequences D_0 and D_f are derived from tournaments T_0 and T_f with certain properties, including that for all $i \in L$, $d_f(i) \geq d_0(i)$ and for all $j \in W$, $d_f(j) \leq d_0(j)$. Instead, we will maximize the ratio over sequences (no longer "degree sequences") D_0 and D_f where D_0 satisfies certain necessary conditions for the degree sequence of a tournament in which W dominates L, and the *only* constraint on D_f is that for all $i \in L$, $d_f(i) \geq d_0(i)$ and for all $j \in W$, $d_f(j) \leq d_0(j)$. We will show that the maximum is at most 4; since this is a relaxation of the original maximization problem, this proves that the original ratio is also at most 4.

We first establish a simple condition on the degree sequence D_0. For convenience, let $\ell_1, \ldots, \ell_{|L|}$ be the nodes of L ordered so that $\mathrm{outd}(\ell_i) \geq \mathrm{outd}(\ell_{i+1})$, so for example ℓ_1 is the best of the losers (or tied for that status). Similarly, let $w_1, \ldots, w_{|W|}$ be the nodes of W ordered so that $\mathrm{outd}(W_j) \geq \mathrm{outd}(w_{j+1})$, so w_1 is the worst of the winners. As W dominates L, it is immediate that $\mathrm{outd}(w_j) \geq |L|$ and $\mathrm{outd}(\ell_i) \leq |L|-1$.

Claim 4.1. *For all i and j, $\mathrm{outd}(w_j) \geq |L| + (j-1)/2$ and $\mathrm{outd}(\ell_i) \leq |L| - (i+1)/2$.*

Proof: Restricting T_0 to W gives a tournament ${T_0}^W$ whose outdegrees are $\mathrm{outd}'(w_j) = \mathrm{outd}(w_j) - |L|$. By Landau's theorem, for any j, $\binom{j}{2} \leq \sum_{k=1}^{j} \mathrm{outd}'(w_k)$, which by the non-decreasing nature of W's degree sequence is $\leq j \cdot \mathrm{outd}'(w_j)$. This gives $(j-1)/2 \leq \mathrm{outd}'(w_j) = \mathrm{outd}(w_j) - |L|$, yielding the claim's first inequality.

Similarly, restricting T_0 to L gives a tournament ${T_0}^L$ with the same outdegrees, $\mathrm{outd}'(\ell_i) = \mathrm{outd}(\ell_i)$. Consider the *indegrees* within ${T_0}^L$, and note that $\mathrm{ind}'(\ell_i) + \mathrm{outd}'(\ell_i) = |L|-1$. Just as above, by Landau's theorem, for any i, $(i-1)/2 \leq \mathrm{ind}'(\ell_i) = |L| - 1 - \mathrm{outd}(\ell_i)$, yielding the claim's second inequality. ∎

Comparing the degree sequences D_0 and D_f, for $i \in \{1, \ldots, |L|\}$ let $x(i) = d_f(\ell_i) - d_0(\ell_i)$ (the amount by which a loser's degree is increased) and for $j \in \{1, \ldots, |W|\}$ let $y(j) = d_0(w_j) - d_f(w_j)$ (the amount by which a winner's degree is decreased). Because W dominates L in T_0 and only edges in $W \times L$ are reversed in constructing T_f, all $x(i)$ and $y(j)$ are ≥ 0.

Note that the summation (2) may be replaced with

$$\sum_{i=1}^{|L|}\sum_{j=1}^{|W|}\mathbf{1}\left(d_0(\ell_i)+x(i), d_0(w_j)-y(j)\right) = \sum_{i\in L}\sum_{j\in W}\mathbf{1}\left(x(i)+y(j), d_0(w_j)-d_0(\ell_i)\right).$$

¿From Claim 4.1,

$$d_0(w_j) - d_0(\ell_i) \geq \left(|L| + \frac{j-1}{2}\right) + \left(\frac{i+1}{2} - |L|\right) = \frac{i+j}{2}.$$

Decreasing the second argument of $\mathbf{1}(\cdot,\cdot)$ can only increase its value, so it follows that (2) is

$$\leq \sum_{i=1}^{|L|}\sum_{j=1}^{|W|}\mathbf{1}\left(x(i)+y(j), \frac{i+j}{2}\right).$$

In our new indexing, (3) is rewritten as

$$\tfrac{1}{2}\left[\sum_{i=1}^{|L|}(d_f(\ell_i)-d_0(\ell_i))+\sum_{j=1}^{|W|}(d_0(\ell_j)-d_f(w_j))\right]. \tag{4}$$

Recall that by definition the sequence $d_0(\ell_i)$ is nonincreasing, and $d_0(w_j)$ nondecreasing. Without loss of generality we may assume the same is true for $d_f(\ell_i)$ and $d_f(w_j)$: in particular, we may

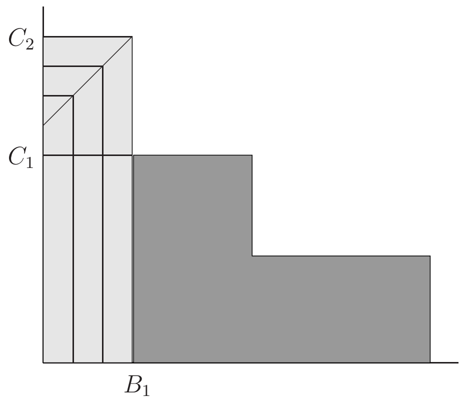

Figure 1: Tall and skinny protrusion

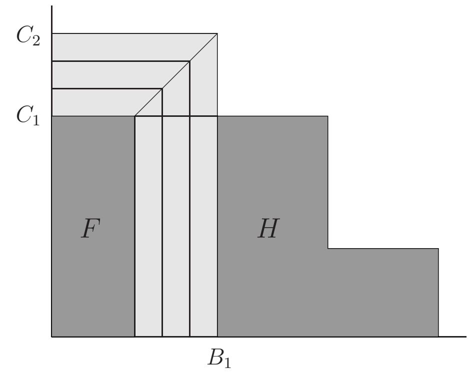

Figure 2: Short and wide protrusion

replace any sequences $d_f(\ell_i)$ and $d_f(w_j)$ with their sorted equivalents. Clearly such a replacement does not affect the value of (4). The only condition we will demand of the sequences $d_f(\ell_i)$ and $d_f(w_j)$ is that every difference $d_f(\ell_i) - d_0(\ell_i)$ and $d_0(\ell_j) - d_f(w_j)$ is non-negative, as is true in the original optimization space. But if this holds for $d_f(\ell_i)$ and $d_f(w_j)$ then it also holds for their sorted equivalents.

Also substituting into (3) with the newly defined $x(i)$ and $y(j)$, the ratio in (1) is at most the maximum of

$$\frac{\sum_{i\in L}\sum_{j\in W}\mathbf{1}\left(x(i)+y(j),\frac{i+j}{2}\right)}{\frac{1}{2}\left(\sum_{i\in L}x(i)+\sum_{j\in W}y(j)\right)}.$$

5 The Suboptimality of Minimization

References

[CLV05] S. Clemencon, G. Lugosi and N. Vayatis. Ranking and Scoring Using Empirical Risk Minimization, *Proceedings of the Eighteenth Annual Conference on Computational Learning Theory* (COLT), 2005.

[CSS99] W. Cohen, R. Schapire, and Y. Singer. Learning to order things, *Journal of Artificial Intelligence Research*, 10: 243–270, 1999.

[CFR06] D. Coppersmith, L. Fleischer, and A. Rudra. Ordering by weighted number of wins gives a good ranking for weighted tournaments, *Proceeding of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms* (SODA), 2006.

[CM04] C. Cortes and M. Mohri. AUC Optimization vs. Error Rate Minimization, *Advances in Neural Information Processing Systems* (NIPS 2003), 2004.

[FIS+03] Y. Freund, R. Iyer, R. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences, *Journal of Machine Learning Research*, 4: 933–969, 2003.

[LB05] J. Langford and A. Beygelzimer. Sensitive Error Correcting Output Codes, *Proceedings of the Eighteenth Annual Conference on Computational Learning Theory* (COLT), 2005.

[LZ05] J. Langford and B. Zadrozny. Estimating Class Membership Probabilities Using Classifier Learners, AI+STATS 2005.

[RCM+05] C. Rudin, C. Cortes, M. Mohri, and R. Schapire. Margin-based ranking meets Boosting in the middle, *Proceedings of the Eighteenth Annual Conference on Computational Learning Theory* (COLT), 2005.

[SHD05] S. Agarwal, S. Har-Peled, and D. Roth. A uniform convergence bound for the area under the ROC curve, *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics* (AISTATS), 2005.

[ZLA03] Bianca Zadrozny, John Langford, and Naoki Abe. Cost Sensitive Learning by Cost-Proportionate Example Weighting, *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM)*, 435–442, 2003.