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 the Area Under the Reciever Operating Characteristic Curve (AUC), to binary classification. The core theorem shows that a binary classification regret of r on the induced binary problem implies an AUC regret of at most $4r$. (The binary problem is to predict, given a random pair of elements being ranked, whether the first element should be ordered before the second.) 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 natural fundamental question is: "Are these problems truly different?" If they are, we require fundamentally different algorithms to optimize these different loss functions. If the answer is "no", 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.

Several basic observations help define this problem.

1. A reductions-style analysis bounds the realized AUC performance in terms of the realized classification performance. Since the analysis is relative, it does not have to rely on any assumptions about the way the world produces data. In particular, it can cope with arbitrary high-order dependencies between examples. This seems particularly important in a number of real-world applications where ranking is of interest.
2. A natural approach to solving ranking is to order examples according to some "score" or estimated conditional class probability. 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 can greatly harm AUC while only slightly affecting classification with respect

to the induced distribution. 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].

Main 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 *arbitrary* distributions over elements. (In particular, we don't assume that examples are drawn i.i.d.) The theorem is a large improvement over other approaches discussed above, which have 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 ($r \to \sqrt{r}$) [LZ05] or multiclass to binary classification ($r \to 4\sqrt{r}$) [LB05].

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 shows that good classification performance implies good ranking performance; thus it can be viewed as a robustness result showing that a large AUC cannot be induced with a small number of mis-classifications.

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 items. 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 [FS97].

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.

2 Formal Setup

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

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 $i \neq j$.)

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

1. Let
$$S' = \{\langle (x_1, x_2), I(y_1 > y_2)\rangle : (x_1, y_1), (x_2, y_2) \in S \text{ and } y_1 \neq y_2\}$$
2. 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.

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,*

$$r_{\text{AUC}}(\text{DEGREE}(D, c), D) \leq 4r(c, \text{AUC-TRAIN}(D)).$$

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 $D(Y_1, \ldots, Y_n \mid x^n)$ 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 $x_1 x_2 \ldots x_n$. 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:

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

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:

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

The proof is done if we can show that the coefficient on $r_Q(k+1,k)$ is 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.

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

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

Example The following example gives a lower bound of (almost) 2 on the ratio. We believe that this is the worst case.

Assume for simplicity that n is even. The adversary can force a total misorder by making the degree of every element in $\{x_{n/2+1}, \ldots, x_n\}$ larger than the degree of every element in $\{x_1, \ldots, x_{n/2}\}$. She can assign degree $n/2$ to every element in the first component, and degree $(n-2)/2$ to all elements in the second component. The number of times $r(x_{k+1}, x_k)$ appears in the AUC regret is thus given by $(n/2)^2$. The sum of out-degrees of nodes in $\{x_1, \ldots, x_{n/2}\}$ is $\frac{n-2}{2} \cdot \frac{n}{2}$, but $\binom{n/2}{2}$ have to be absorbed internally within the component. Thus the number of cross-component edges that the adversary can direct correctly is at most $\frac{n-2}{2} \cdot \frac{n}{2} - \binom{n/2}{2} = \frac{n(\frac{n}{2}-1)}{4}$. giving a bound of

$$\frac{n^2}{n^2 - n(\frac{n}{2}-1)} = \frac{n^2}{\frac{n^2}{2}+n} = 2 - \frac{4}{n+2}$$

on the ratio.

Abstract

Plugin.

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

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

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,*

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

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 of Theorem 2 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 perform the maximization only over 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.

Thus, we may rewrite the ratio in (1) as

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

Instead of maximizing the ratio only over degree sequences corresponding to tournaments satifying the hypotheses of Theorem 2, we will maximize it over the broader class of sequences satisfying the following two conditions:

A The sequence d_0 satisfies Landau's condition (see below); *i.e.*, it is the degree sequence of some tournament T_0.

B For all $i \in L$, $d_f(i) \geq d_0(i)$, and for all $j \in W$, $d_f(j) \leq d_0(j)$.

Note that both conditions are satisfied by tournaments obeying the theorem's hypotheses. This maximization is thus a relaxation of the original maximization problem; we will show that its maximum is at most 4, thus establishing the theorem.

For convenience, let $\ell_1, \ldots, \ell_{|L|}$ be the nodes of L ordered so that $d_0(\ell_i) \geq d_0(\ell_{i+1})$, 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 $d_0(W_j) \geq d_0(w_{j+1})$, so w_1 is the worst of the winners.

Without loss of generality we may assume that $d_f(\ell_i)$ is a nonincreasing sequence (like $d_0(\ell_i)$) and $d_f(w_j)$ is a nondecreasing sequence (like $d_0(w_j)$). In particular, we may 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 the denominator of (2). Also, if the original sequences satisfied condition (B), so do their sorted equivalents.

This simple fact has a nice "structural" consequence for the set of points (i, j) contributing to the numerator, call it $S = \{(i, j) \colon \mathbf{1}\left(d_f(\ell_i), d_f(w_j)\right)\}$. First, if $(i, j) \in S$, then for all $i' \leq i$ and $j' \leq j$, $(i', j') \in S$ as well.

It may be helpful to imagine S as an area drawn in the positive quadrant of a sheet of graph paper: the cell $[i-1, i] \times [j-1, j]$ is filled iff $(i, j) \in S$. The condition just established asserts that in this representation of S there are no "holes": the region is a solid one running from some point on the j axis down in some sort of staircase pattern to some point on the i axis (see Figures **??** and **??**).

Define $\mathsf{L}(i, j) = \{(i', j'), i, j \geq 0 \colon i' = i \text{ and } j' \leq j, \text{ or } j' = j \text{ and } i' \leq i\}$, *i.e.*, the point (i, j) together with all points directly left of it and all points below it. Note that if $(i, j) \in S$ then $\mathsf{L}(i, j) \subset S$.

Claim 4.1. *For any "staircase" region S there exists a (not necessarily perfect) matching C of rows i and columns j such that $S = \bigcup_{(i,j)\in C} \mathsf{L}(i, j)$.*

That is, there is a set of Ls which form a cover of S (it is permissible for them to overlap), and whose defining "corners" all lie in distinct rows and columns.

Proof: We write (i, j) to denote a point in $\mathbb{N}^2$, and $[i, j] = \{i, i+1, \ldots, j\}$ to denote an interval in $\mathbb{N}$. The proof is by induction on the area of S. Consider the topmost protrusion of S, just down to the level of the next "step" to the right. That is, say it extends from $i = 0$ to i_1, and from $j = 0$ to j_2, with j_1 defining the next-highest bit off to the right. Start covering from the protrusion's top-right corner (the point (j_1, j_2)) with nested Ls, working down and left to a point to be specified.

If the protrusion is taller than it is wide (if $j_2 - j_1 > i_1$), go until you bump into the left edge (the j axis). You've now covered the entire leftmost tower (from $(0, 0)$ to (i_1, j_2)) with Ls whose supporting columns are precisely the range $[0, j_1]$ and whose supporting rows are the range $[j_2 - j_1, j_2]$. What's left uncovered is an area right of j_1 and below j_1. By induction it can be covered with Ls with supporting columns right of i_1 and rows below j_1. This second set of Ls can thus be safely unioned with the first set, without any duplication of supporting columns or rows. All the area is covered. (The area $[0, j_1] \times [0, j_1]$ is doubly covered, which is allowed.)

If the protrusion is wider than it is tall, go until you bump into the horizontal line $j = j_1$. This covers the top protrusion, uses up all the rows $[j_1, j_2]$, and also uses up columns $[j_1 - (j_2 - j_1), j_1]$. The remaining area is thus all below j_1, and consists of the rectangle "F" from $(0, 0)$ to $(j_1 - (j_2 - j_1), j_1)$; a "gap G" (of covered area and forbidden rows) from $(j_1 - (j_2 - j_1), 0)$ to (j_1, j_1); and

then some more complex structure "H" to the right of j_1. Glue the first rectangle F together with the area H at the right (deleting the gap). Inductively cover this shape $F|H$ with Ls. Then pull the shape apart again (any L anchored in H now extends across the gap G). The new Ls use rows below j_1 (thus not conflicting with the first set, which were above j_1), and use columns in either F or in H (thus not conflicting with the first set, which were in G). The top rectangle and G are covered by the first set of Ls; F and H are covered by the second set; and thus the whole area is covered. ∎

Corollary 4.1. *If C is a matching covering S (in the sense of Claim 4.1) then the numerator of* (2) *is* $\leq \sum_{(i,j)\in C}(i+j-1)$.

Proof: S is the union of the Ls, and $\mathsf{L}(i,j)$ has cardinality $i+j-1$. ∎

Now we establish a simple condition on the degree sequence D_0. As W dominates L, it is immediate that $d_0(w_j) \geq |L|$ and $d_0(\ell_i) \leq |L|-1$.

Claim 4.2. *For all i and j, $d_0(w_j) \geq |L| + (j-1)/2$ and $d_0(\ell_i) \leq |L| - (i+1)/2$.*

Proof: Restricting T_0 to W gives a tournament ${T_0}^W$ whose outdegrees are $d'_0(w_j) = d_0(w_j) - |L|$. By Landau's theorem, for any j, $\binom{j}{2} \leq \sum_{k=1}^{j} d'_0(w_k)$, which by the nondecreasing nature of W's degree sequence is $\leq j \cdot d'_0(w_j)$. This gives $(j-1)/2 \leq d'_0(w_j) = d_0(w_j) - |L|$, yielding the claim's first inequality.

Similarly, restricting T_0 to L gives a tournament ${T_0}^L$ with the same outdegrees, $d'_0(\ell_i) = d_0(\ell_i)$. Consider the *indegrees* within ${T_0}^L$, and note that $\mathrm{ind}'(\ell_i) + d'_0(\ell_i) = |L| - 1$. Just as above, by Landau's theorem, for any i, $(i-1)/2 \leq \mathrm{ind}'(\ell_i) = |L| - 1 - d_0(\ell_i)$, yielding the claim's second inequality. ∎

Corollary 4.2. *If C is a matching covering S (in the sense of Claim 4.1) then the denominator of* (2) *is* $\geq \frac{1}{4}\sum_{(i,j)\in C}(i+j)$.

Proof: By definition, $(i,j) \in C$ implies $(i,j) \in S$, meaning that

$$d_f(\ell_i) \geq d_f(w_j)$$
$$d_0(\ell_i) + x(i) \geq d_0(w_j) - y(j)$$

and, by Claim 4.2,

$$x(i) + y(j) \geq d_0(w_j) - d_0(\ell_i) \geq \frac{i+j}{2}. \tag{3}$$

In our new notation, the denominator of (2) is simply

$$\tfrac{1}{2}\left[\sum_{i=1}^{|L|} x(i) + \sum_{j=1}^{|W|} y(j)\right] \geq \tfrac{1}{2}\sum_{(i,j)\in C}\left[x(i)+y(j)\right],$$

because C is a matching and the $x(i)$ and $y(j)$ are all nonnegative. From (3), this is

$$\geq \tfrac{1}{2}\sum_{(i,j)\in C}\left[\frac{i+j}{2}\right].$$

∎

The theorem is immediate from Corollary 4.1 and Corollary 4.2.

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.

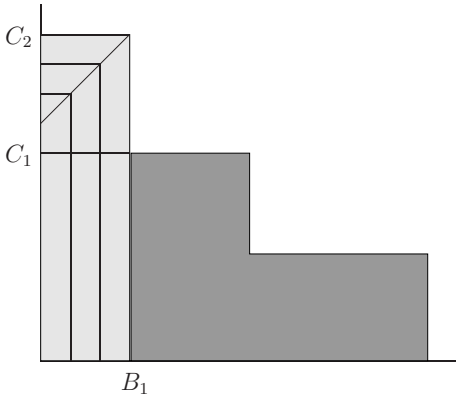

Figure 1: Tall and skinny protrusion

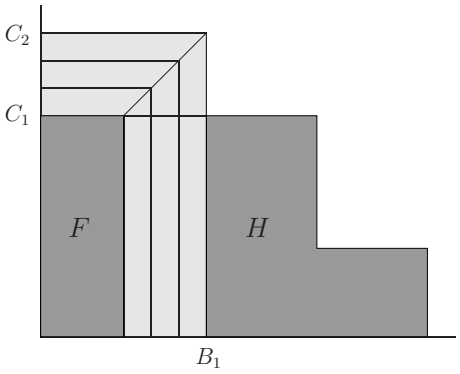

Figure 2: Short and wide protrusion

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

[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.

[FS97] Y. Freund, R. 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

[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.