Perturbation based Large Margin Approach for Ranking

Eunho Yang
University of Texas at Austin

Ambuj Tewari
University of Texas at Austin

Pradeep Ravikumar
University of Texas at Austin

Abstract

We consider the task of devising large-margin based surrogate losses for the learning to rank problem. In this learning to rank setting, the traditional hinge loss for structured outputs faces two main challenges: (a) the supervision consists of instances with multiple training labels instead of a single label per instance, and (b) the label space of the set of all permutations of items is very large, and less amenable to the usual dynamic programming based methods. The most natural way to deal with multiple labels leads, unfortunately, to a non-convex surrogate. We address this by first providing a general class of *convex* perturbation based surrogates as an extension of the large margin method. Our experiments demonstrate that a simple surrogate from this class performs better than other candidate large margin proposals for the learning to rank task.

1 Introduction

The task of ranking a set of instances by their relative relevance is of importance in many contemporary problems including collaborative filtering, text mining and document retrieval. We are interested in a particular formulation of this problem, natural in information retrieval (IR), where the ranking is at the resolution of a data item such as a query. Each query has a list of documents, and the task is to rank these documents in the order of relevance to the query. In the training set, the documents for each query are typically represented as feature vectors derived from the query-document pairs, and are annotated with relevance scores indicating the relative preference of the document in the list for that query. Given any new query, the goal is to rank its documents in an order that best respects their relevance scores according to some ranking evaluation measure. In this paper, we address the task of *learning to rank* Liu (2009) where we *train* a ranking model to fit the popular ranking evaluation measure of Normalized Discounted Cumulative Gain (NDCG) (Järvelin and Kekäläinen, 2000). We note that while we focus on NDCG, our general approach can be extended to other evaluation measures as well. Motivated by user studies, the NDCG evaluation measure evaluates the ranking of the entire list of documents by penalizing errors in higher ranked documents more strongly. While easy to *evaluate*, NDCG is nonetheless a difficult measure to directly use for training.

Surrogates; Large Margin. This discrepancy between ease of evaluation and difficulty in training occurs even in binary classification, with the zero-one loss. Considerable advances have thus been made on *surrogate loss functions* for binary classification which are more amenable to convex optimization. For instance, an exponential loss leads to the method of boosting Friedman et al. (2000), a logistic log-likelihood loss leads to the method of logistic regression, while a *Hinge* loss leads to very popular Support Vector Machines (SVMs) Hastie et al. (2001). In many cases, the construction of such surrogates has been extended to classification in domains where the labels are more structured and complex. For the particular case of the hinge loss, one of the most popular surrogate losses for binary classification, this extension to the case of structured outputs in general has been shown by Tsochantaridis et al. (2004). It is well known that the hinge loss function is intimately connected to the notion of a *large margin.* In binary classification, hinge loss evaluates to zero if the predicted label is not just correct, but *correct with a large margin.* The generalization of hinge loss to structured outputs preserves this intuition, but in this case the margin also depends on the specific label being compared against the true label. The structured hinge loss encourages the classifier to achieve larger margin of separation for labels that disagree more with the correct label. The loss allows smaller

Appearing in Proceedings of the 15^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2012, La Palma, Canary Islands. Volume XX of JMLR: W&CP XX. Copyright 2012 by the authors.

margins in turn for labels with higher agreement with the correct label in the underlying discrete/non-convex loss. The central question we focus on in this paper is whether the large margin approach can be used to derive a surrogate for the NDCG ranking evaluation metric.

Note that the broader question of deriving surrogates for ranking evaluation metrics has attracted considerable attention (Qin et al., 2007; Cao et al., 2007; Xia et al., 2008), motivated in part by successes of such an approach in classification. Moreover, there have been attempts to investigate large-margin approaches specifically for ranking Chapelle et al. (2007), though these suffer from some lacunae as we detail below.

Multiple Labels; Large Margin. The large margin surrogate in Tsochantaridis et al. (2004) was proposed for the general structured output case, but there is a vital caveat to its applicability in the ranking case: they make the assumption, natural in most settings, that there is a single true label in any training example. However, certain ambiguities arise when the supervision consists of *multiple labels*. This has been called the "learning with multiple labels" problem in Jin and Ghahramani (2002): the feedback or supervision available in the training set does not identify a unique correct label for each example. Instead, each example has a *set* of labels associated with it. This multiple label scenario, far from being just a theoretical curiosity, in fact, exactly captures the supervision available in datasets for ranking. In these datasets, each example consists of a query and a list of documents. Instead of a single permutation ranking the set of documents, each document is labelled with a relevance score from a finite set, say $\{1, 2, 3\}$. Since the output space is the set of permutations or rankings, multiple rankings are equally compatible with a given vector of relevance scores. Imagine one of the simplest cases: if one document has relevance 1 and the others 0, all rankings that puts it on the top position are equally good. The ambiguity issue which arises when extending the large margin approach in such settings (which we detail in the later sections) poses a challenge to be overcome if we are to replicate the success of large margin methods to the setting of multiple true labels in general, and in ranking in particular. These difficulties in extending large margin methods to the multiple true label case, and in particular to the ranking setting, have surfaced in recent work Chapelle et al. (2007). There, it was shown that arbitrarily resolving these ambiguities does not solve the problem and we confirm this in our experiments. We note that there has been quite a bit of work in the area of multiple ambiguous labels among which only one is true (Ambroise et al., 2001; Jin and Ghahramani, 2002; Vannoorenberghe and Smets, 2005; Hullermeier and Beringer, 2006; Côme et al., 2008; Cour et al., 2011). But most of the approaches, with the exception of Cour et al. (2011), are not based on convex surrogate minimization. While Cour et al. (2011) do propose a convex surrogate, it cannot scale to ranking problems since it involves a sum over the ambiguous labels.

Perturbation based Large Margin Approach. As we show, the most natural way to deal with multiple labels leads, unfortunately, to a non-convex surrogate. One of the main contributions of this paper is to show how the ambiguities can be resolved *while still preserving convexity* of the derived surrogate. Towards this, we first devise a "perturbation based approach" to large margin methods, and then show how these can be extended to the ranking case while obtaining a convex surrogate. Experiments on LETOR datasets shows that our proposed "Hinge" convex surrogate (a) performs better than the other large margin proposals for such a multiple label case, and (b) indeed compares favourably with one of the state-of-the-art surrogate loss functions in ranking. This shows that the large margin approach can be successfully generalized to optimize the non-convex ranking evaluation metric NDCG. We note that our approach, in keeping with the spirit of NDCG, is truly *listwise*: we do not reduce the ranking problem to *pairwise* binary comparisons or to *pointwise* regression on relevance scores. We also note that the developments in this paper would be independent interest even outside the context of ranking: (a) a perturbation based large margin approach, and (b) efficiently extending this revised large margin approach to the multiple label setting.

2 Problem Setup

In the general setting of supervised learning with structured outputs, we have the space of inputs or features $\mathcal{X}$, and the space of structured outputs $\mathcal{Y}$. We are given a training set $\{X_i, Y_i\}_{i=1}^n$, of input and output pairs, and the goal is to learn a map $h : \mathcal{X} \mapsto \mathcal{Y}$ such that $h(x)$ maps an input x to the true label y. For the specific case where $h(x)$ belongs to the class linear maps, we consider the set of functions, $h(x; w) = \arg\max_{y \in \mathcal{Y}} w^T \Phi(x, y)$, which, given features $\Phi(x, y) \in \mathbb{R}^d$, are indexed by weights $w \in \mathbb{R}^d$. Given some loss function $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$ that captures the discrepancy or disagreement between these labels, empirical risk minimization for the loss ℓ then entails solving the following optimization problem:

$$\min_w \sum_{i=1}^n \ell(h(X_i; w), Y_i). \tag{1}$$

This optimization problem is typically intractable: the objective is typically not convex, and usually not even differentiable, in the weights w. The state of the art supervised learning methods thus use surrogate objectives instead.

Large Margin. A popular class of surrogates are derived using the *large margin approach* which use the following idea: the training error is zero when $w^T\Phi(X^i, Y^i) \geq w^T\Phi(X^i, y)$ for $y \neq Y_i$, so that the true labels have a higher score than the other labels. We can strengthen this by requiring that $w^T\Phi(X^i, Y^i) \geq w^T\Phi(X^i, y) + \ell(Y^i, y)$, for all $y \in \mathcal{Y}$. It is natural to allow for slackness in these constraints, so that we allow $w^T\Phi(X^i, Y^i) \geq w^T\Phi(X^i, y) + \ell(Y^i, y) - \xi_i$, for some $\xi_i \geq 0$, and add a penalty that is linear in these slack variables. This yields the structured SVM formulation of Tsochantaridis et al. (2004) as below:

$$\begin{aligned} \min_{\xi} \quad & \sum_{i=1}^{n} \xi_i \\ \text{s.t.} \quad & w^T\Phi(X^i, Y^i) \geq w^T\Phi(X^i, y) + \ell(Y^i, y) - \xi_i, \forall y \in \mathcal{Y}, \\ & \xi_i \geq 0, \qquad i = 1, \ldots, n, \end{aligned}$$

where we have elided the usual ℓ_2 regularization on the weights.

Score Vectors. The hypothesis set of candidate maps $h : \mathcal{X} \mapsto \mathcal{Y}$ above used the set of linear maps $h(x; w) = \arg\max_{y\in\mathcal{Y}} w^T\Phi(x, y)$. It would be useful in the sequel to generalize this in terms of real-valued score vectors $s \in \mathbb{R}^d$. Thus, we consider maps $h : \mathbb{R}^d \mapsto \mathcal{Y}$ that take score vectors to labels in the label space, and which use a *discriminant* function $f : \mathbb{R}^d \times \mathcal{Y} \mapsto \mathbb{R}$, so that $h(s) \in \arg\max_y f(s, y)$. This discriminant function f captures the compatibility between the score vector and the labels, so that the map h given a score vector assigns the label which is most compatibile to the score vector.

The evaluation loss function $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$ can be naturally extended to allow *score* vectors instead of labels, so that we have a loss function $\boldsymbol{\ell} : \mathbb{R}^d \times \mathcal{Y}$, where the first argument now takes score vectors instead of labels, $\boldsymbol{\ell}(s, y) = \ell(h(s), y)$. We will overload notation and use $\boldsymbol{\ell}$ and ℓ interchangeably. The goal in devising surrogate objectives is to obtain alternative loss functions $\tilde{l}(s, y)$ that are convex and more amenable to optimization. The large margin approach of Tsochantaridis et al. (2004) in particular substitutes the following objective in place of the loss $\ell(h(s), Y)$:

$$\begin{aligned} \phi(s, Y) = \min_{\xi} \quad & \xi \\ \text{s.t.} \quad & f(s, Y) \geq f(s, y) + \ell(y, Y) - \xi, \\ & \xi \geq 0, \quad \forall y \in \mathcal{Y}, \end{aligned}$$

This surrogate "hinge" loss function $\phi : \mathbb{R}^d \times \mathcal{Y} \mapsto \mathbb{R}$ can be rewritten as:

$$\phi(s, Y) = \max_{y\in\mathcal{Y}} \{\ell(y, Y) + f(s, y) - f(s, Y)\}.$$

2.1 Ranking

We are interested in the structured output setting of ranking documents in response to queries. Specifically, each query has a list of documents, and the task is to rank these documents in the order of relevance to the query. In the training set, the documents for each query are typically represented as feature vectors derived from the query-document pairs, and are annotated with relevance values indicating the relative preference of the document in the list for that query. Given any new query, the goal is to rank its documents in an order that best respects their relevance scores according to some ranking evaluation measure.

To simplify notation, we assume that the number of documents for all queries is identically m. Let $\bar{\mathcal{X}} \subseteq \mathbb{R}^d$ be the space of the feature vectors in which the documents are represented (typically derived from the query-document pairs) and $\bar{\mathcal{R}} \subseteq \mathbb{R}$ be the space of the relevance level each document receives. Thus for any query, we have a list $\mathbf{X} = (X_1, \ldots, X_m) \in \mathcal{X} := \bar{\mathcal{X}}^m$ of document feature vectors, and a corresponding list $\mathbf{R} = (R_1, \ldots, R_m) \in \mathcal{R} := \bar{\mathcal{R}}^m$ of document relevance vector. The dataset consists of n $(\mathbf{X}^i, \mathbf{R}^i)$ pairs which we assume to be drawn *iid* from some distribution over $\mathcal{X} \times \mathcal{R}$.

Typical losses for the ranking care only about top ranked documents, since this mirrors the satisfaction or utility functions of typical users. Thus, even the NDCG criterion is usually truncated at a particular level k and $\ell(y, \mathbf{R})$ depends only the first k items according to the permutation y with the following definition:

$$\ell_{\text{NDCG@k}}(y, \mathbf{R}) = -\frac{1}{Z_k(\mathbf{R})} \sum_{j=1}^{k} \frac{G(R_{y^{-1}(j)})}{F(j)}. \qquad (2)$$

Here, F and G are arbitrary monotonically increasing functions; usually, $G(a) = 2^a - 1$, and $F(a) = \log_2(a+1)$. The normalization $Z_k(\mathbf{R})$ is the maximum possible value (over y) of the sum, y^{-1} is the inverse of permutation y (thus $R_{y^{-1}(j)}$ denotes the relevance level of the jth ranked document in the order given by y).

The goal in supervised learning to rank, is to learn a ranking function $h : \mathcal{X} \mapsto \mathcal{Y}$, where $\mathcal{Y}$ is the set of all degree m permutations. Note however, instead of being given a single permutation as the training label, the supervision consists of a vector of relevance levels

$\mathbf{R}$ assigned to the documents. A relevance level vector $\mathbf{R}$ however need not imply any single permutation in $\mathcal{Y}$, since the same relevance levels could be assigned to multiple documents. This is common especially if m is large and it is difficult to provide a full permutation as the training label given the list of m documents given a query. Even the loss function (2) thus has the form $\ell : \mathcal{Y} \times 2^{\mathcal{Y}} \mapsto \mathbb{R}$. In this paper, we investigate how to extend the large margin approach to this multiple label setting.

2.2 Multiple Label Learning

In many settings like ranking above, the training data points have more than one label given an input. In this case, the large margin approach that hinges on separating the true label from the rest of the label suffers from an inherent ambiguity: there is no one true label which can be separated from the rest.

Suppose $\mathbf{Y}$ is the set of training labels for some input. Le et al. (2009) suggest the following natural extension.

$$\phi(s, \mathbf{Y}) = \max_{y \in \mathcal{Y}} \left\{ \ell(y, \mathbf{Y}) + f(s, y) - f(s, \widetilde{y}) \right\}, \qquad (3)$$

where $\widetilde{y} \in \mathbf{Y}$ is one of the labels in the set of true labels. One caveat with this formulation is that this is not even a well-defined function since $\widetilde{y}$ can take any value in the true label set $\mathcal{Y}$. Moreover, Chapelle et al. (2007); Le et al. (2009) observed that in the context of ranking (specifically on the Ohsumed dataset in the LETOR package), this formulation had a vacuous optimal solution, specifically with linear scoring functions the optimum yielded weights equal to zero.

Another natural alternative would be the following:

$$\phi(s, \mathbf{Y}) = \max_{y \in \mathcal{Y}} \left\{ \ell(y, \mathbf{Y}) + f(s, y) - \min_{z \in \mathbf{Y}} f(s, z) \right\}. \qquad (4)$$

This corresponds to requiring that *all* true labels be sufficiently separated from the rest of the labels. This, however is very conservative, especially in settings where an output of *some* true label is typically sufficient. For instance, consider ranking where we are interested in the top k ranks, and there are $m > k$ documents that are highly and equally relevant. We would then be interested in any permutation such that k of these m documents occur in the top k ranks, whereas the loss (4) would not penalize any permutation only if *all of the* relevant documents are ranked above the non-relevant documents. Thus, the variant (4) though convex typically performs poorly, particularly with noisy data where it is typically difficult to train weights so that *all* relevant documents are ranked earlier.

It would be much more natural to instead require that only *some* true label be sufficiently separated from the rest of the labels. This in turn corresponds to the following loss:

$$\phi(s, \mathbf{Y}) = \max_{y \in \mathcal{Y}} \left\{ \ell(y, \mathbf{Y}) + f(s, y) - \max_{z \in \mathbf{Y}} f(s, z) \right\}.$$

This however has the caveat that it is non-convex. Le et al. (2009) provide a concave-convex (also called difference of convex or DC) procedure for solving such an objective.

Besides the multiple true labels issue, point-wise maximum operation over all permutations is expensive because size of $\mathcal{Y}$ increases factorially in m. To solve (3), Le et al. (2009) propose approximation techniques using cutting plane methods (Tsochantaridis et al., 2005).

3 Perturbation based Largin Margin Approach

In this section, we outline two main characteristics of large margin methods, and then proceed to generalize those to devise a novel class of methods; which we show is more amenable to the multiple label setting detailed in the previous section.

The first characteristic of large margin methods is that they allow a perturbation via slack variables in order for the true label to have a higher score than the rest of the labels. In particular, it allows a slack variable $\xi > 0$, such that if Y is the training label, then $f(s, Y) + \xi \geq f(s, y) + \ell(y, Y)$. We will be allowing a simpler notion of perturbation: where we allow direct perturbations $\delta \in \mathbb{R}^m$ to the score vector so that the perturbed score vector $s + \delta$ would satisfy the respective constraints.

The second characteristic of large margin methods is that they use the notion of a large margin to quantify the compatibility of a score vector s to the training label Y. Here, we will be allowing a more general compatibility constraint that $s \in \mathscr{C}(Y)$, where $\mathscr{C}(Y)$ is specified by the class of methods, and is the set of score vectors that are compatible with the training label Y. In the case of large margin methods for instance, we had $\mathscr{C}(Y) = \{s \in \mathbb{R}^d : f(s, Y) + \xi \geq f(s, y) + \ell(y, Y)\}$. In the sequel we will use the notation $s' \rightsquigarrow y$ to indicate the constraints $s' \in \mathscr{C}(y)$ that s' is compatible with y.

We then define the following surrogate loss:

$$\begin{aligned} \phi(s, r) \quad &= \quad \min_{\delta \in \mathbb{R}^m} g(\delta) & (5) \\ \text{s.t.} \quad & \quad \delta \succeq 0 \\ & \quad s' = s + \delta \\ & \quad s' \rightsquigarrow r \end{aligned}$$

Here, g is an arbitrary mapping from $\mathbb{R}^m$ to $\mathbb{R}$.

We first observe that this surrogate loss is convex under natural conditions on g and $\rightsquigarrow$.

Proposition 1. *Consider the loss ϕ as defined in 5. Suppose the constraint sets $\mathscr{C}(y)$ are convex for any $y \in \mathcal{Y}$, and g is convex in δ. We then have that the loss ϕ is convex in its first argument.*

Proof. We define h by

$$h(s, \delta) = \begin{cases} g(\delta) & \text{if } (s, \delta) \text{ satisfies the constraints of (5)} \\ \infty & \text{otherwise.} \end{cases}$$

If $\mathscr{C}(y)$ is convex and g is convex in δ, then h is jointly convex in (s, δ). At the same time, ϕ is the minimum of h over δ in nonempty convex set, and hence is convex (Boyd and Vandenberghe, 2004, p.87). □

Example: Multi-Class Classification. As an example, we consider multi-class classification, and verify that the usual hinge surrogate for multi-class classification (Crammer et al., 2001) is also a member of the perturbation based surrogate family (5). Consider the K-class classification task ($K \geq 2$), where the inputs X_i are drawn from a domain $\mathcal{X} \subseteq \mathbb{R}^d$ and Y_i is an integer from the set $\{1, \ldots, K\}$. Using linear maps as in (Crammer et al., 2001), suppose we have K weight vectors, $\{w_j\}_{j\in[K]}$, one for each class, and form a prediction for any input X using the map $h(X; \mathbf{w}) = \arg\max_{k=1,\ldots,K}\{w_k^T X\}$. In multi-class SVM, we learn these weights $\{w_j\}_{j\in[K]}$ by solving the following optimization problem:

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}\sum_{k=1}^{K} \|w_k\|_2^2 + \frac{C}{n}\sum_{i=1}^{n} \xi_i \qquad (6) \\ \text{s.t.} \quad & w_{Y_i}^T X_i + 1(Y_i = k) - w_k^T X_i \geq 1 - \xi_i \;\forall i, k. \end{aligned}$$

where $\mathbf{w} \in \mathbb{R}^{d\times K}$ is the concatenation of weight vectors and C is the regularization parameter.

The SVM objective used score vectors $s \in \mathbb{R}^K$ formed from a linear map $(w_1^T X_i, \ldots, w_K^T X_i)$. Expressing the SVM hinge loss in (6) in terms of these score vectors, we get

$$\begin{aligned} \phi_{\text{svm}}(s, y) = \min_{\xi} \quad & \xi \\ \text{s.t.} \quad & s_y \geq s_z + 1 - \xi, \forall z \neq y, z \in \mathcal{Y} \\ & \xi \geq 0. \qquad (7) \end{aligned}$$

Consider the following surrogate loss from the perturbation family (5):

$$\begin{aligned} \phi_{\text{class}}(s, y) \quad = \quad & \min_{\delta\in\mathbb{R}^K} \|\delta\|_1 \qquad (8) \\ \text{s.t.} \quad & \delta \succeq 0 \\ & s' = s + \delta \\ & s'_y \geq 1 + s'_z \forall z \neq y, z \in \mathcal{Y} \end{aligned}$$

Thus, using the loss in (8) for empirical risk minimization with score vectors derived from linear maps would yield:

$$\min_{\mathbf{w}} \frac{1}{2}\sum_{k=1}^{K} \|w_k\|_2^2 + \frac{C}{n}\sum_{i=1}^{n} \phi_{\text{class}}(X_i^T \mathbf{w}, Y_i).$$

The following proposition shows that the loss in (8) is equivalent to the SVM based loss (7)

Proposition 2. *Suppose the surrogate losses ϕ_{svm} and ϕ_{class} are defined as in* (7) *and* (8) *respectively. Then for all score vectors $s \in \mathbb{R}^K$, and labels $y \in \{1, \ldots, K\}$,*

$$\phi_{svm}(s, y) = \phi_{class}(s, y).$$

Proof. Suppose δ^* be the minimizer of optimization problem (8). We argue that $z \neq y$, suppose that $\delta_z^* = 0$. To see this, note that if the constraint $s'_y \geq 1 + s_z + \delta_z^*$, is satisfied for $\delta_z > 0$, then it would be satisfied for $\delta_z = 0$ as well, but which would strictly lower the objective $\|\delta\|_1$. Therefore, $\delta_z^* = 0$ for all $z \neq y$, and $\|\delta^*\|_1 = |\delta_y^*|$. The two objectives are then identical, with $|\delta_y^*|$ serving as the slack variable ξ in the multi-class hinge loss. □

Thus the hinge loss for multi-class classification task can be seen to fall under the perturbation based family of surrogate losses.

3.1 Perturbation based Large-Margin Surrogates for Ranking

In this section, we will apply the perturbation based large-margin machinery to derive surrogates for the NDCG ranking loss. Similar to the multi-class SVM case, if we set $g(\delta)$ to $|\delta|_1$ and the constraint set of $s' \rightsquigarrow r$ to include the usual large-margin constraints, then (5) yields the same Hinge loss as the structured SVM formulation of Tsochantaridis et al. (2004), and which is intractable.

To address the multiple training label problem, we propose a novel member of the family from (5) instead. First, we restrict the constraint set to the pairwise constraints, which check the compatability of pairs of coordinates of the score vector s with respect to the relevance vector $\mathbf{R}$. Note however that we set the objective $g(\delta)$ in a *listwise* manner. In contrast to the multi-classification task, where every δ_k had an equal contribution (though this might not be the case for the corresponding loss for cost-sensitive classification), in ranking we have a discounted loss: the higher positions in the ordering matter more. For example, suppose that we have 100 documents to sort and $\mathbf{R}_1 > \cdots > \mathbf{R}_{100}$. Even if for some weight vector $w \in \mathbb{R}^d$ we have scores so that $s_{99} < s_{100}$, and because of which we $\delta_{99} > 0$. This is not as serious problem as the case of $\delta_1 > 0$ (error in the first position). Thus, in the sequel, we use a weighted ℓ_1 norm for $g(\delta)$. Note that while we could use an arbitrary "discounting" function that discounts lower positions, we use a weighted ℓ_1 norm to make it linearly proportional to the margin.

We are now ready to describe the perturbation based surrogate for ranking as follows:

$$\begin{aligned}\phi_{rank}(s,\mathbf{R}) &= \min_{\delta\in\mathbb{R}^m} \langle \nu,\delta\rangle \qquad (9)\\ \text{s.t.} \quad & \delta \succeq 0\\ & s' = s+\delta\\ & s'_i \geq \Delta_{i,j} + s'_j, \text{ if } \mathbf{R}_i > \mathbf{R}_j.\end{aligned}$$

where ν and $\Delta_{i,j}$ are constants depending on $\mathbf{R}$.

With the score vector s arising from a linearly parameterized map with weights $\mathbf{w}$ as earlier, the estimation of the weights via empirical loss minimization can be formulated as

$$\min_w \frac{1}{2}\|w\|_2^2 + \frac{C}{n}\sum_{i=1}^{n} \phi_{rank}(w^T\mathbf{X}^i, \mathbf{R}^i) \qquad (10)$$

Note that superscript i here indexes query id. For each query i, $\mathbf{X}^i$ is the set of m document features, $\mathbf{X}^i \in \mathbb{R}^{d\times m}$ and $w^T\mathbf{X}^i$ will give a list of m scores with linear weight $w \in \mathbb{R}^d$.

As discussed above, extending traditional hinge surrogates for ranking suffer from two issues: (i) non-convexity and (ii) the large size of the label space of permutations. Note that the surrogate derived from the minimum perturbation approach is not only convex but also tractable because the number of constraints for a single query is at most m^2.

Comparison with Ranking SVM (Joachims, 2002). A broad line of work Cao et al. (2007) has focused on breaking the ranking problem down into pointwise, pairwise and listwise problems. In the *pointwise* approach, the ranking problem is viewed as a regression or classification problem of predicting the specific relevance score for any document Xu and Li (2007). In the *pairwise* approach on the other hand, the ranking problem is reduced to the binary classification task of predicting the more relevant document amongst pairs of documents. The caveat with such pointwise and pairwise approaches is that they are ill-suited to evaluation measures as NDCG which are *listwise*: that is their evaluation is a function of the entire list of ranked documents. Cao et al. (2007); Xia et al. (2008) in particular note that methods based on *listwise* loss functions outperform their pointwise and pairwise counterparts.

The Ranking SVM however is an explicitly *pairwise* surrogate. For instance, in the above example with 100 documents, suppose the scores satisfy $s_1 > \ldots > s_{96} > s_{98} > s_{99} > s_{100} > s_{97}$: where all documents before $97-100$ are ranked (correctly), while these are ranked last place (with misordering). However, the pairwise loss would still incur a large penalty for the 3 misclassified pairs, irrespective of their locations. This might hurt the learning performance drastically. On the other hand, our surrogate in (9) is a listwise loss: if ν_i is inversely proportional to the true location of document i based on $\mathbf{R}$, then the objective in (10) will not suffer a large loss for this example.

4 Experiments

In this section, we report empirical results demonstrating the performance of our proposed large-margin ranking surrogate (10) (denoted as *'Min.Perturb'* in the plots). In all experiments, i-th element of ν is set to $mean\{1/F(\min_y y(i)),\ldots,1/F(\max_y y(i))\}/Z_k(\mathbf{R})$ where y is an arbitrary permutation compatible with $\mathbf{R}$ and $y(i)$ the ranking of i-th document by y. For example, suppose we have a list of three document features $\mathbf{X} = (X_1, X_2, X_3)$ and a corresponding relevance vector of $(1, 1, 0)$. Then, $mean\{1/F(1), 1/F(2)\}/Z_3(\mathbf{R})$ for ν_1 and ν_2 and $1/(F(3)Z_3(\mathbf{R}))$ for ν_3. For $\Delta_{i,j}$, we test values from the set $\{10^{-4},\ldots,10^{-1}\}$ on the validation fold to tune the parameter.

As baselines, two other convex large margin surrogates are compared against our proposed one: (i) Following Le et al. (2009), we can randomly pick a permutation in y^* to break ties in (3) (denoted as *'Random'*). (ii) We can require that all true labels be sufficiently separated from the rests as (4) (denoted as *'MaxMax'*).

For the regularization parameter C for all three surrogates, we again use the cross-validation from $\{10^{-5}, 10^{-4},\ldots,10^{0}\}$ to find the best single parameter for all 5 folds. To avoid confusion, please note that in the earlier sections, we used *losses* which we wanted to minimize, instead of *gains*. However, for reporting our results we adhere to reporting the NDCG gain (as is the convention in IR). Thus, in these plots,

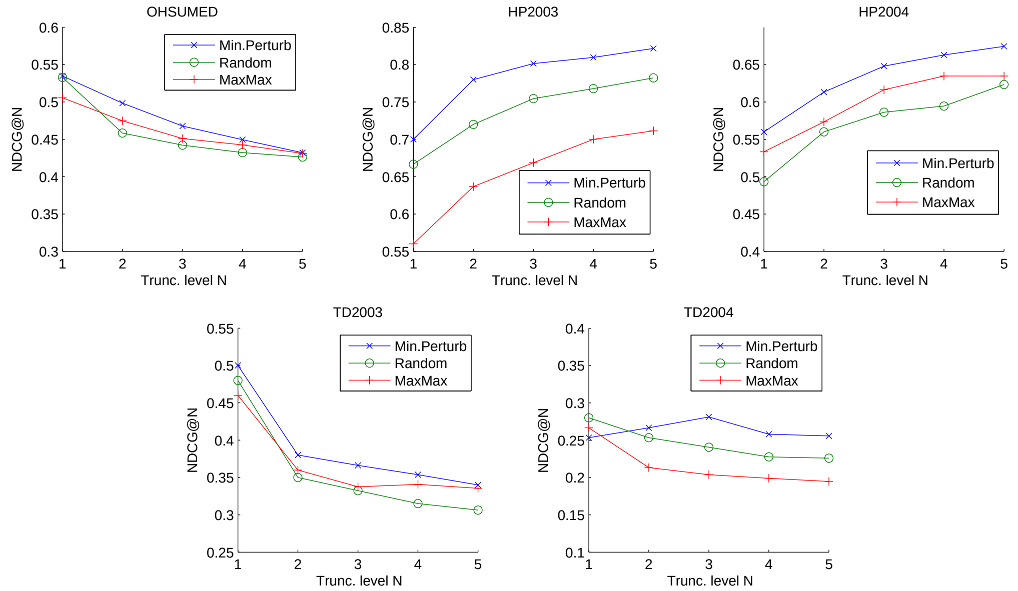

Figure 1: NDCG@1–NDCG@5 results on LETOR small datasets, comparing our large-margin surrogate to the alternative surrogates on LETOR.

higher NDCG values are *better*.

4.1 Synthetic Small Datasets

In the first experiments, we compare two surrogates on the synthetic *small* datasets where we can exactly solve even the optimization problem of (3). For the training set, we randomly choose 5 documents for a query from the LETOR datasets (they will be detailed in the next subsection). For the validation and test sets, all the documents are selected. To solve optimization problems, we use `CVX`, a matlab package for specifying and solving convex programs (Grant and Boyd, 2011). While all the parameters are tuned for NDCG@5, Figure 1 reports NDCG@1 through NDCG@5 results and Table 1 shows their training times in seconds. Since solving optimization problem (10) does not require iterating all possible permutations, training time is much smaller than other alternative large-margin surrogates. Nevertheless, proposed surrogate is consistently better than or equal to alternatives.

Table 1: Training Time (in sec.)

DATASET	Min.Perturb	Random	MaxMax
OHSUMED	**8.6**	150.8	3361.4
HP2003	**77.6**	535.7	25151
HP2004	**45.5**	162.4	8814.8
TD2003	**15.3**	38.2	932.9
TD2004	**25.8**	73.1	2435.4

4.2 LETOR Datasets

In the second set of experiments, we evaluate the surrogates on LETOR (Liu et al., 2007) v3 standard benchmark collections for learning to rank. These benchmarks target four tasks over two collections: 2003-2004 TREC Web track Craswell and Hawking (2005) tasks of (1) Homepage finding, (2) Named page finding, and (3) Topic distillation on the .GOV collection (1.25 million page 2002 crawl of the .gov domain), as well as (4) biomedical search on the older OHSUMED collection (350,000 documents, titles and abstracts without full-text) Hersh et al. (1994). LETOR includes a standard 5-fold partition of each dataset (3 training, 1 validation, and 1 test); our reported results reflect an average over the 5 test folds. Note that out of 7 datasets, we only report the results on 5 excluding 'Named page finding' tasks where mostly only one document for each query is relevant and all surrogates perform comparable.

Since all queries in the LETOR datasets usually have more than 100 documents, optimizing NDCG@k for $k \geq 2$ is computationally intractable. To solve this issue, we use the cutting plane method which boils down to iterating between finding argmax in (3) and solving the optimization problem with selected subset of constraints, as proposed in Le et al. (2009). Here, it is not trivial to solve (4) efficiently even with cutting plane due to the double max operations in (4). Therefore, in Figure 2, we just compare our surrogate, (3) with randomly breaking ties and Ranking SVM (Joachims, 2002), *pairwise* large-margin surrogate for the ranking

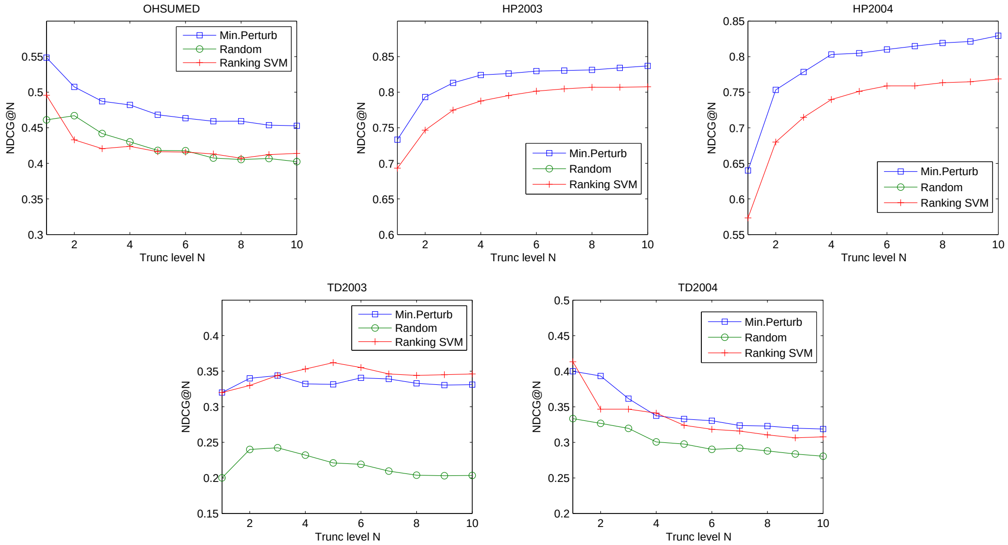

Figure 2: NDCG ranking accuracy achieved across five LETOR (Liu et al., 2007) datasets. Note that, to maintain same y-axis intervals across the datasets, we intentionally exclude the results for ‘Random’ on HP2003 and HP2004 datasets where it performs really poorly (NDCG values are below 0.3).

task. Note that the results of Ranking SVM came from LETOR web page and they are based on much more fine-grained parameter tuning for C. Solving (3) with random tie breaking performs terribly on 3 datasets (HP2003, HP2004 and TD2003) out of 5. We guess it is not only because of the approximation to solve the optimization problem but also because of the multiple true labels issue mentioned in section 2.2. On the other hand, our proposed surrogate is uniformly either better or comparable to the others.

Even though surrogates (3) and (4) usually cannot be solved exactly for a general ℓ, if we use $-$NDCG@1 for ℓ, then ‘max’ operation in them can be replaced with ‘$\sum$’ and therefore we can solve it exactly with reasonable number of constraints because only the top document in the permutation decides the loss. Table 2 compares NDCG@1 values after applying this modification to (3) and (4). Note that in this experiment, parameter C for them is also selected from the cross-validation targeting at NDCG@1.

Table 2: Optimizing NDCG@1

DATASET	Min.Perturb	Random	MaxMax
OHSUMED	0.5734	0.5045	**0.5841**
HP2003	**0.7400**	0.7000	0.7400
HP2004	**0.6400**	0.6080	0.6267
TD2003	**0.3200**	0.2600	0.2200
TD2004	**0.4133**	0.4000	0.3467

Finally, we evaluate NDCG@10 of our surrogate against ListNET Cao et al. (2007), one of the best *listwise* rank algorithms listed in LETOR. Note that values in the Table 3 are the relative improvements on ListNET as a base and negative sign means that ListNET performs better. For most datasets, two surrogates perform comparably.

Table 3: Relative Improvements against ListNET(%), NDCG@10

DATASET	
OHSUMED	2.68
HP2003	-0.02
HP2004	5.72
TD2003	-4.97
TD2004	0.38

5 Conclusions

We studied perturbation based surrogates and showed that existing hinge loss surrogates for multiclass and structured output classification can be cast in this minimum perturbation framework. Finally, we derived a convex surrogate in this class for the ranking problem.

For future work, we will investigate further refinement of our methods with different settings of ν, Δ. We are also interested in pursuing a principled way to tune these constants for different evaluation metrics such as ERR (Chapelle et al., 2009). More interesting work would be to kernelize our large margin approach and thereby learn non-linear functions. Since our surrogate performs comparable to the state-of-the-art methods, even without using kernels, we expect that we can achieve greatly significant improvements, especially if datasets are not linearly separable.

References

C. Ambroise, T. Denoeux, G. Govaert, , and P. Smets. Learning from an imprecise teacher: Probabilistic and evidential approaches. *Applied Stochastic Models and Data Analysis*, 1:100–105, 2001.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. ISBN 0521833787.

Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: from pairwise approach to listwise approach. In *International Conference on Machine learning 24*, pages 129–136. ACM, 2007.

O. Chapelle, Q. Le, and A. Smola. Large margin optimization of ranking measures. In *NIPS Workshop: Machine Learning for Web Search*, 2007.

O. Chapelle, D. Metzler, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *Conference on Information and Knowledge Management (CIKM)*, 2009.

E. Côme, L. Oukhellou, T. Denoeux, and P. Aknin. Mixture model estimation with soft labels. In *International Conference on Soft Methods in Probability and Statistics*, 2008.

T. Cour, B. Sapp, and B. Taskar. Learning from partial labels. *JMLR*, 12:1501–1536, 2011.

Koby Crammer, Yoram Singer, Nello Cristianini, John Shawe-taylor, and Bob Williamson. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:2001, 2001.

N. Craswell and D. Hawking. Overview of the TREC-2004 Web track. In *Proceedings of the 2004 Text REtreival Conference (TREC)*, 2005.

J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, 28:337–374, 2000.

M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. `http://cvxr.com/cvx`, April 2011.

T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer Verlag, 2001.

W. Hersh, C. Buckley, TJ Leone, and D. Hickam. OHSUMED: an interactive retrieval evaluation and new large test collection for research. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, 1994.

E. Hullermeier and J. Beringer. Learning from ambiguously labeled examples. *Intelligent Data Analysis*, 10(5):419–439, 2006.

K. Järvelin and J. Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 41–48. ACM, 2000.

Rong Jin and Zoubin Ghahramani. Learning with multiple labels. In *NIPS '02*, pages 897–904, 2002.

T. Joachims. Optimizing search engines using clickthrough data. In *Proc. of the 8th ACM SIGKDD international conference on Knowledge discovery and data mining*, page 142, 2002.

Q. Le, A. Smola, O. Chapelle, and C. H. Teo. Optimization of ranking measures. *unpublished*, 2009.

T.Y. Liu. Learning to Rank for Information Retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

T.Y. Liu, J. Xu, T. Qin, W. Xiong, and H. Li. Letor: Benchmark dataset for research on learning to rank for information retrieval. In *Proceedings of SIGIR 2007 Workshop on Learning to Rank for Information Retrieval*, pages 3–10, 2007.

T. Qin, X.-D. Zhang, M.-F. Tsai, D.-S Wang, T.-Y. Liu, and H. Li. Query-level loss functions for information retrieval. *Information processing and management*, 2007.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the twenty-first international conference on Machine learning*, 2004.

I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research (JMLR)*, 6:1453–1484, September 2005.

P. Vannoorenberghe and P. Smets. Partially supervised learning by a credal em approach. In *European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 956–967, 2005.

F. Xia, T.-Y. Liu, J. Wang, W. Zhang, and H. Li. Listwise approach to learning to rank: theory and algorithm. In *International Conference on Machine learning 25*, pages 1192–1199, 2008.

J. Xu and H. Li. Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, page 398. ACM, 2007.