Learning Lipschitz-smooth Utility Functions for Transformation Models in Ranking Problems

Abstract

This paper studies a learning approach to infer a transformation model from observations, a problem which is intrinsically related to learning rankings as we shall point out. Implementing empirical risk minimization in this context results in an algorithm which is similar to the Support Vector Machine (SVM) classifier, now based on a Lipschitz smoothness condition replacing the role of the margin. Theoretically, this view allows one to work in the framework of regularization and Rademacher complexities. Practically, the resulting estimate is given by solving a convex quadratic program in $O(n)$ variables (with n the number of samples). The approach can be extended to learning nonlinear functions using Mercer kernels.

1 Introduction

Methods based on ranking continue to challenge researchers in different scientific areas, see e.g. [5, 10]. Problems of learning ranking functions come in different flavors, including ordinal regression, bipartate ranking and discounted ranking studied frequently in research on information retrieval. This paper is concerned with the general problem where the output domain can be arbitrary (with possibly infinite members), but possess a natural ordering relation between the members. This general problem was studied before in [1, 10], and results can be specified to the aforementioned specific settings by proper definition of the domain of the outputs (e.g. restricting its cardinality to $k < \infty$ or $k = 2$).

A main trend is the reduction of a ranking problem to a pairwise classification problem, bringing in all methodology from learning theory. It may however be argued that such an approach deflects attention from the real nature of the ranking problem. It is for example not clear that the complexity control (in a broad sense) which is successful for classification problems is also natural and efficient in the ranking setting. More specifically, it is often taken for granted that the measure of margin - successful in the setting of binary classification - has a natural counterpart in the ranking setting as the measure of *pairwise margin*, although it remains somewhat arbitrary how this is to be implemented exactly, see e.g. [4]. In order to approach such questions, we take an alternative approach: we will try to learn a single function $u : \mathbb{R}^d \to \mathbb{R}$, such that the natural order on $\mathbb{R}$ induces the desired ranking (approximatively). Such a function is often referred to as a scoring, ranking, utility or health function depending on the context - we will use *utility function* in this text.

Instead of considering the problem of learning a utility function which has the best possible concordance with the observed output values [5], this paper starts from considering the *realizable* case first. Here we consider the case where there exists at least one utility function which ranks the data exactly concordant with the observed output values. The realizable case - the case where the underlying (ranking) function belongs to the hypothesis space of the learning algorithm - is theoretically found to have much better convergence results compared to the general case. Specifically, it is well known that (e.g. for pattern recognition) the realizable case gives a $O(1/n)$ convergence, as opposed to the best possible $O(1/\sqrt{n})$ rate in the agnostic (or general) case (see e.g. [7] and references). This theoretical insight formed also a main motivation to consider first the realizable case in practical algorithms, as the support vector machine (SVM) classifier. There is an alternative interpretation one could give for our emphasis on the realizable case: one could assume that in an ideal, transparent world the predictions of the outputs could be made by a deterministic, mechanistic function governing the process of interest. The approach that will be taken essentially tries to capture these rules, despite the random effects perturbing such a clear view.

In the realizable case, techniques as complexity control, regularization or Occam's razor (in a broad sense) give a

guideline to learn a specific function in case there are more functions exactly concordant with the observed data: a simpler function has a better chance of capturing the underlying relation. In short, we will argue that a utility function reproducing the observed order is less complex than another concordant function if the former is more smoothly related to the actual output values. That is, if there is an exact order relation between two variables, one can obviously find (geometrically) a monotonically increasing function between them. This argument relates ranking directly to what is well-studied in the statistical literature as transformation models, see e.g. [6, 13]. Here the monotonically increasing mapping between utility function and output is referred to as the transformation function. Now, we define the complexity of a prediction rule for transformation models as being the Lipschitz constant of this transformation function. When implementing a risk minimization strategy based on these insights, the resulting methods are similar to the binary SVMs, but do differ conceptually and computationally with existing ranking approaches. Also similar in spirit to the non-separable case in SVMs, it is indicated how slack variables can be used to relax the realizable case: we assume that an exactly concordant function can be found, were it not for incomplete observation of the patients' covariates.

An important motivation is the use of transformation models in the context of survival analysis, i.e. the task of finding models for failure-time data and corresponding covariates. Transformation models were described as generalization of Cox' proportional hazard, proportional odds and accelerated failure time models to name a few [12]. On the other hand, the outlined relationship with ranking connects the reasonings directly to recent advances in machine learning techniques for exactly this setting. Specifically, the use of the *concordance index* [9] extends ranking techniques directly to the case of censored observations, a phenomenon which is especially common in survival data.

The following notational conventions are adopted. Lower case letters denote deterministic quantities as indices, constants or functions. Capital letters represent random variables or random vectors. This paper is organized as follows. Section 2 links ranking with transformation models. In Section 3 we describe the use of a kernel-based learning machine in combination with a Lipschitz-based complexity control. Section 4 gives numerical results illustrating the use and effectiveness of the proposed algorithm.

2 Ranking and Transformation Models

2.1 Transformation Models

Consider the statistical model referred to as a *transformation model*, see e.g. [3, 6, 13].

Definition 1 (Transformation Model) *Let $h : \mathbb{R} \to \mathbb{R}$ be a* strictly *increasing function, and let $u : \mathbb{R}^d \to \mathbb{R}$ be a function of the covariates $X \in \mathbb{R}^d$. Let ϵ be a random variable ('noise') independent of X, with cumulative distribution function $F_\epsilon(e) = P(\epsilon \leq e)$ for any $e \in \mathbb{R}$. A Transformation Model (or TM) takes the following form*

$$h(Y) = u(X) + \epsilon. \tag{1}$$

We can specify a transformation model alternatively as

Proposition 1 *Let $g : [0,1] \to \mathbb{R}$ be an appropriate monotonically increasing function, and let $g^- : \mathbb{R} \to [0,1]$ be defined as $g^-(z) = \sup_{x \in [0,1]}(g(x) \leq z)$ for any z. Then we can specify the transformation model (1) equivalently as*

$$g\big(P(Y \leq y | u(X) = u(x))\big) = h(y) - u(x), \tag{2}$$

where $g^-(z) = F_\epsilon(z)$.

Proof: The proof is classical but gives insight in the model specification. Assume h is strictly monotonically increasing, then one can rewrite $h(Y) = u(X) + \epsilon$ as

$$\begin{aligned} &P\left(h(Y) \leq h(y) \middle| u(X) = u(x)\right) \\ &\quad = P\left(u(X) + \epsilon \leq h(y) \middle| u(X) = u(x)\right) \\ &\quad \Leftrightarrow P(Y \leq y | u(X) = u(x)) \\ &\quad = P\left(\epsilon \leq h(y) - u(x)\right) = F_\epsilon(h(y) - u(x)), \end{aligned} \tag{3}$$

where the last implication follows by the assumption that the random variable ϵ is independent of X.

□

We refer to h or h^- as the transformation function, as made clear by the precise context. Transformation models are central in various (statistical) modeling strategies generalizing linear models. They were studied also under the name of single index models (see e.g. [8] and citations), where different semi-parametric estimation methods were studied based on smoothing techniques. An intrinsic difficulty with this approach is to impose the monotonicity of the nonlinearity [8]. Since transformation models basically relate the distribution of the output variable to the covariates, they are in that sense highly related to generalized linear models (GLMs, [14]). Here the transformation function generalizes the link function which however only relates the expected value of the output. The models do also generalize the most powerful statistical models used in the analysis of survival data, see e.g. [6, 13]. We will argue why they can take a central role in the statistical analysis of ranking problems.

2.2 Prediction and Risk

The key idea of this paper is to relate ranking and transformation functions. In the case where two variables $z_i \in \mathbb{R}$ and $y_i \in \mathbb{R}$ with corresponding index $i = 1, \ldots, n$ have perfectly corresponding rankings (or $(z_i - z_j)(y_i - y_j) \geq 0$ for all $i, j = 1, \ldots, n$), one can find a monotonously increasing function $h^- : \mathbb{R} \to \mathbb{R}$ such that

$$y_i = h^-(z_i), \forall i = 1, \ldots, n. \tag{4}$$

To handle inference of the monotone increasing function h^- without imposing any analytical form whatsoever is to restrict attention to this function only at the points where they evaluate in the samples. Hence, the function h^- is represented by the points $\{(z_i, h^-(z_i)\}_{i=1}^n$. The price for doing so is that the function cannot be evaluated at any new point $z \in \mathbb{R}$. However, by exploiting the monotonicity constraints one can still infer $h^-(z)$ reasonably accurate.

Definition 2 (Specification of the Transformation Function) *Let $-\infty < z_{(1)} \leq \cdots \leq z_{(n)} < \infty$ denote the ordered values of $\{z_i\}_{i=1}^n$, where the indexing $_{(i)}$ reflects that $z_{(i-1)} \leq z_{(i)}$ for all $i = 2, \ldots, n$. If a function $h^- : \mathbb{R} \to \mathbb{R}$ is represented by a set of evaluated points $\{(z_i, h^-(z_i))\}_{i=1}^n \subset \mathbb{R} \times \mathbb{R}$, the function takes a value for a new point $x \in \mathbb{R}$ as*

$$h^-(z_{(i-1)}) \leq h^-(z) \leq h^-(z_{(i)}), \tag{5}$$

where $z_{(i-1)} \leq z \leq z_{(i)}$ for an appropriate index i.

This definition is visualized in figure (1).

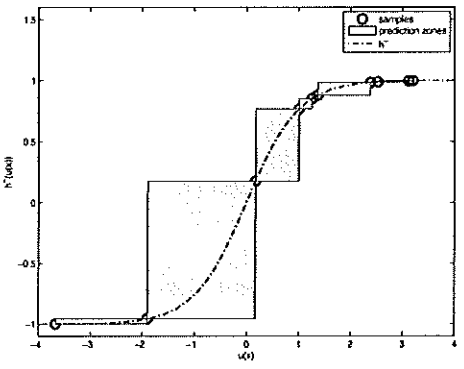

Figure 1: *Visualization of the representation of a monotonically increasing function (dashed lines) through the samples. The yellow zones indicate possible outcomes at places $u(x)$ where the function is not defined explicitly.*

Now, we will derive a matching notion of risk for such a model.

Definition 3 (Risk of (h^-, u)) *The risk one could associate with a monotonically increasing function reflects penalizes discordant samples $u(X)$ and $u(X')$. First, we define the U-kernel c_u for all couples (x, y) and (x', y') as*

$$c_u\left((x, y), (x', y')\right) = I\left[(u(x) - u(x'))(y - y') < 0\right], \tag{6}$$

where $I[z] = 1$ if z is true and 0 otherwise. Then we have the actual risk

$$\mathcal{R}(u; F_{XY}) = P\left((u(X) - u(X'))(Y - Y') < 0\right). \tag{7}$$

The motivation is that this risk does not depend on the transformation function h^-. Its empirical counterpart becomes

$$\mathcal{R}_n(u) = \frac{2}{n(n-1)} \sum_{i<j} I\left[(u(X_i) - u(X_j))(Y_i - Y_j) < 0\right] \tag{8}$$

which is constrained to zero in the realizable case.

Now, this risk criterion corresponds (up to a sign) with the ordinal risk as used in modeling preference relations, or with Kendall's τ.

2.3 Realizable Case

At first, we study the realizable case in some detail as this one does provide faster convergence compared to the general case. Lateron in this subsection, we will illustrate how one can relax the assumption of realizability.

Lemma 1 (Realizable Case) *Consider a finite hypothesis set $|\mathcal{U}| < \infty$. Assume that there exists a function $u \in \mathcal{U}$ with perfect concordance, or the following $n(n-1)$ constraints hold*

$$(u(X_i) - u(X_j))(Y_i - Y_j) \geq 0, \forall i, j = 1, \ldots, n. \tag{9}$$

For any $\delta > 0$, one has for such $u \in \mathcal{U}$ that with probability exceeding $1 - \delta$ that

$$\mathcal{R}_n(u) \leq 2\frac{\ln|\mathcal{U}| - \ln\delta}{n-1}. \tag{10}$$

Proof: This result follows from a simple reduction argument to the classical agnostic case as in [5, 11]. Fix a permutation $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$. Then since the couples $(X_{\pi(i)}, Y_{\pi(i)}); (X_{\pi(n-i)}, Y_{\pi(n-i)})$ are i.i.d., classical results in learning theory [19] state that one has for all $u \in \mathcal{U}$

$$\frac{1}{\lfloor n/2 \rfloor} \sum_{i=1}^{\lfloor n/2 \rfloor} c_u((X_{\pi(i)}, Y_{\pi(i)}), (X_{\pi(n-i)}, Y_{\pi(n-i)})) \leq 2\frac{\ln|\mathcal{U}| - \ln\delta}{n-1}. \tag{11}$$

Now averaging over all possible choices of a permutation π yields the result.

□

Consequently, for a new sample it is unlikely to make too many order violations with the already observed samples.

Corollary 1 (Prediction in the Realizable Case) *Under the previous conditions, the probability of a new sample (X, Y) violating the ordering with respect to the previous samples is bounded as follows. For any $\delta > 0$, one has for such $u \in \mathcal{U}$ that with probability exceeding $1 - \delta$ that*

$$M_n^u(X, Y) \leq 2\frac{\ln|\mathcal{U}| + \ln\frac{2}{\delta}}{n-1} \tag{12}$$

where we define

$$M_n^u(X, Y) = \frac{1}{n}\sum_{i=1}^{n} c_u((X, Y), (X_i, Y_i)). \tag{13}$$

Proof: This result follows from the following equality

$$\begin{aligned}\frac{2}{(n+1)n}&\sum_{i<j=1}^{n+1} c_u((X_i, Y_i), (X_j, Y_j)) \\ &= \frac{2}{n(n-1)}\sum_{i<j=1}^{n} c_u((X_i, Y_i), (X_j, Y_j)) \\ &\quad + \frac{1}{n}\sum_{i=1}^{n} c_u((X_i, Y_i), (X_{n+1}, Y_{n+1})),\end{aligned} \tag{14}$$

where we let $(X_{n+1}, Y_{n+1}) = (X, Y)$. Now we apply the above concentration inequality for all $u \in \mathcal{U}$ twice: once for the term on the left hand side, and once for the first term on the righthand side.

□

This concept is illustrated in Figure 2. This idea appears to be quite powerful: it gives us a means to characterize prediction accuracy on a scale which is induced by the given sample. This should be contrasted with classical data-independent notions of prediction accuracy as the squared error (MSE) or absolute deviation (MAE).

Now Markov' inequality can be used to translate the previous bound on the expectation of a positive random variable into a bound on the probability of an event of interest.

Corollary 2 (Loss to Probability) *Under the previous conditions, the probability of a new sample (X, Y) violating the ordering with respect to the previous samples is bounded as follows. For any $\delta > 0$, one has for such $u \in \mathcal{U}$ that with probability exceeding $1 - \delta$ and for all $t > 0$, that*

$$P\left(M_n^u(X, Y) \geq t\right) \leq \left(\frac{2}{t}\right)\frac{\ln|\mathcal{U}| + \ln\frac{2}{\delta}}{n-1}, \tag{15}$$

since $M_n \geq 0$.

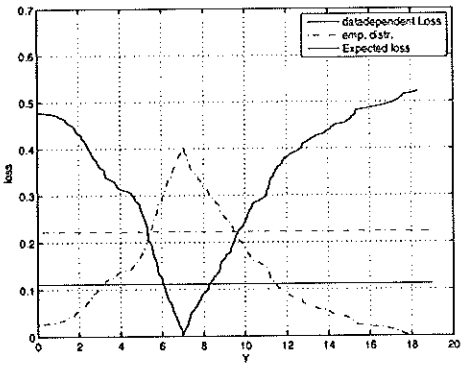

Figure 2: *Illustration of the predictive statements as described in Subsection 2.3. After training an* optimal *utility model, we predict the value of the utility value for a new patient with given covariates $X = x$. One can then compute how many order violations we would have against the observations, if a certain value of Y corresponding to $u(x)$ would come up. The 'V'-shape is then the empirical concordance displayed against such value Y. The solid horizontal line depicts the expected concordance, applied on this value of $u(x)$.*

Note that this results not (yet) in conditional statements in the form 'if we observe $X = x$, the output would be such with high probability', a statement which would be of the form $P\left(M_n^u(X, Y) \geq t \mid X = x\right)$. Instead, it results in a statement on the *average*, if we would make a prediction corresponding to values of X, the predictive rules would be satisfied with high probability. We find this an important observation, as in practice one is rather interested in making an accurate predictive statement for a specific case, rather than predicting well on average. This is especially so if we will specialize to the case of survival data in context of medical analysis. It remains in general an open question how to derive conditional probabilistic statements of the first kind.

These results suggest the following learning strategy for ranking problems. In the realizable case, we have to find $u \in \mathcal{U}$ which gets the orderings of $\{u(X_i)\}_{i=2}^n$ and $\{Y_i\}_{i=2}^n$ exactly similar. In the general case we have to find a $u \in \mathcal{U}$ such that $\{u(X_i)\}_{i=2}^n$ and $\{Y_i\}_{i=2}^n$ are *almost always* concordant. It is still arbitrary how to translate *almost always* in a formal way. Let us introduce a slack variable $\{e_i\}_i$ for each couple $\{(u(X_i), Y_i)\}_{i=1}^n$, and we use the function $\ell : \mathbb{R} \to \mathbb{R}$ to penalize large slack terms. Let the notation $_{(i)}$ re-index the values of $\{(u(X_{(i)}), Y_{(i)})\}_{i=1}^n$ such that $Y_{(i-1)} \leq Y_{(i)}$ for all $i = 2, \ldots, n$. By use of definition 1, let $\mathcal{G}^+ \subset \{h : \mathbb{R} \to \mathbb{R}\}$ which are monotonically increasing. Then a first attempt to write the learning problem

is

$$\min_{u\in\mathcal{U},h\in\mathcal{G}^+}\sum_{i=1}^{n}\ell(u(X_i)-h(Y_i)) \tag{16}$$

In general however, a trivial solution solves (16), namely $u(x)=c$ for any constant $c\in\mathbb{R}$. The following subsection outlines how to avoid this problem by using an appropriate form of complexity control based on Lipschitz-smoothness.

2.4 Managing Infinite Hypothesis Spaces

At this stage, we suggest a formal way to handle infinite hypothesis spaces where $|\mathcal{U}|=\infty$. Learning theory (see e.g. [7, 19] and citations), then learns us that such spaces can be measured in relation to the task at hand by considering appropriate complexity measures. Specifically, it is indicated how apriori (qualitative) assumptions can be translated into formal probabilistic statements that the empirical loss will approach the actual loss (a criterion depending on the underlying P). A convenient way to capture the complexity of a hypothesis space $\mathcal{U}$ is the Rademacher complexity, as employed in [2, 16] and others. The following proposition gives the main argument applied on the risk minimization strategy (16).

Proposition 2 (Rademacher Bound) *Let $\delta>0$ and let $D_n=\{(X_i,Y_i)\}_{i=1}^n$ be an i.i.d. sample from F_{XY}, and let $\delta>0$. Then one has for any $u\in\mathcal{U}$ that with probability exceeding $1-\delta$ that*

$$\mathcal{R}(u;F_{XY})\leq\mathcal{R}_n(u)+\mathrm{R}_n(\mathcal{U},\mathcal{G}^+)+\sqrt{\frac{-8\ln(\delta/2)}{n}} \tag{17}$$

where the Rademacher complexity $\mathrm{R}_n(\mathcal{U})$ equals

$$\mathrm{R}_n(\mathcal{U},\mathcal{G}^+)=E\left[\sup_{u\in\mathcal{U},h\in\mathcal{G}^+}\frac{1}{n}\left|\sum_{i=1}^{n}\sigma_i\ell(u(X_i)-h(Y_i))\right|\mid D_n\right] \tag{18}$$

where $\{\sigma_i\}$ are defined as binomial random variables with $P(\sigma_i=1)=P(\sigma_i=-1)$.

So the goal is now to find a practical mechanism bounding the Rademacher complexity as much as possible, and is *natural* in the sense that functions in a lower complex function class occur relatively more often in practice than functions in a more complex $\mathcal{U}$. The discussion whether such a definition of complexity is natural indeed, is necessarily a case of trial and error, but we claim that the following definition is a good candidate.

Definition 4 (Lipschitz Smooth) *Let $h^-:\mathbb{R}\to\mathbb{R}$ be a monotonically increasing function, and let $u:\mathbb{R}^d\to\mathbb{R}$ be an utility function. Then we say that the function $h^-\circ u$ is first order Lipschitz with constant L if for all $x,x'\in\mathbb{R}^d$ one has*

$$\left|h^-(u(x))-h^-(u(x'))\right|\leq L\left\|x-x'\right\|_2. \tag{19}$$

Consequently, if we consider the class of utility functions $\mathcal{U}^1=\left\{u(x)=w^Tx,\|w\|_2=1\right\}$, for $u\in\mathcal{U}^1$ and a monotonically increasing $h^-:\mathbb{R}\to\mathbb{R}$, we can write equivalently that the Lipschitz property holds for a constant $L\in\mathbb{R}$ if for all $x,x'\in\mathbb{R}^d$ one has

$$\left|h^-(u(x))-h^-(u(x'))\right|\leq L\left\|u(x)-u(x')\right\|_2 \tag{20}$$

after a change of variables (i.e. $v=Lw$), one can write equivalently

$$\left|h^-(u(x))-h^-(u(x'))\right|\leq\left\|v^Tx-v^Tx'\right\|_2,\ \forall x,x'\in\mathbb{R}^d, \tag{21}$$

where $v^Tv\leq L$ and

$$\mathcal{U}^{L,R}=\left\{u:u(x)=w^Tx,w^Tv\leq L,\|x\|_2\leq R\right\}. \tag{22}$$

Figure 3 represents a monotonically increasing function with a Lipschitz constant equal to 1. The final result needs the following proposition as found by specializing Lemma 22 in [2].

Proposition 3 (Lemma 22 in [2]) *Under the above definitions*

$$\mathrm{R}_n\left(\mathcal{U}^{L,R}\right)\leq\frac{2LR}{\sqrt{n}}. \tag{23}$$

From the following inequality which holds for any $x\in\mathbb{R}^d$ and $y\in\mathbb{R}$, and h such that its inverse h^- exists and is Lipschitz with constant $L>0$:

$$|u(x)-h(y)|\leq L\left|h^-(u(x))-y\right|. \tag{24}$$

In summary and with the help of [2], section 3.1, one has

Lemma 2 (Complexity for Lipschitz Smooth functions) *Consider the loss $\ell(e)=|e|$. Let $\mathcal{G}^{+L}\subset\mathcal{G}^+$ for $L\geq0$ where $h\in\mathcal{G}^{+L}$ if h^- is first order Lipschitz with constant L.*

$$\mathrm{R}_n(\mathcal{U}^{1,B},\mathcal{G}^{+L})\leq\frac{8BL^2}{\sqrt{n}} \tag{25}$$

where $\{\sigma_i\}$ are defined as binomial random variables with $P(\sigma_i=1)=P(\sigma_i=-1)$.

This result supports the central role the constant L can play in a learning algorithm for learning rankings. Observe that the bound is highly similar to the celebrated radius-margin bound for binary classifiers [19].

2.5 Pairwise Margin Approach

Consider at first the naive implementation of empirical risk minimization strategy

$$u_n=\arg\min_{u\in\mathcal{U}}\mathcal{R}_n(u), \tag{26}$$

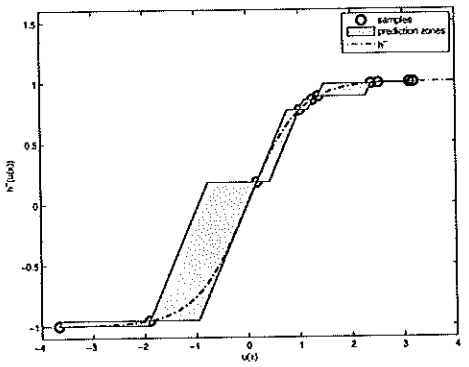

Figure 3: *Visualization of the representation of a monotonically increasing function (dashed lines) satisfying a Lipschitz constraint with constant $L = 1$ through the samples. The yellow zones indicate possible outcomes at places $u(x)$ where the function is not defined explicitly. Observe that the zones of uncertainty (yellow parallelograms) describe a smaller surface than would be the case in the non-constrained analysis as given in Figure (1).*

where $\mathcal{U} \subset \{u\}$ is an appropriate hypothesis set, containing all possible utility functions which are *apriori* realistic. This idea was elaborated before in [18], but obvious computational issues arise as one has to consider all pairs of observations. Also the precise statistical model is not entirely clear, and most of all it is not apparent what a natural notion of *simplicity* or regularization would be. These are precisely the questions which are addressed in this manuscript. A first technical result is given in the following result based on [5].

Proposition 4 (Rademacher Bound [5]) *Let $\delta > 0$ and let $D_n = \{(X_i, Y_i)\}_{i=1}^n$ be an i.i.d. sample from F_{XY}, and let $\delta > 0$. Then one has for any $u \in \mathcal{U}$ that with probability exceeding $1 - \delta$ that*

$$\inf_{u \in \mathcal{U}} \mathcal{R}(u_n; F_{XY}) \leq \inf_{u \in \mathcal{U}} \mathcal{R}(u; F_{XY}) + \mathbf{R}_n{}'(\mathcal{U}) + 4\sqrt{\frac{-\ln(\delta)}{n-1}} \tag{27}$$

where the Rademacher complexity $\mathbf{R}_n{}'(\mathcal{U})$ of the function-class $\mathcal{U}$ is defined as

$$\mathbf{R}_n{}'(\mathcal{U}) = E\left[\sup_{u \in \mathcal{U}} \frac{4}{\lfloor \frac{n}{2} \rfloor} \left| \sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} \sigma_i c_u\left((X_i, Y_i), (X_j, Y_j)\right) \right| \mid D_n \right] \tag{28}$$

where $\{\sigma_i\}$ are defined as binomial random variables with $P(\sigma_i = 1) = P(\sigma_i = -1)$, and as before $c_u\left((X_i, Y_i), (X_j, Y_j)\right) = I\left((u(X_i) - u(X_j))(Y_i - Y_j) < 0\right)$ for all $i, j = 1, \ldots, n$.

The paper [5] improves on this result by using the Hajek projection of U-statistics, but since the order of the involved quantities remains the same, this result is sufficient in the current context.

Remark that this line of results is essentially different from the proposed approach in Subsection 2.4. At first, the difference of the theoretical nature is relected by the difference of the definition of the Rademacher complexities $\mathbf{R}_n(\mathcal{U})$ (eq. 18) and $\mathbf{R}_n{}'(\mathcal{U})$ (eq. 28). The disadvantage of the latter is the non-standard form: past literature describes more structural results and specifications to algorithms [2] than the latter, introduced in [5]. As mentioned already, the focus of the realizable case gives faster rates, and suggests that interpolation between the $O(1/n)$ and $O(1/\sqrt{n})$ regime is possible by imposing natural conditions, similar to Tsybakov's noise condition in the pattern recognition case.

3 MINLIP: A Kernel-based Learning Machine

3.1 Realizable Case

At this point we compile the above outlined reasonings into an (computationally) effective learning scheme. We implement the scheme of (16), in combination with the Lipschitz-based complexity control. The Lipschitz-based complexity control is also in practice a useful tool as it makes the problem (16) well-defined. Consider again first the realizable case, where a function $u \in \mathcal{U}$ exists which reconstructs the ordering between the observations exactly. Since if there is at least one such nontrivial function there are necessarily an infinite number of them, it becomes clear that one needs an additional mechanism to have the learning scheme make a choice between those. Exploiting the model specification, a plausible approach would be to look for a function u and h^- where h^- is as 'regular' as possible. In particular, we propose to minimize the Lipschitz constant $L > 0$ in (16) and (19).

We consider models with linear utility functions as follows

$$h(Y) = u(X) = w^T X. \tag{29}$$

Extension towards non-linear utility functions using Mercer kernels. Since the function $u(x) = w^T x$ can be arbitrary rescaled yielding any Lipschitz constant, we fix the norm $w^T w$ and try to find $u(x) = v^T x$ with $v^T v = 1$. Moreover, the function h^- is only learned explicitly at the points $\{(u(X_i))\}_{i=1}^n$ and the problem of minimizing the Lipschitz constant of h^- can be written as

$$\min_{v,L} L^2 \text{ s.t. } v^T v = 1, \quad |Y_i - Y_j| \leq L\left|v^T X_i - v^T X_j\right|, \ \forall i < j. \tag{30}$$

and equivalently substituting $w = Lv$ as

$$\min_w \frac{1}{2} w^T w \quad \text{s.t. } |Y_i - Y_j| \leq \left| w^T X_i - w^T X_j \right|, \ \forall i < j. \quad (31)$$

At this point, it is seen that the $O(n^2)$ constraints can be replaced by $n-1$ constraints by ordering (and reindexing) the values Y such that $Y_1 \leq \cdots \leq Y_n$. This gives finally the following convex problem

$$\min_w \frac{1}{2} w^T w \quad \text{s.t. } w^T (X_i - X_{i-1}) \geq (Y_i - Y_{i-1}), \ \forall i \geq 2, \quad (32)$$

which goes along similar lines as the hard margin SVM (see e.g.[16]). Remark that there is no need for an intercept term here. Let $\Phi \in \mathbb{R}^{n \times d}$ denote the matrix containing all possible values of X_i for all $i = 1, \ldots, n$. Then we can write the learning problem (32) concisely as

$$\min_w \frac{1}{2} w^T w \quad \text{s.t. } D(\Phi w) \geq DY, \quad (33)$$

with the matrix $D \in \{-1, 0, 1\}^{(n-1) \times n}$ defined as $D_i \Phi = X_i - X_{i-1}$ and $D_i Y = Y_i - Y_{i-1}$ for all rows $i = 2, \ldots, n$. This problem can be solved efficiently as a convex Quadratic Programming (QP) problem. The Lagrange dual problem becomes

$$\min_\alpha \frac{1}{2} \alpha^T (DKD^T) \alpha - \alpha^T DY \quad \text{s.t. } \alpha \geq 0_{n-1} \quad (34)$$

where the kernel matrix $K \in \mathbb{R}^{n \times n}$ contains the kernel evaluations such that $K_{ij} = X_i^T X_j$ for all $i, j = 1, \ldots, n$. The estimated $\hat{u}$ can be evaluated at any point $x \in \mathbb{R}^d$ as

$$\hat{u}(x) = \hat{\alpha}^T D K_n(x), \quad (35)$$

where $\hat{\alpha}$ solves (34), and $K_n(x) = K_n(x) = (K(X_1, x), \ldots, K(X_n, x))^T \in \mathbb{R}^n$

Remark that in view of Subsection 2.5, an alternative approach for constructing a learning machine would be to maximize the pairwise margin (i.e. $\frac{w^T(X_i - X_j)}{w^T w}$ for all $i < j = 1, \ldots, n$). This mechanism was implemented in ranking SVMs [10], and will be compared empirically in Section 4.

3.2 The Agnostic Case

The agnostic case deals with the case where one is not prepared to make the assumption that a function exists which will extract in all cases the most relevant element. To model this, we impute a random variable e with expected value zero, which acts additive on the contribution of the covariates (hence nonadditive on the final output for general function h^-). Hence our model becomes

$$h(Y) = u(X) + e = w^T \varphi(X) + e. \quad (36)$$

Now we suggest how one can integrate the agnostic learning scheme given in (16) with the Lipschitz-based complexity control. We will further specify the loss function $\ell : \mathbb{R} \to \mathbb{R}$ to the absolute value loss, or $\ell(e) = |e|$. The reason for doing so is threefold. At first, this loss function is known to be more robust to model misspecification and outliers (leverage points) than e.g. the squared loss $\ell(e) = e^2$. Secondly, this loss will result in sparse terms, i.e. many of the estimated error terms will be zero. Thirdly, the one-norm loss is found to perform well in the binary classification case as implemented in the SVMs [16]. However, we stress that the choice of this loss is in some sense arbitrary, and should be tailored to the case study at hand.

One can formalize the learning objective for a fixed value of $\gamma > 0$ with errors $e = (e_1, \ldots, e_{n-1})^T \in \mathbb{R}^{n-1}$:

$$\min_{w,e} \frac{1}{2} w^T w + \gamma \|e\|_1 \quad \text{s.t. } D(\Phi w + e) \geq DY, \quad (37)$$

where $\|e\|_1 = \sum_{i=1}^n |e_i|$. A standard argument ([16, 17, 19]) indicates the dual problem formulation

$$\min_\alpha \frac{1}{2} \alpha^T (DKD^T) \alpha - \alpha^T DY \quad \text{s.t.} \quad \begin{cases} -\gamma 1_n \leq D^T \alpha \leq \gamma 1_n \\ \alpha \geq 0_{n-1}, \end{cases} \quad (38)$$

with K as above and the resulting estimate can be evaluated as in (35) without computing explicitly $\hat{w}$.

4 Case Study: Wisconsin Breast Cancer

We report on some numerical results of application of the proposed learning scheme on the Wisconsin Breast Cancer data set[1]. Here, each of the 198 record represents follow-up data for one breast cancer case. These include only those cases exhibiting invasive breast cancer and no evidence of distant metastases at the time of diagnosis. The outcome consist of time of recurrence (if there is recurrence) or time of right-censoring (if no event was observed). 30 features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. We use this dataset to illustrate our methods, and focus for now on jointly modeling the reported times of either event or censoring. The output of our learned model will then be the predicted time to event/censoring. We report numerical accuracies as (1) Kendall's tau between $\{\hat{u}(X_j), Y_j)\}_j$ based on data in a separate test set. (2) The number of discordant pairs $\{\hat{u}(X_j), Y_j)\}_j$ in a test set, compared to $\{\hat{u}(X_j), Y_j)\}_{i=1}^n$ of the training set. The latter expresses the average performance of M_n^u in equation (13). We compare the performance of linear results obtained by **(lsq)** minimizing a least squares criterion. **(lsq_t)** minimizing a least squares criterion after appropriately transforming the outputs (using a rank-transform) [15], **(rankSVM)** rankSVM based on maximizing the pairwise margin. **(minlip)** minlip as

[1] Data available on http://archive.ics.uci.edu/

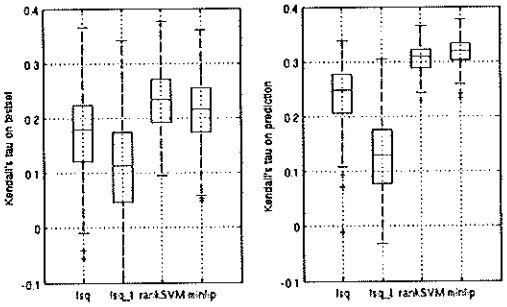

Figure 4: *Numerical results of the study on the Wisconsin Breast Cancer data ($n = 198$, $d = 32$), 100 random splits between training and test set. The left panel shows boxplots of the performance expressed in Kendall's $\tau \in [-1, 1]$. Here rankSVM slightly outperforms minlip. The right panel shows boxplots of the performance expressed in the number of disconcodant pairs in the testset w.r.t the trainingset, as studied in eq. (13). Minlip - especially tuned to this criterion - outperforms the other methods.*

proposed in equation (37). Tuning of the regularization parameter C in (rankSVM) and (minlip) was performed using 10-fold cross-validation. The above method is found to outperform the methods (lsq), (lsq_t) and (rankSVM) in case of the criterion of (13).

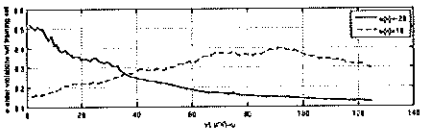

Figure 5: *Probabilities of event Y for two patients, one with prognostic index $\hat{u}(x) = -20$ (dashed line), and the other with $\hat{u}(x) = 10$. The latter is more probale to have a larger response than the first patient.*

5 Conclusions

This paper outlined a general approach to relate the ranking learning problem to a statistical model, the transformation model. The consequence of this symbioses is the definition of a complexity control mechanism, motivated by a Rademacher complexity analysis. A practical learning scheme immediately follows, which - opposed to existing techniques - is given by solving a convex Quadratic Programming problem with $O(n)$ constraints. It is indicated how the linear techniques can be extended to the use of dual Mercer kernels.

References

[1] S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, and D. Roth. Generalization bounds for the area under the ROC curve. *Journal of Machine Learning Research*, 6:393–425, 2005.

[2] P.L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

[3] S.C. Cheng, L.J. Wei, and Z. Ying. Predicting Survival Probabilities with Semiparametric Transformation Models. *Journal of the American Statistical Association*, 92(437), 1997.

[4] W. Chu and S.S. Keerthi. New approaches to support vector ordinal regression. *ICML*, pages 145–152, 2005.

[5] S. Clémençon, G. Lugosi, and N. Vayatis. Ranking and Scoring Using Empirical Risk Minimization. *Learning Theory: 18th Annual Conference on Learning Theory, COLT 2005, Bertinoro, Italy, June 27-30: Proceedings*, 2005.

[6] D.M. Dabrowska and K.A. Doksum. Partial likelihood in transformation models with censored data. *Scandinavian journal of statistics*, 15(1):1–23, 1988.

[7] L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag, 1996.

[8] W, Hardle, P. Hall, and H. Ichimura Optimal Smoothing in Single-Index Models. *Ann. Statist.*, 21(1): 157-178, 1993.

[9] Jr.F.E. Harrell, K.L. Klee, R.M. Califf, D.B. Pryor, and R.A. Rosati. Regression modeling strategies for improved prognostic prediction. *Statistics in Medicine*, 95(9):634–635, 1984.

[10] R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. *Advances in Large Margin Classifiers*, pages 115–132, 2000. MIT Press, Cambridge, MA.

[11] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[12] J.D. Kalbfleisch and R.L. Prentice. *The Statistical Analysis of Failure Time Data*. Wiley series in probability and statistics. Wiley, 2002.

[13] R. Koenker and O. Geling. Reappraising Medfly Longevity: A Quantile Regression Survival Analysis. *Journal of the American Statistical Association*, 96(454), 2001.

[14] P. McCullagh and J.A. Melder. *Generalized Linear Models* Chapman and Hall, 1989.

[15] K. Pelckmans and J.A.K. Suykens and B. De Moor. *A Risk Minimization Principle for a Class of Parzen Estimators*. Advances in Neural Information Processing Systems 20. MIT Press, 2008.

[16] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[17] J.A.K. Suykens, T. van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[18] V. Van Belle, K. Pelckmans, J.A.K. Suykens and S. Vanhuffel. Support Vector Machines for Survival Analysis. *Third International Conference on Computational Intelligence in Medicine and Healthcare, CIMED, 2007, Plymouth, UK, July 25-27: Proceedings*, 2007.

[19] V.N. Vapnik. *Statistical Learning Theory*. Wiley and Sons, 1998.