Tree-based ranking methods

Stéphan Clémençon, Nicolas Vayatis

Tree-based ranking methods

Stéphan Clémençon
Telecom Paristech (TSI) - LTCI UMR Institut Telecom/CNRS 5141
stephan.clemencon@telecom-paristech.fr
Nicolas Vayatis
ENS Cachan & UniverSud - CMLA UMR CNRS 8536
nicolas.vayatis@cmla.ens-cachan.fr

***Abstract*—Recursive partitioning methods are among the most popular techniques in machine learning. The paper investigates how these methods can be adapted to the *bipartite ranking problem*. In ranking, the pursued goal is *global*: based on past data, define an order on the whole input space $\mathcal{X}$, so that positive instances take up the top ranks with maximum probability. The most natural way to order all instances consists of projecting the input data x onto the real line through a real-valued *scoring function* s and use the natural order on $\mathbb{R}$. The accuracy of the ordering induced by a candidate s is classically measured in terms of the ROC curve or the area under the ROC curve (AUC). Here we discuss the design of tree-structured scoring functions obtained by recursively maximizing the AUC criterion. The connection with recursive piecewise linear approximation of the optimal ROC curve both in the L_1-sense and in the L_∞-sense is highlighted. A novel tree-based algorithm, called TREERANK, specifically designed for learning to rank/order instances is proposed. Consistency results and generalization bounds of functional nature are established for this ranking method, when considering either the L_1 or L_∞ distance. Inspired from recent developments in the field of binary classification, we also describe committee-based learning procedures using TREERANK as a "base ranker", in order to overcome obvious drawbacks of such a top-down partitioning technique. Preliminary simulation results are also displayed.**

***Index Terms*—Bipartite Ranking Problem, ROC Curve, AUC Criterion, Decision Tree, Adaptive Piecewise Linear Approximation.**

I. INTRODUCTION

The statistical ranking problem can broadly be considered as the problem of ordering instances from an abstract space $\mathcal{X}$, a high-dimensional Euclidean space typically. This question arises in a large variety of applications, ranging from the design of search engines in information retrieval to medical diagnosis or credit-risk screening. A natural approach consists of "projecting" these instances onto the real line through some real-valued scoring function. Such a function would allow to rank any list of instances in the initial space. Depending on the available information, various approaches can be developed. For instance, both preference learning ([1], [2], [3]) and ordinal regression ([4], [5]) deal with statistical ranking but under different label information. We focus here on the setup where a binary label characterizing each instance is given. This problem is known as the *bipartite ranking problem* ([6], [7], [8]). The calibration of ranking rules can be performed in various ways. In scoring applications, the vast majority of ranking methods are mostly in the spirit of logistic regression and rely on the statistical modeling of the regression function using additive models ([9]). The statistical learning approach is different insofar as it avoids the difficult problem of estimating the distribution in high dimensions and focuses on prediction. Statistical learning strategies can be thought as the optimization of performance measures based on data. In the case of bipartite ranking, the development of the statistical learning approach is involved with AUC maximization. Indeed, a standard performance measure for a scoring function in the presence of classification data is the Receiver Operating Characteristic (ROC) curve, together with the Area Under the ROC Curve, known as the AUC (see [10], [11], [12], [13]). But, since their introduction, ROC curves and the AUC used to serve mostly for validation and not as the basis for optimization principles. More recently, several aspects of AUC maximization have been discussed in the machine learning literature ([14], [15], [16]) and also from a statistical learning perspective ([7], [8], [17]). A particular class of learning algorithms will be at the center of the present paper, namely decision trees in the spirit of CART for classification or regression [18]. The investigation of decision trees in the context of ranking was initiated only recently in the field of machine learning ([19], [20], [21]). The main difficulty relies in the *global* nature of the ranking problem, whereas, in contradistinction, popular classification rules such as those obtained through recursive partitioning of the input space $\mathcal{X}$ are based on the concept of *local learning*. Indeed, for such classification procedures, the predicted label of a given instance $x \in \mathcal{X}$ depends on the data lying in the subregion of the partition containing x solely, while the notion of ranking/ordering would rather involve comparing the subregions to each other.

In this paper, a specific recursive partitioning method (RP) producing piecewise constant scoring functions is proposed and thoroughly investigated. In this approach, alike the RP, the related ordering is *tree-structured*, in a way that (predicted) ranks may be "read from the left to the right" at the bottom of the resulting tree (all instances belonging to the same subregion of the partition having the same rank). This simple top-down algorithm, named TREERANK, may be interpreted as the statistical counterpart of an adaptive and recursive piecewise linear approximation procedure of the optimal ROC curve, in the spirit of finite element methods (FEM). From this angle, the problem of recovering the optimal ROC curve from the perspective of *approximation theory* and the one of

adaptively building a scoring function from training data with a ROC curve close to the approximate version of the optimal one can be addressed simultaneously. As the ROC curve provides a performance measure of functional nature, the approximation can be conceived in a variety of ways depending on the topology equipping the space of ROC curves. Here we shall consider two essential cases: convergence to the optimal ROC curve in the sense of the L_1-metric, which is related to the AUC criterion, but also in a stronger sense carried by the L_∞-distance. In this respect, TREERANK is shown *consistent* (meaning that the ROC curve of the scoring function output by the learning algorithm converges to the optimal one as the training sample size goes to infinity with probability one) for both metrics and *generalization bounds* are established in this functional setup.

It is clear that the top-down strategy of the TREERANK algorithm is very rigid due to its hierarchical nature and shares common drawbacks with classification tree methodologies such as CART. An error in the ordering induced by a certain split will be automatically propagated down to all of the subsequent orderings. Whereas the classification task is local, instability is strongly emphasized by the global nature of the ranking goal: modifying the rank of a given $x \in \mathcal{X}$ may indeed affect the rank of many other instances. Several extensions to the TREERANK approach are thus considered, with the goal of either enhancing the ranking produced by a single tree, or else "combining" many ranking trees in order to improve the overall performance. A preliminary simulation study has also been carried out, aiming essentially at illustrating the practical implementation of these methods.

The article is structured as follows. In Section II, we present a general approach for assessing optimality in the bipartite ranking problem. We also recall the main concepts and discuss the issue of AUC maximization. In Section III, we relate linear-by-parts approximations of the optimal ROC curve to finite-dimensional (piecewise constant) approximations of optimal scoring functions and provide an adaptive tree-structured recursive procedure for which an approximation error result is established. This approximation scheme can be carried out over empirical data by the means of the TREERANK algorithm described in Section IV. The statistical consistency of the method is also investigated and rate bounds are proved. Section V presents various extensions improving the original TREERANK methodology and section VI reports illustrating empirical results. All proofs are postponed to the Appendix section.

II. THE NATURE OF THE RANKING PROBLEM

We start off by describing the optimal elements for the bipartite ranking problem ([6]). The use of the ROC curve as a performance measure for bipartite ranking is then strongly advocated by this enlightening approach, under which the problem boils down to recovering the collection of *level sets* of the regression function.

A. Setup and goal of ranking.

We study the ranking problem for classification data with binary labels. This is also known as the bipartite ranking problem. The data are assumed to be generated as copies of a random pair $(X, Y) \in \mathcal{X} \times \{-1, +1\}$ where X is a random descriptor living in the measurable space $\mathcal{X}$ and Y represents its binary label (relevant vs. irrelevant, healthy vs. sick, ...). We denote by $P = (\mu, \eta)$ the distribution of (X, Y), where μ is the marginal distribution of X and η is the *regression function* (up to an affine transformation): $\eta(x) = \mathbb{P}\{Y = 1 \mid X = x\}$, $x \in \mathcal{X}$. We will also denote by $p = \mathbb{P}\{Y = 1\}$ the proportion of positive labels. In the sequel, we assume that the distribution μ is absolutely continuous with respect to Lebesgue measure.

The goal of a ranking procedure is to provide an ordering of the elements of $\mathcal{X}$ based on their labels. We expect to end up with a list with positive labels at the top and negative labels at the bottom. However, label information does not permit to derive a total order on $\mathcal{X}$ and among relevant (positively labeled) objects in $\mathcal{X}$, some might be more relevant than others. In short, a good ranking should preserve the ordering induced by the likelihood of having a positive label, namely the regression function η. We consider the approach where the ordering can be derived by the means of a *scoring function* $s : \mathcal{X} \to \mathbb{R}$: one expects that the higher the value $s(X)$ is, the more likely the event "$Y = +1$" should be observed. The following definition sets the goal of learning methods in the setup of bipartite ranking.

Definition 1 (Optimal scoring functions)**.** *A scoring function $s^* : \mathcal{X} \to \mathbb{R}$ is said to be optimal if it induces the same ordering over $\mathcal{X}$ as the function $\eta(x) = \mathbb{P}\{Y = 1 \mid X = x\}$, $\forall x \in \mathcal{X}$. In other words:*

$$\forall x, x' \in \mathcal{X} \ , \quad s^*(x) - s^*(x') > 0 \Rightarrow \eta(x) - \eta(x') > 0 \ .$$

According to the previous definition, the next proposition is a trivial characterization of the class of optimal scoring functions.

Proposition 2. *The class of optimal scoring functions is given by the set*

$$\mathcal{S}^* = \{ \ s^* = T \circ \eta \ \mid \ T : [0, 1] \to \mathbb{R} \text{ strictly increasing } \}.$$

Interestingly, it is possible to make the connection between an arbitrary (bounded) optimal scoring function $s^* \in \mathcal{S}^*$ and the distribution P (through the regression function η) completely explicit.

Proposition 3 (Optimal scoring functions representation)**.** *A bounded scoring function s^* is optimal if and only if there exist a nonnegative integrable function w and a continuous random variable V in $(0, 1)$ such that:*

$$\forall x \in \mathcal{X} \ , \quad s^*(x) = \inf_{\mathcal{X}} s^* + \mathbb{E}\left(w(V) \cdot \mathbb{I}\{\eta(x) > V\}\right)$$

Remark 1. In the case of the regression function η, we have the following identity :

$$\forall x \in \mathcal{X} \ , \quad \eta(x) = \mathbb{E}\left(w(U)\mathbb{I}\{\eta(x) > U\}\right)$$

where U is a uniform random variable on $[0, 1]$ and the function w is the indicator of the support of the random variable $\eta(X)$.

A crucial consequence of the last proposition is that solving the bipartite ranking problem amounts to recovering the collection $\{x \in \mathcal{X} \mid \eta(x) > u\}_{u\in(0,1)}$ of level sets of the regression function η. Hence, the bipartite ranking problem can be seen as a collection of *overlaid classification problems*. This view was first introduced in [22]. Moreover, the representation of optimal scoring functions provides the intuition for the approximation procedure of Section III and the subsequent TREERANK algorithm of Section IV. By checking the proof of the Proposition, it looks like the weight function w only plays the role of a scaling function. However, the general representation may suggest various estimations schemes of the Monte-Carlo type in order to recover optimal scoring functions.

B. (True) ROC curves

We now recall the concept of ROC curve and explain why it is a natural choice of performance measure for the ranking problem with classification data. In this section, we only consider *true* ROC curves which correspond to the situation where the underlying distribution is known.

Before recalling the definition, we need to introduce some notations. For a given scoring rule s, the conditional cdfs of the random variable $s(X)$ are denoted by G_s and H_s. We also set, for all $z \in \mathbb{R}$:

$$\bar{G}_s(z) = 1 - G_s(z) = \mathbb{P}\{s(X) > z \mid Y = +1\} \ ,$$
$$\bar{H}_s(z) = 1 - H_s(z) = \mathbb{P}\{s(X) > z \mid Y = -1\} \ .$$

to be the residual conditional cdfs of the random variable $s(X)$. When $s = \eta$, we shall denote the previous functions by G^*, H^*, $\bar{G}^*$, $\bar{H}^*$ respectively. We will also use the notation, for all t:

$$\alpha(t) = \bar{H}^*(t) = \mathbb{P}\{\eta(X) > t \mid Y = -1\} \ ,$$
$$\beta(t) = \bar{G}^*(t) = \mathbb{P}\{\eta(X) > t \mid Y = 1\} \ .$$

We introduce the notation $Q(Z, \alpha)$ to denote the quantile of order $1 - \alpha$ for the distribution of a random variable Z conditioned on the event $Y = -1$. In particular, the following quantile will be of interest:

$$Q^*(\alpha) = Q(\eta(X), \alpha) = \bar{H}^{*-1}(\alpha) \ ,$$

where we have used here the notion of generalized inverse F^{-1} of a càdlàg function F:

$$F^{-1}(z) = \inf\{t \in \mathbb{R} \mid F(t) \geq z\} \ .$$

A classical way to assess the performance of a scoring function s in separating the two populations (positive vs. negative labels) is the *Receiver Operating Characteristic* known as the ROC curve ([11], [12]).

Definition 4 (True ROC curve)**.** *The* ROC *curve of a scoring function s is the parametric curve:*

$$z \mapsto \left(\bar{H}_s(z), \bar{G}_s(z)\right)$$

for thresholds $z \in \mathbb{R}$. It can also be defined as the plot of the function:

$$\alpha \in [0, 1] \mapsto \bar{G}_s \circ \bar{H}_s^{-1}(\alpha) = \bar{G}_s\left(Q(s(X), \alpha)\right) = \mathrm{ROC}(\mathrm{s}, \alpha).$$

By convention, points of the curve corresponding to possible jumps (due to possible degenerate points of H_s or G_s) are connected by line segments, in order that the ROC *curve is always continuous.*
For $s = \eta$, we take the notation $\mathrm{ROC}^*(\alpha) = \mathrm{ROC}(\eta, \alpha)$.

The residual cdf $\bar{G}_s$ is also called the *true positive rate* while $\bar{H}_s$ is the *false positive rate*, so that the ROC curve is the plot of the true positive rate against the false positive rate. Basic properties of ROC curves can be found in the Appendix A.

The ROC curve provides a visual tool for comparing the ranking performance of two scoring rules.

Definition 5. *Consider two scoring functions s_1 and s_2. We say that s_1 provides a better ranking than s_2 when:*

$$\forall \alpha \in (0, 1) \ , \quad \mathrm{ROC}(\mathrm{s}_1, \alpha) \geq \mathrm{ROC}(\mathrm{s}_2, \alpha) \ .$$

Remark 2. (GLOBAL VS. LOCAL PERFORMANCE.) Note that, as a functional criterion, the ROC curve induces a partial order over the space of all scoring functions. Some scoring function might provide a better ranking on some part of the observation space and a worst one on some other. A natural step to take is to consider local properties of the ROC curve in order to focus on best instances but this is not straightforward as explained in [22].

Therefore, we expect optimal scoring functions to be those for which the ROC curve dominates all the others for all $\alpha \in (0, 1)$. The next proposition highlights the fact that the ROC curve is relevant when evaluating performance in the bipartite ranking problem.

Proposition 6. *The class $\mathcal{S}^*$ of optimal scoring functions provides the best possible ranking with respect to the* ROC *curve. Indeed, for any scoring function s, we have:*

$$\forall \alpha \in (0, 1) \ , \quad \mathrm{ROC}^*(\alpha) \geq \mathrm{ROC}(\mathrm{s}, \alpha) \ ,$$

and

$$\forall s^* \in \mathcal{S}^* \ , \ \forall \alpha \in (0, 1) \ , \quad \mathrm{ROC}(\mathrm{s}^*, \alpha) = \mathrm{ROC}^*(\alpha) \ .$$

Moreover, if we set the notations:

$$R_\alpha^* = \{x \in \mathcal{X} \mid \eta(x) > Q^*(\alpha)\}$$
$$R_{s,\alpha} = \{x \in \mathcal{X} \mid s(x) > Q(s(X), \alpha)\}$$

then, for any s and any α such that: (i) the cdfs G_s and H_s are continuous at $Q(s(X), \alpha)$, (ii) the cdfs G^* and H^* are continuous at $Q^*(\alpha)$, and (iii) $Q^*(\alpha) < 1$, we have:

$$\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(\mathrm{s}, \alpha) = \frac{\mathbb{E}(|\eta(X) - Q^*(\alpha)|\ \mathbb{I}\{X \in R_\alpha^* \Delta R_{s,\alpha}\})}{p(1 - Q^*(\alpha))}$$

where Δ denotes the symmetric difference between sets.

The last statement reveals that the pointwise difference between the dominating ROC curve and the one related to

a candidate scoring function s may be interpreted as the error made in recovering the specific level set R^*_α through $R_{s,\alpha}$. To our knowledge, this expression of the deviation between $\mathrm{ROC}^*(\alpha)$ and $\mathrm{ROC}(\mathrm{s},\alpha)$ is entirely new.

A simple consequence of the previous result (and its proof) is that the one-dimensional statistic $\eta(X)$ (instead of the supposedly high-dimensional observation X) suffices to recover the optimal ROC curve. In other words, projecting the original data onto $(0,1)$ using the regression function leaves the ROC curve untouched.

Corollary 7. *Consider the statistical model corresponding to the observation of the random pair $(\eta(X), Y)$. Then the optimal* ROC *curve under this statistical model is exactly the same as the optimal* ROC *curve for the random pair (X, Y).*

The following result will be needed later.

Proposition 8 (Derivative of the ROC)**.** *We assume that the optimal* ROC *curve is differentiable. Then, we have, for any α such that $Q^*(\alpha) < 1$:*

$$\frac{d}{d\alpha}\mathrm{ROC}^*(\alpha) = \frac{1-p}{p} \cdot \frac{Q^*(\alpha)}{1-Q^*(\alpha)} \ .$$

C. AUC *maximization*

Although the ROC curve is a useful graphical tool for evaluating the performance of a scoring function, its use as the target of an optimization strategy to estimate ROC-optimal scoring functions turns out to be quite challenging. Indeed, selecting a scoring function by empirical maximization of the ROC curve over a class $\mathcal{S}$ of scoring functions is a highly complex task because of the functional nature of the ROC curve criterion.

Of course, the closer to ROC^* the ROC curve of a candidate scoring function $s \in \mathcal{S}$, the more pertinent the ranking induced by s. However, various metrics can be considered for measuring the distance between curves. We focus on two essential cases:

- the L_1 metric

$$d_1(s^*, s) = \int_0^1 \{\mathrm{ROC}(\mathrm{s}^*,\alpha) - \mathrm{ROC}(\mathrm{s},\alpha)\}\ \mathrm{d}\alpha.$$

- the L_∞ metric

$$d_\infty(s^*, s) = \sup_{\alpha\in(0,1)} \{\mathrm{ROC}(\mathrm{s}^*,\alpha) - \mathrm{ROC}(\mathrm{s},\alpha)\} \ .$$

Remark 3. In order to avoid a possible confusion due to the notation, we bring to the reader's attention the fact that d_1 and d_∞ do not denote metrics on the space of scoring functions $\mathcal{S}$, but on the set of ROC curves.

As far as we know, the L_∞ metric has not been considered in the literature yet, although it is a natural choice given the view on the goal of ranking previously developed, *i.e.* recovering the collection of level sets $\{R^*_\alpha\}_{\alpha\in(0,1)}$ (see subsection II-B). Of course, L_∞-convergence implies convergence in the L_1-sense, while the reverse is generally false. However, the L_1-metric actually corresponds to a very popular criterion which is at the heart of most practical ranking methods. It is known as the Area Under an ROC Curve (or AUC in abbreviated form, see [13]).

Definition 9 (AUC)**.** *For any scoring function s, define the* AUC *as:*

$$\mathrm{AUC}(\mathrm{s}) = \int_0^1 \mathrm{ROC}(\mathrm{s},\alpha)\ \mathrm{d}\alpha \ ,$$

and set $\mathrm{AUC}^* = \mathrm{AUC}(\eta)$. We then have:

$$d_1(s^*, s) = \mathrm{AUC}^* - \mathrm{AUC}(\mathrm{s}) \ .$$

When it comes to finding a scoring function, based on empirical data, which will perform well with respect to the AUC criterion, various strategies can be considered.

A possible angle is the *plug-in* approach ([23]). The idea of plug-in consists of using an estimate $\hat{\eta}$ of the regression function as a scoring function. It is expected that, whenever $\hat{\eta}$ is close to η in a certain sense, then $\mathrm{ROC}(\hat{\eta},\cdot)$ and ROC^* are also close.

Proposition 10. *Consider $\hat{\eta}$ an estimator of η. We have:*

$$\mathrm{AUC}^* - \mathrm{AUC}(\hat{\eta}) \leq \frac{1}{\mathrm{p}(1-\mathrm{p})}\mathbb{E}\left(|\hat{\eta}(\mathrm{X}) - \eta(\mathrm{X})|\right) \quad a.s.$$

Assume that H^* has a density which is bounded by below on $[0,1]$: $\exists c > 0$ such that $\forall \alpha \in [0,1]$, $\frac{dH^*}{d\alpha}(\alpha) \geq c^{-1}$. Then, for any $\alpha \in [0,1]$ such that (i) $Q^*(\alpha) < 1$ and (ii) $H_{\hat{\eta}}$ is continuous at $Q(\hat{\eta}(X),\alpha)$ with probability 1, we have:

$$\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(\hat{\eta},\alpha) \leq \frac{\mathrm{c}\mathbb{E}\left(|\mathrm{H}^*(\eta(\mathrm{X})) - \mathrm{H}_{\hat{\eta}}(\hat{\eta}(\mathrm{X}))|\right)}{\mathrm{p}(1-\mathrm{Q}^*(\alpha))} \quad a.s.$$

However, plug-in rules face difficulties when dealing with high-dimensional data ([24]). Another drawback of plug-in rules is that they are not consistent with respect to the supremum norm. This observation provides an additional motivation for exploring algorithms based on empirical AUC maximization.

A nice feature of the AUC performance measure is that it may be interpreted in a probabilistic fashion.

Proposition 11 ([17])**.** *For any scoring function s such that H_s and G_s are continuous cdfs, we have:*

$$\begin{aligned}\mathrm{AUC}(\mathrm{s}) &= \mathbb{P}(s(X) > s(X') \mid Y = 1, Y' = -1) \\ &= \frac{1}{2p(1-p)}\mathbb{P}\{(s(X) - s(X'))(Y - Y') > 0\} \ .\end{aligned}$$

where (X, Y) and (X', Y') are i.i.d. copies.

From this observation, ranking can be interpreted as classification of pairs of observations. We refer to [17] for a systematic study of related empirical and convex risk minimization strategies which involve U-statistics. From a machine learning perspective, there is a growing literature in which existing algorithms are adapted in order to perform AUC optimization (such as, for instance: [14], [15], [16]). The tree-based method we propose in the sequel consists of an adaptive recursive strategy for building a piecewise constant scoring function with nearly maximum AUC.

III. Piecewise linear approximation of the optimal ROC curve

In this section, we assume that the distribution, and hence the optimal ROC curve, are known. We also assume that the optimal ROC curve is differentiable and concave (check Proposition 24). We consider the problem of building, in a stepwise manner, a scoring function whose ROC curve is a piecewise linear approximation/interpolation of the optimal curve ROC^*.

A. Piecewise constant scoring functions

The motivation for considering piecewise constant scoring functions comes from the representation result on optimal scoring functions given in Proposition 3. When it comes to approximations of the optimal s^*, a natural idea is to introduce discrete versions and to replace the expectation by a finite sum.

We recall that a partition of $\mathcal{X}$ is a finite class $\mathcal{C}_N = (C_j)_{1\leq j\leq N}$ of sets such that $\bigcup_j C_j = \mathcal{X}$ and $C_i \cap C_j = \emptyset$ for $i \neq j$.

We now introduce D-representation of a piecewise constant scoring function where the 'D' stands for 'disjoint'.

Definition 12 (D-representation)**.** *The D-representation of a piecewise constant scoring function s_N taking values in $\{a_1, \ldots, a_N\}$ is given by:*

$$\forall x \in \mathcal{X} \ , \quad s_N(x) = \sum_{j=1}^{N} a_j \ \mathbb{I}\{x \in C_j\} \ ,$$

for some decreasing sequence $(a_j)_{j\geq 1}$ and some partition *$\mathcal{C}_N = (C_j)_{1\leq j\leq N}$ of $\mathcal{X}$.*

We now list some obvious properties of piecewise constant scoring function.

Proposition 13. *Consider some piecewise constant scoring function s_N taking N different values.*

(i) *The* ROC *curve of s_N is piecewise linear with N linear parts.*

(ii) *The* ROC *curve of s_N does not depend on the particular values of the sequence $(a_j)_{j\geq 1}$ appearing in its D-representation but only on their ordering.*

We introduce the class $\mathcal{S}_N$ of piecewise constant scoring functions which take N distinct values.

Definition 14 (Class $\mathcal{S}_N$)**.** *We define $\mathcal{S}_N$ to be the class of scoring functions with D-representations of order N:*

$$\mathcal{S}_N = \{ s_N = \sum_{j=1}^{N} a_j \ \mathbb{I}_{C_j} \ : \ (C_j)_{j\geq 1} \text{ is a disjoint partition} \ , \ (a_j)_{j\geq 1} \text{ is a decreasing sequence} \} \ .$$

Our purpose in this section is to design an iterative procedure which outputs a piecewise constant scoring function $s_N \in \mathcal{S}_N$ whose ROC curve is as close as possible to the optimal ROC^*. Closeness between ROC curves will be measured both in terms of AUC and in the L_∞-sense. The iterative procedure described in the sequel satisfies the following approximation error result, see the proof of Proposition 15.

Proposition 15. *Assume that the optimal* ROC *curve is twice differentiable and concave and that its second derivative takes its values in a bounded interval which does not contain zero. There exists a sequence of piecewise constant scoring functions $(s_N)_{N\geq 1}$ such that, for any $N \geq 1$, $s_N \in \mathcal{S}_N$ and:*

$$\mathrm{AUC}^* - \mathrm{AUC}(s_N) = \mathrm{d}_1(s^*, s_N) \leq C \cdot N^{-2} \ ,$$
$$d_\infty(s^*, s_N) \leq C \cdot N^{-2} \ ,$$

where the constant C depends only on the distribution.

The proof can be found in the Appendix. The approximation rate $O(N^{-2})$ is actually reached by any piecewise linear approximant provided that the mesh length is of order $O(N^{-1})$. This result is well-known folklore in approximation theory, see [25]. We underline that the piecewise linear approximation method we describe next is adaptive in the sense that breakpoints are not fixed in advance and strongly depend on the target curve (which suggests that this scheme possibly yields a sharper constant C). It highlights the explicit relationship between the ROC^* approximant and the corresponding piecewise constant scoring function. The ranking algorithm proposed in the sequel (Section IV) will appear as a statistical version of this variable knot approximation, where the unknown quantities driving the recursive partitioning will be replaced by their empirical counterparts.

B. An alternative representation of scoring functions

It will be useful to consider another possible representation of piecewise constant scoring functions which is based on increasing sequences of sets.

Definition 16 (Increasing sequence of sets)**.** *We call an* increasing sequence of sets *of $\mathcal{X}$ is a finite class of sets $\mathcal{R}_N = (R_j)_{1\leq j\leq N}$ such that $\bigcup_j R_j = \mathcal{X}$ and $R_i \subset R_j$ for $i < j$. In particular, we have $R_N = \mathcal{X}$.*

Definition 17 (I-representation)**.** *Consider a piecewise constant scoring function s_N taking values in $\{1, \ldots, N\}$. Its I-representation is given by:*

$$\forall x \in \mathcal{X} \ , \quad s_N(x) = \sum_{j=1}^{N} \mathbb{I}\{x \in R_j\} \ ,$$

for some increasing sequence *$\mathcal{R}_N = (R_j)_{1\leq j\leq N}$ of subsets of $\mathcal{X}$.*

The relationship between D- and I-representations is straightforward. Assume that s_N takes values in $\{1, \ldots, N\}$ and consider the sequence $\mathcal{R}_N$ arising from the I-representation. We can then obtain the D-representation by taking $C_1 = R_1$ and:

$$\forall i > 2 \ , \ C_i = R_i \setminus R_{i-1} \quad \text{and} \quad \forall j \ , \ a_j = N - j + 1 \ .$$

In order to explicit the ROC curve of a piecewise constant scoring function, we introduce the following notations: for any

measurable $C \subset \mathcal{X}$,

$$\alpha(C) = \mathbb{P}\{X \in C \mid Y = -1\},$$
$$\beta(C) = \mathbb{P}\{X \in C \mid Y = +1\} \ .$$

Equipped with these notations, the ROC curve of a scoring function with I-representation $s_N(x) = \sum_{j=1}^{N} \mathbb{I}\{x \in R_j\}$ is the broken line that connects the knots $\{(\alpha(R_j), \beta(R_j))\}_{0 \leq j \leq N}$ with $R_0 = \emptyset$ by convention.

Remark 4. (CONCAVIFICATION) The ROC curve of a piecewise constant scoring function s_N is not necessarily concave. Denoting by $\mathcal{C}_N = (C_j)_{1 \leq j \leq N}$ a partition defining a D-representation of s_N, a possible way of remedying this consists of sorting the C_j's by decreasing order of the ratio $\beta(C_j)/\alpha(C_j)$, *i.e.* of considering a permutation σ of $\{1, \ldots, N\}$ such that

$$\frac{\beta(C_{\sigma(1)})}{\alpha(C_{\sigma(1)})} \geq \frac{\beta(C_{\sigma(2)})}{\alpha(C_{\sigma(2)})} \geq \ldots \geq \frac{\beta(C_{\sigma(N)})}{\alpha(C_{\sigma(N)})}.$$

The ROC curve related to the ordering induced by σ, *i.e.* of the scoring function $s_{N,\sigma}(x) = \sum_{j=1}^{N} (N - j + 1)\mathbb{I}\{x \in C_{\sigma(j)}\}$, is indeed concave.

C. One-step approximation to the optimal ROC curve

We now provide some insights on the general construction by describing the one-step modification of a given piecewise constant scoring function s_N. As advocated by Proposition 3, modifications are picked up in the class $\mathcal{G}$ of level sets of the regression function η:

$$\mathcal{G} = \{\{x \in \mathcal{X} \ : \ \eta(x) > t\} : t \in (0, 1)\}.$$

Definition 18 (One-step approximation)**.** *Given $s_N \in \mathcal{S}_N$, we define:*

$$\sigma_N = \arg\max_{\sigma \in \mathcal{G}} d_1(s_N, s_N + \sigma).$$

Then, the one-step approximation sequence to some optimal scoring function s^* is defined as the sequence $(s_N)_{N \geq 1}$ of scoring functions such that:

$$s_1 = \mathbb{I}_{\mathcal{X}},$$
$$s_{N+1} = s_N + \sigma_N \ , \quad N \geq 1 \ .$$

At this point, we shall consider the I-representation of piecewise constant scoring functions. A constructive procedure will rely on a particular choice of subsets $(R_j)_{j \geq 1}$. Following the result from Proposition 3, we focus on partitions with sets of the form:

$$R_j = \{x \in \mathcal{X} \ : \ \eta(x) > u_j\},$$

for some positive decreasing sequence $(u_j)_{j \geq 1}$ with $u_1 > 0$.

First iteration. We initialize the procedure for $N = 1$ with the scoring function:

$$\forall x \in \mathcal{X} \ , \quad s_1(x) = \mathbb{I}\{x \in \mathcal{X}\} \equiv 1 \ ,$$

which ranks all instances equally. It is clear that adding up the indicator of any region of the form $\{\eta(x) > t\}$ for some $t \in (0, 1)$ would provide a piecewise linear approximation of the optimal ROC curve. We choose the one which maximizes the AUC criterion.

Proposition 19 (First iteration)**.** *Assume that the optimal* ROC *curve is differentiable and concave. Then the one-step approximation at the first iteration is given by the piecewise constant scoring function:*

$$\forall x \in \mathcal{X} \ , \quad s_2(x) = \mathbb{I}\{x \in \mathcal{X}\} + \mathbb{I}\{\eta(x) > t^*\} \ ,$$

with $t^* = p$, where $p = \mathbb{P}\{Y = 1\}$. We also have:

$$(d\beta/d\alpha)(t^*) = 1.$$

Remark 5. (RANKING VS. CLASSIFICATION.) We point out that the optimal binary-valued scoring function in the AUC sense does not correspond to the Bayes classifier $g^*(x) = 2\mathbb{I}\{\eta(x) > 1/2\} - 1$, except when $p = 1/2$. Indeed, if we consider classifiers $g_t(x) = 2\mathbb{I}\{\eta(x) > t\} - 1$ of the form and look for the minimizer of the classification error:

$$\mathbb{P}\{Y \neq g_t(X)\} = p(1 - \alpha(t)) + (1 - p)\beta(t),$$

which is minimum for t such that $\frac{d\beta}{d\alpha}(t) = (1 - p)/p$ (if such a value can be reached), and hence $t = 1/2$ by Proposition 8. Denote by $r_{\max} = (d/d\alpha)(\mathrm{ROC}^*)(0)$ and $r_{\min} = (d/d\alpha)(\mathrm{ROC}^*)(1)$. When p falls out of the interval $((1 + r_{\max})^{-1}, (1 + r_{\min})^{-1})$ then one of the two extremal values will give the solution.

It is noteworthy that the one-step approximation obtained by optimization of the AUC criterion is the same as the one obtained through optimization of the sup-norm. The proof of the following proposition is simple and left to the reader.

Proposition 20. *Consider the increments at the first step:*

$$\sigma_1 = \arg\max_{\sigma \in \mathcal{G}} d_1(s_1, s_1 + \sigma),$$
$$\tilde{\sigma}_1 = \arg\max_{\sigma \in \mathcal{G}} d_\infty(s_1, s_1 + \sigma) \ .$$

We have: $\tilde{\sigma}_1 = \sigma_1$.

N-th iteration. Now consider a piecewise constant scoring function $s_N \in \mathcal{S}_N$. The ROC curve of s_N is a broken line with N linear pieces defined by the sequence of points $((\alpha_j, \beta_j))_{0 \leq j \leq N}$ where $(\alpha_0, \beta_0) = (0, 0)$ and $(\alpha_N, \beta_N) = (1, 1)$.

We look for the optimal splitting which would increase the AUC by adding a knot $(\alpha(t), \beta(t))$ such that $\alpha(t)$ is between α_j and α_{j+1}. We take the notation

$$s^{(j)}_{N+1,t}(x) = s_N(x) + \mathbb{I}\{\eta(x) > t\} \ ,$$

with $t \in (Q^*(\alpha_{j+1}), Q^*(\alpha_j))$. The AUC can then be written, for some constant c_j, as:

$$A_{N+1}(t) = \mathrm{AUC}(\mathrm{s}^{(\mathrm{j})}_{\mathrm{N+1,t}})$$
$$= c_j + \frac{1}{2}(\alpha_{j+1} - \alpha_j)\beta(t) - \frac{1}{2}\alpha(t)(\beta_{j+1} - \beta_j) \ ,$$

which is maximized at t^* such that:

$$d\beta(t^*) = \left(\frac{\beta_{j+1}-\beta_j}{\alpha_{j+1}-\alpha_j}\right) d\alpha(t^*) \ .$$

We can set $\alpha_j^* = \alpha(t^*)$ and we get, thanks to Proposition 8, the following relationship:

$$\frac{1-p}{p} \cdot \frac{Q^*(\alpha_j^*)}{1-Q^*(\alpha_j^*)} = \frac{\beta_{j+1}-\beta_j}{\alpha_{j+1}-\alpha_j} \ .$$

This leads to a one-step optimal splitting point (α_j^*, β_j^*) on the ROC curve such that:

$$\alpha_j^* = \bar{H}^*\left(\Delta_j\right) \quad \text{and} \quad \beta_j^* = \bar{G}^*\left(\Delta_j\right)$$

where

$$\Delta_j = \frac{p(\beta_{j+1}-\beta_j)}{(1-p)(\alpha_{j+1}-\alpha_j)+p(\beta_{j+1}-\beta_j)} = t^* \ .$$

Remark 6. (INTERPRETATION IN TERMS OF PARTITIONS.) The insertion of the new knot (α_j^*, β_j^*) is materialized by the splitting of subset R_{j+1} with a subset R_j^* containing R_j and we have:

$$R_j^* = \{x \in \mathcal{X} \ : \ \eta(x) > Q^*(\alpha_j^*)\} \ ,$$

while $R_j = \{x \in \mathcal{X} \ : \ \eta(x) > Q^*(\alpha_j)\}$. In terms of D-representations, we can write:

$$s_N = \sum_{j=1}^{N} (N-j+1) \ \mathbb{I}_{C_j}$$

where

$$C_j = \{x \in \mathcal{X} \ : \ Q^*(\alpha_{j+1}) < \eta(x) \leq Q^*(\alpha_j)\} \ .$$

After the splitting, in the new partition, the set C_{j+1} is replaced by C_j^* and $C_{j+1} \setminus C_j^*$ where

$$C_{j+1} = \{x \in \mathcal{X} \ : \ Q^*(\alpha_{j+1}) < \eta(x) \leq Q^*(\alpha_j^*)\} \ .$$

The previous computations quantify the improvement in terms of AUC after adding one knot for each linear part of the ROC curve at step N. Instead of sticking to one-step approximations, we can introduce an approximation scheme which will add 2^N knots after the N-th iteration.

D. A tree-structured recursive approximation scheme

We now turn to the full recursive procedure. At each step, an adaptively chosen knot is added between all consecutive points of the current meshgrid. We take $N = 2^D$ with $D \geq 0$ and we describe iterations over D for constructing a sequence of piecewise constant scoring functions. It will be easier to work with D-representations of the form:

$$\forall x \in \mathcal{X} \ , \quad s_D(x) = \sum_{k=0}^{2^D-1} (2^D - k) \ \mathbb{I}\{x \in C_{D,k}\} \ ,$$

where, for fixed D, the class of sets $(C_{D,k})_{0 \leq k \leq 2^D-1}$ is a disjoint partition of $\mathcal{X}$.

The iterative procedure goes as follows.

Initialization ($d = 0$ and $d = 1$). For the extremal points, we set:

$$\forall d \in \mathbb{N} \ , \ \alpha_{d,0}^* = \beta_{d,0}^* = 0 \quad \text{and} \quad \alpha_{d,2^d}^* = \beta_{d,2^d}^* = 1 \ ,$$

and for the first iteration points ($d = 1$):

$$\alpha_{1,1}^* = \bar{H}^*(p) \quad \text{and} \quad \beta_{1,1}^* = \bar{G}^*(p) \ ,$$

From d to $d+1$, for $d \geq 1$. We are given the collection of points $\{(\alpha_{d,k}^*, \beta_{d,k}^*)\}_{k=0,\ldots,2^d-1}$. On each interval $(\alpha_{d,k}^*, \alpha_{d,k+1}^*)$, we apply the one-step approximation. Hence, the new point is given by:

$$\begin{array}{rcl} \alpha_{d+1,2k+1}^* & = & \bar{H}^*\left(\Delta_{d+1,2k+1}^*\right), \\ \beta_{d+1,2k+1}^* & = & \bar{G}^*\left(\Delta_{d+1,2k+1}^*\right), \end{array}$$

where

$$\Delta_{d+1,2k+1}^* = \frac{p(\beta_{d,k+1}^* - \beta_{d,k}^*)}{(1-p)(\alpha_{d,k+1}^* - \alpha_{d,k}^*) + p(\beta_{d,k+1}^* - \beta_{d,k}^*)} \ .$$

Moreover, the previous cut-off point is renamed:

$$\alpha_{d+1,2k}^* = \alpha_{d,k}^* \quad \text{and} \quad \beta_{d+1,2k}^* = \beta_{d,k}^* \ ,$$

and also $\Delta_{d+1,2k}^* = \Delta_{d,k}^*$.

Note that, for each level d, the resulting partition is given by the class of sets:

$$C_{d,k}^* = \{x \in \mathcal{X} \ : \ \Delta_{d,k}^* < \eta(x) \leq \Delta_{d,k+1}^*\},$$

for all $k = 0, \ldots, 2^d - 1$ with the convention that $\Delta_{d,0}^* = 0$ and $\Delta_{d,2^d} = 1$ for all $d \geq 0$.

For all $d \in \mathbb{N}$, we also define the sets $R_{d,k}^*$ by: $\forall k \in \{1, \ldots, 2^d - 1\}$, $R_{d,k}^* = C_{d,k}^* \cup R_{d,k-1}^*$ with $R_{d,0}^* = C_{d,0}^*$.

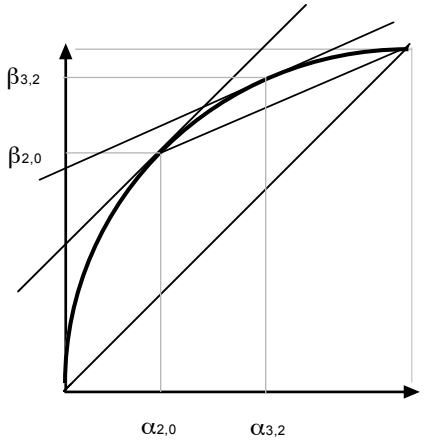

Fig. 1. Piecewise linear approximation of the ROC curve.

Remark 7. (A TREE-STRUCTURED RECURSIVE INTERPOLATION SCHEME.) A nice feature of the recursive approximation procedure is its binary-tree structure. Owing to their crucial

practical advantages regarding implementation and interpretation, tree-structured decision rules have been proved useful for a wide range of statistical tasks and are in particular among the most popular methods for regression and classification (we refer to Chapter 20 in [23] for an excellent account of tree decision rules in the context of classification).

Remark 8. (A PIECEWISE CONSTANT APPROXIMANT OF THE REGRESSION FUNCTION.) Although the angle embraced in this paper consists of directly building a partitioning of the input space corresponding to a nearly optimal ranking in the spirit of popular machine-learning algorithms, we point out that, as a byproduct, the resulting partition provides a stepwise approximation of the regression function:

$$\begin{aligned}\tilde{\eta}(x) &= \sum_{k=0}^{2^D-1} \mathbb{P}(Y=+1 \mid X \in C^*_{D,k})\ \mathbb{I}\{x \in C^*_{D,k}\} \\ &= \sum_{k=0}^{2^D-1} \Delta^*_{D+1,2k+1}\ \mathbb{I}\{x \in C^*_{D,k}\}.\end{aligned}$$

Provided that H^* is strictly increasing, the scoring function $s(x) = H^*(\eta(x))$ is also optimal and is approximated by:

$$\tilde{s}(x) = \sum_{j=0}^{2^D-1} (\alpha^*_{D,j+1} - \alpha^*_{D,j})\ \mathbb{I}\{x \in R^*_{D,j}\},$$

which should be seen as a Riemann's discretization of the integral $\int_0^1 \mathbb{I}\{\eta(x) > Q^*(\alpha)\}\ d\alpha$ (see Remark 1).

In order to provide a closed analytical form for the (linear-by-parts) ROC curve of the stepwise scoring function

$$s^*_D(x) = \sum_{k=0}^{2^D-1} (2^D - k)\ \mathbb{I}\{x \in C^*_{D,k}\},$$

consider the *"hat functions"* defined by

$$\phi^*_{d,k}(.) = \phi(.;\ (\alpha^*_{d,k-1}, \alpha^*_{d,k})) - \phi(.;\ (\alpha^*_{d,k}, \alpha^*_{d,k+1})),$$

for $d \geq 0$ and $1 \leq k \leq 2^d - 1$, with the notation

$$\phi(\alpha;\ (\alpha_1, \alpha_2)) = \frac{\alpha - \alpha_1}{\alpha_2 - \alpha_1} \mathbb{I}\{\alpha \in [\alpha_1, \alpha_2]\}$$

for $-\infty < \alpha_1 < \alpha_2 < \infty$. For notational convenience, we also set

$$\phi^*_{d,2^d}(.) = \phi(.;\ (\alpha^*_{d,2^d-1}, 1)).$$

Equipped with these notations, one may classically write the FEM approximation of the optimal ROC curve based on the meshgrid $\{\alpha^*_{D,k}\}_{0 \leq k \leq 2^D-1}$ as

$$\forall \alpha \in [0,1],\ \mathrm{ROC}(\mathrm{s}^*_{\mathrm{D}}, \alpha) = \sum_{\mathrm{k}=1}^{2^{\mathrm{D}}} \beta^*_{\mathrm{D,k}} \phi^*_{\mathrm{D,k}}(\alpha).$$

It is noteworthy that the approximant is increasing and concave, as the target curve ROC^*. Furthermore, from this representation, one may straightforwardly get the following expression for the corresponding estimate of the optimal AUC:

$$\mathrm{AUC}(\mathrm{s}^*_{\mathrm{D}}) = \frac{1}{2} \sum_{\mathrm{k}=1}^{2^{\mathrm{D}}-1} (\alpha^*_{\mathrm{D,k}+1} - \alpha^*_{\mathrm{D,k}-1}) \beta^*_{\mathrm{D,k}}.$$

As stated in Proposition 15 (see the proof in Appendix B), the deviation between ROC^* and $\mathrm{ROC}(\mathrm{s}^*_{\mathrm{D}}, .)$ is of order 2^{-2D} when measured either in terms of AUC or else in sup norm.

IV. A TREE-STRUCTURED WEAK RANKER

It is time to exploit the theory developed in the previous sections to deal with empirical data. We formulate a practical algorithm which implements a top-down strategy to build a binary tree-structured scoring function. This algorithm mimics the ideal recursive approximation procedure of the optimal ROC curve from Section III, where probabilities are replaced by their empirical counterparts.

A. The TREERANK algorithm

We assume now that a training data set

$$\mathcal{D}_n = \{(X_1, Y_1),\ \ldots,\ (X_n, Y_n)\}$$

of n independent copies of the pair (X, Y) is available. We set

$$n_+ = \sum_{i=1}^n \mathbb{I}\{Y_i = 1\} \quad \text{and} \quad n_- = \sum_{i=1}^n \mathbb{I}\{Y_i = -1\}\ .$$

We introduce the following data-based quantities, for any subset C:

$$\hat{\alpha}(C) = \frac{1}{n_-} \sum_{i=1}^n \mathbb{I}\{X_i \in C,\ Y_i = -1\}$$
$$\hat{\beta}(C) = \frac{1}{n_+} \sum_{i=1}^n \mathbb{I}\{X_i \in C,\ Y_i = +1\}$$

which correspond respectively to the empirical false positive rate and the empirical true positive rate of a classifier predicting +1 on the set C.

For notational convenience, we set $\alpha_{d,0} = \beta_{d,0} = 0$ and $\alpha_{d,2^d} = \beta_{d,2^d} = 1$ for all $d \geq 0$. We assume that we are given a class $\mathcal{C}$ of subsets of $\mathcal{X}$.

TREERANK ALGORITHM

1) **Initialization.** Set $C_{0,0} = \mathcal{X}$.

2) **Iterations.** For $d = 0, \ldots, D-1$ and $k = 0, \ldots, 2^d - 1$:

   a) (OPTIMIZATION STEP.) Set the entropic measure:

$$\begin{aligned}\Lambda_{d,k+1}(C) &= (\alpha_{d,k+1} - \alpha_{d,k})\hat{\beta}(C) \\ &- (\beta_{d,k+1} - \beta_{d,k})\hat{\alpha}(C) \ .\end{aligned}$$

   Find the best subset $C_{d+1,2k}$ of rectangle $C_{d,k}$ in the AUC sense:

$$C_{d+1,2k} = \underset{C \in \mathcal{C},\ C \subset C_{d,k}}{\arg\max} \Lambda_{d,k+1}(C) \ .$$

   Then, set $C_{d+1,2k+1} = C_{d,k} \setminus C_{d+1,2k}$.

   b) (UPDATE.) Set

$$\begin{aligned}\alpha_{d+1,2k+1} &= \alpha_{d,k} + \hat{\alpha}(C_{d+1,2k}) \\ \beta_{d+1,2k+1} &= \beta_{d,k} + \hat{\beta}(C_{d+1,2k})\end{aligned}$$

   and

$$\begin{aligned}\alpha_{d+1,2k+2} &= \alpha_{d,k+1} \\ \beta_{d+1,2k+2} &= \beta_{d,k+1} \ .\end{aligned}$$

3) **Output.** After D iterations, we get the piecewise constant scoring function:

$$s_D(x) = \sum_{k=0}^{2^D-1} (2^D - k)\ \mathbb{I}\{x \in C_{D,k}\},$$

   together with the AUC estimate

$$\begin{aligned}\widehat{\mathrm{AUC}}(s_D) &= \frac{1}{2}\sum_{k=1}^{2^D-1} (\alpha_{D,k+1} - \alpha_{D,k-1})\beta_{D,k} \\ &= \frac{1}{2} + \frac{1}{2}\sum_{k=0}^{2^{D-1}-1} \Lambda_{D-1,k+1}(C_{D,2k})\end{aligned}$$

   and the estimate of the curve $\mathrm{ROC}(\mathrm{s_D}, .)$

$$\widehat{\mathrm{ROC}}(s_D, \alpha) = \sum_{k=1}^{2^D} \beta_{D,k}\phi_{D,k}(\alpha),\ \alpha \in [0,1],$$

   where

$$\begin{aligned}\phi_{D,k}(.) &= \phi(.;\ (\alpha_{D,k-1}, \alpha_{D,k})) \\ &- \phi(.;\ (\alpha_{D,k}, \alpha_{D,k+1})), \\ \phi_{D,2^D}(.) &= \phi(.; (\alpha_{D,2^D-1}, 1)).\end{aligned}$$

Important features of the TREERANK algorithm are listed in the following remarks.

Remark 9. (READING THE RANKS.) The resulting ranking induced by the scoring function s_D may be read from the left to the right looking at the terminal nodes (see Figure 2).

Remark 10. (A SIMPLISTIC STOPPING CRITERION.) If there is more than one subrectangle solution in the OPTIMIZATION STEP, take the larger. Hence, if there is no improvement in terms of AUC maximization when splitting the current rectangle $C_{d,k}$, set $C_{d+1,2k} = C_{d,k}$, so that $C_{d+1,2k+1} = \emptyset$.

Remark 11. (ON THE SPLITTING RULE.) In the context of classification, this splitting rule has been considered previously in [19]. We point out that, in contrast to tree-based classification methods, such as CART, the splitting criterion depends on the node through the parent's false and true positive rates $\hat{\alpha}(C)$ and $\hat{\beta}(C)$. This can be explained by the fact that the goal pursued in the ranking problem is global: one attempts to order all input data with respect to each other.

Remark 12. (LINEAR SPLITS.) The choice of the class $\mathcal{C}$ is a matter of trade-off between representation ability and computation cost. Linear splits lead to a rich class of partitions but practitioners would rather go for orthogonal splits. The choice of orthogonal splits amounts to using a class $\mathcal{R}$ of decision stumps, obtained by cutting a certain coordinate of the input vector X at a certain level (the split variable and the level being chosen so as to maximize the AUC). The subclass to be enumerated is then the intersection of decision stumps with the set represented in the parent node. This choice presents a clear advantage on the algorithmic side but suffers from representation ability as we will see in Section VI.

Remark 13. (TRUE ROC CURVE AND AUC.) We point out that the (true) AUC of the scoring function produced by TREERANK is given by:

$$\mathrm{AUC}(\mathrm{s_D}) = \frac{1}{2}\sum_{\mathrm{k}=1}^{2^{\mathrm{D}}-1} (\alpha(\mathrm{C_{D,k}}) + \alpha(\mathrm{C_{D,k-1}}))\beta(\mathrm{R_{D,k-1}}),$$

where $R_{d,j} = \cup_{k=0}^{j} C_{d,j}$ for all $d \geq 0$, $j \in \{0, \ldots, 2^d - 1\}$. Furthermore, its (true) ROC curve may be written as:

$$\mathrm{ROC}(\mathrm{s_D}, \alpha) = \sum_{\mathrm{k}=1}^{2^{\mathrm{D}}} \beta(\mathrm{R_{D,k-1}})\tilde{\phi}_{\mathrm{D,k}}(\alpha),\ \alpha \in [0,1],$$

where

$$\begin{aligned}\tilde{\phi}_{D,k}(.) &= \phi(.;\ (\alpha(R_{D,k-2}), \alpha(R_{D,k-1}))) \\ &- \phi(.;\ (\alpha(R_{D,k-1}), \alpha(R_{D,k}))), \\ \tilde{\phi}_{D,2^D}(.) &= \phi(.; (\alpha(R_{D,2^D-2}), 1)).\end{aligned}$$

As stated in the next result, another major feature of the TREERANK algorithm is that, similarly to the approximant $\mathrm{ROC}(\mathrm{s^*_D}, .)$, the estimate $\widehat{\mathrm{ROC}}(s_D, .)$ of ROC^* it outputs is necessarily concave as soon as the set $\mathcal{C}$ is *union stable* (whereas this is not necessarily true for the theoretical ROC curve $\mathrm{ROC}(\mathrm{s_D}, .)$).

Proposition 21. *(CONCAVITY OF THE ROC^* ESTIMATE.) Suppose that the class $\mathcal{C}$ of sets is union stable, i.e. $\forall (C, C') \in \mathcal{C}^2$: $C \cup C' \in \mathcal{C}$. Consider the scoring function s_D output by the* TREERANK *algorithm after 2^D iterations. Its empirical* ROC *curve, $\widehat{\mathrm{ROC}}(s_D, .)$, is concave.*

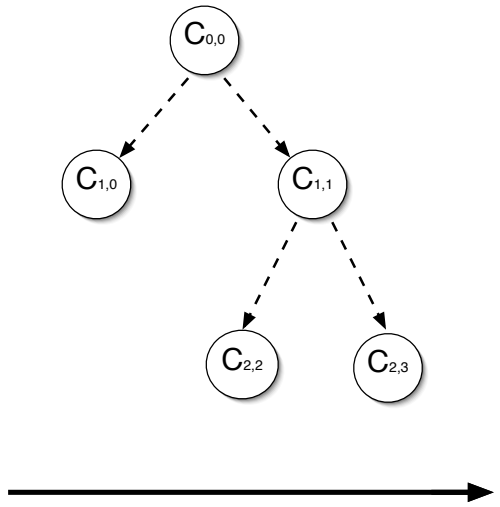

Fig. 2. Numbering of the nodes and order for reading the ranks.

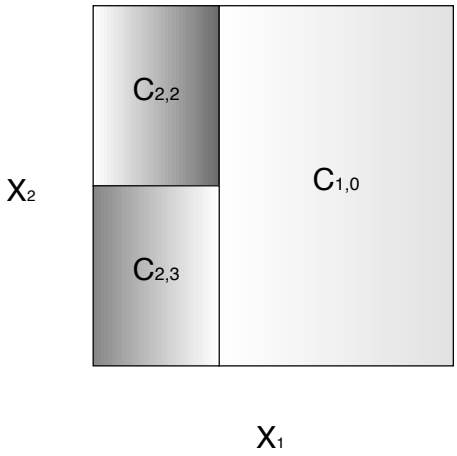

Fig. 3. Partitioning induced by the tree structure with perpendicular splits.

Remark 14. (RANKING AS TESTING FOR HOMOGENEITY) Whereas there is a wide variety of possible approaches in the one-dimensional case, testing for homogeneity in a high-dimensional space is a very challenging task. In this respect, the ROC^* estimate output by the TREERANK algorithm may be useful, insofar that the null assumption boils down to claim that ROC^* is simply the first diagonal of the ROC space. Indeed, suppose we are interested in testing the hypothesis $\mathcal{H}_0 : H = G$ based on sample data. A possible method could consist of first projecting the data onto the real line using s_D and then applying a standard test (based on ranks) for homogeneity between the real-valued samples $\{s_D(X_i') : Y_i' = +1,\ 1 \leq i \leq n'\}$ and $\{s_D(X_i') : Y_i' = -1,\ 1 \leq i \leq n'\}$, where $\mathcal{D}_{n'}' = \{(X_i', Y_i');\ 1 \leq i \leq n'\}$ is a sample of n' i.i.d. copies of the pair (X, Y), independent from $\mathcal{D}_n$.

B. Consistency of TREERANK and rate bounds

We now provide a consistency result for the class of partitions induced by the TREERANK algorithm. The formulation (and the proof) mimics Theorem 21.2 from [23].

Theorem 22. *We consider scoring functions s_n corresponding to partitions $\mathcal{F}_n$ of $\mathcal{X}$. We assume that the $\mathcal{F}_n$'s are random partitions of $\mathcal{X}$ resulting from runs of* TREERANK *with training sets of size n. We also assume that $\mathcal{X}$ is bounded and that the partitions $\mathcal{F}_n$ belong to a VC class of sets with VC dimension V, for any n and any training set. If the diameter of any cell of $\mathcal{F}_n$ goes to 0 when n tends to infinity, then we have that:*

$$\mathrm{AUC}(\mathrm{s}^*) - \mathrm{AUC}(\mathrm{s_n}) = \mathrm{d_1}(\mathrm{s}^*, \mathrm{s_n}) \to 0$$

almost surely, as n goes to ∞.

If we have, in addition that H^* has a density which is bounded by below on $[0,1]$ and that, for any α, $Q^*(\alpha) < 1-\epsilon$, for some $\epsilon > 0$, then:

$$d_\infty(s^*, s_n) \to 0$$

almost surely, as n goes to ∞.

Remark 15. (BOUNDEDNESS OF $\mathcal{X}$.) This assumption is a simplification which can be removed at the cost of a longer proof (the core of the argument can be found in [23]).

Remark 16. (COMPLEXITY ASSUMPTION.) Instead of assuming a finite VC dimension, a weaker assumption on the combinatorial entropy of the class of partitions may be provided (again check [23] for this refinement).

Under additional assumptions, rate bounds can be established for the scoring function produced by TREERANK. We strongly emphasize that the rate bound for $d_\infty(\hat{s}_D, s^*)$ corresponds to a confidence band for $\mathrm{ROC}(\hat{\mathrm{s}}_\mathrm{D}, .)$ in a functional space, namely the space $\mathcal{C}([0,1])$ of real-valued continuous functions on $[0,1]$ equipped with the sup norm, whereas the one for $d_1(\hat{s}_D, s^*)$ yields a confidence interval for the real-valued quantity $\mathrm{AUC}(\hat{\mathrm{s}}_\mathrm{D})$. To our knowledge, it is the first result of this nature available in the statistical learning literature.

Theorem 23. *Assume that conditions of Proposition 15 are fulfilled. Suppose that the class $\mathcal{C}$ of subset candidates contains all level sets R_α^*, $\alpha \in [0,1]$ and is intersection stable, i.e. $\forall (C, C') \in \mathcal{C}^2$: $C \cap C' \in \mathcal{C}$. Assume furthermore that $\mathcal{C}$ has finite* VC *dimension V.*

(i) *For all $\delta > 0$, there exists a constant c_0 and universal constants c_1, c_2 such that, with probability at least $1-\delta$, we have for all $D \geq 1$, $n \in \mathbb{N}$:*

$$d_1(\hat{s}_D, s_D) \leq c_0^D \left\{ \left(\frac{c_1^2 V}{n} \right)^{\frac{1}{2D}} + \left(\frac{c_2^2 \log(1/\delta)}{n} \right)^{\frac{1}{2D}} \right\},$$

$$d_\infty(\hat{s}_D, s_D) \leq c_0^D \left\{ \left(\frac{c_1^2 V}{n} \right)^{\frac{1}{2(D+1)}} + \left(\frac{c_2^2 \log(1/\delta)}{n} \right)^{\frac{1}{2(D+1))}} \right\}.$$

(ii) *Choose $D = D_n$ so that $D_n \sim \sqrt{\log n}$, as $n \to \infty$. Then, for all $\delta > 0$, there exists a constant κ such that,*

with probability at least $1-\delta$*, we have for all* $n \in \mathbb{N}$*:*

$$d_i(\hat{s}_{D_n}, s^*) \leq \exp(-\kappa\sqrt{\log n}), \;\; i \in \{1, \infty\}.$$

Remark 17. (ESTIMATION OF THE OPTIMAL ROC CURVE) It follows from the argument of Theorem 23 that, if one chooses $D_n \sim \sqrt{\log n}$, the empirical ROC curve $\widehat{\mathrm{ROC}}(s_{D_n}, .)$ output by the TREERANK algorithm is a consistent estimator of the optimal curve ROC^*, for both the L_1-distance and the sup norm, the same rate bound as for the true ROC curve $\mathrm{ROC}(\mathrm{s}_{\mathrm{D_n}}, .)$ holding true.

V. BEYOND THE TREERANK ALGORITHM

The TREERANK methodology inherits certain drawbacks from its hierarchical nature, like CART, instability and a lack of smoothness essentially. These drawbacks are emphasized because of the global nature of the ranking goal: indeed, changing the rank/score of an instance $x \in \mathcal{X}$ possibly affects the ranks of many other instances, whereas the classification task is local. In the present section we discuss these issues and propose various modifications of the original TREERANK algorithm. Two types of strategies are considered. The first approach consists of improving the performance of one single tree, while the second one relies on combining several ranking trees following the example of committee-based methods in classification.

A. Pruning a ranking tree

The complexity of a piecewise constant scoring function can naturally be described by the cardinality of the partition involved in its D-representation, see Definition 12. A classical approach in model selection consists of penalizing candidates according to their complexity using an adequate cost function and then choosing the model yielding the best trade-off between performance and complexity cost.

In the ranking setup, a possible strategy could be to grow first a deep ranking tree via TREERANK, producing a scoring function $s_D(x)$ with large depth D, and then considering the ordering induced by "subtrees", the latter being obtained by merging certain neighboring subrectangles $C_{D,k}$. Formally, the ranking induced by a subtree is entirely determined by an element of the set Θ_D of increasing sequences $\theta : \{0, \ldots, 2^D - 1\} \to \{0, \ldots, 2^D - 1\}$ such that $\theta(0) = 0$ and $\forall k \in \{1, \ldots, 2^D - 1\}$, $\theta(k) - \theta(k-1) \geq 1$. We set

$$\forall \theta \in \Theta_D, \;\; s_D^\theta = \sum_{k=0}^{2^D-1} (2^D - \theta(k)) \mathbb{I}\{x \in C_{D,k}\}.$$

The size of the corresponding partition is then $\#\theta = \theta(2^D - 1)$. The idea is to maximize the *complexity-penalized* AUC over Θ_D:

$$\widehat{\mathrm{AUC}}_\lambda(\theta) = \widehat{\mathrm{AUC}}(s_D^\theta) - \lambda \cdot \#\theta.$$

The tuning parameter λ rules the trade-off between ranking performance and ranking-tree size. It may be estimated by cross-validation.

Remark 18. (WEAKEST LINK PRUNING) If one restricts itself to the case where only *siblings* can be merged, a fast bottom-up pruning procedure may be implemented for determining the optimal subtree. We recall that siblings corresponding to subrectangles of the tree which have the same parent node. As for CART in the classification setup, it suffices to collapse the internal node that corresponds to the smallest decrease in terms of AUC, node after node, producing a sequence of embedded subtrees containing the optimal one. We refer to [18] for further details.

B. Shaking the ranking tree

Because of the hierarchical structure of the tree growing procedure, it would be convenient to possibly consider orderings of the subregions of the resulting partition other than the one implicitly obtained by perusing the terminal nodes of the tree from the left to the right. As a matter of fact, due to the specific topology induced by the recursive partitioning, ranking errors induced by a non ideal split cannot be corrected by growing the tree deeper. Indeed, it may happen that a region $C_{d,k}$ of the input space is split into two subregions $R_{d+1,2k} \cup R_{d+1,2k+1}$, in a way that $R_{d+1,2k}$ unfortunately contains a few instances which are less relevant than certain instances of $R_{d+1,2k+1}$. Even though TREERANK keeps on running endlessly, these instances will never be ranked worse than any of the instances of $R_{d+1,2k+1}$. However, it is possible to modify the algorithm so that it encourages recursive partitioning to automatically detect low cardinality groups of instances of low ranks surrounded by instances of high ranks.

Suppose that one disposes of a scoring function with D-representation $s_N(x) = \sum_{i=1}^N (N-i)\mathbb{I}\{x \in C_i\}$. Let σ be an element of the group $\mathcal{G}_N$ of permutations of $\{1, \ldots, N\}$ and consider

$$s_{N,\sigma}(x) = \sum_{i=1}^N (N - \sigma(i))\mathbb{I}\{x \in C_i\}.$$

The ordering of the subregions C_i, $1 \leq i \leq N$, corresponding to the largest AUC corresponds to the permutation

$$\sigma^* = \arg\max_{\sigma \in \mathcal{G}_N} \mathrm{AUC}(\mathrm{s}_{\sigma,\mathrm{N}}).$$

Hence, at each iteration, the partitioning criterion could be enriched in order to evaluate the gain from splitting the current region in terms of the overall AUC, allowing for intercalating the siblings at any possible ranks in the current ordering.

C. On bagging ranking trees

In order to reduce the variability/instability of the ranking rules produced by TREERANK, a possible approach consists of "averaging" many ranking trees, following the *bootstrap* paradigm. This approach, proposed by [26] in the context of binary classification and regression, is known as *bagging*. In our setup, the bagging strategy boils down to generating M independent training data sets by sampling with replacement from the original data, $\mathcal{D}_n^{(1)}, \ldots, \mathcal{D}_n^{(M)}$, running next TREERANK from each of these bootstrap samples, yielding the scoring functions

$$\tilde{s}^{(m)}(c) = \sum_{j=0}^{2^D-1} (\alpha_{D,j+1}^{(m)} - \alpha_{D,j}^{(m)}) \mathbb{I}\{x \in R_{D,j}^{(m)}\},$$

with $m \in \{1, \ldots, M\}$, see Remark 8. The bagging ranker is then given by: $\forall x \in \mathcal{X}$,

$$\tilde{s}_{bag}(x) = \frac{1}{M} \sum_{i=m}^{M} \tilde{s}^{(m)}(x),$$

the predicted score of a given instance being thus the average score from these M tree-based scoring functions. This provides a smoother ranking rule, corresponding to an estimate of $H^*(\eta(x))$ with lower variance than the one of a single scoring function.

D. Boosting tree ranking rules

We suggest that a tree-based ranker could serve as a weak learner and feed a boosting-type algorithm. However, as noticed in [6], extending the notion of aggregating predictors to the ranking problem is far from obvious, due to the fact that what one is trying to predict, the proper ordering on $\mathcal{X}$, is not of binary nature. In this respect, it is noteworthy that the RANKBOOST algorithm indeed proposes an extension of the ADABOOST methodology in the limiting case where weak scoring rules are binary solely.

Nevertheless, building on the approach developed in [8] according to which ranking is viewed as a pairwise classification problem, it is possible to bring back ourselves to the binary setting. An apparently restrictive formulation of the ranking problem consists of determining which one among two instances X and X', independently drawn at random, is "better" *i.e.* predicting the sign of $Y' - Y$, the random variables Y and Y' denoting the respective labels of X and X'. We call a *ranking rule* any antisymmetric predictor: $r : \mathcal{X} \times \mathcal{X} \to \{-1, 0, 1\}$ such that $\forall (x, x') \in \mathcal{X}^2$, $r(x, x') = -r(x', x)$. Since the ranking rule with minimum *ranking risk* $L(r) = \mathbb{P}(r(X, X') \cdot (Y - Y') < 0)$ is $r^*(x, x') = 2 \cdot \mathbb{I}\{\eta(x) > \eta(x')\} - 1$, it is natural to seek for ranking rules of the form $r_s(x, x') = 2 \cdot \mathbb{I}\{s(x) > s(x')\} - 1$ where $s \in \mathcal{S}$. Notice that, equipped with this notation, $AUC(s) = 1 - L(r_s)/(2p(1-p))$.

Reciprocally, one may deduce a scoring function from a ranking rule r. It suffices to consider for instance the function

$$s_r(x) = \mathbb{E}[r(x, X') \mid Y' = -1],$$

which represents the average number of negative instances that are predicted "worse" than x. We point out that one generally has $r \neq r_{s_r}$ unless the rule r is transitive, *i.e.* $\forall (x, x', x'') \in \mathcal{X}^3$, if $r(x, x') = +1$ and $(x', x'') = +1$, then, necessarily, $r(x, x'') = +1$. Observe that $s_{r^*}(x) = 2H^*(\eta(x)) - 1$.

For notational simplicity, consider

$$\{(X_i, Y_i) : \ Y_i = +1 \text{ and } 1 \leq i \leq n\} = \{X_i^+ : \ 1 \leq i \leq n_+\},$$
$$\{(X_i, Y_i) : \ Y_i = -1 \text{ and } 1 \leq i \leq n\} = \{X_j^- : \ 1 \leq j \leq n_-\},$$

TREERANK-BOOST

1) **Initialization.** Assign the weights

$$\omega_i^+ = \frac{1}{n_+}, \ 1 \leq i \leq n_+$$
$$\omega_j^- = \frac{1}{n_-}, \ 1 \leq j \leq n_-,$$

to the data $\{X_i^+\}_{1 \leq i \leq n_+}$ and $\{X_j^-\}_{1 \leq j \leq n_-}$.

2) **Iterations.** For $m = 1, \ldots, M$:

a) (WEAK RANKING.) From the weighted training data, run TREERANK, producing the scoring function $s^{(m)}$ and the associated ranking rule $r^{(m)} = r_{s^{(m)}}$.

b) (RANKING ERROR.) Compute the *weighted rate of discording pairs*

$$L_m = \sum_{i=1}^{n_+} \sum_{j=1}^{n_-} w_i^+ w_j^- \mathbb{I}\{s^{(m)}(X_i^+) < s^{(m)}(X_j^-)\}$$

and set $a_m = \log((1 - L_m)/L_m)$.

c) (UPDATE.) Set

$$\omega_i^+ \leftarrow \omega_i^+ e^{-L_m s^{(m)}(X_i^+)}, \ 1 \leq i \leq n_+,$$
$$\omega_j^- \leftarrow \omega_j^- e^{-L_m s^{(m)}(X_j^-)}, \ 1 \leq j \leq n_-,$$

and normalize the weights so that $\sum_{i=1}^{n_+} \omega_i^+ = 1$ and $\sum_{j=1}^{n_-} \omega_j^- = 1$.

3) **Output.** After B iterations, get the ranking rule

$$r_{Boost}(x, x') = 2 \cdot \mathbb{I}\left\{\sum_{m=1}^{M} a_m r^{(m)}(x, x') > 0\right\} - 1$$

and the scoring function

$$s_{Boost}(x) = \frac{1}{n_-} \sum_{j=1}^{n_-} r_{Boost}(x, X_j^-).$$

Following the view on Boosting developed in [27], the algorithm above may be interpreted in terms of forward stagewise additive modeling for approximating the solution to

$$\min_r \mathbb{E}[e^{r(X, X')} \mid Y = -1, \ Y' = 1].$$

VI. A TOY EXAMPLE

It is not the purpose of this paper to provide a fully practical way of implementing the TREERANK methodology. Related discussions and empirical studies are postponed to a forthcoming companion paper. The simulation example displayed in this section solely serves as an illustration and it should not be considered as more than that. However, we point out that the efficiency of the algorithm is guaranteed by the supposed fact that, at each iteration (d, k), the class of subrectangle candidates is rich enough to contain a good approximation of

the optimal subregion $C^*_{d+1,2k}$. As mentioned in Remark 12, a simple approach would consist of implementing TREERANK with perpendicular splits, in the spirit of the original CART method proposed by [18]. It is the way we proceeded in the example below. But, as for the classification task, many other types of cuts could be pertinently considered, involving combinations of several coordinates for instance. In practice, it may happen that perpendicular splits do not lead to a nearly optimal partitioning and it can be necessary to adapt this naive approach in order to achieve more flexible cuts. A possible key to the design of an efficient implementation of TREERANK could consist of enriching the splitting rule this way: from the current node, grow a subtree with a given depth and then, as previously described, shake and merge the terminal leaves of the subtree in order to produce two siblings only. Clearly, this leads to improve the gain in terms of (empirical) AUC compared to the crude perpendicular splitting. Other variants could naturally be considered, focussing on the nodes it is best to split for instance.

Data description. Each class contains gaussian vectors in $\mathbb{R}^d$ with $d = 5$ with different means and same covariance matrix Σ. Theoretical proportions for each class are equal ($p = 1/2$). We consider samples of size $n = 1000$ and run TreeRank with a depth of five layers.

Results. We consider two situations: (i) the optimal separator between the two classes is a hyperplane orthogonal to one of the axes (Figure 4), (ii) the optimal is a linear separator in arbitrary position (Figure 5). The results illustrate both the potential of the TREERANK algorithm for scoring in high dimensions (case (i)) and the weaknesses of a plain application of TREERANK with orthogonal splits (case (ii)). Indeed, in case (ii), the first split is necessarily bad because an orthogonal split is a bad approximation of an arbitrary hyperplane. Hence, for those points which fall on the wrong side of the first split, the algorithm will never be able to rank them correctly, whatever the depth. These results are promising but also motivate further work in the spirit of Section V.

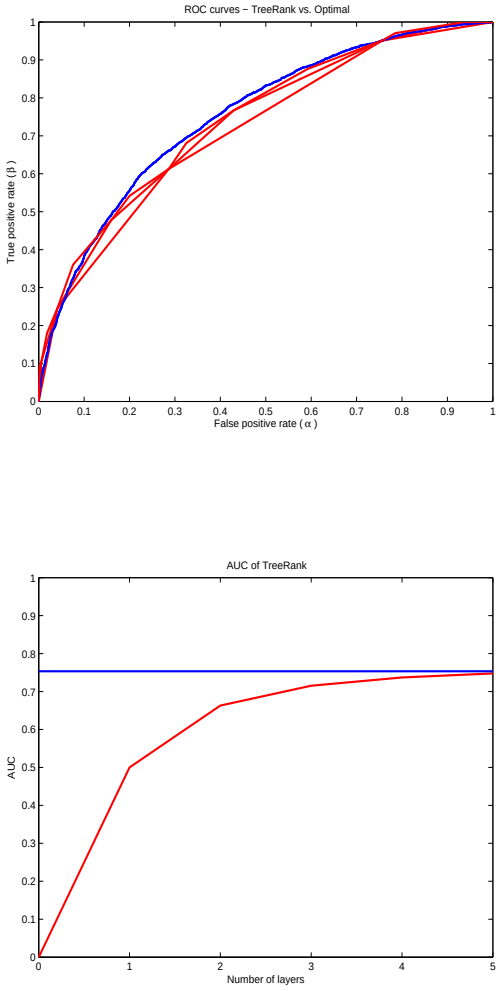

Fig. 4. Case (i) - The linear separator is well-described by a decision stump. Up: Overlaid TREERANK ROC curves over successive iterations (red) vs optimal ROC* curve (blue). Down: TREERANK AUC as a function of the number of layers D (red) compared to the optimal AUC (blue).

VII. CONCLUSION

The ranking problem is characterized by its global nature which is well reflected by function-like optimization criteria such as the ROC curve. The present contribution sets the grounds to develop statistical learning theory for this problem and investigates an algorithm which iteratively builds a piecewise scoring function with a tree-structured partition over the input space. Forthcoming work will attempt to correct the weaknesses of the TREERANK algorithm in the spirit of Section V, but also to further explore curve approximation techniques in the context of ranking methods.

APPENDIX A - PROPERTIES OF ROC CURVES

We now recall some simple properties of ROC curves (see [11], [28]).

Proposition 24 (Properties of the ROC curve)**.** *For any distribution P and any scoring function $s : \mathcal{X} \to \mathbb{R}$, the following properties hold:*

1) **Limit values.** *We have:* $\mathrm{ROC}(s, 0) = 0$ *and* $\mathrm{ROC}(s, 1) = 1$
2) **Invariance.** *For any strictly increasing function $T : \mathbb{R} \to \mathbb{R}$, we have, for all $\alpha \in (0, 1)$:* $\mathrm{ROC}(T \circ s, \alpha) = \mathrm{ROC}(s, \alpha)$.
3) **Concavity.** *If the likelihood ratio dG_s/dH_s is a monotone function then the* ROC *curve is concave.*
4) **Linear parts.** *If the likelihood ratio dG_s/dH_s is constant on some interval in the range of the scoring function s then the* ROC *curve will present a linear part on the corresponding domain. Furthermore,* ROC* *is linear on $[\alpha_1, \alpha_2]$ iff dG/dH is constant on the subset $\{x \in \mathcal{X} / \ Q^*(\alpha_2) \leq \eta(x) \leq Q^*(\alpha_1)\}$.*
5) **Differentiability.** *Assume that the distribution μ of X is*

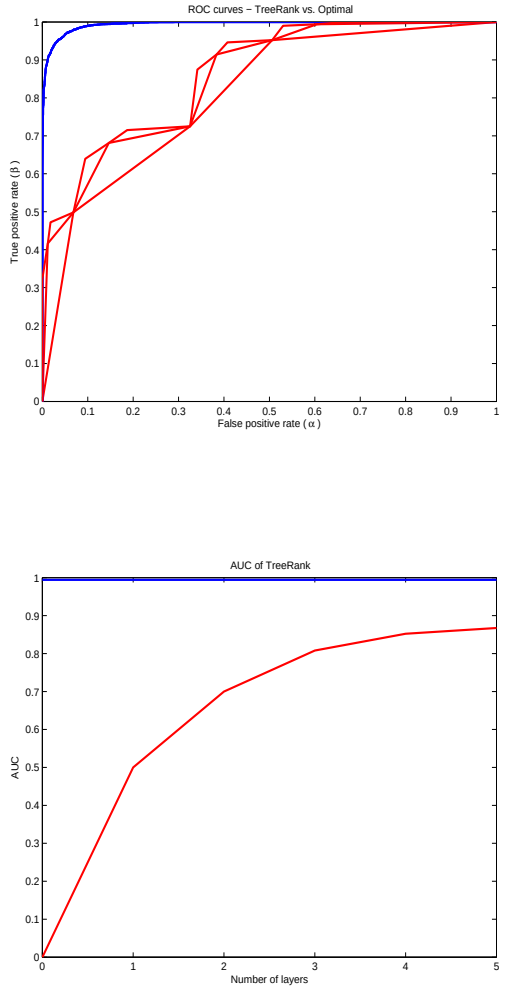

Fig. 5. Case (ii) - The linear separator is in arbitrary position. Up: Overlaid TREERANK ROC curves over successive iterations (red) vs optimal ROC* curve (blue). Down: TREERANK AUC as a function of the number of layers D (red) compared to the optimal AUC (blue).

continuous. Then, the ROC *curve of a scoring function s is differentiable if and only if the conditional distribution of $s(X)$ given Y is continuous.*

APPENDIX B - PROOFS

Proof of Proposition 3

First note that, for any scoring function s with range equal to (m, M), if U has a uniform distribution in (m, M), then:

$$\forall x \in \mathcal{X} \ , \quad \mathbb{E}(\mathbb{I}\{s(x) > U\}) = \frac{s(x) - m}{M - m} \ .$$

Assume that the range of η has no holes. Then for $s^* \in \mathcal{S}^*$ with range equal to $[m, M]$, there exists a strictly increasing function $T : (0, 1) \to [m, M]$ such that $s^* = T \circ \eta$. We have:

$$\forall x \in \mathcal{X} \ , \quad s^*(x) = m + (M - m)\mathbb{E}(\mathbb{I}\{\eta(x) > T^{-1}(U)\}) \ .$$

We can set $V = T^{-1}(U)$ and $w(V) = M - m$, and the 'only if' part is proved in the case where $\eta(X)$ has a support equal to $[0, 1]$. For the general case, we only have to take w to be the indicator of the support of η.

Now assume that s^* has the given form. In order to show that s^* is an optimal scoring function, it suffices to prove that the ordering induced by s on a pair (x, x') is the same as the one induced by η. Denote by ϕ the df of V with respect to the Lebesgue measure. We have:

$$\forall x, x' \in \mathcal{X} \ , \quad s^*(x) - s^*(x') = \int_{\eta(x')}^{\eta(x)} w(v)\phi(v) \ dv \ ,$$

which gives the result since ϕ and w are nonnegative.

Proof of Proposition 6 and Corollary 7

The first part of the proposition is a simple consequence of Neyman-Pearson's lemma formulated in the appropriate setting. For the sake of clarity, we provide a detailed argument. Consider the following hypothesis testing problem: given the observation X, test the null assumption $H_0 \ : \ Y = -1$ against the alternative $H_1 \ : \ Y = +1$. Denote by $p = \mathbb{P}\{Y = 1\}$. The optimal test statistic is then given by the likelihood ratio test:

$$\phi^*(x) = \frac{\mathbb{P}\{X = x \mid Y = 1\}}{\mathbb{P}\{X = x \mid Y = -1\}} = \frac{1 - p}{p} \cdot \frac{\eta(x)}{1 - \eta(x)} \ .$$

Denote by $Q(Z, \alpha)$ the quantile of order $1 - \alpha$ for the distribution of Z conditioned on the event $Y = -1$. By Neyman-Pearson's lemma, we have that among all test statistics $\phi(X)$ with fixed type I error $\alpha = \mathbb{P}\{\phi(X) > Q(\phi(X), \alpha) \mid Y = -1\}$, the test defined by the statistic $\phi^*(X)$ maximizes the power $\beta = \mathbb{P}\{\phi(X) > Q(\phi(X), \alpha) \mid Y = 1\}$. Moreover, the class of distributions $\{\mathbb{P}\{X = x \mid Y = \theta\}\}_{\theta \in \{0,1\}}$ is a monotone likelihood ratio family in $\eta(X)$. Indeed, since the function $u \mapsto \frac{1-p}{p} \cdot \frac{u}{1-u}$ is strictly increasing on $(0, 1)$, the test based on the statistic $\phi^*(X)$ is obviously equivalent to the one based $\eta(X)$. Hence η is an optimal scoring function in the sense of the ROC curve. Any element of the class $\mathcal{S}^*$ will also maximize the ROC curve thanks to the invariance property under strictly increasing transforms.

The last statement of Proposition 6 is proved as follows. First, we use the fact that, for any measurable function h, we have:

$$\mathbb{E}(h(X) \mid Y = +1) = \frac{1 - p}{p}\mathbb{E}\left(\frac{\eta(X)}{1 - \eta(X)} \ h(X) \mid Y = -1\right) \ .$$

We apply this with $h(X) = \mathbb{I}\{X \in R^*_\alpha\} - \mathbb{I}\{X \in R_{s,\alpha}\}$ to get:

$$\begin{aligned} &\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(\mathrm{s}, \alpha) \\ &\qquad = \frac{1 - p}{p}\mathbb{E}\left(\frac{\eta(X)}{1 - \eta(X)} h(X) \mid Y = -1\right) \ . \end{aligned}$$

Then we add and subtract $\frac{Q^*(\alpha)}{1-Q^*(\alpha)}$ and using the fact that

$$1-\alpha = \mathbb{P}\{X \in R_{s,\alpha} \mid Y=-1\} = \mathbb{P}\{X \in R^*_\alpha \mid Y=-1\} \ ,$$

we get:

$$\begin{aligned}&\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(\mathrm{s},\alpha)\\ &= \left(\frac{1-p}{p}\right)\mathbb{E}\left(\left(\frac{\eta(X)}{1-\eta(X)} - \frac{Q^*(\alpha)}{1-Q^*(\alpha)}\right)\ h(X)\middle| Y=-1\right) \ .\end{aligned}$$

We remove the conditioning with respect to $Y=-1$ and using then conditioning on X, we obtain:

$$\begin{aligned}&\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(\mathrm{s},\alpha)\\ &\qquad = \frac{1}{p}\mathbb{E}\left(\left(\frac{\eta(X)-Q^*(\alpha)}{1-Q^*(\alpha)}\right)\ h(X)\right) \ .\end{aligned}$$

It is then easy to see that this expression corresponds to the statement in the Proposition.

Proof of Proposition 8

In the proof of Proposition 6, we saw that the likelihood ratio test statistic was given by:

$$\phi^*(x) = \frac{\mathbb{P}\{X=x \mid Y=1\}}{\mathbb{P}\{X=x \mid Y=-1\}} = \frac{1-p}{p}\cdot\frac{\eta(x)}{1-\eta(x)} \ .$$

Now consider, for any measurable function m, the following conditional expectation with respect to the random variable X given $Y=1$:

$$\mathbb{E}(m(\eta(X)) \mid Y=1) = \mathbb{E}\left(m(\eta(X))\cdot\phi^*(X) \mid Y=-1\right)$$

which can also be expressed as a conditional expectation with respect to the random variable $Z=\eta(X)$ given $Y=1$:

$$\mathbb{E}(m(Z) \mid Y=1) = \mathbb{E}\left(m(Z)\cdot\frac{dG^*}{dH^*}(Z)\middle| Y=-1\right) \ .$$

We can then proceed to the following identification:

$$\phi^*(X) = \frac{dG^*}{dH^*}(\eta(X))$$

We have obtained the following formula for the likelihood ratio of the random variable $\eta(X)$:

$$\forall u \in (0,1) \ , \quad \frac{dG^*}{dH^*}(u) = \frac{1-p}{p}\cdot\frac{u}{1-u} \ ,$$

which gives the result.

Proof of Proposition 10

We recall (see [17]) that:

$$\mathrm{AUC}^* - \mathrm{AUC}(\hat{\eta}) = \frac{\mathbb{E}\left(|\eta(\mathrm{X})-\eta(\mathrm{X}')|\mathbb{I}\{(\mathrm{X},\mathrm{X}')\in\Gamma\}\right)}{2\mathrm{p}(1-\mathrm{p})} \ .$$

where

$$\Gamma = \{(x,x') : \ \mathrm{sgn}(\hat{\eta}(X)-\hat{\eta}(X')) \neq \mathrm{sgn}(\eta(X)-\eta(X'))\}$$

But, one may easily check that:
if $\mathrm{sgn}(\hat{\eta}(X)-\hat{\eta}(X')) \neq \mathrm{sgn}(\eta(X)-\eta(X'))$, then

$$|\eta(X)-\eta(X')| \leq |\eta(X)-\hat{\eta}(X)| + |\eta(X')-\hat{\eta}(X')| \ ,$$

which gives the first part of the result.

Turning to the second assertion, consider the event

$$\mathcal{E} = \{X \in R^*_\alpha \Delta R_{\hat{\eta},\alpha}\}.$$

Notice first that, after Proposition 6, we have:

$$\begin{aligned}\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(\hat{\eta},\alpha) &= \frac{\mathbb{E}(|\eta(X)-Q^*(\alpha)|\ \mathbb{I}_{\mathcal{E}})}{p(1-Q^*(\alpha))}\\ &\leq \frac{c\mathbb{E}(|H^*(\eta(X))-1+\alpha|\ \mathbb{I}_{\mathcal{E}})}{p(1-Q^*(\alpha))}\end{aligned} \ .$$

by virtue of the finite increments theorem. Now, observing that

$$\mathcal{E} = \{\mathrm{sgn}(H^*(\eta(X))-1+\alpha) \neq \mathrm{sgn}(H_{\hat{\eta}}(\hat{\eta}(X))-1+\alpha)\},$$

we have in a similar fashion as above: if $X \in R^*_\alpha \Delta R_{\hat{\eta},\alpha}$, then

$$|H^*(\eta(X))-1+\alpha| \leq |H^*(\eta(X)) - H_{\hat{\eta}}(\hat{\eta}(X))|,$$

which, combined to the previous bound, proves the second part.

Proof of Proposition 15

We now show that the recursive approximation procedure described in Subsection III-D provides a sequence of piecewise constant scoring functions $(s_D)_{D\geq 0}$ with N constant parts which achieves an approximation error rate for the AUC of the order of 2^{-2D}.

For any $\alpha \in (\alpha^*_{D,k}, \alpha^*_{D,k+1})$, we have, for any optimal scoring function s^*, by concavity of η:

$$\begin{aligned}\mathrm{ROC}(\mathrm{s}^*,\alpha) - \mathrm{ROC}(\mathrm{s_D},\alpha) &\leq -\frac{1}{8}(\alpha^*_{D,k+1}-\alpha^*_{D,k})^2\\ &\times \frac{d^2}{d\alpha^2}\mathrm{ROC}(\mathrm{s}^*,\alpha^*_{\mathrm{D,k}}) \ .\end{aligned}$$

By assumption, the second derivative of the optimal ROC is bounded and hence, it suffices to check that, for some constant C, we have:

$$\forall k \ , \quad \alpha^*_{D,k+1} - \alpha^*_{D,k} \leq C\cdot 2^{-D} \ .$$

This inequality follows immediately from a recurrence based on the next lemma.

Lemma 25. *Consider $f : [0,1] \to [0,1]$ a twice differentiable and concave function such that: $m \leq f'' \leq M < 0$. Take x_0, x_1 such that $x_0 < x_1$ and set x_* such that*

$$f'(x_*) = \frac{f(x_1)-f(x_0)}{x_1-x_0} \ .$$

Then, we have:

$$x_* - x_0 \leq C(x_1-x_0)$$

for some constant C which does not depend on x_0, x_1.

Proof: Set the notations: $\Delta f = f(x_1)-f(x_0)$ and $\Delta x = x_1 - x_0$ As f' is continuous and strictly increasing, we can use the following expression for x_*:

$$x_* = f'^{-1}\left(\frac{\Delta f}{\Delta x}\right) \ .$$

By applying the theorem of finite increment to f'^{-1} between and $f'(x_1)$ and $\frac{\Delta f}{\Delta x}$, we have

$$x_* - x_0 = x_1 - x_0 + \left(\frac{\Delta f}{\Delta x} - f'(x_1)\right)(f'^{-1})'(c)$$

for some c. But we also have by Taylor's formula that:

$$\frac{\Delta f}{\Delta x} - f'(x_1) = \frac{1}{2}(x_1 - x_0)f''(c')$$

for some c'. This leads to the result, as $m \leq f'' \leq M < 0$ since:

$$(f'^{-1})' = \frac{f''}{f'' \circ (f')^{-1}} \ .$$

■

Proof of Proposition 19

The ROC curve of $s_{2,t}$ is a broken line with the extremities of the two linear parts being $(0,0)$, $(\alpha(t), \beta(t))$ and $(1,1)$. Hence, the corresponding AUC can be written as:

$$A_2(t) = \frac{1}{2}\left(1 + \beta(t) - \alpha(t)\right) \ .$$

As the ROC curve is differentiable, the maximum of $A_2(t)$ is obtained at the point t^* such that:

$$d\beta(t^*) = d\alpha(t^*) \ ,$$

and hence $\frac{d}{d\alpha}\mathrm{ROC}^*(\alpha^*) = 1$ for $\alpha^* = \alpha(t^*)$. We use Proposition 8 to get $\alpha^* = \bar{H}^*(p)$ and this ultimately leads to $t^* = p$.

Proof of Proposition 21

It suffices to prove the concavity for $D \leq 2$, the general result will be immediately obtained by recurrence. For $D \leq 1$, the result is obviously true. Consider thus the case $D = 2$. By construction, we have for all $C \in \mathcal{C}$,

$$\widehat{\beta}(C) - \widehat{\alpha}(C) \leq \beta_{1,1} - \alpha_{1,1} = \beta_{2,2} - \alpha_{2,2}.$$

Taking successively $C = C_{2,1}$ and $C = R_{2,3} = C_{2,1} \cup C_{2,2} \cup C_{2,3}$ (recall that $\mathcal{C}$ is union stable by assumption), one gets

$$\frac{\beta_{2,3} - \beta_{2,2}}{\alpha_{2,3} - \alpha_{2,2}} \leq 1 \leq \frac{\beta_{2,2} - \beta_{2,1}}{\alpha_{2,2} - \alpha_{2,1}},$$

which yields the desired result.

Proof of Theorem 22 (Sketch of)

The proof of the consistency result in the case of decision trees for classification is based on the control of the excess risk in terms of the L_1-distance between the regression function and its plug-in estimator obtained as a local estimation on one cell. In the case of ranking, we can use a similar argument both for the AUC criterion and the supremum norm over the ROC curves thanks to Proposition 10. For a given sample $\mathcal{D}_n$, consider the sequences of sets $(R_{d,k})_{d,k}$, $(C_{d,k})_{d,k}$ and the sequences $\{(\alpha_{d,k}, \beta_{d,k})\}_{d,k}$ arising from a run of TREERANK with depth $N = 2^D$. We can then deal with the two metrics in a similar way:

- L_1 metric (AUC) - we can consider the following plug-in estimator of the regression function (see Remark 8):

$$\hat{\eta}(x) = \sum_{k=0}^{2^D-1} \Delta_{D+1,2k+1} \ \mathbb{I}\{x \in C_{D,k}\},$$

where

$$\Delta_{D+1,2k+1} = \frac{n_+(\beta_{d,k+1} - \beta_{d,k})}{n_-(\alpha_{d,k+1} - \alpha_{d,k}) + n_+(\beta_{d,k+1} - \beta_{d,k})} \ .$$

Then use the inequality from Proposition 10:

$$\mathrm{AUC}^* - \mathrm{AUC}(\hat{\eta}) \leq \frac{1}{\mathrm{p}(1-\mathrm{p})}\mathbb{E}\left(|\hat{\eta}(\mathrm{X}) - \eta(\mathrm{X})|\right) \ .$$

- L_∞ metric - here we introduce the estimator:

$$\hat{s}(x) = \sum_{j=0}^{2^D-1} (\alpha_{D,j+1} - \alpha_{D,j}) \ \mathbb{I}\{x \in R_{D,j}\}$$

for $H^* \circ \eta$. But we have, by construction:

$$\mathbb{I}\{x \in R_{D,j}\} = \sum_{k=0}^{j} \mathbb{I}\{x \in C_{D,k}\} \ .$$

Then we have, also by Proposition 10, for any α:

$$\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(\hat{\mathrm{s}}, \alpha) \leq \frac{\mathrm{c}\mathbb{E}\left(|\mathrm{H}^*(\eta(\mathrm{X})) - \hat{\mathrm{s}}(\mathrm{X}))|\right)}{\mathrm{p}(1 - \mathrm{Q}^*(\alpha))}.$$

Now denote by j_0 the index of the set such that $x \in C_{D,j_0}$, then $\hat{\eta}(x) = \Delta_{D+1,2j_0+1}$ and $\hat{s}(x) = 1 - \alpha_{D,j_0}$. Note also that

$$\alpha_{D,j_0} = \frac{1}{n_-}\sum_{i=1}^{n} \mathbb{I}\{X_i \in C_{D,j_0}, Y_i = -1\}$$
$$\beta_{D,j_0} = \frac{1}{n_+}\sum_{i=1}^{n} \mathbb{I}\{X_i \in C_{D,j_0}, Y_i = 1\} \ .$$

This observation indicates that the same argument will work for the two metrics. From there, the rest of the proof is exactly as in Theorem 21.2 from [23], except that n_+, n_- are random. We can write, for instance:

$$\frac{1}{n_-} = \frac{1}{n_-} - \frac{1}{n(1-p)} + \frac{1}{n(1-p)} \ ,$$

and we can see that there will be a corrective term of the order of $n^{-1/2}$ which will not affect the convergence.

Proof of Theorem 23

As a first go, we consider the L_1 case, the result concerning the sup norm shall appear as a consequence of the next argument.

The AUC **case.** The proof immediately follows from the next lemma, combined with the proof of Proposition 15.

Lemma 26. *Under the assumptions of Theorem 23, there exist constants κ_1, κ_2, c_1 and c_2 such that, for all $\delta > 0$, we have with probability at least $1 - \delta$: $\forall d \in \mathbb{N}$, $\forall n \in \mathbb{N}$,*

$$|\mathrm{AUC}(\mathrm{s}_\mathrm{d}^*) - \mathrm{AUC}(\mathrm{s}_\mathrm{d})| \leq \kappa_1^{\mathrm{d}-1}\mathrm{B}(\mathrm{d}, \mathrm{n}, \delta),$$

and $\forall k \in \{0,\ldots,2^{d-1}-1\}$,

$$|\alpha(C^*_{d,2k}) - \alpha(C_{d,2k})| + |\beta(C^*_{d,2k}) - \beta(C_{d,2k})| \leq \kappa_2^d B(d+1,n,\delta),$$

where: $\forall (d,n,\delta) \in \mathbb{N}\times\mathbb{N}\times]0,1[$,

$$B(d,n,\delta) = \left(\frac{c_1^2 V}{n}\right)^{\frac{1}{2d}} + \left(\frac{c_2^2\log(1/\delta)}{n}\right)^{\frac{1}{2d}}.$$

For $d=0$, the result is obvious. The general version can be established by recurrence. Here we shall detail the transition from $d=1$ to $d=2$. Let us introduce the notation: for all $C\in\mathcal{C}$, $d\in\mathbb{N}$ and $k\in\{1,\ldots,2^d\}$,

$$\begin{aligned}\Lambda^*_{d,k}(C) &= \alpha(C^*_{d,k-1})\beta(C) - \beta(C^*_{d,k-1})\alpha(C),\\ \widetilde{\Lambda}_{d,k}(C) &= \alpha(C_{d,k-1})\beta(C) - \beta(C_{d,k-1})\alpha(C).\end{aligned}$$

Equipped with this notation, we can bound the deviation $2|\mathrm{AUC}(\mathrm{s}^*_\mathrm{D}) - \mathrm{AUC}(\mathrm{s}_\mathrm{D})|$ by

$$\sum_{k=0}^{2^{D-1}-1} |\Lambda^*_{D-1,k+1}(C^*_{D,2k}) - \widetilde{\Lambda}_{D-1,k+1}(C_{D,2k})|.$$

First iteration $d=1$**.** We have

$$\begin{aligned}2\mathrm{AUC}(\mathrm{s}^*_1) - 1 &= \Lambda^*_{0,1}(C^*_{1,0}) = \beta^*_{1,1} - \alpha^*_{1,1},\\ 2\mathrm{AUC}(\mathrm{s}_1) - 1 &= \Lambda^*_{0,1}(C_{1,0}) = \widetilde{\Lambda}_{0,1}(C_{1,0}).\end{aligned}$$

In the first place, notice that

$$\mathrm{AUC}(\mathrm{s}^*_1) - \mathrm{AUC}(\mathrm{s}_1) \geq 0.$$

Indeed, since ROC^* dominates any true ROC curve everywhere, observe that

$$\beta(C_{1,0}) - \alpha(C_{1,0}) \leq \mathrm{ROC}^*(\alpha(\mathrm{C}_{1,0})) - \alpha(\mathrm{C}_{1,0}).$$

and recall that $\alpha^*_{1,1} = \arg\max_{\alpha\in(0,1)}\{\mathrm{ROC}^*(\alpha)-\alpha\}$.

Now, write

$$2\{\mathrm{AUC}(\mathrm{s}^*_1) - \mathrm{AUC}(\mathrm{s}_1)\} = (\mathrm{I}) + (\mathrm{II}) + (\mathrm{III}), \qquad (1)$$

where

$$\begin{aligned}(I) &= \Lambda^*_{0,1}(C^*_{1,0}) - \Lambda_{0,1}(C^*_{1,0})\\ (II) &= \Lambda_{0,1}(C^*_{1,0}) - \Lambda_{0,1}(C_{1,0})\\ (III) &= \Lambda_{0,1}(C_{1,0}) - \Lambda^*_{0,1}(C_{1,0}).\end{aligned}$$

By definition, one has $(II)\leq 0$, while (I) and (III) are both bounded by

$$\sup_{C\in\mathcal{C}} |\alpha(C) - \hat{\alpha}(C)| + \sup_{C\in\mathcal{C}} |\beta(C) - \hat{\beta}(C)|.$$

Let $\delta>0$. Consequently, using twice the VC inequality for the expectation of a supremum (see [**?**]), we obtain that, with probability at least $1-\delta$: $\forall n\in\mathbb{N}$,

$$\begin{aligned}\mathrm{AUC}(\mathrm{s}^*_1) - \mathrm{AUC}(\mathrm{s}_1) &\leq c_1\sqrt{\frac{V}{n}} + c_2\sqrt{\frac{\log(1/\delta)}{n}}\\ &= B(1,n,\delta),\end{aligned}$$

Using a Taylor-Lagrange expansion of $\alpha\mapsto \mathrm{ROC}^*(\alpha)-\alpha$ around $\alpha^*_{1,1}$ at the second order, we get that $\{\mathrm{ROC}^*(\alpha^*_{1,1}) - \alpha^*_{1,1}\} - \{\mathrm{ROC}^*(\alpha(\mathrm{C}_{1,0})) - \alpha(\mathrm{C}_{1,0})\}$ is equal to

$$-\frac{1}{2}\mathrm{ROC}^{*''}(\tilde{\alpha})(\alpha^*_{1,1} - \alpha(\mathrm{C}_{1,0}))^2,$$

for a certain $\tilde{\alpha}$ between $\alpha(C_{1,0})$ and $\alpha^*_{1,1}$. Besides, using again that ROC^* dominates any other true ROC curve (so that $\beta(C_{1,0}) \leq \mathrm{ROC}^*(\alpha(\mathrm{C}_{1,0}))$), it is also bounded by the deviation $2\{\mathrm{AUC}(\mathrm{s}^*_1) - \mathrm{AUC}(\mathrm{s}_1)\}$). We set $m = -\sup_{\alpha\in[0,1]}\mathrm{ROC}^{*''}(\alpha)$. Combined with the bound previously established, we obtain that, for all $\delta>0$, we have with probability larger than $1-\delta$: $\forall n\in\mathbb{N}$,

$$|\alpha^*_{1,1} - \alpha(C_{1,0})| \leq \frac{2}{\sqrt{m}}\sqrt{B(1,n,\delta)} \leq \frac{2}{\sqrt{m}}B(2,n,\delta).$$

By the triangular inequality, we also have with probability larger than δ: $\forall n\in\mathbb{N}$,

$$|\beta^*_{1,1} - \beta(C_{1,0})| \leq \frac{2}{\sqrt{m}}B(2,n,\delta) + B(1,n,\delta).$$

Hence, $\forall n \geq n_\delta = \max\{c_1^2 V, c_2^2\log(1/\delta)\}$, with probability at least $1-\delta$, we have

$$|\alpha(C^*_{1,1}) - \alpha(C_{1,0})| + |\beta(C^*_{1,1}) - \beta(C_{1,0})| \leq \kappa_2 B(2,n,\delta),$$

with $\kappa_2 = 6/\sqrt{m}$. This suggests that the deviation at the next iteration should be of order $O_{\mathbb{P}}(n^{-1/4})$.

Recurrence. Let $d\geq 1$ be fixed. We temporarily suppose that the bounds stated in Lemma 26 hold for $d-1$. For all $k\in\{0,\ldots,2^{d-1}-1\}$, set

$$\bar{C}_{d,2k} = \arg\max_{C\subset C_{d-1,k}} \widetilde{\Lambda}_{d-1,k+1}(C).$$

Write

$$\Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(C_{d,2k}) = I_{d,2k} + J_{d,2k},$$

where

$$\begin{aligned}I_{d,2k} &= \Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(\bar{C}_{d,2k}),\\ J_{d,2k} &= \widetilde{\Lambda}_{d-1,k+1}(\bar{C}_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(C_{d,2k}).\end{aligned}$$

Reproducing the argument used for $d=1$, we obtain that, for all $\delta>0$, we have with probability at least $1-\delta$: $\forall n\in\mathbb{N}$, $\forall k\in\{0,\ldots,2^{d-1}-1\}$,

$$0 \leq J_{d,2k} \leq B(1,n,\delta).$$

Consider the first term now and write $C^*_{d,2k} = A^*_{d,2k}\cup B^*_{d,2k}$ with

$$\begin{aligned}A^*_{d,2k} &= C^*_{d,2k}\cap(C^*_{d-1,k}\setminus C_{d-1,k}),\\ B^*_{d,2k} &= C^*_{d,2k}\cap C_{d-1,k}.\end{aligned}$$

Similarly, set $C_{d,2k} = A_{d,2k}\cup B_{d,2k}$ where

$$\begin{aligned}A_{d,2k} &= C_{d,2k}\cap(C_{d-1,k}\setminus C^*_{d-1,k}),\\ B_{d,2k} &= C_{d,2k}\cap C^*_{d-1,k}.\end{aligned}$$

By additivity of the entropy measures, we have:

$$\begin{aligned} I_{d,2k} &= \Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(C^*_{d,2k}) \\ &+ \widetilde{\Lambda}_{d-1,k+1}(B^*_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(\bar{C}_{d,2k}) \\ &+ \widetilde{\Lambda}_{d-1,k+1}(A^*_{d,2k}) \\ &\leq \Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(C^*_{d,2k}) \\ &+ \widetilde{\Lambda}_{d-1,k+1}(A^*_{d,2k}). \end{aligned}$$

By virtue of the recurrence assumption, we have, with probability larger than $1-\delta$: $\forall n \geq n_\delta$,

$$\begin{aligned} &\Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(C^*_{d,2k}) \\ &\quad \leq |\alpha(C^*_{d-1,k}) - \alpha(C_{d-1,k})| + |\beta(C^*_{d-1,k}) - \beta(C_{d-1,k})| \\ &\qquad \leq \kappa_2^{d-1} B(d,n,\delta), \end{aligned}$$

and exactly the same bound holds for $\widetilde{\Lambda}_{d-1,k+1}(A^*_{d,2k})$.

Symmetrically,

$$\begin{aligned} I_{d,2k} &= \Lambda^*_{d-1,k+1}(\bar{C}_{d,2k})) - \widetilde{\Lambda}_{d-1,k+1}(\bar{C}_{d,2k}) \\ &+ \Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \Lambda^*_{d,2k}(B_{2,k}) \\ &+ \Lambda^*_{d-1,k+1}(A_{d,2k}) \\ &\geq \Lambda^*_{d-1,k+1}(\bar{C}_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(\bar{C}_{d,k}) \\ &+ \Lambda^*_{d-1,k+1}(A_{d,2k}). \end{aligned}$$

Using the same argument as above, we eventually get that: $\forall n \geq n_\delta$,

$$|\Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \widetilde{\Lambda}_{d-1,k+1}(C^*_{d,2k})| \leq \kappa_2^{d-1} B(d,n,\delta).$$

Hence, we have: $\forall n \geq n_\delta$,

$$\mathrm{AUC}(\mathrm{s}^*_\mathrm{d}) - \mathrm{AUC}(\mathrm{s}_\mathrm{d}) \leq (2\kappa_2)^{\mathrm{d}-1}\mathrm{B}(\mathrm{d},\mathrm{n},\delta).$$

Now it remains to control the deviation $\alpha(C^*_{d,2k}) - \alpha(C_{d,2k})$. Given that the quantity

$$\alpha(C^*_{d-1,k})(\mathrm{ROC}^*(\alpha) - \beta^*_{\mathrm{d}-1,\mathrm{k}}) - \beta(\mathrm{C}^*_{\mathrm{d}-1,\mathrm{k}})(\alpha - \alpha^*_{\mathrm{d}-1,\mathrm{k}})$$

is maximum for $\alpha = \alpha^*_{d,2k+1}$, the deviation

$$\begin{aligned} \Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \alpha(C^*_{d-1,k})\mathrm{ROC}^*(\alpha(\mathrm{B}_{\mathrm{d},2\mathrm{k}}) + \alpha_{\mathrm{d}-1,\mathrm{k}}) - \ldots \\ \ldots - \alpha(C^*_{d-1,k})\beta^*_{d-1,k} - \beta(C^*_{d,k})\alpha(B_{d,2k}) \end{aligned}$$

is positive. Besides, it is also bounded by $\Lambda^*_{d-1,k+1}(C^*_{d,2k}) - \Lambda^*_{d-1,k+1}(B_{d,2k})$. Indeed, using the optimality of ROC^*, we have:

$$\beta(B_{d,2k}) \leq \mathrm{ROC}^*(\alpha(\mathrm{B}_{\mathrm{d},2\mathrm{k}}) + \alpha_{\mathrm{d}-1,\mathrm{k}}) - \beta^*_{\mathrm{d}-1,\mathrm{k}}\ .$$

It is thus bounded by $\kappa_2^{d-1} B(d,n,\delta)$. Now a Taylor expansion of

$$\alpha \mapsto \alpha(C^*_{d,k})(\mathrm{ROC}^*(\alpha) - \beta^*_{\mathrm{d}-1,\mathrm{k}}) - \beta(\mathrm{C}^*_{\mathrm{d},\mathrm{k}})(\alpha - \alpha^*_{\mathrm{d}-1,\mathrm{k}})$$

at the second order around $\alpha^*_{d,2k+1} = \alpha^*_{d-1,k} + \alpha(C^*_{d,2k})$ implies that there exists $\bar{\alpha}$ between $\alpha(C^*_{2,k})$ and $\alpha(B_{d,2k})$ such that

$$\begin{aligned} \Lambda^*_{d-1,k+1}(C^*_{d,2k}) &- \{\alpha^*_{d-1,1}\mathrm{ROC}^*(\alpha(\mathrm{B}_{\mathrm{d},2\mathrm{k}})) - \beta^*_{1,1}\alpha(\mathrm{B}_{\mathrm{d},2\mathrm{k}})\} \\ &= -\frac{1}{2}\alpha(C^*_{d-1,k})\mathrm{ROC}^{*''}(\bar{\alpha} + \alpha^*_{\mathrm{d}-1,\mathrm{k}}) \\ &\times (\alpha(C^*_{d,2k}) - \alpha(B_{d,2k}))^2. \end{aligned}$$

We thus get that: $\forall k \in \{0,\ldots,2^{d-1}-1\}$,

$$\begin{aligned} |\alpha(C^*_{d,2k}) - \alpha(C_{d,2k})| &\leq |\alpha(C^*_{d,2k}) - \alpha(B_{d,2k})| \\ &+ |\alpha(C_{d,2k}) - \alpha(B_{d,2k})| \\ &\leq \sqrt{\frac{2\kappa_2^{d-1}}{\alpha(C^*_{d-1,k})m}} B(d+1,n,\delta) \\ &+ \kappa_2^{d-1} B(d,n,\delta). \end{aligned}$$

Therefore, reproducing the argument used for proving Lemma 26, one may show that there exists a constant $c < \infty$ such that: $\forall d \geq 1$, $\alpha(C^*_{d,k}) = \alpha^*_{d,k+1} - \alpha^*_{d,k} \geq c \cdot 2^{-d}$. We eventually obtain

$$|\alpha(C^*_{d,2k}) - \alpha(C_{d,2k})| \leq \sqrt{\frac{2^d \kappa_2^{d-1}}{c}} B(d+1,n,\delta).$$

It follows that

$$\begin{aligned} |\beta(C^*_{d,2k}) - \beta(C_{d,2k})| &\leq \frac{1}{\alpha(C^*_{d-1,k})}\{\sqrt{\frac{2^d\kappa_2^{d-1}}{c}} B(d+1,n,\delta) \\ &+ \kappa_2^d B(d,n,\delta)\} \\ &\leq \frac{2^d}{c}\sqrt{\frac{2^d\kappa_2^{d-1}}{c}} B(d+1,n,\delta). \end{aligned}$$

Hence, we have $\forall k \in \{0,\ldots,2^{d-1}-1\}$,

$$\begin{aligned} |\alpha(C^*_{d,2k}) - \alpha(C_{d,2k})| &+ |\beta(C^*_{d,2k}) - \beta(C_{d,2k})| \\ &\leq \kappa_2^d B(d+1,n,\delta), \end{aligned}$$

provided that κ_2 is chosen large enough. The desired bounds are thus proved at level d, which establishes the result by recurrence.

The *sup norm* case. The rate bound related to the L_∞ distance follows immediately from the fact that deviation $d_\infty(s_D, s^*_D)$ between the two piecewise linear curves may be bounded by

$$\max_{1 \leq k \leq 2^D - 1} \{\beta^*_{D,k} - \beta(R_{D,k-1}) + \mathrm{ROC}^{*'}(1)(\alpha^*_{\mathrm{D},\mathrm{k}} - \alpha(\mathrm{R}_{\mathrm{D},\mathrm{k}-1}))\},$$

combined with the next lemma.

Lemma 27. *Under the assumptions of Theorem 23, there exists a constant K such that, for all $\delta > 0$, we have with probability at least $1-\delta$: for all $d \geq 1$, $k \in \{1,\ldots,2^{d-1}-1\}$,*

$$|\alpha^*_{d,k} - \alpha(R_{d,k-1})| + |\beta^*_{d,k} - \beta(R_{d,k-1})| \leq K^d B(d+1,n,\delta).$$

It may be easily derived by recurrence from Lemma 26, the proof is omitted.

REFERENCES

[1] R. Herbrich, T. Graepel, P. Bollmann-Sdorra, and K. Obermayer, "Learning a preference relation for information retrieval," in *Proceedings of the AAAI Workshop Text Categorization and Machine Learning*, 1998.

[2] W. Cohen, R. Schapire, and Y. Singer, "Learning to order things," in *NIPS '97: Proceedings of the 1997 conference on Advances in neural information processing systems 10*. Cambridge, MA, USA: MIT Press, 1998, pp. 451–457.

[3] M. desJardins, E. Eaton, and K. Wagstaff, "Learning user preferences for sets of objects," in *Proceedings of the Twenty-Third International Conference (ICML 2006)*, 2006, pp. 273–280.

[4] R. Herbrich, T. Graepel, and K. Obermayer, *Advances in Large Margin Classifiers*. MIT Press, 2000, ch. Large margin rank boundaries for ordinal regression, pp. 115–132.
[5] K. Crammer and Y. Singer, "Pranking with ranking," in *Proceedings of the conference on Neural Information Processing Systems (NIPS)*, 2001.
[6] Y. Freund, R. D. Iyer, R. E. Schapire, and Y. Singer, "An efficient boosting algorithm for combining preferences," *Journal of Machine Learning Research*, vol. 4, pp. 933–969, November 2003.
[7] 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*, vol. 6, pp. 393–425, 2005.
[8] S. Clémençon, G. Lugosi, and N. Vayatis, "Ranking and scoring using empirical risk minimization." in *Proceedings of COLT 2005*, ser. Lecture Notes in Computer Science, P. Auer and R. Meir, Eds., vol. 3559. Springer, 2005, pp. 1–15.
[9] T. Hastie and R. Tibshirani, *Generalized Additive Models*. Chapman & Hall/CRC, 1990.
[10] D.M.Green and J. Swets, *Signal detection theory and psychophysics.* Wiley, 1966.
[11] H. van Trees, *Detection, Estimation, and Modulation Theory, Part I*. John Wiley, 1968.
[12] J. Egan, *Signal Detection Theory and ROC Analysis*. Academic Press, 1975.
[13] J. Hanley and J. McNeil, "The meaning and use of the area under a ROC curve," *Radiology*, no. 143, pp. 29–36, 1982.
[14] C. Cortes and M. Mohri, "Auc optimization vs. error rate minimization," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. MIT Press, Cambridge, MA, 2004.
[15] A. Rakotomamonjy, "Optimizing area under roc curve with svms," in *Proceedings of the First Workshop on ROC Analysis in AI*, 2004.
[16] L. Yan, R. Dodier, M. Mozer, and R. Wolniewicz, "Optimizing classifier performance via an approximation to the wilcoxon-mann-whitney statistic," in *Proceedings of the Twentieth International Conference on Machine Learning (ICML 2003)*, T. Fawcett and N. Mishra, Eds., 2003, pp. 848–855.
[17] S. Clémençon, G. Lugosi, and N. Vayatis, "Ranking and empirical risk minimization of U-statistics," *The Annals of Statistics*, vol. To appear, To appear.
[18] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Wadsworth and Brooks, 1984.
[19] C. Ferri, P. Flach, and J. Hernández-Orallo, "Learning decision trees using the area under the roc curve," in *ICML '02: Proceedings of the Nineteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002, pp. 139–146.
[20] F. Provost and P. Domingos, "Tree induction for probability-based ranking," *Machine Learning*, vol. 52, no. 3, pp. 199–215, 2003.
[21] F. Xia, W. Zhang, and J. Wang, "An effective tree-based algorithm for ordinal regression," *IEEE Intelligent Informatics Bulletin*, vol. 7, no. 1, pp. 22–26, December 2006.
[22] S. Clémençon and N. Vayatis, "Ranking the best instances," *Journal of Machine Learning Research*, vol. 8, pp. 2671–2699, 2007.
[23] L. Devroye, L. Györfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*. Springer, 1996.
[24] L. Györfi, M. Köhler, A. Krzyzak, and H. Walk, *A Distribution-Free Theory of Nonparametric Regression*. Springer, 2002.
[25] R. Devore and G. Lorentz, *Constructive Approximation*. Springer, 1993.
[26] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 26, pp. 123–140, 1996.
[27] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *Annals of Statistics*, vol. 28, pp. 337–407, 2000.
[28] D. Hsieh and B. Turnbull, "Nonparametric and semiparametric estimation of the receiver operating characteristic curve," *The Annals of Statistics*, vol. 24, pp. 25–40, 1996.