Ranking Forests

Stéphan Clémençon

To cite this version:

Stéphan Clémençon. Ranking Forests. 2010. hal-00452577

HAL Id: hal-00452577

https://hal.science/hal-00452577

Preprint submitted on 2 Feb 2010

Ranking Forests

Stéphan Clémençon, *Member, IEEE*

Abstract—It is the goal of this paper to examine how the aggregation and feature randomization principles underlying the algorithm RANDOM FOREST [1], originally proposed in the classification/regression setup, can be adapted to *bipartite ranking*, in order to increase the performance of scoring rules produced by the TREERANK algorithm [2], a recently developed tree induction method, specifically tailored for this global learning problem. Since TREERANK may be viewed as a recursive implementation of a cost-sensitive version of the popular classification algorithm CART [3], with a cost locally depending on the data lying within the node to split, various strategies can be considered for "randomizing" the features involved in the tree growing stage. In parallel, several ways of combining/averaging ranking trees may be used, including techniques inspired from rank aggregation methods recently popularized in Web applications. Ranking procedures based on such approaches are called RANKING FORESTS. Beyond preliminary theoretical background, results of experiments based on simulated data are provided in order to give evidence of their statistical performance.

Index Terms—Bipartite Ranking, data with binary labels, ROC optimization, AUC criterion, tree-based ranking rules, bootstrap, bagging, rank aggregation, median procedure, feature randomization.

1 INTRODUCTION

IN the context of classification/regression, the method called RANDOM FORESTS and introduced in [1] has led to considerable improvements in terms of prediction performance. As highlighted by various empirical studies (see [4], [5] for instance), RANDOM FOREST has emerged as a serious competitor to ADABOOST and is considered as one of the best off-the shelf classification/regression techniques. Our purpose is here to examine how the principles underlying RANDOM FOREST, feature randomization and bootstrap aggregation namely, can be applied in the *bipartite ranking* setup.

In many applications involving input data X in a space $\mathcal{X} \subset \mathbb{R}^q$, $q \geq 1$, assigned to a binary label information $Y \in \{-1,+1\}$, the goal is to order/rank the instances $x \in \mathcal{X}$ by order of magnitude of the posterior distribution $\mathbb{P}\{Y = +1 \mid X = x\}$, rather than simply classifying them as positive or negative. This task is known as *bipartite ranking*. From a practical perspective, orderings are generally derived from a *scoring function* $s : \mathcal{X} \rightarrow \mathbb{R}$, transporting the natural order on the real line onto $\mathcal{X}$, and their accuracy is evaluated through ROC analysis, see [6]. For this reason, the global learning problem that consists in building an accurate ordering from a sample of independent copies of the pair (X, Y) is also called *nonparametric scoring*, sometimes. Though easy to formulate, it covers a wide variety of applications: medical diagnosis, anomaly detection, design of search engines, credit-scoring, *etc.*

In spite of its ubiquitousness, nonparametric scoring is in general considered from the *plug-in* angle: this approach, confronted with the *curse of dimensionality*, consists in inferring the posterior probability from sampling data and using then the resulting estimate as a scoring function. In recent years, nonparametric scoring has been the subject of a good deal of attention in the machine-learning literature, with the aim to extend the *Empirical Risk Minimization* approach to the bipartite ranking setup mainly, see [7], [8], [9], [10] for instance. From a computational perspective, apart from the plug-in approach, the vast majority of learning techniques used for building scoring functions rests on combining classifiers (*i.e.* binary scoring functions), see [11], [12], [13]. However, significant advances have recently been made with the development of the TREERANK algorithm [14], a tree induction method specifically tailored for bipartite ranking. In [2], generalization bounds in the ROC space equipped with the *sup-norm* have been established for the ROC curve of the tree-structured scoring rule output by TREERANK under mild conditions and practical issues related to the splitting procedure, as well as the pruning stage, involved in the ranking algorithm are dealt with in [15].

In the same way RANDOM FOREST makes use of classification trees to drastically reduce misclassification error, we shall examine throughout this paper how to combine feature randomization and bootstrap aggregation techniques based on the ranking trees produced by the TREERANK algorithm in order to increase ranking performance. In constrast to the classification/regression setup, the question of aggregating predictions is far from trivial in the ranking framework. The rank aggregation issue, originally introduced in social choice theory (see [16] and the references therein) and recently "rediscovered" in the context of Web applications [17], can be addressed in a variety of ways, including *metric-based consensus methods*, which the approach we develop here pertains to. In addition, feature randomization can be incorporated at two different levels here: at each node of

- *S. Clémençon is with the LTCI UMR Telecom ParisTech/CNRS No. 5141, 46 rue Barrault, 75634 Paris Cedex, France.*
E-mail: stephan.clemencon@telecom-paristech.fr

the ranking tree and/or at all nodes of the cost-sensitive classification trees describing the splits of the ranking tree leaves. In this paper, beyond the description of a novel ranking methodology, RANKING FOREST, using bagging and random selection of features, some theoretical foundations for rank aggregation in connection with AUC optimization are given, together with simulation results, illustrating how the present approach simultaneously enhances stability and ranking accuracy.

The article is structured as follows. Section 2 sets out the notations and shortly describes the crucial notions required to rigorously formulate the bipartite ranking problem. The concept of *ranking tree* and the TREERANK algorithm, which form the base for the present work, are also briefly reviewed. Section 3 investigates how to extend use of the key ingredients of RANDOM FOREST in order to produce an ensemble of ranking trees, which we call a "ranking forest" and combine them so as to improve on the performance of single ranking trees in regards to the standard AUC criterion. Statistical results guaranteeing the validity of the RANKING FOREST approach are stated in Section 4. In order to explore its performance from an empirical angle, Section 5 presents numerical results based on simulated data. Finally, some concluding remarks are collected in Section 6. Technical details and proofs are deferred to the Appendix.

2 BACKGROUND AND PRELIMINARIES

This section first briefly reviews basic concepts related to bipartite ranking, paying special attention to ROC analysis, a popular way of evaluating the capacity of a given scoring rule to discriminate between two populations [6]. Since our interest here focuses on ranking rules defined through a combination of *tree-structured scoring functions*, we precisely define the notion of tree-based ranking rule and recall the heuristics behind the TREERANK algorithm proposed in [2], [14], a recent recursive partitioning method, specifically designed for ROC optimization and maximizing thus the AUC criterion, the gold standard for summarizing ranking performance in most applications.

2.1 Bipartite Ranking

The probabilistic setup is exactly the same as the one in standard binary classification. The random variable Y is a binary label, valued in $\{-1,+1\}$ say, while the random vector $X=(X^{(1)},\ldots,X^{(q)})$ models some multivariate observation for predicting Y, taking its values in a high-dimensional space $\mathcal{X}\subset\mathbb{R}^q$, $q\geq 1$. The probability measure on the underlying space is entirely described by the pair (μ,η), where μ denotes X's marginal distribution and $\eta(x)=\mathbb{P}\{Y=+1 \mid X=x\}$, $x\in\mathcal{X}$, the posterior probability. Alternatively, it is entirely determined by the triplet (G,H,p) where G (respectively, H) is X's conditional distribution given $Y=+1$ (respectively, given $Y=-1$) and $p=\mathbb{P}\{Y=+1\}$.

An informal way of considering the ranking task is as follows. The goal is to learn from the observation of a sample of independent copies of the pair (X,Y) how to order/rank novel data $X_1,\ldots,X_m$ with no knowledge of their labels, so that positive instances are high on the resulting list with large probability. Undoubtedly, the most natural way of defining a total order on the multidimensional space $\mathcal{X}$ is to transport the natural order on the real line by means of a *scoring function*, *i.e.* a measurable mapping $s:\mathcal{X}\rightarrow\mathbb{R}$. A preorder $\preccurlyeq_s$ on $\mathcal{X}$ is then defined by: $\forall(x,x')\in\mathcal{X}^2$, $x\preccurlyeq_s x'$ iff $s(x)\leq s(x')$. We denote by $\mathcal{S}$ the set of such functions. The capacity of a candidate $s\in\mathcal{S}$ to discriminate between the positive and negative populations is generally evaluated by means of its ROC curve (standing for *Receiver Operating Characteristic* curve), a widely used functional performance measure which we recall below for clarity.

Definition 2.1 (TRUE ROC CURVE) *Let $s\in\mathcal{S}$. The true* ROC *curve of the scoring function s is the "probability-probability" plot given by:*

$$t\in\mathbb{R}\mapsto\left(\mathbb{P}\left\{s(X)>t \mid Y=-1\right\},\mathbb{P}\left\{s(X)>t \mid Y=1\right\}\right).$$

By convention, when a jump occurs, the corresponding extremities of the curve are connected by a line segment, so that $s(x)$'s ROC *curve can be viewed as the graph of a continuous mapping $\alpha\in[0,1]\mapsto\mathrm{ROC}(\mathrm{s},\alpha)$.*

We refer to [2] for a detailed list of properties of ROC curves (see the Appendix section therein). Clearly, this criterion provides a useful visual tool for assessing ranking performance: the closer to the left upper corner of the unit square $[0,1]^2$ the curve $\mathrm{ROC}(\mathrm{s},.)$, the better the scoring function s. It thus leads to a partial order on the set of all scoring functions: for all $(s_1,s_2)\in\mathcal{S}^2$, s_2 is said more accurate than s_1 when $\mathrm{ROC}(\mathrm{s}_1,\alpha)\leq\mathrm{ROC}(\mathrm{s}_2,\alpha)$ for all $\alpha\in[0,1]$. By standard Neyman-Pearson argument, one may classically establish that the most accurate scoring functions are increasing transforms of the regression function, *i.e.* the elements of the set $\mathcal{S}^*=\{T\circ\eta,\ \ T:[0,1]\rightarrow\mathbb{R}$ strictly increasing$\}$, see Proposition 4 in [2].

The performance of a candidate $s\in\mathcal{S}$ is usually summarized by a scalar quantity, the *area under the* ROC *curve* (AUC in short):

$$\mathrm{AUC}(\mathrm{s})=\int_{\alpha=0}^{1}\mathrm{ROC}(\mathrm{s},\alpha)\mathrm{d}\alpha.$$

Beyond the fact that it provides a *total order* on the set $\mathcal{S}$, the major interest of this criterion lies in its well-known probabilistic interpretation. Indeed, recall that

$$\begin{aligned}\mathrm{AUC}(\mathrm{s})=&\ \mathbb{P}\{\mathrm{s}(\mathrm{X}_1)<\mathrm{s}(\mathrm{X}_2)\mid(\mathrm{Y}_1,\mathrm{Y}_2)=(-1,+1)\}\\&+\frac{1}{2}\mathbb{P}\{s(X_1)=s(X_2)\mid(Y_1,Y_2)=(-1,+1)\},\end{aligned}$$

where (X_1,Y_1) and (X_2,Y_2) denote two independent copies of the pair (X,Y), see Proposition 1 in [15] for instance.

The statistical counterparts of $\mathrm{ROC}(\mathrm{s},.)$ and $\mathrm{AUC}(\mathrm{s})$ based on sampling data $\mathcal{D}_n = \{(X_i, Y_i) :\ 1 \leq i \leq n\}$ are obtained by replacing the class distributions by their empirical versions in the definitions. They are denoted by $\widehat{\mathrm{ROC}}(s,.)$ and $\widehat{\mathrm{AUC}}(s)$ in the sequel.

2.2 Tree-structured Scoring Rules

Here we focus on specific piecewise constant scoring rules, namely those defined by a left-right oriented binary tree and "combinations" of the latter (in a sense that will be specified later). For interpretability's sake, it is often desirable in many ranking applications (medical diagnosis, credit-risk, marketing, *etc.*) that the population $\mathcal{X}$ of interest be segmented in various strata. In supervised classification, one may entirely define a prediction rule by means of a partition $\mathcal{P}$ of the input space $\mathcal{X}$ and a training set $\mathcal{D}_n = \{(X_i, Y_i) :\ 1 \leq i \leq n\}$ of i.i.d. copies of the pair (X, Y) through a *majority voting scheme*, *i.e.* by assigning to any instance $x \in \mathcal{X}$ the most frequent label among the observed examples within the cell $\mathcal{C} \in \mathcal{P}$ in which x lies. However, in bipartite ranking, as the nature of the goal pursued is global, the notion of local majority vote makes no sense. In contrast, the matter is to rank the cells of the partition with respect to each other. It is assumed that ties among the ordered cells can be observed in the subsequent analysis and the usual MID-RANK convention is adopted. Refer to Appendix A for a rigorous definition of the notion of *ranking* in the case where some elements can possibly be tied. We also point out that the term *partial ranking* is often used in this context, see [18], [19] for instance.

Hence, when restricting the search of candidates to the collection of piecewise constant scoring rules, the learning problem boils down here to finding a partition $\mathcal{P} = \{\mathcal{C}_k\}_{1\leq k\leq K}$ of $\mathcal{X}$, with $1 \leq K < \infty$, together with a ranking $\preceq_{\mathcal{P}}$ of the $\mathcal{C}_k$'s (*i.e.* a preorder on $\mathcal{P}$), so that the ROC curve of the scoring function given by

$$s_{\mathcal{P},\preceq_{\mathcal{P}}}(x) = \sum_{k=1}^{K}(K - \mathcal{R}_{\preceq_{\mathcal{P}}}(\mathcal{C}_k) + 1) \cdot \mathbb{I}\{x \in \mathcal{C}_k\}$$

be as close as possible of ROC^*, where $\mathcal{R}_{\preceq_{\mathcal{P}}}(\mathcal{C}_k)$ denotes the rank of $\mathcal{C}_k$, $1 \leq k \leq K$, among all cells of $\mathcal{P}$ according to $\preceq_{\mathcal{P}}$.

In this study, particular attention is paid to specific piecewise-constant scoring functions, those related to tree-structured partitions namely. For such a partition, a ranking of the cells can be simply defined by equipping the tree with a left-right orientation. In order to describe how such a ranking tree can be built so as to maximize AUC, further concepts are required. By a *master ranking tree*, here we mean a complete, left-right oriented, rooted binary tree with depth $D \geq 1$. Consider $\mathcal{T}_D$ such a ranking tree. At depth $d = 0$, the entire input space $\mathcal{C}_{0,0} = \mathcal{X}$ forms its root. Every non terminal node (d, k), with $0 \leq d < D$ and $0 \leq k < 2^d$, is in correspondence with a subset $C_{d,k} \subset \mathcal{X}$, and has two children, corresponding each to a piece obtained by splitting $C_{d,k}$: the *left sibling* $C_{d+1,2k}$ is related to the leaf $(d + 1, 2k)$, while the *right sibling* $C_{d+1,2k+1} = C_{d,k} \setminus C_{d+1,2k}$ is related to the leaf $(d + 1, 2k + 1)$ in the tree structure. Distinguishing this way the two siblings of a parent node allows for using any subtree $\mathcal{T} \subset \mathcal{T}_D$ as a ranking rule. A ranking of the terminal cells naturally results from the left-right orientation of the tree, the top of the list being represented by the cell in the bottom left corner of the tree, and is related to the scoring function defined by: $\forall x \in \mathcal{X}$,

$$s_{\mathcal{T}}(x) = \sum_{(d,k):\ \text{terminal node of } \mathcal{T}} (2^D - 2^{D-d}k) \cdot \mathbb{I}\{x \in C_{d,k}\}.$$

The score value $s_{\mathcal{T}}(x)$ can be computed in a top-down manner, using the underlying "heap" structure. Starting from the initial value 2^D at the root node, at each subsequent inner node (d, k), $2^{D-(d+1)}$ is substracted to the current value of the score if x moves down to the right sibling $(d + 1, 2k + 1)$, whereas one leaves the score unchanged if x moves down to the left sibling. The procedure is depicted by Fig. 1.

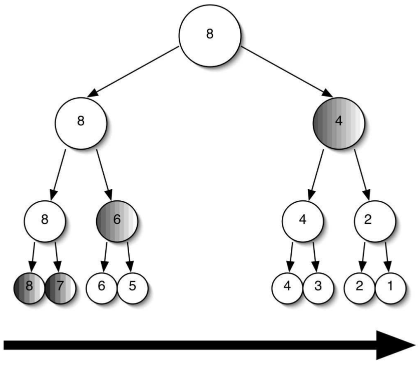


Fig. 1. Ranking tree.

2.3 The TREERANK Algorithm.

Here we briefly review the TREERANK method, on which the procedure we call RANKING FOREST crucially relies. One may refer to [2], [15] for further details as well as rigorous statistical foundations for the algorithm. As for most tree-based techniques, a greedy top-down recursive partitioning stage based on a training sample $\mathcal{D}_n = \{(X_i, Y_i) :\ 1 \leq i \leq n\}$ is followed by a pruning procedure, where children of a same parent node are recursively merged until an estimate of the AUC performance criterion is maximized. A package for R statistical software (see http://www.r-project.com) implementing TREERANK is available at http://treerank.sourceforge.net, see [20].

2.3.1 Growing Stage

The goal is here to grow a master ranking tree of large depth $D \geq 1$ with empirical AUC as large as possible. In order to describe this first stage, we introduce the following quantities. Let $\mathcal{C} \subset \mathcal{X}$, consider the empirical rate of negative (respectively, positive) instances lying in the region $\mathcal{C}$:

$$\begin{aligned}\widehat{\alpha}(\mathcal{C}) &= \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\{X_i \in \mathcal{C},\ Y_i = -1\},\\ \widehat{\beta}(\mathcal{C}) &= \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\{X_i \in \mathcal{C},\ Y_i = +1\},\end{aligned}$$

as well as $n(\mathcal{C}) = n(\widehat{\alpha}(\mathcal{C}) + \widehat{\beta}(\mathcal{C}))$ the number of data falling in $\mathcal{C}$.

One starts from the trivial partition $\mathcal{P}_0 = \{\mathcal{X}\}$ at root node $(0,0)$ (we set $C_{0,0} = \mathcal{X}$) and proceeds recursively as follows. A tree-structured scoring rule $s(x)$ described by an oriented tree, with outer leaves forming a partition $\mathcal{P}$ of the input space, is refined by splitting a cell $\mathcal{C} \in \mathcal{P}$ into two subcells: $\mathcal{C}'$ denoting the left child and $\mathcal{C}'' = \mathcal{C} \setminus \mathcal{C}'$ the right one. Let $s'(x)$ be the scoring function thus obtained. From the perspective of AUC maximization, one is lead to seek for a subregion $\mathcal{C}'$ maximizing the gain $\Delta_{\widehat{\mathrm{AUC}}}(\mathcal{C}, \mathcal{C}')$ in terms of empirical AUC induced by the split, which may be written as:

$$\widehat{\mathrm{AUC}}(s') - \widehat{\mathrm{AUC}}(s) = \frac{1}{2}\{\widehat{\alpha}(\mathcal{C})\widehat{\beta}(\mathcal{C}') - \widehat{\beta}(\mathcal{C})\widehat{\alpha}(\mathcal{C}')\}.$$

Therefore, taking the rate of positive instances within the cell $\mathcal{C}$, $\widehat{p}(\mathcal{C}) = \widehat{\alpha}(\mathcal{C}) \cdot n/n(\mathcal{C})$ namely, as cost for the type I error (*i.e.* predicting label $+1$ when $Y = -1$) and $1 - \widehat{p}(\mathcal{C})$ as cost for the type II error, the quantity $1 - \Delta_{\widehat{\mathrm{AUC}}}(\mathcal{C}, \mathcal{C}')$ may be viewed as the *cost-sensitive empirical misclassification error* of the classifier $C(X) = 2 \cdot \mathbb{I}\{X \in \mathcal{C}'\} - 1$ on $\mathcal{C}$ up to a multiplicative factor, $4\widehat{p}(\mathcal{C})(1 - \widehat{p}(\mathcal{C}))$ precisely. Hence, once the local cost $\widehat{p}(\mathcal{C})$ is computed, any binary classification method can be straightforwardly adapted in order to perform the splitting step. Here, splits are obtained using empirical-cost sensitive versions of the standard CART algorithm with axis-parallel splits, this one-step procedure for AUC maximization being called LEAFRANK in [15]. As depicted by Fig. 2, the growing stage appears as a recursive implementation of a cost-sensitive CART procedure with a cost updated at each node of the ranking tree, equal to the local rate of positive instances within the node to split, see Section 3 of [15].

2.3.2 Pruning Stage

The way the master ranking tree $\mathcal{T}_D$ obtained at the end of the growing stage is pruned is entirely similar to the one described in [3], the sole difference lying in the fact that here, for any $\lambda > 0$, one seeks a subtree $\mathcal{T} \subset \mathcal{T}_D$ that maximizes the penalized empirical AUC

$$\widehat{\mathrm{AUC}}(s_{\mathcal{T}}) - \lambda \cdot |\mathcal{T}|,$$

where $|\mathcal{T}|$ denotes the number of terminal leaves of $\mathcal{T}$, the constant being next picked using N-fold cross validation.

The fact that alternative complexity-based penalization procedures, inspired from recent nonparametric model selection methods and leading to the concept of *structural* AUC *maximization*, can be successfully used for pruning ranking trees has also been pointed up in subsection 4.2 of [15]. However, the resampling-based technique previously mentioned is preferred to such pruning schemes in practice, insofar as it does not require, in contrast, to specify any tuning constant. Following in the footsteps of [21] in the classification setup, estimation of the ideal penalty through bootstrap methods could arise as the answer to this issue. This question is beyond the scope of the present paper but will soon be tackled.

2.3.3 Some Practical Considerations

Like other types of decision trees, ranking trees (based on perpendicular splits) have a number of crucial advantages. Concerning interpretability first, it should be noticed that they produce ranking rules that can be easily visualized through the binary tree graphic, see Fig. 2, the rank/score of an instance $x \in \mathcal{X}$ being obtained through checking of a nested combination of simple rules of the form "$X^{(k)} \geq t$" or "$X^{(k)} < t$". In addition, ranking trees can adapt straightforwardly to situations where some data are missing and/or some predictor variables are categorical and some monitoring tools helping to evaluate the relative importance of each predictor variable $X^{(k)}$ or to depict the partial dependence of the prediction rule on a subset of input variables are readily available. These facets are described in section 5 of [15]. From a computational perspective now, we also underline that the tree structure makes the computation of consensus rankings much more tractable, we refer to Appendix B for further details.

3 RANKING FOREST

In this section, we introduce most of the ingredients required for subsequent description of the RANKING FOREST algorithm. We investigate how to adapt the ideas grounding RANDOM FOREST, bagging and random selection of features, in order to grow and combine an ensemble of ranking trees so as to improve upon the accuracy of single ranking trees.

3.1 Feature Randomization in TREERANK

Whereas the concept of **b**ootstrap **agg**regat**ing** technique had already been introduced in [22], the major novelty in the RANDOM FOREST method consists in randomizing the features used for recursively splitting the nodes of the classification/regression trees involved in the committee-based prediction procedure. As recalled in subsection 2.3, the left and right siblings of the ranking

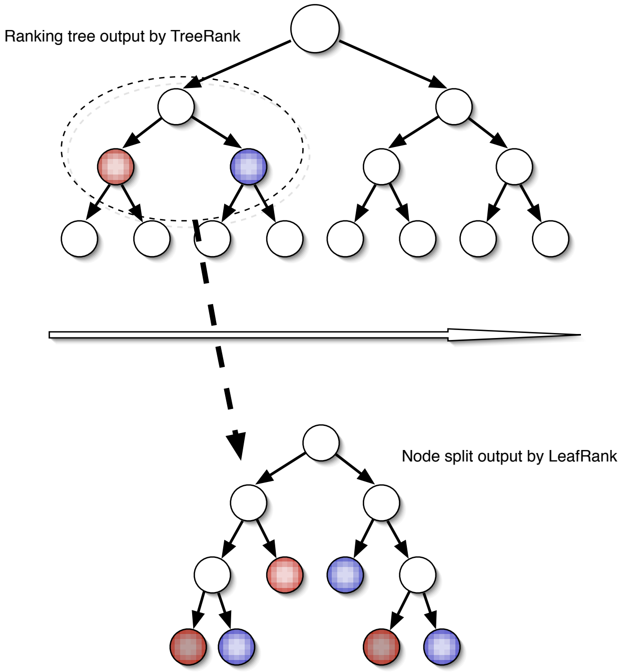

Ranking tree output by TreeRank
Node split output by LeafRank

Fig. 2. THE TREERANK ALGORITHM AS A RECURSIVE IMPLEMENTATION OF COST-SENSITIVE CART.

tree nodes are themselves obtained through classification trees. We thus propose two possible feature randomization schemes for TREERANK.

- $\mathcal{FR}_1$: *At each node (d,k) of the master ranking tree $\mathcal{T}_D$, draw at random a set of $q_0 \leq q$ indexes $\{i_1, \ldots, i_{q_0}\} \subset \{1, \ldots, q\}$. Implement the* LEAFRANK *splitting procedure based on the descriptor $(X_{i_1}, \ldots, X_{i_{q_0}})$ to split the cell $C_{d,k}$.*
- $\mathcal{FR}_2$: *For each node (d,k) of the master ranking tree $\mathcal{T}_D$, at each node of the cost-sensitive classification tree describing the split of the cell $\mathcal{C}_{d,k}$ into two children, draw at random a set of $q_1 \leq q$ indexes $\{j_1, \ldots, j_{q_1}\} \subset \{1, \ldots, q\}$ and perform an axis-parallel cut using the descriptor $(X_{j_1}, \ldots, X_{j_{q_1}})$.*

We underline that, of course, these randomization methods do not exclude each other. At each node (d,k) of the ranking tree, one may first draw at random a collection $\mathcal{F}_{d,k}$ of q_0 features and then, when growing the cost-sensitive classification tree describing $C_{d,k}$'s split, divide each node based on a sub-collection of $q_1 \leq q_0$ features drawn at random among $\mathcal{F}_{d,k}$.

3.2 Aggregation of Ranking Trees

In recent years, the issue of summarizing/aggregating various rankings has been the subject of a good deal of attention in the machine-learning literature, mainly motivated by practical problems in the Web context: design of meta-search engines, collaborative filtering, spam-fighting, *etc.* Refer to [17], [23], [24], [25] for instance. Such problems have lead to a variety of results, ranging from the generalization of the mathematical concepts introduced in social choice theory (see [16] and the references therein) for defining relevant notions of *consensus* between rankings [19], to the development of efficient procedures for computing such "consensus rankings" [26], [27], [28], through the study of probabilistic models over sets of rankings [29], [30]. Here we revisit the rank aggregation issue with a view to extend the bagging approach to ranking trees. In order to formulate our answer to this problem, further definitions and notations are required.

Given two partitions $\mathcal{P}$ and $\mathcal{P}'$ of the feature space $\mathcal{X}$, we will say that $\mathcal{P}'$ is a *subpartition* of $\mathcal{P}$, and then denote $\mathcal{P}' \subset \mathcal{P}$, when any nonempty cell $\mathcal{C} \in \mathcal{P}$ can be obtained as union of cells of $\mathcal{P}'$.

Suppose now that we dispose of an ensemble of $B \geq 1$ ranking trees $\mathcal{T}_1, \ldots, \mathcal{T}_B$. Each ranking tree $\mathcal{T}_b$, $1 \leq b \leq B$, is associated to a partition $\mathcal{P}_b$ and a ranking $\preceq_b$ of its cells. We consider the partition $\mathcal{P}_B^*$ made of nonempty subsets $\mathcal{C} \subset \mathcal{X}$ satisfying the two constraints:

(i) there exists $(\mathcal{C}_1, \ldots, \mathcal{C}_B) \in \mathcal{P}_1 \times \cdots \times \mathcal{P}_B$ such that:

$$\mathcal{C} = \bigcap_{b=1}^{B} \mathcal{C}_b,$$

(ii) for all nonempty subset $\mathcal{C}'$ belonging to some partition $\mathcal{P}_b$, $b \in \{1, \ldots, B\}$: if $\mathcal{C}' \subset \mathcal{C}$, then $\mathcal{C}' = \mathcal{C}$.

One may easily see that $\mathcal{P}_B^*$ is a subpartition of all the $\mathcal{P}_b$'s, and the largest one in the sense that any partition $\mathcal{P}$ such that $\mathcal{P} \subset \mathcal{P}_b$ for all $b \in \{1, \ldots, B\}$ is a subpartition of $\mathcal{P}_B^*$. We denote $\mathcal{P}_B^* = \bigcap_{b \leq B} \mathcal{P}_b$. Incidentally, it should be noticed that, from a computational perspective, the underlying tree structures considerably help getting $\mathcal{P}_B^*$'s cells explicitely, refer to Appendix B for further details.

Each ranking tree $\mathcal{T}_b$ naturally induces a ranking (or a *preorder*) $\preceq_b^*$ on the partition $\mathcal{P}_B^*$: precisely, for all $(\mathcal{C}, \mathcal{C}') \in \mathcal{P}_B^{*2}$, one write by definition $\mathcal{C} \preceq_b^* \mathcal{C}'$ (respectively, $\mathcal{C} \prec_b^* \mathcal{C}'$) iff $\mathcal{C}_b \preceq_b^* \mathcal{C}'_b$ (respectively, $\mathcal{C}_b \prec_b^* \mathcal{C}'_b$) where $(\mathcal{C}_b, \mathcal{C}'_b) \in \mathcal{P}_b^2$ are such that $\mathcal{C} \times \mathcal{C}' \subset \mathcal{C}_b \times \mathcal{C}_b$. We denote by $\mathcal{R}_b^*(\mathcal{C})$ the rank of the cell $\mathcal{C} \in \mathcal{P}_B^*$ as defined by $\preceq_b^*$, for $b = 1, \ldots, B$.

Now, based on this family of B rankings on $\mathcal{P}_B^*$, called a *profile* in voting theory, we would like to define a "central ranking" or a *consensus*. Whereas the mean or the median naturally provides such a summary when considering scalar data, various meanings can be given to this notion for rankings. It is precisely the purpose of this subsection to review possible ways of aggregating a finite collection of rankings on a same set of finite cardinality.

3.2.1 Agreement between rankings

The most widely used approach to the *rank aggregation* issue relies on the concept of *measure of agreement* between rankings or *(pseudo-) metrics* equivalently, depending on whether one chooses to measure similarity or dissimilarity. Since the seminal contribution of [31], numerous ways of measuring agreement have been proposed in the literature. Here we review three popular choices, originally introduced in the context of *nonparametric statistical testing*; see [24] for instance.

Let $\preceq$ and $\preceq'$ be two rankings on a finite set $\mathcal{Z} = \{z_1, \ \ldots, \ z_K\}$.

Kendall τ. Consider the quantity $d_\tau(\preceq, \preceq')$, obtained by summing up all the terms

$$\begin{aligned} U_{i,j}(\preceq, \preceq') = \mathbb{I}\{(\mathcal{R}_{\preceq}(z_i) - \mathcal{R}_{\preceq}(z_j))(\mathcal{R}_{\preceq'}(z_i) - \mathcal{R}_{\preceq'}(z_j)) < 0\} \\ + \frac{1}{2}\mathbb{I}\{\mathcal{R}_{\preceq}(z_i) = s_{\preceq}(z_j), \ \mathcal{R}_{\preceq'}(z_i) \neq \mathcal{R}_{\preceq'}(z_j)\} \\ + \frac{1}{2}\mathbb{I}\{\mathcal{R}_{\preceq'}(z_i) = \mathcal{R}_{\preceq'}(z_j), \ \mathcal{R}_{\preceq}(z_i) \neq \mathcal{R}_{\preceq}(z_j)\} \end{aligned}$$

over all pairs (z_i, z_j) such that $1 \leq i < j \leq K$. It counts, among the $K(K-1)$ pairs of $\mathcal{Z}$'s elements, how many are "discording", assigning the weight $1/2$ when two elements are tied in one ranking but not in the other. The Kendall τ is obtained by renormalizing this distance:

$$\tau(\preceq, \preceq') = 1 - \frac{4}{K(K-1)} d_\tau(\preceq, \preceq'). \tag{1}$$

Large values of $\tau(\preceq, \preceq')$ indicate agreement (or similarity) between $\preceq$ and $\preceq'$: it ranges from -1 (full disagreement) to 1 (full agreement). It is worth noticing that it can be computed in $O((K \log K)/\log\log K)$ time, refer to [32].

Spearman footrule. Another natural distance between rankings is defined by considering the l_1-metric between the corresponding rank vectors:

$$d_1(\preceq, \ \preceq') = \sum_{i=1}^{K} |\mathcal{R}_{\preceq}(z_i) - \mathcal{R}_{\preceq'}(z_i)|.$$

The affine transformation given by

$$F(\preceq, \preceq') = 1 - \frac{3}{K^2 - 1} d_1(\preceq, \ \preceq'). \tag{2}$$

is known as the Spearman footrule measure of agreement and takes its values in $[-1, +1]$.

Spearman rank-order correlation. Considering instead the l_2-metric

$$d_2(\preceq, \ \preceq') = \sum_{i=1}^{K} \left(\mathcal{R}_{\preceq}(z_i) - \mathcal{R}_{\preceq'}(z_i)\right)^2$$

leads to the Spearman ρ coefficient:

$$\rho(\preceq_1, \preceq_2) = 1 - \frac{6}{K(K^2 - 1)} d_2(\preceq, \ \preceq'). \tag{3}$$

Remark 1 (EQUIVALENCE.) *It should be noticed that these three measures of agreement are equivalent in the sense that:*

$$\begin{aligned} c_1\,(1 - \rho(.,.)) \leq \quad & (1 - F(.,.))^2 && \leq c_2\,(1 - \rho(.,.))\,, \\ c_3\,(1 - \tau(.,.)) \leq \quad & 1 - F(.,.) && \leq c_4\,(1 - \tau(.,.))\,, \end{aligned}$$

with $c_2 = K^2/(2(K^2-1)) = Kc_1$ *and* $c_4 = 3K/(2(K+1)) = 2c_3$*; see Theorem 13 in [19].*

We point out that many fashions of measuring agreement or distance between rankings have been considered in the literature, see [33]. Well-known alternatives to the measures recalled above are the Cayley/Kemeny distance [31] and variants for top k-lists [19], in order to focus on the "best instances" [34]. Many other distances between rankings could naturally be deduced through suitable extensions of *word metrics* on the symmetric groups on finite sets, see [35] or [36].

3.2.2 Probabilistic measures of scoring agreement

Now the concept of (pseudo-) metric for rankings on a finite set has been recalled, it is easy to relate it to a notion of dissimilarity between preorders on a general space $\mathcal{X}$, when the latter are induced by piecewise constant scoring functions. Consider two scoring functions s_1 and s_2, defining preorders $\preccurlyeq_{s_1}$ and $\preccurlyeq_{s_2}$ on $\mathcal{X}$, both constant on each cell of a partition $\mathcal{P} = \{\mathcal{C}_k\}_{1 \leq k \leq K}$. They also naturally define rankings $\preceq_{s_1}$ and $\preceq_{s_2}$ on the finite set $\mathcal{P}$: $\forall (k, l) \in \{1, \ \ldots, \ K\}$, $\mathcal{C}_k \preceq_{s_i} \mathcal{C}_l$ iff $\exists (x, x') \in \mathcal{C}_k \times \mathcal{C}_l$ such that $s_i(x) \leq s_i(x')$ for $i = 1, \ 2$. Given a pseudo-metric d on the set of rankings of $\mathcal{P}$'s elements, one may thus quantify the disagreement between the preorders they induce on $\mathcal{X}$ by:

$$\tilde{d}(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}) \overset{def}{=} d(\preceq_{s_1}, \preceq_{s_2}).$$

As we shall see now, in some specific situations, this may lead to measures of closeness between $\preccurlyeq_{s_1}$ and $\preccurlyeq_{s_2}$, that can be alternately defined with absolutely no reference to the partition $\mathcal{P}$, by contrast with the expression $d(\preceq_{s_1}, \preceq_{s_2})$. As mentioned above, most widely used measures of agreement between rankings arise from the field of nonparametric statistics and the quantities defined above have well-known probabilistic counterparts. In this regard, recall that the theoretical Kendall τ related to two random variables (Z_1, Z_2) defined on the same probability space is $\widetilde{\tau}(Z_1, Z_2) = 1 - 2d_{\tilde{\tau}}(Z_1, Z_2)$, with:

$$\begin{aligned} d_{\tilde{\tau}}(Z_1, Z_2) = \mathbb{P}\{(Z_1 - Z_1') \cdot (Z_2 - Z_2') < 0\} \\ + \frac{1}{2}\mathbb{P}\{Z_1 = Z_1', \ Z_2 \neq Z_2'\} \\ + \frac{1}{2}\mathbb{P}\{Z_1 \neq Z_1', \ Z_2 = Z_2'\}. \end{aligned}$$

where (Z_1', Z_2') is an independent copy of the pair (Z_1, Z_2). Notice, incidentally, that the Kendall τ for the pair $(s(X), Y)$ is related to $\mathrm{AUC(s)}$ through:

$$\begin{aligned} \frac{1}{2}\left(1 - \tilde{\tau}(s(X), Y)\right) = 2p(1-p)\left(1 - \mathrm{AUC(s)}\right) \\ + \frac{1}{2}\mathbb{P}\{s(X) \neq s(X'), \ Y = Y'\}. \end{aligned}$$

Hence, a natural way of measuring the similarity between the preorders on $\mathcal{X}$ induced by the scoring functions s_1 and s_2 is to consider the probabilistic quantity $\tau_X(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}) = \tilde{\tau}(s_1(X), s_2(X))$, or, equivalently, the probability that s_1 and s_2 rank two independent copies X and X' in the same order. We also set $d_{\tau_X}(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}) =$

$d_{\tilde{\tau}}(s_1(X), s_2(X))$. The next result reveals the connection between this quantity and a specific notion of agreement between the rankings $\preceq_{s_1}$ and $\preceq_{s_2}$ they induce on $\mathcal{P}$. The proof is straightforward and thus omitted.

Lemma 3.1 *Let $(s_1, s_2) \in \mathcal{S}^2$, we have:*

$$d_{\tau_X}(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}) = 2 \sum_{1 \leq k < l \leq K} \mu(\mathcal{C}_k)\mu(\mathcal{C}_l) \cdot U_{k,l}(\preceq_{s_1}, \preceq_{s_2}). \tag{4}$$

Hence, $\tau_X(\preccurlyeq_{s_1}, \preccurlyeq_{s_2})$ may be viewed as a "weighted version" of the rate of concording pairs measured by $\tau(\preceq_{s_1}, \preceq_{s_2})$. Notice that, when all cells have the same weight with respect to μ, *i.e.* when $\forall (k, l) \in \{1, \ldots, K\}^2$, $\mu(\mathcal{C}_k) = \mu(\mathcal{C}_l) = 1/K$, we have $d_{\tau_X}(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}) = 2d_\tau(\preceq_{s_1}, \preceq_{s_2})/K^2$. In order to avoid confusion, we shall use the term "probabilistic Kendall τ" to refer to the quantity $\tau_X(\preccurlyeq_{s_1}, \preccurlyeq_{s_2})$.

The following proposition shows that the AUC deviation between two scoring functions is controlled by the related probabilistic Kendall tau in a very simple fashion. It is essentially for this reason that the Kendall τ criterion plays a large part in the theoretical analysis carried out in Section 4.

Proposition 3.2 (AUC AND KENDALL τ) *Let $p = \mathbb{P}\{Y = +1\} \in (0, 1)$. For any scoring functions s_1 and s_2 on $\mathcal{X}$, we have:*

$$|\mathrm{AUC(s_1) - AUC(s_2)}| \leq \frac{1 - \tau_\mathrm{X}(\preccurlyeq_{\mathrm{s_1}}, \preccurlyeq_{\mathrm{s_2}})}{\mathrm{4p(1-p)}}.$$

We point out that it is generally vain to look for a reverse control: indeed, scoring functions yielding different rankings may have exactly the same AUC. However, the following result guarantees that a scoring function with a nearly optimal AUC is close to optimal scoring functions in Kendall sense, under the additional assumption that the noise condition introduced in [37] is fulfilled.

Proposition 3.3 (AUC AND KENDALL τ (BIS)) *Assume that the r.v. $\eta(X)$ is continuous and there exists $\epsilon \in (0, 1/2)$ such that $\epsilon \leq \eta(X) \leq 1 - \epsilon$ with probability one. Suppose also that there exist $c < \infty$ and $a \in (0, 1)$ such that*

$$\forall x \in \mathcal{X}, \;\; \mathbb{E}\left[|\eta(X) - \eta(x)|^{-a}\right] \leq c. \tag{5}$$

Then, we have for all $(s, s^*) \in \mathcal{S} \times \mathcal{S}^*$,

$$1 - \tau_X(\preccurlyeq_{s^*}, \preccurlyeq_s) \leq C \cdot (\mathrm{AUC}^* - \mathrm{AUC(s)})^{a/(1+a)},$$

with $C = 2 \cdot \max\{c^{1/(1+a)}, \; p(1-p)/\epsilon^2\}$.

Remark 2 (ON THE NOISE CONDITION.) *Recall that condition (5) is rather weak. Indeed, it is fulfilled for any $a \in (0, 1)$ as soon $\eta(X)$'s density is bounded, see Corollary 8 in [37]. Notice in addition that the condition related on $\eta(X)$'s range means that the likelihood ratio of the class distributions $\phi(X) = dG/dH(X)$ is bounded and bounded away from zero, recall indeed that $\phi(X) = ((1-p)/p) \cdot (\eta(X)/(1 - \eta(X)))$, or, equivalently, that the slope of the tangent of the optimal* ROC *curve at the origin is not infinite and that at the opposite vertex $(1, 1)$ of the unit square is non zero ; see [38].*

A statistical version of $\tau_X(\preccurlyeq_{s_1}, \preccurlyeq_{s_2})$ is obtained by replacing the $\mu(\mathcal{C}_k)$'s by their empirical counterparts in Eq. (4). It may be classically expressed as

$$\widehat{\tau}_X(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}) = 1 - 2\widehat{d}_{\tau_X}(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}), \tag{6}$$

where $\widehat{d}_{\tau_X}(\preccurlyeq_{s_1}, \preccurlyeq_{s_2}) = 2/(n(n-1)) \sum_{i<j} K(X_i, X_j)$ is a U-statistic of degree 2 with symmetric kernel given by:

$$\begin{aligned} K(x, x') = \; & \mathbb{I}\{(s_1(x) - s_1(x')) \cdot (s_2(x) - s_2(x')) < 0\} \\ & + \frac{1}{2}\mathbb{I}\{s_1(x) = s_1(x'), \; s_2(x) \neq s_2(x')\} \\ & \qquad + \frac{1}{2}\mathbb{I}\{s_1(x) \neq s_1(x'), \; s_2(x) = s_2(x')\}. \end{aligned}$$

Alternately, one could measure the dissimilarity between $\preccurlyeq_{s_1}$ and $\preccurlyeq_{s_2}$ by considering the theoretical Spearman correlation coefficient, $\tilde{\rho}(s_1(X), s_2(X))$, that is the linear correlation coefficient between the r.v.'s $F_{s_1}(s_1(X))$ and $F_{s_2}(s_2(X))$, where F_{s_i} denotes the cdf of $s_i(X)$, $i \in \{1, \; 2\}$. Based on a sample $(X_1, \; \ldots, \; X_n)$ of i.i.d. copies of the r.v. X, the empirical counterpart is the empirical linear correlation between the rank vectors of $(s_1(X_1), \; \ldots, \; s_1(X_n))$ and $(s_2(X_1), \; \ldots, \; s_2(X_n))$ respectively. Notice that, when s_1 and s_2 are constants on the cells of a partition $\mathcal{P} = \{\mathcal{C}_k\}_{1 \leq k \leq K}$ such that $\mu(C_k) = 1/K$ for $1 \leq k \leq K$, we have $\tilde{\rho}(s_1(X), s_2(X)) = \rho(\preceq_{s_1}, \preceq_{s_2})$.

3.2.3 Median Rankings

The method for aggregating rankings we consider here relies on the so-termed *median procedure*, which belongs to the family of *metric aggregation procedures*, see [16] for further details. Let $d(.,.)$ be some metric or dissimilarity measure on the set of rankings on a finite set $\mathcal{Z}$. By definition, a *median ranking* among a profile $\Pi = \{\preceq_k : 1 \leq k \leq K\}$ with respect to d is any ranking $\preceq_{med}$ on $\mathcal{Z}$ that minimizes the sum $d_\Pi(\preceq) \overset{def}{=} \sum_{k=1}^K d(\preceq, \preceq_k)$ over the set $\mathbf{R}(\mathcal{Z})$ of all rankings $\preceq$ on $\mathcal{Z}$:

$$d_\Pi(\preceq_{med}) = \min_{\preceq: \text{ ranking on } \mathcal{Z}} d_\Pi(\preceq). \tag{7}$$

Notice that, when $\mathcal{Z}$ is of cardinality $N < \infty$, there are

$$\#\mathbf{R}(\mathcal{Z}) = \sum_{k=1}^N (-1)^k \sum_{m=1}^k (-1)^m \binom{k}{m} m^N$$

possible rankings on $\mathcal{Z}$[1] and in most cases, the computation of (metric) median rankings leads to solve NP-hard combinatorial optimization problems, see [39], [40], [41] and the references therein. From a practical perspective, acceptably good solutions can be computed in a reasonable amount of time by means of (probabilistic) metaheuristics such as simulated annealing, genetic

[1] Let $1 \leq k \leq N$. We recall that the number of surjective mappings from $\{1, \ldots, N\}$ to $\{1, \ldots, k\}$ is $(-1)^k \sum_{m=0}^k (-1)^m \binom{k}{m} m^N$.

algorithms or tabu search; see [42]. Refer to [43], [44] for instance.

When it comes to preorders on a set $\mathcal{X}$ of infinite cardinality, defining a notion of aggregation becomes harder. Given a pseudo-metric such as $d_{\tilde{\tau}}(.,.)$ for instance and $K \geq 1$ scoring functions $s_1, \ \ldots, \ s_K$ on $\mathcal{X}$, the existence of $\bar{s}$ in $\mathcal{S}$ such that $\sum_{k=1}^{K} d_{\tilde{\tau}}(\bar{s}, s_k) = \min_{s \in \mathcal{S}} \sum_{k=1}^{K} d_{\tilde{\tau}}(s, s_k)$ is by no means guaranteed in general. However, when considering scoring functions that are constant on each cell of a given finite partition $\mathcal{P}$ of $\mathcal{X}$, the corresponding preorders are in one-to-one correspondence with rankings on $\mathcal{P}$ and the minimum distance is thus effectively attained, by a *median scoring function* that is also constant on $\mathcal{P}$'s cells.

Remark 3 (THE ORDINAL APPROACH.) *We point out that metric aggregation procedures are not the sole way to summarize a profile of rankings. The so-termed "ordinal approach" provides a variety of alternative techniques for combining rankings (or, more generally, preferences), returning to the famous "Arrow's voting paradox" and consisting of a series of duels (i.e. pairwise comparisons) as in Condorcet's methods or successive tournaments as in the well-known proportional voting Hare system, see [45]. Such approaches have recently been the subject of a good deal of attention in the context of preference learning ("Ranking by Pairwise Comparison" methods); see [46] for instance.*

Remark 4 (ON UNIQUENESS.) *It is worth noticing that a median ranking is far from being unique in general. One may immediately check for instance that any ranking among the profile made of all rankings on $\mathcal{Z} = \{1, \ 2\}$ is a median in Kendall sense, i.e. for the metric d_τ.*

3.2.4 Aggregation: Ranks vs. Rankings

Let $s_1, \ldots, s_K$ be $K \geq 1$ base piecewise constant scoring functions and $\mathbf{X}^{(n)} = \{X_1, \ldots, X_n\}$ a collection of $n \geq 1$ i.i.d. copies of the input variable X. When it comes to rank the X_i's "consensually", two strategies can be considered:

1) compute a "median ranking rule" based on the K rankings of the largest subpartition's cells and use it for ranking the new data as previously described,

2) alternatively, compute, for each scoring rule s_k, the related rank vector of $(X_1, \ldots, X_n)$ or, in other words, the ranking induced by s_k on the set $\mathbf{X}^{(n)}$, and then a "median rank vector", *i.e.* a median ranking on the set $\mathbf{X}^{(n)}$ (data lying in a same cell of the largest subpartition being tied).

Although they are not equivalent, these two methods generally produce similar results, especially when n is large. Indeed, considering probabilistic Kendall τ medians for instance, it is sufficient to notice that the Kendall τ distance d_τ between rankings on $\mathbf{X}^{(n)}$ induced by two piecewise constant scoring functions s and s' can be viewed as an empirical estimate of $d_{\tau_X}(\preccurlyeq_s, \preccurlyeq_{s'})$ based on the dataset $\mathbf{X}^{(n)}$. Therefore, the median computation approach 1) relies on analogous quantities except they are not based on the data to be ranked but on the training dataset. However, when both the size of the training sample and that of the dataset $\mathbf{X}^{(n)}$ are large, the two approaches lead to optimize close quantities, except that, in case 1), optimization is performed over rankings of the largest subpartition's cells, while in case 2), rankings on $\mathbf{X}^{(n)}$, such that data belonging to a same cell of the largest subpartition are tied, are considered.

3.3 The Algorithm

Now that the rationale behind the RANKING FOREST procedure has been given, we recapitulate its successive steps in detail. Based on a training sample $\mathcal{D} = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, the algorithm is performed in three stages, as follows.

RANKING FOREST

1) **Parameters.** B number of bootstrap replicates, n^* bootstrap sample size, TREERANK tuning parameters (depth D and presence/absence of pruning) $\mathcal{FR}$ feature randomization strategy, d pseudo-metric.

2) **Bootstrap profile makeup.**
   a) (RESAMPLING STEP.) Build B independent bootstrap samples $\mathcal{D}_1^*, \ \ldots, \ \mathcal{D}_B^*$, by drawing with replacement $n^* \times B$ pairs among the original training sample $\mathcal{D}$.
   b) (RANDOMIZED TREERANK.) For $b = 1, \ \ldots, \ B$, run TREERANK combined with the feature randomization method $\mathcal{FR}$ based on the sample $\mathcal{D}_b^*$, yielding the ranking tree $\mathcal{T}_b^*$, related to the partition $\mathcal{P}_b^*$ of the space $\mathcal{X}$.

3) **Aggregation.** Compute the largest subpartition partition $\mathcal{P}^* = \bigcap_{b=1}^{B} \mathcal{P}_b^*$. Let $\preceq_b^*$ be the ranking of $\mathcal{P}^*$'s cells induced by $\mathcal{T}_b^*$, $b = 1, \ \ldots, \ B$. Compute a median ranking $\preceq^*$ related to the bootstrap profile $\Pi^* = \{\preceq_b^* \colon 1 \leq b \leq B\}$ with respect to the metric d on $\mathbf{R}(\mathcal{P}^*)$:

$$\preceq^* = \mathop{\arg\min}_{\preceq \in \mathbf{R}(\mathcal{P}^*)} d_{\Pi^*}(\preceq),$$

as well as the scoring function $s_{\preceq^*, \mathcal{P}^*}$(x).

Before setting theoretical grounds for use of the RANKING FOREST method, a few remarks are in order.

Remark 5 (ON TUNING PARAMETERS.) *As mentioned in 3.2.3, aggregating ranking rules is computationally expensive. The empirical results displayed in Section 5 suggest to aggregate several dozens of randomized ranking trees of moderate or even small depth built from bootstrap samples of size $n^* \leq n$.*

Remark 6 ("PLUG-IN" BAGGING.) *As pointed out in [2] (see Remark 6 therein), given an ordered partition $(\mathcal{P},\mathcal{R}_\mathcal{P})$ of the feature space $\mathcal{X}$, a "plug-in" estimate of the (optimal scoring) function $S=H_\eta\circ\eta$ can be automatically deduced from any ordered partition (or piecewise constant scoring function equivalently) and the data $\mathcal{D}$, where H_η denotes the conditional cdf of $\eta(X)$ given $Y=-1$. This scoring function is somehow canonical in the sense that, given $Y=-1$, $H(X)$ is distributed as a uniform r.v. on $[0,1]$. Considering a partition $\mathcal{P}=\{\mathcal{C}_k\}_{1\le k\le K}$ equipped with a ranking $\mathcal{R}_\mathcal{P}$, the plug-in estimate is given by*

$$\widehat{S}_{\mathcal{P},\mathcal{R}_\mathcal{P}}(x)=\sum_{k=1}^{K}\widehat{\alpha}(R_k)\cdot\mathbb{I}\{x\in\mathcal{C}_k\},\quad x\in\mathcal{X},\tag{8}$$

where $R_k=\bigcup_{l:\ \mathcal{R}(k)\le\mathcal{R}(l)}C_l$. Notice that, as a scoring function, $\widehat{S}_{\mathcal{P},\mathcal{R}_\mathcal{P}}$ and yields the same ranking as $s_{\mathcal{P},\mathcal{R}_\mathcal{P}}$, provided that $\widehat{\alpha}(\mathcal{C}_k)>0$ for all $k=1,\ \dots,\ K$. Adapting the idea proposed in subsection 6.1 of [22] in the classification context, an alternative to the rank aggregation approach proposed here naturally consists in computing the average of the piecewise-constant scoring functions $\widetilde{S}^*_{\mathcal{T}^*_b}$ thus defined by the bootstrap ranking trees and consider the rankings induced by the latter. This method we call "plug-in bagging" is however less effective in many situations, due to the inaccuracy/variability of the probability estimates involved.

Ranking stability. Let $\Theta=\mathcal{X}\times\{-1,+1\}$. From the view developed in this paper, a ranking algorithm is a function $\mathbf{S}$ that maps any data sample $\mathcal{D}\in\Theta^n$, $n\ge 1$, to a scoring function $\mathbf{S}_\mathcal{D}$. In the ranking context, we will say that a learning algorithm is "stable" when the preorder on $\mathcal{X}$ it outputs is not much affected by small changes in the training set. We propose a natural way of measuring ranking stability, through the computation of

$$\mathbf{Stab}_n(\mathbf{S})=\mathbb{E}\left[d_{\tau_X}\left(\preccurlyeq_{\mathbf{S}_\mathcal{D}},\preccurlyeq_{\mathbf{S}_{\mathcal{D}'}}\right)\right],\tag{9}$$

where the expectation is taken over two independent training samples $\mathcal{D}$ and $\mathcal{D}'$, both made of n i.i.d. copies of the pair (X,Y). We highlight the fact that the bootstrap stage of RANKING FOREST can be used for assessing the stability of the base ranking algorithm: indeed, the quantity

$$\widehat{\mathbf{Stab}}_n(\mathbf{S})=\frac{2}{B(B-1)}\sum_{1\le b<b'\le B}\widehat{d}_{\tau_X}\left(\preccurlyeq_{\mathbf{S}_{\mathcal{D}^*_b}},\preccurlyeq_{\mathbf{S}_{\mathcal{D}^*_{b'}}}\right),$$

can be possibly interpreted as a bootstrap estimate of (9).

We finally underline that the outputs of the RANKING FOREST can also be used for monitoring ranking performance, in an analogous fashion to RANDOM FOREST in the classification/regression context, see subsection 3.1 in [1] and the references therein. An *out-of-bag* estimate of the AUC criterion can be obtained by considering, for all pairs (X,Y) and (X',Y') in the original training sample, those ranking trees that are built from bootstrap samples containing neither of them, avoiding this way the use of a test dataset.

4 SOME THEORETICAL BACKGROUND

The purpose of this section is to set preliminary statistical grounds for the aggregation procedure in the ranking context. Precisely, following in the footsteps of [47], from which some of the notations are borrowed, we study the AUC consistency of scoring rules that are obtained as a median over a profile of consistent *randomized scoring functions* for the (probabilistic) Kendall τ distance. Here, a randomized scoring function is of the form $\mathbf{S}_{\mathcal{D}_n}(.,Z)$, where $\mathcal{D}_n=\{(X_1,Y_1),\ \dots,\ (X_n,Y_n)\}$ denotes the training sample and Z a random variable taking its values in some measurable space $\mathcal{Z}$ that describes the randomization mechanism.

Suppose that a randomized scoring function $\mathbf{S}_{\mathcal{D}_n}(.,Z)$ is given and consider its generalization AUC, $\mathrm{AUC}(\mathbf{S}_{\mathcal{D}_\mathrm{n}}(.,\mathrm{Z}))$ namely, which is given by:

$$\mathbb{P}\{\mathbf{S}_{\mathcal{D}_n}(X,Z)<\mathbf{S}_{\mathcal{D}_n}(X',Z)\mid(Y,Y')=(-1,+1)\}$$
$$+\frac{1}{2}\mathbb{P}\{\mathbf{S}_{\mathcal{D}_n}(X,Z)=\mathbf{S}_{\mathcal{D}_n}(X',Z)\mid(Y,Y')=(-1,+1)\},$$

where the conditional probabilities are taken over independent copies (X,Y) and (X,Y'), independent from the training data $\mathcal{D}_n$. It is said AUC-consistent (respectively, strongly AUC-consistent), when the convergence

$$\mathrm{AUC}(\mathbf{S}_{\mathcal{D}_\mathrm{n}}(.,\mathrm{Z}))\to\mathrm{AUC}^*\ \text{as}\ \mathrm{n}\to\infty,$$

holds in probability (respectively, almost-surely), where $\mathrm{AUC}^*=\mathrm{AUC}(\mathrm{s}^*)$ for $s^*\in\mathcal{S}^*$ denotes the maximum AUC. Let $m\ge 1$. Conditioned upon $\mathcal{D}_n$, one may draw m i.i.d. copies $Z_1,\ \dots,\ Z_m$ of Z, yielding the collection Σ_m of scoring functions $\mathbf{S}_{\mathcal{D}_n}(.,Z_j)$, $1\le j\le m$. Let $\mathcal{S}_0\subset\mathcal{S}$ be a collection of scoring functions and suppose that $\bar{\mathbf{S}}_m(.,Z^{(m)})$ is a median scoring function with respect to $\mathcal{S}_0$, in the sense that:

$$\Delta_{\Sigma_m}(\bar{\mathbf{S}}_m(.,Z^{(m)}))=\min_{s\in\mathcal{S}_0}\Delta_{\Sigma_m}(s),$$

where $\Delta_{\Sigma_m}(s)=\sum_{j=1}^m d_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)})$ for $s\in\mathcal{S}$.

The next result shows that, whenever it exists, AUC consistency is preserved for such a median ranking rule (a look at the proof given in Appendix C.3 shows that the convergence rate is also preserved).

Theorem 4.1 (AUC-CONSISTENCY AND AGGREGATION.) *Assume that assumptions of Proposition 3.3 are fulfilled. Suppose that the randomized scoring function $\mathbf{S}_{\mathcal{D}_n}(.,Z)$ is* AUC*-consistent (respectively, strongly* AUC*-consistent). Let $\mathcal{S}_0\subset\mathcal{S}$ such that $\mathcal{S}^*\cap\mathcal{S}_0\neq\emptyset$ and $m\ge 1$. Suppose that, for all $n\ge 1$, there exists a median $\bar{\mathbf{S}}_m\in\mathcal{S}_0$ of m independent replications of the randomized scoring function given $\mathcal{D}_n$, in Kendall sense with respect to $\mathcal{S}_0$. Then, the aggregated scoring rule $\bar{\mathbf{S}}_m$ is* AUC*-consistent (respectively, strongly* AUC*-consistent).*

From a practical perspective, median computation is based on empirical versions of the probabilistic Kendall τ's involved, see Eq. (6). The following result reveals that this leads to scoring functions that are asymptotically median with respect to d_{τ_X}, provided that the class $\mathcal{S}_0$ over which the median is computed is not too complex.

Theorem 4.2 *(*EMPIRICAL MEDIAN COMPUTATION.*) Let $\Sigma = \{S_1, \ \ldots, \ S_K\}$ be a collection of $K \geq 1$ scoring functions and $\mathcal{S}_0 \subset \mathcal{S}$ a class of scoring functions with finite* VC *dimension. For any $s \in \mathcal{S}$, define*

$$\widehat{\Delta}_N(s) = \sum_{k=1}^{K} \widehat{d}_{\tau_X}(\preccurlyeq_s, \preccurlyeq_{S_K}),$$

where the estimate $\widehat{\tau}_X(.,.)$ of $\tau_X(.,.)$ is based on $N \geq 1$ independent copies of the r.v. X. Suppose that $\hat{\hat{s}}_N \in \mathcal{S}$ is such that $\widehat{\Delta}_N(\hat{\hat{s}}_N) = \min_{s \in \mathcal{S}_0} \widehat{\Delta}_N(s)$. Then, as $N \to \infty$, we have

$$\Delta_\Sigma(\hat{\hat{s}}_N) \to \min_{s \in \mathcal{S}_0} \Delta_\Sigma(s) \text{ with probability one,}$$

where $\Delta_\Sigma(s) = \sum_{k=1}^{K} d_{\tau_X}(\preccurlyeq_s, \preccurlyeq_{S_k})$. In addition, this convergence takes place at the rate $O_{\mathbb{P}}(n^{-1/2})$.

Combining the two preceding theorems, we finally obtain the following corollary.

Corollary 4.3 *Suppose that assumptions of Theorem 4.1 are fulfilled. Let $\mathcal{S}_0 \subset \mathcal{S}$ be of finite* VC *dimension and assume that the* AUC*-consistent randomized scoring function $\mathbf{S}_{\mathcal{D}_n}(., Z)$ belongs to $\mathcal{S}_0$. Let $m \geq 1$ and suppose in addition that, for all $n \geq 1$, there exist median scoring rules $\bar{\mathbf{S}}_m$ and $\hat{\hat{\mathbf{S}}}_m$ in $\mathcal{S}_0$ of m independent replications of the randomized scoring function given the training sample $\mathcal{D}_n$, with respect to the probabilistic Kendall τ distance and its empirical version respectively. Then, the empirical aggregated scoring rule $\hat{\hat{\mathbf{S}}}_m$ is* AUC*-consistent as n tends to ∞. Additionally, if $\mathbf{S}_{\mathcal{D}_n}(., Z)$'s convergence rate is of order $O_{\mathbb{P}}(v_n)$ with $v_n \searrow 0$, that of $\hat{\hat{\mathbf{S}}}_m$ is of order $O_{\mathbb{P}}(\sup\{v_n, 1/\sqrt{n}\})$.*

Naturally, the results stated above straightforwardly extend to any median based on a dissimilarity measure d (on the set of preorders on $\mathcal{S}$) equivalent to d_{τ_X}, *i.e.* such that $c_1 d_{\tau_X}(.,.) \leq d(.,.) \leq c_2 d_{\tau_X}(.,.)$ for $0 < c_1 \leq c_2 < \infty$. Notice additionally that more general complexity assumptions about the class $\mathcal{S}_0$ over which optimization is performed could be considered, following in the footsteps of the results established in [37]. The finite *VC* dimension case is however enough when considering ranking trees with a given maximum depth and based on a LEAFRANK procedure with a fixed maximum number of perpendicular splits for instance, see subsection 4.2 in [15].

5 NUMERICAL EXPERIMENTS

Various simulation studies have been carried out, providing considerable empirical evidence of the effectiveness of the approach proposed in this paper. It is the purpose of this section to describe some of the experimental results obtained, demonstrating how the RANKING FOREST method improves upon single ranking trees produced by the TREERANK algorithm in these situations.

Three examples are considered here, called *RF 10*, *RF 20* and *RF 10 sparse*. Datasets *RF 10* and *RF 20* have been generated using two Gaussian class distributions $H(dx)$ and $G(dx)$, on $\mathbb{R}^{10}$ and $\mathbb{R}^{20}$ respectively. For *RF 10*, both distributions have the same means ($\mu_+ = \mu_- = 0$) but different covariance matrices $\Sigma_+ = \mathbf{Id}_{10}$ and $\Sigma_- = 1.023 \cdot \mathbf{Id}_{10}$, while in case *RF 20*, the class distributions have different means ($||\mu_+ - \mu_-|| = 0.9$) and covariance matrices, $\Sigma_+ = \mathbf{Id}_{20}$ and $\Sigma_- = 1.23 \cdot \mathbf{Id}_{20}$, both at the same time. In the third situation, *RF 10 sparse* namely, $G(dx)$ and $H(dx)$ are still Gaussian probability distributions but have a 6-d marginal in common, the regression function $\eta(x)$ depending on four components of the input vector X only. In each situation, an estimate of the optimal ROC curve using a test set of size 3000 has been plotted in red line, see graphs a., b. and c. in Fig. 5.

In order to quantify the impact of bagging and random feature selection on the accuracy/stability of the resulting ranking rule, in each situation the algorithm has been run under various configurations (see Table 1) on 30 independent training samples of size $n = 2000$ and the ranking rule thus output have all been evaluated on a same test sample of size 3000. The first row of each subtable of Table 1 displays the results obtained when running the TREERANK algorithm without any bagging nor feature selection method. The enveloppe in $|| \cdot ||_\infty$ norm based on the related 30 "test" ROC curves is displayed in blue dotted line for each example in Fig. 5.

Here, RANKING FOREST has been implemented with $B = 50$ bootstrap samples of size 2000 (*i.e.* equal to the original sample size). Strategies $\mathcal{FR}_1$ and $\mathcal{FR}_2$ for random selection of features have been combined. For each bootstrap data sample, the TREERANK algorithm has been run using a fixed number of input components, selected at random, at each node of the *master ranking tree* (indicated by column "*TRK rfs*") and a fixed number of variables at each node of the subtrees describing the splits of the master ranking tree, randomly selected among those chosen at the level of the corresponding master tree node (indicated by column "*LRK rfs*"). Then, the test data have been ranked using an approximate (empirical) median scoring rule in Kendall sense, optimization being performed by means of a *simulated annealing* method (with the ranking of $\mathcal{P}_B^*$'s cells obtained by taking the median or mean ranks over the B rankings as starting point). This procedure has been repeated for each of the 30 training samples. The corresponding enveloppes in the ROC space are plotted in Figure 5, for the configurations ("*RF 10*", case 3), ("*RF 20*", case 9), and ("*RF 10 sparse*", case 3), in dotted red line. In all examples, the *master ranking tree* is of maximum depth 10, as well as the "split subtrees", except for the "*RF 10 sparse*" experiments, where the latter have depth less than 8.

Results are collected in Table 1. In each situation, the test AUC of the optimal scoring rule ($\widehat{\mathrm{AUC}}^*$), the mean of the test AUC of the scoring function produced by the ranking algorithm over the 30 training samples ($\overline{\text{AUC}}$) and its standard deviation ($\widehat{\sigma(\text{AUC})}$) have been computed. Additionally, two other indicators of the stability of the ranking algorithms considered are displayed: *Env. Red.* gives the amount of reduction (in percentage) of the area delineated by the enveloppe in the ROC space resulting from the RANKING FOREST parametrization compared with TREERANK, while $\mathbf{Stab}_\tau$ provides an estimate of the stability indicator (9) based on the probabilistic Kendall τ.

The figures speak volume. These experiments clearly show the impact of the resampling and the random feature procedures. Indeed, generally speaking, both prediction accuracy and stability are improved : the test AUC is clearly enhanced, while at the same time, ranking variability globally decreases. Notice in particular that, on the graphs displayed in Fig. 5, RANKING FOREST enveloppes are higher than those corresponding to single ranking trees at every point of the false positive rate, and their areas are much smaller. In addition, the following fact, confirmed by many other experimental simulations, is worth noticing : an adequate amount of randomization permits to increase stability, while preserving, to a certain extent, the enhancement in ranking accuracy induced by the aggregation stage. Observe in this regard that, even when applied to very "weak" tree-based ranking rules, with only one node all told, probabilistic Kendall τ aggregation undoubtedly improves upon TREERANK in the *RF 10 sparse* example (see Case No. 7 in Table 1). As explained in [38] (see also Section 3 in [15]), the possible success of a ranking algorithm lies in its ability to approximate accurately a collection of level sets of the regression function η. As the latter are quadratic in these example, they are, in general, poorly approximated by union of rectangles with axis parallel sides, such as those TREERANK builds here and one may thus reasonably guess that the refinements induced by the aggregation of *simpler* rules (due to randomization) produced more accurate level set estimates.

These empirical results only aim at illustrating the effect of the combination of rank aggregation and random feature selection on ranking accuracy/stability, the sole goal pursued here being to show how this improves upon single ranking trees built using the TREERANK algorithm. A complete and detailed empirical analysis of the merits and limitations of RANKING FOREST is beyond the scope of this paper and is one of the main subjects of a forthcoming article, where the impact of the choice of the pseudo-metric used for aggregating preorders and that of the bootstrap sample size are investigated at length among other things and comparisons with competitors such as those studied in [48], [49], [11], [50] are carried out, on real datasets in particular.

6 CONCLUDING REMARKS

The major contribution of this article is of methodological order. We showed how to apply the principles of the RANDOM FOREST approach to the ranking task. Several ways of randomizing and aggregating ranking trees, such as those produced by the TREERANK algorithm, have been rigorously described. We proposed a specific notion of *stability* in the ranking setup and provided some preliminary backround theory for ranking rule aggregation. Encouraging experimental results based on artificial data have also been obtained, demonstrating how bagging combined with feature randomization may significantly enhance ranking accuracy and stability both at the same time. Truth be told, theoretical explanations for RANKING FOREST's success in these situations are left to be found. Results obtained by [51] or [52] for the bagging approach in the classification/regression context suggest possible lines of research in this regard. At the same time, further experiments, based on real datasets in particular, will be carried out in a dedicated article in order to determine precisely the situations in which RANKING FOREST is competitive, compared to alternative ranking methods.

APPENDIX A
RANKING RULES

Throughout this paper, we call a *ranking* of the elements of a set $\mathcal{Z}$ any *total preorder* on $\mathcal{Z}$, *i.e.* a binary relation $\preceq$ for which the following axioms are checked.

1) (TOTALITY) For all $(z_1, z_2) \in \mathcal{Z}^2$, either $z_1 \preceq z_2$ or else $z_2 \preceq z_1$ holds.
2) (TRANSITIVITY) For all (z_1, z_2, z_3): if $z_1 \preceq z_2$ and $z_2 \preceq z_3$, then $z_1 \preceq z_3$.

When the assertions $z_1 \preceq z_2$ and $z_2 \preceq z_1$ hold both at the same time, we write $z_1 \asymp z_2$ and $z_1 \prec z_2$ when solely the first one is true. Assuming in addition that $\mathcal{Z}$ has finite cardinality $\#\mathcal{Z} < \infty$, the rank of any element $z \in \mathcal{Z}$ is given by

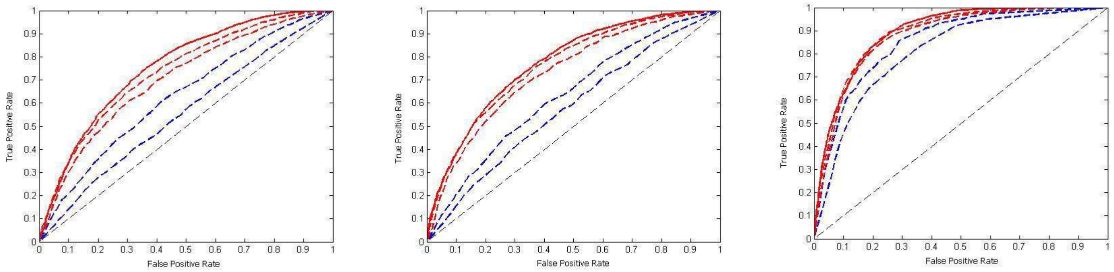
$$\mathcal{R}_{\preceq}(z) = \sum_{z' \in \mathcal{Z}} \left\{ \mathbb{I}\{z' \prec z\} + \frac{1}{2}\mathbb{I}\{z' \asymp z\} \right\},$$

when using the standard MID-RANK convention [53], *i.e.* by assigning to tied elements the average of the ranks they cover.

Any scoring function $s : \mathcal{Z} \to \mathbb{R}$ naturally defines a ranking $\preceq_s$ on $\mathcal{Z}$: $\forall (z_1, z_2) \in \mathcal{Z}^2$, $z_1 \preceq_s z_2$ iff $s(z_1) \leq s(z_2)$. Equipped with these notations, it is clear that $\preceq_{\mathcal{R}_\preceq}$ coincides with $\preceq$ for any ranking $\preceq$ on a finite set $\mathcal{Z}$.

APPENDIX B
ON COMPUTING THE LARGEST SUBPARTITION

We now briefly explain how to make crucial use of the fact that the partitions of $\mathcal{X}$ we consider here are tree-structured to compute the largest subpartition they induce. Let $\mathcal{P}_1 = \{\mathcal{C}_k^{(1)}\}_{1 \leq k \leq K_1}$ and $\mathcal{P}_2 = \{\mathcal{C}_k^{(2)}\}_{1 \leq k \leq K_2}$



a. *RF 10*: case No. 3.   b. *RF 20*: case No. 9.   c. *RF 10 sparse*: case No. 3.

Fig. 3. ROC curves and enveloppes: optimal curves (red line), RANKING FOREST enveloppes (red dotted line) and TREERANK enveloppes (blue dotted line).

TABLE 1
Impact of Resampling and Random Feature Selection on the TREERANK algorithm.

RF 10

	TRK rfs	*LRK rfs*	$\widehat{\text{AUC}}^*$	$\overline{\text{AUC}}$	$\sigma(\widehat{\text{AUC}})$	*Red. Env.*	$\mathbf{Stab}_\tau$
TreeRank	10	10	0.756	0.588	0.127	$\emptyset$	0.0084
Case No. 1	5	5	0.756	0.721	0.003	58%	0.0026
Case No. 2	10	5	0.756	0.717	0.003	55%	0.0027
Case No. 3	8	5	0.756	0.718	0.003	56%	0.0027

RF 20

	TRK rfs	*LRK rfs*	$\widehat{\text{AUC}}^*$	$\overline{\text{AUC}}$	$\sigma(\widehat{\text{AUC}})$	*Red. Env.*	$\mathbf{Stab}_\tau$
TreeRank	20	20	0.773	0.605	0.011	$\emptyset$	0.0134
Case No. 1	15	15	0.773	0.744	0.003	47%	0.0043
Case No. 2	20	15	0.773	0.743	0.002	50%	0.0042
Case No. 3	18	15	0.773	0.744	0.003	50%	0.0042
Case No. 4	10	10	0.773	0.748	0.003	54%	0.0043
Case No. 5	20	10	0.773	0.743	0.003	49%	0.0043
Case No. 6	15	10	0.773	0.744	0.004	46%	0.0043
Case No. 7	5	5	0.773	0.75	0.004	48%	0.0047
Case No. 8	20	5	0.773	0.744	0.003	46%	0.0043
Case No. 9	8	5	0.773	0.747	0.002	47%	0.0044

RF 10 sparse

	TRK rfs	*LRK rfs*	$\widehat{\text{AUC}}^*$	$\overline{\text{AUC}}$	$\sigma(\widehat{\text{AUC}})$	*Red. Env.*	$\mathbf{Stab}_\tau$
TreeRank	10	10	0.89	0.83	0.007	$\emptyset$	0.0068
Case No. 1	5	5	0.89	0.88	0.002	67%	0.0016
Case No. 2	10	5	0.89	0.88	0.001	73%	0.0014
Case No. 3	8	5	0.89	0.88	$7 \cdot 10^{-4}$	73%	0.0015
Case No. 4	3	3	0.89	0.87	0.002	58%	0.0023
Case No. 5	10	3	0.89	0.88	0.001	71%	0.0016
Case No. 6	5	3	0.89	0.88	0.002	63%	0.0019
Case No. 7	1	1	0.89	0.83	0.009	10%	0.0053
Case No. 8	10	1	0.89	0.87	0.002	60%	0.0026
Case No. 9	3	1	0.89	0.86	0.003	53%	0.0031

be two partitions of $\mathcal{X}$, related to (ranking) trees $\mathcal{T}_1$ and $\mathcal{T}_2$ respectively. For any $k \in \{1, \ldots, K_1\}$, the collection of subsets of the form $\mathcal{C}_k^{(1)} \cap \mathcal{C}_l^{(2)}$, $1 \leq l \leq K_2$, can be obtained by extending the $\mathcal{T}_1$ tree structure the following way. At the $\mathcal{T}_1$'s terminal leave defining the cell $\mathcal{C}_k^{(1)}$, add a subtree corresponding to $\mathcal{T}_2$ with root $\mathcal{C}_k^{(1)}$: the terminal nodes of the resulting subtree, starting at the global root $\mathcal{X}$, correspond to the desired collection of subsets (notice that some of these can be empty), see Fig. 4 below. Of course, this scheme can be iterated in order to recover all the cells of the subpartition induced by $B > 2$ tree-structured partitions. For obvious reasons of computational nature, one should start with the most complex tree and bind progressively less and less complex trees as one goes along.

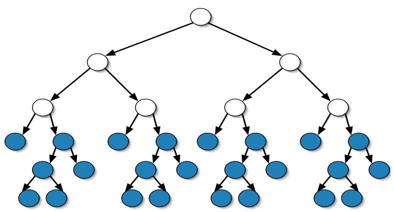


Fig. 4. CHARACTERIZING THE LARGEST SUBPARTITION INDUCED BY TREE-STRUCTURED PARTITIONS.

Appendix C
Technical Proofs

C.1 Proof of Proposition 3.2

Recall that $\tau_X(\preccurlyeq_{s_1},\preccurlyeq_{s_2}) = 1-2d_{\tau_X}(\preccurlyeq_{s_1},\preccurlyeq_{s_2})$, where $d_{\tau_X}(\preccurlyeq_{s_1},\preccurlyeq_{s_2})$ is given by:

$$\begin{aligned}\mathbb{P}\{(s_1(X)-s_1(X'))\cdot(s_2(X)-s_2(X'))<0\}\\ +\frac{1}{2}\mathbb{P}\{s_1(X)=s_1(x'),\ s_2(X)\neq s_2(X')\}\\ +\frac{1}{2}\mathbb{P}\{s_1(X)\neq s_1(x'),\ s_2(X)=s_2(X')\}.\end{aligned}$$

Observe first that, for all $s\in\mathcal{S}$, $\mathrm{AUC(s)}$ may be written as:

$$\begin{aligned}\mathbb{P}\{(s(X)-s(X'))\cdot(Y-Y')>0\}/(2p(1-p))+\\ \mathbb{P}\{s(X)=s(X'),\ Y\neq Y'\}/(4p(1-p)).\end{aligned}$$

Notice also that, using Jensen's inequality, one easily obtain that $2p(1-p)|\mathrm{AUC(s_1)}-\mathrm{AUC(s_2)}|$ is bounded by the expectation of the random variable

$$\begin{aligned}\mathbb{I}\{(s_1(X)-s_1(X'))\cdot(s_2(X)-s_2(X'))>0\}+\\ \frac{1}{2}\mathbb{I}\{s_1(X)=s_1(X')\}\cdot\mathbb{I}\{s_2(X)\neq s_2(X')\}+\\ \frac{1}{2}\mathbb{I}\{s_1(X)\neq s_1(X')\}\cdot\mathbb{I}\{s_2(X)=s_2(X')\},\end{aligned}$$

which is equal to $d_{\tau_X}(\preccurlyeq_{s_1},\preccurlyeq_{s_2})=(1-\tau_X(\preccurlyeq_{s_1},\preccurlyeq_{s_2}))/2$.

C.2 Proof of Proposition 3.3

Recall first that, for all $s\in\mathcal{S}$, the AUC deficit $2p(1-p)\{\mathrm{AUC}^*-\mathrm{AUC(s)}\}$ may be written as

$$\begin{aligned}\mathbb{E}\left[|\eta(X)-\eta(X')|\cdot\mathbb{I}\{(X,X')\in\Gamma_s\}\right]\\ +\mathbb{P}\{s(X)=s(X'),\ (Y,Y')=(-1,+1)\},\end{aligned}$$

with

$$\Gamma_s=\{(x,x')\in\mathcal{X}^2:\ (s(x)-s(x'))\cdot(\eta(x)-\eta(x'))<0\},$$

refer to Example 1 in [37] for instance. Now, Hölder inequality combined with noise condition (5) shows that $\mathbb{P}\{(X,X')\in\Gamma_s\}$ is bounded by

$$\left(\mathbb{E}\left[|\eta(X)-\eta(X')|\cdot\mathbb{I}\{(X,X')\in\Gamma_s\}\right]\right)^{a/(1+a)}\times c^{1/(1+a)}.$$

Therefore, we have for all $s^*\in\mathcal{S}^*$:

$$d_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{s^*})=\mathbb{P}\{(X,X')\in\Gamma_s\}+\frac{1}{2}\mathbb{P}\{s(X)=s(X')\}.$$

Notice that $p(1-p)\mathbb{P}\{s(X)=s(X')\mid(Y,Y')=(-1,+1)\}$ can be written as $\mathbb{E}[\mathbb{I}\{s(X)=s(X')\}\cdot\eta(X')(1-\eta(X))]$, which is larger than $\epsilon^2\cdot\mathbb{P}\{s(X)=s(X')\}$ by assumption. Using concavity of $t\geq 0\mapsto t^{a/(1+a)}$ and the bound previously established, we eventually obtain the desired result.

C.3 Proof of Theorem 4.1

By virtue of Proposition 3.2, we have:

$$\mathrm{AUC}^*-\mathrm{AUC}(\bar{\mathbf{S}}_{\mathrm{m}})\leq\frac{\mathrm{d}_{\tau_{\mathrm{X}}}(\preccurlyeq_{\mathrm{s}^*},\preccurlyeq_{\bar{\mathbf{S}}_{\mathrm{m}}})}{2\mathrm{p}(1-\mathrm{p})},$$

for any $s^*\in\mathcal{S}^*$. Using now triangular inequality, one gets

$$\begin{aligned}d_{\tau_X}(\preccurlyeq_{s^*},\preccurlyeq_{\bar{\mathbf{S}}_m})\leq d_{\tau_X}(\preccurlyeq_{s^*},\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)})\\ +d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{\bar{\mathbf{S}}_m}),\end{aligned}$$

for all $j\in\{1,\ \dots,\ m\}$. Averaging then over j and using the fact that, if one chooses s^* in $\mathcal{S}_0$,

$$\sum_{j=1}^{m}d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{\bar{\mathbf{S}}_m})\leq\sum_{j=1}^{m}d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{s^*}),$$

one obtains that

$$d_{\tau_X}(\preccurlyeq_{s^*},\preccurlyeq_{\bar{\mathbf{S}}_m})\leq\frac{2}{m}\sum_{j=1}^{m}d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{s^*}).$$

The desired result finally follows from Proposition 3.3 combined with the consistency assumption of the randomized scoring function.

C.4 Proof of Theorem 4.2

Observe that we have:

$$\begin{aligned}\Delta_\Sigma(\hat{\bar{s}}_N)-\min_{s\in\mathcal{S}_0}\Delta_\Sigma(s)\leq 2\cdot\sup_{s\in\mathcal{S}_0}|\widehat{\Delta}_N(s)-\Delta_\Sigma(s)|\\ \leq 2\sum_{k=1}^{K}\sup_{s\in\mathcal{S}_0}|\widehat{d}_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{S_k})-d_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{S_k})|.\end{aligned}$$

Now, it results from the strong Law of Large Numbers for U-processes stated in Corollary 5.2.3 in [54] that $\sup_{s\in\mathcal{S}_0}|\widehat{d}_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{S_k})-d_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{S_k})|\to 0$ as $N\to\infty$, for all $k=1,\ \dots,\ K$. The convergence rate $O_{\mathbb{P}}(n^{-1/2})$ follows from the Central Limit Theorem for U-processes given in Theorem 5.3.7 in [54].

C.5 Proof of Corollary 4.3

Reproducing the argument of Theorem 4.1, one gets:

$$\begin{aligned}d_{\tau_X}(\preccurlyeq_{s^*},\preccurlyeq_{\hat{\bar{\mathbf{S}}}_{N,m}})\leq\frac{1}{m}\sum_{j=1}^{m}d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{s^*})\\ +\frac{1}{m}\sum_{j=1}^{m}d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{\hat{\bar{\mathbf{S}}}_{N,m}}).\end{aligned}$$

As in Theorem 4.2's proof, we also have:

$$\begin{aligned}\frac{1}{m}\sum_{j=1}^{m}\{d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{\hat{\bar{\mathbf{S}}}_{N,m}})-d_{\tau_X}(\preccurlyeq_{\mathbf{S}_{\mathcal{D}_n}(.,Z_j)},\preccurlyeq_{\bar{\mathbf{S}}_{N,m}})\}\\ \leq 2\cdot\sup_{(s,s')\in\mathcal{S}_0^2}|\widehat{d}_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{s'})-d_{\tau_X}(\preccurlyeq_s,\preccurlyeq_{s'})|.\end{aligned}$$

Using again Corollary 5.2.3 in [54], we obtain that the term on the right hand side of the bound above vanishes as $N\to\infty$. Now the desired result immediately follows from Theorem 4.1.

References

[1] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[2] S. Clémençon and N. Vayatis, "Tree-based ranking methods," *IEEE Transactions on Information Theory*, vol. 55, no. 9, pp. 4316–4336, 2009.

[3] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Wadsworth and Brooks, 1984.

[4] V. Svetnik, A. Liaw, C. Tong, J. Culberson, R. Sheridan, and B. Feuston, "Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling," *J. Chem. Inf. Comput. Sci.*, vol. 43, no. 6, pp. 1947–1958, 2003.

[5] R. Diaz-Uriarte and S. A. de Andrés, "Gene selection and classification of microarray data using random forest," *BMC Bioinformatics*, vol. 7, no. 3, 2006.

[6] J. Egan, *Signal Detection Theory and ROC Analysis*. Academic Press, 1975.

[7] S. Clémençon, G. Lugosi, and N. Vayatis, "Ranking and scoring using empirical risk minimization," in *Proceedings of COLT*, 2005.

[8] S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, and D. Roth, "Generalization bounds for the area under the ROC curve," *J. Mach. Learn. Res.*, vol. 6, pp. 393–425, 2005.

[9] A. Rakotomamonjy, "Optimizing Area Under Roc Curve with SVMs," in *Proceedings of the First Workshop on ROC Analysis in AI*, 2004.

[10] S. Clémençon and N. Vayatis, "Empirical performance maximization based on linear rank statistics," in *NIPS*, ser. Lecture Notes in Computer Science, vol. 3559. Springer, 2009, pp. 1–15.

[11] 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, 2003.

[12] 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*, 2002, pp. 139–146.

[13] S. Clémençon and N. Vayatis, "Overlaying classifiers: a practical approach for optimal ranking." in *Advances in Neural Information Processing Systems 21. Proceedings of NIPS'08*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., 2009, pp. 313–320.

[14] ——, "Tree-structured ranking rules and approximation of the optimal ROC curve," in *Proceedings of ALT*. Springer, 2008.

[15] S. Clémençon, M. Depecker, and N. Vayatis, "Adaptive partitioning schemes for bipartite ranking," HAL, Tech. Rep. hal-00268068, 2009. [Online]. Available: /hal.archives-ouvertes.fr/hal-00268068/fr/

[16] J. Barthélémy and B.Montjardet, "The median procedure in cluster analysis and social choice theory," *Mathematical Social Sciences*, vol. 1, pp. 235–267, 1981.

[17] D. Pennock, E. Horvitz, and C. Giles, "Social choice theory and recommender systems: analysis of the foundations of collaborative filtering," in *National Conference on Artificial Intelligence*, 2000, pp. 729–734.

[18] P. Diaconis, "A generalization of spectral analysis with application to ranked data," *The Annals of Statistics*, vol. 17, no. 3, pp. 949–979, 1989.

[19] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, and E. Vee, "Comparing partial rankings," *SIAM J. Discrete Mathematics*, vol. 20, no. 3, pp. 628–648, 2006.

[20] N. Baskiotis, S. Clémençon, M. Depecker, and N. Vayatis, "R-implementation of the TreeRank algorithm," *Submitted for publication*, 2009.

[21] S. Arlot, "Model selection by resampling techniques," *Electronic Journal of Statistics*, vol. 3, pp. 557–624, 2009.

[22] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 26, pp. 123–140, 1996.

[23] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, "Rank aggregation methods for the Web," in *Proceedings of the 10th International WWW conference*, 2001, pp. 613–622.

[24] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, and E. Vee, "Comparing and aggregating rankings with ties," in *Proceedings of the 12-th WWW conference*, 2003, pp. 366–375.

[25] I. Ilyas, W. Aref, and A. Elmagarmid, "Joining ranked inputs in practice," in *Proceedings of the 28th International Conference on Very Large Databases*, 2002, pp. 950–961.

[26] N. Betzler, M. Fellows, J. Guo, R. Niedermeier, and F. Rosamond, "Computing kemeny rankings, parameterized by the average kt-distance," in *Proceedings of the 2nd International Workshop on Computational Social Choice*, 2008.

[27] B. Mandhani and M. Meila, "Tractable search for learning exponential models of rankings," in *Proceedings of AISTATS, Vol. 5 of JMLR:W&CP 5*, 2009.

[28] M. Meila, K. Phadnis, A. Patterson, and J. Bilmes, "Consensus ranking under the exponential model," in *Conference on Artificial Intelligence (UAI)*, 2007, pp. 729–734.

[29] M. Fligner and J. Verducci (Eds.), *Probability Models and Statistical Analyses for Ranking Data*. Springer, 1993.

[30] G. Lebanon and J. Lafferty, "Conditional models on the ranking poset," in *Proceedings of NIPS'03*, 2003.

[31] J. G. Kemeny, "Mathematics without numbers," *Daedalus*, no. 88, pp. 571–591.

[32] M. Bansal and D. Fernandez-Baca, "Computing distances between partial rankings," *Information Processing Letters*, vol. 109, pp. 238–241, 2009.

[33] P. Mielke and K. Berry, *Permutation methods*. Springer, 2001.

[34] S. Clémençon and N. Vayatis, "Ranking the best instances," *Journal of Machine Learning Research*, vol. 8, pp. 2671–2699, 2007.

[35] J. Howie, "Hyperbolic groups," *In Groups and Applications, edited by V. Metaftsis, Ekdoseis Ziti, Thessaloniki*, pp. 137–160, 2000.

[36] M. Deza and E. Deza, *Encyclopedia of Distances*. Springer, 2009.

[37] S. Clémençon, G. Lugosi, and N. Vayatis, "Ranking and empirical risk minimization of U-statistics," *The Annals of Statistics*, vol. 36, no. 2, pp. 844–874, 2008.

[38] S. Clémençon and N. Vayatis, "Overlaying classifiers: a practical approach to optimal scoring," *To appear in Constructive Approximation*, 2009.

[39] Y. Wakabayashi, "The complexity of computing medians of relations," *Resenhas*, vol. 3, no. 3, pp. 323–349, 1998.

[40] O. Hudry, "Computation of median orders: complexity results," *Annales du LAMSADE: Vol. 3. Proceedings of the workshop on computer science and decision theory, DIMACS*, vol. 163, pp. 179–214, 2004.

[41] ——, "NP-hardness results for the aggregation of linear orders into median orders," *Ann. Oper. Res.*, vol. 163, pp. 63–88, 2008.

[42] J. Spall, *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. John Wiley & Sons, 2003.

[43] I. Charon and O. Hudry, "Lamarckian genetic algorithms applied to the aggregation of preferences," *Annals of Operations Research*, vol. 80, pp. 281–297, 1998.

[44] V. C. M. Laguna, R. Marti, "Intensification and diversification with elite tabu search solutions for the linear ordering problem," *Computers and Operations Research*, vol. 26, no. 12, pp. 1217–1230, 1999.

[45] P. Fishburn, *The Theory of Social Choice*. University Press, Princeton, 1973.

[46] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, "Label ranking by learning pairwise preferences," *Artificial Intelligence*, vol. 172, pp. 1897–1917, 2008.

[47] G. Biau, L. Devroye, and G. Lugosi, "Consistency of Random Forests," *J. Mach. Learn. Res.*, vol. 9, pp. 2039–2057, 2008.

[48] J. T., "Optimizing search engines using clickthrough data," in *KDD'02 - Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM*, 2002, pp. 133–142.

[49] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender, "Learning to rank using gradient descent," in *Proceedings of the 22nd International Conference on Machine Learning*. ACM International Conference Proceeding Series **119**, 2005, pp. 89–96.

[50] T. Pahikkala, E. Tsivtsivadze, A. Airola, J. Boberg, and T. Salakoski, "Learning to rank with pairwise regularized least-squares," in *Proceedings of SIGIR 2007 Workshop on Learning to Rank for Information Retrieval*, 2007, pp. 27–33.

[51] J. Friedman and P. Hall, "On bagging and non-linear estimation," *Journal of statistical planning and inference*, vol. 137, no. 3, pp. 669–683, 2007.

[52] Y. Grandvalet, "Bagging Equalizes Influence," *Machine Learning*, vol. 55, pp. 251–270, 2004.

[53] M. Kendall, "The Treatment of Ties in Ranking Problems," *Biometrika*, no. 33, pp. 239–251, 1945.

[54] V. de la Pena and E. Giné, *Decoupling: from Dependence to Independence*. Springer, 1999.

PLACE
PHOTO
HERE

Stéphan Clémençon received the Ph.D. degree in applied mathematics from the University Denis Diderot, Paris 7, France, in 2000. In October 2001, he joined the faculty of the University Paris X as Associate Professor and successfully defended his habilitation thesis in 2006. He worked as Director of Research for the Department of Mathematics and Computer Science of the National Institute of Research in Agronomy between 2005 and 2007. Since October 2007, he has been professor and researcher with Telecom ParisTech, the leading school in the field of information technologies in France. His research interests include machine-learning, Markov processes, computational harmonic analysis, and nonparametric statistics.