Error curves for evaluating the quality of feature rankings

Ivica Slavkov[1], **Matej Petković**[1,2], **Pierre Geurts**[3], **Dragi Kocev**[1,2], **and Sašo Džeroski**[1,2]

[1]**Jožef Stefan Institute, Ljubljana, Slovenia**
[2]**Jožef Stefan International Postgraduate School, Ljubljana, Slovenia**
[3]**University of Liège, Liège, Belgium**

Corresponding author:
Matej Petković[1,2]

Email address: matej.petkovic@ijs.si

ABSTRACT

In this paper, we propose a method for evaluating feature ranking algorithms. A feature ranking algorithm estimates the importance of descriptive features when predicting the target variable, and the proposed method evaluates the correctness of these importance values by computing the error measures of two chains of predictive models. The models in the first chain are built on nested sets of top-ranked features, while the models in the other chain are built on nested sets of bottom ranked features. We investigate which predictive models are appropriate for building these chains, showing empirically that the proposed method gives meaningful results and can detect differences in feature ranking quality. This is first demonstrated on synthetic data, and then on several real-world classification benchmark problems.

INTRODUCTION

In the era of data abundance, we face high-dimensional problems increasingly often. Sometimes, prior to applying predictive modeling (e.g., classification) algorithms to such problems, dimensionality reduction may be necessary for a number of reasons, including computational reasons. By keeping only a limited number of descriptors (features), a classifier can also achieve better predictive performance, since typically, a portion of the features strongly influence the target variable, and the others can be understood as (mostly) noise. This dimensionality reduction corresponds to the task of feature selection (Guyon et al., 2002). A task related to it is feature ranking. This is a generalization of feature selection where, in addition to simply telling apart relevant features from irrelevant ones (Nilsson et al., 2007), one also assesses how relevant are they for predicting the target variable.

In machine learning, feature ranking is typically seen either as a preprocessing or as a postprocessing step. In the former case, one actually tackles the feature selection problem by first computing the feature relevance values, and then keeping only the features whose relevance is above some user defined threshold. In the second case, feature ranking is obtained after building a predictive model in order to explain it, e.g., (Arceo-Vilas et al., 2020). For black box models, such as neural networks, this may be the only way to understand their predictions.

In some application domains, such as biology or medicine, feature ranking may be the main point of interest. If we are given data about the expression of numerous genes, for a group of patients, and the patients' clinical state (diseased/healthy), one can find good candidate genes that influence the health status of the patients, which gives us a deeper understanding of the disease.

Due to the prominence of the feature ranking task, there exist many feature ranking methods. Simpler methods assess the relevance of each feature independently ignoring the other features (χ^2 statistics, mutual information of the feature and the target variable) and their possible interactions. A prominent example that shows the myopic nature of such approaches is the case when the target variable y is defined as $y = \text{XOR}(x_1, x_2)$ where x_1 and x_2 are two binary features. Ignoring x_1 when computing the relevance of x_2 (and vice-versa) would result in assessing x_1 as completely irrelevant, i..e, as random noise. More

46 sophisticated methods assess relevance of each feature in the context of the others. They are typically
47 based on some predictive model, e.g., Random Forest feature ranking (Breiman, 2001), or optimisation
48 problem (Nardone et al., 2019), but not necessarily, cf. e.g., ReliefF (Robnik-Šikonja and Kononenko,
49 2003) and the work of Li and Gu (2015).
50   However, there is no unified definition of feature importance, and actually, every feature ranking
51 algorithm comes with its own (implicit) definition. Therefore, different methods typically introduce
52 different feature importance scores: Deciding which of them is the best is a very relevant, but also very
53 challenging task that we would like to address in this paper. More precisely, we continue and extend
54 our previous work (Slavkov et al., 2018), where we proposed and evaluated a quantitative score for the
55 assessment of feature rankings. Here, we propose a new feature ranking evaluation method that can
56 evaluate feature rankings in a relative sense (deciding which of the feature rankings is better), or in an
57 absolute sense (assessing how good is a feature ranking). The method is based on constructing two
58 chains of predictive models that are built from the top-ranked and bottom-ranked features. The predictive
59 performances of the models in the chain are then shown on graphs of so called forward feature addition
60 (FFA) and reverse feature addition (RFA) curves, which reveal how the relevant features are distributed in
61 the ranking(s). An important property of the proposed method is that it does not need any prior ground
62 truth knowledge of the data.
63   We investigate the performance of the proposed evaluation approach under a range of scenarios. To
64 begin with, we prove the potential of the FFA and RFA curves by using them in setting which employs
65 synthetic data. Next, we investigate the use of different types of predictive models for constructing the
66 curves, thus considerably extending the preliminary experiments by Slavkov et al. (2018). Furthermore,
67 we apply the proposed evaluation approach to a large collection of benchmark datasets. Compared to
68 Slavkov et al. (2018), we have included 11 new high-dimensional datasets. The results of the evaluation,
69 in a nutshell, show that the FFA and RFA curves are able to discern the best ranking among multiple
70 proposed feature rankings.
71   The remainder of this paper is organized as follows. Sec. *Related Work* outlines related work, Sec. *A*
72 *Method for Evaluating Feature Rankings* describes in detail the proposed method for constructing error
73 curves. Next, Sec. *Empirical Evaluation of FFA/RFA Curve Properties* discusses the properties of the
74 error curves when applied to synthetic data. We then give the results of the experimental evaluation on
75 benchmark datasets in Sec. *Feature Ranking Comparison on Real-Worlds Datasets*. Sec. *Conclusions*
76 concludes with a summary of our contributions and an outline of possible directions for further work. In
77 the appendices, we give additional information about generating synthetic data (Appendix A1), measuring
78 distance between rankings (Appendix A2), and comparative evaluation of feature ranking methods
79 (Appendix A3). In Appendix A4, detailed experimental results are given.

RELATED WORK

81 The evaluation of feature rankings is a complex and unsolved problem. Typically, feature rankings are
82 evaluated on artificially generated problems, while evaluation on real world problems remains an issue
83 approached indirectly. To begin with, when the ground truth ranking is known, one can transform the
84 problem of feature ranking evaluation into an evaluation of classification predictive model (Jong et al.,
85 2004) as follows. First, a ranking is computed. Then, for every threshold, the numbers of relevant features
86 (true positives) and irrelevant features (false positives) with the feature relevance above the threshold are
87 computed. From these values, a ROC curve can be created and the area underneath it computed.
88   Another possible approach is to compute separability (Robnik-Šikonja and Kononenko, 2003), i.e.,
89 the minimal difference between the feature importance of a relevant feature and the feature importance
90 of an irrelevant feature. If this difference is positive, then the relevant features are separated from the
91 irrelevant ones, otherwise they are mixed.
92   However, both approaches are more applicable to feature selection problems and are too coarse
93 for feature rankings problem, since they only differentiate between relevant and irrelevant features.
94 Spearman's rank correlation coefficient between the computed and the ground truth ranking might be
95 more appropriate.
96   The main shortcoming of the upper approaches is that they demand the ground truth ranking. In
97 real world scenarios, this is not known, which makes the upper approaches useless. Nevertheless, using
98 synthetic data and the controlled environment offers a good starting point for showing the usefulness of a
99 feature ranking evaluation method, as we shall also see later.

An approach that overcomes the issue of unknown ground truth ranking bases on selecting k top-ranked features and building a predictive model that uses only these features to predict target variable. The ranking whose top-ranked features result in the model with the highest predictive performance, is proclaimed the best. Since it is now always clear which value of k should be chosen, this can be done for multiple values of k (Guyon et al., 2002; Furlanello et al., 2003; Paoli et al., 2005; Verikas et al., 2011).

In addition to correctness, rankings stability is sometimes also part of the evaluation. The stability of a ranking algorithm can be measured by comparing the feature rankings obtained, for example, from the different bootstrap replicates of a dataset or from the folds in cross-validation (Guzmán-Martínez and Alaiz-Rodríguez, 2011; Kalousis et al., 2007; Jurman et al., 2008). In (Saeys et al., 2008) both stability and predictive performance are combined into a single feature ranking quality index.

Also, notions similar to FFA curves (without any particular name, though) as the feature ranking evaluation method can be found in the literature ((Liu et al., 2003), (Duch et al., 2004), (Biesiada et al., 2005) and (Liang et al., 2008)). However, to the best of our knowledge, there is no discussion and detailed investigation why FFA curves are an appropriate method for comparing feature rankings, nor which learning methods should (or should not) be used for constructing them.

A METHOD FOR EVALUATING FEATURE RANKINGS

First of all, every feature ranking method should be able to tell apart relevant features from irrelevant ones (Nilsson et al., 2007). In addition to that, the method should order the features with respect to the target variable, awarding the most relevant ones the top ranks.

If ground truth ranking exists, the method should return this ranking in the optimal case. The worst case is more complicated and has two possible answers. One is the inverse of the ground truth ranking. However, since the ground truth ranking is typically not known in real-world scenarios, a more useful definition of the worst ranking is random ranking. This ranking also contains as little information about the distribution of the relevant features in the ranking as possible. Moreover, this distribution can be always assessed and is the cornerstone of our ranking evaluation method.

The Evaluation Method

First, we define the notation used in the rest of the paper: $\mathscr{D}$ denotes a dataset whose columns are input features F_i that form a set $\mathscr{F}$, and the target feature F_t. A feature ranking method takes the dataset as an input, and returns a list $\mathbf{R} = (F_{(1)}, \ldots, F_{(n)})$ as the output, where $F_{(i)}$ is the feature with the rank i.

We evaluate a ranking $\mathbf{R}$ by evaluating different subsets $\boldsymbol{S}$ of features $\mathscr{F}$. This is done by building a predictive model $\mathscr{M}(\boldsymbol{S}, F_t)$ and assessing its predictive power. The evaluation of the predictive model provides a cumulative assessment of the information contained in the feature set $\boldsymbol{S}$ and it can be quantified with an error measure $\mathrm{err}(\mathscr{M}(\boldsymbol{S}, F_t))$. The question is how to generate the feature subsets from the feature ranking, so that the error estimates can provide insight into the correctness of the feature ranking and constitute an evaluation thereof.

The construction of the feature subsets should be guided by the feature ranking $\mathbf{R}$. Starting from the top ranked feature $F_{(1)}$ and going towards the bottom ranked feature $F_{(n)}$, the feature relevance should decrease. Following this basic intuition, we propose two methods for constructing feature subsets from the feature ranking: *forward feature addition* (FFA) and *reverse feature addition* (RFA).

FFA constructs the feature subsets $\boldsymbol{S}^i$ by considering the i highest ranked features, starting with $\boldsymbol{S}^1 = \{F_{(1)}\}$. The next set $\boldsymbol{S}^{i+1}$ is constructed by adding the next lower-ranked feature, namely $\boldsymbol{S}^{i+1} = \boldsymbol{S}^i \cup \{F_{(i+1)}\}$. The process continues until $i = n$ and $\boldsymbol{S}^n$ contains all of the features from $\mathbf{R}$.

RFA produces feature sets $\boldsymbol{S}_i$ constructed in an opposite manner to FFA. We start with $\boldsymbol{S}_1 = \{\mathscr{F}_{(n)}\}$ that contains only the lowest ranked feature. The next feature set $\boldsymbol{S}_{i+1}$ is constructed by adding the lowest-ranked feature which is not in $\boldsymbol{S}_i$, namely $\boldsymbol{S}_{i+1} = \boldsymbol{S}_i \cup \{F_{(n-i)}\}$. In the same way as for FFA, the process of RFA continues until we include all of the features, i.e., $\boldsymbol{S}_n = \mathscr{F}$.

Note that FFA can be viewed as backward feature elimination. Starting from $\boldsymbol{S}^n = \mathscr{F}$, at each step we remove the least relevant feature from $\boldsymbol{S}^i$ to obtain $\boldsymbol{S}^{i-1}$. Similarly, RFA can be viewed as forward feature elimination. Finally, it holds that $\mathscr{F} = \boldsymbol{S}^{n-i} \cup \boldsymbol{S}_i$ for all i.

For each i and each constructed feature subset $\boldsymbol{S}^i$ or $\boldsymbol{S}_i$, we build predictive models $\mathscr{M}(\boldsymbol{S}^i, F_t)$ and $\mathscr{M}(\boldsymbol{S}_i, F_t)$. We then estimate their respective prediction errors, err^i and err_i. This results in two error curves. We name them FFA and RFA curves, each constructed by the corresponding FFA/RFA feature subset construction method. The value for each point of the FFA curve is defined as $\mathrm{FFA}(i) = \mathrm{err}^i$, while

Algorithm 1 Generation of the FFA and RFA curves.

Input: Feature Ranking $\mathbf{R} = (F_{(1)}, \ldots, F_{(n)})$, Target Feature F_t, type of curve (FFA or RFA)

$\quad S \leftarrow \emptyset$
$\quad E \leftarrow$ list of length n
$\quad$**for** $i = 1, 2, \ldots, n$ **do**
$\quad\quad$**if** curve type is FFA **then**
$\quad\quad\quad S \leftarrow S \cup \{F_{(i)}\}$
$\quad\quad$**else**
$\quad\quad\quad S \leftarrow S \cup \{F_{(n-i+1)}\}$
$\quad\quad E[i] \leftarrow \mathrm{err}(\mathcal{M}(S, F_t))$
$\quad$**return** E
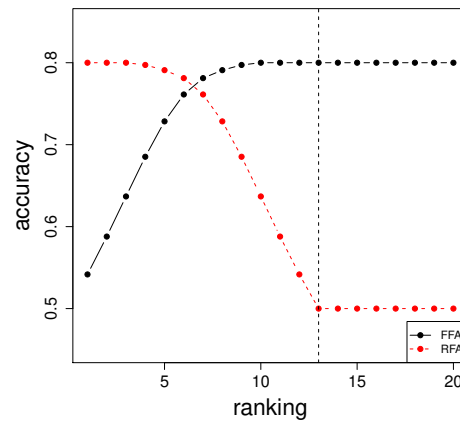
Figure 1. Sample FFA and RFA curves

for the RFA curve as $\mathrm{RFA}(i) = err_i$. The process of FFA/RFA curve construction is summarised in Algorithm 1.

The computational complexity of the proposed algorithm for constructing a single (FFA or RFA) curve is $\mathcal{O}(n(M + T))$, where n is the number of features, $M = M(n)$ is the cost of constructing the predictive model and $T = T(n)$ is the cost of its evaluation. It should be noted that M and T are dependent on the specific learning method used for inducing the model and on the procedure used for evaluating it.

Typically, the points $\mathrm{FFA}(i)$, $\mathrm{FFA}(i+1)$, $\cdot$, do not differ considerably, for i large enough, since it expected that only a small proportion of the features is relevant when the data is high-dimensional. This means that we can make the algorithm more efficient if we construct the set $S^{i+\delta(i)}$ from the set S^i by including $\delta(i)$ features into it. Analogously, we speed up the construction of the RFA curves.

Interpretation of Error Curves

The visualization and interpretation of the FFA and RFA curves can be explained by considering the examples of FFA and RFA curves given in Fig. 1. The y-axis for both curves is the same and depicts the error estimate of a feature subset. Point i at the x-axis corresponds to the moment when the feature $F_{(i)}$ is first included in the predictive model: S^i for the FFA curve and S_{n-i+1} for the RFA curve. Thus, the FFA curve in Fig. 1 is constructed from left-to-right as the top-ranked features are at the beginning of the ranking. In contrast, the RFA curve is constructed from right-to-left starting with the end of the ranking and going towards its beginning.

Let us first focus on the FFA curve. We can observe that as the number k of features increases, the accuracy of the predictive models also increases. This can be interpreted as follows: By adding more and more of the top-k ranked features, the number of relevant features in the constructed feature subsets increases, which is reflected in the improvement of the accuracy (error) measure.

Next, for the RFA curve in Fig. 1, if we inspect it from right-to-left, we can notice that it is quite different from the FFA curve at the beginning. Namely, the accuracy of the models constructed with the bottom ranked features is minimal, which means the ranking is correct in the sense that it puts only
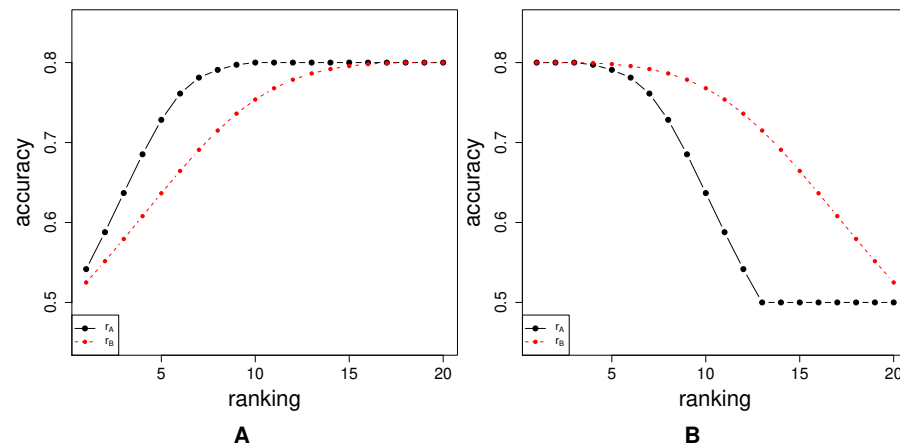
Figure 2. Comparison of FFA curves (A) and RFA curves (B) of two ranking methods r_A and r_B.

178 irrelevant features at the bottom of the ranking. As the number of bottom-k features increases, some
179 relevant features are included and the accuracy of the models increases.

180   We now consider the complete Fig. 1. The FFA and RFA curve essentially provide an estimate of how
181 the relevant features are spread throughout the feature ranking. Namely, the FFA curve provides us with
182 an estimate of where the relevant features appear at the top of the ranking, while the RFA curve provides
183 an estimate of where relevant features appear at the bottom of the ranking. In the specific case depicted in
184 Fig. 1, the relevant features are located between the 1st and the 13th ranked feature.

185   Besides providing an estimate of the spread of the relevant features across the feature ranking, the real
186 utility of the FFA/RFA curves becomes apparent if we consider them in a relative, or more precisely, a
187 comparative context. Let us consider two arbitrary feature ranking methods r_A and r_B, which produce
188 feature rankings $\mathbf{R}_A$ and $\mathbf{R}_B$, respectively. For these two rankings, we present the corresponding FFA/RFA
189 curves in Fig. 2.

190   We first inspect the FFA curves visually. We find that the values of the FFA curve of the ranking
191 method r_A are (most of the time) above the FFA curve of the other ranking method r_B. This can be
192 interpreted in the following way: for an arbitrary k, when considering the top-k features of the feature
193 rankings $\mathbf{R}_A$ and $\mathbf{R}_B$, more relevant features are included in the top-k features of ranking $\mathbf{R}_A$ than the top-k
194 features of ranking $\mathbf{R}_B$. This implies that ranking algorithm r_A produces a better ranking as compared to
195 the ranking algorithm r_B.

196   A similar discussion applies to the RFA curve. When one considers the bottom-k features of a given
197 feature ranking, most of the time, feature ranking $\mathbf{R}_A$ includes less relevant features than feature ranking
198 $\mathbf{R}_B$, i.e., the predictive models constructed are less accurate. Here, because the opposite logic of the
199 FFA curve applies, one can also conclude that the feature ranking algorithm r_A produces a better feature
200 ranking than the feature ranking algorithm r_B.

Expected FFA and RFA Curves

202 When one wants to assess the quality of a single feature ranking in a real-world application, its forward
203 (reverse) feature addition curves can be only compared to the curves that belong the ranking, generated
204 uniformly at random, since the ground-truth ranking is not known. As discussed before, the random
205 ranking $\mathbf{R}_{\mathrm{RND}}$ is the worst-case ranking, since it contains no information about the distribution of the
206 relevant features. As such, it can also serve as a baseline.

207   The expected values of the points that define FFA curve of the ranking $\mathbf{R}_{\mathrm{RND}}$ coincide with the
208 expected values of the RFA curve of this ranking, since the corresponding values only depend on the data
209 itself and the number of features i at a given point of the curves. Thus, *expected curves* can be the common
210 name for both types of the curves that belong to $\mathbf{R}_{\mathrm{RND}}$. Computing the exact average error estimations
211 $\mathbb{E}_{\boldsymbol{S}}[\mathrm{err}^i] = \mathbb{E}_{\boldsymbol{S}}[\mathrm{err}_i]$, where $\boldsymbol{S} \subseteq \mathscr{F}, |\boldsymbol{S}| = i$, may be unfeasible if the number of features n is large (e.g., for
212 $i = n/2$, $\mathscr{O}((2n)!/(n!)^2)$ models have to be evaluated), but one can overcome this by sampling the sets $\boldsymbol{S}$.

sufficient detail provided

Stability of feature ranking

An important aspect of feature ranking algorithms is their stability (Nogueira et al., 2017) or, more specifically, the stability of the ranked feature lists that they produce. Once we have the set $\mathscr{R}$ of m rankings $\boldsymbol{R}_t$ that were induced from the different samples of the dataset $\mathscr{D}$, the stability index $S(\mathscr{R})$ can be calculated as

$$St(\mathscr{R}) = \frac{1}{\binom{m}{2}} \sum_{t=1}^{m-1} \sum_{s=t+1}^{m} S_M(\boldsymbol{R}_t, \boldsymbol{R}_s),$$

i.e., the stability index is the average of pairwise similarities S_M for each pair of rankings. In general, the function S_M can be any (dis)similarity measure, e.g., the Spearman rank correlation coefficient (Saeys et al., 2008; Khoshgoftaar et al., 2013), the Jaccard distance (Saeys et al., 2008; Kalousis et al., 2007), an adaptation of the Tanimoto distance (Kalousis et al., 2007), Fuzzy (Goodman and Kruskal's) gamma coefficient (Boucheham and Batouche, 2014; Henzgen and Hüllermeier, 2015), etc.

To assess the stability of feature ranking in our experimental work, we set $S_M = \text{Ca}$, where Ca is the Canberra distance (Lance and Williams, 1966, 1967; Jurman et al., 2008). This is a weighted distance metric that puts bigger emphasis on the stability of the top ranked features. If we have two feature rankings $\boldsymbol{R}_A$ and $\boldsymbol{R}_B$ of n features, then the Canberra distance is defined as

$$\text{Ca}(\boldsymbol{R}_A, \boldsymbol{R}_B) = \sum_{j=1}^{n} \frac{|rank_A(F_j) - rank_B(F_j)|}{rank_A(F_j) + rank_B(F_j)}. \tag{1}$$

However, we do not only estimate the stability of the ranking as a whole. Rather, we also estimate the stability of the partial rankings based on the features from $\boldsymbol{S}^i$. In order for the distance to be applicable to such partial rankings with $i < n$ features, the following adaptation is proposed: instead of using the ranks $rank_{A,B}(F)$, we use $rank_{A,B}^i(F) = \min\{rank_{A,B}(F), i+1\}$, i.e., all features with rank higher than i are treated as if they had rank $i+1$:

$$\text{Ca}(\boldsymbol{R}_A, \boldsymbol{R}_B) = \sum_{j=1}^{n} \frac{|rank_A^i(F_j) - rank_B^i(F_j)|}{rank_A^i(F_j) + rank_B^i(F_j)}. \tag{2}$$

Additionally, we would like the stability indicator to be independent of specific values of i and n. Hence, we normalize it by the expected Canberra distance between random rankings, denoted by $\text{Ca}(n, i)$. It can be approximated (Jurman et al., 2008) as

$$\text{Ca}(n, i) \approx \frac{(i+1)(2n-i)}{n} \log 4 + \frac{i(1+i)}{n} + 2i - 3, \tag{3}$$

which we make use of when $i \geq 8$ and the computation of the exact value becomes intractable. For $i \geq 8$, the relative error of the approximation (3) is smaller than 1%. Our final stability indicator is thus the curve consisting of points calculated as

$$\left(i, \frac{St(\boldsymbol{S}^i)}{\text{Ca}(n, i)} \right), \tag{4}$$

for $1 \leq i \leq n$, that represent the relative change of distance between top-i lists w.r.t. the expected top-i distance.

EMPIRICAL EVALUATION OF FFA/RFA CURVE PROPERTIES

We start with the experiments on synthetic datasets. In such laboratory conditions, one has full control over the data, can establish the ground truth feature ranking, and produce rankings of different quality. Such a setting will facilitate proper assessment of our proposed feature ranking evaluation method. Before proceeding to the experiments, we briefly describe the constructed synthetic datasets. The detailed description of the datasets is given in Appendix A1.

238   We construct three datasets named `single`, `pair` and `combined`. Each of them consists of 1000
239 instances and 100 features. The relevant features in `single` dataset are individually correlated to target,
240 the relevant features in the `pair` dataset are related to the target via XOR relation, and the relevant
241 features in the `combined` dataset are the union of the relevant features in the `single` and `pair` datasets.
242 The rest of the features in the datasets are random noise.

Evaluation by Randomising the Ground Truth Ranking

244 The appropriateness of the proposed method is first demonstrated on a family of feature rankings that
245 contain more and more noise. By doing that, we can show that the lower and lower quality of feature
246 rankings is reflected in the FFA and RFA curves, and thus detected by the method.
247   We start with the ground-truth ranking $\mathbf{R}_{GT}$ and perturb it as follows. First, we select a proportion θ
248 of randomly selected features which are then assigned random relevances, drawn uniformly from the unit
249 interval. The other features preserve their ground truth relevance. This results in a ranking $\mathbf{R}_\theta$.

Experimental setup

251 We use the aforementioned `single`, `pair` and `combined` datasets. The following amounts θ of
252 noise are introduced into the ground truth ranking: $\theta \in \{0.05, 0.1, 0.15, 0.2, 0.3, 0.5, 1\}$. The value $\theta = 1$
253 corresponds to a completely random ranking.
254   For every value of θ, we estimate the expected values of the FFA/RFA curves that belong to the
255 ranking $\mathbf{R}_\theta$, by first generating $m = 100$ realizations of the ranking, and second, (point-wise) averaging of
256 the error estimates of the obtained predictive models.
257   For constructing FFA/RFA curves, SVMs were used, as noted and justified at the end of the section
258 *Analysis of Different Learning Methods to Construct FFA and RFA Curves*. The curves were constructed
259 via 10-times stratified 10-fold cross validation, using different random seeds.

Results

261 The obtained FFA and RFA curves are shown in Fig. 3 that gives the results for the dataset `combined`.
262 The results for the datasets `single` and `pair` are similar. In addition to the curves that belong to the
263 rankings $\mathbf{R}_\theta$ with different amounts of noise, ground truth ranking is also shown.
264   Both, FFA curves (Fig. 3A) and RFA curves (Fig. 3B) correctly detect different amount of noise θ:
265 the higher the θ, the more distant are FFA and RFA curve of $\mathbf{R}_\theta$ to the curves of ground truth ranking.
266 The independent confirmation of these results are given in Appendix A2.
267   Additionally, note that FFA curves cannot give all the information about the ranking: Had we not
268 plotted the RFA curves in Fig. 3B, we would not have a proof that all of the rankings misplace some
269 relevant features (check the considerable decrease in accuracy just before the 100th feature).
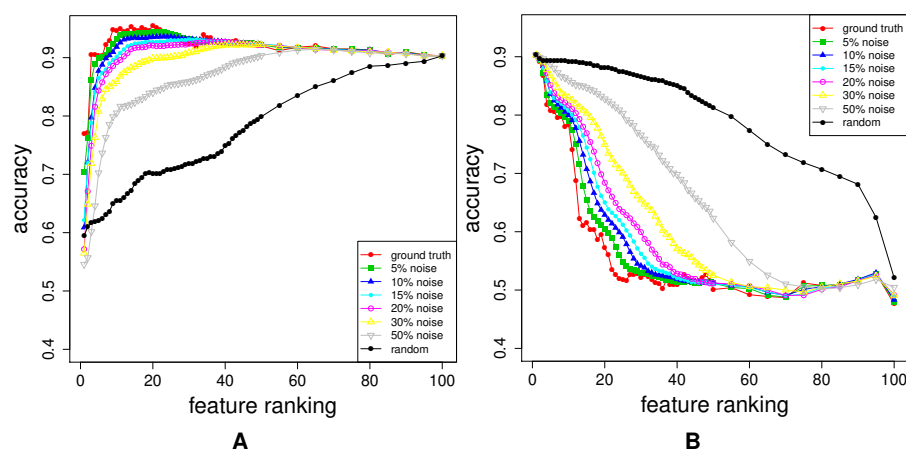
Figure 3. Dataset `combined`: Forward feature addition curves (A), and reverse feature addition curves (B). The curves for the noisy rankings $\mathbf{R}_\theta$ ($0.05 \le \theta \le 1$) and the ground truth ranking are shown.
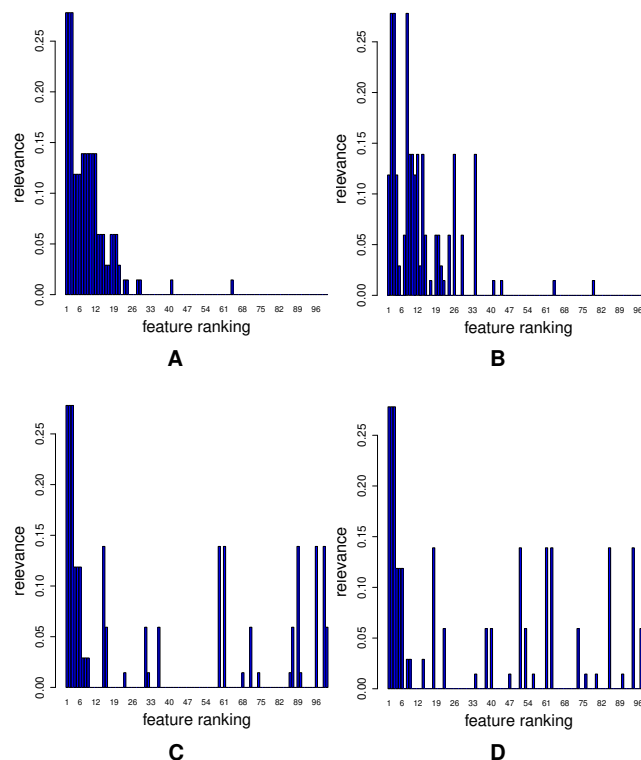
Figure 4. Distribution of relevant features for each of the four ranking methods: ReliefF (A), Random Forests (B), Info Gain (C) and SVM-RFE (D).

Analysis of Different Learning Methods to Construct FFA and RFA Curves

According to Algorithm 1, the error curve estimates depend not only on the feature ranking method, but also on the learning method used to construct the predictive models. In this section, we investigate which learning methods (learners) are suitable to construct the FFA and RFA curves. Note that we are not searching for a learner that would produce the most accurate predictive models. Rather, the requirement for the learner to be used in this context is that it should produce predictive models that exploit all the information that the features contain about the target concept, and can thus distinguish between feature rankings of different quality.

Experimental Setup

When comparing the FFA and RFA curves of different ranking methods, constructed with different learners, we used the `combined` dataset described in detail in Appendix A1. We consider the following four feature ranking methods.

Information gain, where we calculate the information gain of each feature F_i as $IG(F_t, F_i) = H(F_t) - H(F_t|F_i)$.

SVM-RFE uses a series of support vector machines (SVMs) to recursively compute a feature ranking. A linear SVM was employed, as proposed by (Guyon et al., 2002). Following (Slavkov et al., 2018), we set $\varepsilon = 10^{-12}$ and $c = 0.1$

ReliefF algorithm as proposed by Robnik-Šikonja and Kononenko (Robnik-Šikonja and Kononenko, 2003). The number of neighbors was set to 10, and the number of iterations was set to 100%.

Random Forests, which can be used for estimating feature relevance as described by Breiman (Breiman, 2001). A forest of 100 trees was used, constructed by randomly choosing rounded up $\log_2$ of the number of features.

We compare the above ranking methods by using different learners to produce classifiers and generate error estimates for the FFA and RFA curves. More specifically, we consider **Naïve Bayes** (John and Langley, 1995); **Decision Trees** (Quinlan, 1993); **Random Forests** (Breiman, 2001): the number of trees was set to 100, and in each split $\log_2$ of the number of features are considered; **SVMs** (Cortes and Vapnik,

296 1995): a polynomial (quadratic) kernel was employed, with the $\varepsilon = 10^{-12}$ and $c = 0.1$; **k-NN** (Aha and
297 Kibler, 1991) with a value of $k = 10$.

298   The curves were constructed via 10-times stratified 10-fold cross validation, using different random
299 seeds. The obtained FFA and RFA curve comparisons of the four feature ranking methods obtained by
300 each of the five learning methods, are presented in the following section.

301 ### *Results*

302 The rankings are shown in Fig. 4, where each graph represents the distribution of the ground truth
303 relevance values. The y-axis depicts the ground truth relevance value (5). Each point, i, represents the i-th
304 ranked feature, $F_{(i)}$, as determined by the feature ranking method.

305   We can see that the rankings fall into two groups: in Figs. 4A and 4B, highly relevant features are
306 concentrated on the left, while in the Figs. 4C and 4D, they are evenly spread.

307   ReliefF and Random Forests (Figs. 4A and 4B) are thus clearly better than Info Gain and SVM-RFE
308 (Figs. 4C and 4D). Hence, the FFA and RFA curves should at least differentiate between the two groups
309 of the rankings. However, there should be visible difference also between Relief and Random Forests
310 at the beginning of the ranking. The detailed comparative evaluation of the obtained feature rankings is
311 given in Appendix A3.

312   In the case of FFA curves, the learners can be divided into two groups: FFA curves produced by Naïve
313 Bayes, Decision Trees and Random Forests cannot capture any difference between rankings, whereas
314 those produced by SVMs and k-NN can. It suffices to show one representative graph for each group
315 (those for Naïve Bayes and k-NN are shown in Fig. 5), since there are no significant differences among
316 the learning methods in the same group[1]. The FFA curves produced by these two learners have all the
317 desirable properties: the curves for Relief and Random Forests are better than those of Info Gain and
318 SVM-RFE. Additionally, at the beginning, the Random Forest curve is under the curve of Relief.

319   The reason why, for example, the Naïve Bayes classifier does not show any difference between
320 rankings, is the fact that it can not use the information from the interactions of higher order. Namely, it
321 assumes feature independence. Hence, it is not appropriate for use in the considered context.
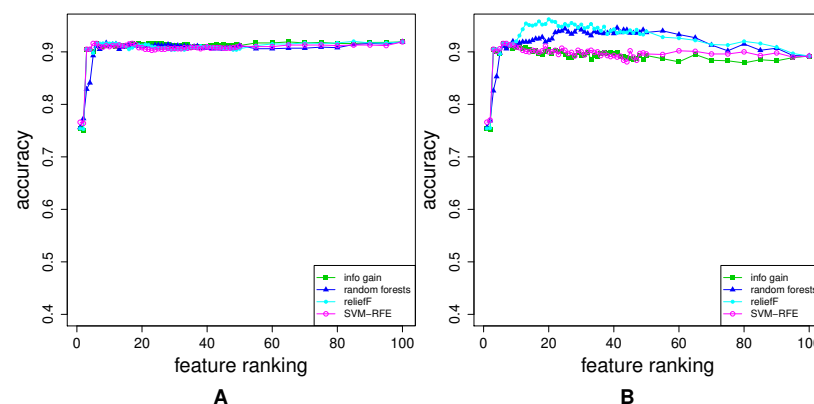
Figure 5. Comparison of **FFA** curves for the four different ranking methods for the `combined` dataset.
The curves were obtained by using the Naïve Bayes (A) and k-NN (B).

322   If we proceed to RFA curves, again, the Naïve Bayes classifier does not show any difference between
323 rankings, whereas the other four methods do. We prefer Random Forests, SVMs and k-NN over Decision
324 Trees in the case of RFA curves, because Decision Trees generate quite unstable curves, as shown in Fig.
325 6A. In Fig. 6B, the RFA curves of k-NN are shown. Again, there is no quantitative difference between
326 them and the RFA curves generated by SVMs[2].

327   Tu sum up, one can use

328 • SVMs and k-NN models, for constructing FFA curves,

---

[1] Compare, for example, Fig. 5B (obtained by k-NN) with Fig. A1A (obtained by SVMs), which is given in Appendix A3 (note
that Fig. A1A also contains the random ranking curve).

[2] Compare Fig. 6B to Fig. A1B (given in Appendix A3).
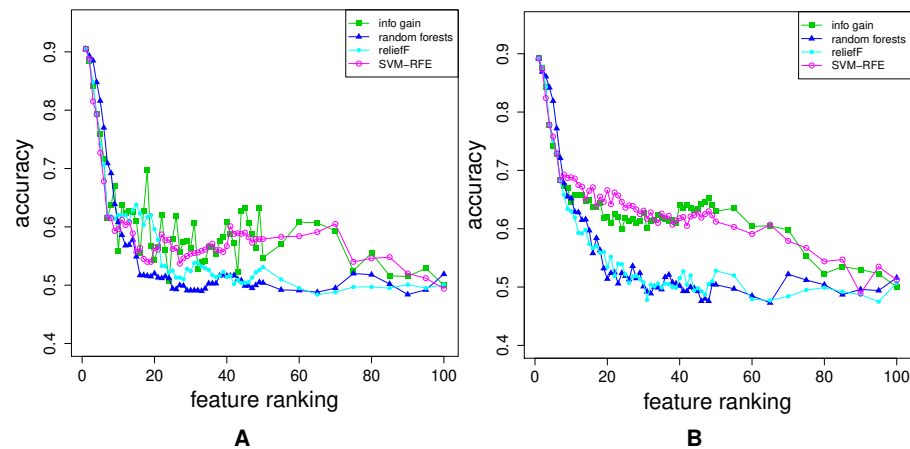
Figure 6. Comparison of RFA curves for the four different ranking methods for the `combined` dataset. The curves were obtained by using Decision Trees (A) and k-NN (B).

329   • SVMs, k-NN and Random Forests, for constructing RFA curves.

330 Thus, only k-NN and SVMs are appropriate for constructing both FFA and RFA curves. Since one should
331 typically use approximate k-NN when the number of features is extremely high (Muja and Lowe, 2009),
332 we use SVMs (with the settings described here) as the learner for constructing the FFA/RFA curves in all
333 the remaining experiments in this work.

Discussion

335 We have to give some additional notes about choosing the best method, when, for example, different
336 learning methods prioritize different rankings, which is possible since some learning method might make
337 use of some features, whereas another learning method can make better use of some others.

338   If we have computed feature rankings to learn a classifier that uses only a subset of (top-ranked)
339 features and we have already decided on which classifier to use, we should use the same (type of) classifier
340 to construct the curves, because we want to use the features that the chosen learning method prioritizes.

341   Second, if our motivation for computing the feature rankings is to discover all relevant features for a
342 given problem (e.g., the genes that influence the patients' clinical state), and learning method A prioritizes
343 the ranking $\mathbf{R} = (x_1, x_2, \dots)$ over the ranking $\mathbf{R}' = (x_1', x_2', \dots)$, whereas learning method B prioritizes
344 $\mathbf{R}'$ over $\mathbf{R}$, this means that x_1, x_2, x_1' and x_2' are important (provided that both learners achieve similar
345 accuracy), so we can include them all in the subsequent experiments (and thus use both learning methods).

346   The decision about which among the two appropriate methods to use – k-NN or SVMs – might also
347 depend on the properties of the dataset at hand. As mentioned before, k-NN could be too time-consuming
348 when the number of features is extremely high. On the other hand, if the number of instances is high,
349 SVMs could be too time-consuming, but speed-ups are possible (Tsang et al., 2005). As for the noise,
350 both methods are quite robust (Wang et al., 2018; Xu et al., 2009), so this is not among the most influential
351 factors.

FEATURE RANKING COMPARISON ON REAL-WORLDS DATASETS

353 In this section, we move from the synthetic data and show the appropriateness of the proposed feature
354 ranking evaluation method on the real-world data with unknown relevant and irrelevant features. To be
355 consistent with the synthetic-data experiments, we evaluate the same four feature ranking methods as
356 before, and compare them to the random feature rankings which now serve as the only baseline.

Datasets Description

358 In this extensive study, 35 classification benchmark problems are used. They come in two bigger groups.
359 The first group has been part of the experiments in (Slavkov et al., 2018) and except for `aapc` (Džeroski
360 et al., 1997), `water` and `diversity` (Džeroski et al., 2000), mostly originates from the UCI data
361 repository (Newman and Merz, 1998). This benchmark problems have higher number of instances (up

362 to 5000) and not extremely high number of features (up to 280). This problems cover various domains:
363 health, ecology, banking etc.

364 The second group is newly included and contains 11 high-dimensional micro-array benchmark
365 problems(Mramor et al., 2007) (up to 12625 features) with lower number of examples (up to 110). The
366 main properties of the data are given in Tab. 1.

Table 1. Properties of the benchmark datasets: number of instances, number of features, number of discrete/numeric attributes, and number of different class values. The datasets with a considerably high number of features are listed under the dashed line.

Dataset	#Inst.	#Feat.	(D/N)	#Cl.
aapc	335	84	(83/1)	3
arrhythmia	452	280	(73/206)	16
australian	690	14	(8/6)	2
balance	625	4	(0/4)	3
breast-cancer	286	9	(9/0)	2
breast-w	699	9	(9/0)	2
car	1728	6	(6/0)	4
chess	3196	36	(36/0)	2
diabetes	768	8	(0/8)	2
diversity	292	86	(0/86)	5
german	1000	20	(13/7)	2
heart	270	13	(6/7)	2
heart-c	303	13	(7/6)	5
heart-h	294	13	(7/6)	5
hepatitis	155	19	(13/6)	2
image	2310	19	(0/19)	7
ionosphere	351	34	(0/34)	2
iris	150	4	(0/4)	3
sonar	208	60	(0/60)	2
tic-tac-toe	958	9	(9/0)	2
vote	435	16	(16/0)	2
water	292	80	(0/80)	5
waveform	5000	21	(0/21)	3
wine	178	13	(0/13)	3
amlPrognosis	54	12625	(0/12625)	2
bladderCancer	40	5724	(0/5724)	3
breastCancer	24	12625	(0/12625)	2
childhoodAll	110	8280	(0/8280)	2
cmlTreatment	28	12625	(0/12625)	2
colon	62	2000	(0/2000)	2
dlbcl	77	7070	(0/7070)	2
leukemia	72	5147	(0/5147)	2
mll	72	12533	(0/12533)	3
prostate	102	12533	(0/12533)	2
srbct	83	2308	(0/2308)	4

Experimental Setup

368 We construct the curves that base on the feature ranking methods described in experimental setup part
369 of Sec. *Analysis of Different Learning Methods to Construct FFA and RFA Curves*, and the curves that
370 belong to the completely random ranking (i.e., expected curves) which serve as a baseline. For the actual
371 construction of the curves (once the ranking is obtained), support vector machines were used, as described
372 and justified in the results in Section *Analysis of Different Learning Methods to Construct FFA and RFA*
373 *Curves*. The curves were constructed via 10-times stratified 10-fold cross validation, using different
374 random seeds.

375 The expected error curves for random rankings were produced by generating 100 random rankings for
376 each dataset under consideration. For each random ranking, error curves were produced and the average
377 of the error values was used as the expected error. This was done in a manner similar to the one described
378 in Sec. *Evaluation by Randomising the Ground Truth Ranking*. As mentioned in Sec. *The Evaluation*
379 *Method*, building FFA/RFA curves by adding the features one by one to large feature subsets S^i and S_i,
380 might be too costly when n is big enough. In this set of experiments, we use the following procedure.
381 We add $\delta(i)$ features to the subset, where $\delta(i)$ is defined as follows: $\delta(i) = 1$ if $1 \le i \le 50$, $\delta(i) = 5$ if
382 $50 < i \le 500$, and $\delta(i) = n//20$ otherwise, where $//$ denotes integer division.
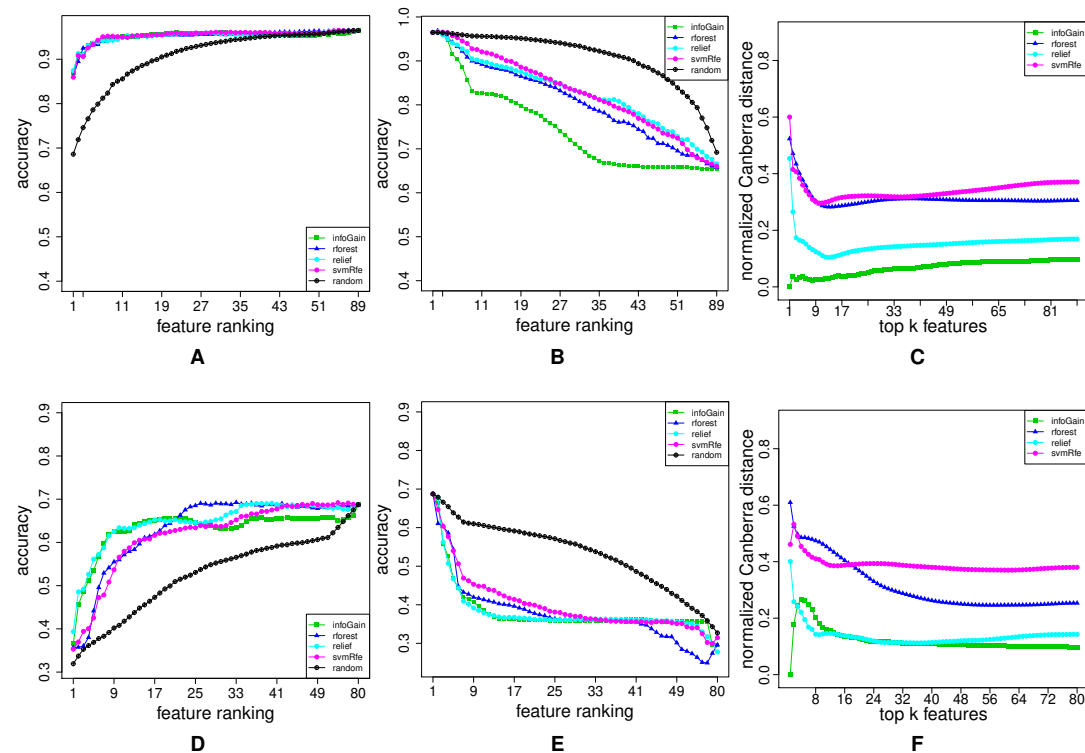
Figure 7. Ranking quality assessment for datasets `breast-w` (first row) and `water` (second row) in terms of the FFA (first column) and RFA curves (second column), and rankings' stability estimates (third column). The FFA/RFA curves are obtained by using SVMs.

Results

In this section, we show representative examples of three types of curves: FFA, RFA and stability curves. The curves are shown for two datasets with lower and two with higher number of features. The graphs for the other datasets can be found in Appendix A4 in Figs. A2, A3, A4, A5, A6, A7, A8, A9, A10, A11, and A12.

We start with the `breast-w` dataset. The FFA/RFA curves in Figs. 7A and 7B show that both types of curves are needed in order to evaluate the ranking completely. The FFA-curves suggest that all feature ranking algorithms (except for the random ranking) place only the relevant features at the beginning, since there is practically no difference if we compare the accuracy of the 89-feature (all) model and, e.g., the 11-feature model. However, the RFA-curves show that all feature ranking algorithms - except for Info Gain - misplace some relevant features, since the Info-Gain-ranking-based models have the lowest accuracy by far in the RFA curves. Also, in the case of Info Gain, the accuracy cease to decrease after only cca. 40 top-ranked features were removed.

Fig. 7C shows that Info Gain produces also the most stable rankings. We can see that the top-ranked feature is always the same, since the stability index of the Info Gain equals 0 at the point $k = 1$. The second most stable algorithm is ReliefF, the third is Random Forest and the least stable is SVM-RFE, but the difference between Random Forest and SVM-RFE is not considerable.

Let us now take a look at the curves for the dataset `water`. From the FFA-curves in Fig. 7D, we see that ReliefF, Info Gain and Random Forest ought to have the same 21 top-ranked features, and as a consequence, the same last $59 = 80 - 21$ features too. However, the first 21 features are ordered better by Info Gain and ReliefF, while the last 59 are more properly ordered by Random Forest. We can conclude that none of the rankings is ideal, but we can come close to the ideal one (in terms of FFA-curves), if we combine the first part of the ReliefF (or Info Gain) and the second part of Random Forest. This claim is also confirmed by the RFA-curves of Info Gain and ReliefF (Fig. 7E): these two algorithms indeed misplace some relevant features, since the accuracy of the model abruptly decreases and the end of the ranking.
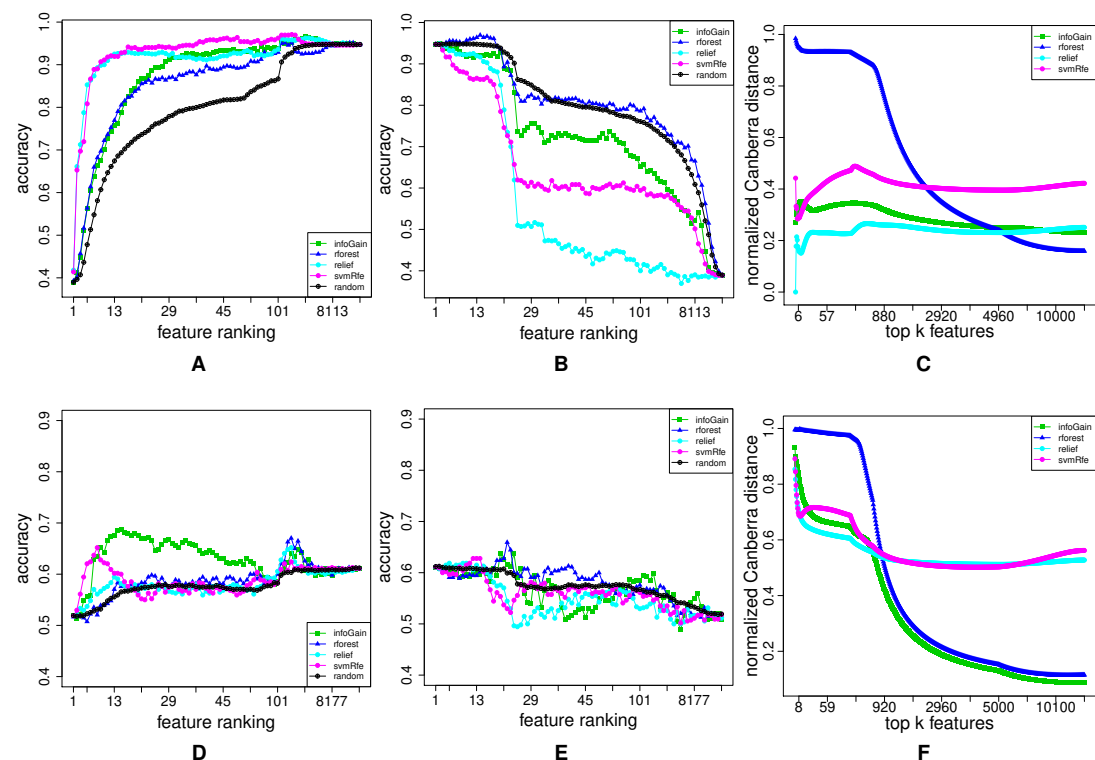
Figure 8. Ranking quality assessment for datasets `mll` (first row) and `amlPrognosis` (second row) in terms of the FFA (first column) and RFA curves (second column), and rankings' stability estimates (third column). The FFA/RFA curves are obtained by using SVMs.

⁴⁰⁹   Fig. 7F suggests that we should prefer Info Gain and ReliefF over Random Forest since they are more
⁴¹⁰ stable. However, we can also notice that Random Forest is the least stable at the beginning of the ranking
⁴¹¹ but its stability increases when the number of features gets larger.

⁴¹²   We begin the analysis of the high-dimensional datasets with `mll`. Fig. 8B shows that Random Forest
⁴¹³ completely misplaces some relevant features, since its RFA-curve mostly goes above the random-ranking
⁴¹⁴ one. Even though it is evident from Random Forest's FFA-curves that some relevant features are also
⁴¹⁵ successfully captured, Random Forest produces the worst ranking. Info Gain is slightly better, whereas
⁴¹⁶ ReliefF and SVM-RFE are again the best algorithms. From the FFA-curves, we can conclude that SVM-
⁴¹⁷ RFE places more features with higher relevance at the beginning of the ranking (its curve is higher than
⁴¹⁸ ReliefF's), while RFA-curves reveal that SVM-RFE also misses some quite relevant features: ReliefF's
⁴¹⁹ curve is far below SVM-RFE's. Fig. 8C shows that ReliefF is considerably more stable than SVM-RFE,
⁴²⁰ hence we prefer the former over the latter on the `mll` dataset.

⁴²¹   The last example will show that sometimes, the understanding of the results is not that easy. In the
⁴²² Figs. 8D and 8E, the FFA and RFA curves for the `amlPrognosis` dataset are presented. In this case,
⁴²³ only Info Gain performs considerably better than random ranking in terms of FFA-curves. SVM-RFE is
⁴²⁴ also able to find some relevant features at the beginning (peak of its curve at 10 features), but after that,
⁴²⁵ the models' accuracy decreases, hence mostly noisy features are positioned here. Some relevant features
⁴²⁶ are again placed by ReliefF, Info Gain and Random Forest also around the 2000th place (local peak of
⁴²⁷ their curves in the right part of the FFA-curve). RFA-curves confirm that there is indeed much noise in
⁴²⁸ these data, since removing features does not result in an (at least approximately) decreasing curve.

⁴²⁹   It may not come as a surprise that all ranking algorithms produce rankings that are very unstable at the
⁴³⁰ beginning (Fig. 8F), but it is interesting that after approximately 1000 features, Info Gain and Random
⁴³¹ Forest produce quite stable rankings even though they have low quality. The reason for both low quality
⁴³² of rankings and their instability is probably the low number of instances accompanied by a high number
⁴³³ of features (54 and 12625 respectively).

CONCLUSIONS

We have proposed a method for evaluating and comparing feature rankings. The method is based on constructing two chains of predictive models that are built on two families of nested subsets of features. The first family of subsets are the sets of top-ranked features, while the second family consists of sets of bottom-ranked features. The predictive performances of the models form a forward feature addition (FFA) curve in the former case, and reverse feature addition (RFA) curve in the latter case.

We show in our experiments that both types of curves are necessary when comparing the rankings: FFA curves detect whether important features are placed at the beginning of the ranking, whereas RFA curves detect whether important features can still be found at the end of the ranking.

In the set of experiments, we show the usefulness of the proposed evaluation method and its sensitivity to the rankings of different quality on synthetic data. The second set of experiments shows which of the learning methods are appropriate for building the FFA and RFA curves (SVMs, k-NN) and which are not (Naïve Bayes, Decision Tree, Random Forest). In the third set of experiments on synthetic data, we test several feature ranking algorithms and examine their properties. Considering data with different properties, we show that ReliefF algorithm outperforms the other investigated approaches, both in terms of detecting relevant features and in terms of stability of the feature rankings it produces.

Moreover, we show the usefulness of the proposed approach in real-world scenarios. We evaluate feature rankings computed by four feature ranking algorithms on 35 classification benchmark problems. The results reveal that there is no feature ranking algorithm that would dominate the others on every dataset.

A possible disadvantage of the proposed method is that it can be computationally quite intensive, if we want to construct the curves in full resolution. Namely, every point of a FFA or RFA curve comes at the cost of building and evaluating a predictive model. However, as justified in the method description, the full resolution is, especially when the number of features is really high, not necessary, and, moreover, the construction of the curve can be also easily parallelized.

The work presented in this paper can continue in many directions. First, of all, the proposed methodology could use other error measures, since accuracy is appropriate only for the task of classification when the distribution of target variable is approximately uniform. The strong modularity of the FFA/RFA curves allows for their use in any other predictive modeling task, e.g., for the task of regression, we could use root mean squared error instead of accuracy. However, even though there exists a regression version of most of the learners, which are considered for constructing the curves, experiments should be repeated on those cases, since the conclusions about, for example, the most appropriate learner for constructing the curves, can be different. Moreover, the method can be adapted not only to the regression setting, but also to different tasks of structured output prediction (Bakır et al., 2007) and time series prediction.

REFERENCES

Aha, D. and Kibler, D. (1991). Instance-based learning algorithms. *Machine Learning*, 6:37–66.

Arceo-Vilas, A., Fernandez-Lozano, C., Pita, S., Pértega-Díaz, S., and Pazos, A. (2020). A redundancy-removing feature selection algorithm for nominal data. *PeerJ Computer Science*.

Bakır, G. H., Hofmann, T., Schölkopf, B., Smola, A. J., Taskar, B., and Vishwanathan, S. V. N., editors (2007). *Predicting Structured Data*. The MIT Press, Cambridge, Massachusetts.

Biesiada, J., Duch, W., Kachel, A., and Paucha, S. (2005). Feature ranking methods based on information entropy with parzen windows. In *In International Conference on Research in Electrotechnology and Applied Informatics*, Katowice, Poland.

Boucheham, A. and Batouche, M. (2014). Robust biomarker discovery for cancer diagnosis based on meta-ensemble feature selection. In *2014 Science and Information Conference*, pages 452–560.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Duch, W., Wieczorek, T., and Biesiada, J. (2004). Comparison of feature ranking methods based on information entropy. In Fawcett, T. and Mishra, N., editors, *IEEE International Conference on Neural Networks - Conference Proceedings*, volume 2, pages 1415–1419. IEEE.

Džeroski, S., Demšar, D., and Grbović, J. (2000). Predicting chemical parameters of river water quality from bioindicator data. *Applied Intelligence*, 13:7–17.

Džeroski, S., Potamias, G., Moustakis, V., and Charissis, G. (1997). Automated revision of expert rules

for treating acute abdominal pain in children. In *Artificial intelligence in medecine - AIME, LNCS 1211*, pages 98–109.

Furlanello, C., Serafini, M., Merler, S., and Jurman, G. (2003). Entropy-based gene ranking without selection bias for the predictive classification of microarray data. *BMC Bioinformatics*, 4:54.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, 46:389–422.

Guzmán-Martínez, R. and Alaiz-Rodríguez, R. (2011). Feature selection stability assessment based on the Jensen-Shannon divergence. *Lecture Notes in Computer Science*, 6911:597–612.

Henzgen, S. and Hüllermeier, E. (2015). Weighted rank correlation: A flexible approach based on fuzzy order relations. In Appice, A., Rodrigues, P. P., Santos Costa, V., Gama, J., Jorge, A., and Soares, C., editors, *Machine Learning and Knowledge Discovery in Databases*, pages 422–437. Springer International Publishing.

John, G. H. and Langley, P. (1995). Estimating continuous distributions in Bayesian classifiers. In *proc. Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Mateo, CA. Morgan Kaufmann.

Jong, K., Mary, J., Cornuéjols, A., Marchiori, E., and Sebag, M. (2004). Ensemble feature ranking. In *PKDD - LNCS 2302*, pages 267–278.

Jurman, G., Merler, S., Barla, A., Paoli, S., Galea, A., and Furlanello, C. (2008). Algebraic stability indicators for ranked lists in molecular profiling. *Bioinformatics*, 24:258–264.

Kalousis, A., Prados, J., and Hilario, M. (2007). Stability of feature selection algorithms: A study on high-dimensional spaces. *Knowledge and Information Systems*, 12:95–116.

Khoshgoftaar, T. M., Fazelpour, A., Wang, H., and Wald, R. (2013). A survey of stability analysis of feature subset selection techniques. In *IEEE 14th International Conference on Information Reuse Integration (IRI)*, pages 424–431.

Lance, G. N. and Williams, W. T. (1966). Computer programs for hierarchical polythetic classification ('similarity analyses'). *The Computer Journal*, 9:60–64.

Lance, G. N. and Williams, W. T. (1967). Mixed-data classificatory programs i - agglomerative systems. *Australian Computer Journal*, 1.

Li, Z. and Gu, W. (2015). A redundancy-removing feature selection algorithm for nominal data. *PeerJ Computer Science*.

Liang, J., Yang, S., and Winstanley, A. (2008). Invariant optimal feature selection: A distance discriminant and feature ranking based solution. *Pattern Recognition*, 41:1429–1439.

Liu, T., Liu, S., Chen, Z., and Ma, W.-Y. (2003). An evaluation on feature selection for text clustering. In Fawcett, T. and Mishra, N., editors, *ICML*, pages 488–495. The AAAI Press, Menlo Park, California.

Mramor, M., Leban, G., Demšar, J., and Zupan, B. (2007). Visualization-based cancer microarray data classification analysis. *Bioinformatics*, 23(16):2147.

Muja, M. and Lowe, D. G. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. In Ranchordas, A. and Araújo, H., editors, *VISAPP (1)*, pages 331–340. INSTICC Press.

Nardone, D., Ciaramella, A., and Staiano, A. (2019). A redundancy-removing feature selection algorithm for nominal data. *PeerJ Computer Science*.

Newman, C. B. D. and Merz, C. (1998). UCI repository of machine learning databases. https://archive.ics.uci.edu/ml/datasets.html. Accessed on: 2015-12-13.

Nilsson, R., Peña, J. M., Björkegren, J., and Tegnér, J. (2007). Consistent feature selection for pattern recognition in polynomial time. *Journal of Machine Learning Research*, 8:589–612.

Nogueira, S., Sechidis, K., and Brown, G. (2017). On the stability of feature selection algorithms. *Journal of Machine Learning Research*, 18(1):6345–6398.

Paoli, S., Jurman, G., Albanese, D., Merler, S., and Furlanello, C. (2005). Semisupervised profiling of gene expressions and clinical data. In *Proc. Sixth International Conference on Fuzzy Logic and Applications*, pages 284–289.

Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA.

Robnik-Šikonja, M. and Kononenko, I. (2003). Theoretical and empirical analysis of ReliefF and RReliefF. *Machine Learning*, 53:23–69.

Saeys, Y., Abeel, T., and de Peer, Y. V. (2008). Robust feature selection using ensemble feature selection techniques. In *ECML/PKDD, LNCS 5212*, pages 313–325.

Slavkov, I., Petković, M., Kocev, D., and Džeroski, S. (2018). Quantitative score for assessing the quality of feature rankings. *Informatica*, 42(1):43–52.

Tsang, I. W., Kwok, J. T., and Cheung, P.-M. (2005). Core vector machines: Fast svm training on very large data sets. *Journal of Machine Learning Research*, 6:363–392.

Verikas, A., Gelzinis, A., and Bacauskiene, M. (2011). Mining data with random forests: A survey and results of new tests. *Pattern Recognition*, 44:330–349.

Wang, Y., Jha, S., and Chaudhuri, K. (2018). Analyzing the robustness of nearest neighbors to adversarial examples. In Dy, J. G. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 5120–5129. PMLR.

Xu, H., Caramanis, C., and Mannor, S. (2009). Robustness and regularization of support vector machines. *Journal of Machine Learning Research*, 10:1485–1510.

A1

In this section, we explain how we generate our synthetic datasets. For simplicity, we take both the features F_i and the target F_t to be binary and take values from the set $\{0,1\}$. We then partition the set of features $\mathscr{F}$ into feature interaction subsets $\mathscr{F}_{\text{int}}$ of cardinality one and two. The feature sets with cardinality one are single features F_i that are in an individual interaction with the target F_t, while the features from the interaction sets with cardinality two determine the value of the target by the *XOR* relation.

The examples are generated as follows. For each example, we first randomly (from a uniform distribution) set the value of the target feature F_t. After that, if $\mathscr{F}_{\text{int}} = \{F_i\}$, then the value of feature F_i is randomly chosen, so that $P(F_i = F_t) = p$. Otherwise, we have $\mathscr{F}_{\text{int}} = \{F_i, F_j\}$, and the values of the features F_i and F_j are randomly chosen, so that $P(\text{XOR}(F_i, F_j) = F_t) = p$.

We consider the probability values $p \in \{0.8, 0.7, 0.6, 0.5\}$. The feature sets with $p = 0.5$ are in fact independent of the target F_t, and can be considered as irrelevant features.

With combinations of these feature interaction sets, three datasets were generated, each of them consisting of 1000 instances and 100 features in total.

The first dataset (named `single`) comprises only individually correlated features. The second dataset (named `pair`) contains relevant features related to the target via the XOR relation, as well as irrelevant features. The third (named `combined`) is a combination of the first two. It contains XOR-related features and individually correlated features.

In order to simulate the redundancy of features, which occurs in real datasets, the three datasets are constructed in the following way: If the set $\mathscr{F}_{\text{int}}$ of relevant features is included in the dataset, we also include two redundant copies of $\mathscr{F}_{\text{int}}$ in the dataset. Irrelevant features are realized independently of each other.

The properties of the generated datasets are summarized in Table 2, from which we can observe that there are 9 relevant features in the `single` dataset, 18 in the `pair` dataset, and 27 in the `combined` dataset.

Table 2. Properties of the synthetic datasets. If $p > 0.5$, #copies denotes the number of copies of the interaction set. In the last row where $p = 0.5$, #copies corresponds to the number of independently realised irrelevant features.

$\|\mathscr{F}_{\text{int}}\|$	p	# copies in `single`	# copies in `pair`	# copies in `combined`
	0.8	3		3
1	0.7	3		3
	0.6	3		3
	0.8		3	3
2	0.7		3	3
	0.6		3	3
1	0.5	91	82	73

The ground-truth feature relevances $\text{rel}(F_i)$ of the features F_i are defined as follows. First, the relevance of each feature group $\mathscr{F}_{\text{int}}$ is defined as the mutual information between the group and F_t, namely $\text{rel}(\mathscr{F}_{\text{int}}) = I(\mathscr{F}_{\text{int}}; F_t)$. Second, for $F_i \in \mathscr{F}_{\text{int}}$, feature importances are obtained as

$$\text{rel}(F_i) = \text{rel}(\mathscr{F}_{\text{int}}) / |\mathscr{F}_{\text{int}}|. \tag{5}$$

For the particular three datasets, this ground-truth ranking R_{GT} should also result in the optimal FFA and RFA curves, but this may not be the case in general. In the next section, we give the results of comparing it to the other feature rankings.

A2

When discussing the results in the subsection *Evaluation by Randomising the Ground Truth Ranking*, we showed that, when the level of noise θ in the ranking increases, then i) the quality of the ranking $\mathbf{R}_\theta$ decreases, and ii) the rankings $\mathbf{R}_{GT}$ and $\mathbf{R}_\theta$ become more and more distant. However, for the second point, we need to define a distance $\text{dist}(\mathbf{R}_{GT}, \mathbf{R}_\theta)$ between a noisy and the ground truth ranking. In the definition of $\text{dist}(\mathbf{R}_{GT}, \mathbf{R}_\theta)$ we use the average Spearman rank correlation coefficient $\rho(\mathbf{R}_A, \mathbf{R}_B)$ which is calculated as

$$\frac{1}{n-1} \sum_{i=1}^{n} \frac{(rank_A(F_i) - \overline{rank_A})(rank_B(F_i) - \overline{rank_B})}{\sigma_A \sigma_B}$$

where n is the number of features, therefore the average ranks $\overline{rank_A}$ and $\overline{rank_B}$ equal $(n+1)/2$. Standard variations $\sigma_{A,B}$ are computed as $\sigma_{A,B} = \sqrt{\sum_{i=1}^{n}(rank_{A,B}(F_i) - \overline{rank_{A,B}})^2 / (n-1)}$. For a given θ, the distance between rankings $\mathbf{R}_{GT}$ and $\mathbf{R}_\theta$ is then computed as

$$\text{dist}(\mathbf{R}_{GT}, \mathbf{R}_\theta) = 1 - \frac{1}{m} \sum_{t=1}^{m} \rho(\mathbf{R}_{GT}, \mathbf{R}_{\theta,t}) \tag{6}$$

where m is the number of different realizations $\mathbf{R}_{\theta,t}$ of the noisy ranking $\mathbf{R}_\theta$.

Tab. 3 lists the values of the distances between the ground truth ranking $\mathbf{R}_{GT}$ and its noisy versions $\mathbf{R}_\theta$. Note that, for all three synthetic datasets, the distances indeed increase when the amount of noise θ increases.

Table 3. The distances $\text{dist}(\mathbf{R}_{GT}, \mathbf{R}_\theta)$ for different θ values, for each of the three synthetic datasets.

θ	0.05	0.1	0.15	0.2	0.3	0.5	1
single	0.219	0.34	0.449	0.51	0.628	0.81	1.056
pair	0.126	0.239	0.327	0.397	0.519	0.726	1.091
combined	0.1	0.171	0.252	0.32	0.432	0.652	1.048

A3

In Sec. *Evaluation by Randomising the Ground Truth Ranking*, we analyzed rankings of different quality (with different amounts of noise) by comparing them to the ground truth ranking. In a real-world setting, the ground truth ranking is unknown and the feature rankings are induced directly from data. Therefore, in this section we analyze feature rankings, produced by the four feature ranking methods from Sec. *Analysis of Different Learning Methods to Construct FFA and RFA Curves*, and the synthetic data described in Sec. *Generating Synthetic Data*.

hen comparing the rankings, stability should also be taken into account as discussed earlier. Therefore, the stability indicator (4) is also included in the analysis.

Experimental Setup
We have used the same parameter settings for the feature ranking algorithms as in Sec. *Analysis of Different Learning Methods to Construct FFA and RFA Curves*. As noted and justified in the corresponding Sec. *Results*, SVMs were used for constructing the FFA/RFA curves. The curves were constructed via 10-times stratified 10-fold cross validation, using different random seeds. To complement them, we also estimate the stability of each feature ranking algorithm by using the stability indicator described in Sec. *Stability of feature ranking*. All feature ranking methods were tested only on the `combined` dataset.
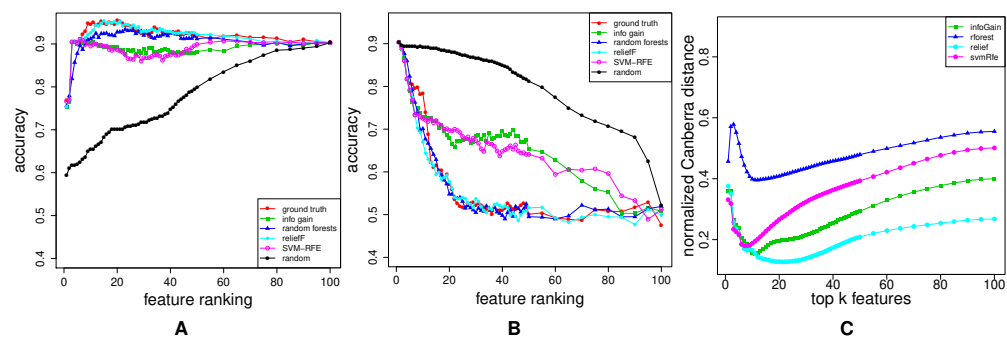
Figure A1. Ranking quality assessment for dataset `combined` in terms of the FFA (A) and RFA curves (B), and rankings' stability estimates (C). The FFA/RFA curves are obtained by using SVMs.

Results

For our analysis, we consider three types of graphs. The first two types are FFA curves (Fig. A1A) and RFA curves (Fig. A1B). The third is the stability estimate graph (Fig. A1C) where the y-axis refers to the value of the stability indicator (4): the higher the value, the less stable the ranking method. Each point, k, on the x-axis represents the size of the considered feature subsets, consisting of the top ranked k features.

Upon a visual inspection of the overall results in Fig. 4, we can conclude that all of the feature ranking methods can correctly detect the features individually related to the target. However, Info Gain and SVM-RFE (Figs. 4C and 4D, respectively) exhibit random behavior for the XOR features, i.e., are unable to assign proper relevance values to them. Random Forests (Fig. 4B) separate relevant from irrelevant features, but the ordering of the relevant features is mixed. Finally, ReliefF (Fig. 4A) provides the ranking that is the closest to the ground truth.

These differences in behavior among the different ranking methods are also clearly reflected in the FFA and RFA curves in Figs. A1A and A1B. In Fig. A1B, the RFA curves for Info Gain and SVM-RFE have a similar behavior: Namely, a linearly increasing accuracy (from right to left) in the region where the relevant features are randomly distributed and a sharp increase in accuracy in the region where the individually relevant features are included. On the other hand, the RFA curves of both random forests and ReliefF remain first constant and then increase abruptly when the top-ranked features are included. These two groups of methods can be also distinguished from the FFA curves. The FFA curves of all methods are first increasing abruptly and then slightly decreasing but the FFA curves of ReliefF and random forests increase during more steps and reach higher accuracy than Info Gain and SVM-RFE. This clearly indicates that while Info Gain and SVM manage to identify a proportion of the relevant features and put them at the top of the ranking, this proportion is nevertheless smaller than the one identified by random forests and ReliefF.

FFA and RFA curves undoubtedly allow us to compare the quality of the different ranking algorithms. The FFA/RFA curves of all methods are clearly better than the curves of the random ranking. The ReliefF ranking algorithm, however, clearly outperforms the other methods. It has the best error curves, i.e., the curves that are the closest to the ground truth ranking. The second best method are random forests: they exhibit very similar performances, but show a slightly worse FFA curve. Both Info Gain and SVM-RFE are clearly inferior in terms of both FFA and RFA curves.

Stability-wise, as seen in Fig. A1C, all of the algorithms are stable in the region of the relevant features that they can detect, except for random forests, which has an instability peak exactly in this region. This means that random forests are in this case capable of detecting all the relevant features, but are highly unstable in the estimation of their ordering. Further inspection reveals that Relief generates not only the best rankings in terms of FFA/RFA curves, but also the most stable ones.

A4

In this section, we provide detailed per dataset results from the experimental study.
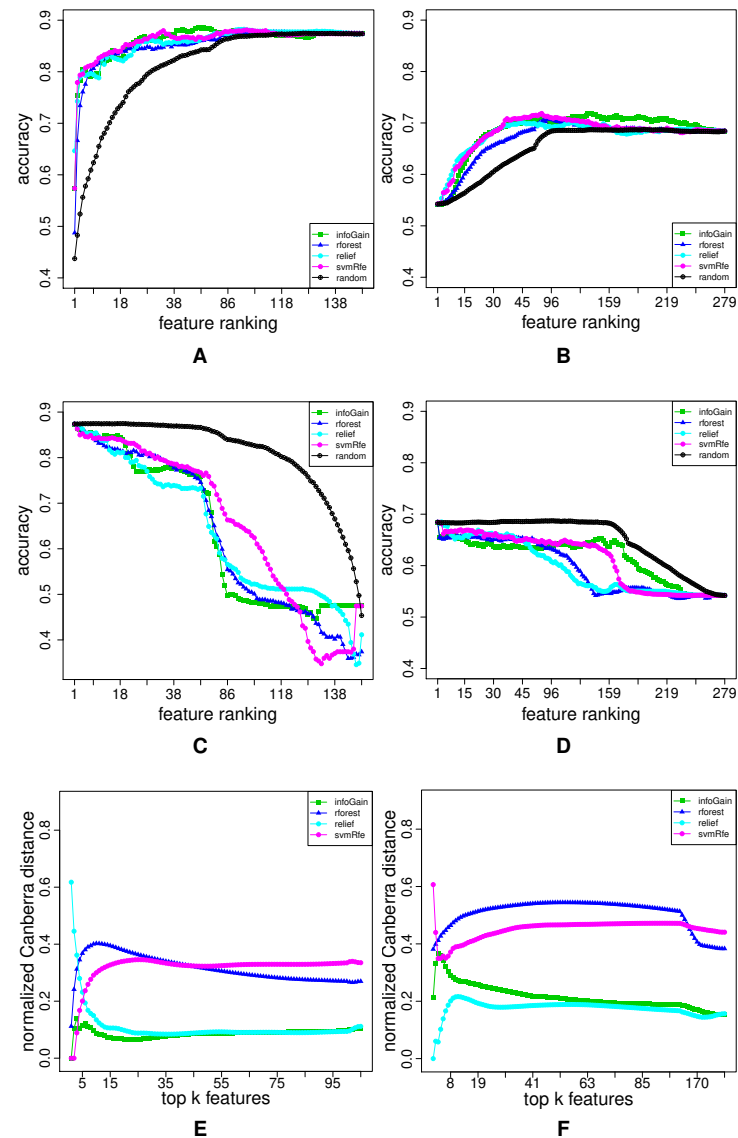
Figure A2. Ranking quality assessment for datasets `aapc` (first column) and `arrhythmia` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
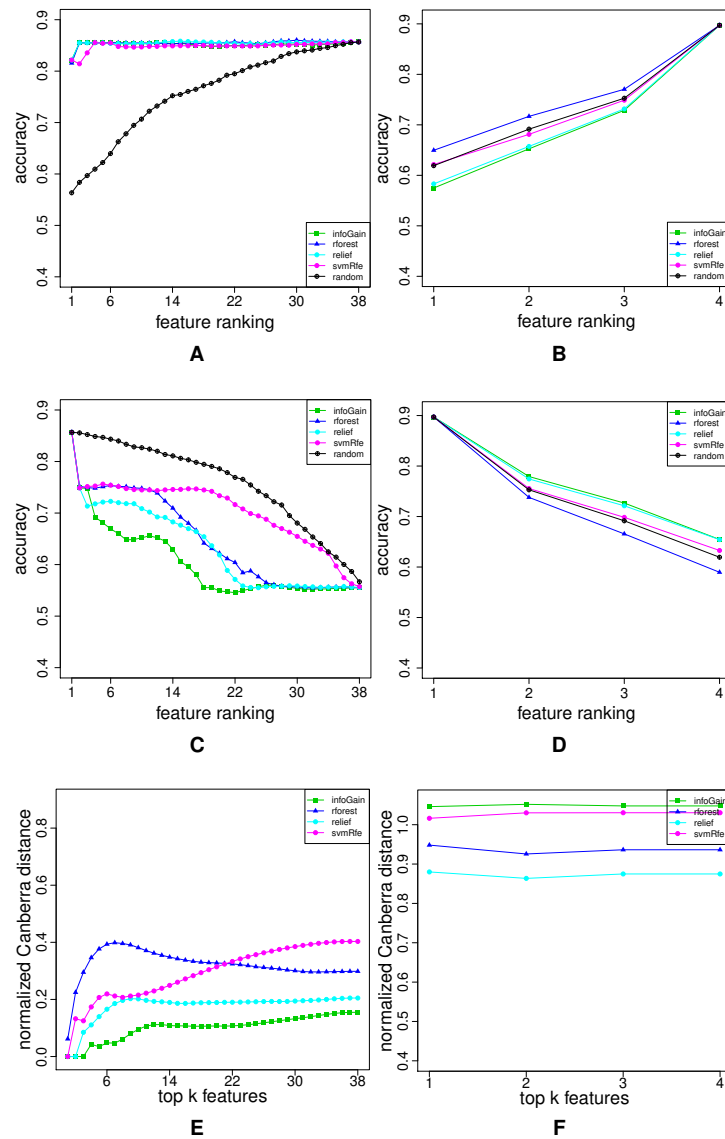
Figure A3. Ranking quality assessment for datasets `australian` (first column) and `balance` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
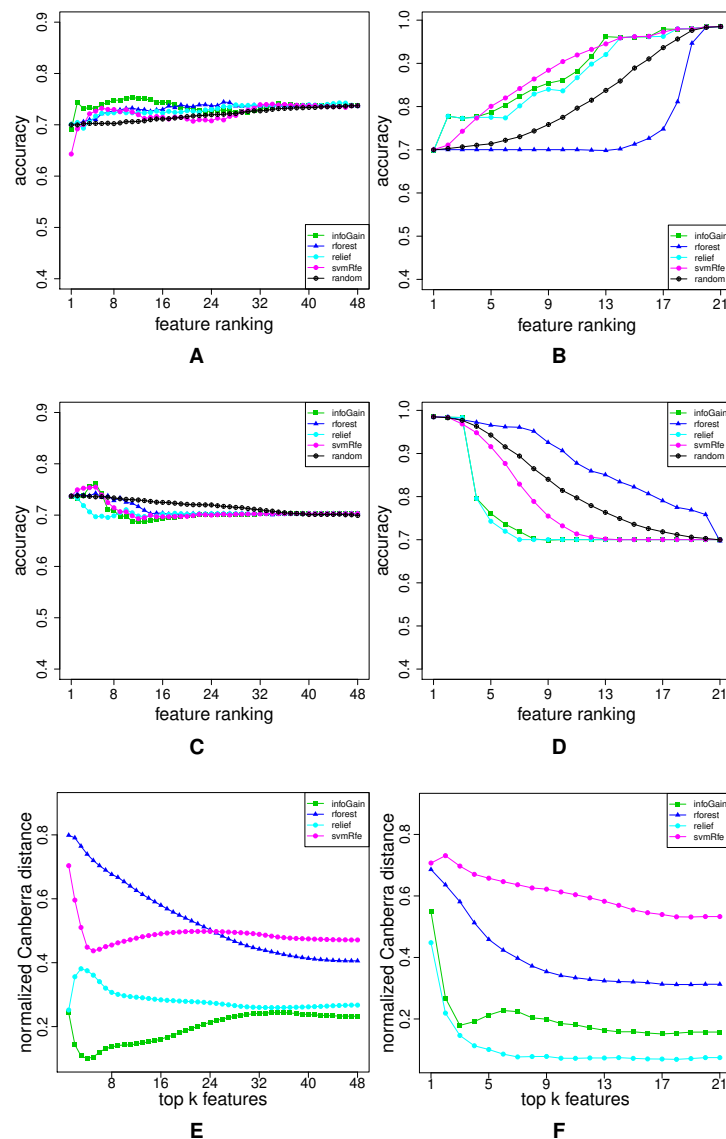
Figure A4. Ranking quality assessment for datasets `breast-cancer` (first column) and `car` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
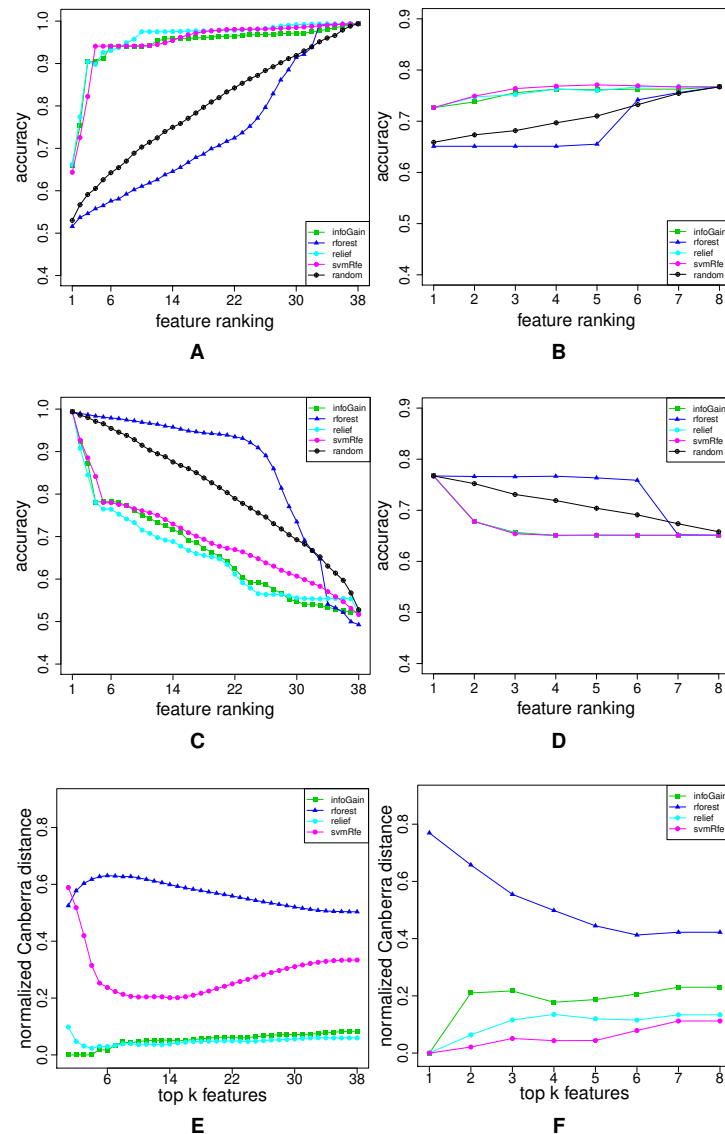
Figure A5. Ranking quality assessment for datasets `chess` (first column) and `diabetes` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
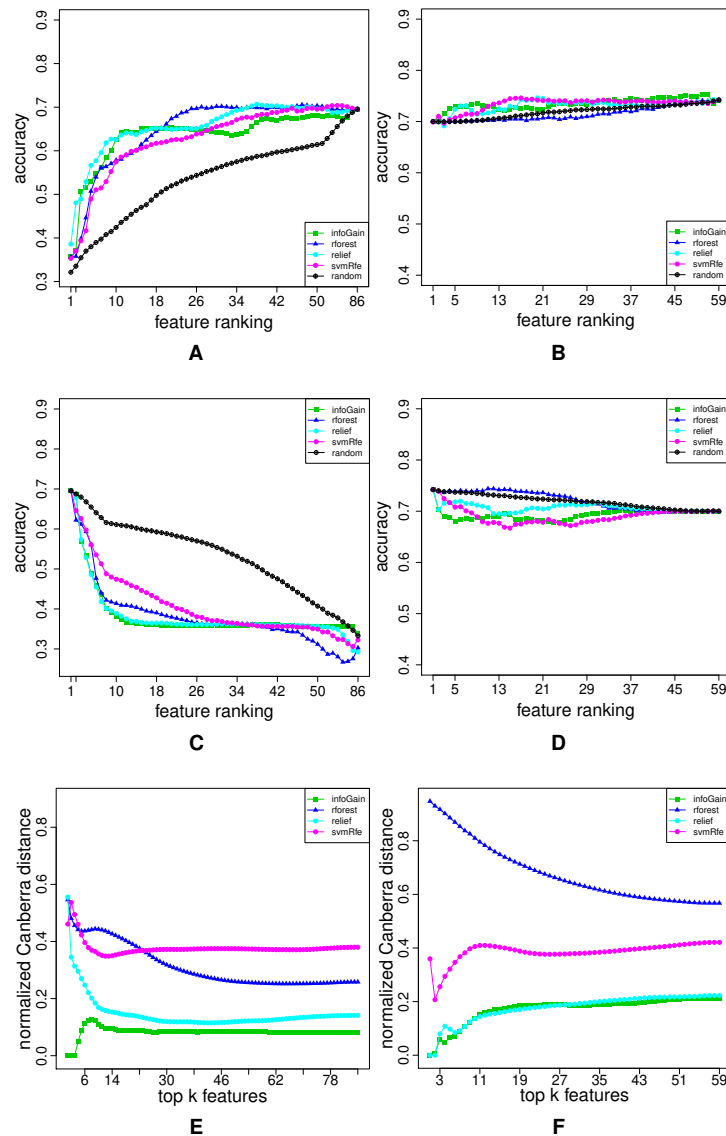
Figure A6. Ranking quality assessment for datasets `diversity` (first column) and `german` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
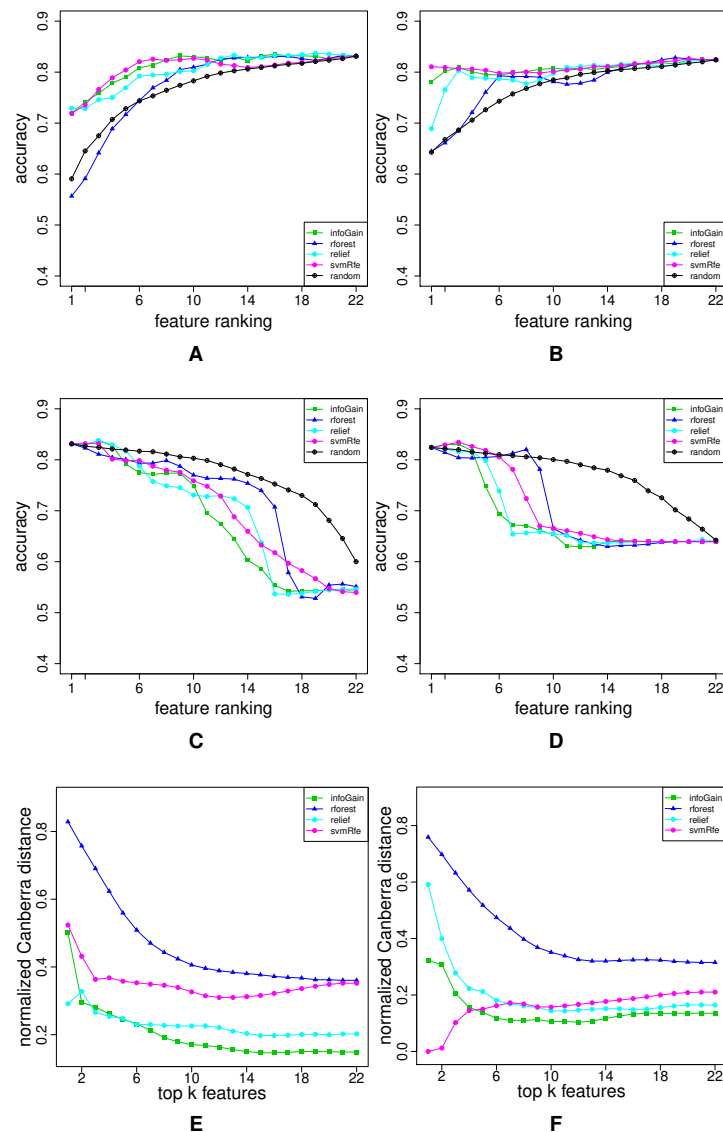
Figure A7. Ranking quality assessment for datasets `heart-c` (first column) and `heart-h` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
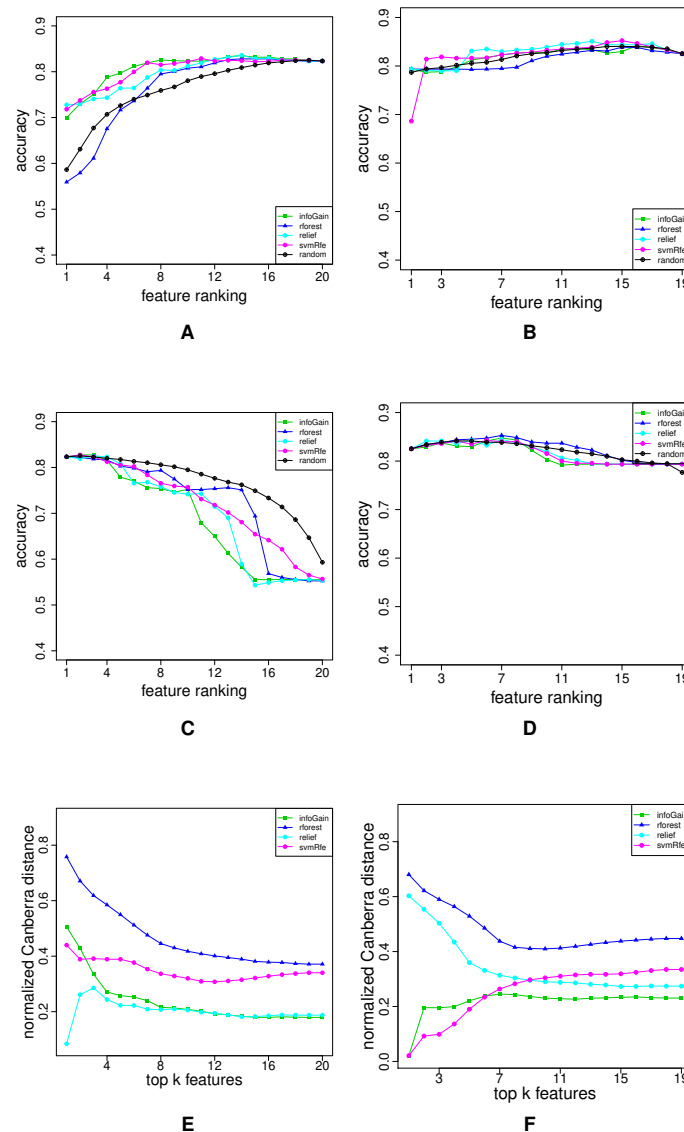
Figure A8. Ranking quality assessment for datasets `heart` (first column) and `hepatitis` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
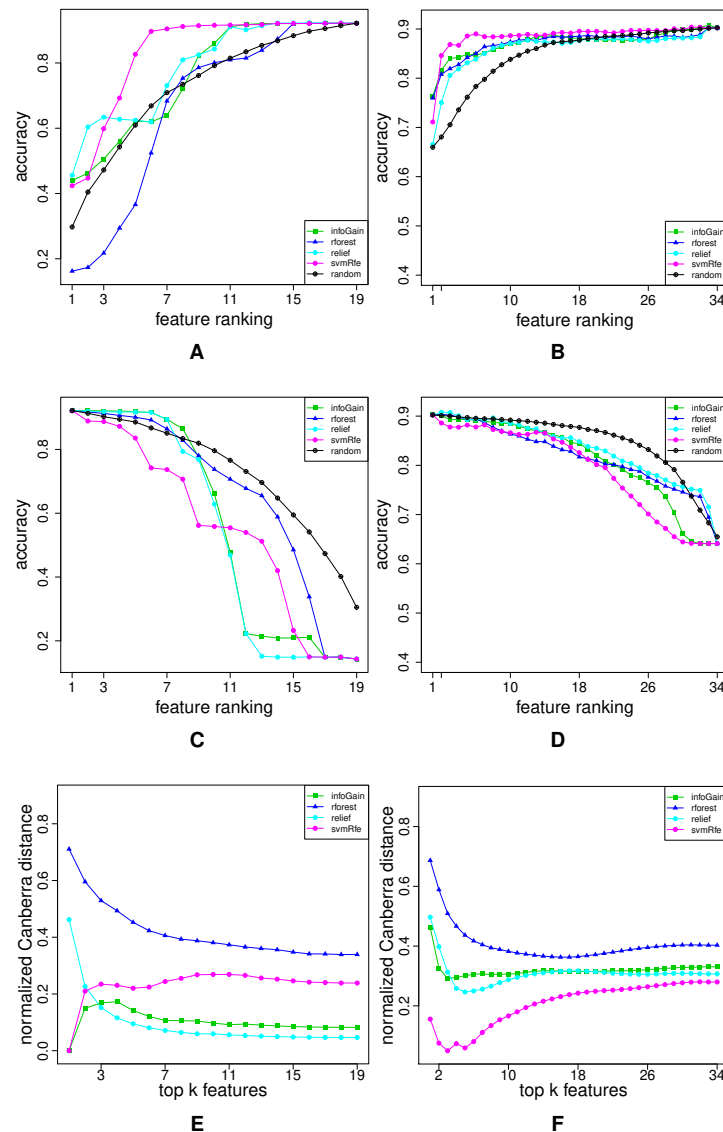
Figure A9. Ranking quality assessment for datasets `image` (first column) and `ionosphere` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
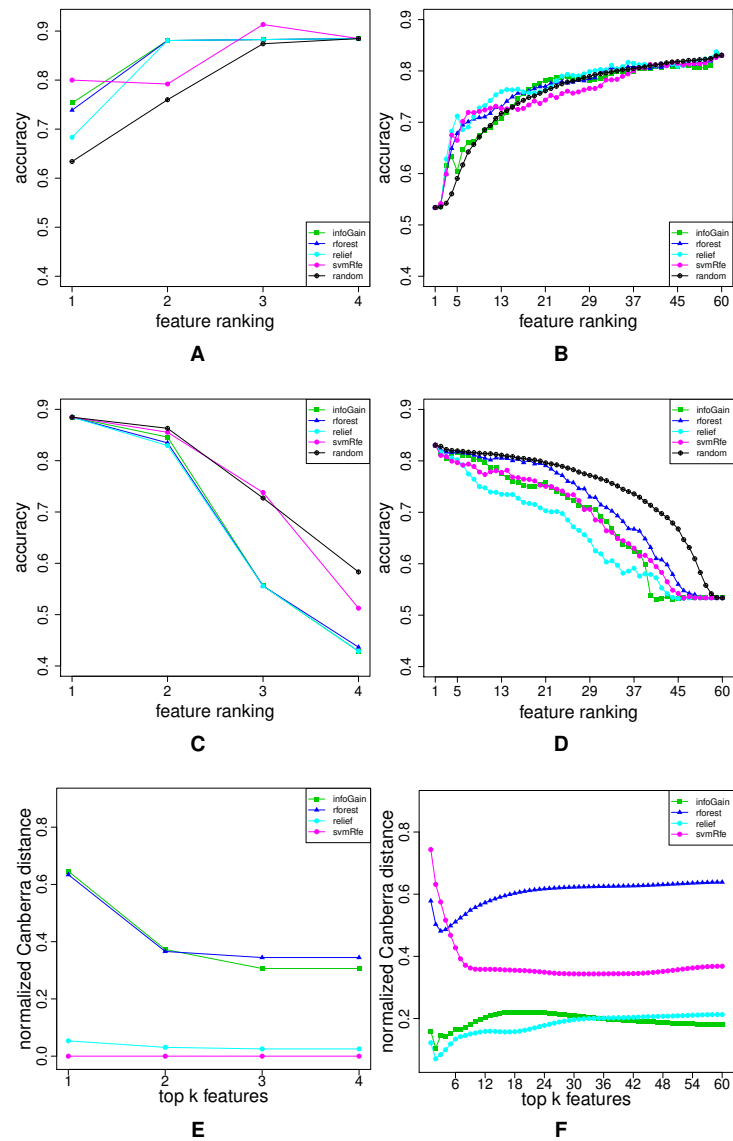
Figure A10. Ranking quality assessment for datasets `iris` (first column) and `sonar` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
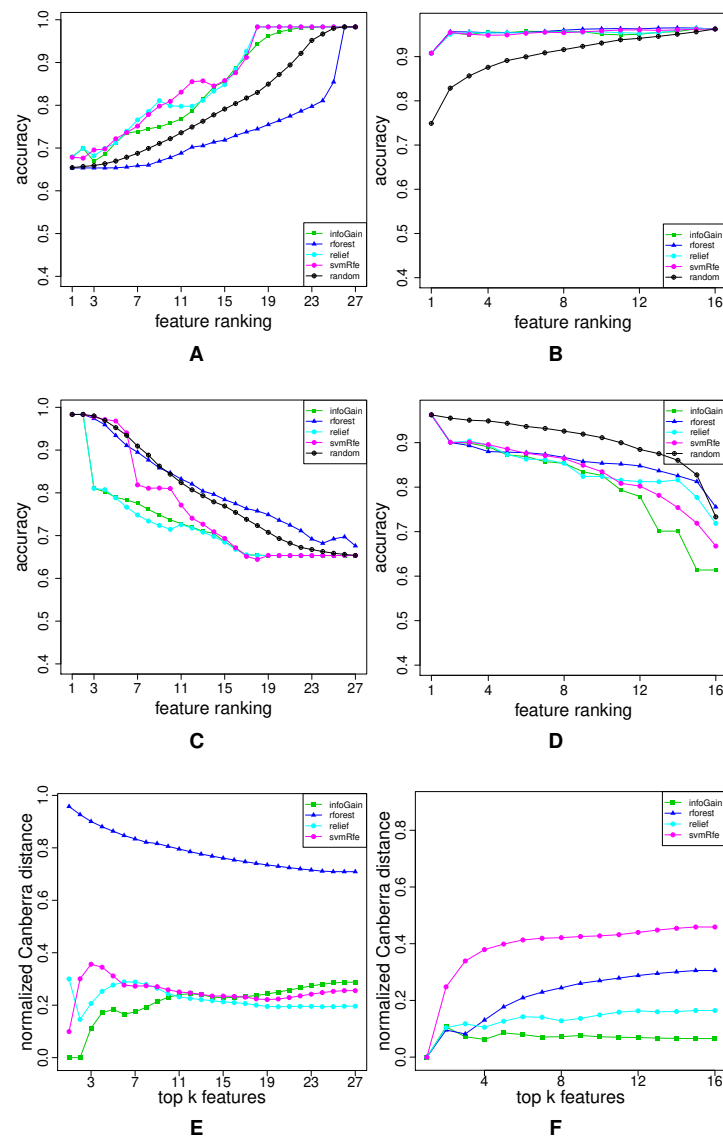
Figure A11. Ranking quality assessment for datasets `tic-tac-toe` (first column) and `vote` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.
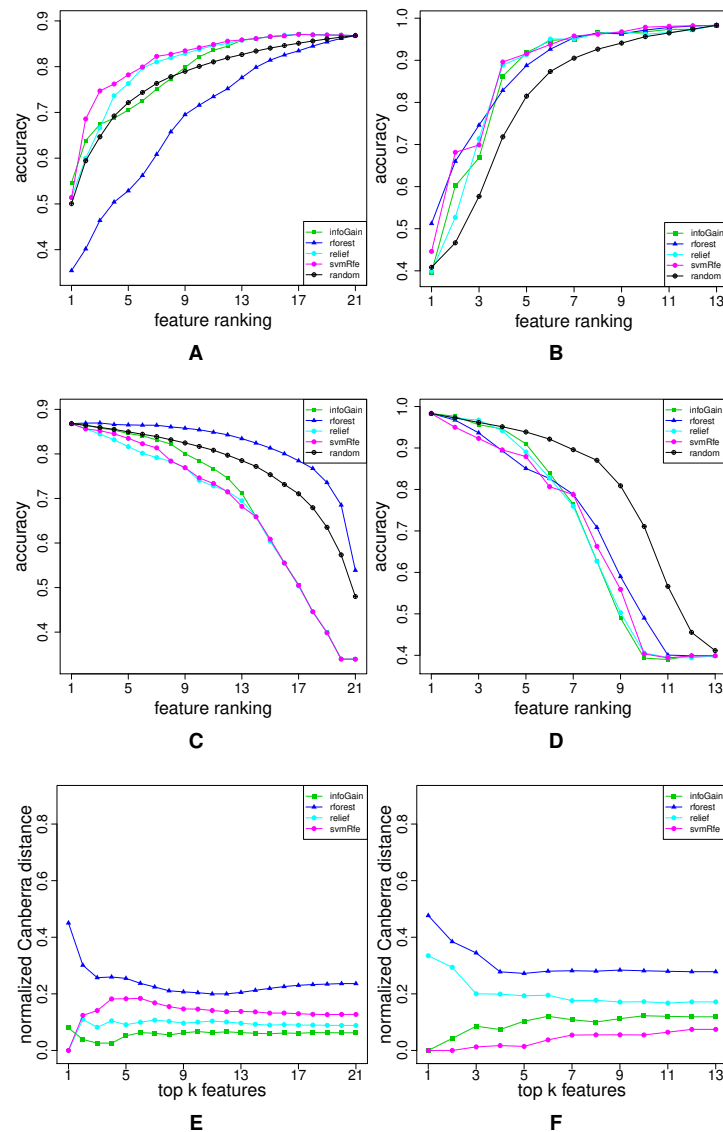
Figure A12. Ranking quality assessment for datasets `waveform` (first column) and `wine` (second column) in terms of the FFA (first row) and RFA curves (second row), and rankings' stability estimates (third row). The FFA/RFA curves are obtained by using SVMs.