Exploiting the ensemble paradigm for stable feature selection: A case study on high-dimensional genomic data

Barbara Pes*, Nicoletta Dessì and Marta Angioni

Università degli Studi di Cagliari, Dipartimento di Matematica e Informatica,
Via Ospedale 72, 09124 Cagliari, Italy
pes@unica.it, dessi@unica.it, martaangioni@gmail.com

**corresponding author*

Abstract
Ensemble classification is a well-established approach that involves fusing the decisions of multiple predictive models. A similar "ensemble logic" has been recently applied to challenging feature selection tasks aimed at identifying the most informative variables (or features) for a given domain of interest. In this work, we discuss the rationale of ensemble feature selection and evaluate the effects and the implications of a specific ensemble approach, namely the *data perturbation strategy*. Basically, it consists in combining multiple selectors that exploit the same core algorithm but are trained on different perturbed versions of the original data. The real potential of this approach, still object of debate in the feature selection literature, is here investigated in conjunction with different kinds of core selection algorithms (both univariate and multivariate). In particular, we evaluate the extent to which the ensemble implementation improves the overall performance of the selection process, in terms of predictive accuracy and stability (i.e., robustness with respect to changes in the training data). Furthermore, we measure the impact of the ensemble approach on the final selection outcome, i.e. on the composition of the selected feature subsets. The results obtained on ten public genomic benchmarks provide useful insight on both the benefits and the limitations of such ensemble approach, paving the way to the exploration of new and wider ensemble schemes.

Keywords: Ensemble paradigm, Feature selection, Data perturbation, Selection stability, High-dimensional genomic data.

1. Introduction

In the context of hybrid intelligent systems [1], different approaches have been proposed that use a suitable combination of computational methods and techniques to handle real world complex problems involving imprecision, uncertainty and high-dimensionality of data. In particular, multi-classifier systems seek to exploit the strengths of diverse classifier models, obtaining enhanced performance by their combination. This approach, also referred as *ensemble learning* paradigm, has received extensive coverage in pattern recognition and machine learning literature [2-6], with a fast increasing number of reported applications [1, 7-10].

In the last years, significant research efforts [11-15] have also explored the extension of the above paradigm to the *feature selection* process [16], which is crucial to the analysis of high dimensional datasets coming from a number of application areas such as text processing and biomedicine. Indeed, by removing features that may be either redundant or irrelevant to the problem at hand, feature selection methods have become indispensable in several knowledge discovery tasks [17,18] that require the identification of a small subset of informative variables. Moreover, feature selection may lead to better predictors, either in terms of learning speed, generalization capability as well as interpretability of the induced model.

As highlighted by recent literature [11,19], a good selection algorithm should achieve an optimal trade-off between *predictive performance*, i.e. the capacity of identifying the most relevant/predictive features, and *stability*, i.e. the robustness of results with respect to changes in the dataset composition (e.g., adding or removing a given percentage of training samples should not affect the selection outcome in a significant way). Neglected until a few years ago, stability is now recognized as a very important issue, especially if subsequent analysis or validation of the selected features are costly.

Since existing selection algorithms are often deficient in stability [20], research activity is increasingly focusing on new approaches that may improve the robustness of selection results. Among them, *ensemble feature selection* has been recommended [21] as a very promising paradigm that does not require complex transformations of the original feature space or prior knowledge on the underlying domain. Basically, the rationale is to exploit the outputs of a set of different selectors, instead of using a single one: indeed, as the combination of multiple classifiers can lead to a better predictive system, similarly the combination of multiple selectors could allow to achieve more reliable and robust selection results.

Actually, a number of experimental studies [11,12,14] have shown that ensemble approaches can really overcome standard selection algorithms in terms of stability, especially in the context of high-dimensional/small sample size domains (such as genomics and biomedicine). However, in this field, the potential and the implications of the ensemble paradigm have not been exhaustively investigated yet, and there is no consensus on the superiority of ensemble methods over standard techniques [22] nor clear indications on when, and to which extent, an ensemble approach may be convenient in terms of both predictive performance and stability [23,24]. This motivates, in our opinion, further and deeper analyses, involving different types of selection algorithms.

Extending our previous research in this field [22,25], we present here an extensive study aimed at investigating the effects of a *data perturbation* ensemble strategy [21], already consolidated in the context of multi-classifiers systems and recently explored within high dimensional feature selection tasks. Basically, it involves the application of a core selection algorithm to different perturbed versions of a dataset, each originating a specific selection outcome (e.g. a feature subset or a ranked list of features): the different outcomes are then combined (through a suitable aggregation function) to obtain the final output. In our study, the effectiveness of this ensemble implementation is systematically evaluated (for different aggregation functions and different number of selected features) against the direct application of the core algorithm to the dataset at hand.

Specifically, we consider ten selection methods, representatives of both *univariate* and *multivariate* approaches: in the first case, each single feature is evaluated independently from the others; in the second one, the inter-dependencies among features are taken into account. For each method, we compare the effectiveness of simple and ensemble implementations, in terms of both predictive performance and stability. Moreover, as further contribution, we measure the similarity among the feature subsets produced in the simple and in the ensemble setting, in order to evaluate the exact extent to which the ensemble approach affects the selection outcome, i.e. the composition of the selected subsets.

As benchmarks for our experiments, we consider ten genomic datasets coming from DNA micro-array experiments [26]: due to the large number of features, coupled with a comparatively small number of samples, they have proved to be very challenging for standard selection algorithms, hence providing an interesting test-bed for ensemble methods.

The results of our experiments give insight on both benefits and limitations of ensemble selection techniques and could represent a useful starting point to best understand the behavior of these techniques as well as the extent of their applicability to specific real-world problems.

The rest of this work is organized as follows. Section 2 provides relevant background concepts as well as a survey of current literature. Section 3 describes the adopted methodology, while section 4 gives more details about the feature selection techniques and the datasets involved in the proposed case study. The experimental results are presented and discussed in section 5. Finally, section 6 outlines concluding remarks and future research directions.

2. Background and literature survey

Feature selection is a critical preprocessing step in data mining. Indeed, it allows to reduce the dimensionality (i.e. the number of features) of a given dataset, making the overall analysis more manageable and often more productive too (since irrelevant and confounding factors are removed). Within high dimensional classification tasks [27,28], it is ordinarily used to extract a subset of features highly correlated to the target class and hence potentially predictive. In this context, feature selection methods can be broadly divided into three groups [29,30]:

- *Filter methods* estimate the relevance of features by looking only at the intrinsic properties of the data, without interacting with the learning algorithm (classifier) that will be ultimately used to infer a model.
- *Wrapper methods,* in contrast, interact with the classifier by using its classification performance as evaluation criterion to select the best feature subset within a space of candidate subsets. Tailored to a specific learning algorithm, wrappers may ensure better results than filters, but with an increased computational cost.
- *Embedded methods* exploit the internal parameters of a suitable classifier to derive the level of significance of the features, and usually achieve a good trade-off between computational cost and performance.

Though many studies have investigated the strengths and weaknesses of existing feature selection algorithms [18,29,31,32], the choice of the most appropriate approach for a given task remains difficult. Furthermore, with the aim of devising suitable solutions for specific problem settings, new proposals are constantly appearing in the feature selection literature. In what follows, we discuss some of these proposals, with specific reference to the selection stability issue (sub-section 2.1) and to the ensemble selection paradigm (sub-section 2.2), both relevant to the study presented in this work.

2.1 Selection stability

By identifying the most representative attributes for a given domain, feature selection often aims at discovering useful knowledge from data, not just at producing an accurate classifier. In this case, the *robustness* of the selection process is equally important as good model performance. Specifically, the term robustness (or *stability*) is used to describe the extent to which a feature selection algorithm is sensitive to changes in the training data [19]: a robust algorithm is capable of producing (almost) the same results when the composition of the dataset is modified to some extent, hence allowing domain experts to have more confidence in the selected features.

In more detail, given a set of training records, each described by a vector of N features, the stability of a selection algorithm can be measured:

(i) by constructing M training sets, each derived by perturbing to some extent (e.g. through a proper resampling procedure) the original set of records;

(ii) by applying the algorithm to each training set: this originates, for each set, an output which can take the form of a subset of features, or of a weighting-scoring or a ranking of the features;

(iii) by comparing, through a proper similarity measure, the M resulting outputs: the comparison is typically conducted on a pair-wise basis, and the resulting similarities values are averaged over the $M(M\text{-}1)/2$ pair-wise comparisons.

Recent literature has focused on suitable experimental procedures for constructing the M training sets (the available protocols essentially differ for the way the original data are perturbed) [20,33] as well as on proper similarity measures to quantify the effect of data perturbations on the selection results [19,21,34,35]. For example, the Pearson's correlation coefficient can be used if the selection output is expressed as a weighting of the features, the Spearman's rank correlation coefficient if the output is a ranking of the features, the Tanimoto distance or the Kuncheva index if the output is a subset of features.

The behavior of the existing protocols and stability measures has been tested in a number of experimental studies [36-40] which focus on high-dimensional datasets such as the ones here considered.

Furthermore, a number of selection approaches have been proposed that explicitly incorporate the stability requirement in their design, i.e. *group feature selection* [41], *prior feature relevance* [42], *sample injection* [21] and *ensemble feature selection* [11]. In particular, the ensemble approach has been suggested as a general-purpose solution for ensuring a good trade-off between stability and classification accuracy.

2.2 Ensemble feature selection

Similarly to the context of supervised learning [2], where multiple classifiers are combined to achieve a better performance, the ensemble selection techniques exploit different selectors under the assumption that "two - or more - heads are better than one". Indeed, as observed in [11], different selection methods may give feature subsets that can be considered local optima in the search space of candidate subsets, and ensemble selection might give a better approximation to the optimal set of features.

Basically, ensemble feature selection involves two main steps:
(i) creating a set of different selectors *(ensemble components)*;
(ii) aggregating the results of the different selectors into a single final decision.

Different approaches have been experimented for generating the ensemble components. The so-called *data perturbation* strategy [11,12,43] entails applying a single selection method to different sampled versions of a given dataset (analogously to bagging or boosting procedures in the field of ensemble learning). On the other hand, the *function perturbation* approach [14,44,45] involves applying different selection methods to the same dataset, but hybrid strategies [46,47] are also possible where perturbation is injected both at the data level and at the function level (e.g., different selection methods are applied to different versions of a dataset).

The aggregation of the results produced by the different selectors, in turn, can be made using different strategies. A general framework for combining several feature subsets into a single "ensemble subset" is presented in [45], where the focus is on filter methods that do not rely on a ranking approach., i.e. do not assign a score to each single feature. When the ensemble components exploit a ranking procedure, on the other hand, the combination of the results produced by the different selectors is usually modeled as a *rank aggregation* problem [15] (i.e., the final output is given in the form of a consensus *ranked list*, as discussed in next section). In this context, a number of aggregation functions have been experimented [48], but it is not clear which of them may be more appropriate for a given task.

Despite an increasing research activity in this field, important concerns still need to be addressed. Indeed, a number of ensemble approaches have been discussed in recent literature [49], but only few guidelines are available [44,47,50] on how to exploit the potential of ensemble feature selection in practical real-world applications (e.g. which ensemble strategy should be used in a specific context? which selection methods should be involved?).

On the one hand, combining different selection algorithms (*function perturbation* approach) can be beneficial in terms of predictive performance [45,51-53], but the best choice of the methods to be combined is often dataset dependent, and the effectiveness of this approach in terms of selection stability is yet to be investigated.

On the other hand, the use of a *data perturbation* strategy has been proposed as the most effective approach to handle selection instability, especially in high dimensional/small sample size domains, such as bioinformatics. Indeed, among the most cited studies in this field, [11,12] have shown that this strategy can produce more stable results than standard selection techniques. In particular, [12] shows that the ensemble approach is strongly beneficial both in terms of stability and predictive performance, as compared with a single SVM-based selector. However, as observed in [24], this strategy has been so far experimented for a limited number of selection methods, with no evidence of a generalized superiority of the ensemble approach over simple techniques.

Moreover, due to its inherently higher computational cost (tied to the use of a set of re-sampled datasets, instead of a single one), the *data perturbation* approach poses greater implementation issues, requiring suitable methodologies to jointly measure both model performance and selection stability, as well as a proper tuning of parameters (such as the number of re-sampled datasets, i.e. the number of selectors to be included in the ensemble [54]). More details about this kind of ensemble strategy, which is the focus of our study, will be provided in next section.

3. Methodological approach

The ensemble approach here explored involves using, as ensemble components, a set of ranking-based selectors (*rankers*), typically falling in the category of filter or embedded selection methods. Extensively applied in high-dimensional domains, rankers produce as output a list (*ranked list*) where the original features appear in descending order of relevance: this list is usually cut at a proper threshold point (*cut-off*) in order to obtain a subset of highly predictive features. In an ensemble selection perspective, different ranked lists can be combined into a single *ensemble list* where the rank (i.e., the relevance) of each feature depends on the feature's rank across all the ensemble components. This ensemble list, in turn, generates the final feature subset. Note that, according to the *data perturbation* strategy discussed in section 2, the different ranked lists to be combined are here obtained by applying the same selection method to different versions of a given dataset: their diversity hence comes from the diversity of the training data.

The aim of our study is to provide more insight on the effectiveness and the implications of such ensemble approach, still object of debate in literature [22,23,49,55]. Specifically, we compare some popular ranking methods with their ensemble counterparts along different dimensions:

(i) *stability* and *predictive performance* of selected subsets;
(ii) *similarity* of selected subsets.

The comparison is carried out for different values of the cut-off threshold, namely for different subset sizes, according to the methodology outlined in sub-sections 3.1 and 3.2.

3.1 Joint evaluation of stability and predictive performance

The predictive performance of a feature subset, i.e. its capacity of discriminating the target class, can be measured by inducing a classification model on that subset and using a proper test set to evaluate this model in terms of metrics such as accuracy or AUC. This is usually done in a cross-validation setting, although it can produce overoptimistic results on small sample size domains [12,56]. On the other hand, research work on designing suitable protocols for stability evaluation is still ongoing [20,21,35], and often stability is not evaluated in conjunction with predictive performance but in independent experiments. Due to the increasing need of achieving a good trade-off between accuracy and robustness [12], we adopt here a methodological approach that involves a single unified framework to jointly evaluate these aspects, in the context of both simple and ensemble ranking.

Simple ranking

Given the original dataset D (with N features), we exploit a sub-sampling procedure to create a number M of reduced datasets D_j (j = 1, 2, …, M), each containing a fraction X of instances randomly drawn from D (Fig. 1). A simple ranking method R is then applied to each dataset D_j, in order to obtain a ranked list L_j as well as a feature subset FS_j containing the n most predictive features (i.e. the highest ranked ones). The resulting M subsets are then compared with each other: indeed, the more similar they are, the more stable the ranking method R.

In more detail, we use the *Kuncheva consistency index* [57] to derive a similarity value for each pair of subsets:

$$similarity(FS_i, FS_j) = \frac{\left|FS_i \cap FS_j\right| - n^2/N}{n - n^2/N} \tag{1}$$

This value expresses the degree of overlapping between the subsets (i.e., $|FS_i \cap FS_j|/n$), with a correction term reflecting the probability that a feature is included in both subsets simply by chance (this probability grows as the subset size n approaches the dimensionality N of the original dataset). The resulting similarity values are then averaged over all pair-wise comparisons, in order to obtain a global evaluation of the degree of stability of the ranking method R.

To incorporate predictive performance evaluation in the above protocol, we also employ each dataset D_j (j = 1, 2, …, M) to induce a classification model that leverages only the features in FS_j: the model performance is estimated on a test set T_j containing the fraction (1-X) of the original instances not included in D_j (Fig. 1). By averaging the accuracy/AUC performance of the resulting M models, we can evaluate the effectiveness of the ranker R in identifying highly predictive features.

***Fig. 1**. Simple ranking: joint evaluation of stability and predictive performance*

Ensemble ranking

The same analysis, in terms of both stability and accuracy/AUC evaluation, is also conducted on the ensemble version of the ranker R. Specifically, as illustrated in Fig. 2, each of the reduced datasets D_j (j = 1, 2, …, M) is further sampled (but with replacement) to get a number K of *bootstrap samples* from which to obtain K ranked lists (each produced by R) to be aggregated into an ensemble list $L_{j\text{-}ensemble}$. Each ensemble list, in turn, produces a feature subset $FS_{j\text{-}ensemble}$ containing the n highest ranked features.

The resulting M subsets are then managed according to the methodological approach previously described, i.e.:

- they are compared on a pair-wise basis (using the Kuncheva index), and their average similarity is used as a measure of the ensemble stability;
- they are employed to build classification models whose average accuracy/AUC is used to evaluate the ensemble capacity of identifying highly predictive features.

This approach enables us to evaluate, within a unified framework, if the overall effectiveness of a given ranking method improves significantly when it is used in an ensemble fashion.

***Fig. 2**. Ensemble ranking: joint evaluation of stability and predictive performance*

3.2 Similarity analysis

To get more insight on the effects of an ensemble selection strategy, we also perform a detailed similarity analysis on the feature subsets produced in the simple and in the ensemble setting. In more detail:

a) We compare, for a given dataset, the feature subsets produced by a simple ranker R and its ensemble version $R_{ensemble}$ (*intra-method similarity*). This kind of analysis can be extended to different ensemble versions, implemented with different aggregation strategies, of the same ranker R, so as to evaluate the extent to which the selection outcome is affected by the choice of the aggregation function.

b) We measure, for a given dataset, the similarity among the subsets produced by a number of different rankers R^i (i = 1, 2, …, B) used in their simple form (*inter-method similarity*). This analysis is then performed on the subsets selected by the same rankers in their ensemble version, $R^i_{ensemble}$ (i = 1, 2, …, B), in order to evaluate the influence of the ensemble approach on the pattern of agreement among different ranking methods.

In both intra-method and inter-method studies, we employ the Kuncheva measure [57] as consistency index; indeed, it has proved to be a good option also in the context of such a similarity analysis [58,59].

4. The case study: datasets, methods and settings

According to the methodological approach presented in previous section, we conducted an extensive study involving different datasets (sub-section 4.1) and different ranking methods (sub-section 4.2). For each ranking method, we implemented three ensemble versions based on different strategies for the aggregation of the ensemble components (sub-section 4.3). Some further details on the experimental settings adopted in our study are given in sub-section 4.4.

4.1 Genomic benchmarks

Biomarker discovery from high dimensional genomic data is a challenging benchmark to test the behavior of ensemble approaches. Within this domain, we considered ten datasets [60-69] deriving from DNA micro-array experiments [26]; specifically, five of them represent binary classification tasks while the other five are multi-class problems. The considered datasets are reported in Table 1 along with their main characteristics (number of features, number of instances and number of classes).

Table 1. *Micro-array datasets used in the experiments.*

In terms of feature selection, the task here is to identify the genes most useful in discriminating the target class (e.g., normal vs tumor or a type of pathology vs the other ones). Note that all the above datasets are characterized by a large number of features (genes) and a small number of samples, which makes it difficult to achieve a good trade-off between predictive performance and robustness.

4.2 Ranking methods

In our experiments, we considered ten ranking methods. Specifically, we used both *univariate* techniques, which evaluate each feature independently from the others, and *multivariate* techniques, which take into account inter-dependencies among features.

As representatives of univariate approaches, we employed:

- *Information Gain* (IG), *Symmetrical Uncertainty* (SU) and *Gain Ratio* (GR) that rely on the information-theoretical concept of entropy [70]. In particular, IG measures the amount by which the entropy of the class (i.e., the uncertainty about its prediction) decreases when the value of a given feature is known. Both SU and GR refine the IG definition by introducing proper normalization factors.
- *Chi Squared* (χ^2) that leverages the chi-squared statistic [71]. Specifically, for each feature, the chi-squared statistic is evaluated with respect to the class: the larger the chi-squared, the more important the feature is for the predictive task at hand.
- *OneR* (OR) that exploits a simple rule-based classifier. Following the approach proposed in [72], a classification rule is induced for each feature; the accuracy of each rule is then estimated, and the features are ranked according to the quality of the corresponding rules.

Moreover, as representatives of multivariate approaches, we considered:

- *ReliefF* (*RF*) and *ReliefF-W* (*RFW*) that evaluate the level of significance of input features based on their ability to distinguish between instances that are near to each other [73]. Specifically, given a probe instance, RF compares the value a feature takes in that instance and its nearest neighbors (one for each class); this comparison is extended to a suitable number of probe instances to obtain a measure of the feature's discriminative power. In the RFW approach, a weighting mechanism is introduced to take account of the neighbors' distance.
- *SVM-ONE* that employs a linear SVM classifier to derive a weight for each feature: this weight corresponds to the absolute value of the feature's coefficient in the hyperplane equation induced by SVM [74].
- *SVM-RFE* that, in turn, relies on a linear SVM classifier. Differently from SVM-ONE, it builds the final ranking of features by applying a backward elimination strategy [75]: the features with the lowest weights are iteratively removed and the overall weighting process is repeated on the remaining features. The percentage of features removed at each iteration has a great impact on the computational cost of this approach: in our experiments, we set this parameter as 10% (*SVM-RFE10*) and 50% (*SVM-RFE50*).

4.3 Aggregation strategies

In the context of ensemble feature ranking, a suitable aggregation function is needed that assigns an overall score to each feature based on the feature's rank across all the lists (ensemble components) to be combined. In our study, we evaluated the following approaches:

- *Mean aggregation*. For each feature, the rank value is averaged over all the original lists (note that the feature's rank in a given list corresponds to its ranking position: the most relevant feature has rank 1, the least relevant rank *N*). The resulting mean value is then used as the feature's overall score: the smaller this score, the higher the overall importance of the feature.

- *Median aggregation*. Similar to the previous one, this strategy involves finding the median rank value across all the original lists.
- *Exponential aggregation.* For each feature, a local score is calculated, within each of the original lists, as an exponentially decreasing function of the rank, namely:

$$exp(-rank/t) \tag{2}$$

  where t is a suitable threshold [23,48]. Then, the feature's overall score is derived by summing up the corresponding local scores: in this case, the higher the overall score, the higher the relevance of the feature.

Based on the above strategies, the features can be finally ordered (from the most important to the least important) into a single ensemble list. Though more complicated aggregation functions have been proposed in recent literature [48], they do not seem to improve the ensemble performance in a significant way [76].

4.4 Experimental settings

In our empirical study, we compared each of the ranking methods previously described (sub-section 4.2) with its ensemble counterpart. In more detail, for each method, we implemented three ensemble versions: *ensemble-mean*, *ensemble-median* and *ensemble-exponential* (sub-section 4.3). Specifically, based on preliminary tuning experiments, the threshold t of the exponential aggregation function was set as 5% of the original number of features (N).

The other parameters of our methodology were set as follows:

- number of reduced datasets: $M = 50$;
- fraction of original instances included in each reduced dataset: $X = 0.90$;
- cut-off value (i.e. subset size): we explored a range of values, from $n = 0.3\%$ to $n = 5\%$ of the original number of features (N);
- number of bootstrap samples: $K = 50$.

Note that the ensemble implementation here adopted has a computational cost that depends linearly on the number K of bootstrap samples (i.e. on the number of the ensemble components). In setting this critical parameter, we relied on the recommendations in recent literature [54] as well as on a number of tuning experiments: it turned out that using more than 50 bootstrap samples affects the computational cost without improving the ensemble performance in a significant way.

Finally, in evaluating the predictive performance of the selected feature subsets, we tested two classification algorithms, i.e. a Support Vector Machine (SVM) classifier [77] and a Random Forest classifier [78,79], which have proved to be "best of class" algorithms for the high-dimensional benchmarks here considered [80,81]. Specifically, for the SVM classifier we chose a linear kernel, while the Random Forest classifier was parametrized using 100 trees and a $log_2 n+1$ random features, based on common practice in this domain.

The overall analysis was carried out using a software package built on top of the WEKA library [82].

5. Experimental results

Though the *data perturbation* ensemble strategy has been suggested (see section 2) as a primary avenue for improving the performance of standard selection algorithms, especially in terms of stability, our empirical study clearly shows that the ensemble approach is not always and necessarily beneficial in itself, but only in dependence on the "intrinsic" effectiveness of the considered method.

In what follows, we give an overview of the most significant experimental results. In particular, a summary of AUC/accuracy patterns is presented in sub-section 5.1, while sub-section 5.2 summarizes the stability patterns and sub-section 5.3 further discusses these patterns in light of the findings of the similarity analysis (see sub-section 3.2).

5.1. AUC/accuracy patterns

While primarily conceived to obtain more stable selection results, the ensemble approach here investigated can sometimes be beneficial in terms of predictive performance. For example, when looking at the Colon tumor dataset (a noisy benchmark [83] whose classes are not linearly separable [84]), we have observed a positive impact of the ensemble approach, both in terms of AUC and accuracy, but limited to the selection methods that exhibit the worst behavior in the simple form. In contrast, methods that show a comparatively better performance in the simple setting do not take significant advantage of the ensemble implementation.

Specifically, for both SVM and Random Forest classifiers, Fig. 3 shows the AUC patterns of the OR method, representative of the univariate category, as well as those of the SVM-ONE method, representative of the multivariate category. Both of them exhibit a better behavior in the ensemble version, with no significant differences among the three aggregation strategies (i.e., mean, median and exponential).

In more detail, but limited to the SVM classifier, Fig. 4(a) shows the AUC patterns of all the univariate methods, in their simple (left) and ensemble version (right). Note that, for more clarity and readability, only the curves obtained with the mean aggregation function have been reported here. Similarly, Fig. 4(b) shows the AUC patterns of all the multivariate methods, in their simple (left) and ensemble-mean (right) version. It is clear that the strongest methods, i.e. those achieving the best AUC performance in the simple setting, do not take advantage of the ensemble implementation. Rather, it seems that the main effect of the ensemble approach is to reduce the differences among the original methods, leading the weakest algorithms to reach (in terms of AUC performance) the strongest ones. Similar considerations hold for the accuracy patterns, as we can see in Fig. 5.

The corresponding AUC/accuracy curves for the Random Forest classifier have been here omitted for the sake of space, but the results can be found in tabular form in the attached supplementary material. It is interesting to note that, in general, the Random Forest classifier outperforms the SVM classifier in terms of AUC, while the SVM performance seems to be slightly better in terms of accuracy. This trend can be observed, for example, in Figs. 6 and 7 that compare the SVM and Random Forest classifiers when used in conjunction with OR and SVM-ONE selection methods.

The significance of the AUC/accuracy differences shown in Figs. 3-7 has been evaluated using a *two-tailed paired t-test*, as in numerous similar studies. We have found that AUC differences in the order of 0.04 can be considered significant at a confidence level of at least 95%. Higher differences, for example those observed in Fig.3 for the SVM-ONE method, have also proved to be significant with the *corrected resampled test* [70], which is a more restrictive version of the standard t-test.

As regards the other benchmarks included in this study (see Table 1), the overall results are provided as supplementary material. To summarize, we can remark that the ensemble approach does not have a generalized positive impact in terms of predictive performance, but can be nonetheless able to strengthen, to some extent, the weakest methods. When viewed along with the stability patterns, where the impact of the ensemble approach is quite stronger (see sub-section 5.2), these results provide a better understanding of the data diversity ensemble strategy, so far applied to a limited number of selection methods [12,23].

Fig.3. *AUC patterns on the Colon dataset, for both SVM (a) and Random Forest (b) classifiers. The performance of OR (left) and SVM-ONE (right) methods is evaluated in the simple and in the ensemble implementation. For both methods, three ensemble versions are considered, i.e. mean, median and exponential.*

Fig. 4. *AUC patterns on the Colon dataset using the SVM classifier. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

Fig. 5. *Accuracy patterns on the Colon dataset using the SVM classifier. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

Fig. 6. *AUC patterns on the Colon dataset. Comparison between SVM and Random Forest classifiers using OR (a) and SVM-ONE (b) methods in their simple (left) and ensemble-mean (right) version.*

Fig. 7. *Accuracy patterns on the Colon dataset. Comparison between SVM and Random Forest classifiers using OR (a) and SVM-ONE (b) methods in their simple (left) and ensemble-mean (right) version.*

5.2. Stability patterns

The stability analysis revealed a very strong impact of the ensemble approach, at least for some selection methods. For example, the OR and SVM-RFE50 rankers, respectively in the univariate and in the multivariate category, systematically benefit from the ensemble implementation irrespective of the considered dataset, as shown in Figs. 8 and 9. Again, the choice of the aggregation function does not affect the results in a significant way.

Overall, the adoption of an ensemble strategy is really beneficial only for the methods that are less stable in the simple setting, in particular GR and OR in the univariate group and the SVM-based rankers (especially SVM-RFE10 and SVM-RFE50) in the multivariate group. Fig. 10 and Fig.11, for example, show the stability patterns of all the selection methods, in their simple and ensemble-mean setting, respectively for the Prostate dataset (a binary problem) and for the MLL dataset (a multi-class problem). Analogously to what observed for the AUC/accuracy patterns (Figs. 4 and 5), we can see here that the ensemble approach induces a gain in stability that, in some way, is "inversely proportional" to the stability of the original method. Indeed, the methods that benefit to a greater extent from the ensemble implementation are those that perform worse in the simple setting. Conversely, the methods that are more stable in the simple version (i.e. χ^2, IG and SU in the univariate group; RF and RFW in the multivariate group) benefit to a limited (or null) extent from the ensemble approach.

Once again, similarly to what observed in Figs. 4 and 5, the differences among the original methods are significantly reduced in the ensemble setting, since the weakest algorithms tend to reach the strongest ones. Hence, it turns out that injecting diversity into the training data, which is the rationale of the ensemble approach here explored, has the effect of producing more uniform stability (as well as AUC/accuracy) patterns. Remarkably, even quite dissimilar methods, when trained in a sufficiently diversified data space, can produce very similar (sometimes coincident) results in terms of overall predictive/stability performance. This is a very interesting, so far neglected, implication of the ensemble selection paradigm.

***Fig. 8**. Stability patterns for the OR method, in its simple and ensemble version. For each dataset, three ensemble versions are considered, i.e. mean, median and exponential.*

***Fig. 9**. Stability patterns for the SVM-RFE50 method, in its simple and ensemble version. For each dataset, three ensemble versions are considered, i.e. mean, median and exponential.*

***Fig. 10**. Stability patterns on the Prostate dataset. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

***Fig. 11**. Stability patterns on the MLL dataset. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

5.3. Similarity analysis and discussion

The results discussed in both sub-sections 5.1 and 5.2 significantly extend some well-known studies [11,12,23] that use a similar methodology but involve a lower number of genomic benchmarks as well as a lower number of selection methods. As further contribution, we also performed a detailed similarity analysis among the feature subsets produced by the considered ranking methods. According to the methodology presented in sub-section 3.2, we conducted this analysis along two dimensions: (i) we compared, for each method, the feature subsets obtained in the simple and in the ensemble setting (*intra-method similarity*); (ii) we evaluated the pattern of agreement among the different methods (i.e. the extent to which they produce overlapping subsets), both in case of simple and ensemble implementation (*inter-method similarity*).

The findings of the intra-method analysis have been summarized in terms of similarity matrices. As an example, Table 2 and Table 3 show some significant results for the DLBCL dataset. Within a given matrix, each cell contains the Kuncheva similarity value for a pair of subsets selected by two versions of the same ranker. Specifically, for each ranker, we considered three ensemble versions (mean, median and exponential) as well as the simple version. Note that the subsets here compared contain 1% of the original features, but the same analysis has been performed for feature subsets of different sizes. Overall, these results (as well as those obtained on the other benchmarks, here omitted for the sake of space), help to best understand the patterns presented in the previous sub-sections.

Indeed, for both the univariate (Table 2) and the multivariate (Table 3) methods, it turns out that the stronger the simple ranker (in terms of predictive performance/stability), the higher its similarity (in terms of selected features) with the corresponding ensemble rankers. For example, for both IG and RF, which have a relatively good performance in the simple version without benefiting from the ensemble approach, the simple and the ensemble subsets are quite similar to each other. In contrast, for the weakest methods, whose performance is significantly improved by the ensemble implementation, the degree of overlapping between the simple and the ensemble subsets is lower.

This is the case of GR and OR (in the univariate group) as well as SVM_RFE10 and SVM_RFE50 (in the multivariate one).

This kind of similarity analysis quantifies, for each single method, the real impact of the ensemble implementation on the final selection outcome (i.e. the selected feature subsets), integrating (and partially explaining) the results of the AUC/accuracy and stability analysis. However, the most interesting aspect to investigate is what emerged by the comparison of the patterns of the different methods (see sub-sections 5.1 and 5.2), in particular the fact that they tend to exhibit a very similar behavior (despite their inner algorithmic differences) when used in the ensemble version.

In this regard, it is useful to evaluate the inter-method similarity among the simple rankers as well as among the ensemble ones. In case of simple ranking, Table 4 shows an example of similarity matrix for the Prostate dataset: each entry represents the Kuncheva consistency value for the pair of subsets selected by the rankers in the corresponding row and column (here, we are considering subsets containing 1% of the original features). Different shades of gray are used to highlight different similarity ranges (the darker is the gray, the higher the similarity).

The corresponding similarity matrix for the ensemble rankers (specifically, the ensemble-mean version) is shown in Table 5. Overall, we can see that the degree of overlapping among the ensemble subsets is higher than that among the simple subsets: indeed, the average similarity over all pairwise comparisons turns out to be 0.53 in Table 5, while in Table 4 it is 0.42. We also derived the average similarity trend for feature subsets of increasing size (Fig. 12): it is clear that the ensemble approach (red curve) reduces the dissimilarity among the different rankers, though the degree of overlapping among the ensemble subsets is still partial.

Despite the specificities of each dataset, the above considerations are also valid for the other genomic benchmarks included in our study. Hence, the inter-method similarity analysis has shown that different ranking methods become more similar to each other (in terms of selection outcome) when used in the ensemble version. However, even in the ensemble setting, important differences still remain, especially between univariate and multivariate approaches (and, within the multivariate approaches, between RF/RFW and SVM-based methods). On the other hand, despite the above differences, the ensemble rankers turn out to be comparable (often equivalent) in terms of predictive performance and stability, as discussed in sub-sections 5.1 and 5.2. Thus, we can conclude that they are capable of providing different (at least to some extent) but equally good (and hence potentially complementary) representations of the underlying domain. This is especially important in the biomedical field here considered, where different sets of genomic markers can exist for a given pathological condition.

Table 2. *DLBCL dataset: intra-method similarity matrices for the univariate methods IG, GR and OR.*

Table 3. *DLBCL dataset: intra-method similarity matrices for the multivariate methods RF, SVM-RFE10 and SVM-RFE50.*

Table 4. *Prostate dataset: inter-method similarity among the simple rankers.*

Table 5. *Prostate dataset: inter-method similarity among the ensemble rankers.*

Fig. 12. *Prostate dataset: average inter-method similarity for feature subsets of increasing size.*

6. Concluding remarks and future research directions

In this work, we explored the effects and the potential benefits of ensemble feature selection in the context of biomarker discovery from high-dimensional genomic data. Specifically, using a methodological approach that leverages best practices from literature, we extensively evaluated the effectiveness of a *data perturbation* ensemble strategy, which entails enlarging and diversifying the original space of training data.

Our main contribution is twofold:

(i) We analyzed, in a joint manner, both the stability and the predictive performance of different selection algorithms, in their simple and ensemble implementation, while so far most of the studies have focused on only one aspect at a time (especially accuracy). Overall, our results indicate that the beneficial impact of the ensemble approach is “inversely proportional” to the strength of method itself, i.e. only the least stable/effective methods really take advantage of a computationally expensive ensemble setting. This may explain the (apparently) discordant findings in recent literature, where different studies seem to achieve different conclusions about the beneficial impact of ensemble feature selection [11,12,22,23,55]. Interestingly, the main effect of the ensemble strategy here explored, based on injecting diversity into the training data, is to narrow the gap between the weakest and the strongest methods, leading to almost uniform patterns in terms of accuracy and stability.

(ii) We also explored the extent to which the ensemble implementation affects the selection outcome, i.e. the composition of the selected subsets. On the one hand, this analysis confirmed that the weak (especially in terms of stability) methods are far more affected by the ensemble approach. Furthermore, it turned out that different methods, when used in the ensemble version, tend to produce more similar subsets but this does not explain, actually, the fact that their accuracy/stability patterns become almost coincident. Indeed, the “ensemble subsets” still overlap only to some extent, giving somewhat different but equally good (in terms of overall performance) solutions for the problem at hand. This can provide a deeper understanding of complex domains such as the one here considered (where different sets of genetic markers can exist for a given pathological state). To the best of our knowledge, no study in literature has so far adopted such a similarity-based point of view in the analysis of the ensemble methods.

From this starting point, our work can be extended in a number of directions. First, further insight could be gained by analyzing datasets from different real-world scenarios. In addition, it would be interesting to explore the effects of wider ensemble schemes, such as hybrid approaches that combine the *data perturbation* strategy (here considered) with a suitable *function perturbation* strategy aimed at jointly exploiting the strengths of different selection algorithms. Indeed, our study has shown that different selection methods, in their data-perturbed ensemble version, tend to achieve a comparable performance although selecting (partially) different solutions: these (equally good) solutions, in turn, could be combined in order to achieve a more complete representation of the underlying domain. Our future research will be devoted to explore the potential of such enlarged ensemble approach.

References

[1] M. Woz´niak, M. Graña, E. Corchado, A survey of multiple classifier systems as hybrid systems, Information Fusion 16 (2014) 3-17.

[2] T. Dietterich, Ensemble methods in machine learning, in: Multiple Classifier Systems, Lecture Notes in Computer Science, vol. 1857, Springer, Berlin, Heidelberg, 2000, pp. 1–15.

[3] L. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms, Wiley-Interscience, 2004.

[4] R. Polikar, Ensemble based systems in decision making, IEEE Circuits and Systems Magazine 6 (3) (2006) 21–45.

[5] L. Rokach, Taxonomy for characterizing ensemble methods in classification tasks: a review and annotated bibliography, Computational Statistics and Data Analysis 53 (12) (2009) 4046–4072.

[6] G. Seni, J. Elder, Ensemble Methods in Data Mining: Improving Accuracy Through Combining Predictions, Morgan and Claypool Publishers, 2010.

[7] N. Oza, K. Tumer, Classifier ensembles: select real-world applications, Information Fusion 9 (1) (2008) 4–20.

[8] P. Yang, Y.H. Yang, B.B. Zhou, A.Y. Zomaya, A review of ensemble methods in bioinformatics, Current Bioinformatics, 5(4) (2010) 296-308.

[9] F. Enríquez, F.L. Cruz, F.J. Ortega, C.G. Vallejo, J.A. Troyano, A comparative study of classifier combination applied to NLP tasks, Information Fusion 14 (2013) 255–267.

[10] J.A. Sáez, M. Galar, J. Luengo, F. Herrera, INFFC: An iterative class noise filter based on the fusion of classifiers with noise sensitivity control, Information Fusion 27 (2016) 19–32.

[11] Y. Saeys, T. Abeel, Y. Van de Peer, Robust Feature Selection Using Ensemble Feature Selection Techniques, in: Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science, vol. 5212, Springer, Berlin, Heidelberg, 2008, pp. 313-325.

[12] T. Abeel, T. Helleputte, Y. Van de Peer, P. Dupont, Y. Saeys, Robust biomarker identification for cancer diagnosis with ensemble feature selection methods, Bioinformatics, 26(3) (2010) 392–398.

[13] S. Van Landeghem, T. Abeel, Y. Saeys, Y. Van de Peer, Discriminative and informative features for biomolecular text mining with ensemble feature selection, Bioinformatics 26(18) (2010) i554-60.

[14] F. Yang, K.Z. Mao, Robust Feature Selection for Microarray Data Based on Multicriterion Fusion, IEEE/ACM Transactions on Computational Biology and Bioinformatics 8(4) (2011) 1080 – 1092.

[15] W. Altidor, T.M. Khoshgoftaar, J. Van Hulse, A. Napolitano, Ensemble Feature Ranking Methods for Data Intensive Computing Applications, in: B. Furth, A. Escalante (Eds.), Handbook of Data Intensive Computing, Springer, New York, 2011, pp. 349-376.

[16] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of Machine Learning Research 3 (2003) 1157-1182.

[17] V. Kumar, S. Minz, Feature Selection: A literature Review, Smart Computing Review 4(3) (2014) 211-229.

[18] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, A review of feature selection methods on synthetic data, Knowledge and Information Systems 34(3) (2013) 483-519.

[19] A. Kalousis, J. Prados, M. Hilario, Stability of feature selection algorithms: a study on high-dimensional spaces, Knowledge and Information Systems 12(1) (2007) 95-116.

[20] W. Awada, T.M. Khoshgoftaar, D. Dittman, R. Wald, A. Napolitano, A review of the stability of feature selection techniques for bioinformatics data, in: IEEE 13th International Conference on Information Reuse and Integration, IEEE, 2012, pp. 356–363.

[21] H. Zengyou, Y. Weichuan, Stable feature selection for biomarker discovery, Computational Biology and Chemistry 34 (2010) 215–225.

[22] N. Dessì, B. Pes, Stability in Biomarker Discovery: Does Ensemble Feature Selection Really Help?, in: Current Approaches in Applied Artificial Intelligence, Lecture Notes in Computer Science, vol. 9101, Springer International Publishing, 2015, pp. 191-200.

[23] A.C. Haury, P. Gestraud, J.P. Vert, The Influence of Feature Selection Methods on Accuracy, Stability and Interpretability of Molecular Signatures, PLOS ONE 6(12) (2011) e28210.

[24] L.I. Kuncheva, C.J. Smith, Y. Syed, C.O. Phillips, K.E. Lewis, Evaluation of Feature Ranking Ensembles for High-Dimensional Biomedical Data: A Case Study, in: IEEE 12th International Conference on Data Mining Workshops, IEEE, 2012, pp. 49 – 56.

[25] N. Dessì, B. Pes, M. Angioni, On Stability of Ensemble Gene Selection, in: Intelligent Data Engineering and Automated Learning – IDEAL 2015, Lecture Notes in Computer Science, vol. 9375, Springer International Publishing, 2015, pp. 416-423.

[26] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, J.M. Benítez, F. Herrera, A review of microarray datasets and applied feature selection methods, Information Sciences 282 (2014) 111–135.

[27] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Feature Selection for High-Dimensional Data, Springer International Publishing, 2015.

[28] N. Dessì, B. Pes, Similarity of feature selection methods: An empirical study across data intensive classification tasks, Expert Systems with Applications 42(10) (2015) 4632–4642.

[29] J. Tang, S. Alelyani, H. Liu, Feature selection for classification: A review, in: C.C. Aggarwal (Ed.), Data classification: Algorithms and applications, CRC Press, 2014, pp. 37–64.

[30] G. Chandrashekar, F. Sahin, A survey on feature selection methods, Computers and Electrical Engineering 40 (2014) 16–28.

[31] M.A. Hall, G. Holmes, Benchmarking attribute selection techniques for discrete class data mining, IEEE Transactions on Knowledge and Data Engineering 15(6) (2003) 1437–1447.

[32] Y. Saeys, I. Inza, P. Larranaga, A review of feature selection techniques in bioinformatics, Bioinformatics 23(19) (2007) 2507–2517.

[33] S. Alelyani, Z. Zhao, H. Liu, A Dilemma in Assessing Stability of Feature Selection Algorithms, in: IEEE 13th International Conference on High Performance Computing and Communications, IEEE, 2011, pp. 701 – 707.

[34] P. Somol, J. Novovičová, Evaluating stability and comparing output of feature selectors that optimize feature subset cardinality, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (2010) 1921–1939.

[35] D. Dernoncourt, B. Hanczar, J.D. Zucker, Analysis of feature selection stability on high dimension and small sample data, Computational Statistics and Data Analysis 71 (2014) 681–693.

[36] G. Stiglic, P. Kokol, Stability of Ranked Gene Lists in Large Microarray Analysis Studies, Journal of Biomedicine and Biotechnology 2010 (2010), Article ID 616358.

[37] D. Dittman, T.M. Khoshgoftaar, R. Wald, H. Wang, Stability Analysis of Feature Ranking Techniques on Biological Datasets, in: 2011 IEEE International Conference on Bioinformatics and Biomedicine, 2011, pp. 252 – 256.

[38] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, An ensemble of filters and classifiers for microarray data classification, Pattern recognition 45(1) (2012) 531-539.

[39] P. Moulos, I. Kanaris, G. Bontempi, Stability of Feature Selection Algorithms for Classification in High-Throughput Genomics Datasets, in: 2013 IEEE 13th International Conference on Bioinformatics and Bioengineering (BIBE), 2013, pp 1–4.

[40] N. Dessì, E. Pascariello, B. Pes, A Comparative Analysis of Biomarker Selection Techniques, BioMed Research International 2013 (2013), Article ID 387673.

[41] S. Loscalzo, L. Yu, C. Ding, Consensus group stable feature selection, in: Proceeding of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, 2009, pp. 567–576.

[42] T. Sanavia, F. Aiolli, G. Da San Martino, A. Bisognin, B. Di Camillo, Improving biomarker list stability by integration of biological knowledge in the learning process, BMC Bioinformatics 2012, 13(Suppl 4):S22.

[43] P. Yang, J.W.K. Ho, Y.H. Yang, B.B. Zhou, Gene-gene interaction filtering with ensemble of filters, BMC Bioinformatics 2011, 12(Suppl 1):S10.

[44] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Data classification using an ensemble of filters, Neurocomputing 135 (2014) 13–20.

[45] L. Rokach, B. Chizi, O. Maimon, A methodology for improving the performance of non-ranker feature selection filters, International Journal of Pattern Recognition and Artificial Intelligence 21(05) (2007) 809-830.

[46] A. Boucheham, M. Batouche, Massively parallel feature selection based on ensemble of filters and multiple robust consensus functions for cancer gene identification, in: Intelligent Systems in Science and Information, Springer International Publishing, 2015, pp 93-108.

[47] D. Dittman, T.M. Khoshgoftaar, R. Wald, A. Napolitano, Comparing Two New Gene Selection Ensemble Approaches with the Commonly-Used Approach, in: 11th International Conference on Machine Learning and Applications, IEEE, 2012, pp. 184 – 191.

[48] R. Wald, T.M. Khoshgoftaar, D. Dittman, W. Awada, A. Napolitano, An Extensive Comparison of Feature Ranking Aggregation Techniques in Bioinformatics, in: IEEE 13th International Conference on Information Reuse and Integration, IEEE, 2012, pp. 377–384.

[49] D. Guan, W. Yuan, Y.K. Lee, K. Najeebullah, M.K. Rasel, A review of ensemble learning based feature selection, IETE Technical Review 31(3) (2014) 190-198.

[50] H. Wang, T.M. Khoshgoftaar, A. Napolitano, Software measurement data reduction using ensemble techniques, Neurocomputing 92 (2012) 124–132.

[51] J. Xu, L. Sun, Y. Gao, T. Xu, An ensemble feature selection technique for cancer recognition, Bio-Medical Materials and Engineering 24(1) (2014) 1001-1008.

[52] B. Seijo-Pardo, V. Bolón-Canedo, I. Porto-Díaz, A. Alonso-Betanzos, Ensemble Feature Selection for Rankings of Features, in: Advances in Computational Intelligence, Lecture Notes in Computer Science, vol. 9095, Springer International Publishing, 2015, pp 29-42.

[53] T. Latkowski, S. Osowski, Data mining for feature selection in gene expression autism data. Expert Systems with Applications, 42 (2015) 864–872.

[54] D. Dittman, T.M. Khoshgoftaar, R. Wald, A. Napolitano, Determining the Number of Iterations Appropriate for Ensemble Gene Selection on Microarray Data, in: 11th International Conference on Machine Learning and Applications, IEEE, 2012, pp. 82 – 89.

[55] P. Yang, B.B. Zhou, J.Y. Yang, A.Y. Zomaya, Stability of Feature Selection Algorithms and Ensemble Feature Selection Methods in Bioinformatics, in: Biological Knowledge Discovery Handbook: Preprocessing, Mining, and Postprocessing of Biological Data, John Wiley & Sons, 2014.

[56] U.M. Braga-Neto, E.R. Dougherty, Is cross-validation valid for small-sample microarray classification?, Bioinformatics 20(3) (2004) 374–380.

[57] L.I. Kuncheva, A Stability Index for Feature Selection, in: Proceedings of the 25th IASTED International Multi-Conference: artificial intelligence and applications, ACTA Press, Anaheim, CA, USA, 2007, pp. 390-395.

[58] L.M. Cannas, N. Dessì, B. Pes, Assessing similarity of feature selection techniques in high-dimensional domains, Pattern Recognition Letters 34(12), 2013, pp. 1446–1453.

[59] D. Dittman, T.M. Khoshgoftaar, R. Wald, A. Napolitano, Similarity analysis of feature ranking techniques on imbalanced DNA microarray datasets, in: Proceedings of the 2012 IEEE international conference on bioinformatics and biomedicine, 2012, pp. 1–5.

[60] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, A.J. Levine, Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays, PNAS 96(12) (1999) 6745-6750.

[61] M.A. Shipp, K.N. Ross, P. Tamayo, A.P. Weng, J.L. Kutok, R.C. Aguiar, M. Gaasenbeek, et al., Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning, Nature Medicine 8(1) (2002) 68–74.

[62] D. Singh, P.G. Febbo, K. Ross, D.G. Jackson, J. Manola, C. Ladd, P. Tamayo, et al., Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 1 (2) (2002) 203–209.

[63] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, et al., Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science 286 (5439) (1999) 531–537.

[64] L. van't Veer, H. Dai, M. Van De Vijver, Y. He, A. Hart, M. Mao, H. Peterse, K. van der Kooy, M. Marton, A. Witteveen, et al., Gene expression profiling predicts clinical outcome of breast cancer, Nature 415 (6871) (2002) 530–536.

[65] A.A. Alizadeh, M.B. Eisen, R.E. Davis, C. Ma, I.S. Lossos, A. Rosenwald, J.C. Boldrick, H. Sabet, T. Tran, X. Yu, et al., Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, Nature 403 (2000) 503–511.

[66] T. Li, C. Zhang, M. Ogihara, A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression, Bioinformatics 20 (2004) 2429–2437.

[67] S.A. Armstrong, J.E. Staunton, L.B. Silverman, R. Pieters, M.L. den Boer, M.D. Minden, S.E. Sallan, E.S. Lander, T.R. Golub, S.J. Korsmeyer, MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. Nat. Genet. 30 (2002) 41–47.

[68] J. Khan, J. Wei, M. Ringner, L. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. Antonescu, C. Peterson, et al., Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks, Nat. Med. 7 (6) (2001) 673–679.

[69] A. Bhattacharjee, W.G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, et al., Classification of Human Lung Carcinomas by mRNA Expression Profiling Reveals Distinct Adenocarcinoma Subclasses, PNAS 98(24) (2001) 13790-13795.

[70] I.H. Witten, E. Frank, M.A. Hall, Data mining: Practical machine learning tools and techniques, 3rd ed., Morgan Kaufmann, San Francisco, 2011.

[71] H. Liu, R. Setiono, Chi2: Feature selection and discretization of numeric attributes, in: Proceedings of the 7th international conference on tools with artificial intelligence, IEEE, 1995, pp. 338–391.

[72] R.C. Holte, Very simple classification rules perform well on most commonly used datasets, Machine Learning 11 (1993) 63–91.

[73] M. Robnik-Sikonja, I. Kononenko, Theoretical and empirical analysis of relieff and rrelieff, Machine Learning 53 (2003) 23–69.

[74] A. Rakotomamonjy, Variable selection using SVM based criteria, Journal of Machine Learning Research 3 (2003) 1357–1370.

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

[76] R. Wald, T.M. Khoshgoftaar, D. Dittman, Mean Aggregation versus Robust Rank Aggregation for Ensemble Gene Selection, in: 11th International Conference on Machine Learning and Applications, IEEE, 2012, pp. 63–69.

[77] V.N. Vapnik, Statistical learning theory, Wiley, New York, 1998.

[78] L. Breiman, Random forests, Machine Learning 45 (2001) 5–32.

[79] L. Rokach, Decision forest: Twenty years of research, Information Fusion 27 (2016) 111–125.

[80] A. Statnikov, L. Wang, C.F. Aliferis, A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification, BMC Bioinformatics 2008, 9:319.

[81] R. Díaz-Uriarte, S.A. de Andrés, Gene selection and classification of microarray data using random forest, BMC Bioinformatics 7:3 (2006).

[82] R.R. Bouckaert, E. Frank, M.A. Hall, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, WEKA – Experiences with a java open-source project, Journal of Machine Learning Research 11 (2010) 2533–2541.

[83] J. Ye, T. Li, T. Xiong, R. Janardan, Using uncorrelated discriminant analysis for tissue classification with gene expression data, IEEE/ACM Trans Comput Biol Bioinform 1(4) (2004) 181–90.

[84] V. Bolón-Canedo, L. Morán-Fernández, A. Alonso-Betanzos, An insight on complexity measures and classification in microarray data, in: 2015 International Joint Conference on Neural Networks (IJCNN), July 2015, IEEE, pp. 1-8.

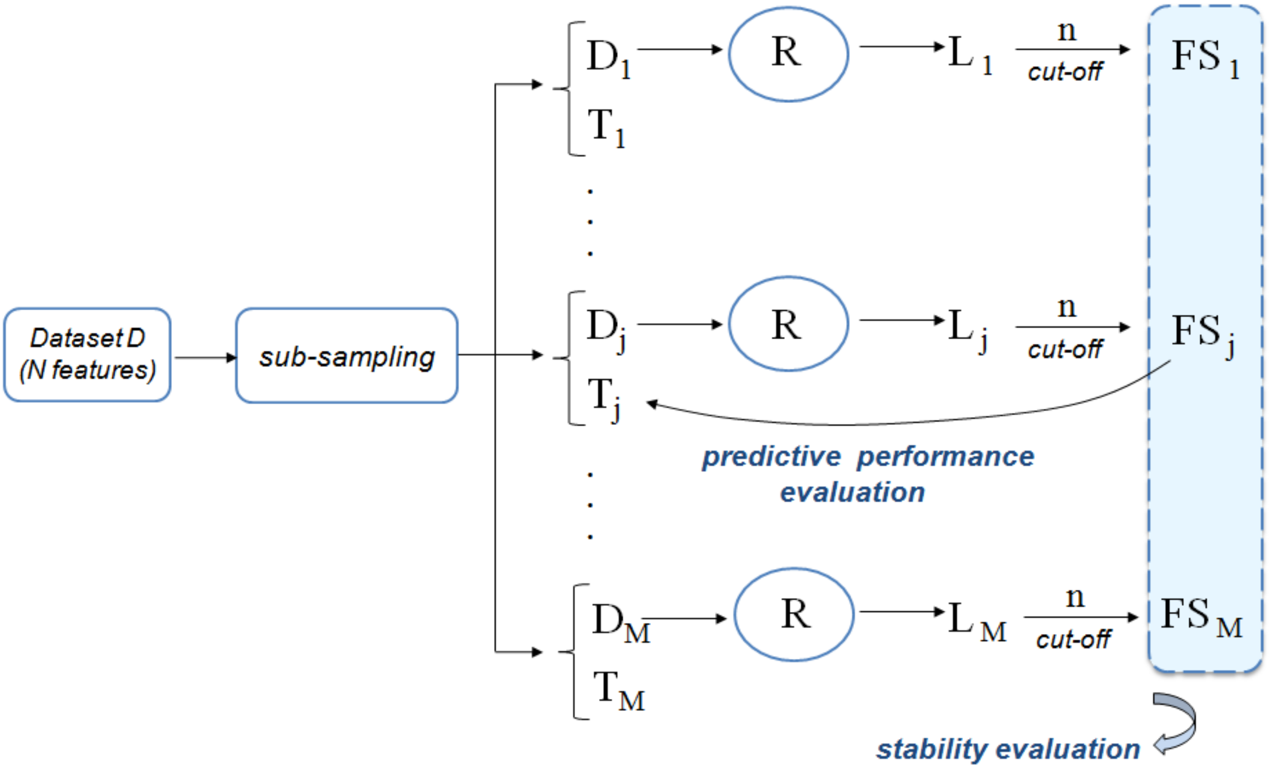

Fig. 1. *Simple ranking: joint evaluation of stability and predictive performance*

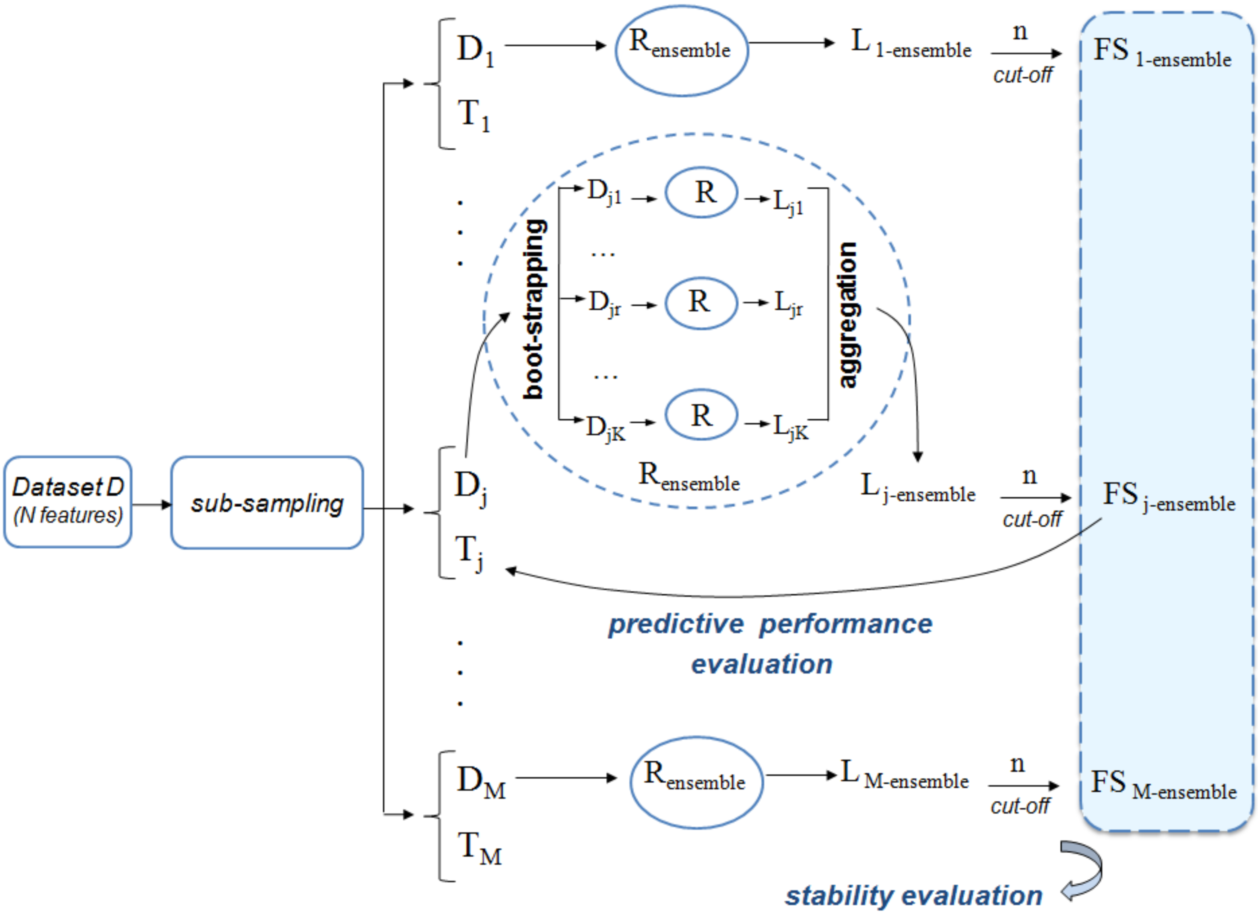

Fig. 2. *Ensemble ranking: joint evaluation of stability and predictive performance*

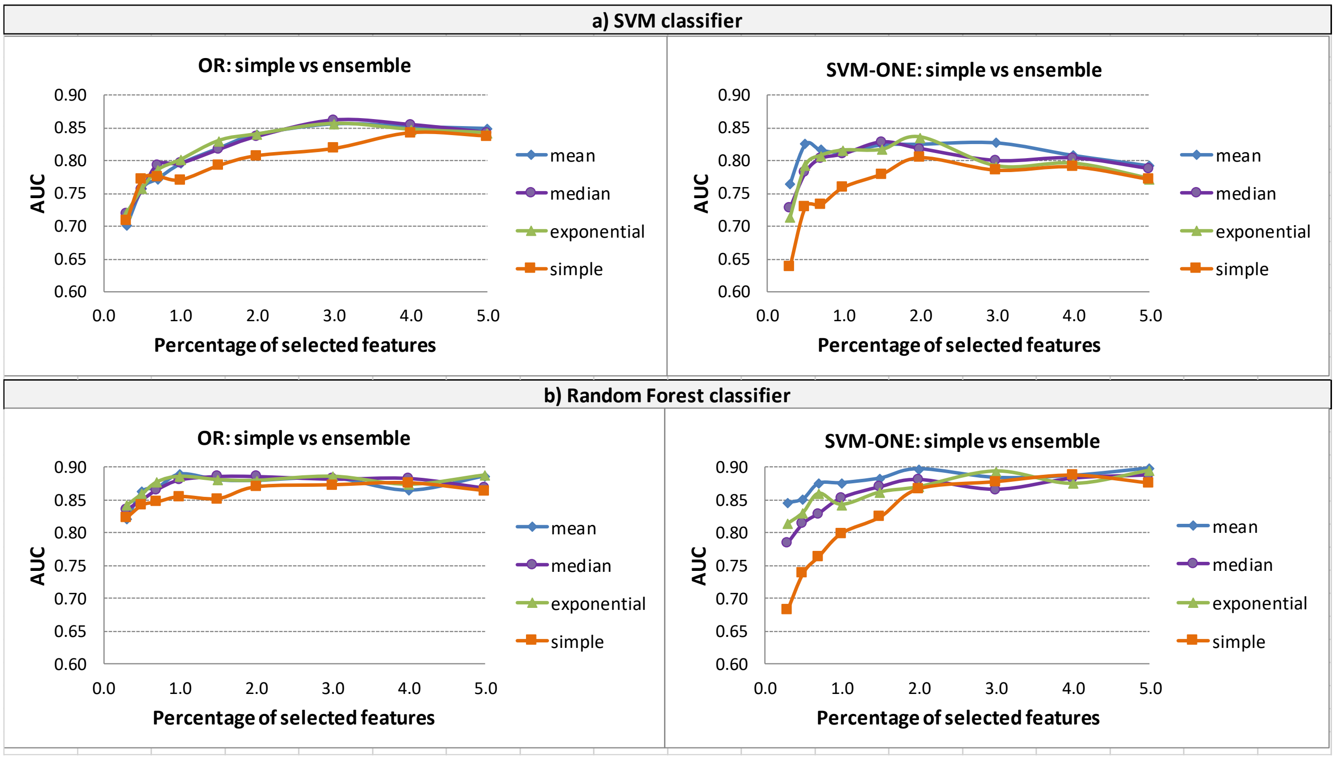

Fig.3. *AUC patterns on the Colon dataset, for both SVM (a) and Random Forest (b) classifiers. The performance of OR (left) and SVM-ONE (right) methods is evaluated in the simple and in the ensemble implementation. For both methods, three ensemble versions are considered, i.e. mean, median and exponential.*

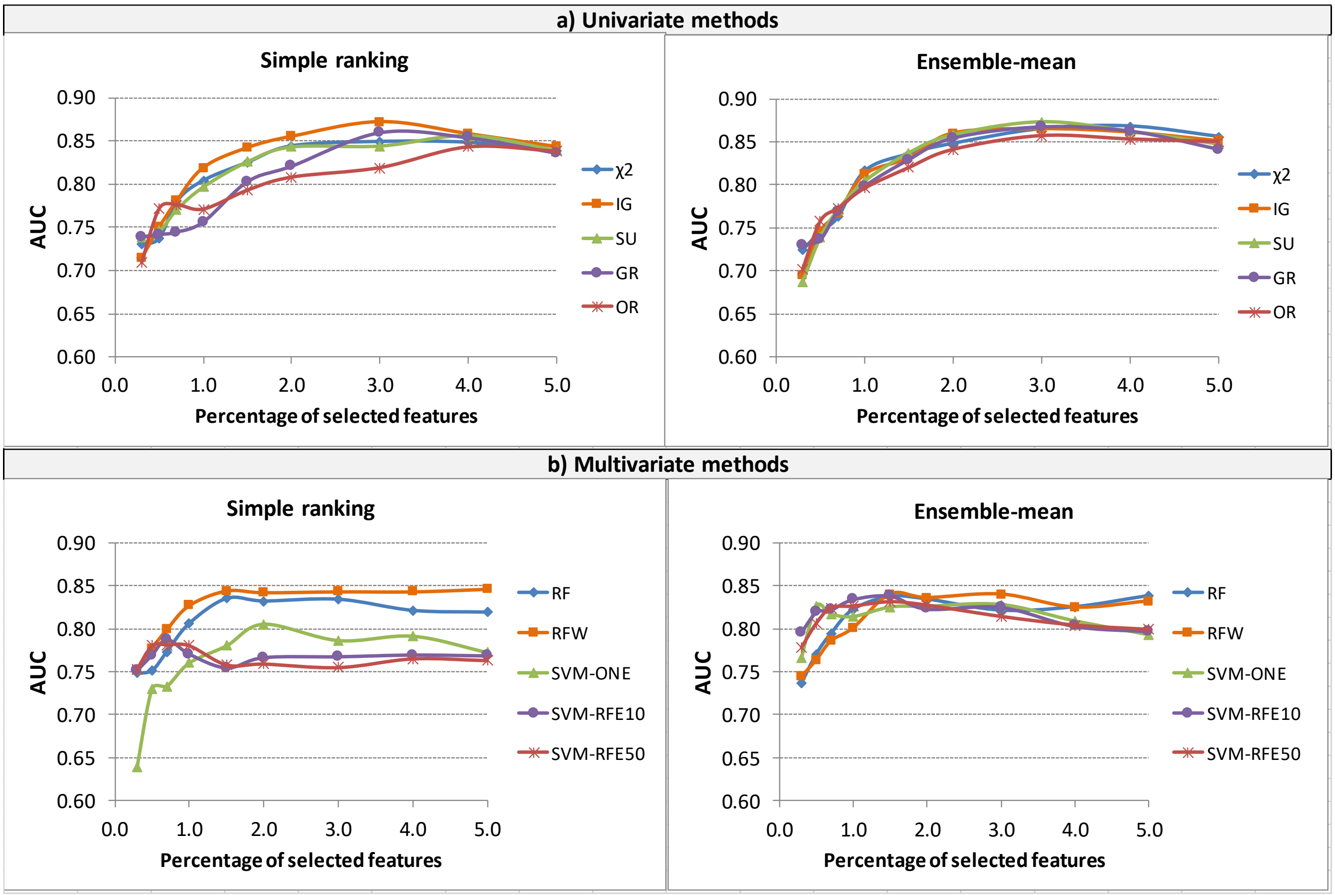

***Fig. 4**. AUC patterns on the Colon dataset using the SVM classifier. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

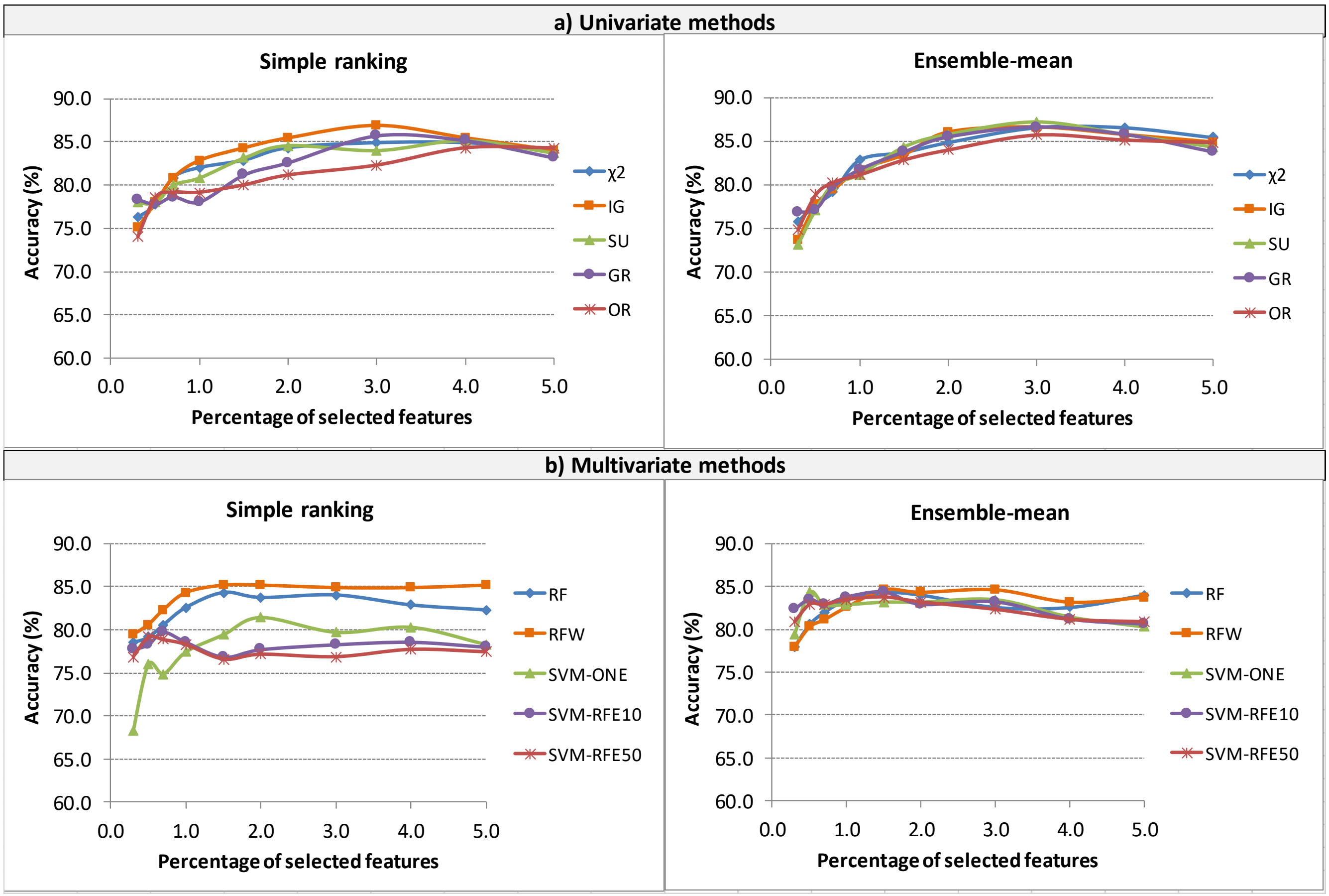

Fig. 5. *Accuracy patterns on the Colon dataset using the SVM classifier. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

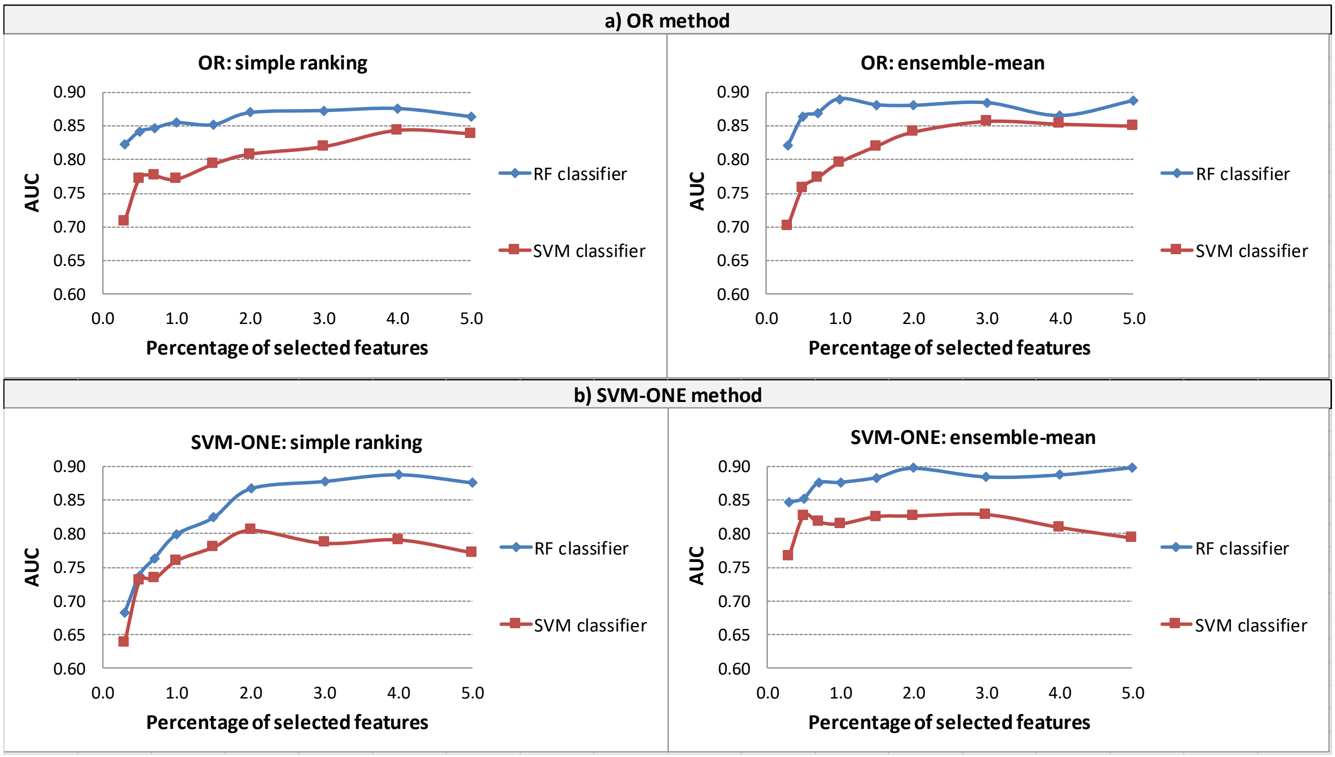

Fig. 6. *AUC patterns on the Colon dataset. Comparison between SVM and Random Forest classifiers using OR (a) and SVM-ONE (b) methods in their simple (left) and ensemble-mean (right) version.*

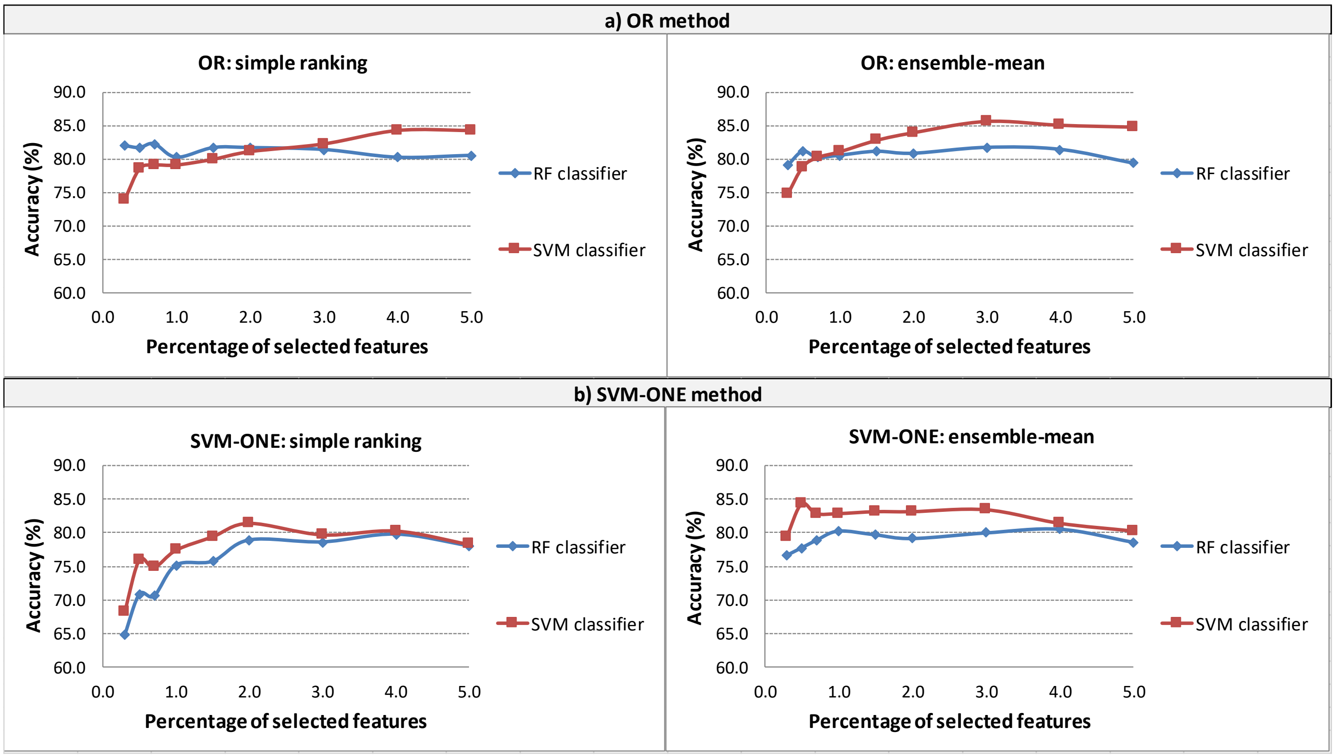

Fig. 7. Accuracy patterns on the Colon dataset. Comparison between SVM and Random Forest classifiers using OR (a) and SVM-ONE (b) methods in their simple (left) and ensemble-mean (right) version.

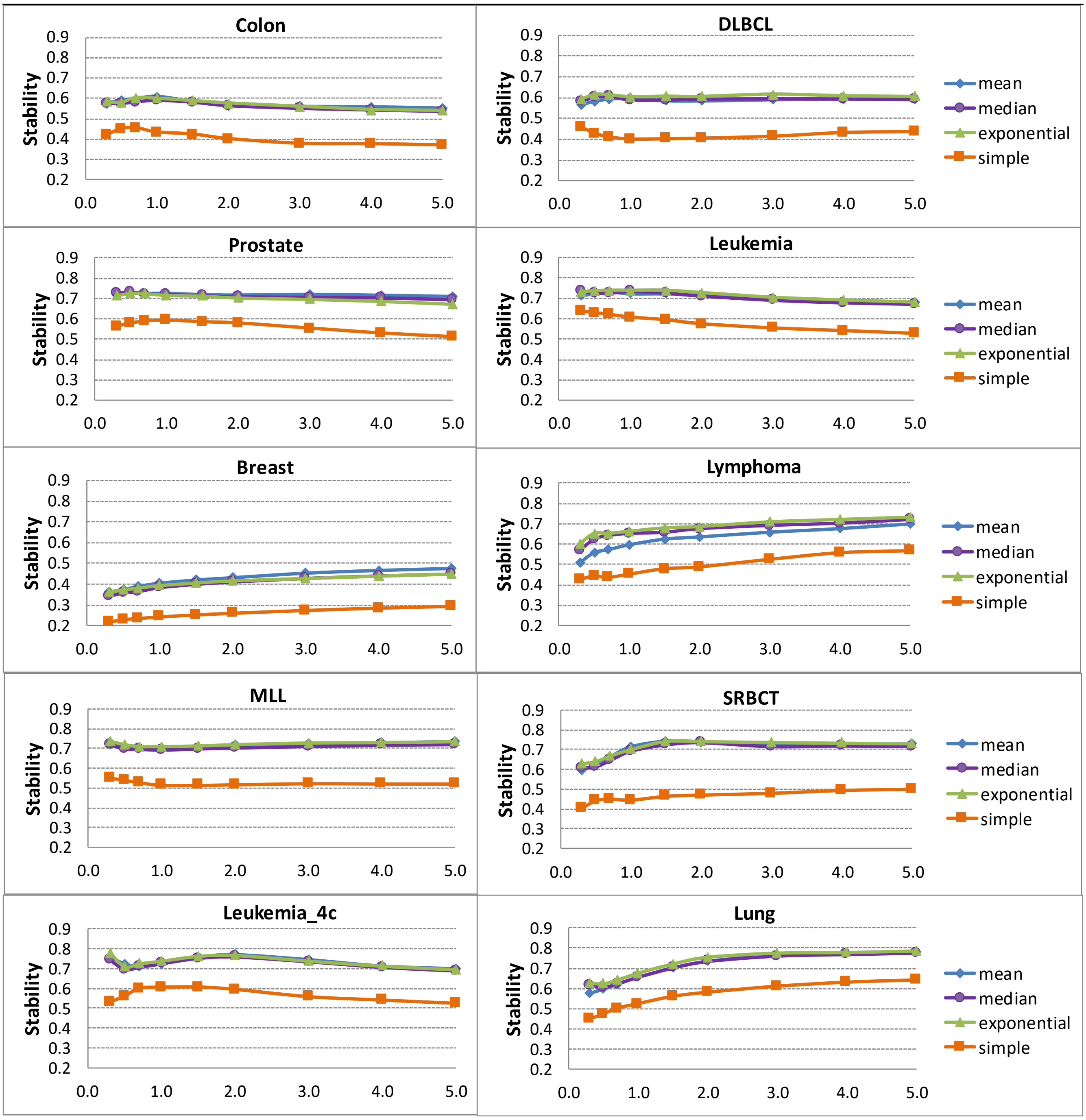

Fig. 8. *Stability patterns for the OR method, in its simple and ensemble version. For each dataset, three ensemble versions are considered, i.e. mean, median and exponential.*

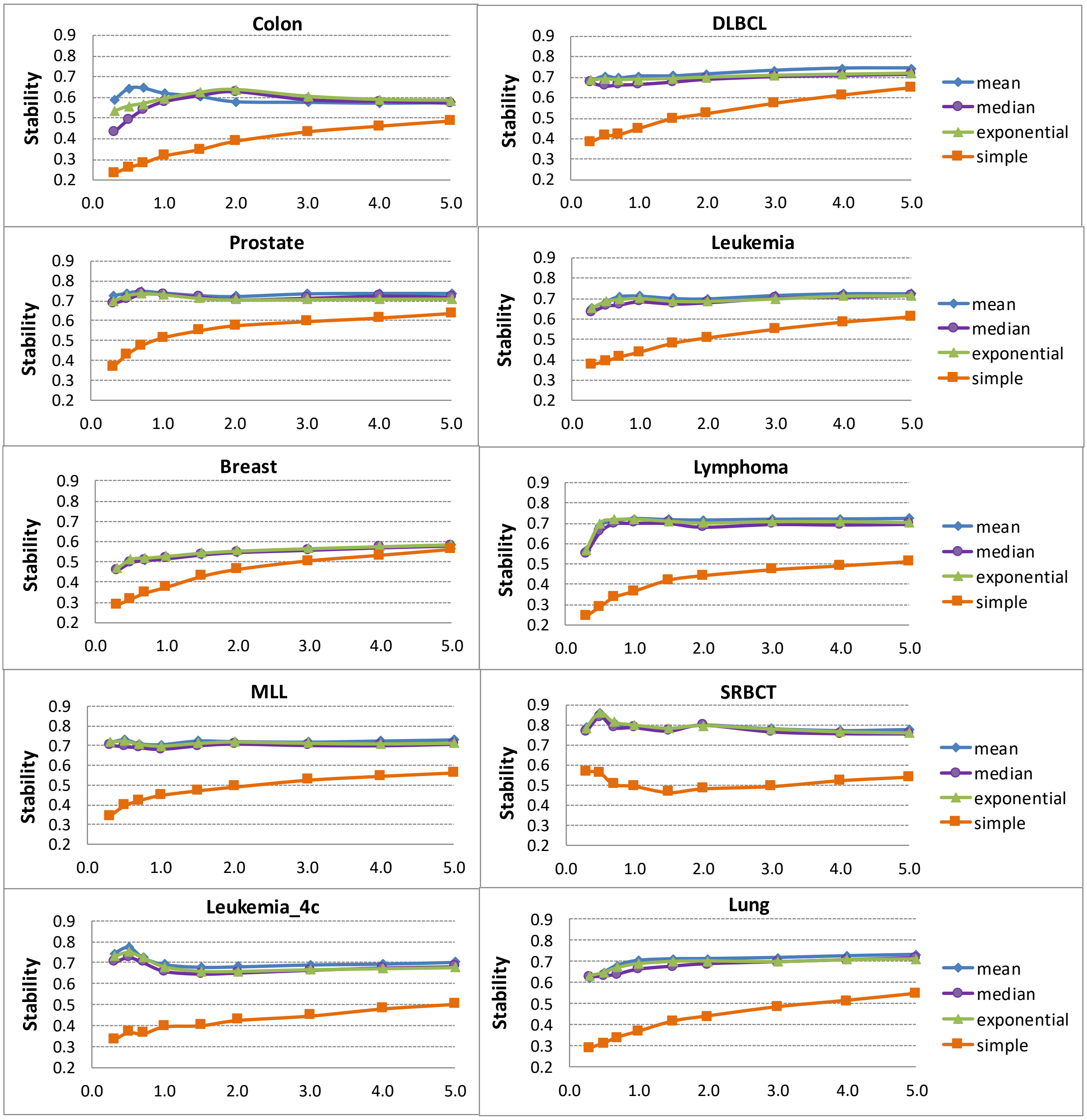

Fig. 9. *Stability patterns for the SVM-RFE50 method, in its simple and ensemble version. For each dataset, three ensemble versions are considered, i.e. mean, median and exponential.*

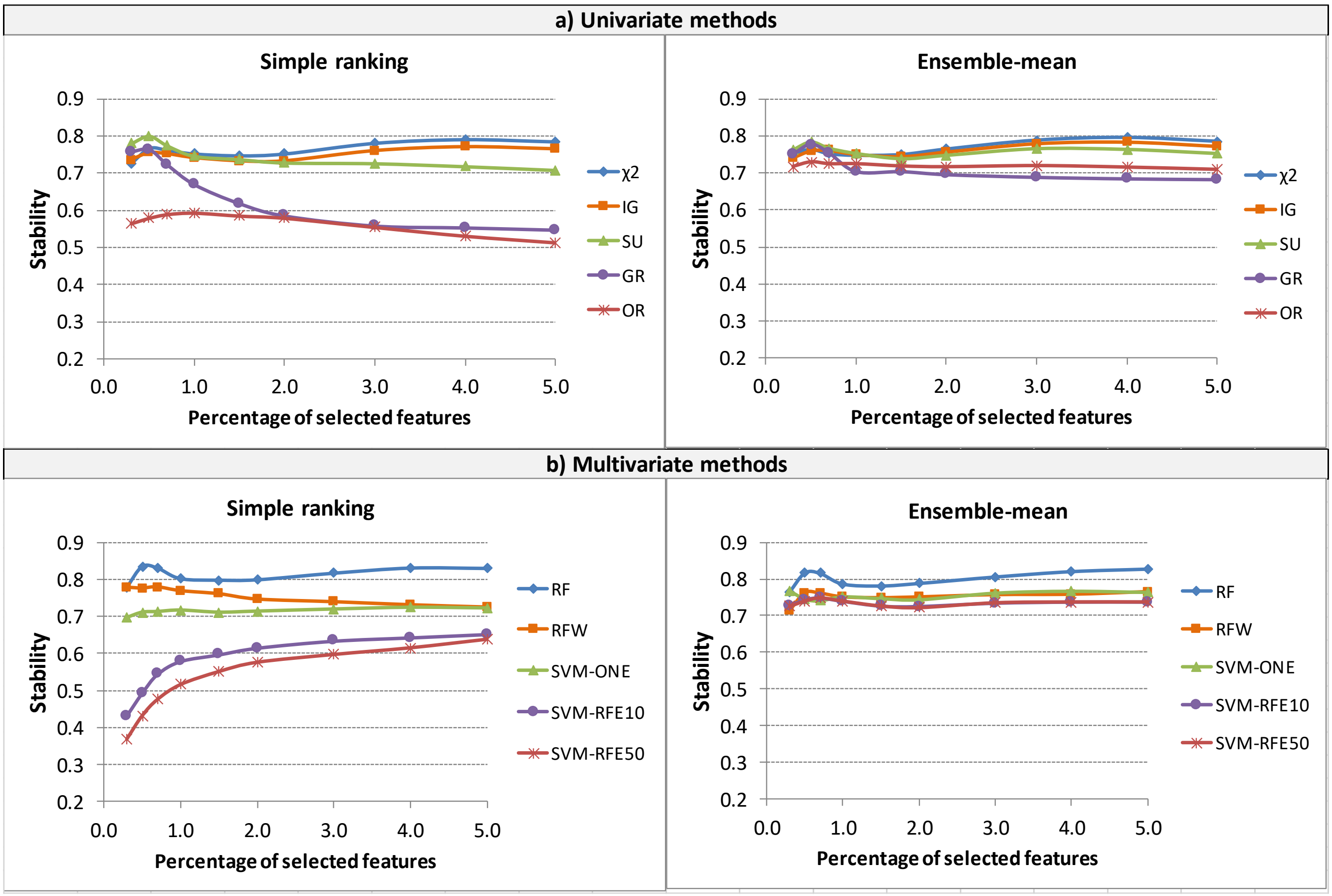

Fig. 10. *Stability patterns on the Prostate dataset. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

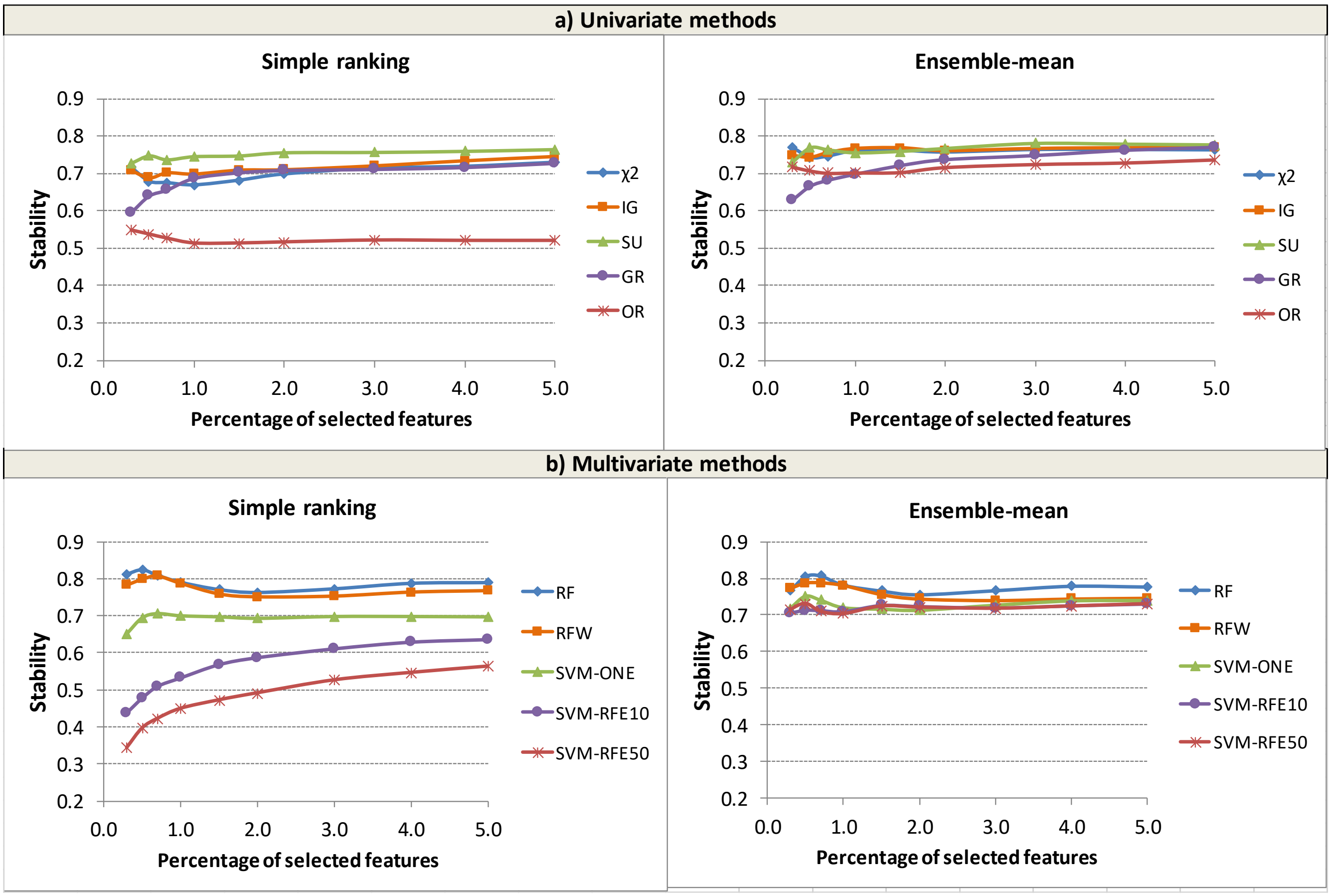

***Fig. 11**. Stability patterns on the MLL dataset. Both the univariate (a) and the multivariate (b) methods are evaluated in their simple (left) and ensemble-mean (right) version.*

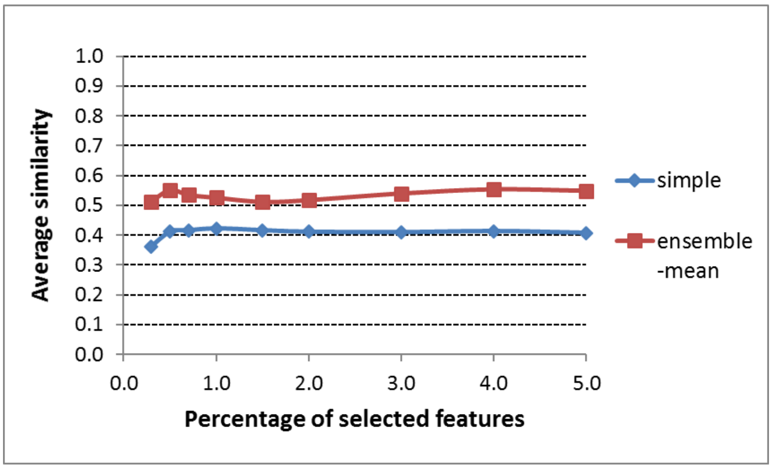

Fig. 12. *Prostate dataset: average inter-method similarity for feature subsets of increasing size.*

Table 1. *Micro-array datasets used in the experiments.*

Dataset	***Number of features***	***Number of instances***	***Number of classes***	***Refs***
Colon	2000	62	2	[60]
DLBCL	7129	77	2	[61]
Prostate	12600	102	2	[62]
Leukemia	7129	72	2	[63]
Breast	24481	97	2	[64]
Lymphoma	4026	66	3	[65, 66]
MLL	12582	72	3	[67]
SRBCT	2308	83	4	[68]
Leukemia_4c	7129	72	4	[63, 66]
Lung	12600	203	5	[69]

Table 2*. DLBCL dataset: intra-method similarity matrices for the univariate methods IG, GR and OR.*

	IG-mean	**IG-median**	**IG-exponential**	**IG-simple**
IG-mean	-	0.92	0.94	0.82
IG-median	0.92	-	0.96	0.82
IG-exponential	0.94	0.96	-	0.82
IG-simple	0.82	0.82	0.82	-

	GR-mean	**GR-median**	**GR-exponential**	**GR-simple**
GR-mean	-	0.79	0.83	0.47
GR-median	0.79	-	0.90	0.61
GR-exponential	0.83	0.90	-	0.55
GR-simple	0.47	0.61	0.55	-

	OR-mean	**OR-median**	**OR-exponential**	**OR-simple**
OR-mean	-	0.82	0.86	0.49
OR-median	0.82	-	0.89	0.52
OR-exponential	0.86	0.89	-	0.49
OR-simple	0.49	0.52	0.49	-

Table 3*. DLBCL dataset: intra-method similarity matrices for the multivariate methods RF, SVM-RFE10 and SVM-RFE50.*

	RF-mean	**RF-median**	**RF-exponential**	**RF-simple**
RF-mean	-	0.87	0.93	0.89
RF-median	0.87	-	0.93	0.94
RF-exponential	0.93	0.93	-	0.93
RF-simple	0.89	0.94	0.93	-

	SVM-RFE10 -mean	**SVM-RFE10 -median**	**SVM-RFE10 -exponential**	**SVM-RFE10 -simple**
SVM-RFE10-mean	-	0.85	0.89	0.55
SVM-RFE10-median	0.85	-	0.93	0.64
SVM-RFE10-exponential	0.89	0.93	-	0.62
SVM-RFE10-simple	0.55	0.64	0.62	-

	SVM-RFE50 -mean	**SVM-RFE50 -median**	**SVM-RFE50 -exponential**	**SVM-RFE50 -simple**
SVM-RFE50-mean	-	0.82	0.89	0.54
SVM-RFE50-median	0.82	-	0.93	0.57
SVM-RFE50-exponential	0.89	0.93	-	0.54
SVM-RFE50-simple	0.54	0.57	0.54	-

Table 4. *Prostate dataset: inter-method similarity among the simple rankers.*

	χ^2	IG	SU	GR	OR	RF	RFW	SVM-RFE10	SVM-RFE50	SVM-ONE
χ^2	-	0.92	0.81	0.58	0.70	0.53	0.45	0.14	0.18	0.20
IG	0.92	-	0.85	0.63	0.66	0.53	0.44	0.15	0.19	0.20
SU	0.81	0.85	-	0.74	0.62	0.53	0.44	0.17	0.20	0.21
GR	0.58	0.63	0.74	-	0.50	0.45	0.36	0.19	0.22	0.24
OR	0.70	0.66	0.62	0.50	-	0.45	0.38	0.16	0.21	0.18
RF	0.53	0.53	0.53	0.45	0.45	-	0.76	0.25	0.31	0.34
RFW	0.45	0.44	0.44	0.36	0.38	0.76	-	0.27	0.31	0.37
SVM-RFE10	0.14	0.15	0.17	0.19	0.16	0.25	0.27	-	0.74	0.62
SVM-RFE50	0.18	0.19	0.20	0.22	0.21	0.31	0.31	0.74	-	0.62
SVM-ONE	0.20	0.20	0.21	0.24	0.18	0.34	0.37	0.62	0.62	-

Table 5. *Prostate dataset: inter-method similarity among the ensemble rankers.*

	χ^2	IG	SU	GR	OR	RF	RFW	SVM-RFE10	SVM-RFE50	SVM-ONE
χ^2	-	0.91	0.86	0.78	0.83	0.52	0.37	0.31	0.30	0.32
IG	0.91	-	0.92	0.83	0.79	0.52	0.37	0.32	0.31	0.32
SU	0.86	0.92	-	0.90	0.78	0.49	0.35	0.32	0.31	0.32
GR	0.78	0.83	0.90	-	0.73	0.49	0.35	0.32	0.31	0.33
OR	0.83	0.79	0.78	0.73	-	0.51	0.37	0.30	0.29	0.30
RF	0.52	0.52	0.49	0.49	0.51	-	0.72	0.43	0.42	0.45
RFW	0.37	0.37	0.35	0.35	0.37	0.72	-	0.49	0.50	0.54
SVM-RFE10	0.31	0.32	0.32	0.32	0.30	0.43	0.49	-	0.98	0.89
SVM-RFE50	0.30	0.31	0.31	0.31	0.29	0.42	0.50	0.98	-	0.89
SVM-ONE	0.32	0.32	0.32	0.33	0.30	0.45	0.54	0.89	0.89	-