Neurocomputing 470 (2022) 365–375

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

How important is data quality? Best classifiers vs best features

Laura Morán-Fernández *, Verónica Bólon-Canedo, Amparo Alonso-Betanzos

CITIC, Universidade da Coruña, A Coruña, Spain

ARTICLE INFO

Article history:
Received 5 January 2021
Revised 7 April 2021
Accepted 3 May 2021
Available online 24 July 2021

Keywords:
Feature selection
Filters
Preprocessing
High dimensionality
Classification
Data analysis

ABSTRACT

The task of choosing the appropriate classifier for a given scenario is not an easy-to-solve question. First, there is an increasingly high number of algorithms available belonging to different families. And also there is a lack of methodologies that can help on recommending in advance a given family of algorithms for a certain type of datasets. Besides, most of these classification algorithms exhibit a degradation in the performance when faced with datasets containing irrelevant and/or redundant features. In this work we analyze the impact of feature selection in classification over several synthetic and real datasets. The experimental results obtained show that the significance of selecting a classifier decreases after applying an appropriate preprocessing step and, not only this alleviates the choice, but it also improves the results in almost all the datasets tested.

1. Introduction

Classification is a type of supervised learning that aims at extracting models from a set of training data that are able to identify to which set of predefined categories a new test data belongs. This task has many applications and, together with clustering, is an example of the more general problem of pattern recognition, being essential to data analytics and machine learning. The data instances (e.g. a patient potentially having cancer) used by the classification algorithms for learning are described by feature vectors of measurable properties of the instance (e.g., several tumor markers can be substances found in the blood, urine, stool, other bodily fluids, or tissues of the patient). The model constructed aims at making predictions of the class variable known as the class (e.g., the patient has/has not cancer) using one or more of the other features, which can be either categorical or numeric. To summarize, classification aims to learn the relationship between a set of feature variables and a target variable of interest, also named response variable, e.g. whether the patient has a benign or a malignant tumor.

There are a number of classification models that can be used for a given problem, such as logistic regression, decision trees, support vector machines, random forest, multilayer perceptrons or Naive-Bayes to name just a few (see [2,1,31]). Nevertheless, for a given dataset there are only a few indications that one can follow to know in advance which classifier will obtain the best results. There have been several works that have studied the relationship between the performance of several classifiers and the complexity measures of the datasets used (overlapping of the classes, shape of the decision boundary, linear separability, etc.), such as in [18,33,30]. However, as each complexity measure by definition also depends on the characteristics of the dataset as the number of features and instances, it may well happen that two datasets with very different characteristics can present the same metric value, and when the same classifier is used for both, the accuracies obtained are not related at all. Thus, the common way to proceed is to test several classifiers over a suite of several datasets, in order to select the ones that behave the best. Still, if the collection of datasets varies (is enlarged or reduced), the best classifier might change, and this is the basis of the No-Free-Lunch theorem, that is, the best classifier will not be the same for all the datasets [44]. Aiming at shedding some light in this problem, the authors in Fernández-Delgado et al. [12] carried out an exhaustive evaluation in which 179 classifiers from 17 different families were tested over a large collection of 121 datasets, with different sample sizes, number of features and number of classes. They concluded that the classifiers most likely to obtain the best results were Random Forest and Support Vector Machines. Later on, another study by Wainberg et al. [43] showed that the results obtained by the previous authors Fernández-Delgado et al. [12] were biased by the lack of a held-out test set and the exclusion of trials with errors, calling into question that conclusion.

In this paper, we aim at a different direction. Our goal is to try to establish if the application of an adequate preprocessing step using

* Corresponding author.
E-mail addresses: laura.moranf@udc.es (L. Morán-Fernández), veronica.bolon@udc.es (V. Bólon-Canedo), ciamparo@udc.es (A. Alonso-Betanzos).

https://doi.org/10.1016/j.neucom.2021.05.107
0925-2312/

feature selection might alleviate the decision on which classifier is the most appropriate. In other words, we would like to check if the use of the appropriate features would cause different classifiers to obtain similar results, as a consequence of working on higher quality data. The rationale of this idea is that the data collected contains usually some level of noise, and feature selection can help removing those noisy and irrelevant features [4] and help classifiers to obtain better results. To show just a few examples, decision trees, such as C4.5, or instance-based methods, such as *k*NN, degrade their performance when faced with many irrelevant features. In Langley and Iba [27] the authors showed that the number of training samples needed to produce a predetermined level of performance for instance-based learning increases exponentially with the number of irrelevant features. Nevertheless, algorithms such as Naive Bayes are robust with respect to irrelevant features, degrading their performance very slowly when more irrelevant features are added [24]. However, their performance deteriorates quickly when redundant features are added, even if they are relevant to the concept. Thus, researchers use feature selection methods to reduce the number of input variables of the dataset with the aim of retaining those most useful for the model in order to accurately predict the target variable [13]. Hence a new question arises, regarding the effect that feature selection has over classification in the sense that if the application of an adequate preprocessing step using feature selection can attenuate the relevance of the decision of selecting the best classification algorithm. In order to do this, we have studied the impact of the process over the accuracy of the classifier using a suite of 10 synthetic and 30 real datasets.

The rest of the paper is organized as follows: Section 2 provides some information on relevant previous works on feature selection and their influence on classification, emphasizing the main aims of our study. Sections 3 and 4 provide the description of the different feature selection techniques and the description of classifiers and the synthetic and real datasets employed in the study, respectively. Section 5 details the experimental study carried out over several real and synthetic datasets and the results obtained, including several case studies in order to analyze whether parameter tuning affects the results obtained, as well as if our conclusions hold when different classifiers (beside the ones already employed in the experiments) are tested. Finally, Section 6 contains our concluding remarks and proposals for future research.

2. Background

The first research works in feature selection date back to the 1960 [20]. It was in the 1990s when notable advances were made in the field with the aim of solving machine learning problems and, nowadays, it is acknowledged to play a crucial role in reducing the dimensionality of real problems, with the added benefit of enhancing interpretability [5]. Feature selection has attracted interest in processes such as clustering or regression, but most of the published works are related to classification problems.

The success of feature selection applications is mainly based on the benefits that it implies for classification problems, since sometimes using less features improves the classification performance. Several works have reviewed the most widely used feature selection methods in the last years. Molina et al. [32] assessed the performance of fundamental feature selection algorithms in a controlled scenario, taking into account dataset relevance, irrelevance and redundancy. Saeys et al. [38] created a basic taxonomy of classical feature selection techniques, discussing their use in bioinformatics applications, as well as Bolón-Canedo et al. [6] which reviewed feature selection for microarray data. Other works evaluate the performance of feature selection methods on synthetic data, such as [19,4]. Brown et al. [9] presented a unifying framework for information theoretic feature selection, bringing almost two decades of research into heuristic filter criteria under a single theoretical umbrella. The advent of Big Data and the necessity of dealing with thousands (or even millions) of features has posed tremendous challenges for feature selection researchers, as studied in [5,46,29].

However, although several works studied the different feature selection methods, their adequacy to be applied to different problems, and even which feature selection methods are more appropriate to use in conjunction with a given classifier, there are no attempts to study if provided that the best features are selected, the choice of classifier is not so critical. Our hypothesis is that, when a classifier is fed with data of enough quality (i.e. after removing the irrelevant or redundant features), there would be a slight difference in the behavior of different state-of-the-art classifiers. If our hypothesis is true, this finding will open the door to put more emphasis on the data curation phase, instead of relying on complex (and often highly computationally expensive) classifiers.

3. Feature selection techniques

Feature selection methods have received a great deal of attention in the classification literature. Broadly, they can be divided into [14]: (i) filters, which are independent of the induction algorithm and establish the importance of the features by using metrics such as mutual information or statistics such as Chi^2; (ii) wrappers, which use the induction algorithm accuracy to determine the importance of subsets of features; and (iii) embedded methods, which perform the selection of features during the training process of the induction algorithm. In addition to this, feature selection methods can also be divided into univariate methods (when they compute the relevance of a single feature with respect to the predictive class); and multivariate (when they take into account the interactions among subsets of features). Since wrapper and embedded methods interact with the classifier, we opted for filter methods. Besides, filter methods are a common choice in the new Big Data scenario, mainly due to their low computational cost compared to wrapper or embedded methods.

Filter methods evaluate the goodness of data subsets by observing only intrinsic data characteristics and evaluating a single feature or subset against the class label. Below we describe the seven filters used in our experimental study, where two of the feature selection methods are univariate (Mutual Information Maximisation and Information Gain) and the other five (Correlation-based Feature Selection, INTERACT, ReliefF, Joint Mutual Information and minimum Redundancy Maximum Relevance) are multivariate methods:

- **Correlation-based Feature Selection** (CFS) is a simple multivariate filter algorithm that ranks feature subsets according to a correlation-based heuristic evaluation function [16]. This function is biased towards subsets containing features that are highly correlated with the class and uncorrelated with each other. Irrelevant features with low correlation with the class are ignored. Redundant features are screened out as they would be highly correlated with one or more of the remaining features.
- The **INTERACT** (INT) algorithm is based on symmetrical uncertainty and it also includes the consistency contribution [47]. It consists of two main parts. In the first part, the features are ranked in descending order based on their symmetrical uncertainty values. In the second part, features are evaluated one by one starting from the end of the feature ranking. If the consistency contribution of a feature is less than an established threshold, the feature is removed, otherwise it is selected.

- **Information Gain** (IG) filter evaluates the features according to their information gain and considers a single feature at a time [17]. It provides an orderly classification of all features, and then a threshold is required to select a certain number of them according to the order obtained.
- **ReliefF** algorithm (RelF) [25], an extension of the original Relief [23], adds the ability of dealing with noisy, incomplete and multiclass datasets. The key idea of this algorithm is to estimate features according to how well their values distinguish among the instances that are near to each other. This method may be applied in all situations, has low bias, includes iteration among features and may capture local dependencies which other methods miss.
- **Mutual Information Maximisation** (MIM) [28] ranks the features by their mutual information score, and selects the top k features, where k is decided by some predefined need for a certain number of features or some other stopping criterion.
- **Joint Mutual Information** (JMI) [45] is another feature selection method based on mutual information, and it adopts a new criterion to evaluate the candidate features. JMI chooses the feature that has the maximum cumulative summation of joint mutual information with the selected features in each step and adds it to the subset S until the number of selected features reaches k.
- The **minimum Redundancy Maximum Relevance** (mRMR) [34] feature selection method selects features that have the highest relevance with the target class and are also minimally redundant, i.e. it selects features that are maximally dissimilar to each other. Both optimization criteria (maximum-relevance and minimum-redundancy) are based on mutual information.

4. Materials and methods

Seven different feature selection methods are tested and compared in this work in order to draw useful conclusions. Their behavior will be tested according to the classification error obtained by five different classifiers over 10 synthetic and 30 real datasets.

4.1. Classifiers

Five different classifiers, each belonging to a different family, were used as tested for analyzing the effects of the seven feature selection algorithms. The classifiers employed were: two linear (naive Bayes and Support Vector Machine using a linear kernel) and three nonlinear (C4.5, k-Nearest Neighbor and Random Forest). All five classifiers were executed using the Weka [15] tool, employing default values for their parameters.

- **Naive Bayes** (NB) is a simple probabilistic classifier [37] based on applying Bayes' theorem with strong (naive) independence assumptions. This classifier assumes that the presence or absence of a particular feature is unrelated to the presence or absence of any other feature, given the class variable.
- **Support Vector Machine** (SVM) is a learning algorithm, used for classification, regression and other tasks, which constructs a hyperplane or set of hyperplanes in a high —or finite— dimensional space [42]. Intuitively, good separation is achieved by the hyperplane with the greatest distance to the nearest training data point of any class, since in general, the larger the margin, the lower the generalization error of the classifier.
- **C4.5** was developed by Quinlan [36] as an extension of the ID3 algorithm (both are based on decision tree concepts). A decision tree classifies a pattern by means of a descending filtering until is found a leaf, that points to the corresponding classification.
- **k-Nearest Neighbor** (k-NN) is a classification strategy that is an example of a "lazy learner" [2]. An object is classified by majority vote of its neighbors and is assigned to the most common class among its k nearest neighbors. In this work, $k = 3$.
- **Random Forest** [7] consists of a combination of tree classifiers where each classifier is generated using a random vector sampled independently from the input vector, and each tree casts a unit vote for the most popular class to classify an input.

4.2. Datasets

In order to evaluate empirically the effect that feature selection has over classification (in the sense that if the application of an adequate preprocessing step using feature selection can attenuate the relevance of the decision of selecting the best classification algorithm), we employed 10 synthetic datasets and 30 real datasets, where 17 of them are microarray datasets. The features within each dataset have a variety of characteristics: some are binary/discrete, and some are continuous. Continuous features were discretized, using an equal-width strategy in 5 bins, while features already with a categorical range were left untouched.

4.2.1. Synthetic datasets

The datasets chosen for this study (Table 1) try to cover different problems: increasing number of irrelevant features, redundancy, noise, alteration of the inputs, nonlinearity of the data etc. These factors complicate the task of the feature selection methods, which are very affected by them. Besides, SD datasets have a significantly higher number of features than samples, which implies an added difficulty for the correct selection of the relevant features.

- **CorrAL-100**. The CorrAL [21] dataset has six binary features (i.e., $f_1, f_2, f_3, f_4, f_5, f_6$) and its class value is $(f_1 \wedge f_2) \vee (f_3 \wedge f_4)$. Feature f_5 is irrelevant and f_6 is correlated to the class label by 75%. The correlated feature is redundant if the four relevant features are selected and, besides, it is correlated with the class label by 75%, so if one applies a classifier after the feature selection process, a 25% of error will be obtained. CorrAL-100 [22] was constructed by adding 93 irrelevant binary features to the previous CorrAL dataset.
- **XOR-100**. XOR-100 [22] has 2 relevant binary features and 97 irrelevant binary features (randomly generated). Features f_1 and f_2 are correlated with the class value with XOR operation (i.e. class equals $f_1 \oplus f_2$).
- **Parity3 + 3**. The parity problem is a classic problem where the output is $f(x_1, \ldots, x_n) = 1$ if the number of $x_i = 1$ is odd and $f(x_1, \ldots, x_n) = 0$ otherwise. The Parity3 + 3 dataset is a modified version of the original parity dataset. The target concept is the parity of three bits. It contains 12 features among which 3 are relevant, another 3 are redundant (repeated) and other 6 are irrelevant (randomly generated).
- **Monk3**. The MONK's problems [41] rely on an artificial robot domain, in which robots are described by six different attributed $(x_1, \ldots, x_6)$. The learning task is a binary classification task. The logical description of the class of the third problem (Monk3) is the following: $(x_5 = 3 \wedge x_4 = 1 \vee (x_5 \neq 4 \vee x_2 \neq 3)$. Among the 122 samples, 5% are misclassifications, i.e., noise in the target.
- **Madelon**. The Madelon dataset [14] is a 2 class problem originally proposed in the NIPS'2003 feature selection challenge. The relevant features are situated on the vertices of a five-dimensional hypercube. Five redundant features were added, obtained by multiplying the useful features by a random matrix. Some of the previously defined features were repeated to create 10 more features. The other 480 features are drawn from a Gaussian distribution and labeled randomly. This dataset pre-

Table 1
Summary of the synthetic datasets. It shows the number of samples (#sam.), the number of features (#feat.), the relevant features (rel-feat.) and the number of classes (#cl.), as well as the presence of correlation (#corr.), noise and no linearity. G_i means that the feature selection method must select one feature within the i-th group of features.

Dataset	#sam.	#feat.	rel-feat.	Corr.	Noise	No linear	#cl.
CorrAL-100	32	99	1–4	✓			2
XOR-100	50	99	1–2			✓	2
Parity3 + 3	64	12	1–3			✓	2
Monk3	122	6	2,4,5		✓		2
Madelon	2400	500	1–5		✓	✓	2
Led-25	50	24	1–7		✓		10
Led-100	50	99	1–7		✓		10
SD1	75	4020	G_1, G_2				3
SD2	75	4040	$G_1 - G_4$				3
SD3	75	4060	$G_1 - G_6$				3

sents high dimensionality both in number of features and in number of samples and the data were distorted by adding noise, flipping labels, shifting and rescaling.

- **The LED problem**. The LED problem [8] is a simple classification task that consists of, given the actived LEDs on a seven segments display, identifying the digit that the display is representing. Thus, the classification task to be solved is described by seven attributes and ten possible classes available ($C = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9$). A 1 is an attribute which indicates that the LED is active, and a 0 indicates that it is not active. Two versions of the LED problem will be used: the first one, Led25, adding 17 irrelevant features (with random binary values) and the second one, Led100, adding 92 irrelevant attributes.
- **SD1, SD2 and SD3**. These three synthetic datasets [48] are challenging problems because of their high number of features (around 4000) and the small number of samples (75), besides of a high number of irrelevant attributes. SD1, SD2 and SD3 are three-class datasets with 75 samples (each containing 25 samples). Each synthetic dataset consists of both relevant and irrelevant features. The relevant features in each dataset are generated from a multivariate normal distribution using mean and covariance matrixes. Besides, 4000 irrelevant features are added to each dataset, where 2000 are drawn from a normal distribution of $N(0, 1)$ and the other 2000 are sampled with a uniform distribution $U[-1, 1]$.

4.2.2. Real datasets

In order to obtain significant conclusions about the effect of feature selection on classification, we also used 30 real datasets. 13 datasets were downloaded from the UCI repository [3], with the restriction of having at least 50 features, and also 17 microarray datasets were used due to their high dimensionality [33]. Tables 2 and 3 profile the main characteristics of the datasets used in this research in terms of the number of samples, features and classes.

5. Experiments

In this section, the results obtained after applying seven different feature selection methods over ten synthetic and 30 real datasets will be presented. While two of the feature selection methods return a feature subset (CFS and INTERACT), the other five (IG, ReliefF, MIM, JMI and mRMR) are ranker methods, so a threshold is mandatory in order to obtain a final subset of features. In this work we have opted for retaining the top 10%, 20% and $log_2(n)$ [40] of the most relevant features of the ordered ranking, where n is the number of features in a given dataset. In the case of SD and microarray datasets, due to the mismatch between dimensionality and sample size, the thresholds selected the top 5%, 10% and $log_2(n)$ features, respectively. To estimate the error rate we computed a 3×5-fold cross validation (i.e. 3 repetitions of a cross validation with 5 folds), including both feature selection and classification steps in a single cross-validation loop, as recommended in Kuncheva et al. [26] (see Fig. 1).

In order to check if the importance of choosing a specific classifier decreases after applying a good preprocessing step, we analyzed the standard deviation of the classification error obtained by the five classifiers in average. We consider that a lower value of standard deviation represents a lower influence of the classifier selected. The rationale of this idea is the following: using directly the datasets with all features, there will probably be classifiers that exhibit a good performance, while others will perform poor. In this case, the standard deviation in the suite of classifiers might be high. However, using feature selection as preprocessing step will feed best quality inputs to the classifiers and thus the standard deviation of the suite will be lower. This will prove a lower influence on the choice of the appropriate classifier, enhancing the need for a previous feature selection.

5.1. Dealing with synthetic datasets

The first step to test the effectiveness of a feature selection method should be on synthetic data, since the knowledge of the optimal features and the chance to modify the experimental conditions allows to draw more useful conclusions.

To explore the statistical significance of our classification results, we analyzed the standard deviation by using a Friedman test with the Nemenyi post hoc test. Fig. 2 presents the critical different diagrams, introduced by Demšar [11], where groups of methods that are not significantly different (at $\alpha = 0.10$) are connected. The top line in the critical difference diagram is the axis on which we plot the average ranks of methods. The axis is turned

Table 2
Characteristics of the 13 real datasets. It shows the number of samples (#sam.), features (#feat.) and classes (#cl.).

Dataset	#sam.	#feat.	#cl.	Dataset	#sam.	#feat.	#cl.
arrhythmia	452	279	13	optdigits	5620	64	10
conn-bench-sonar	208	60	2	ozone	2536	72	2
gisette	7000	5000	2	semeion	1593	256	10
hill-valley	606	100	2	sonar	208	60	2
low-res-spect	531	100	9	splice	3175	60	3
molec-biol-promoter	106	57	2	USPS	9298	256	10
molec-biol-splice	3190	60	3				

Table 3
Characteristics of the 17 microarray datasets. It shows the number of samples (#sam.), features (#feat.) and classes (#cl.).

Dataset	#sam.	#feat.	#cl.	Dataset	#sam.	#feat.	#cl.
9-tumors	60	5726	9	gli85	85	22283	2
11-tumors	174	12533	11	leukemia-1	72	5327	3
brain	21	12625	2	leukemia-2	72	11225	3
brain-tumor-1	90	5920	5	lung-cancer	203	12600	5
brain-tumor-2	50	10367	4	ovarian	253	15154	2
CLL-SUB-111	111	11340	3	smk	187	19993	2
CNS	60	7129	2	SRBCT	83	2308	4
colon	62	2000	2	TOX-171	171	5748	4
DLBCL	47	4026	2				

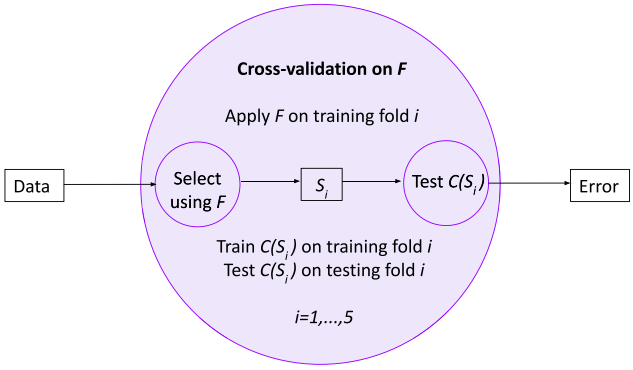

Fig. 1. Diagram of the protocol for feature selection. Boxes represent inputs and outputs; and circles represent procedures. *S* is the selected subset of features; Error refers to the classification error predicted through cross-validation for the classifier *C*, and *F* is the feature selection method chosen. This process is repeated three times.

so that the lowest (best) ranks are to the right since we perceive the methods on the right side as better. As we are dealing with synthetic datasets, the relevant features of each dataset are known (Table 1). Thus, firstly we compared the results obtained by the classifiers over the original datasets (All feats), i.e. without feature selection, and then the datasets with the relevant features (Relevant). As can be seen in Fig. 2(a), the classifiers performed better on average over the datasets with only the relevant features but with no statistical significance over the classifier using the original data. However, these results might be obscured by the fact that three nonlinear problems were tested: XOR-100, Parity3 + 3 and Madelon. Then, for these datasets, the classification errors obtained

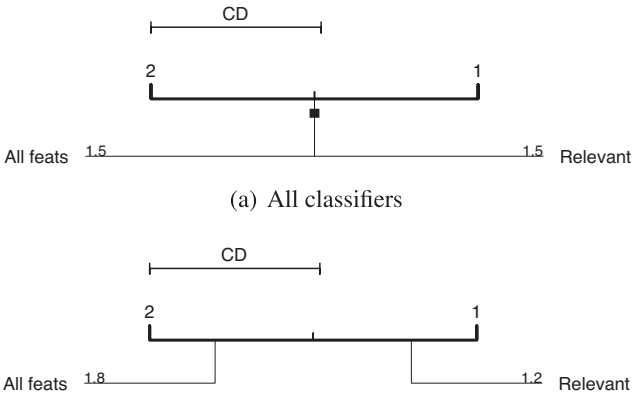

Fig. 2. Critical difference diagram showing the difference in terms of standard deviation between the error obtained by the five classifiers over the ten synthetic datasets.

by the linear classifiers (naive Bayes and SVM, which cannot solve non-linear problems) are not taken into account in Fig. 2(b). As a result, statistical significance appeared now between the two approaches, thus supporting our initial hypothesis.

In Fig. 2 we have used the features that we already know are relevant. However, it might well be that the different feature selection algorithms are not able to identify all of them correctly. Thus, we have applied all the feature selection methods with different thresholds over the 10 synthetic datasets. Then the five classifiers were applied, and the standard deviation of the suite was obtained, as shown in Table 4. It can be seen that for all datasets and feature selection methods, the standard deviation is most of the times considerably lower (at least for one of the thresholds for each feature selection algorithm) than for the case of using the classifiers with all features.

Finally, we analyzed if the application of feature selection improves classification accuracy. Fig. 3 compared the performance of the feature selection methods over the five different classifiers. For ranker methods (IG, RelF, MIM, mRMR and JMI), the results with the best threshold (the one that achieves the lowest misclassification) are shown. It can be seen that, although in average feature selection methods do not achieve results which are not statistically significant different with respect to using all features, the performance obtained is better, and with fewer features.

5.2. Dealing with real datasets

In order to check if the results obtained on synthetic data can be extrapolated to real world problems, 30 real datasets were chosen. Firstly, we analyzed if the application of an adequate preprocessing step using feature selection alleviates the decision on which classifier is the most appropriate. Therefore, Table 5 shows the standard deviation of the classification errors obtained by the five classifiers. As can be seen, for all datasets and feature selection methods, the standard deviation is most of the times considerably lower than for the case when no feature selection was performed, showed in the last row and labeled as "All feats".

Once we have been reconfirmed that the use of the appropriate features causes different classifiers to obtain similar results, we proceed to analyze if the application of feature selection improves the classification performance. Thus, the same five classifiers as above were employed to compute the classification error. As can be seen in Fig. 4 for the seven feature selection methods—only the results with the best threshold (the one that achieves the lowest misclassification) are shown for rankers—, using the whole set of features is not significantly better than using the reduced datasets obtained by the feature selectors, which is a good result because feature selection has the added benefits of producing simpler and more understandable models, increasing explainability, and reducing storage requirements. Moreover, there are three filters that achieved better results on average. The worst results were obtained by the univariate methods, Information Gain and MIM, which ignore feature dependencies.

Table 4
Standard deviation of the classification errors obtained by the five classifiers over each of the ten synthetic datasets. For feature selection methods that require a threshold, the option to keep 5/10% is indicated by '−10', the option to stay with 10/20% is indicated by' −20', and the option '−log' refers to use log_2. Lower standard deviations obtained by the filters methods versus the 'All feats' approach are highlighted in bold.

	CorrAL	XOR	Parity	Monk3	Madelon	Led-25	Led-100	SD1	SD2	SD3	Average
CFS	**0.91**	**1.02**	**1.20**	**7.10**	**2.54**	**7.61**	**8.06**	**4.09**	**2.75**	**2.39**	**3.77**
INT	**1.58**	**1.02**	**1.20**	**7.10**	**2.14**	**7.85**	**8.02**	**1.65**	**1.88**	**2.87**	**3.53**
IG-10	**2.20**	2.69	**1.58**	**0.41**	**5.71**	**0.89**	**6.11**	**1.47**	**1.99**	**4.29**	**2.73**
IG-20	**1.83**	2.14	**2.11**	**0.41**	**7.04**	**4.84**	**14.55**	**4.67**	**1.06**	**1.98**	**4.06**
IG-log	**2.46**	1.76	**7.09**	**7.02**	**4.17**	**4.84**	**8.39**	**3.49**	**5.19**	**5.40**	**4.98**
ReliF-10	5.22	7.06	**0.30**	**0.70**	**2.59**	**0.89**	**6.37**	**4.97**	8.39	**3.04**	**3.95**
ReliF-20	6.29	5.39	**0.30**	**0.70**	**4.95**	**4.84**	**14.61**	**4.09**	6.93	**3.71**	**5.18**
ReliF-log	**4.72**	3.08	**2.12**	**6.49**	**3.58**	**4.84**	**6.42**	**4.19**	**0.80**	**3.64**	**3.99**
MIM-10	6.04	3.79	**0.00**	**0.41**	**4.32**	**0.89**	**7.02**	**3.89**	**5.18**	6.54	**3.81**
MIM-20	**4.63**	4.07	**2.24**	**0.41**	**5.94**	**4.84**	**14.66**	**2.41**	**5.75**	6.84	**5.18**
MIM-log	6.73	**1.39**	**2.63**	**6.40**	**4.24**	**4.84**	**5.81**	4.68	**2.75**	**2.00**	**4.15**
mRMR-10	6.18	3.36	**0.00**	**0.41**	**6.31**	**0.89**	**8.62**	**2.58**	**3.52**	**4.56**	**3.64**
mRMR-20	**4.55**	4.07	**2.24**	**0.41**	**5.77**	**4.84**	**14.32**	**3.71**	6.12	6.80	**5.28**
mRMR-log	**4.25**	3.29	**2.88**	**6.40**	**3.35**	**4.84**	**5.71**	**2.08**	**3.21**	**2.28**	**3.83**
JMI-10	**3.89**	**0.67**	**0.00**	**0.41**	**2.73**	**0.89**	**8.07**	**3.79**	**3.78**	**3.75**	**2.80**
JMI-20	5.59	3.33	**1.78**	**0.41**	**4.75**	**4.84**	**15.53**	**2.58**	**2.84**	**5.20**	**4.69**
JMI-log	**2.82**	3.71	**1.69**	**6.25**	**4.24**	**4.84**	**5.49**	**3.28**	**2.10**	**1.51**	**3.59**
All feats	4.75	1.76	14.44	9.05	10.74	14.38	24.54	4.65	5.99	6.10	9.64

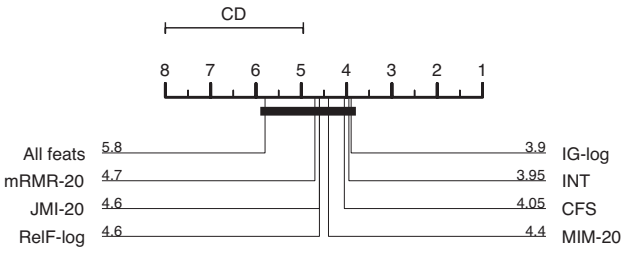

Fig. 3. Critical difference diagrams showing the average classification error ranks after applying feature selection over the 10 synthetic datasets. For filters methods that require a threshold, the option to keep 5/10% is indicated by '−10', the option to keep with 10/20% is indicated by' −20', and the option '−log' refers to use log_2.

5.2.1. Dealing with high dimensional datasets

Microarray technology is used to collect information from tissue and cell samples regarding gene expression differences that could be useful for diagnosing diseases [33]. The classification of this type of data has been viewed as a particular challenge for machine learning researchers, mainly due to the mismatch between dimensionality and sample size. The existence of many features relative to few samples means that false positives findings due to chance are very likely in terms of both identifying relevant genes and building predictive models [35]. Moreover, several studies have demonstrated that most of the genes measured in a DNA microarray experiment do not actually contribute to efficient sample classification. To avoid this "curse of dimensionality", feature selection is advisable so as to identify the specific genes that enhance classification accuracy.

Following the same study as for the previous datasets, and trying to establish if the application of feature selection might alleviate the decision on which classifier is the most appropriate, Table 6 shows the standard deviation of the classification errors obtained by the five classifiers over the 17 microarray datasets. It can be seen again that, for all microarray datasets and feature selection methods, the standard deviation is most of the times considerably lower (at least for one of the thresholds for each ranker method) than for the case of using the classifiers with all features (All feats).

Finally, we compared the classification accuracy achieved by the seven feature selection methods over the five different classifiers. Fig. 5 shows that, except for JMI filter, the results obtained by all the feature selection methods are significantly better than the results obtained over the version using all the features of the dataset. This reflects the importance of feature selection on high dimensional datasets.

5.3. Case studies

So far we have proven that our hypothesis holds and the fact of using an adequate feature selection method alleviates the choice of the classifier. However, there are other specific aspects that can influence our results, such as the effects of parameter tuning, the choice of kernel in SVMs, or the use of classifiers for which feature selection is not typically applied. We will try to shed light on these issues in the following subsections:

5.3.1. Case study I: The influence of parameter tuning

In machine learning problems, it is generally necessary to set the parameters used by the classification algorithms in order to achieve the best possible model and, consequently, the best results. Experiments show a substantial decrease in error when the right parameters are used. However, there is an associated problem in adjusting the hyperparameters of most machine learning methods. This parameter tuning involves a high computational cost or even the risk of relying on assumptions that may bias the results. In order to analyze the possible effect of parameter tuning on our hypothesis, we consider two of the previously used classifiers, *k*-Nearest Neighbor and C4.5.

5.3.1.1. k-Nearest Neighbor classifier. In our previous experiments, a standard value of $k = 3$ was set for *k*-Nearest Neighbor classifier, avoiding even numbers to prevent ties and $k = 1$ for triviality. In this case study, we vary the number of neighbors with values 3, 5, 7, 9 and 11. The datasets used are those described in Table 2, and the standard deviation of the suite of the five classifiers was obtained, as shown in Table 7. It can be seen that, regardless of the number of neighbors set in *k*NN, the standard deviation is considerably lower than for the case of using the classifiers with all features.

Table 5
Standard deviation of the classification errors obtained by the five classifiers for the real datasets tested. For feature selection methods that require a threshold, the option to keep 10% is indicated by '−10', the option to stay with 20% is indicated by' −20', and the option '−log' refers to use log_2. Lower standard deviations obtained by the filters methods versus the 'All feats' approach are highlighted in bold.

	arrhythmia	conn-bench-sonar	gisette	hill-valley	low-res-spect	molec-biol-prometer	molec-biol-splice	optdigits	ozone	semeion	sonar	splice	USPS	Average
CFS	**1.80**	**3.84**	**1.07**	**1.63**	**2.60**	**3.05**	**10.26**	**3.84**	**8.16**	**5.21**	**4.63**	**8.56**	**3.87**	**4.50**
INT	**1.35**	**4.36**	2.93	**1.52**	**2.94**	**3.50**	**10.45**	**3.53**	**7.97**	**3.47**	**4.76**	**8.81**	**4.11**	**4.59**
IG-10	**1.78**	**2.18**	**2.04**	**1.97**	**2.16**	**2.93**	**4.95**	**3.12**	**10.44**	**2.22**	**2.93**	**3.07**	**6.56**	**3.57**
IG-20	**2.52**	5.59	**1.80**	**1.11**	**3.17**	**3.36**	**6.38**	**3.81**	**10.06**	**3.03**	**4.23**	**4.42**	**5.59**	**4.24**
IG-log	**1.84**	**2.18**	**1.90**	**1.02**	**1.99**	**2.93**	**4.95**	**3.12**	**9.27**	**1.01**	**2.93**	**3.07**	**6.43**	**3.28**
RelF-10	**2.41**	**3.62**	3.10	**1.91**	**3.24**	**3.05**	**3.93**	**3.24**	12.08	**1.84**	**2.02**	**2.17**	**6.39**	**3.77**
RelF-20	**3.67**	**4.75**	2.58	**3.43**	**3.23**	**4.41**	**6.33**	4.11	14.64	**3.52**	**4.31**	**4.47**	**5.22**	**4.97**
RelF-log	**0.33**	**3.62**	2.71	**2.13**	**3.24**	**3.05**	**3.93**	**3.24**	**10.60**	**1.88**	**2.02**	**2.17**	**6.12**	**3.46**
MIM-10	**2.49**	**2.09**	**2.08**	**1.22**	**1.87**	**3.10**	**4.95**	**3.10**	**11.02**	**2.22**	**2.21**	**3.07**	**6.28**	**3.52**
MIM-20	**2.98**	**4.67**	**1.84**	**2.06**	**3.06**	**3.94**	**6.38**	**3.79**	**10.70**	**3.03**	**4.57**	**4.46**	**5.59**	**4.39**
MIM-log	**1.57**	**2.09**	**1.90**	**1.30**	**2.10**	**3.10**	**4.95**	**3.10**	**9.61**	**1.01**	**2.21**	**3.07**	**6.63**	**3.28**
mRMR-10	**2.85**	**2.48**	**1.75**	**1.51**	**2.62**	**4.20**	**4.91**	**2.18**	**6.07**	**1.69**	**2.01**	**3.03**	**4.34**	**3.05**
mRMR-20	**2.89**	**4.68**	**1.76**	**2.66**	**2.92**	**4.13**	**6.33**	**3.85**	**10.70**	**3.15**	**4.48**	**4.42**	**5.61**	**4.43**
mRMR-log	**2.55**	**2.48**	**0.92**	**1.69**	**2.18**	**4.20**	**4.91**	**2.18**	**4.94**	**0.47**	**2.01**	**3.03**	**3.35**	**2.69**
JMI-10	**2.52**	**3.30**	**1.87**	**2.20**	**2.47**	**4.30**	**4.91**	**3.19**	**4.86**	**1.70**	**1.98**	**3.03**	**4.52**	**3.14**
JMI-20	**2.54**	**3.83**	**1.63**	3.68	**3.61**	**3.78**	**6.39**	**3.94**	**8.16**	**2.72**	**4.21**	**4.36**	**4.57**	**4.11**
JMI-log	**2.66**	**3.30**	**2.63**	**1.77**	**2.10**	**4.30**	**4.91**	**3.19**	**3.93**	**0.39**	**1.98**	**3.03**	**3.97**	**2.94**
All feats	4.52	5.90	2.35	3.52	3.56	7.05	13.47	4.07	11.36	6.90	6.38	9.53	7.64	6.63

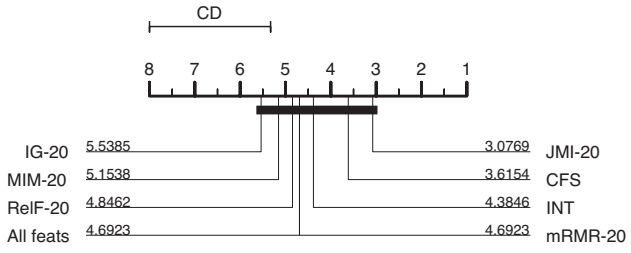

Fig. 4. Critical difference diagrams showing the average classification error ranks after applying feature selection over the 13 real datasets. For filters methods that require a threshold, the option to keep 10% is indicated by '−10', the option to keep with 20% is indicated by' −20', and the option '-log' refers to use log_2.

5.3.1.2. C4.5 classifier. The decision tree classifier selected was J48 [36], an improved version of the C4.5 classification algorithm, which has several parameters but only two of which influence the amount of pruning: the pruning confidence (C) and the minimum number of instances per leaf (M). The pruning confidence defines, for each pruning operation, the probability of error in the hypothesis for which the deterioration due to this operation is significant. The lower this value, the more pruning operations allowed. For previous experiments, we employed the default parameters $C = 0.25$ and $M = 2$ for C4.5 classifier. In this case study, the settings used were: C (0.1, 0.25 and 0.5) and M (2 and 10), that is, a total of six different combinations of parameters for each real dataset (Table 2). The standard deviation of the classification errors obtained by the five classifiers is shown in Table 8. As can be seen, regardless of the values taken by the parameters C and M in C4.5 the classifier, the standard deviation is lower than for the case when no feature selection was performed (All feats).

In light of these results, it seems that parameter tuning does not have any influence on our hypothesis, since—regardless of the values taken by the parameters in kNN and C4.5 classifiers—a lower standard deviation of the classification error is achieved by the different classifiers when feature selection is performed.

5.3.2. Case study II: The choice of kernel in SVM classifiers

In application to classification problems, SVMs can produce models with different kinds of decision borders—it depends on the parameters used (especially on the kernel type). Thus, the borders can be linear or highly nonlinear. A nonlinear kernel allows to solve nonlinear problems, but at the expense of being more computationally demanding. In this case of study, in addition to the SVM with linear kernel already used in previous experiments, a Gaussian Radial Basis Function (RBF) with values $C = 1$ and $\gamma = 0.01$ was applied on the SVM classifier [10]. Since SVMs were initially constructed to solve binary classification problems, and to avoid the possible different strategies to deal with multiclass problems, the datasets used are those binary datasets described in Table 2.

Table 9 shows the standard deviation of the classification errors obtained by the six classifiers—the five previously used classifiers plus SVM with the Gaussian kernel—over the six binary real datasets. It can be seen that, for all real datasets and feature selection methods, the standard deviation is (at least for one of the thresholds for each ranker method) lower than for the case of using the classification algorithms with all features. Thus, it is confirmed again that the application of an adequate feature selection method can attenuate the relevance of the decision of selecting the best classifier, regardless of the kernel type.

5.3.3. Case study III: Intrinsic extracted features vs. selected features

Neural networks, and deep learning algorithms in general, have the capability of automating feature extraction (the extraction of

Table 6
Standard deviation of the classification errors obtained by the five classifiers over the 17 microarray datasets. For feature selection methods that require a threshold, the option to keep 5% is indicated by '−5', the option to stay with 10% is indicated by' −10', and the option '-log' refers to use log_2. Lower standard deviations obtained by the filters methods versus the 'All feats' approach are highlighted in bold.

	9-tumors	11-tumors	brain	brain-tumor-1	brain-tumor-2	CLL-SUB-111	CNS	colon	DLBCL	gli85	leukemia-1	leukemia-2	lung-cancer	ovarian	smk	SRBCT	TOX-171	Average
CFS	**9.42**	**8.13**	**9.60**	**6.81**	**4.84**	**5.33**	**5.76**	**5.34**	**9.09**	**6.11**	**3.87**	**3.41**	**3.75**	**1.04**	**4.39**	**7.24**	**11.56**	**6.22**
INT	**7.67**	**9.19**	**6.49**	8.77	**8.85**	**3.95**	**5.79**	**2.00**	**10.54**	**4.84**	**4.02**	**3.61**	**3.94**	**1.10**	**3.63**	**5.87**	**9.85**	**5.89**
IG-5	**10.81**	**7.86**	**9.01**	**5.88**	**5.36**	**4.13**	**3.57**	**5.43**	**8.43**	**4.21**	**2.48**	**3.82**	**2.31**	**2.27**	**3.48**	**6.32**	**13.48**	**5.81**
IG-10	**10.40**	**8.68**	**11.92**	**5.93**	**7.10**	**4.69**	**3.51**	**4.61**	**8.60**	**5.12**	**2.61**	**3.62**	**2.30**	**2.45**	**4.41**	**6.38**	**13.32**	**6.21**
IG-log	**4.71**	**7.21**	**9.77**	**3.26**	**4.49**	**3.57**	**4.22**	**3.09**	**8.58**	**3.25**	**3.59**	**5.65**	**1.83**	**0.83**	**1.84**	**5.27**	**5.04**	**4.48**
ReIF-5	11.05	**7.56**	**12.37**	**5.53**	**5.35**	**4.01**	5.38	**3.60**	**8.13**	**3.74**	**2.16**	**3.24**	**2.18**	**1.49**	**2.65**	**6.06**	**13.40**	**5.76**
ReIF-10	12.84	**8.38**	15.88	**6.06**	**8.83**	**4.38**	5.52	**3.05**	**8.10**	**4.76**	**2.26**	**3.30**	**2.69**	**2.19**	**3.72**	**5.99**	**13.90**	**6.58**
ReIF-log	**2.81**	**6.27**	**5.70**	**3.30**	**6.81**	**1.17**	**3.24**	**2.61**	**7.48**	**3.63**	**2.79**	**3.08**	**3.37**	**0.81**	**3.22**	**3.70**	**2.52**	**3.68**
MIM-5	11.08	**11.33**	**4.92**	**6.76**	**7.14**	**4.40**	**2.87**	**3.19**	**9.16**	**4.57**	**2.30**	**3.54**	**2.74**	**2.53**	**4.12**	**5.66**	**12.94**	**5.84**
MIM-10	11.77	**10.19**	**4.25**	**5.53**	**6.68**	**4.14**	**3.20**	**3.37**	**9.00**	**4.55**	**2.67**	**3.72**	**3.62**	**2.38**	**3.50**	**6.21**	**14.50**	**5.84**
MIM-log	**3.92**	**5.18**	**9.97**	**3.39**	**4.59**	**2.77**	**3.86**	**4.54**	**7.18**	**4.18**	**3.85**	**4.49**	**2.67**	**0.66**	**1.91**	**4.69**	**3.18**	**4.18**
mRMR-5	**10.54**	**10.59**	**5.51**	**5.79**	10.59	**3.93**	**1.42**	**5.49**	**9.18**	**4.99**	**2.32**	**4.49**	**2.63**	**1.06**	**3.73**	**6.10**	**13.86**	**6.01**
mRMR-10	11.80	**10.22**	**2.67**	**5.62**	11.22	**5.36**	**3.20**	**4.04**	**8.64**	**4.05**	**2.67**	**4.43**	**3.16**	**2.46**	**4.05**	**6.02**	**14.57**	**6.13**
mRMR-log	**0.88**	**4.52**	**3.88**	**6.38**	**5.27**	**3.89**	**3.03**	**2.82**	**6.82**	**3.57**	**3.18**	**2.90**	**1.72**	**0.66**	**1.49**	**4.84**	**3.71**	**3.50**
JMI-5	**10.23**	14.14	**5.34**	**7.35**	**9.53**	**6.93**	**4.47**	**4.31**	**7.34**	**5.27**	**3.53**	**5.99**	**3.85**	**0.99**	**4.02**	**5.88**	**13.20**	**6.61**
JMI-10	11.42	14.36	**5.74**	**6.03**	**9.42**	**7.56**	**4.83**	**4.52**	**7.09**	**6.43**	**2.81**	**5.74**	**4.68**	**2.07**	**5.50**	**7.02**	**13.17**	**6.96**
JMI-log	**2.93**	**5.78**	**10.12**	**3.30**	**2.72**	**3.96**	**3.95**	**4.18**	**6.08**	**4.16**	**2.65**	**4.11**	**3.56**	**1.01**	**2.00**	**3.96**	**5.84**	**4.14**
All feats	10.89	11.40	14.10	8.56	9.77	7.97	4.99	9.15	10.57	6.45	7.53	6.28	5.10	3.52	5.16	9.40	14.62	8.56

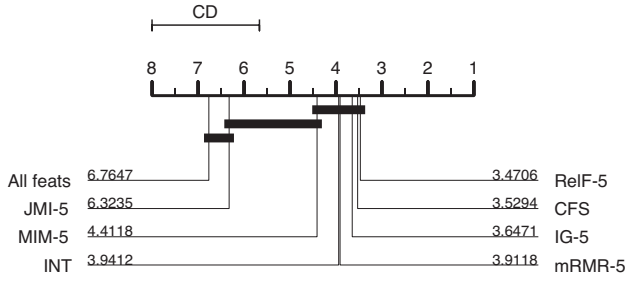

Fig. 5. Critical difference diagrams showing the average classification error ranks after applying feature selection over the 17 microarray datasets. For filters methods that require a threshold, the option to keep 5% is indicated by '−5', the option to keep with 10% is indicated by' −10', and the option '−log' refers to use log_2.

representations) from the data. The representation is learned through the data which is fed directly into deep nets without human knowledge. Thus, in this case study, we include a Multilayer Perceptron among the classification algorithms in order to evaluate if, also for this neural network, using feature selection as preprocessing step can attenuate the relevance of the decision of selecting the best classification algorithm.

Multilayer Perceptron (MLP) is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate outputs [39]. The MLP is also known as the initial deep architecture, and it can structure a more efficient model on separating non-linear problems. The experimental setup considers the real datasets described in Table 2, except Gisette, due to its space complexity (7000 samples and 5000 features). Table 10 shows the results in terms of the standard deviation of the classification errors obtained by the six classifiers—the five previously used classifiers plus MLP—over the 12 real datasets. As can be seen, for all the datasets and filter methods, the standard deviation is considerably lower than for the case when no feature selection was performed, confirming our hypothesis.

6. Conclusions

Feature selection has been widely used as a preprocessing step that reduces the dimensions of a problem and in some cases it even improves classification accuracy. Although the benefits of feature selection in a plethora of applications have been widely proved in the literature, in this work we aim to go one step further and check our hypothesis that, provided that the best features are selected, the choice of the best classifier is not so critical. In particular, for testing this hypothesis, we expected to have little variation among the different classification errors obtained by different classifiers when using the right features.

Thus, in this paper, we analyzed the effect of this preprocessing task on the classification performance over several synthetic and real datasets. In light of the results, we can conclude that: (i) the choice of a classifier is less critical if we apply a good feature selection method before the classification task and (ii) feature selection not only alleviates the choice but also improves the predictive accuracy and reduces the complexity of machine learning models, specially on high dimensional datasets. Besides, and concerning the different feature selection methods, CFS appears to be the best performing filter regardless of the nature of the dataset. We think that the results of this paper will open the door to put more effort on the data curation stages, since it remained demonstrated that improving data quality is highly relevant for obtaining good performance in subsequent learning stages. Furthermore, having good quality data has the potential

Table 7
Standard deviation of the classification errors obtained by the five classifiers—with different number of k neighbors for k NN classifier—over the 13 real datasets. Lower standard deviations obtained by the filters methods versus the 'All feats' approach are highlighted in bold.

	$k = 3$	$k = 5$	$k = 7$	$k = 9$	$k = 11$
CFS	**4.614**	**4.537**	**4.497**	**4.436**	**4.441**
INT	**4.724**	**4.615**	**4.565**	**4.503**	**4.496**
IG-10	**3.475**	**3.580**	**3.596**	**3.619**	**3.675**
IG-20	**4.323**	**4.292**	**4.330**	**4.351**	**4.306**
IG-log	**3.444**	**3.601**	**3.621**	**3.648**	**3.688**
RelF-10	**3.814**	**3.865**	**3.941**	**3.988**	**4.058**
RelF-20	**4.821**	**4.734**	**4.826**	**4.819**	**4.831**
RelF-log	**3.532**	**3.607**	**3.672**	**3.690**	**3.710**
MIM-10	**3.577**	**3.654**	**3.657**	**3.675**	**3.713**
MIM-20	**4.408**	**4.396**	**4.398**	**4.466**	**4.448**
MIM-log	**3.407**	**3.518**	**3.525**	**3.560**	**3.607**
mRMR-10	**2.939**	**2.899**	**2.897**	**2.924**	**2.948**
mRMR-20	**4.313**	**4.296**	**4.296**	**4.366**	**4.341**
mRMR-log	**2.698**	**2.680**	**2.668**	**2.693**	**2.709**
JMI-10	**3.116**	**3.059**	**3.077**	**3.078**	**3.137**
JMI-20	**4.063**	**3.971**	**3.970**	**3.910**	**3.921**
JMI-log	**2.917**	**2.919**	**2.909**	**2.921**	**2.942**
All feats	6.367	6.314	6.066	6.049	6.086

Table 8
Standard deviation of the classification errors obtained by the five classifiers—with different values of C and M parameters for C4.5 classifier—over the 13 real datasets. Lower standard deviations obtained by the filters methods versus the 'All feats' approach are highlighted in bold.

	$C = 0.1$ $M = 2$	$C = 0.1$ $M = 10$	$C = 0.25$ $M = 2$	$C = 0.25$ $M = 10$	$C = 0.5$ $M = 2$	$C = 0.5$ $M = 10$
CFS	**4.656**	**4.985**	**4.628**	**4.985**	**4.627**	**5.025**
INT	**4.695**	**4.981**	**4.669**	**4.974**	**4.690**	**5.014**
IG-10	**3.464**	**3.627**	**3.443**	**3.624**	**3.444**	**3.631**
IG-20	**4.268**	**4.501**	**4.303**	**4.478**	**4.307**	**4.520**
IG-log	**3.451**	**3.546**	**3.402**	**3.539**	**3.390**	**3.548**
RelF-10	**3.811**	**3.939**	**3.813**	**3.923**	**3.796**	**3.914**
RelF-20	**4.869**	**5.193**	**4.856**	**5.177**	**4.871**	**5.184**
RelF-log	**3.536**	**3.576**	**3.527**	**3.566**	**3.541**	**3.556**
MIM-10	**3.568**	**3.815**	**3.544**	**3.827**	**3.505**	**3.796**
MIM-20	**4.459**	**4.733**	**4.497**	**4.692**	**4.484**	**4.652**
MIM-log	**3.365**	**3.547**	**3.351**	**3.560**	**3.314**	**3.531**
mRMR-10	**3.008**	**3.247**	**2.979**	**3.252**	**2.969**	**3.256**
mRMR-20	**4.318**	**4.585**	**4.351**	**4.544**	**4.335**	**4.504**
mRMR-log	**2.714**	**2.874**	**2.669**	**2.873**	**2.642**	**2.882**
JMI-10	**3.101**	**3.306**	**3.045**	**3.274**	**3.030**	**3.237**
JMI-20	**4.035**	**4.274**	**4.049**	**4.269**	**4.046**	**4.282**
JMI-log	**2.884**	**2.999**	**2.842**	**2.984**	**2.831**	**2.972**
All feats	6.471	6.758	6.437	6.753	6.430	6.782

of avoiding the use of more complex classifiers, which usually come with high computational costs.

While the initial findings are promising, further research is necessary. As future research, we plan to study in depth whether, when analyzing the standard deviation, the ranker methods that obtain a higher value are those with a higher threshold, possibly due to the fact that choosing a higher number of features entails more variability.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been supported by the National Plan for Scientific and Technical Research and Innovation of the Spanish Government (Grant PID2019-109238 GB-C2), and by the Xunta de Galicia (Grant ED431C 2018/34) with the European Union ERDF funds. CITIC, as Research Center accredited by Galician University System, is funded by "Consellería de Cultura, Educación e Universidades from Xunta de Galicia", supported in an 80% through ERDF Funds, ERDF Operational Programme Galicia 2014–2020, and the remaining 20% by "Secretaría Xeral de Universidades" (Grant ED431G 2019/01). Funding for open access charge: Universidade da Coruña/CISUG.

Table 9
Standard deviation of the classification errors obtained by the five classifiers plus SVM with Gaussian kernel over each of the six binary real datasets. Lower standard deviations obtained by the filters methods versus the 'All feats' approach are highlighted in bold.

	conn-bench-sonar	gisette	hill-valley	molec-biol-promoter	ozone	sonar	Average
CFS	**4.80**	**18.17**	**2.09**	**2.74**	**7.51**	**4.61**	**6.65**
INT	**4.08**	**18.15**	**2.09**	**2.68**	**7.49**	**3.87**	**6.39**
IG-10	**3.22**	19.02	**1.02**	**2.49**	10.45	**2.21**	**6.40**
IG-20	**5.12**	19.10	**0.85**	**2.68**	**9.22**	**4.37**	**6.89**
IG-log	**3.22**	**2.49**	**0.94**	**2.49**	**9.20**	**2.21**	**3.42**
RelF-10	**4.00**	18.85	**2.27**	**2.24**	11.08	**1.83**	**6.71**
RelF-20	**4.68**	19.02	**2.69**	**3.13**	14.52	**4.14**	**8.03**
RelF-log	**4.00**	**3.98**	**2.01**	**2.24**	**9.64**	**1.83**	**3.95**
MIM-10	**2.23**	19.00	**1.46**	**3.31**	10.76	**1.97**	**6.46**
MIM-20	**4.66**	19.13	**2.15**	**3.01**	**9.60**	**4.13**	**7.12**
MIM-log	**2.23**	**2.52**	**1.27**	**3.31**	**9.61**	**1.97**	**3.48**
mRMR-10	**3.00**	19.11	**1.52**	**3.30**	**5.08**	**2.86**	**5.81**
mRMR-20	**4.69**	19.13	**2.59**	**3.40**	**9.62**	**4.17**	**7.27**
mRMR-log	**3.00**	**2.51**	**1.95**	**3.30**	**4.03**	**2.86**	**2.94**
JMI-10	**2.53**	19.09	**2.62**	**4.05**	**4.39**	**1.84**	**5.75**
JMI-20	**4.39**	19.16	**2.79**	**3.38**	**7.40**	**4.02**	**6.85**
JMI-log	**2.53**	**3.17**	**1.20**	**4.05**	**3.52**	**1.84**	**2.72**
All feats	5.48	18.81	2.75	4.26	10.31	6.60	8.04

Table 10
Standard deviation of the classification errors obtained by the five classifiers plus the Multilayer Perceptron for the 12 real datasets tested. Lower standard deviations obtained by the filters methods versus the 'All feats' approach are highlighted in bold.

	arrhythmia	conn-bench-sonar	hill-valley	low-res-spect	molec-biol-prometer	molec-biol-splice	optdigits	ozone	semeion	sonar	splice	USPS	Average
CFS	**1.60**	**3.11**	**0.88**	**3.16**	**2.81**	**9.41**	**3.35**	**7.39**	**5.15**	**4.30**	**7.74**	**3.88**	**4.40**
INT	**1.13**	**3.99**	**0.94**	**3.37**	**3.23**	**9.29**	**3.14**	**7.46**	**3.76**	**3.96**	**7.78**	**3.69**	**4.31**
IG-10	**1.71**	**1.54**	**1.33**	**2.64**	**2.89**	**4.39**	**2.73**	**9.22**	**1.67**	**2.58**	**2.88**	**6.05**	**3.30**
IG-20	**2.74**	**4.18**	**0.93**	**3.61**	**4.17**	**5.78**	**3.36**	**9.61**	**3.01**	**3.63**	**4.02**	**5.14**	**4.18**
IG-log	**2.07**	**1.54**	**1.14**	**2.54**	**2.89**	**4.39**	**2.73**	**8.82**	**0.89**	**2.58**	**2.88**	**5.55**	**3.17**
RelF-10	**2.70**	**2.87**	**2.16**	**3.73**	**2.38**	**3.60**	**2.77**	11.07	**1.65**	**2.26**	**2.06**	**6.04**	**3.61**
RelF-20	**3.21**	**4.17**	**2.09**	**3.95**	**4.14**	**5.81**	**3.74**	13.64	**3.44**	**3.76**	**4.14**	**4.83**	**4.74**
RelF-log	**0.81**	**2.87**	**2.16**	**3.82**	**2.38**	**3.60**	**2.77**	**9.57**	**1.84**	**2.26**	**2.06**	**5.68**	**3.32**
MIM-10	**2.41**	**1.85**	**1.40**	**1.90**	**3.15**	**4.39**	**2.69**	10.33	**1.67**	**2.06**	**2.90**	**5.96**	**3.39**
MIM-20	**2.72**	**3.97**	**2.07**	**3.48**	**3.42**	**5.80**	**3.36**	**9.79**	**3.01**	**3.60**	**4.07**	**5.18**	**4.21**
MIM-log	**1.90**	**1.85**	**0.95**	**2.05**	**3.15**	**4.39**	**2.69**	**9.15**	**0.89**	**2.06**	**2.90**	**6.02**	**3.17**
mRMR-10	**1.95**	**2.39**	**1.50**	**3.03**	**3.29**	**4.32**	**2.23**	**5.70**	**1.48**	**1.89**	**2.89**	**3.94**	**2.88**
mRMR-20	**2.66**	**3.88**	**2.09**	**3.43**	**3.40**	**5.70**	**3.40**	**9.78**	**2.95**	**3.52**	**4.09**	**5.18**	**4.17**
mRMR-log	**2.44**	**2.39**	**1.31**	**2.61**	**3.29**	**4.32**	**2.23**	**4.89**	**0.35**	**1.89**	**2.89**	**3.04**	**2.64**
JMI-10	**2.50**	**2.40**	**1.93**	**3.20**	**3.07**	**4.35**	**2.65**	**4.45**	**1.67**	**2.03**	**2.91**	**4.17**	**2.94**
JMI-20	**1.96**	**3.82**	**3.41**	**3.88**	**2.89**	**5.70**	**3.48**	**7.36**	**2.57**	**3.47**	**4.06**	**4.36**	**3.91**
JMI-log	**2.29**	**2.40**	**1.88**	**2.68**	**3.07**	**4.35**	**2.65**	**3.66**	**0.38**	**2.03**	**2.91**	**3.83**	**2.68**
All feats	4.26	5.19	3.80	4.39	4.54	12.22	3.69	10.21	6.80	5.89	8.54	7.27	6.40

References

[1] C.C. Aggarwal, Data classification. Algorithms and applications, CRC Press, Taylor & Francis Group, 2015.
[2] D.W. Aha, D. Kibler, M.K. Albert, Instance-based learning algorithms, Mach. Learn. 6 (1991) 37–66.
[3] K. Bache, M. Linchman, UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences. [Online; accessed December 2020]. URL:http://archive.ics.uci.edu/ml/..
[4] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, A review of feature selection methods on synthetic data, Knowledge Inf. Syst. 34 (2013) 483–519.
[5] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Recent advances and emerging challenges of feature selection in the context of big data, Knowl.-Based Syst. 86 (2015) 33–45.
[6] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, J.M. Benítez, F. Herrera, A review of microarray datasets and applied feature selection methods, Inf. Sci. 282 (2014) 111–135.
[7] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.
[8] L. Breiman, Classification and regression trees, Routledge, 2017.
[9] G. Brown, A. Pocock, M.J. Zhao, M. Luján, Conditional likelihood maximisation: a unifying framework for information theoretic feature selection, J. Mach. Learn. Res. 13 (2012) 27–66.
[10] C.C. Chang, C.J. Lin, Libsvm: a library for support vector machines, ACM Trans. Intell. Syst. Technol. (TIST) 2 (2011) 1–27.
[11] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J.Mach Learn. Res. 7 (2006) 1–30.
[12] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems?, J Mach. Learn. Res. 15 (2014) 3133–3181.
[13] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.
[14] I. Guyon, S. Gunn, M. Nikravesh, L.A. Zadeh, Feature extraction: foundations and applications, volume 207, Springer, 2008.
[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, The weka data mining software: an update, ACM SIGKDD explorations newsletter 11 (2009) 10–18.

[16] M.A. Hall, Correlation-based feature selection for machine learning, 1999..
[17] M.A. Hall, L.A. Smith, Practical feature subset selection for machine learning, 1998..
[18] T.K. Ho, M. Basu, Complexity measures of supervised classification problems, IEEE Trans. Pattern Anal. Mach. Intell. (2002) 289–300.
[19] J. Hua, W.D. Tembe, E.R. Dougherty, Performance of feature-selection methods in the classification of high-dimension data, Pattern Recogn. 42 (2009) 409–424.
[20] G. Hughes, On the mean accuracy of statistical pattern recognizers, IEEE Trans. Inf. Theory 14 (1968) 55–63.
[21] G.H. John, R. Kohavi, K. Pfleger, Irrelevant features and the subset selection problem, in: Machine Learning Proceedings 1994. Elsevier, pp. 121–129..
[22] G. Kim, Y. Kim, H. Lim, H. Kim, An mlp-based feature subset selection for hiv-1 protease cleavage site analysis, Artif. Intell. Med. 48 (2010) 83–89.
[23] K. Kira, L.A. Rendell, The feature selection problem: Traditional methods and a new algorithm, in: Aaai, 1992, pp. 129–134.
[24] R. Kohavi, G.H. John, et al., Wrappers for feature subset selection, Artif. Intell. 97 (1997) 273–324.
[25] I. Kononenko, Estimating attributes: analysis and extensions of relief, in: European conference on machine learning, Springer., 1994, pp. 171–182.
[26] L.I. Kuncheva, J.J. Rodríguez, On feature selection protocols for very low-sample-size data, Pattern Recogn. 81 (2018) 660–673.
[27] P. Langley, W. Iba, Average-case analysis of a nearest neighbor algorithm, in: IJCAI, Citeseer, 1993, p. 889.
[28] D.D. Lewis, Feature selection and feature extraction for text categorization, in: Proceedings of the workshop on Speech and Natural Language, Association for Computational Linguistics, 1992, pp. 212–217..
[29] J. Li, H. Liu, Challenges of feature selection for big data analytics, IEEE Intell. Syst. 32 (2017) 9–15.
[30] A.C. Lorena, L.P. Garcia, J. Lehmann, M.C. Souto, T.K. Ho, How complex is your classification problem? a survey on measuring classification complexity, ACM Comput. Surv. (CSUR) 52 (2019) 1–34.
[31] R. Mitchell, J. Michalski, T. Carbonell, An artificial intelligence approach, Springer, 2013.
[32] L.C. Molina, L. Belanche, À. Nebot, Feature selection algorithms: A survey and experimental evaluation, in: 2002 IEEE International Conference on Data Mining, 2002. Proceedings., IEEE, 2002, pp. 306–313..
[33] L. Morán-Fernández, V. Bolón-Canedo, A. Alonso-Betanzos, Can classification performance be predicted by complexity measures? a study using microarray data, Knowl. Inf. Syst. 51 (2017) 1067–1090.
[34] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2005) 1226–1238.
[35] G. Piatetsky-Shapiro, P. Tamayo, Microarray data mining: facing the challenges, ACM SIGKDD Explorations Newsletter 5 (2003) 1–5.
[36] J.R. Quinlan, C4. 5: programs for machine learning, Elsevier, 2014.
[37] I. Rish, et al., An empirical study of the naive bayes classifier, in: IJCAI 2001 workshop on empirical methods in artificial intelligence, 2001, pp. 41–46..
[38] Y. Saeys, I. Inza, P. Larrañaga, A review of feature selection techniques in bioinformatics, Bioinformatics 23 (2007) 2507–2517.
[39] W.S. Sarle, Neural networks and statistical models, 1994..
[40] B. Seijo-Pardo, V. Bolón-Canedo, A. Alonso-Betanzos, On developing an automatic threshold applied to feature selection ensembles, Inf. Fusion 45 (2019) 227–245.
[41] S.B. Thrun, J. Bala, E. Bloedorn, I. Bratko, B. Cestnik, J. Cheng, K. De Jong, S. Dzeroski, S.E. Fahlman, D. Fisher, et al., The monk's problems a performance comparison of different learning algorithms, 1991..
[42] V. Vapnik, The nature of statistical learning theory, Springer science & business media, 2013.
[43] M. Wainberg, B. Alipanahi, B.J. Frey, Are random forests truly the best classifiers?, J Mach. Learn. Res. 17 (2016) 3837–3841.
[44] D.H. Wolpert, The lack of a priori distinctions between learning algorithms, Neural Comput. 8 (1996) 1341–1390.
[45] H.H. Yang, J. Moody, Data visualization and feature selection: New algorithms for nongaussian data, in: Advances in neural information processing systems, 2000, pp. 687–693.
[46] Y. Zhai, Y.S. Ong, I.W. Tsang, The emerging "big dimensionality", IEEE Comput. Intell. Mag. 9 (2014) 14–26.
[47] Z. Zhao, H. Liu, Searching for interacting features in subset selection, Intell. Data Anal. 13 (2009) 207–228.
[48] Z. Zhu, Y.S. Ong, J.M. Zurada, Identification of full and partial class relevant genes, IEEE/ACM Trans. Comput. Biol. Bioinf. 7 (2010) 263–277.

Laura Morán-Fernández received her B.S. (2015) and Ph.D. (2020) degrees in Computer Science from the University of A Coruña (Spain). She is currently an Assistant Lecturer in the Department of Computer Science and Information Technologies of the University of A Coruña. She received the Frances Allen Award (2021) from the Spanish Association of Artificial Intelligence (AEPIA). Her research interests include machine learning, feature selection and big data. She has co-authored three book chapters, and more than 15 research papers in international journals and conferences.

Verónica Bolón-Canedo received her B.S. (2009), M.S. (2010) and Ph.D. (2014) degrees in Computer Science from the University of A Coruña (Spain). After a postdoctoral fellowship in the University of Manchester, UK (2015), she is currently an Assistant Professor in the Department of Computer Science of the University of A Coruña. She received the Best Thesis Proposal Award (2011) and the Best Spanish Thesis in Artificial Intelligence Award (2014) from the Spanish Association of Artificial Intelligence (AEPIA). She has extensively published in the area of machine learning and feature selection. On these topics, she has co-authored two books, seven book chapters, and more than 80 research papers in international conferences and journals. Her current research interests include machine learning, feature selection and big data. She serves as Secretary of the Spanish Association of Artificial Intelligence and is member of the Spanish Young Academy.

Amparo Alonso-Betanzos received the PhD degree for her work in the area of medical expert systems in 1988 at the University of Santiago de Compostela (Spain). Later, she was a postdoctoral fellow in the Medical College of Georgia, Augusta (USA). She is currently a Full Professor in the Department of Computer Science, University of A Coruña (Spain). Her main current areas are intelligent systems, scalable machine learning, explainable AI and feature selection.