Journal of Intelligent Information Systems
https://doi.org/10.1007/s10844-023-00823-y

RESEARCH

Finding a needle in a haystack: insights on feature selection for classification tasks

Laura Morán-Fernández[1] · **Verónica Bolón-Canedo**[1]

Received: 8 May 2023 / Revised: 2 October 2023 / Accepted: 3 October 2023

Abstract
The growth of Big Data has resulted in an overwhelming increase in the volume of data available, including the number of features. Feature selection, the process of selecting relevant features and discarding irrelevant ones, has been successfully used to reduce the dimensionality of datasets. However, with numerous feature selection approaches in the literature, determining the best strategy for a specific problem is not straightforward. In this study, we compare the performance of various feature selection approaches to a random selection to identify the most effective strategy for a given type of problem. We use a large number of datasets to cover a broad range of real-world challenges. We evaluate the performance of seven popular feature selection approaches and five classifiers. Our findings show that feature selection is a valuable tool in machine learning and that correlation-based feature selection is the most effective strategy regardless of the scenario. Additionally, we found that using improper thresholds with ranker approaches produces results as poor as randomly selecting a subset of features.

Keywords Dimensionality reduction · Feature selection · Filters · Classification

1 Introduction

Artificial intelligence has made significant breakthroughs in recent years, owing to recent developments in algorithms, computing power, and big data. In particular, machine learning has had a lot of success due to its outstanding capacity to evaluate massive volumes of data automatically. Classification is one of the most important tasks in machine learning, as it allows for the prediction of events in a wide range of applications, from medical to finance. However, when faced with a high number of irrelevant and/or redundant features, several of the most popular classification algorithms can deteriorate their performance. This

✉ Laura Morán-Fernández
laura.moranf@udc.es

Verónica Bolón-Canedo
veronica.bolon@udc.es

[1] CITIC, Universidade da Coruña, A Coruña, Spain

Published online: 03 November 2023

phenomenon is known as *curse of dimensionality* and is the reason why dimensionality reduction methods play an important role in preprocessing the data.

Feature selection is one of these dimensionality reduction approaches, which is described as the process of selecting relevant features and rejecting irrelevant or redundant ones. There are numerous noisy and meaningless features that are frequently gathered or generated by various sensors and algorithms, all of which consume a significant amount of computational resources. As a result of this, feature selection is critical in the context of machine learning, as it allows for the removal of nonsense features while keeping a small subset of features to reduce computational complexity.

Feature selection methods can be classified into three categories based on their relationship to the induction algorithm (Guyon et al., 2008): (i) filters, which are independent of the induction algorithm and use metrics like mutual information or statistics like Chi^2 to determine the importance of the features; (ii) wrappers, which use the induction algorithm accuracy to determine the importance of the features; and (iii) embedded methods, which perform feature selection in the process of training and are usually specific to given learning machines. Furthermore, feature selection approaches are classified as univariate (when they compute the relevance of a single feature to the predictive class) and multivariate (when they take into account the interactions among subsets of features).

Unlike other dimensionality reduction techniques that are gaining popularity, such as feature extraction based on embeddings or deep neural networks (Salau & Jain, 2019; Kasongo & Sun, 2020), there are various applications where finding relevant features is required. In bioinformatics (for example, to discover a few important biomolecules that account for the majority of a phenotype (Climente-González et al., 2019)), in terms of decision-making fairness (e.g., instead of focusing on the fairness of the choice outcomes, locate the input features employed in the decision process (Grgic-Hlaca et al., 2018)), or in nanotechnology (for example, to establish the most important experimental conditions and physicochemical features to take into account when making a nanotoxicology risk assessment (Furxhi et al., 2020)). These applications all have one thing in common: they are not pure classification problems. In fact, knowing which features are relevant is just as crucial as correctly classifying them, because these features may provide new information about the underlying system.

However, there are numerous feature selection methods to choose from, and most researchers agree that the best feature selection approach does not exist (Bolón-Canedo et al., 2013). On top of this, new feature selection methods are appearing every year, which makes us ask the questions: do we really need so many feature selection methods? Which ones are the best to use for each type of data? In light of these concerns, the purpose of this paper is to examine the most common feature selection approaches in two scenarios: synthetic and real datasets, using random selection as a baseline. Our goal is to analyze if there are some methods that produce results that are not considerably better than those obtained by randomly selecting a subset of features. Differently from our previous work (Morán-Fernández & Bolón-Canedo, 2021), in this paper we (a) include seven synthetic datasets, trying to check the behavior of the feature and random selection when the relevant features are known, (b) examine the effects of including different levels of noise in the inputs, (c) analyze the impact of discretization on feature selection through a case study involving the variation of the number of bins in the Equal-width method, (d) compare the results obtained by applying the rough set attribute method QuickReduct with the Correlation-based Feature Selection method and (e) perform an illustrative example of feature selection over Mnist dataset.

The remainder of the paper is organized as follows: Section 3 presents the different feature selection methods employed in the study and provides a brief description of the 7 synthetic

and 55 real datasets used to reduce data dimensionality. Section 4 details the experimental results carried out, including several case studies. Finally, Section 5 contains our concluding remarks and proposals for future research.

2 Background

Machine learning researchers face an interesting dilemma when datasets expand in size; to cite Donoho (2000) *"our task is to find a needle in a haystack, teasing the relevant information out of a vast pile of glut"*. Ultra-high dimensionality necessitates a large amount of memory and a significant training computational cost. Furthermore, what is known as the "curse of dimensionality" undermines generalization abilities. As a result, in a society where huge amounts of data and features are required in a variety of fields, new solutions for dealing with the critical issue of feature selection are urgently needed (Bolón-Canedo et al., 2015).

The initial studies on feature selection date back to the 1960s (Hughes, 1968), but it was not until the 1990s that significant advancements in feature selection for solving machine learning problems were made. Because of its capacity to improve the performance of learning algorithms, feature selection has gained popularity in the field of machine learning, particularly in supervised and unsupervised processes like clustering, regression, and classification. However, the most widely used feature selection approaches were created years ago, and they are currently facing significant hurdles that could negatively impact their performance. Feature selection is a difficult task, since for a dataset with m features, the total number of possible alternatives for a feature subset is $2^m - 1$.

Furthermore, feature-to-feature correlations are common. There are a variety of two-way, three-way, and more complex correlations. A weak correlation between two features may become a strong correlation, when they are combined with other features. Furthermore, the most common types of search in feature selection, such as sequential forward or sequential backward selection, suffer from local convergence issues and significant computational costs. Table 1 shows the computational cost of some of the most popular FS methods.

As can be seen, the most sophisticated methods have quadratic complexity with the number of features, an expensive calculation process usually derived from computing the correlation of pairs of features. In this paper we will try to answer the question of if it is worth paying the price of an expensive calculation for better performance results.

Table 1 Popular filter methods and their theoretical complexity where n is the number of samples, m is the number of features, $c + c$ is the double hashing time cost and $P(m)$ is the power-set of conditional features

Method	Complexity
Information gain (Hall & Smith, 1998)	nm
ReliefF (Kononenko, 1994)	n^2m
minimum Redundancy Maximum Relevance (Peng et al., 2005)	nm^2
Joint Mutual Information (Yang & Moody, 2000)	nm^2
Correlation-based Feature Selection (Hall, 1999)	nm^2
INTERACT (Zhao & Liu, 2009)	nm^2
Mutual Information Maximisation (Lewis, 1992)	nm
QuickReduct (Shen & Chouchoulas, 2000)	$n(c + c)P(m)$

3 Methods and materials

3.1 Feature selection techniques

In the classification literature, feature selection approaches have garnered a lot of attention, and they can be divided into three categories based on their interaction with the induction algorithm (Guyon et al., 2008; Shahrjooihaghighi & Frigui, 2021): filters, wrappers, and embedding methods. We choose filter methods over wrapper and embedded methods because we want to avoid the interaction with the classifier. Furthermore, filter methods are a popular choice in the new Big Data environment, owing to their lower computing cost as compared to wrapper or embedded approaches. The seven filters used in the experiment are described below, where two of them are univariate (Information Gain and Mutual Information Maximisation) and the other five are multivariate.

- **Correlation-based Feature Selection** (CFS) is a simple multivariate filter technique that ranks feature subsets using a heuristic evaluation function based on correlation (Hall, 1999). This function tries to find subsets of features that present correlation with the class but not with one another. The idea is to remove those attributes whose correlation with the class is low (and thus they are considered irrelevant), as well as those redundant (correlation among them).
- The **INTERACT** (INT) algorithm works on the idea of symmetrical uncertainty (SU) and adds a contribution for consistency (Zhao & Liu, 2009). This method works on two steps. First, features are sorted in descending order according to their value of SU. In the second step, the algorithm starts taking those features at the end of the feature ranking, and it evaluates each feature one by one. If a feature's consistency contribution is below a predetermined threshold, it is deleted; otherwise, it is selected.
- **Information Gain** (IG) filter analyses a single feature at a time and evaluates it based on its information gain (Hall & Smith, 1998). It gives an ordered classification of all features, after which a threshold is used to choose a particular number of them based on the order.
- **ReliefF** algorithm (RelF) (Kononenko, 1994) adds the ability to deal with noisy, incomplete, and multiple class datasets to the original Relief algorithm. This algorithm's key idea is to estimate features based on how well their values discriminate between examples that are close to each other.
- **Mutual Information Maximisation** (MIM) (Lewis, 1992) obtains a ranking of attributes according to their mutual information score and selects the top k features, where k is determined by a predefined need for a certain number of features or another criterion.
- The **minimum Redundancy Maximum Relevance** (mRMR) (Peng et al., 2005) approach selects features that fulfill two conditions: they are highly relevant to the target class but no redundant among each other. Both the maximum-relevance and minimum-redundancy optimization criteria are based on mutual information.
- Another feature selection approach based on mutual information is **Joint Mutual Information** (JMI) (Yang & Moody, 2000), which uses a new criterion to evaluate candidate features. In each phase, JMI selects the feature with the highest cumulative sum of joint mutual information with the selected features and adds it to the subset S, until the number of selected features exceeds k.

In addition, for case study III (see Section 4.3.3), we will use a method belonging to the family of rough set attribute reduction algorithms:

- **QuickReduct** (QR) (Shen & Chouchoulas, 2000; Chouchoulas & Shen, 2001) employs a forward selection approach, utilizing a non-exhaustive hill-climbing search that may encounter local optima, lacking a guarantee of global optimality. It evaluates attribute subsets based on rough set dependency values. The objective is to reach a state where the search identifies the highest achievable dependency value for the dataset.

3.2 Synthetic and real datasets

To investigate the effect of feature selection empirically, we used 7 synthetic datasets and 55 real datasets, 17 of which were microarray datasets. There are a range of features for each dataset, some of which are binary/discrete and others which are continuous. Using the Equal-width method, continuous features were discretized into 5 bins, while categorical features were left unchanged.

The synthetic datasets used in this work (Table 2) are designed to address a variety of issues, such as an increasing amount of irrelevant features, redundancy, noise, input variations, data nonlinearity, etc. These factors make the task of feature selection methods, which are heavily influenced by them, more complicated.

We also examined 55 real datasets to make significant findings about the impact of feature selection. There were 38 datasets with at least nine features downloaded from the UCI repository (Bache & Linchman, 2013), as well as 17 microarray datasets due to their high dimensionality (Morán-Fernández et al., 2017; Remeseiro & Bolón-Canedo, 2019). Tables 3 and 4 depict key properties of the datasets used in this investigation, such as sample size, number of features and classes.

4 Experimental results

The different experiments consist of comparing the application of each of the seven feature selection approaches individually, as well as random selection (represented as 'Ran' in the tables/figures), which will serve as the comparison baseline. While two of the feature selection methods (CFS and INTERACT) produce a feature subset, the remaining five (IG, ReliefF, MIM, JMI, and mRMR) are ranker methods, requiring a threshold to acquire a subset of features. We chose to keep the top 10%, 20%, and $log2(n)$ of the most significant features in the ordered ranking in this study, where n is the number of features in a given dataset.

Table 2 Summary of the seven synthetic datasets

Dataset	#samples	#features	Relevant features	Correlation	Noise	No linear	#classes
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

It shows the number of samples, the number of features, the relevant features and the number of classes, as well as the presence of correlation, noise and no linearity

Table 3 Summary of the 38 real datasets

Dataset	#sam.	#feat.	#cl.	Dataset	#sam.	#feat.	#cl.
arrhythmia	452	279	13	molec-biol-promoter	106	57	2
bc-wisc-diag	569	30	2	molec-biol-splice	3190	60	3
bc-wisc-prog	198	33	2	musk-2	6598	166	2
breast	569	30	2	optdigits	5620	64	10
coil20	1440	1024	20	ozone	2536	72	2
congress	435	16	2	page-blocks	5473	10	5
conn-bench-sonar	208	60	2	parkinsons	195	22	2
connect-4	67557	42	2	pendigits	10992	16	10
dermatology	366	34	6	satimage	6435	36	6
gisette	7000	5000	2	segmentation	2310	19	7
glass	214	9	6	semeion	1593	256	10
heart	270	13	2	sonar	208	60	2
hill-valley	606	100	2	soybeansmall	47	36	4
ionosphere	351	35	2	spect	267	23	2
isolet	7797	617	2	splice	3175	60	3
krvskp	3196	36	2	USPS	9298	256	10
landstat	5435	36	6	waveform	5000	40	3
libras	360	90	15	wine	178	13	3
low-res-spect	531	100	9	zoo	101	17	7

It shows the number of samples (#sam.), features (#feat.) and classes (#cl.)

Because of the mismatch between dimensionality and sample size in microarray datasets, the thresholds picked the top 5%, 10%, and $log_2(n)$ features, respectively. To estimate the error rate, we used 3×5 cross validation.

The best classifier will not be the same for all datasets, according to the No-Free-Lunch theorem (Wolpert, 1996). As a result, the behavior of feature selection approaches will be evaluated using the classification error acquired from five different classifiers, each of which belongs to a different family. Two linear classifiers (naive Bayes and Support Vector Machine with a linear kernel) and three nonlinear classifiers (C4.5, k-Nearest Neighbor with $k = 3$,

Table 4 Summary of the 17 DNA microarary datasets

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				

It shows the number of samples (#sam.), features (#feat.) and classes (#cl.)

and Random Forest) were used. The Matlab (2022b) and Weka (3.8) tools were used to run the experiments on Windows 10 operating system (Intel(R) Core(TM) i7-4790 CPU @ 3.60GHz 16GB RAM). The QuickReduct algorithm was executed using the package (Scully & Jensen, 2011) for Weka.

4.1 Synthetic datasets

The initial step in determining the efficacy of a feature selection approach should be to use synthetic data, because knowing the optimal features and having the ability to change the experimental conditions allows for more useful conclusions to be drawn. Thus, the experimental findings obtained by the different feature selection approaches over the seven synthetic datasets, depending on the classifier, are reported in this section. To investigate the statistical significance of our classification results, we used a Friedman test with a Nemenyi post-hoc test to examine the classification error. The following figures present the critical different (CD) diagrams, proposed by Demšar (2006), where how 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 so that the lowest (best) ranks are to the right since we perceive the methods on the right side as better.

By working with synthetic datasets, we know what their relevant features are. Therefore, apart from the results obtained by the different feature selection methods and the random selection (Ran), those obtained when the relevant features are used are also presented (labeled in the figures/tables as "Relevant"). Thus, we can see in Fig. 1 that, regardless of the classifier used, the lowest classification errors are obtained when the model is trained with the known relevant features. If we look at the different feature selection methods, INTERACT (INT) seems to be one of the most appropriate for this type of datasets. Besides, if we analyze the results of the univariate methods, MIM obtains competitive results (and sometimes even better) than the multivariate methods. This makes this method an appropriate choice for scenarios where it is important to consider the computational cost. Regarding the threshold of the rankers that achieves lower classification errors, the results are highly variable depending on the classifier, with 10% and the logarithm generally standing out. The synthetic datasets used have a number of relevant features between 2 and 7, far from the average of 25 features that are selected when using the 20% threshold. In this case, irrelevant and/or redundant features are surely being included that make the classification task difficult.

On the other hand, random selection along with thresholds of 10 and 20 percent show the worst classification results. However, it appears that random selection shows competitive results against other feature selection methods when selecting the $log_2(n)$ features of the dataset. Thus, and due to the drawbacks of the traditional tests of contrast of the null hypothesis pointed up by Benavoli et al. (2017), we have chosen to apply the Bayesian hypothesis test (Kuncheva, 2020), in order to analyze the classification results achieved by "Ran-log" and the ranker methods. A previous step is required for this type of study, which is the defining of the Region of Practical Equivalence (Rope). If the mean differences between two approaches for a given metric are smaller than a predefined threshold, they are considered practically equal in practice. In our situation, if the difference in error is less than 1%, we will consider two methods as equivalent.

For the whole benchmark and each pair of methods, we calculate the probability of the three possibilities: (i) with a difference greater than *rope*, random selection (Ran) wins over filter method, (ii) filter method wins over random selection with a difference larger than

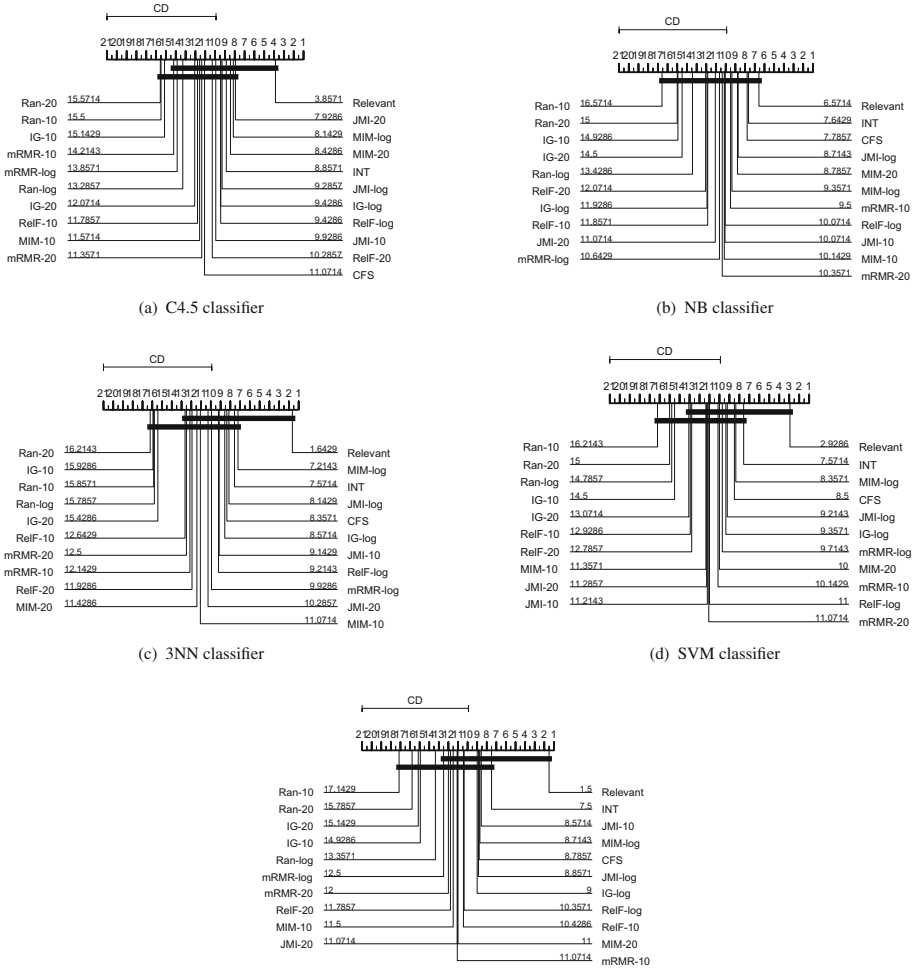
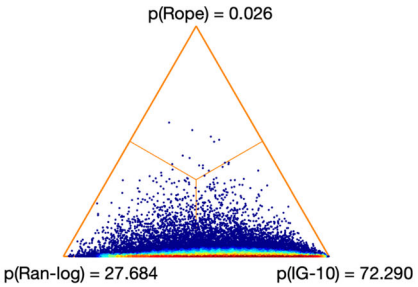


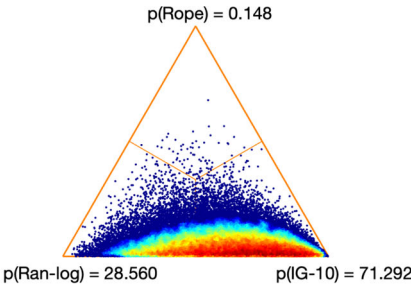
Fig. 1 Critical difference diagrams showing the ranks after applying feature selection over the seven synthetic datasets. For feature selection methods that require a threshold, the option to keep 10% is indicated by '-10', the option to stay with 20% is indicated by '-20', and the option '-log' refers to use log_2

rope, and (iii) the difference between the outcomes is inside the *rope* area. We consider a substantial difference if one of these probabilities is greater than 95%. As a result, using simplex graphs, Fig. 2 depicts the distribution of differences between each pair of methods. As can be seen, although the CD diagrams showed a slight superiority of the random selection together with the 20% threshold compared to the ranker Information Gain (IG), the simplex graphs show that there are no significant differences. In fact, facing only these two methods, the probabilities are 75% skewed to the feature selection method. This may be due to the fact that, in their attempt to compare all the proposed methods, the CD diagrams ignore the individual confrontations carried out by means of the simplex graphs.

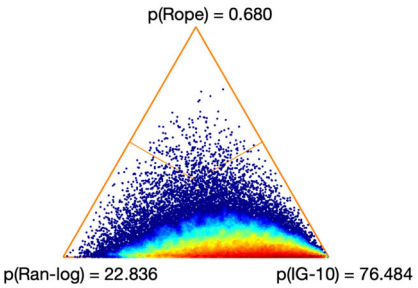
Table 5 displays the classification error obtained by the five classifiers and the eight feature selection methods—the seven filters and random selection—over the seven synthetic



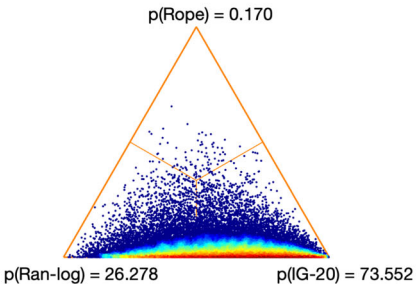
(a) C45 classifier



(b) NB classifier



(c) 3NN classifier



(d) Random Forest classifier

Fig. 2 Simplex graphs for pair comparison of each feature selection method and the baseline random selection (Ran) over the seven real datasets using Bayesian hierarchical tests: random selection (left) and filter method (right)

Table 5 Classification errors obtained by the five classifiers for the seven synthetic datasets tested

	C4.5	NB	3NN	SVM	RF
Relevant	9.04	25.55	10.15	25.63	**3.80**
CFS	30.80	31.17	34.00	33.34	**27.95**
INT	29.74	31.14	32.50	33.09	**26.79**
IG-10	40.50	43.49	45.36	42.49	**39.19**
IG-20	**34.86**	41.71	44.20	39.04	37.57
IG-log	25.94	35.54	28.01	36.10	**23.90**
RelF-10	36.92	42.49	41.53	41.68	**35.51**
RelF-20	**33.72**	38.92	39.49	39.52	34.46
RelF-log	**30.03**	34.77	33.18	36.80	30.15
MIM-10	35.41	37.94	37.72	37.00	**34.51**
MIM-20	**30.25**	35.24	37.46	35.01	31.53
MIM-log	**29.13**	31.67	31.02	32.00	29.17
mRMR-10	37.58	37.9	39.67	36.24	**34.79**
mRMR-20	**31.76**	34.91	37.94	35.62	32.93

Table 5 continued

	C4.5	NB	3NN	SVM	RF
mRMR-log	**32.12**	32.15	35.15	32.72	32.53
JMI-10	34.03	38.61	36.07	38.04	**32.54**
JMI-20	**29.27**	35.65	35.75	35.33	32.69
JMI-log	**28.17**	32.30	31.24	33.49	28.85
Ran-10	57.99	59.79	58.40	**57.72**	59.11
Ran-20	**54.28**	56.18	58.72	56.19	56.36
Ran-log	**52.29**	54.53	55.51	54.23	52.61

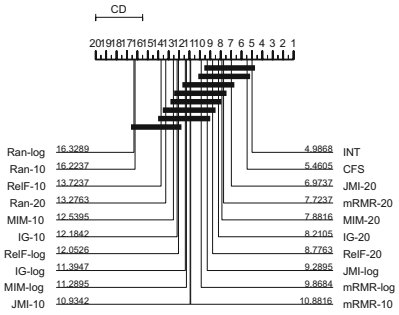
The first row (*Relevant*) corresponds to the error obtained by the model when the known relevant features are used

datasets using the five different classifiers (the lowest classification error obtained for each feature selection method is in bold). As can be seen, the lowest classification errors have been obtained by the non-linear classifiers C4.5 and Random Forest. Let us remember that within the seven synthetic datasets used, three of them (XOR-100, Parity3+3 and Madelon), represent non-linear scenarios. Therefore, and taking into account that SVM and naive Bayes are linear classifiers (a linear kernel is being used for SVM), good results were not expected. Furthermore, it can also be clearly observed that random selection obtains the worst classification results, with hardly any differences across the different classifiers.

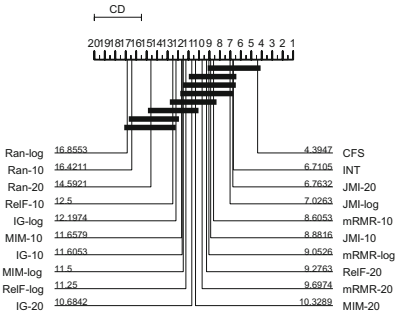
4.2 Real datasets

In this section we will perform experiments on real datasets, to check if the results are similar to those obtained on synthetic data. For this task, we selected a suite of 38 real datasets. CFS and INTERACT, regardless of the classifier used, appear to be the most suitable feature selection methods for this type of datasets, as shown in Fig. 3. Apart from obtaining good results, these two feature selection methods have an added advantage: they do not require to establish a threshold for the number of features to keep. When it comes to ranker methods (which do need a threshold), a percentage of 20% appears to be the best option overall. Moreover, as for the synthetic datasets, the univariate MIM method achieves good results despite its simplicity. Although in this case it only manages to obtain better classification results than the ReliefF multivariate method.

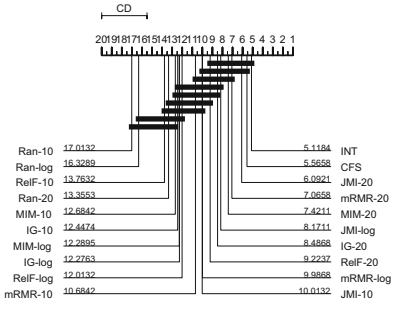
Now, we proceed to compare the results obtained by the feature selection methods tested with the baseline, which we established as the Random selection (Ran). As expected, the random selection is the worst option, when using the thresholds logarithmic and 10%. Nevertheless, when we allow the random selection to keep more features (threshold 20%), it is interesting to see that the random selection is competitive when compared with the other methods. Thus, using simplex graphs as we did with synthetic datasets, Fig. 4 depicts the distribution of differences between random selection (with a 20% threshold) and the ranker methods Information Gain, ReliefF, and MIM (with a 10% threshold). Even although the random selection with a 20% threshold is not statistically significant when compared to the outcomes of numerous ranker methods, it consistently outperforms them. This indicates that the ranker methods (ReliefF, InfoGain, and MIM) are very dependant of the chosen threshold, so a bad choice of threshold produces results that are similar to randomly choosing 20% of features. These findings highlight the importance of choosing a proper threshold, which is



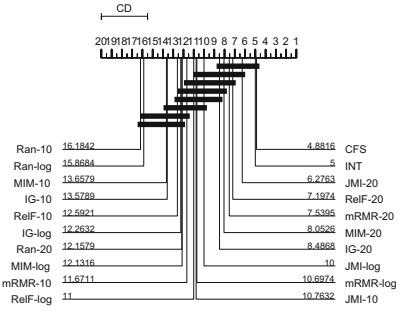
(a) C4.5 classifier



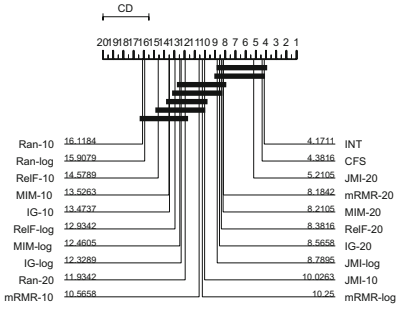
(b) NB classifier



(c) 3NN classifier



(d) SVM classifier

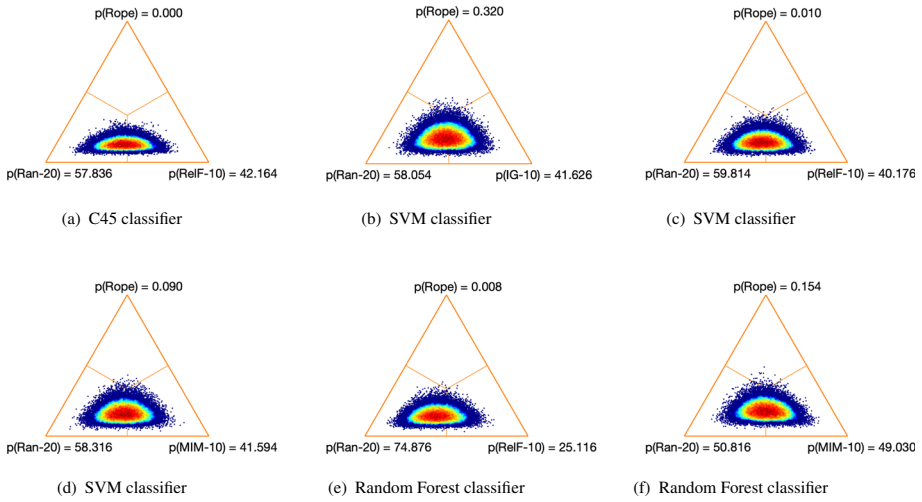


(e) Random Forest classifier

Fig. 3 Critical difference diagrams showing the ranks after applying feature selection over the 38 real datasets. 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

a difficult procedure that is frequently dependent on the problem to solve (and sometimes, even the classifier that is subsequently used).

Table 6 displays the classification error on 38 real datasets, for each classifier and feature selection method (seven filters and the random selection). A total of five classifiers were employed, and lowest classification errors are marked in bold face. Despite the fact that there are no significant differences among the feature selection methods, it is worth highlighting that Random Forest seems to be the best classifier in this setting.



(a) C45 classifier (b) SVM classifier (c) SVM classifier

(d) SVM classifier (e) Random Forest classifier (f) Random Forest classifier

Fig. 4 Simplex graphs for pair comparison of each feature selection method and the baseline random selection (Ran) over the 38 real datasets using Bayesian hierarchical tests: random selection (left) and filter method (right)

Table 6 Classification errors obtained by the five classifiers for the 38 real datasets tested

	C4.5	NB	3NN	SVM	RF
CFS	15.17	18.05	14.83	14.85	**13.06**
INT	15.01	18.87	14.99	14.98	**12.80**
IG-10	22.05	26.51	21.96	24.93	**21.12**
IG-20	18.17	23.52	18.20	19.88	**16.88**
IG-log	21.65	27.30	21.96	25.84	**20.92**
RelF-10	23.66	27.67	23.88	25.13	**22.87**
RelF-20	19.86	24.39	19.84	20.33	**18.11**
RelF-log	23.57	28.12	23.40	26.27	**22.67**
MIM-10	22.08	26.64	22.24	25.08	**21.23**
MIM-20	18.13	23.55	17.92	19.88	**16.69**
MIM-log	21.88	27.37	22.23	26.04	**20.98**
mRMR-10	20.79	24.15	20.64	23.19	**19.56**
mRMR-20	18.10	23.35	17.88	19.66	**16.57**
mRMR-log	19.48	23.79	19.31	22.93	**18.39**
JMI-10	20.34	23.29	19.95	22.44	**19.02**
JMI-20	16.84	20.70	16.40	17.95	**15.05**
JMI-log	18.89	22.43	18.55	21.98	**17.64**
Ran-10	30.34	34.87	30.87	32.08	**29.45**
Ran-20	23.66	29.15	24.12	24.96	**22.13**
Ran-log	29.16	34.66	29.69	32.66	**28.57**

4.2.1 Microrrray datasets

The mismatch between dimensionality and sample size has been seen as a specific issue for machine learning researchers when it comes to DNA microarray classification. Several studies have shown that the majority of genes detected in microarray experiments do not contribute to accurate sample classification (Bolón-Canedo et al., 2014). Feature selection is recommended to avoid the *curse of dimensionality* by identifying the specific genes that improve classification accuracy.

Figure 5 illustrates the critical difference diagrams for each classification algorithm, based on the same study as for the previous datasets, in order to examine the ranks of the feature

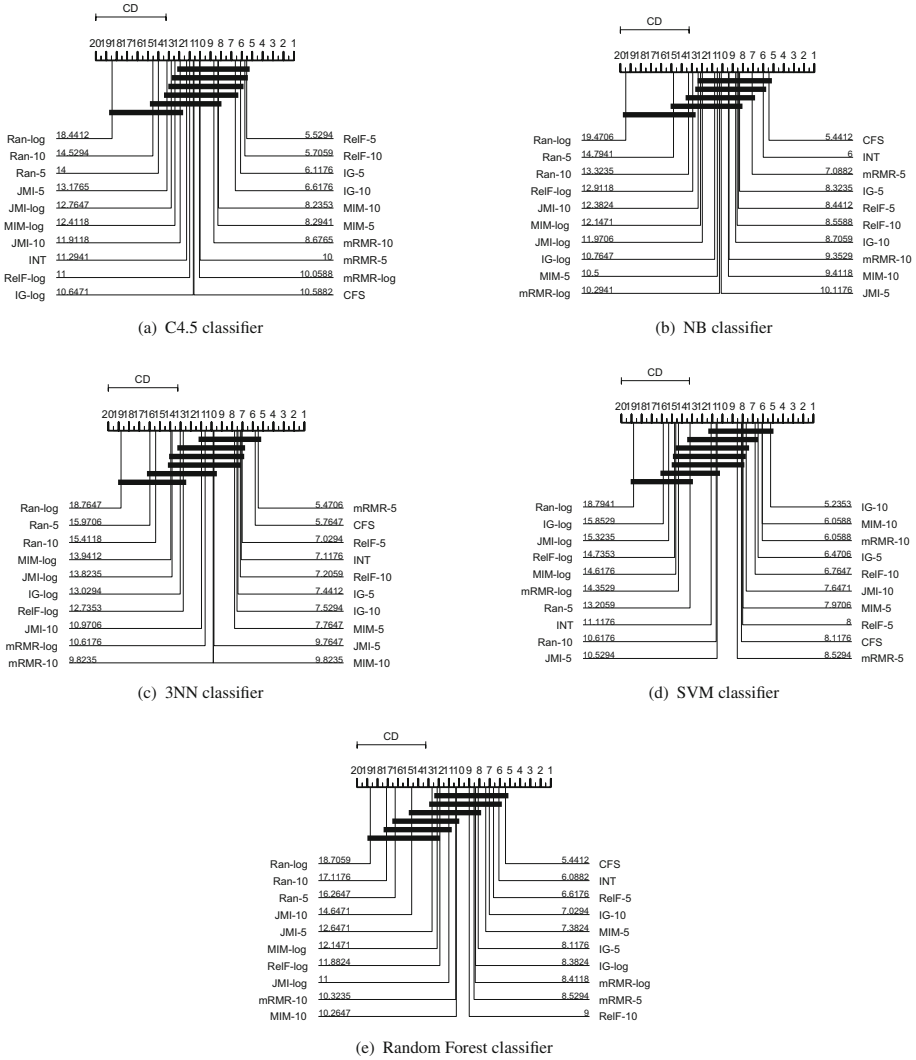


Fig. 5 Critical difference diagram showing the ranks after applying feature selection over the 17 microarray datasets. For feature selection methods that require a threshold, the option to keep 5% is indicated by '-5', the option to stay with 10% is indicated by '-10', and the option '-log' refers to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

selection methods throughout the 17 DNA microarray datasets. The ideal feature selection strategy varies depending on the classifier, as can be seen. In general, though, we can say that CFS is the best option. Regarding the results obtained by the univariate methods, and unlike with the synthetic and real datasets, the IG method seems to work better than MIM, also achieving results similar to those of other more complex multivariate methods. The percentage that maintains 5% of the features appears to be the best fit for these high-dimensional datasets among the many thresholds used by ranker algorithms.

Random selection gives the worst classification accuracy in the C4.5, NB, 3NN, and Random Forest classifiers, both for the thresholds that retain 5 and 10% and for the logarithm, according to the statistical test findings. The results of the SVM reveal a very striking pattern. When the number of features is low (in contrast to the dataset's initial size), this classification strategy appears to perform poorly (Miller, 2002). Remember that if the ranker approaches

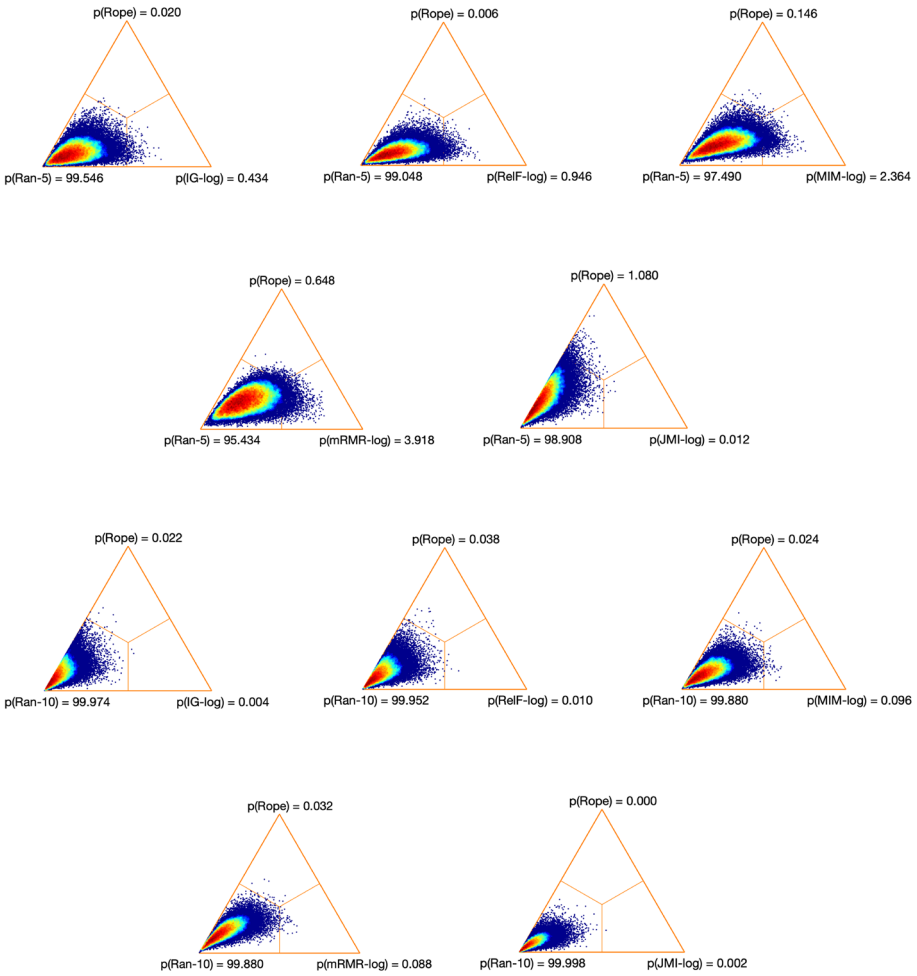


Fig. 6 Simplex graphs for pair comparison of each feature selection method and the baseline random selection (Ran) over the 17 microarray datasets for SVM classifier using Bayesian hierarchical tests: random selection (left) and filter method (right)

utilize a threshold to select the top $log2(n)$ features, the number of features used to train the model for these datasets will be limited to 15 (less than 1% of the original microarray dataset's features). Figure 6 depicts the distribution of the differences between random selection—with 5% and 10% thresholds—and the ranker methods with the logarithm threshold using simplex graphs, just as it does with real datasets. Random selection outperforms ranker methods that keep the top $log2(n)$ features on average and statistically significant, as can be seen. These data illustrate, once again, and much more clearly in this example, that when utilizing ranker methods, an incorrect threshold decision can result in performance comparable to a random selection of features. This is a challenging problem to tackle because the only way to be sure we are using the right threshold is to attempt a large number of them and compute the classification performance for that subset of features, which would result in inadmissible computation times.

The classification error produced by the five classifiers and the eight feature selection methods over the 17 DNA microarray datasets is shown in Table 7 (the lowest error rates highlighted in bold). As mentioned in Navarro (2011), these results demonstrate the superiority in performance of SVM over other classifiers in this domain.

4.3 Case studies

After presenting the experimental results, and before discussing and analyzing them in detail, we will describe several cases of study.

Table 7 Classification errors obtained by the five classifiers for the 17 DNA microarray datasets tested

	C4.5	NB	3NN	SVM	RF
CFS	30.15	19.77	19.49	**17.53**	22.52
INT	30.40	20.26	19.56	**18.46**	22.56
IG-5	27.10	21.98	20.08	**15.88**	23.73
IG-10	27.52	22.05	20.55	**15.73**	23.52
IG-log	30.54	**23.37**	24.73	25.60	23.98
RelF-5	27.46	22.99	19.00	**16.90**	23.16
RelF-10	27.10	23.01	19.04	**16.81**	24.81
RelF-log	31.76	27.24	**25.73**	27.30	26.91
MIM-5	29.08	23.73	20.37	**16.70**	24.40
MIM-10	28.83	22.94	21.15	**15.82**	25.28
MIM-log	31.90	24.95	25.78	**24.86**	27.00
mRMR-5	30.07	21.67	18.92	**16.74**	24.63
mRMR-10	29.45	22.94	21.15	**15.82**	25.97
mRMR-log	30.33	**23.56**	23.71	24.31	24.84
JMI-5	32.72	24.17	23.19	**17.89**	27.77
JMI-10	32.06	25.19	23.68	**16.72**	29.36
JMI-log	32.51	**25.91**	27.21	26.28	27.16
Ran-5	33.00	28.08	28.22	**19.62**	32.08
Ran-10	32.69	26.66	28.11	**17.83**	32.96
Ran-log	43.70	43.00	41.62	41.47	**41.35**

4.3.1 Case study I: Dealing with noise in the inputs

There are various scenarios that can obstruct the feature selection process, including the presence of irrelevant and redundant features, attribute interaction, and data noise. Therefore, in this case study we will analyze the influence of the presence of noise at the input for the Led-25 dataset (see Table 2). The LED dataset requires properly identifying seven LEDs that correspond to values ranging from 0 to 9. The Led-25 dataset was created by adding 17 irrelevant features. Different levels of noise in the inputs (10 and 20 percent) were added to make this dataset more challenging. It is worth noting that, because the features are binary, adding noise results in the wrong value being assigned to the relevant features.

Figure 7 depicts the behavior of feature selection approaches in response to different levels of noise, as measured by the classification error of SVM and Random Forest classifiers. For the rankings methods, only the thresholds of 10 and 20 percent are shown, since in the case of this dataset, the number of features retained by the 20% threshold is the same as that of the logarithm. As we would expect, the classification error grows as the level of noise increases, regardless of the feature selection method used. Furthermore, it is interesting to see that as the noise level at the input is increased, the difference in terms of classification error between the feature selection methods and the random selection is markedly reduced. In fact, when the noise level is 20%, random selection achieves better classification results than various feature selection methods.

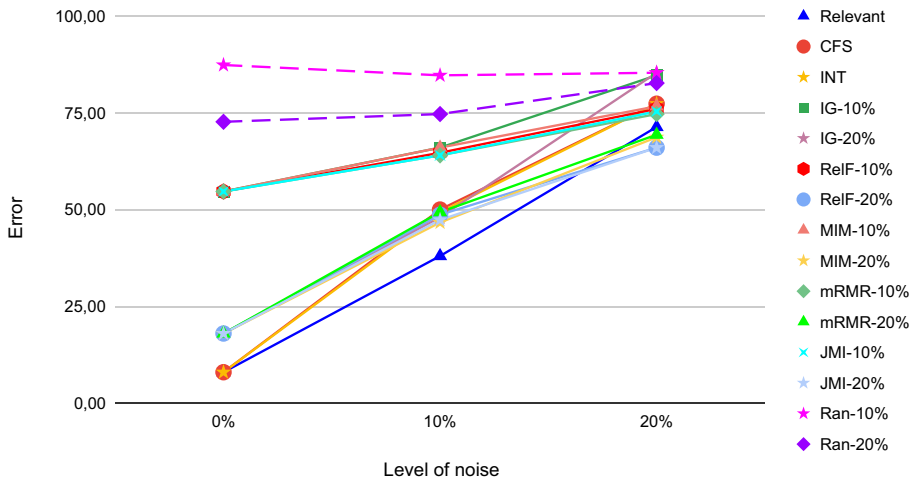
This highlights the low robustness of feature selection methods to noise in the inputs, where the methods most resistant are ReliefF, mRMR, and JMI, while the subsets filters (CFS and INTERACT) and the univariate approach Information Gain are the most affected by noise.

4.3.2 Case study II: Influence of discretization

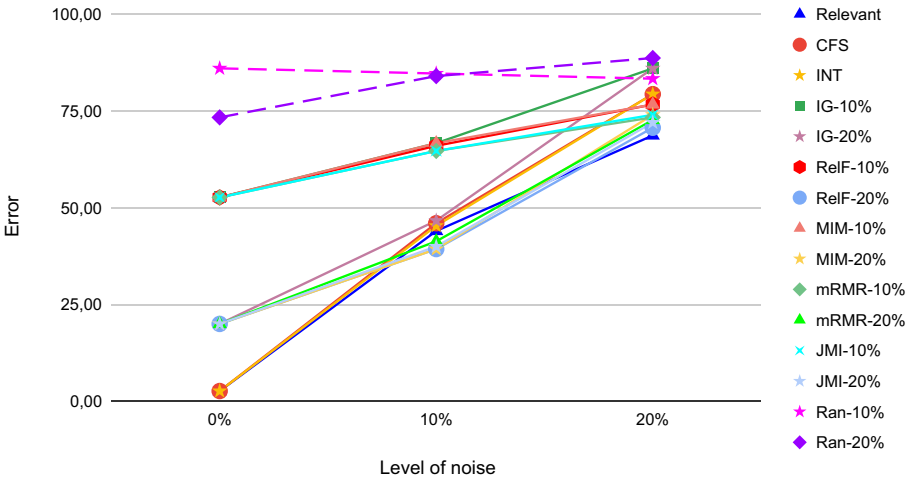
Many feature selection algorithms are designed to handle only discrete data (Bolón-Canedo et al., 2011). To apply these algorithms to numeric features, a common practice is to discretize the data before conducting feature selection. However, the choice of how to group continuous values, the number of intervals to generate, and the positioning of interval cut points on the continuous attribute scale may differ among various discretization methods. Among the discretization methods available in the literature, we opted to use Equal-width due to its widespread popularity. Equal-width divides the number line between v_{min} and v_{max} into b intervals (or bins) of equal width, where b is a user-predefined parameter.

To investigate the impact of discretization, particularly the Equal-width method, on the feature selection process, we will conduct a case study. During this study, we will systematically vary the number of bins, exploring the effects of 5, 10, 15, and 20 bins on the analysis. For this purpose, we choose seven DNA microarray datasets from Table 4, namely 9-tumors, brain-tumor-1, CNS, DLBCL, leukemia-1, SRBCT and TOX-171. Table 8 presents an overview of the classification results obtained. The first observation indicates that the version using 5 bins yields the most favorable outcomes. Concerning the feature selection methods, ReliefF demonstrates superior performance on average, followed closely by CFS and IG. Among the ranking-based methods, the 5% threshold consistently achieves the lowest errors across the five methods, with ReliefF showing a tie with the 10% threshold in this regard.

Finally, SVM stands out as the optimal classifier for this particular type of data, as indicated in Table 9. Therefore, we can affirm that, overall, despite the variation in the number of bins and the observed impact of discretization on feature selection, the conclusions drawn in Section 4.2.1 remain consistent.



(a) SVM classifier



(b) Random Forest classifier

Fig. 7 Classification error (%) for LED-25 dataset with different levels of noise (0%, 10% and 20%). Random selection is indicated by a dashed line

4.3.3 Case study III: CFS vs the rough set attribute method QuickReduct

Rough Set Theory (Pawlak, 1991; Kopczynski & Grzes, 2022) is a formal mathematical technique that aids in reducing dataset dimensionality by quantifying the information content concerning a specific classification. Within rough set theory, the notion of an attribute reduct holds significance, referring to a subset of attributes that, when taken together, effectively retain a specific property of the dataset while ensuring that each attribute, individually, is essential for this preservation. Thus, in order to analyse other feature selection methods

Table 8 Average of the classification errors obtained by the five classifiers on the 7 microarray datasets for the different feature selection methods and number of bins of the Equal-width discretization method

FS method	Equal-width discretization				Average
	5 bins	10 bins	15 bins	20 bins	
CFS	24.07	23.53	23.64	23.44	23.67
INT	25.72	25.09	25.08	24.63	25.13
IG-5	23.11	22.06	23.19	23.55	22.98
IG-10	23.48	23.14	23.31	23.91	23.46
IG-log	30.10	28.75	29.96	28.42	29.31
RelF-5	23.33	23.05	23.24	21.96	22.89
RelF-10	23.18	23.24	22.25	22.89	22.89
RelF-log	30.03	29.61	29.28	27.86	29.20
MIM-5	22.70	23.97	26.65	27.20	25.13
MIM-10	23.99	25.32	25.65	26.29	25.31
MIM-log	29.50	30.39	30.51	31.42	30.45
mRMR-5	23.16	24.73	24.68	25.73	24.58
mRMR-10	23.70	24.84	25.76	26.42	25.18
mRMR-log	27.58	29.43	30.76	32.49	30.06
JMI-5	25.93	27.33	29.20	30.39	28.21
JMI-10	25.87	27.91	29.94	29.65	28.34
JMI-log	28.74	34.01	35.20	39.69	34.41
Average	25.54	26.26	26.96	27.41	

that return a set of features, in this case study we will compare the CFS method, explained above, and the QuickReduct method, belonging to the family of rough set attribute reduction algorithms.

For the experiments, we selected seven DNA microarray datasets from Table 4. The initial part of our analysis presents the classification results obtained by the five previously used classifiers after applying the CFS and QuickReduct feature selection methods, as shown in Table 10. The results demonstrate that in nearly all datasets and classifier scenarios, CFS consistently yields the lowest classification errors, often exhibiting a significant difference compared to QuickReduct. The possible reason behind this can be observed in Table 11, where QuickReduct selects significantly fewer features compared to CFS. The number of features selected by QuickReduct is notably insufficient since the microarray datasets used in this case study have a range of features between 2308 and 7129.

However, while CFS exhibits superiority in terms of the achieved classification results, it comes at the expense of a longer execution time, as evidenced in Table 12.

Table 9 Average of the classification errors obtained by the five classifiers on the 7 microarray datasets and feature selection methods for the different number of bins of the Equal-width discretization method. Lower error rates highlighted in bold

Equal-width discretization	Classifier				
	C4.5	NB	3NN	SVM	RF
5 bins	34.01	24.99	22.75	**18.54**	27.41
10 bins	35.03	24.82	23.98	**19.68**	27.79
15 bins	36.22	25.78	24.58	**19.80**	28.41
20 bins	33.81	26.54	26.16	**20.86**	29.67

Table 10 Classification errors obtained by the five classifiers for the CFS and QuickReduct feature selection methods and the DNA microarray datasets 9-tumors, brain-tumor-1, CNS, DLBCL, leukemia-1, SRBCT and TOX-171. Lower error rates highlighted in bold

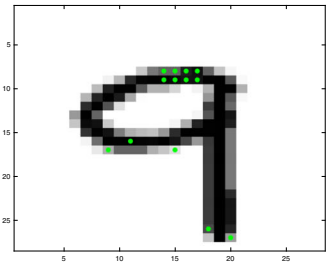
Classifier	FS method	Dataset						
		9-tum	brain-1	CNS	DLBCL	leuk-1	SRBCT	TOX-171
C4.5	CFS	**68.33**	31.11	48.33	**23.56**	**9.81**	**19.49**	**39.75**
	QR	85.00	31.11	**33.33**	40.67	33.43	34.63	49.06
NB	CFS	**55.00**	**18.89**	**38.33**	**0.00**	**6.86**	**1.25**	**23.34**
	QR	66.67	23.33	58.33	40.67	33.14	32.28	47.90
3NN	CFS	**58.33**	20.00	46.67	**6.67**	**6.95**	**0.00**	**14.00**
	QR	80.00	20.00	**35.00**	28.44	33.52	39.71	43.75
SVM	CFS	**46.67**	18.89	41.67	**2.22**	**5.52**	**0.00**	**9.36**
	QR	86.67	18.89	**35.00**	40.89	41.52	44.56	47.28
RF	CFS	**50.00**	**20.00**	43.33	**6.44**	**4.10**	**2.43**	**16.92**
	QR	85.00	21.11	**31.67**	36.67	35.05	29.78	49.66

Table 11 Number of features selected by the CFS and QuickReduct methods for the DNA microarray datasets 9-tumors, brain-tumor-1, CNS, DLBCL, leukemia-1, SRBCT and TOX-171

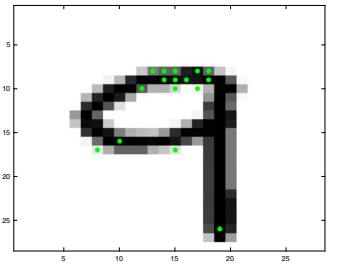
FS method	Dataset						
	9-tum	brain-1	CNS	DLBCL	leuk-1	SRBCT	TOX-171
CFS	47.40	149.20	45.20	61.40	93.80	108.20	115.60
QR	5.80	40.20	1.40	7.40	3.40	2.20	13.80

Table 12 Runtimes (in seconds) for the CFS and QuickReduct methods for the DNA microarray datasets 9-tumors, brain-tumor-1, CNS, DLBCL, leukemia-1, SRBCT and TOX-171

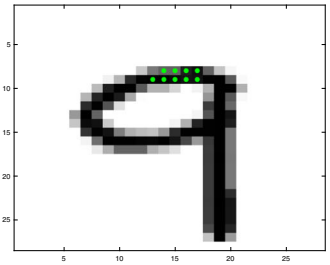
FS method	Dataset						
	9-tum	brain-1	CNS	DLBCL	leuk-1	SRBCT	TOX-171
CFS	248.34	1253.76	398.29	84.02	574.81	115.46	1339.91
QR	141.43	976.07	113.84	75.23	112.10	28.50	864.79



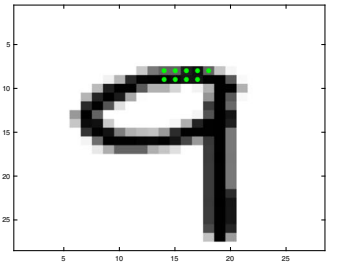
(a) CFS



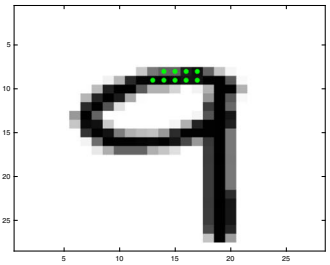
(b) INT



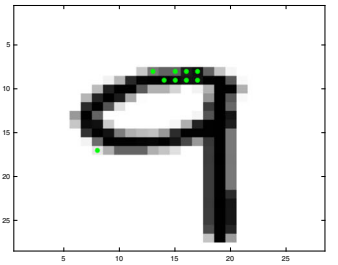
(c) IG-log



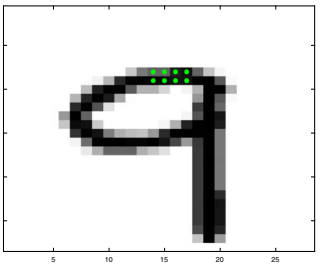
(d) RelF-log



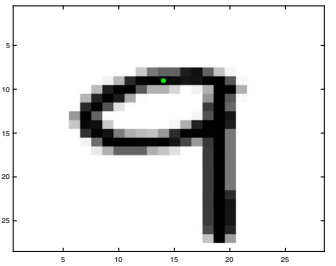
(e) MIM-log



(f) mRMR-log

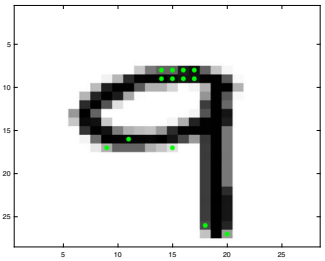


(g) JMI-log

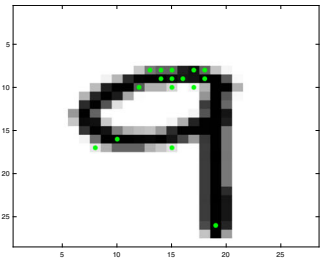


(h) Ran-log

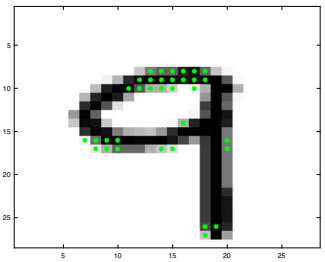
Fig. 8 An example of the use of feature selection (ranker methods with logarithm threshold) for one sample of the class "9" digit. Green dots mark selected features. For the sake of a clear visualization, those features that correspond with pixels that are always in the white area are not marked



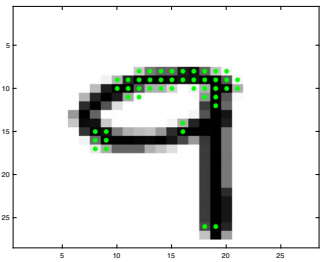
(a) CFS



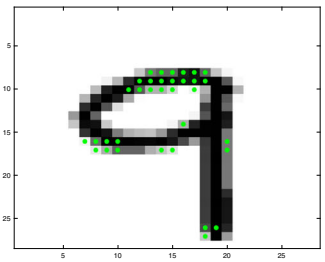
(b) INT



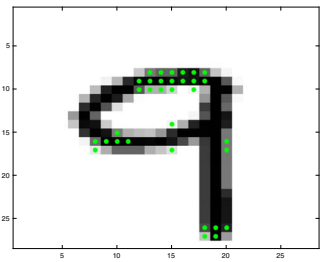
(c) IG-10%



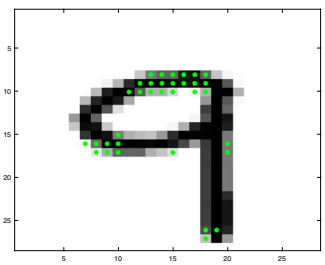
(d) RelF-10%



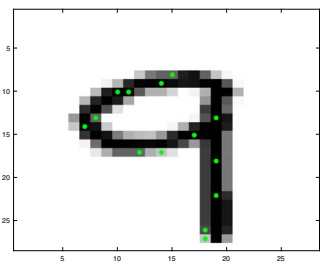
(e) MIM-10%



(f) mRMR-10%



(g) JMI-10%



(h) Ran-10%

Fig. 9 An example of the use of feature selection (ranker methods with 10% threshold) for one sample of the class "9" digit. Green dots mark selected features. For the sake of a clear visualization, those features that correspond with pixels that are always in the white area are not marked

4.3.4 Case study IV: An illustrative example of feature selection over Mnist dataset

In this case of study, we will illustrate the feature selection process over the MNIST dataset (LeCun et al., 1998), in which only two confusable classes are used: digits 4 and 9, because of the small distinctions between them. We thus have 4000 examples per class available. In the original representation, each digit image has 28×28 gray level pixels (784 features).

For these experiments, in the case of the ranker methods, we have used both the 10% and the logarithm for the threshold. Thus, in the following figures we can observe the features selected by the different feature selection methods—marked in green—, as well as by the random selection over the original digit image. As can be seen in Fig. 8, where the logarithm threshold is used for the ranker methods, all feature selection methods select features that illustrate the distinguishable part with the digit 4 (that is, the closed upper part of 9). However, this does not happen in the case of random selection, which fails to define the area, where only one of the 10 selected features falls on the representation of the digit. This makes the classification task to distinguish digits 4 and 9 really complex. When we select the top 10% of features for the ranking methods—i.e. we are left with 78 features—a greater part of the digit is defined (see Fig. 9). In the case of feature selection methods, they continue to select a greater number of features in the area distinguishable with the digit 4 (especially in the case of ReliefF). Meanwhile, the random selection continues to select many features that fall outside the representation of the digit, thus not leaving the upper part of the digit 9 defined.

5 Conclusions and future work

The goal of this research is to thoroughly examine the most common approaches in the field of feature selection, make appropriate comparisons, as well as to determine if there exist some methods that are not able to outperform those results obtained by the random selection. We tested 62 synthetic and real datasets (including the challenging family of DNA microarray datasets) and found that feature selection is effective in general, and that feature selection approaches are superior than random selection in most circumstances, as expected. Our experiments revealed, in particular, that:

- CFS is an excellent choice for any dataset. As a result, when having no knowledge of the specifics of the problem to be solved, we recommend using the CFS method, which has the extra benefit of not requiring the establishment of a threshold. However, if we take into account the computational cost of the feature selection methods used, the univariate filter MIM seems an appropriate choice, which manages to obtain competitive results compared to other more complex multivariate methods.
- Regarding the use of different thresholds, it seems that 10% is more appropriate for the synthetic datasets. For real datasets, the 20% criterion for normal datasets (although worse than the subset approaches, which are the winning option for this type of dataset) and the 5% threshold for microarray datasets appear to be more appropriate. Indeed, when using ranker feature selection methods, the threshold selection is crucial, as our research demonstrated. For some thresholds, in particular, the outcomes were as poor as if some features were chosen at random.
- Despite the fact that the classification results obtained were not significantly different between the feature selection methods used—as discussed in Morán-Fernández et al. (2020)—, we can conclude that Random Forest in the case of synthetic and real datasets and SVM in the case of microarrays were the ones that obtained the best results in terms of

classification precision in a general way across all datasets used, as Fernández-Delgado et al. (2014) concluded in their study.
- With respect to the presence of noise, and as we would have expected, the classification accuracy decreases when the level of noise increases. Besides, the feature selection methods have not proved to be very robust to noise, obtaining classification errors similar to those given by random selection. This highlights the importance of working with quality data.
- Concerning the impact of discretization on feature selection, and particularly in this study, the choice of 5 bins in the Equal-width method yields the most favorable results.

As previously stated, determining an appropriate threshold for ranker-type approaches is a major issue in feature selection that has yet to be solved. As a result, we plan to test a wider number of thresholds in the future, as well as establish an automatic threshold for each dataset type. Another interesting line of research would be to develop feature selection methods more robust to noise, as well as testing other discretization methods to gain further insights into their potential effects on feature selection.

Acknowledgements Not applicable.

Author Contributions All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Laura Morán-Fernández. The first draft of the manuscript was written by Laura Morán-Fernández and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. This research has been financially supported in part by the Spanish Ministerio de Ciencia e Innovación MCIN/AEI/10.13039/501100011033 and "NextGenerationEU"/PRTR under Grants [PID2019-109238GB-C22; TED2021-130599A-I00], and by the Xunta de Galicia (ED431C 2022/44) with the European Union ERDF funds. CITIC, as a Research Center of the University System of Galicia, is funded by Consellería de Educación, Universidade e Formación Profesional of the Xunta de Galicia, Spain through the European Regional Development Fund (ERDF) and the Secretaría Xeral de Universidades (Ref. ED431G 2019/01).

Data Availibility All datasets used in this paper to support the findings are publicly available. Links are reported in the bibliography.

Declarations

Ethical approval Not applicable.

Conflicts of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

Bache, K., & Linchman, M. (2013). UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences. [Online; accessed December 2022]. http://archive.ics.uci.edu/ml/

Benavoli, A., Corani, G., Demšar, J., et al. (2017). Time for a change: a tutorial for comparing multiple classifiers through bayesian analysis. *The Journal of Machine Learning Research, 18*(1), 2653–2688.

Bolón-Canedo, V., Sánchez-Maroño, N., & Alonso-Betanzos, A. (2011). Feature selection and classification in multiple class datasets: An application to kdd cup 99 dataset. *Expert Systems with Applications, 38*(5), 5947–5957. https://doi.org/10.1016/j.eswa.2010.11.028

Bolón-Canedo, V., Sánchez-Maroño, N., & Alonso-Betanzos, A. (2013). A review of feature selection methods on synthetic data. *Knowledge and Information Systems, 34*(3), 483–519. https://doi.org/10.1007/s10115-012-0487-8

Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A., et al. (2014). A review of microarray datasets and applied feature selection methods. *Information Sciences, 282*, 111–135. https://doi.org/10.1016/j.ins.2014.05.042

Bolón-Canedo, V., Sánchez-Maroño, N., & Alonso-Betanzos, A. (2015). Recent advances and emerging challenges of feature selection in the context of big data. *Knowledge-Based Systems, 86*, 33–45. https://doi.org/10.1016/j.knosys.2015.05.014

Chouchoulas, A., & Shen, Q. (2001). Rough set-aided keyword reduction for text categorization. *Applied Artificial Intelligence, 15*(9), 843–873. https://doi.org/10.1080/088395101753210773

Climente-González, H., Azencott, C. A., Kaski, S., et al. (2019). Block hsic lasso: model-free biomarker detection for ultra-high dimensional data. *Bioinformatics, 35*(14), i427–i435. https://doi.org/10.1093/bioinformatics/btz333

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research 7*(Jan), 1–30

Donoho, D. L., et al. (2000). High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Challenges Lecture, 1*(2000), 32.

Fernández-Delgado, M., Cernadas, E., Barro, S., et al. (2014). Do we need hundreds of classifiers to solve real world classification problems? *The Journal of Machine Learning Research, 15*(1), 3133–3181.

Furxhi, I., Murphy, F., Mullins, M., et al. (2020). Nanotoxicology data for in silico tools: a literature review. *Nanotoxicology, 1–26*,. https://doi.org/10.1080/17435390.2020.1729439

Grgic-Hlaca, N., Zafar, M. B., & Gummadi, K. P. et al (2018). Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In: AAAI, (pp. 51–60). https://doi.org/10.1609/aaai.v32i1.11296

Guyon, I., Gunn, S., Nikravesh, M., et al. (2008). Feature extraction: foundations and applications, vol 207. *Springer, New York*. https://doi.org/10.1007/978-3-540-35488-8

Hall, MA. (1999). Correlation-based feature selection for machine learning. PhD thesis, The University of Waikato

Hall, MA., & Smith, L. A. (1998). Practical feature subset selection for machine learning. C McDonald (Ed), Computer Science'98 Proceedings of the 21st Australasian Computer Science Conference ACSC'98

Hughes, G. (1968). On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory, 14*(1), 55–63. https://doi.org/10.1109/TIT.1968.1054102

Kasongo, S. M., & Sun, Y. (2020). A deep learning method with wrapper based feature extraction for wireless intrusion detection system. *Computers & Security, 92*, 101752. https://doi.org/10.1016/j.cose.2020.101752

Kononenko, I. (1994). Estimating attributes: analysis and extensions of relief. In: European conference on machine learning, Springer, 171–182. https://doi.org/10.1007/3-540-57868-4_57

Kopczynski, M., & Grzes, T. (2022). Fpga supported rough set reduct calculation for big datasets. *Journal of Intelligent Information Systems, 59*(3), 779–799. https://doi.org/10.1007/s10844-022-00725-5

Kuncheva, L. I. (2020). Bayesian-analysis-for-comparing-classifiers. https://github.com/LucyKuncheva/Bayesian-Analysis-for-Comparing-Classifiers

LeCun, Y., Cortes, C., Burges, C. (1998). Mnist database of handwritten digits. [Online; accessed December 2022]. http://yann.lecun.com/exdb/mnist/

Lewis, D. D. (1992). Feature selection and feature extraction for text categorization. In: Proceedings of the workshop on Speech and Natural Language, Association for Computational Linguistics, 212–217. https://doi.org/10.3115/1075527.1075574

Miller, A. (2002). *Subset selection in regression*. New York: CRC Press.

Morán-Fernández, L., Bolón-Canedo, V. (2021). Dimensionality reduction: Is feature selection more effective than random selection? In: International Work-Conference on Artificial Neural Networks, Springer, 113–125. https://doi.org/10.1007/978-3-030-85030-2_10

Morán-Fernández, L., Bolón-Canedo, V., & Alonso-Betanzos, A. (2017). Can classification performance be predicted by complexity measures? a study using microarray data. *Knowledge and Information Systems, 51*(3), 1067–1090. https://doi.org/10.1007/s10115-016-1003-3

Morán-Fernández, L., Bolón-Canedo, V., & Alonso-Betanzos, A. (2020). Do we need hundreds of classifiers or a good feature selection? In: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 399–404

Navarro, F. F. G. (2011). Feature selection in cancer research: microarray gene expression and in vivo 1h-mrs domains. PhD thesis, Universitat Politècnica de Catalunya (UPC)

Pawlak, Z. (1991). Rough sets: Theoretical aspects of reasoning about data, vol 9. *Springer Science & Business Media*. https://doi.org/10.1007/978-94-011-3534-4

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*(8), 1226–1238. https://doi.org/10.1109/TPAMI.2005.159

Remeseiro, B., & Bolón-Canedo, V. (2019). A review of feature selection methods in medical applications. *Computers in Biology and Medicine, 112*, 103375. https://doi.org/10.1016/j.compbiomed.2019.103375

Salau, A. O., & Jain, S. (2019). Feature extraction: a survey of the types, techniques, applications. In: 2019 International Conference on Signal Processing and Communication (ICSC), IEEE, 158–164. https://doi.org/10.1109/ICSC45622.2019.8938371

Scully, P. M. D., & Jensen, R. K. (2011). Investigating rough set feature selection for gene expression analysis (BSc Computer Science dissertation). [Online; accessed July 2023]. https://petescully.co.uk/2015/08/28/weka-package-rsarsubseteval/

Shahrjooihaghighi, A., & Frigui, H. (2021). Local feature selection for multiple instance learning. *Journal of Intelligent Information Systems, 1–25*,. https://doi.org/10.1007/s10844-021-00680-7

Shen, Q., & Chouchoulas, A. (2000). A modular approach to generating fuzzy rules with reduced attributes for the monitoring of complex systems. *Engineering Applications of Artificial Intelligence, 13*(3), 263–278. https://doi.org/10.1016/S0952-1976(00)00010-5

Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural Computation, 8*(7), 1341–1390. https://doi.org/10.1162/neco.1996.8.7.1341

Yang, H. H., & Moody, J. (2000). Data visualization and feature selection: New algorithms for nongaussian data. In: Advances in Neural Information Processing Systems, pp 687–693

Zhao, Z., & Liu, H. (2009). Searching for interacting features in subset selection. *Intelligent Data Analysis, 13*(2), 207–228. https://doi.org/10.3233/IDA-2009-0364