CONNECTION SCIENCE, 2016
VOL. 28, NO. 3, 242–257
http://dx.doi.org/10.1080/09540091.2016.1149146

Merging subsets of attributes to improve a hybrid consistency-based filter: a case of study in product unit neural networks

A. J. Tallón-Ballesteros[a], J. C. Riquelme[a] and R. Ruiz[b]

[a]Department of Languages and Computer Systems. University of Seville, Seville, Spain; [b]Area of Computer Science, Pablo de Olavide University, Seville, Spain

ABSTRACT

This paper presents a quality enhancement of the selected features by a hybrid filter-based jointly on feature ranking and feature subset selection (FR-FSS) using a consistency-based measure via merging new features which are obtained applying other FR-FSS evaluated with a correlation metric. The goal is to overcome the accuracy of a neural network classifier containing product units as hidden nodes combined with a feature selection pre-processing step by means of a single consistency-based FR-FSS filter. Neural models are trained with a refined evolutionary programming approach called two-stage evolutionary algorithm. The experimentation has been carried out in eight complex classification problems, seven out of them from UCI (University of California at Irvine) repository and one real-world problem, with high test error rates (around 20%) with powerful classifiers such as 1-nearest neighbour or C4.5. Non-parametric statistical tests revealed that the new proposal significantly improves the accuracy of the neural models.

ARTICLE HISTORY
Received 23 May 2015
Accepted 24 January 2016

KEYWORDS
Artificial neural networks; feature selection; classification; product units; filters; feature subset selection

1. Introduction

A good number of approaches has been proposed to tackle supervised machine learning problems (Dougherty, 2012). Classifiers can be grouped in different categories such as decision trees, classifiers based on nearest neighbours, artificial neural networks, rule-based and Bayes classifiers.

It is straightforward that feature selection (FS) is becoming an important and common activity in most of the current classification problems. Its motivation is treble: simplifying the classifier, improving the accuracy of the classifier; and reducing the dimensionality for the classifier. The last issue is particularly relevant in the context of classifiers based on artificial neural networks.

Our goal is to improve the accuracy of the classification models obtained with product unit neural networks (PUNN) trained with a two-stage evolutionary algorithm. Recently, we have experimented with this classifier by means of a prior pre-processing step with FS (Tallón-Ballesteros, Hervás-Martínez, Riquelme, & Ruiz, 2013) reaching satisfactory results.

CONTACT A. J. Tallón-Ballesteros atallon@us.es

From this starting point we deepen into new possibilities to be applied to our neural classifier.

This paper is organised as follows: Section 2 describes the methodology including FS and evolutionary product unit neural networks; Section 3 introduces the proposal; Section 4 details the experimental process; Section 5 shows and analyses the results obtained; finally, Section 6 states the concluding remarks.

2. Methodology

2.1. Feature selection

Pattern recognition is inherently tied to an information reduction in order to extract the present knowledge inside data. Features can be useful, redundant or irrelevant. An irrelevant feature does not influence on the underlying structure of the data in any way. A redundant feature does not provide anything new in explaining that structure (Guyon & Elisseeff, 2003). FS can be defined as the problem of choosing a small subset of features that is ideally necessary and sufficient to describe the target concept. It can be formulated as a task of searching for an optimal subset of features from all available features (Dash & Liu, 1997). An optimal subset is relative to a certain evaluation function. Typically, an evaluation function tries to measure the discriminating ability of a feature or a subset to distinguish the different class labels.

FS methods can be divided into two broad groups (filter Liu & Setiono, 1996 and wrapper models) based on their use of an inductive algorithm in FS or not (Blum & Langley, 1997). More recently, Liu and Yu (2005) provides a common taxonomy where a large list of FS methods are categorised. Evaluation functions can be grouped into five categories: distance, information (or uncertainty), dependence, consistency and classifier error rate.

Depending on the generation procedure, FS can be divided into individual feature ranking (FR) and feature subset selection (FSS) (Blum & Langley, 1997; Guyon & Elisseeff, 2003). FR measures the feature-class relevance, then rank features by their scores and select the top-ranked ones. These methods are widely used due to its simplicity, scalability, and good empirical success (Golub et al., 1999; Guyon & Elisseeff, 2003). However, FR is criticised because they can only capture the relevance of features to the target concept, but redundancy and basic interactions between features are not discovered, besides, the number of retained features is difficult to determine, a threshold is required. In contrast, FSS attempts to find a set of features with good performance. They integrate the metric to measure the feature-class relevance and the feature–feature interactions. Different algorithms address these issues distinctively. We found different search strategies, namely *exhaustive*, *heuristic* and *random* search, combined with several types of measures to form different algorithms (Dash, Liu, & Motoda, 2000). The time complexity is exponential in terms of data dimensionality for exhaustive search and quadratic for heuristic search. The complexity can be linear to the number of iterations in a random search, but experiments show that in order to find the best feature subset, the number of iterations required is usually at least quadratic to the number of features (Liu & Setiono, 1998).

As well as the categorisation of the previous paragraph, a hybrid model was proposed to handle large data sets to take advantage of the above two approaches (FR, FSS). These methods (Ruiz, Riquelme, & Aguilar-Ruiz, 2006; Yu & Liu, 2004) decouple relevance analysis

and redundancy analysis, and they have been proved to be more effective than ranking methods and more efficient than subset evaluation methods on many traditional high-dimensional data sets. In the context of hybrid FS, BIRS (Best Incremental Ranked Subset) (Ruiz et al., 2006) was the proposed method to obtain relevant features and to remove redundancy. Figure 1 overviews BIRS in the context of the FS taxonomy.

On the one hand, the purpose of a feature subset algorithm is to identify relevant features according to a definition of relevance. However, the notion of relevance in machine learning has not yet been rigorously defined in common agreement (Bell & Wang, 2000). Kohavi and John (1997) include three disjointed categories of feature relevance: strong relevance, weak relevance and irrelevance. Bell and Wang (2000) make use of Information Theory concepts to define the entropic or variable relevance of a feature with respect to the class. Blum and Langley (1997) collect several relevance definitions.

On the other hand, notions of feature redundancy are usually defined in terms of feature correlation. It is widely accepted that two features are redundant to each other if their values are completely correlated. There are two widely used types of measures for the correlation between two variables: linear and nonlinear. In the linear case, the Pearson correlation coefficient is used, and in the nonlinear case, many measures are based on the concept of entropy, or measure of the uncertainty of a random variable. However, it may not be as straightforward in determining feature redundancy when one is correlated with a set of features. Koller and Sahami (1996) apply a technique based on cross-entropy, named Markov blanket filtering, to eliminate redundant features. This idea was formalised using the notion of conditionally independent attribute, which can be defined by several approaches (Xing, Jordan, & Karp, 2001; Yu & Liu, 2004). The minimal-redundancy-maximal-relevance (mRMR) (Peng, Long, & Ding, 2005) metric, which is based on mutual information (Yao, 2003), is also another interesting strategy that is very widespread.

Between filter and wrapper model, there is an intermediate type of methods that are called embedded FS methods (Lal, Chapelle, Weston, & Elisseeff, 2006). It is particularly frequent in the context of support vector machine (SVM) classifier (Vapnik, 2013); a remarkable approach among them is the recursive feature elimination (RFE) (Guyon, Weston, Barnhill, & Vapnik, 2002) selection method that uses SVM classifier to evaluate the goodness of the obtained subset removing at every step a single feature.

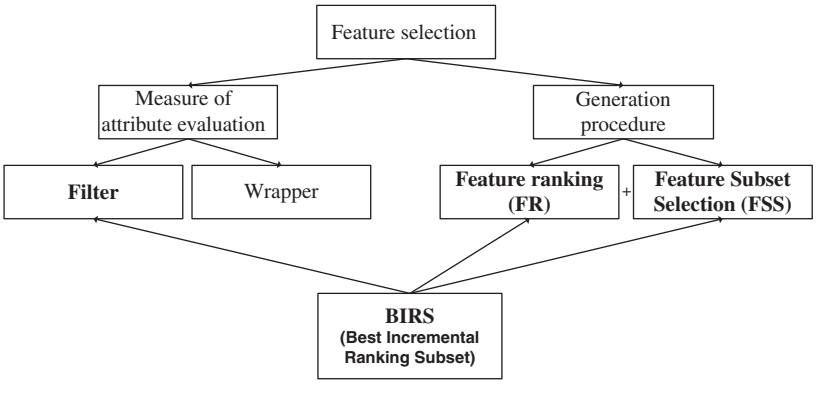

Figure 1. BIRS within FS taxonomy.

The selection process in BIRS is composed of two stages. In stage one, features are evaluated individually, providing a ranking based on a criterion. In stage two, a feature subset evaluator is applied to a certain number of features in the previous ranking following a search strategy. BIRS can include any kind of evaluator in the two phases. In the current work, BIRS utilises as a subset evaluator consistency and correlation concepts, CNS (consistency-based measure (Liu & Setiono, 1996)) and CFS (Correlation-based Feature Selection (Hall, 2000)), respectively, at the second phase, and SOAP (Selecion Of Attributes by Projection (Ruiz, Riquelme, & Aguilar-Ruiz, 2002)) measure and the own subset evaluator at the first phase as a ranking evaluator in the case of CFS.

To guide the search, CFS evaluates the quality of a feature subset taking into account the hypothesis that good feature subsets contain features highly correlated to the class. CNS is based on the consistency measure, which estimates, for a given subset of features, the number of sources that match all but their class labels. The inconsistency rate is then used to assess its quality. SOAP is a deterministic attribute selection criterion based on this basic principle: to count the label changes of examples projected onto each feature. If the attributes are in ascending order according to the number of label changes we will have a list that defines the priority of selection, from greater to smaller importance. The main advantages are its speed and simplicity in the evaluation of the attributes.

BIRS considers that the relevance and the redundancy concepts are included in the following "incremental usefulness" (Caruana & Freitag, 1994) definition: Given a sample of data, an evaluation measure L, a feature space $\mathbf{F}$ and a feature subset $\mathbf{S}$ ($\mathbf{S} \subseteq \mathbf{F}$), the feature F_i is incrementally useful to L with respect to $\mathbf{S}$ if the evaluation of the hypothesis that L produces using the group of features $\{F_i\} \cup \mathbf{S}$ is better than the evaluation achieved using just the subset of features $\mathbf{S}$. That is, if F_i is not incrementally useful to L with respect to $\mathbf{S}$, then the evaluation value given the subset $\mathbf{S}$ is equal or better than the subset evaluation result known $\{F_i\} \cup \mathbf{S}$. It suggests that F_i gives no information beyond what is already in $\mathbf{S}$, therefore, F_i could be removed safely, or in this case, F_i would not be added to $\mathbf{S}$. However, since the computational complexity to determine all possible interactions between features is very high (mainly in high-dimensional domains), BIRS operates using a guided search over an ordered list of attributes.

BIRS deals with the incremental ranked usefulness in order to devise an approach to explicitly identify relevant features and do not bear in mind redundant features. The idea is to choose the feature F_i from a ranked list one by one in the following way: firstly, the features are ranked according to some evaluation measure (SOAP, CFS, CNS); and secondly, BIRS deals with the list of features once, crossing the ranking from the beginning to the last ranked feature. It is obtained the evaluation result using CFS or CNS with the first feature in the list and it is marked as selected. Again, it is obtained the result, but now with the first and second features. The second feature will be marked as selected depending on whether the evaluation obtained is statistically significant better. The process is repeated until the last feature on the ranked list is reached. Finally, the algorithm returns the best subset found, and it can be stated that it will not contain irrelevant or redundant features. Therefore, in the experiments, spBIRS indicates that SOAP will be used as an individual measure in the first part of BIRS and the indicated subset selection as a subset evaluator in the second part. In the same way, cnBIRS denotes that CNS evaluator will be used in both part of the BIRS algorithm.

2.2. Evolutionary product unit neural networks

We have considered a single-hidden-layer feed-forward network architecture, that is, a neural structure with one input layer, one hidden layer and one output layer. We focus on feed-forward neural networks (Bishop, 1995) containing in the hidden layer product units that are neurons with their output is based in the multiplication of terms instead of the sum. Particularly, these terms are expressed in the form of a value pow to a real number. The training of the aforementioned networks has been carried out by a refined evolutionary programming approach (Yao & Liu, 1997) which was named two-stage evolutionary algorithm and introduced in Tallón-Ballesteros and Hervás-Martínez (2011) . Such as an evolutionary programming model, the population is only subjected to replication and mutation operators (Moriarty & Miikkulainen, 1995). Parametric and structural mutations have been used and follow the expressions and details given in the aforementioned paper. The main characteristic of this algorithm is to be based on the use of two populations that are evolved at the beginning of the evolution shortly. Then, the best half of both populations is merged into a new population. On the new intermediate population a full evolutionary cycle is carried out. Table 1 summarises the main TSEA parameters.

In a recent work the TSEAFS (TSEA with FS) framework was proposed (Tallón-Ballesteros et al., 2013). In summary, at the beginning there is a pre-processing step on the training set to obtain a list of selected features. The training of the neural network is performed with the reduced training set via TSEA. Then, the list of selected features is projected into the test set and the performance of the neural models is evaluated in the resulting reduced test set. TSEA does not carry out any kind of FS. Now, we briefly summarise the different configurations and properties of TSEA and TSEAFS. There are two different configurations in TSEA, named $1*$ and $2*$. The TSEAFS features are the following: a) PUNN have been employed, with a number of neurons in the input layer equal to the number of variables in the problem after FS; a hidden layer with a number of nodes that depends on the data set to be classified and the number of selected features; and the number of nodes in the output layer equal to the number of classes minus one because a softmax-type probabilistic approach has been used; b) two experiments have been performed for each problem with two different values for $\boldsymbol{\alpha}_2$, that is associated with the residual of the updating expression of the output-layer coefficients; this parameter controls the diversity of the individuals and has a great impact over the performance (Tallón-Ballesteros & Hervás-Martínez, 2011); c) two different configurations ($1 * \sharp$ and $2 * \sharp$) are applied to subsets obtained with each one of the selectors, for each data set. The parameters of each configuration are *neu*$\sharp$, *gen*$\sharp$ and $\boldsymbol{\alpha}_2$. The first two ones take specific values depending on the data set and the last one depends

Table 1. TSEA general parameters.

Parameter	Value
Population size (N)	1000
Gen-without-improving	20
Interval for the exponents w_{ji}/coefficients β_j^l	$[-5, 5]$
Initial value of α_1	0.5
Initial value of α_2	1
Normalisation of the input data	[1, 2]
Number of nodes in node addition and node deletion operators	[1, 2]

Table 2. Description of the TSEA/TSEAFS configurations.

Methodology	Config.	Num. of neurons in each pop.	Size of each pop.	Num. Gener. in each pop.	α_2
TSEA	1∗	*neu and neu* + 1	1000	0.1 ∗ *gen*	1
TSEA	2∗	*neu and neu* + 1	1000	0.1 ∗ *gen*	1.5
TSEAFS	1 ∗ ♯	*neu*♯ *and neu*♯ + 1	1000	0.1 ∗ *gen*♯	1
TSEAFS	2 ∗ ♯	*neu*♯ *and neu*♯ + 1	1000	0.1 ∗ *gen*♯	1.5

on the configuration number (1 ∗ ♯, . . .). Table 2 shows the main aspects of TSEA/TSEAFS configurations.

3. Proposal

On real-world problems, the attributes may have any kind of main relationship but it is not a fact that this is the single option. So far, the consistency and correlation measures have been considered in an isolated way. The cooperation between different groups of attributes is the key idea of the current paper. According to the literature, the most common measure in the context of FS is the correlation analysis (Hair, 2010). On the other hand, consistency is not so frequent nowadays, basically because in the majority of applications the data are correlated and the detection of inconsistencies (Aggarwal, 2015) is not always the easiest and promising path to classify new patterns in the future.

Now, we describe a motivating example. Lymphography problem (Kononenko, Bratko, & Roskar, 1984) contains data that were obtained from the Institute of Oncology of the University Medical Center in Ljubljana, Yugoslavia about lymph nodes and lymphatic vessels labelled in four classes: *normal find*, *metastases*, *malign lymph* and *fibrosis* (Bramer, 1999). At first sight it seems an interesting data set because there are four classes. The problem was prepared using a hold-out cross validation in two sets: one to train and another to test the classifier performance. We applied a filter based on consistency and the results improved only around a 1% compared to the case without data pre-processing. After that, we used a filter based on correlation and the results were even worse than with all the feature set. In that point, we felt that the analysis of the selected features was necessary. We detected that five attributes were selected by both methods. We taught that the union operator may be a good action because the number of selected attributes would be increased a bit, but the number of selected attributes will be below 15 and it would be worth evaluating our classifier based on product unit neural networks. Surprisingly, the combination succeeded and the performance was a trade-off between increasing the number of features and classifying better.

It has been shown that TSEA performance can be improved by means of FS. Two relevant measures to assess the quality of the features in terms of redundancy and relevance are correlation and consistency. Particularly, the hybrid filter BIRS has an exceptional behaviour in conjunction with TSEA.

This paper proposes to extend the list of selected features by BIRS based on CNS. Among the good number of approaches that can be considered we have chosen CFS using SOAP measure (Tallón-Ballesteros, Hervás-Martínez, Riquelme, & Ruiz, 2011) in order to put in a small number of features. It is extremely important to take into account that the base list of features is obtained by the FR-FSS BIRS using a consistency-based measure. After that some new features may be added if different characteristics are selected with the correlation

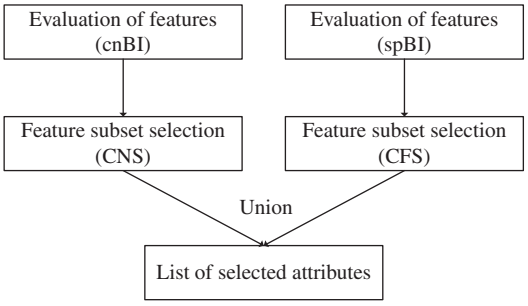

Figure 2. BIRS(CNS+) framework.

measure. In other words, the proposal is the combination of two FR-FSS methods with a good performance in order to get an extended list of attributes. Concretely, we add more attributes to the obtained list by CNS-based FR-FSS BIRS (henceforth, BIRS(CNS)). We have entitled the new proposal BIRS(CNS+) which means that the features selected are those obtained by BIRS using CNS with extra features. Figure 2 depicts the BIRS(CNS+) framework.

Generally speaking, the increase of the number of features would be very small due to the small size of the list obtained by CFS. The new methodology could be applied to any kind of classification problem with any number of features and classes, however the single constraint is that there would not be total overlapping between the lists obtained by each one of the two FR-RSS methods to be combined. In order to meet this condition we recommend that the number of features would be in the original problem at least more than one or two tens. The justification of the proposal is based on the fact that once we choose a evaluation function for a FS method a possible lost of information may happen due the focus is related with the own criterion of the function. Thus, the idea is try to save at least a few more attributes and let the classifier to decide which attributes from this extended subset are included in the classification model.

To the best of our knowledge, there are not contributions that just mix several FR-FSS methods or simply broaden the list of characteristics provided by a FR-FSS filter in any way. A related work belonging to a wrapper model was proposed by Somol, Grim, Novovičová, and Pudil (2011) ; concretely, a k-Nearest Neighbour (k-NN) classifier with four values for *k* was used to determine the inclusion or a removal of single features with a stride value of one in relation with the current feature set. The authors of the aforementioned paper concluded that the actual gain is problem dependent and can not be guaranteed, although the improvement on some data sets is substantial.

4. Experimentation

Table 3 describes the data sets employed together with the parameter values for the base configuration of TSEA and TSEAFS (last two columns). Most of them are publicly available at the UCI repository (Bache & Lichman, 2013) from the University of California. They belong to different domains of application such as Finances, Molecular Biology, Social Sciences, Environment, Oncology, Hand Movement Recognition and Analysis of Olive Oils. The following eight have been used: Statlog (*German* credit), *Labor* Relations, *Libras* Movement, *Lymphography*, Olive Oil (*Olitos*), Molecular Biology (*Promoter* Gene Sequences), Statlog (*Satellite*)

Table 3. Summary of the data sets used and parameter values for TSEA and TSEAFS methodologies.

Data set	Size	Features	Inputs	Classes	Neu; Gen	*Neu♯; Gen♯*
German	1000	22	61	2	6; 500	6; 500
Labor	57	16	29	2	6; 300	5; 300
Libras	360	90	90	15	8; 1000	8; 1000
Lymphography	148	18	38	4	6; 500	6; 100
Olitos	120	25	25	4	6; 300	6; 300
Promoter	106	58	114	2	11; 500	6; 300
Satellite	6435	36	36	6	8; 1000	8; 1000
Water	527	37	37	3	7; 300	7; 300

and *Water* Treatment Plant. Olitos data set is related with olive oil processing and is a complex real-world problem representing different kind of olive oils whose properties can be read in Armanino, Leardi, Lanteri, and Modi (1989) . The size of the data sets ranges from over fifty-five to more than six thousands. The number of features depends on the problem and varies between 16 and 90, while the number of classes is between 2 (for only 3 cases) and 15.

As account of we are using neural networks, all nominal variables have been converted to binary ones; due to this, often the number of inputs is greater than the number of features. Regards the number of inputs it ranges between 26 and 114. Also, the missing values have been replaced in the case of nominal variables by the mode or, when concerning continuous variables, by the mean, taking into account the full data set. These data sets have in common that present important error rates in test phase around 20% or above with reference and robust classifiers such as 1NN or C4.5.

The values of the parameters were chosen with a previous experimental design via a five-fold cross validation with five repetitions using the training set for each data set. For the number of generations, four kind of values were defined (100, 300, 500 and 1000) and in regard to the maximum number of neurons in the hidden layer the range [4–12] were considered.

In relation to the experimental design we have followed a three-fold stratified cross validation (Hjorth, 1993), whereby data set is divided into three parts and subsequently a partition is the test set and the two remaining ones are pooled as the training data. For stochastic algorithms, for each cross validation fold we perform 30 iterations and since we have three folds the results are averaged from 90 runs in order to get a good reliability level. On the other hand, to evaluate the classification models we have chosen the accuracy measure (Kohavi 1995) that can be defined as the probability of correctly classifying a randomly select pattern. Sometimes, it is called as the number of successful hits (Witten, Frank, & Mark, 2011).

Table 4 depicts the FR-FSS methods utilised in the experimentation. There are two ones with and one without FS that belong respectively to TSEAFS and TSEA methodologies. As

Table 4. List of methods employed in the experimentation with and without FS based on FR-FSS.

FR-FSS filter	Ranking method	Subset evaluation	Methodology	Abb.name
–	None	None	TSEA	F0
BIRS(CNS)	cnBIRS	CNS	TSEAFS	F1
BIRS(CNS+)	cnBIRS+spBIRS	CNS+CFS	TSEAFS	F2

Note: cn and sp stand for consistency and SOAP measures, respectively

stated before, feature selectors are FR-FSS filters. Last column defines an abbreviated name for each of them that is employed in next sections.

5. Results

This section details the results obtained, measured in Accuracy or Correct Classification Ratio in the test set or in the test subset depending on that FS has been considered or not. First of all, we present the results obtained, including also the average number of inputs for the three train folds of cross validation, with TSEA and TSEAFS by means of F1 and F2. After that, a statistical analysis compares them to determine whether there are significant differences between applying a combination of two FR-FSS methods and a single FR-FSS. Lastly, we present the results with other additional classifiers together with a global plot of them.

5.1. Results applying TSEA and TSEAFS with F1 and F2

The results obtained by applying the initial TSEA methodology (Tallón-Ballesteros & Hervás-Martínez, 2011), including none FS, are presented, along with those obtained with TSEAFS for the methods F1 and F2. Table 5 shows the average number of inputs, the mean and standard deviation (SD) of the test accuracies for each data set for a total of 90 runs. The best results without and with FS appear in boldface for each data set. It is important to highlight that every data fold has the same number of inputs without any kind of pre-processing via

Table 5. Results obtained in eight data sets applying TSEA and TSEAFS with F1 and F2.

			Mean ± SD	
Data set	Method	Inputs	*Config* 1 ∗ /1 ∗ ♯	*Config* 2 ∗ /2 ∗ ♯
German	F0	61.00	**72.24 ± 2.29**	71.30 ± 2.44
	F1	16.00	72.29 ± 1.59	**72.57 ± 1.30**
	F2	17.33	**73.41 ± 1.61**	72.47 ± 1.36
Labor	F0	29.00	79.77 ± 8.71	**81.58 ± 8.54**
	F1	3.00	**90.58 ± 3.39**	88.83 ± 5.40
	F2	4.67	**91.40 ± 4.35**	90.47 ± 3.94
Libras	F0	90.00	36.67 ± 10.41	**40.25 ± 10.95**
	F1	29.33	**45.68 ± 7.06**	44.98 ± 7.52
	F2	40.00	**47.32 ± 8.53**	47.03 ± 8.83
Lymphography	F0	38.00	**77.09 ± 6.13**	77.02 ± 5.98
	F1	9.33	**76.61 ± 7.46**	75.51 ± 4.70
	F2	11.33	**80.91 ± 7.35**	78.11 ± 5.47
Olitos	F0	25.00	**67.81 ± 6.21**	67.75 ± 7.26
	F1	10.67	**68.81 ± 5.36**	68.67 ± 6.98
	F2	13.00	69.47 ± 6.78	**69.86 ± 4.82**
Promoter	F0	114.00	57.49 ± 8.99	**63.74 ± 9.17**
	F1	7.33	**77.69 ± 5.86**	77.11 ± 3.94
	F2	9.00	76.77 ± 5.32	**77.23 ± 5.02**
Satellite	F0	36.00	**83.48 ± 2.50**	82.08 ± 4.49
	F1	14.67	**83.53 ± 2.62**	83.42 ± 2.40
	F2	25.00	83.14 ± 1.80	**83.74 ± 1.88**
Water	F0	37.00	**83.90 ± 3.53**	83.37 ± 3.15
	F1	12.33	**84.26 ± 3.54**	83.31 ± 2.42
	F2	14.33	83.85 ± 3.10	**84.76 ± 3.80**

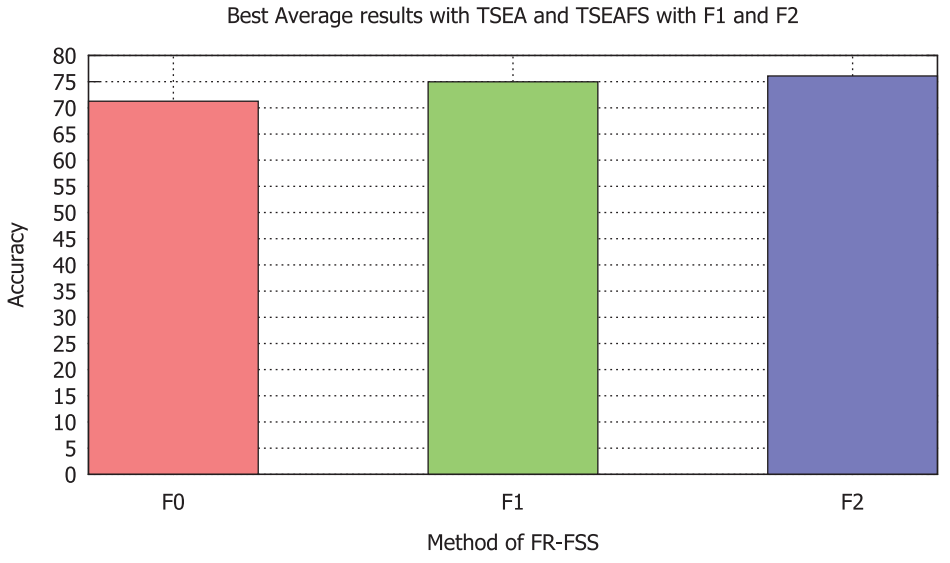

Figure 3. Plot with best average results for TSEA and TSEAFS with F1 and F2.

FS ; nevertheless with F1 and F2 the FS process is conducted for each fold and hence the number of inputs may be different among folds.

From the analysis of the data, it can be concluded, from a purely descriptive point of view, that the TSEAFS methodology with F2 obtains best results for almost all data sets. In most of cases, the SD reduction with F2 versus F1 is clear and it expresses more homogeneous results. We complete the results with a graphical representation of the mean best average results for TSEA and TSEAFS with F1 and F2. As can be seen in Figure 3 F2 improves the performance of F1. Moreover, statistical tests are conducted in the next subsection to detect whether there are significant differences.

5.2. Statistical analysis

We follow the guidelines pointed out by Demsar (2006) to perform non-parametric statistical tests. To determine the statistical significance of the differences in rank observed for two methods with several data sets, a non-parametric test might be used. We apply the Wilcoxon signed-ranks test (1945) which ranks the differences in performances of two classifiers for each data set, ignoring the signs, and compares the ranks for the positive and the negative differences. It is based in the computations of the sum of ranks for positive and negative differences. The difference between two classifiers is significant if the smaller of the sums is equal or less than the critical value (Hollander, Wolfe, & Chicken, 2013).

The test results between F2 (the current proposal) and F1 are summarised in Table 6. R+ and R− are found with the sum of all positive ranks and all negative ranks under the *rank* column, respectively. The minimum of R+ and R− is the T value that in our case is 2. Since there are 8 data sets, the T value at $\alpha = 0.05$ should be less or equal than 4 (the critical value) to reject the null hypothesis. That is, F2 is significantly better than F1.

Table 6. Statistical test results between F1 and F2.

Data set	F2	F1	Difference	Rank
German	73.41	72.57	0.84	5
Labor	91.40	90.58	0.82	4
Libras	47.32	45.68	1.65	7
Lymphography	80.91	76.61	4.31	8
Olitos	69.86	68.81	1.06	6
Promoter	77.23	77.69	−0.46	2
Satellite	83.74	83.53	0.20	1
Water	84.76	84.26	0.50	3
	$T = \min\{34, 2\} = 2$			

5.3. Complexity analysis of the neural models

Once the accuracy results have been compared by means of non-parametric statistical tests, we move on to the complexity analysis in terms of the number of connections that are contained in the neural network models. The best accuracy results for each data set and FS method (F0, F1 and F2) are now extracted from Table 5 and analysed via the number of links. Table 7 represents the average number of connections along with the average test accuracy and the average number of inputs to the neural network classifier for each faced problem. According to the provided information, the number of inputs with F2 has suffered an average increase of around a 30% in contrast to F1. On the other hand, the number of

Table 7. Complexity analysis of the neural models obtained in eight data sets utilising TSEA and TSEAFS with F0, F1 and F2.

Data set	FR-FSS method	Inputs	Av. Accuracy ± SD	Av. no. connections ± SD
German	F0	61.00	72.24 ± 2.29	89.81 ± 18.75
	F1	16.00	72.57 ± 1.30	45.20 ± 9.55
	F2	17.33	73.41 ± 1.61	43.61 ± 10.90
Labor	F0	29.00	81.58 ± 8.54	45.86 ± 10.79
	F1	3.00	90.58 ± 3.39	16.77 ± 1.82
	F2	4.67	91.40 ± 4.35	17.37 ± 2.47
Libras	F0	90.00	40.25 ± 10.95	279.37 ± 63.29
	F1	29.33	45.68 ± 7.06	148.57 ± 16.14
	F2	40.00	47.32 ± 8.53	188.53 ± 17.72
Lymphography	F0	38.00	77.09 ± 6.13	76.36 ± 13.85
	F1	9.33	76.61 ± 7.46	42.51 ± 4.14
	F2	11.33	80.91 ± 7.35	42.46 ± 4.25
Olitos	F0	25.00	67.81 ± 6.21	63.89 ± 11.37
	F1	10.67	68.81 ± 5.36	48.78 ± 6.70
	F2	13.00	69.86 ± 4.82	49.48 ± 8.73
Promoter	F0	114.00	63.74 ± 9.17	502.80 ± 54.35
	F1	7.33	77.69 ± 5.86	27.83 ± 3.65
	F2	9.00	77.23 ± 5.02	26.67 ± 3.95
Satellite	F0	36.00	83.48 ± 2.50	118.78 ± 16.94
	F1	14.67	83.53 ± 2.62	79.92 ± 7.90
	F2	25.00	83.74 ± 1.88	91.52 ± 9.38
Water	F0	37.00	83.90 ± 3.53	68.89 ± 11.15
	F1	12.33	84.26 ± 3.54	50.97 ± 6.47
	F2	14.33	84.76 ± 3.80	41.62 ± 7.37
Average	F0	53.75	71.26 ± 6.17	155.72 ± 25.06
	F1	12.83	74.97 ± 4.57	57.57 ± 7.04
	F2	16.83	76.08 ± 4.67	62.66 ± 8.10

connections have been only raised in a 10%. The number of connections after the application of the FS via F2 is almost a 60% lower than the number of links without any kind of filtering. We have to bear in mind that the input size of the initial situation has been reduced in a very outstanding way, about a 75% with F2 filter.

5.4. Results obtained with a variety of filters and classifiers

Now, a comparison, applying the current proposal (F2), the baseline FR-FSS (F1), mRMR and SVM-RFE, is performed between TSEAFS and other machine learning algorithms. These methods are C4.5, k-NN, -where k is 1-, PART (Frank & Witten, 1998) and SVM. Since these methods are implemented in Weka tool (Frank et al., 2010), we have conducted the experiments and used the same cross-validation, thus the same instances in each of the partitions, that in the reported results in previous sections.

Regarding the parameters, the algorithms have been run with the Weka default values because those are the recommended ones by the own authors of the algorithms.

mRMR and SVM-RFE require to prompt a threshold or the number of attributes to be selected. Our choice has been to consider a common relative value of attributes for all the data sets, that is to select the 70% of the initial number of inputs. In addition, for SVM-RFE the parameter associated with the complexity (*C*) has been set to 0.

We have outlined in Table 8 the average results with F1, F2, mRMR and SVM-RFE for each data set and algorithm. Due to we have used FR-FSS for the FS, the same reduced features set is applied to all classifiers. Concerning TSEAFS, on the one hand for F1 and F2 is reported the best mean value of the two configurations shown in Table 5; for mRMR and SVM-RFE we have tested two configurations with the same parameters that have been used without FS and we have only depicted the results of the best configuration. Really, the number of inputs in the case of mRMR and SVM-RFE is just closer to the starting situation and a training of the neural models with a number of neurons similar to those related F1 and F2 could not learn enough to be able to generalise in an appropriate way.

From an analysis of the results, we can assert the following. Generally speaking, F2 has a higher performance than F1; with the exceptions of TSEAFS and C4.5, the improvements are not very strong. mRMR works very fine with SVM and PART. SVM-RFE is a good option to be combined with C4.5. Next, we analyse every classifier one by one. C4.5 gets the best average results with SVM-RFE followed by F2; according to punctual results SVM-RFE wins in three problems followed by two times for F2. Classifier 1NN achieves, in average, the best results with F2 and after that F1; individually, F2 and F1 achieves the best performance in one data set. PART algorithm sheds light on the best average results with mRMR, followed by F2; mRMR gets the best results for three problems and F2 for two out of eight data set included in the test-bed. SVM operates excellently with mRMR and good with SVM-RFE; in contrast, the individual performance is in the other way round because SVM-RFE wins four times by the three times of mRMR. From the point of view of the average feature subset size, the lower number of attributes is caught by F1 and F2; on the contrary, SVM-RFE and mRMR select the highest number of features but alternatively the input parameter is customisable to have a greater or lower number of attributes.

Now, we move on to the global perspective of the top best individual results for each data set. The pair (SVM-RFE, SVM) achieves the global best results for *German* and *Lymphography* problems. The combination of mRMR with SVM get the best behaviours for *Olitos* and

Table 8. Global test accuracy results for several classifiers with the FR-FSSs F1 and F2, mRMR and SVM-RFE.

Data set	Filter method	Av. feature subset size	C4.5	1NN	PART	SVM	TSEAFS
German	F1	16.00	71.30 ± 2.24	69.50 ± 1.30	69.40 ± 0.90	73.50 ± 0.79	72.57 ± 1.30
	F2	17.33	70.60 ± 3.12	68.90 ± 2.25	71.70 ± 1.91	73.00 ± 0.64	73.41 ± 1.61
	mRMR	43.00	71.97 ± 3.24	68.57 ± 0.46	71.17 ± 2.57	74.98 ± 0.87	72.38 ± 1.71
	SVM-RFE	43.00	71.17 ± 4.20	69.67 ± 2.35	69.37 ± 1.20	75.18 ± 0.69	72.08 ± 1.62
Labor	F1	3.00	85.96 ± 6.08	89.47 ± 5.26	85.96 ± 6.08	92.98 ± 3.04	90.58 ± 3.39
	F2	4.67	85.96 ± 6.08	94.74 ± 5.26	85.96 ± 6.08	92.98 ± 3.04	91.40 ± 4.35
	mRMR	20.00	85.96 ± 6.08	91.23 ± 6.08	85.96 ± 6.08	91.23 ± 8.04	89.70 ± 5.40
	SVM-RFE	20.00	87.72 ± 3.04	92.98 ± 3.04	85.96 ± 3.04	92.98 ± 3.04	90.05 ± 6.98
Libras	F1	29.33	37.78 ± 9.73	64.72 ± 4.35	44.17 ± 10.93	53.06 ± 10.29	45.68 ± 7.06
	F2	40.00	46.11 ± 10.88	66.67 ± 15.43	41.39 ± 11.38	58.89 ± 10.62	47.32 ± 8.53
	mRMR	63.00	46.67 ± 12.01	67.22 ± 14.82	45.00 ± 12.83	56.94 ± 12.14	33.41 ± 9.67
	SVM-RFE	63.00	51.67 ± 14.17	67.50 ± 16.73	42.50 ± 8.21	60.56 ± 14.49	34.50 ± 10.23
Lymphography	F1	9.33	71.59 ± 9.01	77.69 ± 2.25	76.98 ± 6.08	81.05 ± 6.67	76.61 ± 7.46
	F2	11.33	72.95 ± 6.67	75.65 ± 3.77	75.62 ± 7.54	81.05 ± 4.41	80.91 ± 7.35
	mRMR	27.00	72.11 ± 7.17	79.59 ± 5.40	77.55 ± 8.90	82.99 ± 5.14	76.46 ± 6.87
	SVM-RFE	27.00	75.51 ± 9.35	80.95 ± 6.56	76.87 ± 6.56	83.67 ± 7.36	75.94 ± 6.33
Olitos	F1	10.67	55.83 ± 3.82	72.50 ± 2.50	60.83 ± 1.44	71.67 ± 3.82	68.81 ± 5.36
	F2	13.00	55.00 ± 4.33	70.83 ± 1.44	61.67 ± 1.44	75.83 ± 3.82	69.86 ± 4.82
	mRMR	18.00	58.33 ± 1.44	75.83 ± 7.64	63.33 ± 8.04	85.00 ± 4.33	65.33 ± 7.54
	SVM-RFE	18.00	54.17 ± 3.82	65.83 ± 7.22	62.50 ± 12.99	79.17 ± 5.20	67.30 ± 7.50
Promoter	F1	7.33	76.43 ± 1.24	82.14 ± 8.42	78.33 ± 2.89	77.43 ± 7.06	77.69 ± 5.86
	F2	9.00	78.33 ± 4.06	79.29 ± 3.98	77.41 ± 4.49	79.26 ± 1.28	77.23 ± 5.02
	mRMR	80.00	71.43 ± 10.30	69.52 ± 3.30	74.29 ± 2.86	87.62 ± 3.30	74.31 ± 6.08
	SVM-RFE	80.00	73.33 ± 5.95	74.29 ± 4.95	70.48 ± 10.03	81.91 ± 5.95	75.34 ± 5.91
Satellite	F1	14.67	83.67 ± 2.52	86.28 ± 1.06	83.64 ± 2.15	69.40 ± 0.90	83.53 ± 2.62
	F2	25.00	84.13 ± 2.70	87.60 ± 0.81	84.17 ± 1.93	71.70 ± 1.91	83.74 ± 1.88
	mRMR	25.00	83.96 ± 2.30	87.42 ± 1.12	84.59 ± 2.10	84.14 ± 1.20	82.09 ± 2.33
	SVM-RFE	25.00	84.11 ± 3.03	87.88 ± 1.05	84.64 ± 2.19	84.53 ± 1.83	82.49 ± 2.14
Water	F1	12.33	81.39 ± 2.55	81.38 ± 0.31	80.42 ± 1.73	82.35 ± 2.90	84.26 ± 3.54
	F2	14.33	81.00 ± 3.16	81.77 ± 2.85	82.15 ± 2.25	82.93 ± 4.61	84.76 ± 3.80
	mRMR	26.00	77.78 ± 1.76	81.99 ± 1.45	80.84 ± 3.32	83.91 ± 4.70	82.40 ± 1.59
	SVM-RFE	26.00	80.65 ± 2.02	82.18 ± 3.20	78.35 ± 1.33	83.53 ± 4.24	82.17 ± 1.61
Average	F1	12.83	70.49 ± 4.65	77.96 ± 3.18	72.47 ± 4.03	75.18 ± 4.43	74.97 ± 4.57
	F2	16.83	71.76 ± 5.13	78.18 ± 4.47	72.51 ± 4.63	76.96 ± 3.79	76.08 ± 4.67
	mRMR	37.75	71.03 ± 5.54	77.67 ± 5.03	72.84 ± 5.83	80.85 ± 4.97	72.01 ± 5.15
	SVM-RFE	37.75	72.29 ± 5.70	77.66 ± 5.64	71.33 ± 5.69	80.19 ± 5.35	72.48 ± 5.29

Promoter data sets. SVM-RFE together with 1NN reaches the best results with *Libras* and *Satellite* problems. The couple F2 and 1NN is the best approach for *Labor* data set. Lastly, F2 and TSEA is the most suitable tandem for *Water* problem.

SVM-RFE with SVM gets the best results in the half of the test-bed which indicates that is a very interesting approach but we have to take into account that the setting is not fully automatic and the best or worst performance may be influenced by the input value; although SVM-RFE is an embedded method the behaviour is not remarkably better than other general purpose filters like mRMR. F2 with TSEA achieves the best results in seven out of eight problems which is a very outstanding fact. The remaining classifiers may have good punctual performances with mRMR, SVM-RFE, F2 or F1 but the trend is neither clear nor homogeneous.

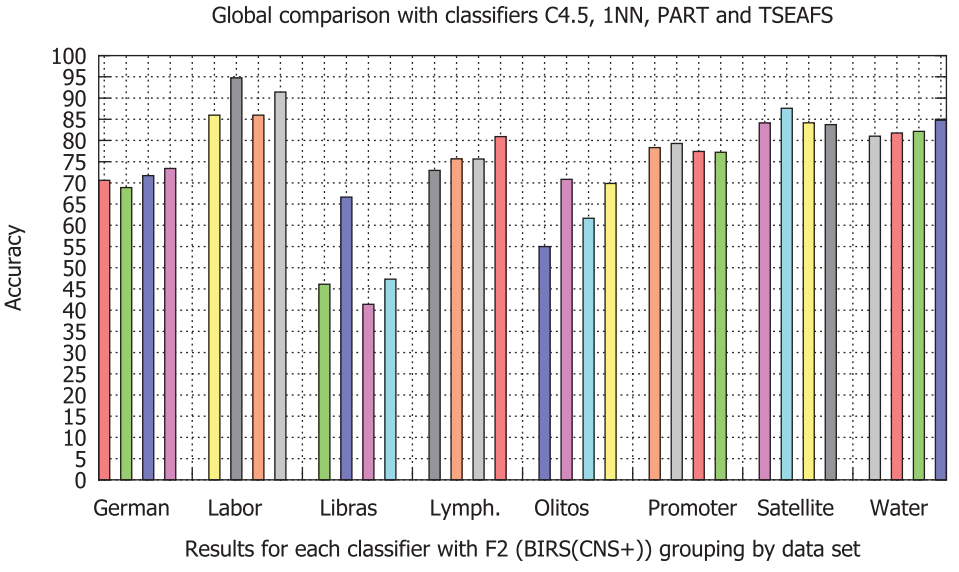

Figure 4. Plot with global average results for a good number of classifiers with F2.

It has been shown that F2 is not very appropriate for SVM. For this reason, it will be excluded in the next summarising plot about F2. Figure 4 shows a box-plot with the average results for C4.5, 1NN, PART and TSEAFS with F2 grouping by data set. As can be seen 1NN and TSEAFS obtain the most relevant results. PART in some cases has a better behaviour than C4.5.

6. Conclusions

This paper introduced an approach based on broadening the subset of attributes obtained with a FR-FSS filter based on consistency via adding some few more attributes that are determined with other FR-FSS using a correlation measure. The experimental results and the statistical test shown that the accuracy improvement is significant in relation with the base approach in the context of product unit neural networks trained with an evolutionary programming algorithm. The proposed FS algorithm has been compared with mRMR and SVM-RFE. The new contribution exhibits a better performance than the last two aforementioned methods with product unit networks. Also other kinds of classifiers, such as C4.5, 1NN, PART and SVM were tested. A relevant conclusion is that the new approach may sometimes be helpful for C4.5 and PART classifiers. According to the reported results, consistency and correlation measures might be considered to a certain extent as complementary metrics that could operate in cooperation.

Authors believe that the presented approach will be challenging to data mining practitioners in order to try to evaluate similar approaches to their own classifiers. As future work, it may be interesting to apply the current proposal, and then other kind of pre-processing technique taking into account that a more limited lost of information than usual is caused by the FS method itself. The empirical study were conducted on eight binary and multi-class classification problems on real-world applications with test error rates around a 20% in the best cases without any kind of pre-processing via FS.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work has been partially subsidised by TIN2011-28956-C02-02 and TIN2014-55894-C2-R projects of the Spanish Inter-Ministerial Commission of Science and Technology (MICYT), FEDER funds and P11-TIC-7528 projects of the "Junta de Andalucía" (Spain).

References

Aggarwal, C. C. (2015). Data preparation. In C. C. Aggarwal (Ed.), *Data mining* (pp. 27–62). Cham: Springer.

Armanino, C., Leardi, R., Lanteri, S., & Modi, G. (1989). Chemometric analysis of Tuscan olive oils. *Chemometrics and Intelligent Laboratory Systems*, *5*(4), 343–354.

Bache, K., & Lichman, M. (2013). *UCI machine learning repository*. Retrieved from http://archive.ics.uci.edu/ml

Bell, D., & Wang, H. (2000). A formalism for relevance and its application in feature subset selection. *Machine Learning*, *41*(2), 175–195.

Bishop, C. (1995). *Neural networks for pattern recognition*. New York, NY: Oxford University Press.

Blum, A. L., & Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence*, *97*(1–2), 245–271.

Bramer, M. A. (1999). *Knowledge discovery and data mining* (No. 1). London: Iet.

Caruana, R., & Freitag, D. (1994). How useful is relevance? In R. Greiner and D. Subramanian (Eds.), *Working notes of the AAAI fall symp. on relevance* (pp. 25–29). N. Orleans, LA: AAAI Press.

Dash, M., & Liu, H. (1997). Feature selection for classification. *Intelligent Data Analisys*, *1*(3), 131–156.

Dash, M., Liu, H., & Motoda, H. (2000). Consistency based feature selection. In P. Langley (Ed.), *Pacific-Asia conference on knowledge discovery and data mining* (pp. 98–109). Kyoto: Morgan Kaufmann.

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

Dougherty, G. (2012). *Pattern recognition and classification: An introduction*. New York, NY: Springer.

Frank, E., Hall, M., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I., & Trigg, L. (2010). Weka-a machine learning workbench for data mining. In O. Maimon & L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (pp. 1269–1277). New York, NY: Springer.

Frank, E., & Witten, I.H. (1998). *Generating accurate rule sets without global optimization*. Proceedings of the fifteenth International Conference on Machine Learning (ICML 1998). San Francisco, CA: Morgan Kaufmann, pp. 144–151.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., . . . Lander, E. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, *286*, 531–537.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, *46*(1–3), 389–422.

Hair, J. F. Jr, Black, W. C., Babin, B. J., & Anderson, R. E. (2010). *Multivariate data analysis*. Upper Saddle River: Prentice Hall.

Hall, M. (2000). Correlation-based feature selection for discrete and numeric class machine learning. In P. Langley (Ed.), *Proceedings of the seventeenth international conference on machine learning (ICML 2000)* (pp. 359–366). San Francisco, CA: Morgan Kaufmann.

Hjorth, J. (1993). *Computer intensive statistical methods: Validation, model selection, and bootstrap*. New York, NY: CRC Press.

Hollander, M., Wolfe, D., & Chicken, E. (2013). *Nonparametric statistical methods* (Vol. 751). Hoboken: John Wiley & Sons.

Kohavi, R. (1995). *A study of cross-validation and bootstrap for accuracy estimation and model selection*. Proceedings of the fourteenth International Joint Conference on Artificial Intelligence (IJCAI 1995), Vol. 2, Montreal, pp. 1137–1145.

Kohavi, R., & John, G. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, *1–2*, 273–324.

Koller, D., & Sahami, M. (1996). Toward optimal feature selection. In L. Saitta (Ed.), *13th International conference on machine learning* (pp. 284–292). Bari: Morgan Kaufmann.

Kononenko, I., Bratko, I., & Roskar, E. (1984). *Experiments in automatic learning of medical diagnostic rules*. Technical Report, Jozef Stefan Institute, Ljubljana, Yugoslavia.

Lal, T. N., Chapelle, O., Weston, J., & Elisseeff, A. (2006). Embedded methods. In I. Gyon, M. Nikravesh, S. Gunn, & L. A. Zadeh (Eds.), *Feature extraction* (pp. 137–165). Berlin: Springer.

Liu, H., & Setiono, R. (1996). A probabilistic approach to feature selection – a filter solution. In L. Saitta (Ed.), *Proceedings of the thirteenth International Conference on Machine Learning (ICML 1996)* (pp. 319–327). Italy: Morgan Kaufmann.

Liu, H., & Setiono, R. (1998). Incremental feature selection. *Applied Intelligence*, *9*(3), 217–230.

Liu, H., & Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, *17*(3), 1–12.

Moriarty, D., & Miikkulainen, R. (1995). Discovering complex othello strategies through evolutionary neural networks. *Connection Science*, *7*(3–4), 195–210.

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.

Ruiz, R., Riquelme, J., & Aguilar-Ruiz, J. (2002). Projection-based measure for efficient feature selection. *Journal of Intelligent and Fuzzy Systems*, *12*(3–4), 175–183.

Ruiz, R., Riquelme, J., & Aguilar-Ruiz, J. (2006). Incremental wrapper-based gene selection from microarray expression data for cancer classification. *Pattern Recognition*, *39*(12), 2383–2392.

Somol, P., Grim, J., Novovičová, J., & Pudil, P. (2011). Improving feature selection process resistance to failures caused by curse-of-dimensionality effects. *Kybernetika*, *47*(3), 401–425.

Tallón-Ballesteros, A. J., & Hervás-Martínez, C. (2011). A two-stage algorithm in evolutionary product unit neural networks for classification. *Expert Systems with Applications*, *38*(1), 743–754.

Tallón-Ballesteros, A., Hervás-Martínez, C., Riquelme, J., & Ruiz, R. (2011). Improving the accuracy of a two-stage algorithm in evolutionary product unit neural networks for classification by means of feature selection. In J. M. Ferrández, J. R. Álvarez Sánchez, F. de la Paz, & F. J. Toledo (Eds.), *New challenges on bioinspired applications* (pp. 381–390). Heidelberg: Springer.

Tallón-Ballesteros, A. J., Hervás-Martínez, C., Riquelme, J., & Ruiz, R. (2013). Feature selection to enhance a two-stage evolutionary algorithm in product unit neural networks for complex classification problems. *Neurocomputing*, *114*, 107–117.

Vapnik, V. (2013). *The nature of statistical learning theory*. New York: Springer Science & Business Media.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics*, *1*(6), 80–83.

Witten, I., Frank, E., & Mark, A. (2011). *Data mining: Practical machine learning tools and techniques*. San Francisco: Morgan Kaufmann.

Xing, E., Jordan, M., & Karp, R. (2001). Feature selection for high-dimensional genomic microarray data. In C. E. Brodley & A. P. Danyluk (Eds.), *Proceedings of 18th International Conference on Machine Learning*, (pp. 601–608). San Francisco, CA: Morgan Kaufmann.

Yao, X., & Liu, Y. (1997). A new evolutionary system for evolving artificial neural networks. *IEEE Transactions on Neural Networks*, *8*(3), 694–713.

Yao, Y. (2003). Information-theoretic measures for knowledge discovery and data mining. In *Entropy measures, maximum entropy principle and emerging applications* (pp. 115–136). Heidelberg: Springer.

Yu, L., & Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, *5*, 1205–1224.