Feature Selection Algorithms in Classification Problems: An Experimental Evaluation

MICHAEL DOUMPOS, ATHINA SALAPPA
Department of Production Engineering and Management
Technical University of Crete
University Campus, 73100 Chania
Greece

Abstract: Feature selection (FS) is a major issue in developing efficient pattern recognition systems. FS refers to the selection of the most appropriate subset of features that describes (adequately) a given classification task. The objective of this paper is to perform a thorough analysis of the performance and efficiency of feature selection algorithms (FSAs). The analysis covers a variety of important issues with respect to the functionality of FSAs, such as: (a) their ability to identify relevant features, (b) the performance of the classification models developed on a reduced set of features, (c) the reduction in the number of features, and (d) the interactions between different FSAs with the techniques used to develop a classification model. The analysis considers a variety of FSAs and classification methods.

Key-Words: Feature Selection, Knowledge Discovery, Pattern Recognition, Machine Learning.

1 Introduction

A classification problem involves the assignment of some objects to a set of predefined classes. Each object i is assumed to be a multivariate vector $\mathbf{x}_i=(x_{i1}, x_{i2}, \ldots, x_{in})$, where x_{ij} is the description of object i on feature x_j. Essentially, the objective in a classification problem is to identify an unknown mapping function $f(\mathbf{x})$ that assigns each object to one of the predefined classes as accurately as possible. The development of f is based on a training sample consisting of m objects $(\mathbf{x}_1, c_1)$, $(\mathbf{x}_2, c_2)$, …, $(\mathbf{x}_m, c_m)$, where c_i denotes the class assignment for object i. Given such a training sample, the specification of f can be performed in many different ways using well-known methods.

The appropriate specification of the classification model f depends strongly on the quality of the training data. This is mainly related to the number of training objects and the adequacy of the features used in the analysis. FS involves the latter issue. The FS problem refers to the selection of the appropriate features that should be introduced in the analysis in order to maximize the performance of the resulting model. This has significant implications for issues such as [7]: (1) noise reduction through the elimination of noisy features, (2) reduction of the computational effort required to develop and implement an appropriate model, (3) simplification of the resulting models, and (4) facilitation of the easy use and updating of the models.

FS is usually performed as a preprocessing stage prior to model development, using special algorithms. The development of FSAs has been an active research topic in data mining and machine learning. FSAs are computational processes, which are used to select a set of features that optimizes an evaluation measure representing the quality of the features.

The research on this topic has been mainly focused on algorithmic developments, experimental evaluations and real world applications. However, most of the previous studies on the evaluation of FSAs' performance has focused on a limited number of algorithms and a limited number of methods.

This paper provides an extensive analysis of FSAs' performance in an experimental context using both real-world data sets as well as artificially generated data with pre-specified characteristics. The contribution of the paper compared to previous studies involves the analysis of a variety of FSAs combined with different popular classification methods including statistical and machine learning techniques. Such an analysis enables the investigation of the interactions between FSAs and the methods used for model development, as well as between the FSAs performance and the data set characteristics.

The rest of the paper is organized as follows: section 2 outlines the main characteristics and functionalities of FSAs. Section 3 describes the experimental setup, section 4 presents the obtained results, whereas section 5 concludes the paper and discusses possible future research directions.

2 Feature selection algorithms

An FSA is characterized by the strategy used to search for appropriate subsets of features, the feature selection process, the evaluation measure used to assess the quality of the features and the interaction with the classification method used to develop the final model [3], [10].

2.1 Search strategies

All FSAs employ a search strategy aiming at the specification of a feature weighting vector $\mathbf{w}=(w_1, w_2, \ldots, w_n)$ which has either a binary or real-valued form. In the former case, each element w_j of the weight vector is a 0/1 variable indicating whether feature x_j should be used in the analysis ($w_j=1$), or not ($w_j=0$). Alternatively, each w_j can be defined as a real-valued variable (usually in [0, 1]) representing the importance of feature x_j. In general, the search for the optimal weight vector can be performed through three main strategies [12]:

- Exponential strategies involving an exhaustive search of all feasible solutions.
- Sequential strategies based on a local search over solutions defined by the current solution state.
- Random strategies that employ randomness to avoid local optimal solutions.

2.2 Feature selection process

The feature selection process is usually implemented through forward, backward, compound, weighting and random schemes [9].

Forward techniques iteratively build the appropriate set of features beginning from an empty set. At each iteration, a feature that optimizes a pre-specified quality measure is selected and added to the current set of selected features. Backward processes begin with the full set of attributes and iteratively eliminate attributes so that the pre-specified quality measure is maximized. Compound techniques combine forward and backward selection, thus enabling both the addition and the elimination of features to/from the selected subset of features. Weighting approaches assign weights to features which act as proxies for the relevance of each feature, whereas random selection enables the construction of any possible feature subset from the current solution state.

2.3 Evaluation measures

The evaluation measure used in an FSA defines the way that the quality of each possible solution (subset of features) is assessed. The evaluation measure can be considered in relation to the classification method used for model development or it can be independently defined. In the former case, the minimization of the expected classification error rate is often used, but alternative measures are also applicable (e.g., the ROC curve). Alternatively, the quality of each subset of features can be assessed independently of the method used. Such approaches include measures such as the interclass distance [2], divergence measures [13] as well as consistency [1], and entropy-based measures [15].

2.4 Interaction schemes

A final important characteristic of FSAs involves the way they interact with the classification method used for model development. Three main interaction schemes have been explored:

- Filter schemes involving FSAs that do not interact with the classification method during feature selection. Within this context, FS is performed as a pure preprocessing stage prior to model development in order to filter out the irrelevant or redundant features from the analysis. Therefore, the selection of the most appropriate features is not related to the classification method used to build the model. This is the main disadvantage of such algorithms, since the peculiarities of the method are ignored during the FS process. However, filter techniques have been quite popular mainly due their computational efficiency even for large data sets. Some popular filter FSAs include the FOCUS algorithm [1], the RELIEF algorithm [6], Las Vegas algorithms [11], sequential forward and backward generation algorithms [4], [14], branch and bound techniques [5], etc.
- Wrapper schemes involving FSAs that use a classification method to assess the quality of the features [8]. Cross validation and bootstrapping techniques are used within this context in order to ensure a more reliable estimation of the features' quality. Such an approach explicitly considers the interactions between the selected features and the method used for model development, but this often requires increased computational effort.
- Embedded schemes involve procedures that implicitly perform FS as part of a classification method. Such approaches are quite popular is many machines learning methods such CART, rough sets, ID3, C4.5, etc. Similar approaches are also used in neural networks (weight decay) and support vector machines.

3 Experimental analysis

3.1 FSAs and methods

The main characteristic of the analysis in this paper involves the wide variety of FSAs and classification methods that are explored. Nine FSAs are considered in the analysis including both filter and wrapper methods:

- Las Vegas algorithms: (Las Vegas Filter-LFV, Las Vegas Incremental-LVI, Las Vegas Wrapper-LVW). The LVF algorithm employs a random search strategy to select a subset of features that optimize an evaluation measure, taking also into consideration the number of the selected features. The LVI algorithm is an extension of LVF that uses a portion of the training data, of gradually increased size depending on the quality of the selected features. The LVW algorithm is the wrapper version of LVF.
- FOCUS: The FOCUS algorithm is a filter algorithm which is based on an exponential search strategy to select a subset of features that optimize an evaluation measure.
- Sequential Forward and Backward Generation Filter/Wrapper (SFGF, SBGF, SFGW, SBGW): The SFGF and SBGF algorithms use a filter approach based on a sequential search strategy with a forward/backward feature selection process to build the appropriate subset of features. The SFGW and SBGW are the corresponding wrapper versions of the previous algorithms.
- RELIEF: The RELIEF algorithm follows the filter scheme based on a random search strategy to select features that maximize the separation of the classes measured through a distance measure.

The evaluation measure used for all filter FSAs is divergence which enables the consideration of both qualitative and quantitative features. For wrapper FSAs a 5-fold cross validation approach is used together with a classification method to assess the quality of the features.

The methods used in the analysis include linear discriminant analysis (LDA), logistic regression (LR), the CART algorithm, the nearest neighbor algorithm (NN), probabilistic neural networks (PNN) and support vector machines with an exponential kernel (SVM).

The consideration of such a variety of FSAs and classification methods enable the investigation of the performance of the FSAs for different methods which are extensively used to develop classification models. Furthermore, within this context the interactions between FSAs and the methods can also be explored.

3.2 Experimental setup

The experimental analysis is performed in two main directions. The first involves the analysis of 15 data sets taken from the UCI machine learning repository. The characteristics of the data sets used in this stage of the analysis are summarized in Table 1. All data sets are analyzed through a 10-fold cross validation experiment.

Table 1: UCI machine learning data sets

Data sets	Objects	Features
Bupa liver disorders	350	6
Hepatitis	160	19
Credit screening	690	14
Ionosphere	350	33
Mushroom	8120	6
Pima Indians Diabetes	770	8
Tic tac toe	960	9
Thyroid	2800	26
Breast cancer Wisconsin	570	30
Voting	440	16
German credit	1000	20
Heart disease	270	13
Monks-1	550	6
Monks-2	600	6
Monks-3	550	6

The second part of the analysis involves artificially generated two-class data, designed on the basis of four factors. Despite the restriction to two-class problems, the analysis has also implications to multi-class problems (multi-class problems are addressed by decomposing them two multiple two-class problems [6]).

The first factor defines the type of class separation (linear or quadratic). All generated data sets involve two-class problems, where the classes are defined on the basis of a linear or quadratic separating function. In both cases the coefficients for the relevant attributes in the separating function are modeled as uniformly distributed random variables in [1, 2] for the linear terms and [0, 0.25] for the quadratic terms. A normally distributed constant term with zero mean and unit variance is added to the separating function and an additional 10% noise is also imposed to the classification rule.

The second factor defines the number of features in the data sets. Three cases are considered in the analysis involving 10, 15 and 20 features.

The third factor defines whether irrelevant or redundant features are present in the data sets. In both cases relevant features are those with non-zero coefficients in the separating function. The presence of irrelevant features is considered with features that are not used in the separating function and have the same mean for both classes. On the other hand, redundant features are modeled as features which are not used in the separating function but are correlated to the relevant features.

Finally, the fourth factor defines the proportion of relevant features in the data as opposed to the total number of features. This proportion is varied between 40% and 80%, with a 20% increment.

Henceforth F1 will be used to denote the factor involving the presence of irrelevant or redundant features, F2 will denote the factor that defines the type of class separation, F3 will denote the factor defining the number of attributes and F4 will denoted the factor that specifies the proportion of features that are relevant.

For each combination of the factors (36 combinations overall) 10 training and validation sets are constructed from the standard normal distribution. Each data set consists of 500 objects which are equally distributed in two classes.

4 Results

4.1 UCI machine learning data sets

The first aspect of the analysis involves the accuracy of the models developed using the features selected from the FSAs compared to the models developed on the full set of features. Table 2 reports the average ratios (in %) of the accuracy of the reduced models to the accuracy of the full models. The number of data sets where the average 10-fold CV accuracy exceeded the corresponding accuracy of the full models is also reported in parentheses. The results show that CART and SVM are the two methods that most benefit from FS. In both cases, all FSAs led to the improvement of the classification accuracy. For the rest of the methods, the accuracies of the reduced models are slightly inferior to the complete models, but the differences are, in most cases, limited. It is also interesting to observe that, on average, wrapper FSAs outperform filter FSAs.

Table 2: Comparison of reduced to complete models in terms of classification accuracy

	CART	LDA	PNN	KNN	SVM	LR
LVF	98.7 (7)	99.6 (5)	91.5 (5)	92.8 (6)	105.6 (9)	98.8 (6)
LVI	98.8 (3)	100.2 (6)	94.4 (6)	94.3 (7)	104.9 (8)	99.7 (8)
FOCUS	100.1 (7)	99.6 (7)	96.9 (8)	97.1 (7)	101.5 (8)	99.3 (6)
SBGF	98.9 (5)	98.9 (6)	91.4 (7)	91.7 (6)	106.8 (7)	98.7 (5)
SFGF	98.7 (5)	99.5 (6)	92.5 (6)	92.5 (6)	107.1 (10)	99.1 (5)
RELIEF	93.8 (6)	95.0 (4)	90.6 (5)	86.5 (3)	101.8 (8)	93.2 (4)
LVW	102.0 (9)	99.6 (5)	95.8 (5)	97.0 (5)	107.4 (12)	99.7 (6)
SBGW	101.9 (10)	100.0 (5)	100.4 (5)	103.8 (9)	106.2 (10)	99.2 (5)
SFGW	101.5 (7)	99.7 (9)	102.1 (10)	102.9 (9)	111.6 (14)	99.4 (8)
Average	99.4	99.1	95.1	95.4	105.9	98.6

Two additional important issues in FS involve the reduction in the number of features, as well as the computational time required to select the appropriate subset of features. Table 3 presents the average proportion of the selected features compared to the complete number of features (PSF) and the average CPU times (in sec.) for the six filter FSAs. Similar results are reported in Table 4 for the wrapper FSAs, where the results differ according to the method with which each FSA interacts.

Table 3: Proportion of selected features and CPU times for the filter FSAs

	PSF	CPU time
LVF	49.16	1.89
LVI	61.10	7.79
FOCUS	78.48	0.36
SBGF	43.79	0.48
SFGF	44.46	0.17
RELIEF	41.76	2.34

Table 4: Proportion of selected features and CPU times for the wrapper FSAs

	PSF			CPU time		
	LVW	SBGW	SFGF	LVW	SBGW	SFGW
CART	36.09	55.17	39.22	15.9	15.7	30.8
LDA	50.87	62.08	42.23	1.8	3.7	2.8
PNN	44.39	65.57	53.99	19.5	69.7	61.2
KNN	47.42	69.91	52.20	11.2	26.6	22.3
SVM	22.29	73.53	34.07	5.4	15.7	9.1
LR	44.53	69.22	43.47	19.2	109.1	46.4
Average	40.93	65.91	44.20	12.2	40.1	28.8

According to the above results the algorithms LVI, FOCUS and SBGW seem to be the least successful FSAs in reducing the number of features used for model development. For all the other algorithms, the average number of selected features is lower that 50% of the complete set of features. In terms of computational efficiency the two sequential algorithms (SBGF, SFGF) as well as FOCUS clearly outperform the rest of the algorithms. Of course, as expected wrapper FSAs, are computationally more intensive compared to filter approaches, mainly in cases of data sets with a large number of features. However, as noted earlier, this is compensated by their increased performance in terms of the accuracy of the resulting models. Finally, it is interesting to notice that the most important reduction in the number of the selected features is observed for CART and SVM using two wrapper methods (LVW, SFGW). For the same combination of FSAs-methods the highest improvements in accuracy were also observed (cf. Table 2).

4.2 Artificial data sets

The analysis of the artificial data sets enables the investigation of the performance of the FSAs with respect the characteristics of the data.

In terms of classification accuracy, the most important factors involve the presence of irrelevant or redundant features (F1) and the proportion of factors that are relevant (F4). Table 5 summarizes the corresponding results for all FSAs. The presented results involve the ratios (in %) of the accuracy of the reduced models to the accuracy of the full models, averaged over all methods. The percentage of experiments where the reduced models outperformed the complete models is also reported (in parentheses) for each FSA. The results show that the success of the reduced models decreases when redundant features are present in the data as opposed to the case of irrelevant features. On average, in almost 54% of the cases with irrelevant features the reduced models outperformed the full models. On the other hand, when redundant features are present, the reduced models outperformed the full models in almost 31% of the cases. This is a first indication that FSAs are more successful in identifying irrelevant features than redundant ones. Furthermore, FSAs are more successful in cases where the relevant features are a small portion of the complete set of features. In almost 65% of the cases where only 40% of the features are relevant, the reduced models outperformed the full models; the same was observed in only 26% of the cases where 80% of the features were relevant. Overall, the wrapper FSAs seem to provide more robust results compared to filter FSAs, but they are not always superior compared to filter techniques.

Table 5: Comparison of reduced to complete models in terms of classification accuracy (artificial data)

	F1		F4		
	Irrelevant	Redundant	40%	60%	80%
LVF	101.96	99.07	101.74	100.16	99.31
	(66.7)	(16.7)	(50.0)	(33.3)	(16.7)
LVI	101.75	99.30	101.73	100.19	99.37
	(66.7)	(16.7)	(66.7)	(33.3)	(16.7)
FOCUS	101.05	100.03	100.74	100.52	100.23
	(66.7)	(33.3)	(50.0)	(50.0)	(33.3)
SBGF	103.46	98.73	102.92	100.68	99.15
	(66.7)	(16.7)	(66.7)	(33.3)	(16.7)
SFGF	103.69	99.74	103.25	101.33	100.10
	(66.7)	(16.7)	(83.3)	(66.7)	(33.3)
RELIEF	94.48	98.12	97.55	95.61	95.45
	(16.7)	(0.0)	(33.3)	(0.0)	(0.0)
LVW	98.83	99.54	100.04	99.10	98.42
	(0.0)	(50.0)	(66.7)	(33.3)	(33.3)
SBGW	101.22	100.66	101.60	100.85	100.30
	(83.3)	(66.7)	(83.3)	(100.0)	(50.0)
SFGW	100.89	100.70	102.22	100.54	99.51
	(50.0)	(66.7)	(83.3)	(66.7)	(33.3)
Average	100.81	99.54	101.31	99.89	99.09
	(53.7)	(31.5)	(64.8)	(46.3)	(25.9)

Except for the performance of the FSAs in terms of the classification accuracy, their ability to identify relevant features is also considered using the artificial data sets. This is analyzed in terms of two measures. The first involves the number of relevant features that are selected, expressed as percentage of the total number of relevant features. Similarly, the second measure involves the number of irrelevant/redundant features that are selected, expressed as percentage of the total number of irrelevant/redundant features. The difference between the two measures defines a hit rate in [-100%, 100%] for each FSA. A hit rate equal to 100% corresponds to the case where an FSA selects all the relevant features but none of the irrelevant/ redundant features. A hit rate equal to -100% corresponds to the case where an FSA does not select any relevant features but it does select all the irrelevant/redundant ones.

Table 6 presents the FSAs' hit rates (in %) for all the four factors used in the analysis. The results show all factor have a significant impact on results. The most significant differences are observed for the factors F1 and F4. In particular, the FSAs are more successful in discriminating between relevant and irrelevant features, than between relevant and redundant features. When the task becomes more complex (non-linear separation, large number of features, small portion of relevant features), the hit rates decrease. Also, the hit rates for the filter methods are generally higher to the wrapper techniques.

Table 6: Hit rates for the selection of relevant features by the FSAs

	F1		F2		F3			F4		
	1	2	1	2	1	2	3	1	2	3
LVF	40.9	20.1	31.7	29.3	38.9	28.6	24.0	15.2	29.8	46.5
LVI	38.5	20.1	31.8	26.7	35.8	28.6	23.5	8.8	29.5	49.6
FOCUS	31.1	24.1	30.0	25.2	30.9	27.0	25.0	-5.9	28.5	60.3
SBGF	51.0	23.0	40.9	33.1	42.4	35.2	33.5	26.8	36.8	47.5
SFGF	51.0	22.7	40.2	33.5	42.7	34.7	33.2	24.3	37.3	49.0
RELIEF	18.6	16.0	18.2	16.4	18.9	17.4	15.5	-0.5	17.5	35.0
LVW	25.5	14.9	22.7	17.8	26.6	18.8	15.4	4.8	19.9	36.0
SBGW	33.3	24.3	29.7	27.9	33.4	28.4	24.7	4.1	28.7	53.7
SFGW	35.8	19.2	30.6	24.5	35.4	26.5	20.7	15.0	27.3	40.2
Average	36.2	20.5	30.6	26.1	33.9	27.2	23.9	10.3	28.4	46.4

Table 7 presents some additional results on the hit rates for the wrapper FSAs in connection with the classification methods. The results show that LVW provides consistently worst results compared to the two sequential algorithms. In most cases SBGW is the most successful algorithm followed by SFGW. The interaction of the algorithms with SVM provide the most successful results, whereas the combination with CART performs poorly.

Table 7: Hit rates for the selection of relevant features by wrapper FSAs

	CART	LDA	PNN	KNN	SVM	LR
LVW	13.30	22.33	11.90	23.55	27.43	22.95
SBGW	23.19	30.59	29.61	28.56	29.78	31.19
SFGW	15.88	29.27	24.22	27.36	38.38	30.00

5 Conclusions

FS is a major research topic for the development of classification methods. This study presented experimental results on the performance of several FSAs for different popular classification methods, using both real world data and artificial data.

The results show that most FSAs lead to significant reductions in the dimensionality of the data, without sacrificing the performance of the resulting models. Generally, wrapper techniques were found to be more successful in terms of classification accuracy, but this comes with increased computational cost. SVM was found as the method that benefited the most from FS. In terms of the efficiency of the algorithms in identifying features that are actually relevant, the analysis showed that, generally, filter techniques provide better results. However, it should be emphasized that throughout the experimental analysis significant differences were observed in the features selected by the FSAs, thus indicating that relying on a single algorithm may not be a good choice when making inferences on which features are crucial for a classification task.

Further analysis could involve the consideration of additional FS methodologies such as evolutionary techniques (e.g., genetic algorithms) as well as branch and bound methods and linear programming techniques. It would also be interesting to explore the combination of different FSAs in order to obtain improved results both in terms of classification accuracy, as well as in terms of the identification of the relevant attributes for the analysis.

References:

[1] Almuallim, H. and Dietterich, T.G. (1994), “Learning boolean concepts in the presence of many irrelevant features”, *Artificial Intelligence*, 69(1-2), 279-305.

[2] Ben-Bassat, M. (1982), “Use of distance measures, information measures and error bounds in feature evaluation”, in: Krishnaiah, P.R. and Kanal, L.N. (eds.), *Handbook of Statistics,* North Holland, 773-791.

[3] Blum, A.L. and Langley, P. (1997), “Selection of relevant features and examples in machine learning”, *Artificial Intelligence*, 97(1-2), 245-271.

[4] Choubey, S. K., Deogun, J. S., Raghavan, V. V. and Sever, H. (1996), “A comparison of feature selection algorithms in the context of rough classifiers”, in: *Proceedings of the 5th IEEE International Conference on Fuzzy Systems (vol. 2)*, New Orleans, LA, 1122-1128.

[5] Dash, M. and Liu, H. (1998), “Hybrid search of feature subsets”, in: Lee, H.Y. and Motoda, H. (eds.), *Proceedings of the 15th Pacific Rim International Conference on Artificial Intelligence*, Springer Verlag, Singapore, 22-27.

[6] Dietterich, T.G. and Bakiri G. (1995), “Solving multiclass learning problems via error-correcting output codes”, *Journal of Artificial Intelligence Research*, 2, 263-286.

[7] Kira, K. and Rendell, L. (1992), “The feature selection problem: Traditional methods and a new algorithm”, in: *Proceedings of AAAI-92*, AAAI Press, 129-134.

[8] Kohavi, R. and John, G.H. (1997), “Wrappers for feature subset selection”, *Artificial Intelligence*, 97(1-2), 273–324.

[9] Koller, D. and Sahami, M. (1996), “Toward optimal feature selection”, in: Saitta, L. (ed.), *Proceedings of the 13th International Conference on Machine Learning,* Morgan Kaufmann, San Francisco, 284-292.

[10] Liu, H. and Motoda, H. (1998), *Feature Selection for Knowledge Discovery and Data Mining*, Kluwer Academic Publishers, London, GB.

[11] Liu, H. and Setiono, R. (1998), “Scalable feature selection for large sized databases”, in: *Proceedings of the 4th World Congress on Expert Systems*, Morgan Kaufmann, 68-75.

[12] Molina, L.C., Belanche, L. and Nebot, A. (2002), “Feature selection algorithms: A survey and experimental evaluation”, in: *Proceedings of the 2002 IEEE International Conference on Data Proceedings of the 2002 IEEE International Conference on Data Mining*, IEEE Computer Society, 306-313.

[13] Narendra, P. and Fukunaga, K. (1977), “A branch and bound algorithm for feature subset selection”, *IEEE Transactions on Computers,* 26(9), 917-922.

[14] Pudil, P., Novovicova, J. and Kittler, J. (1994), “Floating search methods in feature selection”, *Pattern Recognition Letters*, 15(11), 1119-1125.

[15] Quinlan, J.R. (1993) *C4.5: Programs for Machine Learning,* Morgan Kaufmann.