Proceedings of the 25th IASTED International Multi-Conference
ARTIFICIAL INTELLIGENCE AND APPLICATIONS
February 12–14, 2007, Innsbruck, Austria
ISBN Hardcopy: 978-0-88986-629-4 / CD: 978-0-88986-631-7

A STABILITY INDEX FOR FEATURE SELECTION

Ludmila I. Kuncheva
School of Informatics
University of Wales, Bangor, LL57 1UT, UK
e-mail: l.i.kuncheva@bangor.ac.uk

ABSTRACT

Sequential forward selection (SFS) is one of the most widely used feature selection procedures. It starts with an empty set and adds one feature at each step. The estimate of the quality of the candidate subsets usually depends on the training/testing split of the data. Therefore different sequences of features may be returned from repeated runs of SFS. A substantial discrepancy between such sequences will signal a problem with the selection. A stability index is proposed here based on cardinality of the intersection and a correction for chance. The experimental results with 10 real data sets indicate that the index can be useful for selecting the final feature subset. If stability is high, then we should return a subset of features based on their total rank across the SFS runs. If stability is low, then it is better to return the feature subset which gave the minimum error across all SFS runs.

KEY WORDS

Pattern recognition, Feature selection, Stability index, Sequential Forward Selection (SFS)

1 Introduction

Feature selection has been a focus of pattern recognition for a long time now [13, 4, 9, 2, 3, 7, 12, 10, 11]. It has been well documented that a subset of features may work better than the entire set [6]. This phenomenon, termed "peak effect", is explained by the fact that constructing the true minimum-error classifier from finite training data is impossible, and the (imperfect) approximation is adversely affected by irrelevant features. Hence selecting a suitable subset of features is not only computationally desirable but can also lead to better classification accuracy.

In this paper we are interested in the *wrapper* approach to feature selection. The quality of a feature subset is measured by an estimate of the classification accuracy of a chosen classifier trained on the candidate subset. The problem addressed here comes from the fact that the estimate of the classification accuracy is a random variable. Therefore, when two feature subsets are compared, the decision as to which one should be preferred involves uncertainty. This is particularly important in the sequential feature selection methods which augment or reduce the selected subset at each step. A flip in the decision at an earlier step may lead to a completely different selection path, and result in a very different subset of features being selected. Stability of the selected features is an important aspect when the task is knowledge discovery, not merely returning an accurate classifier. The domain experts will try to reason why the returned subset of features contains the most relevant discriminatory information. If there have been numerous different subsets of approximately equal quality, presenting the user with only one subset may be misleading.

It is curious that little attention has been devoted to the stability of the estimate of the criterion value, an issue independently raised by Dunne et al. [5] and Kalousis et al. [8]. Sima et al. [14] argue that differences in performances among feature selection algorithms are less significant than the performance differences among the error estimators used to implement the algorithms.

Here we propose to look at the agreement between the sequences as a measure of stability of the selected feature subsets. The rest of the paper is organised as follows. Section 2 introduces the stability index. In Section 3, we discuss the problem of how to choose a feature subset in case of multiple selection sequences. Experimental results with 10 data sets are given in Section 4, and Section 5 concludes the paper.

2 A Stability Index

2.1 The stability problem in Sequential Forward Selection (SFS)

Consider the *sequential forward selection* (SFS) procedure [15]. Let $X = \{x_1, \dots, x_n\}$ be the original set of features and $J(S)$ be a measure of quality of a subset $S \subseteq X$. Starting with an empty subset, S, one feature is added at each step. To choose this feature, all possible subsets of $S \cup \{x_i\}$ are evaluated, where x_i is a feature from X which is not in S. The best feature to add is taken to be $x^* = \arg\max_{x_i \in X \setminus S} J(S \cup \{x_i\})$.

The problem is that we do not have the exact value of $J(S \cup \{x_i\})$ but only an approximation thereof evaluated on a part of the training data. Thus the choice of x^* depends on the accuracy of this estimate. If a large training set is available or if one can afford a large number of data shuffling runs, so that the variance of $J(S \cup \{x_i\})$ is small, the estimate will be reliable enough and the choice of x^* will be unequivocal. However, when this is not possible, we have to account for the variability of $J(S \cup \{x_i\})$ in other ways. If SFS is run to the end, the result is a sequence of features entering the subset. If a subset of d features is required, the first d features of the sequence will be returned. Suppose that we carry out K runs of SFS and record the sequences $S_1, S_2, \ldots, S_K$. The question is how similar these sequences are and whether this similarity can help us choose the final subset to be returned to the user.

2.2 Consistency between a pair of subsets

Let A and B be subsets of features, $A, B \subset X$, of the same size (cardinality), k. Let $r = |A \cap B|$ be the cardinality of the intersection of the two subsets. A list of desirable properties of a consistency index for a pair of subsets is given below

1. **Monotonicity.** For a fixed subset size, k, and number of features, n, the larger the intersection between the subsets, the higher the value of the consistency index.
2. **Limits.** The index should be bound by constants which do not depend on n or k. The maximum value should be attained when the two subsets are identical, i.e., for $r = k$.
3. **Correction for chance.** The index should have a constant value for independently drawn subsets of features of the same cardinality, k.

A general form of such index is

$$\frac{\text{Observed } r - \text{Expected } r}{\text{Maximum } r - \text{Expected } r}. \tag{1}$$

Maximum r equals k, achieved when A and B are identical subsets. To evaluate the expected cardinality of the intersection, consider r to be a random variable obtained from randomly drawn A and B of size k from a set X of size n (without replacement). We can think of subset A as fixed. Suppose that the elements of $X \setminus A$ are colored in white and those in A are colored in black. A set B of size k is selected without replacement from X. The number of objects from A (black) selected also in B is a random variable Y with hypergeometric distribution with probability mass function

$$P(Y = r) = \frac{\binom{k}{r}\binom{n-k}{k-r}}{\binom{n}{k}}. \tag{2}$$

The expected value of Y for given k and n is $\frac{k^2}{n}$.

Definition 1. *The Consistency Index for two subsets $A \subset X$ and $B \subset X$, such that $|A| = |B| = k$, where $0 < k < |X| = n$, is*

$$I_C(A, B) = \frac{r - \frac{k^2}{n}}{k - \frac{k^2}{n}} = \frac{rn - k^2}{k(n-k)}. \tag{3}$$

This index satisfies the three properties above. First, for fixed k and n, $I_C(A, B)$ increases with increasing r. Second, the maximum value of the index, $I_C(A, B) = 1$, is achieved when $r = k$. The minimum value of the index is bound from below by -1. The limit value is attained for $k = \frac{n}{2}$ and $r = 0$. Note that $I_C(A, B)$ is not defined for $k = 0$ and $k = n$. These are the trivial cases where either no feature is selected or all features are selected. They are not interesting from the point of view of comparing feature subsets, so the lack of values for $I_C(A, B)$ in these cases is not important. For completeness we can assume $I_C(A, B) = 0$ for both cases. Finally, $I_C(A, B)$ will assume values close to zero for independently drawn A and B because r is expected to be around $\frac{k^2}{n}$.

2.3 An example

Consider the following two hypothetical sequences of features obtained from two runs of SFS on a data set with 10 features.

$$\begin{aligned} S_1 &= \{x_9, x_7, x_2, x_1, x_3, x_{10}, x_8, x_4, x_5, x_6\} \\ S_2 &= \{x_3, x_7, x_9, x_{10}, x_2, x_4, x_8, x_6, x_1, x_5\} \end{aligned}$$

Denote by $S_i(k)$ the subset of the first k features of sequence S_i. The cardinality of the intersection of $S_1(3)$ and $S_2(3)$ is $|\{x_7, x_9\}| = 2$. Then

$$I_C(S_1(3), S_2(3)) = \frac{2 \times 10 - 3^2}{3(10-3)} = \frac{11}{21} \approx 0.5238.$$

Figure 1 shows $I_C(S_1(k), S_2(k))$ against the set size k. By introducing the correction for chance the consistency index I_C differs from two indices proposed previously. Kalousis et al. (2005) introduce the similarity index between two subsets of features, A and B, as

$$S_S(A, B) = 1 - \frac{|A| + |B| - 2|A \cap B|}{|A| + |B| - |A \cap B|} = \frac{|A \cap B|}{|A \cup B|}, \tag{4}$$

where $|\cdot|$ denotes cardinality, '$\cap$' denotes intersection and '$\cup$' denotes union of sets. Dunne et al. (2002) suggest to measure the stability using the relative Hamming distance between the masks corresponding to the two subsets, which in set notation is

$$S_H(A, B) = 1 - \frac{|A \setminus B| + |B \setminus A|}{n}, \tag{5}$$

where '$\setminus$' is the set-minus operation and n is the total number of features. The two indices were calculated for $S_1(k), S_2(k)$ from the example above, where k was varied from 1 to n. The results are plotted also in Figure 1. While all three indices detect the dip at $k = 4$ features, S_S and S_H have a tendency to increase when the size of the selected set approaches the total number of features n. The point of view advocated here is that consistency should have high value only if it exceeds the consistency by chance or by design.

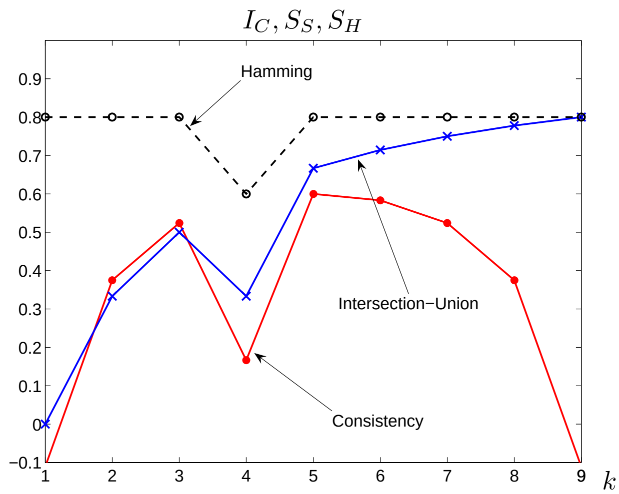

Figure 1. Consistency index I_C and similarities S_S, S_H for sequences S_1 and S_2 (the example in the text) plotted against the subset size k

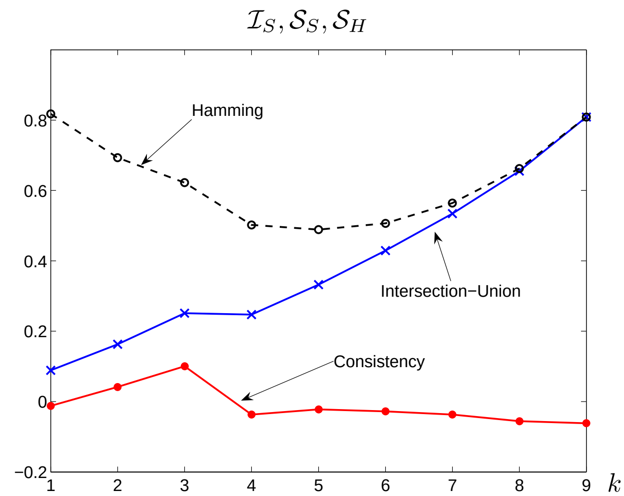

Figure 2. Consistency index $\mathcal{I}_S$ and similarities $\mathcal{S}_S, \mathcal{S}_H$ for 10 random sequences plotted against the subset size k

2.4 A stability index for K sequences

Let $S_1, S_2, \ldots, S_K$ be the sequences of features obtained from K runs of SFS on a given dataset.

Definition 2. *The Stability Index for a set of sequences of features, $\mathcal{A} = \{S_1, S_2, \ldots S_K\}$, for a given set size, k, is the average of all pairwise consistency indices*

$$\mathcal{I}_S(\mathcal{A}(k)) = \frac{2}{K(K-1)} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} I_C(S_i(k), S_j(k)). \tag{6}$$

Averaging the pairwise similarities to arrive at a single index is also the approach adopted for both S_S and S_H [5, 8]. Denote the averaged indices by $\mathcal{S}_S$ and $\mathcal{S}_H$, respectively. To strengthen the argument for correction for chance, Figure 2 shows $\mathcal{I}_S$, $\mathcal{S}_S$ and $\mathcal{S}_H$ across all pairs of 10 independently generated random sequences. Only $\mathcal{I}_S$ gives consistency around zero for any number of features k. Similarity $\mathcal{S}_S$ favours large subsets and $\mathcal{S}_H$ favours large and small but not medium-size subsets. For these reasons only $\mathcal{I}_S$ is considered in the rest of the paper.

To illustrate the operation of $\mathcal{I}_S$ on real data, we chose the Spam dataset from the UCI repository [1] (2 classes, 57 features, 4601 objects). The problem is to distinguish between spam and legitimate e-mail. The first 54 features are the frequencies of each of 48 words and each of 6 characters. The remaining 3 features are the average length of uninterrupted sequences of capital letters, the length of the longest uninterrupted sequence of capital letters and total number of capital letters in the e-mail. All features are continuous-valued. This set was chosen because the importance of the features can only be guessed at the time the data collection was started. This means that many features may turn out to be redundant, so feature selection is paramount. Figure 3 shows $\mathcal{I}_S$ against k for two experiments, each producing $K = 10$ sequences In the first experiment, for each subset candidate S, the criterion $J(S)$ was obtained as the average of $T = 10$ evaluations while in the second experiment we used $T = 100$.

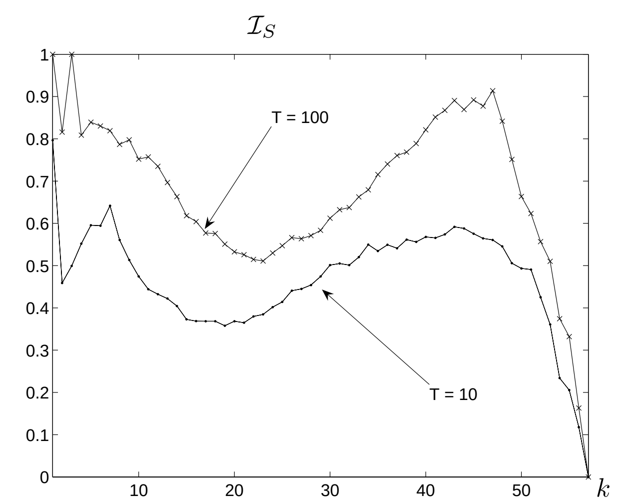

Figure 3. Consistency index $\mathcal{I}_S$ for the Spam dataset with 10 and 100 evaluations for each $J(S)$

As expected, when the variance of the estimate of $J(S)$ is lower ($T = 100$), the stability $\mathcal{I}_S$ is higher. There is a dip in the consistency after the first selected feature (x_{53} = the frequency of character '$' in the message), on which all the selection runs agree for $T = 100$. It seems that the first feature is important on its own and is not involved in a distinctly informative pair with another feature. The sequences are also fairly consistent toward the end, which singles out a set of redundant features, e.g., the frequency of words such as "you", "your", "e-mail" and "receive".

3 Feature selection from multiple sequences

The four basic issues in feature selection, formulated by Langley (1994), are

(1) Starting point. In this study the standard sequential forward selection (SFS) is applied, therefore the starting point for each S_i is the empty set.

(2) Construction of candidate subsets. To derive $S_i(k+1)$, all subsets of size $k+1$ containing $S_i(k)$ and another feature are formed and checked. The best subset is retained.

(3) Criterion for evaluating the candidate subsets. In this study we adopt a wrapper approach whereby the quality of a feature subset, $S_i(k)$, is gauged by an estimate of the classification error, $J(S_i(k))$, of a classifier built using only the k features in $S_i(k)$. Any classifier model can be employed. For the purposes of illustrating the feature selection process from a collection of sequences $S_i(k)$, $i = 1, \ldots, K$, we picked the simple *nearest mean classifier*. Each object is labelled to the class with the nearest mean (Euclidean distance is used throughout the experiment). The means are estimated from the training data.

(4) Stopping criterion. The peak effect which normally occurs through the selection process suggests that there is an optimal number of features for the chosen classifier. The minimum of the error is found and the corresponding feature subset is returned. Searching for a minimum can be done on-line so that the selection is stopped as soon the next feature does not improve on $J(S_i(k))$. However, knowing that the error is non-monotonic on k in principle, as well as knowing that the obtained $J(S_i(k))$ is only an estimate of the true value, stopping at the first minimum found may be far off the optimal solution. The alternative is to run the selection until all features enter the optimal subset, i.e., $S_i(n) = X$, for all $i = 1, \ldots, K$. The optimal number of features is decided afterwards.

Treating J as a random variable and taking the result of multiple runs raise new questions and require a complete re-thinking of (3) and (4) in the above list. For example, given the set of sequences $\mathcal{A} = \{S_1, S_2, \ldots, S_K\}$, what final subset of features should we return to the user? Should $\mathcal{I}_S$ be taken into account in deciding whether a final subset should be returned to the user at all? If so, how? Knowing that $\mathcal{I}_S$ does not depend on J, can the two be used in conjunction to determine a stopping point for the search?

3.1 Choosing a final sequence of features

In order to cut a subset of features to return to the user, a final sequence S^* has to be chosen. Given $\mathcal{A}$, there are various intuitive options for choosing S^* and the number of features. The following two options comply with the current practices

• Rank the features in each sequence S_i so that the best feature (the one starting the sequence) is assigned rank 1, the second is assigned rank 2, etc. Sum up the K ranks for each feature. Order the features in ascending order by the total ranks to get a final sequence S^*_{rank}.

• Find the sequence with the minimum local error.

$$S^*_{\min} = \arg\min_i \left\{ \min_k J(S_i(k)) \right\}.$$

Having a set of sequences instead of a single one opens up a multitude of choices with respect to selecting the number of features d. Some of the possible way to pick d are

(a.) Using the final sequence, pick the number d such that $d = \arg\min_k J(S^*(k))$.

(b.) Using the final sequence, fit a polynomial to $J(S^*(k))$ as a function of k. Find the minimum analytically using the coefficients. The integer value of k closest to the minimum is retrieved as d. The benefit of this calculation is that the criterion curve will be smoothed. Fluctuations of $J(S^*(k))$ are expected to occur due to estimation errors. Such fluctuations will represent noise in the selection process and so should be eliminated.

(c.) Apply (a) on the mean error across the K sequences.

(d.) Apply (b) on the mean error across the K sequences.

(e.) Find the suggested number of features for each S_i by either (a) or (b). Let d_i be this number for sequence S_i, $i = 1, \ldots, K$. Compare all d_i and derive a final d based on median, mode or mean.

(f.) Find a set of “consistent values” d, for which the sequences agree, i.e.,

$$D_{\mathrm{cons}} = \{ d \,|1 \le d \le n, \mathcal{I}_S(\mathcal{A}(d)) > \theta \},$$

where θ is a predefined threshold on $\mathcal{I}_S$. Choose d to be the one with the smallest $J(S^*(d))$ within D_{cons}.

All the experiments in the next section were carried out with option (a). If the final sequence is $S^*_{\min}$, then picking d is straightforward because the errors for $k = 1, 2, \ldots, n$ are available from the training run producing $S^*_{\min}$. If S^*_{rank} is chosen, another evaluation run has to be carried out in order to find the validation error of each subset $S^*_{\mathrm{rank}}(k)$.

4 Experimental results

The experiment in this paper is rather exploratory. We seek to find a relationship between the stability of the selected sets and the choice between S^*_{rank} and $S^*_{\min}$. Intuitively, if stability is high, S^*_{rank} would be better because it will smooth out the small discrepancies between the selected subsets. If stability is low, perhaps there have been runs which have discovered by chance irregular troughs of the error criterion J, accounting for a set of dependent and useful features. In that case it may be better to use $S^*_{\min}$ and return the best feature subset across all individual runs.

4.1 Data

Table 1 shows a summary of the ten data sets chosen for the feature selection experiment:- 8 datasets from UCI [1] and 3 additional medical datasets: Scrapie, Laryngeal and Contractions.

4.2 Experimental protocol

The following steps were carried out with each data set

1. The data set was divided into a training part (2/3), and a testing part (1/3).

2. SFS was run 10 times on the training part only, with the nearest mean classifier (NMC) within the wrapper approach. Thus $K = 10$ sequences of features were obtained.

3. To evaluate J, for each candidate subset, an NMC was trained on 90% of the training data, called "pre-training set", and tested on the remaining 10% of the *training data*, called a "validation set". To examine the effect of stability of the estimate, this procedure was repeated T times for each candidate subset with a different random split of the training data into 90% pre-training and 10% validation parts. The resultant value $J(S)$ for a candidate subset S was taken to be the average validation error across the T runs. Two values of T were tried: $T = 10$ and $T = 100$. Thus two sets of sequences were obtained, denoted $\mathcal{A}_{10}$ and $\mathcal{A}_{100}$, respectively.

4. A random set of 10 sequences, $\mathcal{A}_{\mathrm{rand}}$, was generated for comparison. Each of the 10 sequences was a random permutation of the integers from 1 to n drawn independently of the other permutations.

5. The stability index $\mathcal{I}_S(\mathcal{A}(k))$ was calculated for $\mathcal{A}_{10}$ and $\mathcal{A}_{100}$.

6. Final sequences S^*_{rank} and S^*_{min} were obtained from for each of $\mathcal{A}_{10}$ and $\mathcal{A}_{100}$, as explained above.

7. The suggested number of features, d was obtained through method (a). The classification error for each final sequence and each k was evaluated on the *testing set*. First an NMC was trained on the whole training set using the features in $S^*(k)$ and then tested on the *testing set*.

Table 2 shows the results from the experiments. There is no clear favourite between S^*_{rank} and S^*_{min}. However, an interesting tendency can be observed. For $\mathcal{I}_S \geq 0.5$, S^*_{rank} provides, in general, better solutions than S^*_{min}. This confirms the intuition that when high stability is detected, smoothing the discrepancies is beneficial. Conversely, if stability is low $\mathcal{I}_S < 0.5$, the best individual run should be chosen. The exceptions from this observations are Sonar with $T = 10$ and $T = 100$, Spam with $T = 10$ and Vehicle with $T = 100$. As there are 8 out of the 10 cases supporting the tendency for each T, based on this experiment, we can suggest that the tendency is not due to chance, with level of significance $p < 0.10$. This observation is valid for the chosen parameters of the experiment: SFS, nearest mean classifier, $K = 10$ sequences, and $T = 10$ or $T = 100$. Given that the experiment was of exploratory nature, it would be too premature to speculate and generalise.

5 Conclusion

A consistency index is proposed here to help feature selection when multiple selection sequences are available. It can be used to find out whether a final sequence based on the feature ranks will be a good solution. If stability is high ($\mathcal{I}_S \geq 0.5$), the rank order of the features, S^*_{rank}, appeared to be better than choosing the subset with the minimum error found during training. On the other hand, if stability is low, the best individual runs can be more useful. Alternatively, in the case of low consistency, we may abstain from selecting any features.

References

[1] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

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

[3] M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1:131–156, 1997.

[4] P. A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach*. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1982.

[5] K. Dunne, P. Cunningham, and F. Azuaje. Solution to instability problems with sequential wrapper-based approaches to feature selection. Technical Report TCD-CS-2002-28, Department of Computer Science, Trinity College, Dublin, Ireland, 2002.

[6] A. K. Jain and B. Chandrasekaran. Dimensionality and sample size considerations in pattern recognition practice. In P. R. Krishnaiah and L. N. Kanal, editors, *Handbook of Statistics*, pages 835–855. North Holland, 1982.

[7] A. K. Jain and D. Zongker. Feature selection: evaluation, application and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):153–158, 1997.

[8] A. Kalousis, J. Prados, and M. Hilario. Stability of feature selection algorithms. In *Proc. 5th IEEE International Conference on Data Mining (ICDM'05)*, pages 218–225, 2005.

[9] P. Langley. Selection of relevant features in machine learning. In *Proc AAAI Fall Symposium on Relevance*, pages 140–144, 1994.

[10] H. Liu, R. Stine, and L. Auslender, editors. *Proceedings of the Workshop on Feature Selection for Data Mining*. Newport Beach, CA, 2005.

[11] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17(4):491–502, 2005.

[12] L. C. Molina, L. Belanche, and A. Nebot. Feature selection algorithms: A survey and experimental evaluation. In *Proc. the IEEE International Conference on Data Mining (ICDM'02)*, Japan, 2002.

[13] E. A. Patrick. *Fundamentals of Pattern Recognition*. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1972.

[14] C. Sima, S. Attor, U. Brag-Neto, J. Lowey, E. Suh, and E. R. Dougherty. Impact of error estimation on feature selection. *Pattern Recognition*, 38:2472–2482, 2005.

[15] S. Stearns. On selecting features for pattern classifiers. In *3-d International Conference on Pattern Recognition*, pages 71–75, Coronado, CA, 1976.

Table 1. Data description and error rates for $d = 0$ selected features and $d = n$ (all) selected features

Dataset	Features n	Classes c	Objects N	Error with no features ($d = 0$)	Error with all features ($d = n$)	
Contractions	2	27	98	0.500	0.173	$\pm$0.026
Image	7	19	210	0.857	0.297	$\pm$0.020
Ionosphere	2	34	351	0.359	0.276	$\pm$0.015
Laryngeal1	2	16	213	0.620	0.367	$\pm$0.020
Scrapie	2	14	3113	0.829	0.312	$\pm$0.005
Sonar	2	60	208	0.534	0.325	$\pm$0.021
Spam	2	57	4601	0.394	0.321	$\pm$0.005
Spect	2	44	349	0.728	0.324	$\pm$0.014
Vehicle	4	18	846	0.749	0.612	$\pm$0.009
Votes	2	16	232	0.534	0.095	$\pm$0.012

Table 2. Number of selected features d, error rates E and stability $\mathcal{I}_S$ obtained from a random sequence of features, R, the ranked sequence $S^*_{\rm rank}$ and the minimum-selection sequence $S^*_{\rm min}$, for $T = 10$ and $T = 100$ evaluation runs for J

		$T = 10$				$T = 100$			
Dataset	Method	d	E	95%CI	$\mathcal{I}_S$	d	E	95%CI	$\mathcal{I}_S$
Contractions	R	9	0.257	$\pm$0.046		6	0.187	$\pm$0.041	
	$S^*_{\rm rank}$	8	0.180	$\pm$0.050	0.001	14	0.163	$\pm$0.042	0.219
	$S^*_{\rm min}$	3	0.060	$\pm$0.025		7	0.090	$\pm$0.033	
Image	R	2	0.503	$\pm$0.039		9	0.464	$\pm$0.035	
	$S^*_{\rm rank}$	11	0.294	$\pm$0.035	0.242	8	0.199	$\pm$0.027	0.549
	$S^*_{\rm min}$	8	0.167	$\pm$0.027		11	0.281	$\pm$0.030	
Ionoshpere	R	6	0.135	$\pm$0.017		12	0.190	$\pm$0.022	
	$S^*_{\rm rank}$	3	0.128	$\pm$0.017	0.569	4	0.145	$\pm$0.021	0.547
	$S^*_{\rm min}$	16	0.183	$\pm$0.019		13	0.191	$\pm$0.024	
Laryngeal	R	11	0.157	$\pm$0.024		6	0.200	$\pm$0.027	
	$S^*_{\rm rank}$	3	0.174	$\pm$0.024	0.152	9	0.184	$\pm$0.027	0.317
	$S^*_{\rm min}$	7	0.161	$\pm$0.024		3	0.163	$\pm$0.028	
Scrapie	R	1	0.187	$\pm$0.008		11	0.276	$\pm$0.008	
	$S^*_{\rm rank}$	1	0.198	$\pm$0.020	0.258	2	0.189	$\pm$0.008	1.000
	$S^*_{\rm min}$	1	0.178	$\pm$0.007		1	0.193	$\pm$0.014	
Sonar	R	25	0.266	$\pm$0.034		5	0.293	$\pm$0.035	
	$S^*_{\rm rank}$	16	0.299	$\pm$0.031	0.095	4	0.217	$\pm$0.031	0.298
	$S^*_{\rm min}$	23	0.311	$\pm$0.034		7	0.281	$\pm$0.030	
Spam	R	54	0.099	$\pm$0.005		57	0.095	$\pm$0.005	
	$S^*_{\rm rank}$	43	0.082	$\pm$0.004	0.592	32	0.074	$\pm$0.004	0.637
	$S^*_{\rm min}$	26	0.079	$\pm$0.004		25	0.080	$\pm$0.004	
Spect	R	25	0.277	$\pm$0.025		41	0.286	$\pm$0.028	
	$S^*_{\rm rank}$	16	0.278	$\pm$0.023	0.114	4	0.308	$\pm$0.025	0.566
	$S^*_{\rm min}$	12	0.274	$\pm$0.028		6	0.313	$\pm$0.025	
Vehicle	R	11	0.548	$\pm$0.021		17	0.564	$\pm$0.018	
	$S^*_{\rm rank}$	4	0.452	$\pm$0.018	0.707	4	0.443	$\pm$0.015	0.936
	$S^*_{\rm min}$	8	0.478	$\pm$0.020		4	0.420	$\pm$0.016	
Votes	R	9	0.094	$\pm$0.019		6	0.085	$\pm$0.017	
	$S^*_{\rm rank}$	1	0.034	$\pm$0.012	1.000	1	0.033	$\pm$0.011	1.000
	$S^*_{\rm min}$	2	0.043	$\pm$0.013		4	0.044	$\pm$0.013	