

Hybrid Feature Selection Method for Supervised Classification Based on Laplacian Score Ranking

Saúl Solorio-Fernández, J. Ariel Carrasco-Ochoa, and José Fco. Martínez-Trinidad

National Institute for Astrophysics, Optics and Electronics
Luis Enrique Erro # 1, 72840, Santa María Tonantzintla, Puebla, Mexico
{sausolofer, ariel, fmartine}@ccc.inaoep.mx

Abstract. In this paper, we introduce a new hybrid filter-wrapper method for supervised feature selection, based on the Laplacian Score ranking combined with a wrapper strategy. We propose to rank features with the Laplacian Score to reduce the search space, and then we use this order to find the best feature subset. We compare our method against other based on ranking feature selection methods, namely, Information Gain Attribute Ranking, Relief, Correlation-based Feature Selection, and additionally we include in our comparison a Wrapper Subset Evaluation method. Empirical results over ten real-world datasets from the UCI repository show that our hybrid method is competitive and outperforms in most of the cases to the other feature selection methods used in our experiments.

Keywords: Supervised Feature Selection, Laplacian Score, Feature Ranking.

1 Introduction

Feature selection has been an active research area in Pattern Recognition, Data Mining and Machine Learning. The main idea of feature selection is to get a subset of features for representing the data, so that those features with low relevance are eliminated and therefore they are not taken into account for further analysis. The main goal of feature selection for supervised classification is to find a feature subset that would produce high classification accuracy [25]. Feature selection not only reduces the size of the data and run-time of learning algorithms, but also leads to more compact learning models and possibly with better generalization capability [13].

There are two main approaches for feature selection in supervised classification: filter methods and wrapper methods. The filter methods [28,30], perform the selection based on inherent properties of data such as variance, entropy, mutual information or correlation, these methods are generally fast and scalable [16]. Meanwhile, wrapper methods [12,20,26] select feature subsets based on the precision of a specific classifier. Wrapper methods are often characterized by a high quality of the selected feature subsets, but these methods have a high computational cost [16]. Additionally, hybrid filter-wrapper methods have been developed for supervised feature selection [21,24], these methods attempt to have a reasonable compromise between efficiency (computational effort) and effectiveness (accuracy of classification).

In this paper, we propose a new hybrid filter-wrapper feature selection method for supervised classification based on the Laplacian Score ranking combined with a wrapper strategy. Experiments show that this combination selects relevant features that provide good accuracy.

The remainder of this paper is organized as follows. Section 2, gives a brief description of each one of the supervised feature selection methods compared in our experiments. Section 3 describes the Laplacian Score. In section 4, we present the proposed method. Section 5 shows experimental results. Finally, section 6 provides some conclusions and future work.

2 Feature Selection Techniques in Supervised Classification

There is a considerable amount of literature about feature selection [14,15,20,24,25] for supervised classification, an excellent survey about feature selection algorithms can be found in [11]. In this section, we only present some ranking based methods for supervised feature selection, which are some of the most used in the last years of feature selection for Pattern Recognition and Machine Learning [18]. We also include the description of the WPR method, which is also used in our experiments.

2.1 Information Gain Attribute Ranking (IG)

This is one of the simplest (and fastest) feature ranking methods and it is often used in text categorization [22]. The idea behind IG is to select features that reveal most information about the classes. Ideally, such features are highly discriminative and their values occur mostly in a single class.

2.2 Relief (RLF)

Relief is an instance based feature ranking scheme introduced by Kira and Rendell [30] and later enhanced by Kononenko [28]. Relief works by randomly sampling an instance from the data and then locating its nearest neighbor from the same and opposite classes. In Relief a relevance weight is given to each feature for reflecting its ability to discern between classes. An overview of this algorithm can be found in [5].

2.3 Correlation-Based Feature Selection (CFS)

Correlation-Based Feature Selection [23] evaluates feature subsets rather than individual features. The heart of the algorithm is a subset evaluation heuristic that takes into account the usefulness of individual features for predicting the class along with the level of intercorrelation among them. The heuristic of CFS assigns high scores to subsets containing features that are highly correlated with the class and have low intercorrelation with each other.

2.4 Wrapper Subset Evaluation (WRP)

As described at the section 1, wrapper feature selection methods use a classifier algorithm to estimate the worth of feature subsets [26]. The WRP method performs a

greedy forward search through the space of feature subsets, starting with the empty set and stopping when the addition of any remaining features results in an accuracy decrease. In this method, Cross-Validation is used to estimate the accuracy of the classifier for a set of features.

3 Laplacian Score

Recently, a family of feature selection algorithms based on the spectral graph theory has been developed [1,3,4,7,9]. This family of algorithms evaluates features according to their agreement with the graph Laplacian matrix of similarities of the data.

Formally, given a dataset consisting of m vectors $\{\mathbf{x}_i\}_{i=1}^m$, a matrix of similarities $W_{m \times m}$ that represents the similarity or adjacency between \mathbf{x}_i and \mathbf{x}_j data points (instances) can be constructed. We can interpret W as a weighted graph, whose nodes represent the instances, and the set of edges contains a connection for each pair of nodes i, j with weight w_{ij} , depending on the type of the graph one wants to use¹, such as the k-nearest neighbor or the fully connected graph [6] (Laplacian Score generally use k-nearest neighbor). The Laplacian matrix L is defined as:

$$L = D - W \tag{1}$$

Where D is a diagonal matrix such that $d_{ii} = \sum_{j=1}^m w_{ij}$.

Given a graph G , the Laplacian matrix L is a linear operator on a vector $\mathbf{f} \in \mathbb{R}^m$ [6,7], where:

$$\mathbf{f}^T L \mathbf{f} = \frac{1}{2} \sum_{i \neq j} w_{ij} (f_i - f_j)^2 \tag{2}$$

Equation (2) quantifies how much the vector \mathbf{f} locally varies in G [1]. This fact motivates to use L to measure on a vector of values of a feature, the consistency of this feature regarding the structure of the graph G . A feature is consistent with the structure of a graph if it takes similar values for instances that are near each other in the graph, and dissimilar values for instances that are far from each other. Thus a consistent feature would be relevant to separate the classes [7,6].

The Laplacian Score [4], proposed by X. He et al. [9], assesses the significance of individual features taking into account the local preserving power and its consistency with the structure of the similarity graph.

If we denote $\mathbf{f}_r = (f_{r1}, f_{r2}, \dots, f_{rm})^T$ with $r = 1, 2, \dots, n$, as the r -th feature and its values for the m instances. Then the Laplacian Score for \mathbf{f}_r is calculated as:

$$L_r = \frac{\tilde{\mathbf{f}}_r^T L \tilde{\mathbf{f}}_r}{\tilde{\mathbf{f}}_r^T D \tilde{\mathbf{f}}_r} \tag{3}$$

Where $\tilde{\mathbf{f}}_r = \tilde{\mathbf{f}}_r - (\mathbf{f}_r^T D \mathbf{1} / \mathbf{1}^T D \mathbf{1}) \mathbf{1}$, L is the Laplacian matrix of the graph G , D is the degree diagonal matrix, and $\tilde{\mathbf{f}}_r$ represents the deviation from the mean of all observations of the \mathbf{f} vector.

¹ For the Laplacian Score, we construct the graph putting an edge between two nodes if they belong to the same class and they are among the k nearest neighbors of each other.

For the Laplacian Score, the local structure of the data space is more important than the global structure [9]. In order to model the local structure, this method constructs a k -nearest neighbor graph, where k is the degree of neighborhood for each instance in the graph (see [9] for details). This value must be specified a priori by the user and it models local neighborhood relations between instances.

According to the Laplacian Score, a "good" feature should have a small value for L_r [9]. Thus, the features are arranged in a list according to their relevance. Those features that are at the top of the list are those with smaller values for L_r ; these features will be considered as the most important.

4 Proposed Method

In this section, we introduce the proposed feature selection method, which follows a hybrid filter-wrapper strategy and consists of two basic steps: I) Feature Ranking. II) A wrapper selection of a subset of relevant features that provides high accuracy for classification. In the first step (filter strategy), features are sorted according to their relevance by applying the Laplacian Score. In this step, we aim to identify the features that are consistent with the structure of the data and they are sorted according to their relevance, in order to narrow the search space of possible subsets of features

```

HIBRID LAPLACIAN SCORE FEATURE SELECTION METHOD (dataset, gnd, kLS)

{dataset: Dataset with  $m$  instances and  $n$  features
 gnd: label information for each data point
 kLS: number of neighbors for the Laplacian Score Graph }

Begin
  ACCBest  $\leftarrow -\infty$ 
  indSbest  $\leftarrow \emptyset$ 
  S  $\leftarrow \emptyset$ ;
  {rank of features with Laplacian Score}
  indRank  $\leftarrow \text{laplacianScore}(\text{dataset}, \text{gnd}, k_{LS})$ 

  for  $i=1$  to  $n$  do
    S  $\leftarrow S \cup \text{indRank}[i]$ 
    dataCand  $\leftarrow \text{dataset}[S]$ 
    {Run Classifier with 10-fold cross-validation for dataCand}
    AverageACC  $\leftarrow \text{Classifier}(\text{dataCand}, \text{gnd})$ 
    If AverageACC > ACCBest then
      ACCBest  $\leftarrow$  AverageACC
      indSbest  $\leftarrow S$ 
    end if
  end for
  return {indBest, ACCBest}
end

```

Fig. 1. Pseudocode for the Hybrid Laplacian Score Feature Selection method (LS-FS)

(2^n subsets) and starting the second step with a good approximation. In the second step (wrapper stage), the idea is to evaluate the features considered as a subset rather than individually; in this step, the algorithm builds n feature subsets: the first set only contains the top ranked feature, the second set contains the two top-ranked features, the third set contains the three top-ranked features, and so on (see algorithm in Fig. 1). In our method, to evaluate the accuracy of the subsets of features, we use the target classifier. The accuracy of the classification model generated by each feature subset is estimated using 10-fold cross-validation.

The pseudocode of our method is described in the algorithm of figure 1, called Hybrid Laplacian Score Feature Selection Method (LS-FS). In the algorithm of figure 1, *Classifier* can be any supervised classifier.

5 Experimental Results

In order to show the performance of the proposed method, we compared it against Information Gain Attribute Ranking [22], Relief [30], Correlation-based Feature Selection [23], and Wrapper Subset Evaluation [26] methods. For the comparison, we used ten real datasets taken from the UCI repository [8]. The details of these datasets are shown in Table 1.

In order to compare the effectiveness for feature selection, the feature sets chosen by each selector were tested with three classifiers: a decision tree (C4.5 [29]), a probabilistic classifier (naïve Bayes [27]) and k-NN [31], an instance-based classifier. These three classifiers were chosen because they represent three quite different approaches for supervised classification.

In order to evaluate the accuracy with and without feature selection the percentage of correct classification (Accuracy), averaged over ten-fold cross validation, was calculated for each selector-dataset combination before and after feature selection.

For IG and RLF selectors, which are based on ranking, dimensionality reduction was accomplished using the wrapper step of our proposed method.

To perform the experiments we used Weka² [2] (Waikato Environment for Knowledge Analysis) an open-source Java-based machine learning workbench.

Table 1. Details of the used datasets

Datasets	No. of Instances	No. of Features	No. of Classes
Iris	150	4	3
Wine	178	13	3
Ionosphere	351	34	2
Sonar	208	60	2
Pima	768	8	2
Wdbc	568	30	2
Glass	213	9	7
Monks-3	432	6	2
Parkinsons	194	22	2
Vehicle	845	18	4

In the Laplacian Score, the only parameter that can vary is k (the number of neighbors considered for building the graph), which could affect the selected subsets

² <http://www.cs.waikato.ac.nz/~ml/weka/>

of features, usually k must be equal to or greater than 5 [9]. In our experiment we have taken $k = 33\%$ of the size of the dataset, this value according to Liu Rongyan [3] is close to the "optimum".

Table 2 shows the results for feature selection with naive Bayes. This table shows how often each feature selection method performs significantly better (denoted as *) or worse (denoted as ●) than performing no feature selection (column 2). We consider that the results are significantly different if the difference is statistically significant at a 5% level according to a paired two-sided t test [19, 10]. From Table 2 it can be seen that all the tested selectors are on average better than no selection, although WRP is the best selector in this experiment, which improves significant the performance on four datasets. LS-FS (proposed method) got the second best average with two significant improvements and no degradation. IG got the third best average, which improves the performance on one datasets. RLF and CFS were the fourth and fifth best feature selection methods respectively for the Naïve Bayes Classifier.

Table 2. Results of feature selection for naive Bayes

Datasets	NB (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	82.62	83.19	87.76	89.76	88.62*	90.90*
Iris	96.00	96.66	95.33	95.33	96.00	94.00
Monks-3	92.35	95.60	94.69	93.99	95.85	94.67
Pima	76.31	76.30	75.14	75.65	77.86	75.13
Sonar	67.88	78.36*	74.52	70.69	68.36	75.07
Wdbc	93.15	94.19	93.32	94.55	94.73	96.66*
Wine	96.63	97.19	97.19	97.19	97.22	96.60
Glass	46.00	49.76	48.35	45.06	46.90	57.27
Parkinsons	69.07	79.38*	83.95*	82.34*	77.71	80.32*
Vehicle	46.03	46.03	45.68	49.46*	45.55	53.96*
Average	76.60	79.66	79.59	79.40	78.88	81.45

*, ● statistically significant improvement or degradation

Table 3 shows the results for feature selection with k-NN (using $k=1$). We can see that the best results were obtained by our proposed method (LS-FS), which significant improves the performance on two datasets, and it was the best on average. Similar results were obtained with $k=3$ and $k=5$ (see table 4 and table 5) respectively, in which the proposed method was the best on average compared with the other features selectors.

Table 3. Results of feature selection for k-NN (using $k=1$)

Datasets	k-NN (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	86.33	89.74	88.90	89.17	89.17	87.75
Iris	95.33	96.0	94.00	94.00	96.00	94.0
Monks-3	76.18	100.0*	97.21*	100.00*	96.07*	100.00*
Pima	70.17	70.44	68.88	68.74	70.97	69.79
Sonar	86.57	87.50	86.57	87.48	84.57	82.6
Wdbc	95.25	96.12	95.07	95.07	95.08	95.07
Wine	94.97	97.19	97.19	94.93	96.08	94.9
Glass	69.52	77.46*	75.63	74.22	78.46*	69.96
Parkinsons	95.29	96.39	94.24	96.37	88.18	90.24
Vehicle	70.28	71.24	71.35	69.46	61.18●	67.68
Average	83.99	88.20	86.90	86.94	85.58	85.2

*, ● statistically significant improvement or degradation

Table 4. Results of feature selection for k-NN (using k=3)

Datasets	k-NN (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	86.60	88.31	89.76	90.33	88.90	90.04
Iris	95.33	96.66	95.33	95.33	94.67	94.67
Monks-3	97.21	100.0*	97.21	97.21	96.07	100.0*
Pima	72.65	74.08	70.05	73.17	73.70	72.65
Sonar	86.02	85.57	85.55	83.14	83.17	77.38●
Wdbc	96.48	97.18	96.66	96.13	95.60	95.43
Wine	94.97	97.19*	97.16	96.60	97.22*	93.79
Glass	69.48	75.11*	73.77	74.24	69.09	73.77
Parkinsons	95.36	94.32	95.84	88.68	94.82	95.84
Vehicle	70.29	74.55	73.84	57.15●	68.16	73.84
Average	86.44	88.29	87.52	85.20	86.14	86.74

*, ● statistically significant improvement or degradation

Table 5. Results of feature selection for k-NN (using k=5)

Datasets	k-NN (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	84.90	88.88	90.04	90.03	87.48	90.90*
Iris	95.33	96.66	95.33	95.33	96.67	93.33
Monks-3	92.12	100.0*	97.21*	97.21*	96.07*	100.00*
Pima	73.18	74.68	73.17	73.69	72.92	72.40
Sonar	84.62	86.05	80.71	79.29	83.12	81.69
Wdbc	96.13	97.18	96.66	96.48	96.66	95.78
Wine	95.52	96.62	97.16	95.46	96.11	96.63
Glass	69.09	72.30	71.39	71.39	71.39	71.41
Parkinsons	92.74	94.32	90.74	94.29	88.66	92.21
Vehicle	71.12	71.83z	71.48	72.67	61.77●	68.04
Average	85.48	87.85	86.39	86.58	85.08	86.24

*, ● statistically significant improvement or degradation

In table 6, we can observe that the proposed method (LS-FS) improves significantly on one dataset and also was the best on average for C4.5 classifier. RLF was the second best on average although there was no significant improvement or degradation in any dataset. IG was the third best selector with one degradation. WPR was the fourth best selector with no significant improvement or degradation. On the contrary, CFS degraded the performance in two datasets; this selector was the worst for C4.5 classifier.

Table 6. Results of feature selection for C4.5

Datasets	C4.5 (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	91.46	91.45	92.89	92.6	89.75	91.46
Iris	96.00	96.00	93.33●	94.0	96.00	92.67
Monks-3	100.0	100.0	100.00	100.0	96.07●	100.0
Pima	73.83	74.47	73.18	73.71	75.01	73.44
Sonar	71.17	79.80*	77.86	75.93	73.50	74.05
Wdbc	94.02	94.71	94.38	95.78	94.38	94.02
Wine	93.86	94.94	96.08	94.41	93.86	94.41
Glass	67.60	71.83	69.44	68.48	68.94	65.24
Parkinsons	85.56	87.72	87.11	90.18	89.16	87.13
Vehicle	71.95	72.66	71.46	71.82	65.78●	71.11
Average	84.55	86.36	85.57	85.69	84.25	84.35

*, ● statistically significant improvement or degradation

Finally, in table 7 we report the run-time³ required by each one of the selectors. In this table we can see that the best results are achieved by the proposed method for the three classifiers. It is important to highlight the superiority of our method over the other methods.

Table 7. Run-time of feature selection methods (in seconds)

Datasets	LS-FS			IG			RLF			CFS			WRP		
	Bayes	KNN	C4.5	Bayes	KNN	C4.5	Bayes	KNN	C4.5	Bayes	KNN	C4.5	Bayes	KNN	C4.5
Ionosphere	3.21	1.94	8.88	4.02	14.51	36.33	4.19	14.23	37.48	4.10	14.62	37.51	6.39	29.08	43.4
Iris	0.14	0.10	0.13	0.05	0.14	0.14	0.05	0.17	0.13	0.06	0.15	0.14	0.09	0.35	0.22
Monks-3	0.27	0.31	0.28	0.24	0.91	0.19	0.27	1.20	0.26	0.21	1.00	0.18	0.35	2.43	0.47
Pima	0.63	0.94	1.15	0.47	3.99	3.51	0.76	4.44	3.51	0.46	3.80	3.62	1.06	7.86	4.15
Sonar	5.61	2.98	15.23	6.49	17.87	52.55	6.95	17.89	57.02	5.98	18.83	57.13	4.65	59.76	66.6
Wdbc	3.88	3.03	9.08	5.61	20.30	31.39	6.06	22.24	34.43	5.67	21.91	37.97	6.91	81.15	40.0
Wine	0.56	0.29	0.78	0.40	0.83	1.61	0.44	0.92	1.65	0.40	0.86	1.86	1.63	4.06	3.19
Glass	0.43	0.22	0.95	0.30	0.61	2.19	0.33	0.78	2.26	0.31	0.66	2.64	0.48	2.99	6.08
Parkinsons	1.11	0.59	2.25	0.81	2.10	5.77	0.95	2.30	6.51	0.90	2.28	6.89	1.03	9.24	8.38
Vehicle	2.41	2.55	8.26	2.05	12.33	24.90	3.04	14.44	28.91	2.55	16.75	28.58	3.36	63.50	82.5
Average	1.83	1.30	4.70	2.04	7.36	15.86	2.30	7.86	17.22	2.07	8.08	17.65	2.59	26.04	25.53

6 Conclusion and Future Work

In this paper, we have presented a new hybrid filter-wrapper method for supervised feature selection based on the Laplacian Score ranking.

From the experiments we can conclude that the proposed method performs better using the k-NN and C.45 classifiers, where it is better than the other evaluated feature selectors, reaching on average a precision around 87%. For the Naïve Bayes classifier, the proposed method was the second best feature selector after WRP, reaching on average precision of 79.6%. Also, it is found that the proposed method requires less run-time compared to other feature selection methods.

As future work, we will explore the use of other classifiers and other feature subset search strategies in the wrapper step of our method.

References

1. García, D.G., Rodríguez, R.S.: Spectral clustering and feature selection for microarray data. In: Fourth International Conference on Machine Learning and Applications, pp. 425–428 (2009)
2. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations 11(1) (2009)
3. Liu, R., Yang, N., Ding, X., Ma, L.: An unsupervised feature selection algorithm: Laplacian Score combined with distance-based entropy measure. In: Workshop on Intelligent Information Technology Applications, vol. 3, pp. 65–68 (2009)

³ The run-times reported in this paper were obtained using a computer with an Intel Core i5 2.27GHz, with 4MB RAM.

4. Nijjima, S., Okuno, Y.: Laplacian linear discriminant analysis approach to unsupervised feature selection. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 6(4), 605–614 (2009)
5. Jensen, R., Shen, Q.: Computational intelligence and feature selection: rough and fuzzy approaches, pp. 61–84. Wiley, Chichester (2008)
6. von Luxburg, U.: A tutorial on spectral clustering. *Statistics and Computing* 17(4), 395–416 (2007)
7. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: *ICML '07: Proceedings of the 24th International Conference on Machine learning*, pp. 1151–1157. ACM, New York (2007)
8. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository. School of Information and Computer Science. University of California, Irvine (2007), <http://www.ics.uci.edu/~mlearn/MLRepository.html>
9. He, X., Cai, D., Niyogi, P.: Laplacian Score for feature selection. In: Weiss, Y., Schölkopf, B., Platt, J. (eds.) *Advances in Neural Information Processing Systems*, vol. 18, pp. 507–514. MIT Press, Cambridge (2006)
10. Witten, I.H., Frank, E.: *Data Mining: Practical machine learning tools and techniques*, 2nd edn. Morgan Kaufmann, San Francisco (2005)
11. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering* 17(4), 491–502 (2005)
12. Loughrey, J., Cunningham, P.: Using Early-Stopping to Avoid Overfitting in Wrapper-Based Feature Selection Employing Stochastic Search. Technical Report (TCD-CS-2005-37). Department of Computer Science, Trinity College Dublin, Dublin, Ireland (2005)
13. Pal, S.K., Mitra, P.: *Pattern Recognition Algorithms for Data Mining*, pp. 59–82. Chapman & Hall/CRC (2004)
14. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. *J. Mach. Learn. Res.* 5, 1205–1224 (2004)
15. Zhang, L., Sun, G., Guo, J.: Feature selection for pattern classification problems. In: *International Conference on Computer and Information Technology*, pp. 233–237 (2004)
16. Guyon, I.: An introduction to variable and feature selection. *Journal of Machine Learning Research* 3, 1157–1182 (2003)
17. Kim, Y.S., Nick Street, W., Menczer, F.: Feature selection in data mining, pp. 80–105 (2003)
18. Hall, M.A., Holmes, G.: Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data Engineering* 15, 1437–1447 (2003)
19. Nadeau, C., Bengio, Y.: Inference for the generalization error. *Mach. Learn.* 52(3), 239–281 (2003)
20. Xing, E.P., Jordan, M.I., Karp, R.M.: Feature selection for high-dimensional genomic microarray data. In: *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 601–608 (2001)
21. Das, S.: Filters, wrappers and a boosting-based hybrid for feature selection. In: *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 74–81. Morgan Kaufmann Publishers Inc., San Francisco (2001)
22. Dumais, S., Platt, J., Heckerman, D., Sahami, M.: Inductive learning algorithms and representations for text categorization. In: *Proceedings of the International Conference on Information and Knowledge Management*, pp. 148–155 (1998)
23. Hall, M.A.: Correlation-based feature selection for machine learning. PhD thesis, Department of Computer Science, University of Waikato, Hamilton, New Zealand (1998)

24. Dash, M., Liu, H.: Hybrid search of feature subsets. In: PRICAI'98: Topics in Artificial Intelligence, pp. 238–249 (1998)
25. Dash, M., Liu, H.: Feature selection for classification. *Intelligent Data Analysis* 1, 131–156 (1997)
26. Kohavi, R., John, G.H.: Wrappers for feature subset selection. *Artificial Intelligence* 97, 273–324 (1997)
27. John, G.H., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers. In: Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, pp. 338–345 (1995)
28. Kononenko, I.: Estimating attributes: Analysis and extensions of relief. In: Proceedings of the Seventh European Conference on Machine Learning, pp. 171–182. Springer, Heidelberg (1994)
29. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo (1993)
30. Kira, K., Rendell, L.: A practical approach to feature selection. In: Proceedings of the Ninth International Conference on Machine Learning, pp. 249–256. Morgan Kaufmann, San Francisco (1992)
31. Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. *IEEE Transactions on Information Theory* 13(1), 21–27 (1967)