Journal of Intelligent & Fuzzy Systems 18 (2007) 157–169
IOS Press

Hierarchical fuzzy filter method for unsupervised feature selection

Yun Li[a,*], Bao-Liang Lu[a] and Zhong-Fu Wu[b]
[a]*Department of Computer Science and Engineering, Shanghai JiaoTong University, 800 Dongchuan Rd, Minhang, Shanghai, P. R. China, 200240*
[b]*College of Computer Science, ChongQing University, 174 Shazheng Rd, ChongQing, P. R. China, 400044*

Abstract. The problem of feature selection has long been an active research topic within statistics and pattern recognition. So far, most methods of feature selection focus on supervised data where class information is available. For unsupervised data, the related methods of feature selection are few. The presented article demonstrates a way of unsupervised feature selection, which is a two-level filter model removing the redundant and irrelevant features, respectively. The redundant features are eliminated using any clustering algorithm, and a new method is proposed to remove the irrelevant features: first rank the features according to their relevance to cluster and then a subset of relevant features is selected using the Fuzzy Feature Evaluation Index (FFEI) with some changes and extensions. The experimental results have shown the effectiveness of the proposed method for high-dimensional data. Our major contributions are: (1) to present a new hierarchical filter method for unsupervised feature selection; (2) to propose a new algorithm for removing the irrelevant features; (3) to extend the FFEI, and present a method for calculating the approximate weight of feature in FFEI, which improves the efficiency and robustness of the method.

Keywords: Unsupervised feature selection, fuzzy set, ranking index, filter method

1. Introduction

One of the key problems that arise in a great variety of fields, including pattern recognition and machine learning, is the so-called feature selection. Feature selection not only obtains better accuracy of the predictor and improved scalability but also leads to better data visualization and understanding, reduction of measurement and storage requirements, and reduction of training and inference time. Finding the optimal set of features is intractable, and many problems related to feature selection have been shown to NP-hard [1]. As a result, we are forced to find heuristic methods that represent a compromise between solutions quality and cost.

Algorithms that perform feature selection can generally be categorized into two classes: Filters and Wrappers. The former considers the feature selection as a preprocessing step and independent of the learning algorithm. For the latter, feature selection is wrapped around the learning algorithm and the learning result is used as the evaluation criterion. In general, the characteristics of Filters are low time cost and not better effect. On the contrary, the time cost of Wrappers is high for its calling the learning algorithm to evaluate candidate subset of considered features, but the effect is better to predetermined learning algorithm. In recent years, data have become increasingly larger in the number of both instances and features. When the number of features is large, the Filters model is usually chosen due to its computation efficiency.

When class labels of the data are available we use supervised feature selection, otherwise unsupervised feature selection is appropriate. In many applications, class labels are unknown, thereby indicating the significance of unsupervised feature selection there. For unsupervised feature selection, traditional feature selection algorithms for classification [18] do not work.

[*] Corresponding author. Tel.: +86 21 34204421; Fax: +86 21 34205422; E-mail: liyun_mail@sjtu.edu.cn.

1064-1246/07/$17.00 © 2007 – IOS Press and the authors. All rights reserved

Dimensionality reduction or feature extraction methods (e.g. Principal Components Analysis, Karhunen-Loeve transformation or Singular Value Decomposition) are commonly used. They have drawbacks such as: (1) it is difficult to understand the data using the extracted features, and (2) the original features remain, as they are required to determine the extracted features [19]. Some methods for unsupervised feature selection have been developed, such as methods that measure feature similarity to detect redundant features [23]. In [11,12, 20] the normalized log-likelihood and cluster separability are used to evaluate the quality of clusters obtained with different feature subsets, and the algorithm described in [20] is a filter method. In [19], the clustering performance of each feature is evaluated by an entropy index. A genetic algorithm is used in [26] for feature selection in k-means clustering. In [16], feature selection for symbolic data is addressed by assuming that irrelevant features are uncorrelated with the relevant features. The notion of "category utility" for feature selection in a conceptual clustering task is described in [21]. Recently a neuro-fuzzy approach [25] was developed and an EM (Expectation-Maximization) algorithm to estimate the importance of different features was proposed in [4]. A Bayesian approach to unsupervised feature selection is addressed in [24]. They all at least have one of the shortcomings listed below: only remove redundant features; only eliminate irrelevant features; low performance on high-dimensional data set; expensive computation cost for high-dimensional data or sensitive to noisy data. However, most real-world data sets have irrelevant, redundant features and noisy data. So we propose a new system for the problem of selecting a subset of important features for unsupervised learning, which is a two-level filter system including unsupervised redundant feature filter and unsupervised irrelevant feature filter to remove redundant and irrelevant features, respectively.

The organization of the article is as follows: in the next section, we describe the proposed system and the adopted algorithms in each level. In Section 3, a new irrelevant feature filter is presented. In Section 4, the detailed experimental results for synthetic and real-world data sets along with comparisons are presented. The paper ends with acknowledgements and conclusions in Section 5.

2. The proposed system

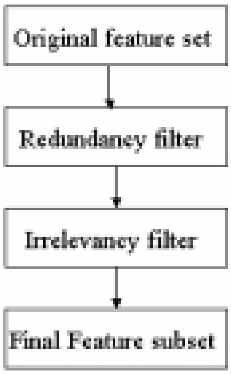


Fig. 1. Two-level filter model architecture.

The goal of our system is to reduce a large set of features to a small subset of features without significantly reducing the system's ability. Our approach, shown in Fig. 1, is a two-level filter system. First the redundant features are removed, and then the irrelevant features are eliminated. The idea is that each step uses a filter to reduce the number of candidate features, until finally only a small subset remains. Of course, the order of the filters partly depends on the time complexity of the adopted concrete algorithm in each filter. In other words, the algorithm with low time complexity should be implemented first to get the lowest total time cost, which is discussed in our earlier work [27].

For this system, the first filter is to remove redundant feature from original feature set, which can be completed using any clustering algorithm, such as C-means [9]. Here, we will eliminate the redundant features using a specific algorithm newly developed in [23] (we call it as Mitra's) for its high efficiency and low time cost, which is to find the subsets of feature that are highly correlated based on the k nearest neighbors principle. First compute k nearest features for each feature, among them the feature having the most compact subset, i.e. having the largest similarity to the farthest neighbor, is selected, and its k neighbors are discarded. The process is repeated for the remaining features until all of them are considered. For determining the k nearest neighbors of features, assigning a constant error threshold (ε), which is equal to the distance of kth nearest neighbor of selected feature in the first iteration. In the subsequent iterations, checking the distance between features whether it is greater than ε or not. If yes, then decreases the value of k. So k may be changing over iterations. In the algorithm k controls the degree of cluster, since k determines the error threshold (ε). It has shown, in many cases, $k + d \approx n$, where d is the size of selected feature set, n is the number of original features. Let the number of training instances be s, then the time complexity of Mitra's is $O\left(n^2 s\right)$.

The second filter is to remove the irrelevant features. In [19], Dash and Liu presented an unsupervised method to solve the problem, which is a wrapper method with huge computation cost. An EM algorithm to detect relevant features was proposed in [4], however, it cannot deal with high-dimensional data. In [16], Talavera addressed a method to select relevant symbolic feature. Relief algorithm [15] and its extensions [7] that identifying statistically relevant features have been reported, which are approximate to supervised feature selection. Here, we will propose a new unsupervised filter method-AIF. Then according to the system structure, an algorithm-MAIF is proposed, which integrates the Mitra's with AIF to eliminate the redundant and irrelevant features, i.e., to get the maximum relevance and minimum redundancy. It also can be represented as: Mitra's + AIF, the output feature subset of Mitra's is the input feature set of AIF.

3. The irrelevancy filter

To eliminate the irrelevant features with a filter model, we first rank the features according to their importance on clustering, i.e., their relevance to cluster. Then an evaluation criterion independent of the learning algorithm is used to get the relevant feature set. Feature ranking is a filter method: it is a preprocessing step, independent of the choice of the predictor. Still, under certain independence or orthogonal assumptions, it may be optimal with respect to a given predictor. Even when feature ranking is not optimal, it may be preferable to other variable subset selection methods because of its computational and statistical scalability: computationally, it is efficient since it requires only the computation of n scores and sorting the scores. Statistically, it is robust against overfitting because it introduces bias but it may have considerably less variance [5]. The adopted ranking index belongs to exponential entropy. The evaluation criterion is Fuzzy Feature Evaluation Index (FFEI), which is defined in [8,9,25].

3.1. Ranking index

Let $S_{p,q}$ be the similarity between two instances X_p and X_q, $S_{p,q}$ is high if the two instances are very close and $S_{p,q}$ is low if the two instances are far away. Let N be the number of samples on which the feature ranking index is computed, and the feature ranking index is defined as:

$$H = \sum_{p=1}^{N}\sum_{q=1}^{N} \left(S_{p,q} \times e^{(1-S_{p,q})} + (1 - S_{p,q}) \times e^{S_{p,q}}\right) \tag{1}$$

where $S_{p,q}$ takes value in [0–1]. When $S_{p,q} \rightarrow 0(1)$, the H decreases, however, $S_{p,q} \rightarrow 0.5$, H increases. In other words, the index H decreases as the similarity between two instances belonging to the same cluster or dissimilarity between two instances belonging to different cluster increases in the feature space. This is appropriate to character clustering performance of the selected feature set. The similarity measure is defined as in [19]:

1). For numeric data, we use Euclidean distance to calculate the similarity. Mathematically similarity for numeric data is given as: $S_{p,q} = e^{-\alpha D_{p,q}}$, where α is a parameter. In a multi-dimensional space, distance $D_{p,q}$ is defined as:

$$D_{p,q} = \left[\sum_{k=1}^{n}\left(\frac{X_{pk} - X_{qk}}{\max_k - \min_k}\right)^2\right]^{1/2} \tag{2}$$

where n is the number of features, X_{pk} and X_{qk} are values of kth feature of pth and qth patterns, $\max_k$ and $\min_k$ is the maximum and minimum values of the kth dimension in the feature space, respectively.

The interval in the kth dimension is normalized through dividing it by the maximum interval $(\max_k - \min_k)$ before calculating the distance. In this work, α is calculated automatically by assigning 0.5 in equation $\overline{S} = e^{-\alpha \times \overline{D}}$, the uncertainty is maximum for this value, so we get: $\alpha = -\ln^{0.5}\big/\overline{D}$, where $\overline{D}$ is the average distance among the training patterns.

2). For nominal data, we use the Hamming distance. The similarity between two data points is given as:

$$S_{p,q} = \frac{\sum\limits_{k=1}^{n} |X_{pk} = X_{qk}|}{n} \tag{3}$$

where $|X_{pk} = X_{qk}|$ is 1 if X_{pk} is equal to X_{qk} and 0 otherwise.

3). For data with both numeric and nominal features, we can discretize numeric values before utilizing the measure defined in 2).

3.2. Ranking algorithm

For ranking features we can use H in the following way: Each feature is removed in turn and H is calculated. If the removal of a feature results in the minimum

H, the feature is the least relevant; and vice versa. The minimum H indicates the removed feature has the least effect on the distribution of sample in the data set, so it has least influence on the cluster. The pseudocode of the ranking algorithm (RANK) is described as follows:

```
RH = H values for $n$ features
FOR $k = 1$ TO $n$
    $RH_k$ = CalH($F_k$)
END
OUTPUT Rank(RH)
```

In the algorithm, CalH(F_k) calculates the H value of the training data set after discarding feature F_k. On the other hand, how to deal with data with large size and high dimension? The experimental results in Section 4 will indicate the proposed algorithm is efficient to handle high dimensional data. For the data set with large number of data points, our ranking measure may not be practical as the complexity is $O(N^2n^2)$, so we use a scalable method that is based on random sampling for that a reasonably small random samples retain the original cluster information in most cases [19]. However, note that for H measure to work well the cluster structure needs to be retained and it is largely independent of the number of data points. The pseudocode of ranking algorithm for large size data (SRANK) is described as follows:

```
For all features $F_k'$s Overall Rank, $OR_k$ =0
FOR $l = 1$ TO $t$ # t is the number of random samples
    Take a sample $L_l$
    Run RANK to find rankings $R_l$
    FOR $k$= 1 TO $n$
        $OR_k$=$OR_k + R_{l_k}$
    END
END
OUTPUT OR
```

3.3. Evaluation criterion

Now, the problem of feature selection is how many features we should choose from the ranked feature list. There are several methods can be used to select the features from ranked list, such as: 1). If one knows the number of important features required, just pick them starting with the most important one. However, it is not practical without any prior knowledge. 2). In [19], Dash and Liu adopt a clustering algorithm and choose the subset maximizes the clustering quality, which is a wrapper method with high computational cost. At the same time, there is no commonly accepted evaluation approach of clustering performance. 3). In [6], using the RFE (Recursive Feature Elimination) to remove the feature with smallest ranking criterion. This iterative procedure is an instance of backward feature elimination. However, RFE has no effect on our method since the H is not invariant with respect to different numbers of features.

We adopt filter method for its low cost, and consider the Fuzzy Feature Evaluation Index (FFEI) as the evaluation criterion for several reasons: first, it has solid theory basis and can get better performance; second, the ranking index H and FFEI are all using Euclidean distance, so there is no bias; third, we propose a method to compute the weighting coefficient of the features in the evaluated set, which is achieved using the ranking index H of the feature. Then the feature ranking is seamlessly integrated with the evaluation criterion, which can improve the efficiency and robustness. When use the FFEI, we make some changes to it and extend it to the nominal data.

3.3.1. Fuzzy feature evaluation index

Let, μ_{pq}^{O} be the degree that both the pth and qth patterns belong to the same cluster in the n-dimensional original feature space, and μ_{pq}^{T} be that in the n'-dimensional ($n' \leqslant n$) transformed feature space. μ value determine how similar a pair of patterns are in the respective feature spaces, in other words, it may be interpreted as the membership value of a pair of patterns belonging to the fuzzy set "similar". Let s be the number of samples on which the feature evaluation index is computed. The FFEI is defined as:

$$\mathrm{FFEI} = \frac{2}{s\,(s-1)} \sum_{p} \sum_{p \neq q} \frac{1}{2} \left[\mu_{pq}^{T}\left(1-\mu_{pq}^{O}\right) + \mu_{pq}^{O}\left(1-\mu_{pq}^{T}\right)\right] \tag{4}$$

$$\mu_{pq} \begin{cases} = 1 - \frac{d_{pq}}{D} & \text{if } d_{pq} \leqslant D \\ = 0 & \text{otherwise} \end{cases}$$

where d_{pq} is the distance measure which indicates similarity between the pth and qth patterns in the feature space. D is a parameter that reflects the minimum separation between a pair of patterns belonging to two different clusters.

1). For numeric data: $D = \beta d_{\max}$, here $\beta \in (0,1)$ is a user defined constant parameter which determines the degree of flattening of the membership function Eq. (4), and is difficult to determine without any prior knowledge.

$$d_{\max} = \left[\sum_{k} \left(\max_k - \min_k\right)^2\right]^{1/2} \tag{5}$$

Table 1
The description of data sets and the ranking results as well as feature selection results

Data Sets	No. Features	No. Classes	Important Features' number	Ranking (Descending)	Selection Results
Iris	4	3	3,4	{3,4},2,1	3,4
Corral	6	2	1,2,3,4	{6,3,1,4,2,},5	6,3,1,4,2
Monk3	6	2	2,4,5	{5,4,2},1,6,3	5,4,2
Syndata2_6	6	2	1,2,3	{1,2,3},6,4,5	1,2,3
Syndata3_11	11	3	1–6	{2,3,6,1,4,5},8, . . .	2,3,6,1,4,5
Syndata4_15	15	4	1–5	{1,3,4,2,5},12, . . .	1,3,4,2,5
Syndata6_22	22	6	1–7	{5,6,1,2,7,3,4},9, . . .	5,6,1,2,7,3,4
Parity3 +3	12	2	{1,7}, {2,8}, {3,9}	{9,3,8,2,7,1},4, . . .	9,3,8,2,7,1

$$d_{pq} = \left[\sum_k \omega_k^2 \left(X_{pk} - X_{qk} \right)^2 \right]^{1/2} \tag{6}$$

where $\omega_k \in [0, 1]$ represents weighting coefficient corresponding to kth feature. X_{pk}, X_{qk}, $\max_k$ and $\min_k$ are the same as in Section 3.1.

2). For nominal data: D represents the minimum number of features to separate a pair of patterns, which is specified by user.

$$d_{pq} - \sum_k |X_{pk} = X_{qk}| \tag{7}$$

where $|X_{pk} = X_{qk}|$ is 1 if X_{pk} equals to X_{qk} and 0 otherwise.

The index decreases as the similarity between two patterns belonging to the same cluster or dissimilarity between two patterns belonging to different clusters increases in the original feature space. This means, if the inter-cluster/intra-cluster distances in the transformed space increase/decrease, the feature evaluation index of the corresponding set of feature decreases. Therefore, the objective of feature selection is to select those features for which the evaluation index becomes minimum.

3.3.2. *Computation of weight coefficient*

In order to break the assumption of the hyperspherical clusters and get better results, we consider the weighting coefficient of feature in the calculation of distance d_{pq}. How to calculate the weighting coefficient of feature? We propose a method (CalWeight) to get the approximate weight of feature as below:

Suppose we get a ranked feature set ($RF_1, \ldots,$ RF$_m$) ordered by the descending H value, where m is the number of features. LET RH_k be the H value of feature RF_k in the ranked feature set, certainly, RH_m is the minimum H value. SET Overall Difference of H value, *ODH* $= 0$, and the Difference of H values between feature RF_k and RF_m, $DH_k = 0$. The feature weight is its contribution to the overall difference of H value.

```
$DH_m = 1$
FOR $k = 1$ TO $m - 1$
  $DH_k = RH_k - RH_m$
  $ODH = ODH + DH_k$
END FOR
$ODH = ODH + DH_m$
FOR $k = 1$ TO $m$
  $\omega_k = DH_k / ODH$
END FOR
```

Because the difference of Hvalue between a ranked feature and RF_m is always large, so we initialize the DH_m =1 to represent the least influence of the feature RF_m to cluster. Of course, we can select other small number to initialize the DH_m. On the one hand, RH_k is the H value of feature $RF_{k.}$ and represents the clustering performance of the feature RF_k, which depends on the calculation of Euclidean distance. FFEI also utilizes the Euclidean distance to compute the similarity of patterns. On the other hand, from the Eqs (4), (5) and (6), we can get that the feature weight ω influences the similarity of patterns and the degree of patterns belonging to the same cluster. Then the feature weight ω also indicates the clustering performance of feature and has the resemble meaning to H value, so we can use the H value to calculate the feature weight, and then CalWeight works well.

3.4. *Algorithm of Irrelevancy Filter* (*AIF*)

In our case, features are already ranked according to their relevance, so the task of searching through the feature subset space of 2^n is avoided. The proposed search process is forward selection. In the algorithm we use the shorthand notation *FFEI*(*fs*) to denote the value of FFEI as computed with all the features in feature subset *fs* and use ϕ as a user specified threshold representing the minimum acceptable decrease in FFEI with each added feature. The process of selecting relevant feature set is described as follows:

Table 2
The ranking results of high dimensional synthetic data set Syndata5–100

# Run	Sample Sizes		
	0.25%	0.50%	1.0%
1	{1,4,11,8,18,9,20,17,10,12,19,7,3,14, 16,15,2,5,13,6},27,83,91,...	{12,17,19,11,3,4,9,1,5,16,7,10,20,13, 15,6,14,18,8,2},44,43,94,...	{4,1,10,11,17,8,15,2,3,19,18,7,6,14, 12,20,16,9,13,5},69,64,52,...
2	{10,1,14,12,7,8,20,3,6,11,9,5,18,17,4, 13,19,16,15,2},60,25,99,...	{2,8,6,3,13,4,14,12,16,9,17,1,5,19,7,15, 18,20,10,11},92,93,44,...	{16,8,10,9,12,18,20,14,4,5,3,11,2,15,1, 17,19,7,13,6},91,65,51,...
3	{16,9,7,8,15,18,11,17,4,6,19,12,1,3,20, 13,2,14,10,5},41,92,71,...	{6,2,15,19,4,9,13,3,18,12,14,8,16,7,17, 10,20,1,11,5},21,64,48,...	{7,13,4,20,6,10,5,1,12,3,14,15,8,2,17,9, 16,18,11,19},82,25,72,...
The Feature Selection Results	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20		

```
Run RANK to get ranked features $OR = (RF_1, RF_2, \ldots, RF_n)$ and $RH_k$, $k = 1, \ldots, n$
  Let $fs = \{RF_1\}$
    FOR $k = 2$ TO $n$
      Calculate $FFEI(fs')$ where $fs' = fs \cup RF_k$
        IF $(FFEI(fs) - FFEI(fs')) > \phi$
          $fs = fs'$
          CONTINUE
        ELSE
          BREAK
        END IF
      END FOR
```

Return *fs* as the candidate subset. In case of large data set, we run SRANK instead of RANK.

The threshold ϕ is very difficult to determine for different data sets without the prior knowledge. However, from the theoretic analysis above and experimental results below, FFEI value decreases initially and once all important features are added, it either goes up or remains relatively unchanged for any addition of the unimportant features. The point at which FFEI gets minimum value or remains approximately unchanged is not difficult to detect visually, hence we can manually determine the stopping criterion through finding this point on the plot of the FFEI change instead of prespecifying the threshold ϕ. This method is practical and easy to implement. Of course, the fully automatic feature selection is our target.

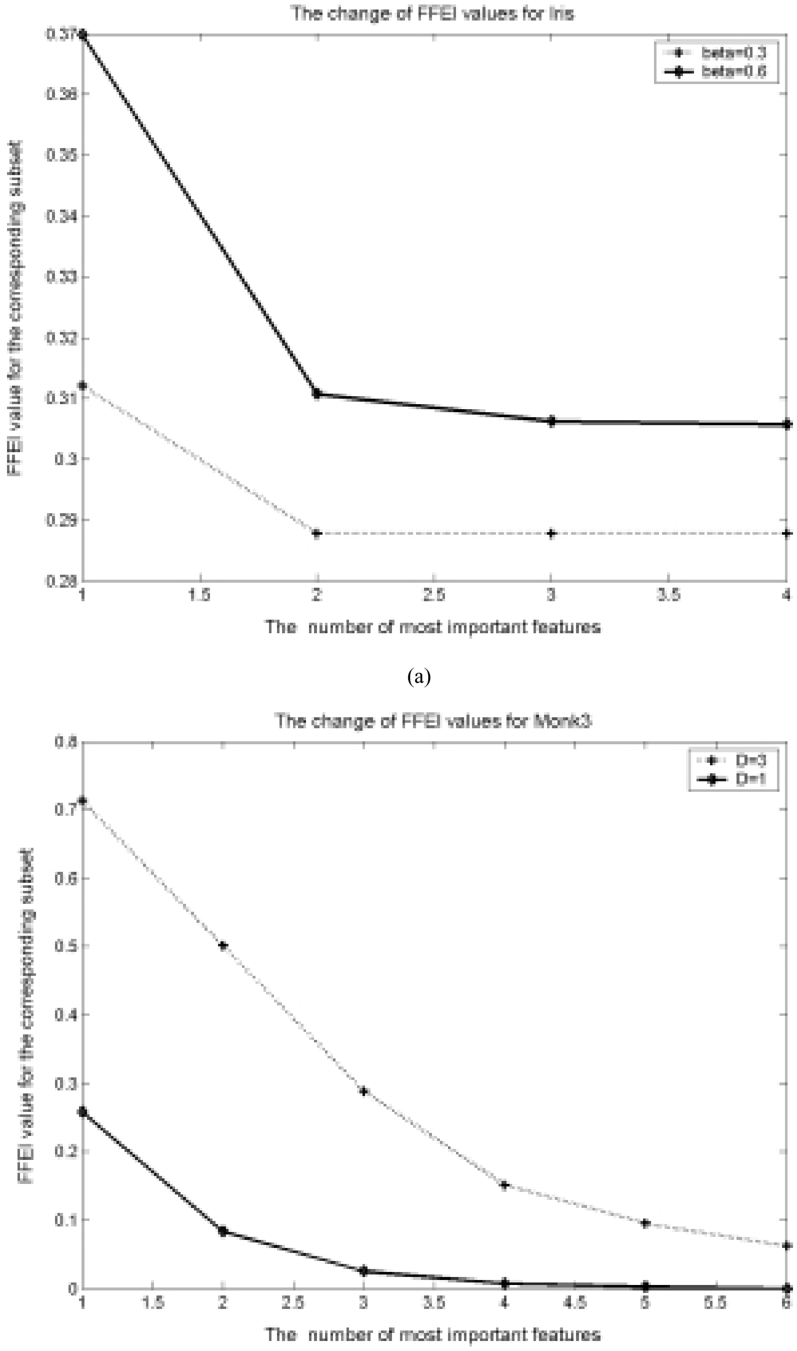
Computation Complexity Analysis: For RANK, let N be the number of sample, n be the number of features. When one feature is removed, the distance between all pairs of samples should be calculated, the time complexity is $O(N^2(n-1))$. For every feature, the total time complexity is $O(N^2n^2)$, and the time complexity of ranking features is $O(n)$, so the time complexity of RANK is $O(N^2n^2) + O(n) \approx O(N^2n^2)$. In FFEI, Let s be the number of samples on which the feature evaluation index is computed. For one evaluated feature subset with m features ($m = 1, 2, \ldots, n$), the distance between all pairs of samples should be calculated, the time complexity is $O(s^2m)$. For every evaluated feature subset, the total time cost is $O(s^2)+O(2s^2)+\ldots+O(ns^2) \approx O(s^2n^2)$. In the end, the time complexity of AIF is $O(N^2n^2) + O(s^2n^2)$. If $N \leqslant s$, $O(N^2n^2) + O(s^2n^2) \approx O(s^2n^2)$. In practice, the size of the selected subset is much lesser than n and the complexity is significantly low. In addition, when the distance computation is done in parallel, then the actual time cost is low.

4. Experiments

We empirically tested our feature selection method on different data sets. The whole experiment including two parts: one for the test of irrelevancy filter-AIF, the other for the integrated two-level filter method-MAIF. First, experiments are conducted on benchmark and synthetic data sets to check the correctness of our claim that AIF can select relevant features, as we know well about these data sets. Tests are also conducted on large high-dimensional data sets to test the performance of SRANK and MAIF. We used a MATLAB random function to generate synthetic data. For synthetic data sets a few features are chosen as important and these features follow Gaussian distribution. Each cluster is of equal size if not mentioned otherwise. Clusters are usually overlapping. Unimportant features are added which take uniformly random values.

4.1. Data sets

Low dimensional data sets: Four synthetic low-dimensional data sets (Syndata2_6, Syndata3_11, Syndata4_15, Syndata6_22) are generated using the method described above with different numbers of clusters and features. Benchmark data sets (both numeric and nominal) are selected from UCI machine learning repository [2]. The Monks and Corral are discrete data sets. The details of these data sets are described in Table 1, the second and third column (from the left) describes the number of features and classes of the data sets, re-



(a)

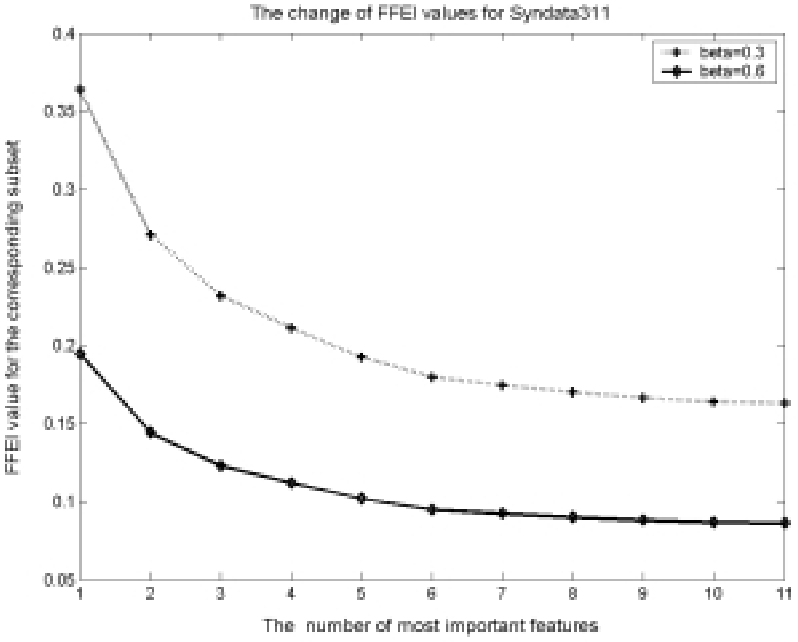
(b)

Fig. 2. The change of FFEI values for (a) Iris, (b) Monk3, (c) Syndata3_11.

spectively. We have chosen these data sets from the repository for which prior information is available regarding relevance of features and, the relevant features' number in original data set is listed in the fourth column instead of the concrete name of feature for clarity. Although for these benchmark data sets class information is available, in our experiments we have removed the class labels in feature selection. Parity3 + 3 has 3

Table 3
The description of data sets and experimental results (DDR: Dimensionality Reduction Rate, OFS: Original Feature Set, M: Mean, SD: Standard Deviation)

Data Sets	Data	No.	No.	Algorithms	Accuracy Rate%		DRR%
	Set Size	Features	Classes		M	SD	
Ionosphere	351	34	2	Mitra's	84.27	0.30	47.1
				MAIF	84.87	0.30	70.6
				OFS	83.94	0.31	0
Sonar	208	61	2	Mitra's	75.92	0.30	37.7
				MAIF	77.43	0.30	50.1
				OFS	78.62	0.30	0
Multi-features	2000	649	10	Mitra's	93.43	0.20	49.9
				MAIF	92.39	0.22	77.0
				OFS	93.35	0.21	0



(c)

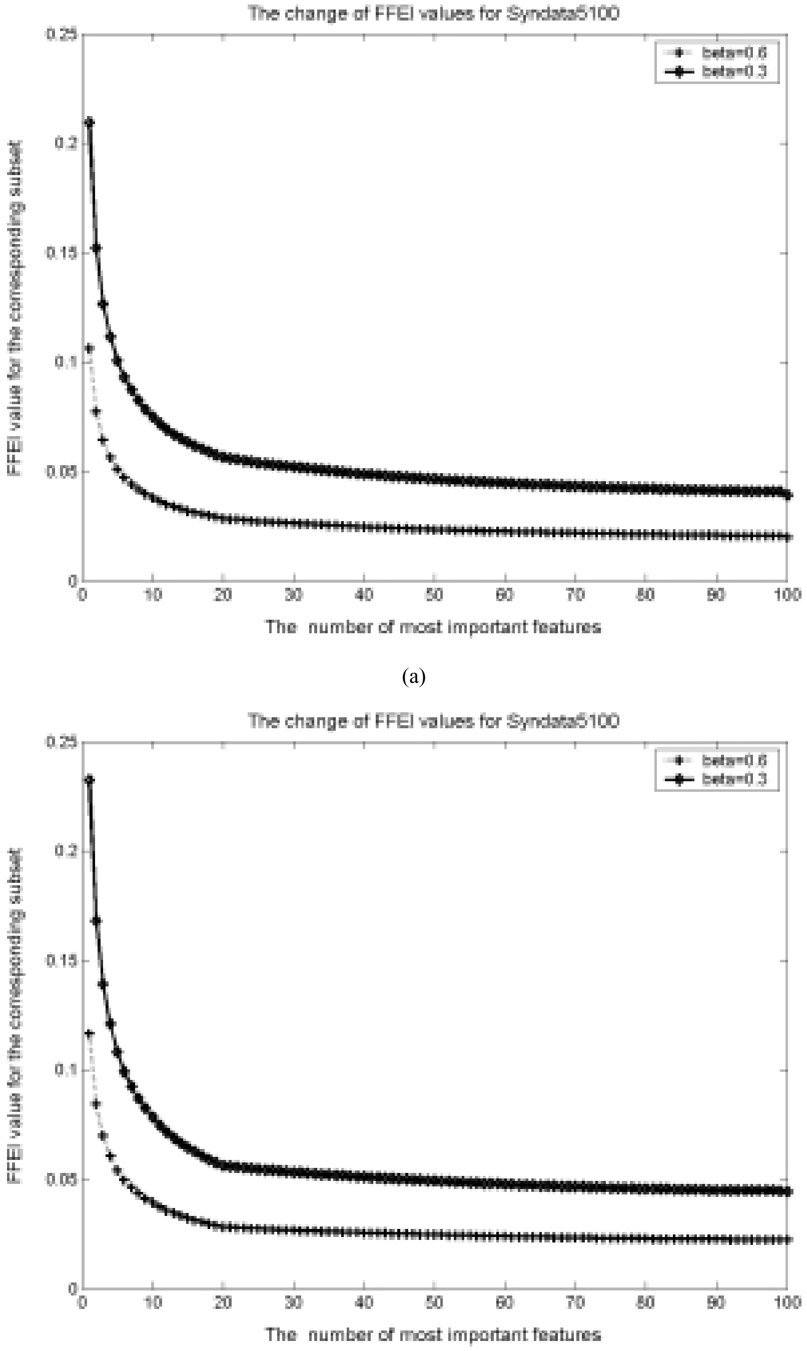
Fig. 2. continued.

relevant, 3 redundant, and 6 irrelevant features.

High dimensional data sets: One synthetic large and high-dimensional data set (Syndata5_100) is generated. The data set has 100 features (first 20 features are relevant and the next 80 features irrelevant), 5 clusters, each cluster created by Gaussian distribution, and irrelevant features take uniformly random values. Each cluster has 20,000 points and the data set has 5000 noisy data points. Three real-world high-dimensional data sets (Ionosphere, Sonar, Multi-features) are selected from [2], which are described in Table 3, the second, third and fourth column (from the left) describes the number of samples, features and classes, respectively. Their class labels are also removed in the feature selection.

4.2. Performance of AIF

The results for ranking low dimensional data sets in descending order are shown in Table 1 in the fifth column (from the left). Here, we also use the feature's number instead of the concrete name. From the Table 1, we can conclude that our method is able to rank the relevant features in the top ranks for all data. For Corral, our method ranks the sixth feature higher. The sixth feature is correlated to the class label 75% of the data



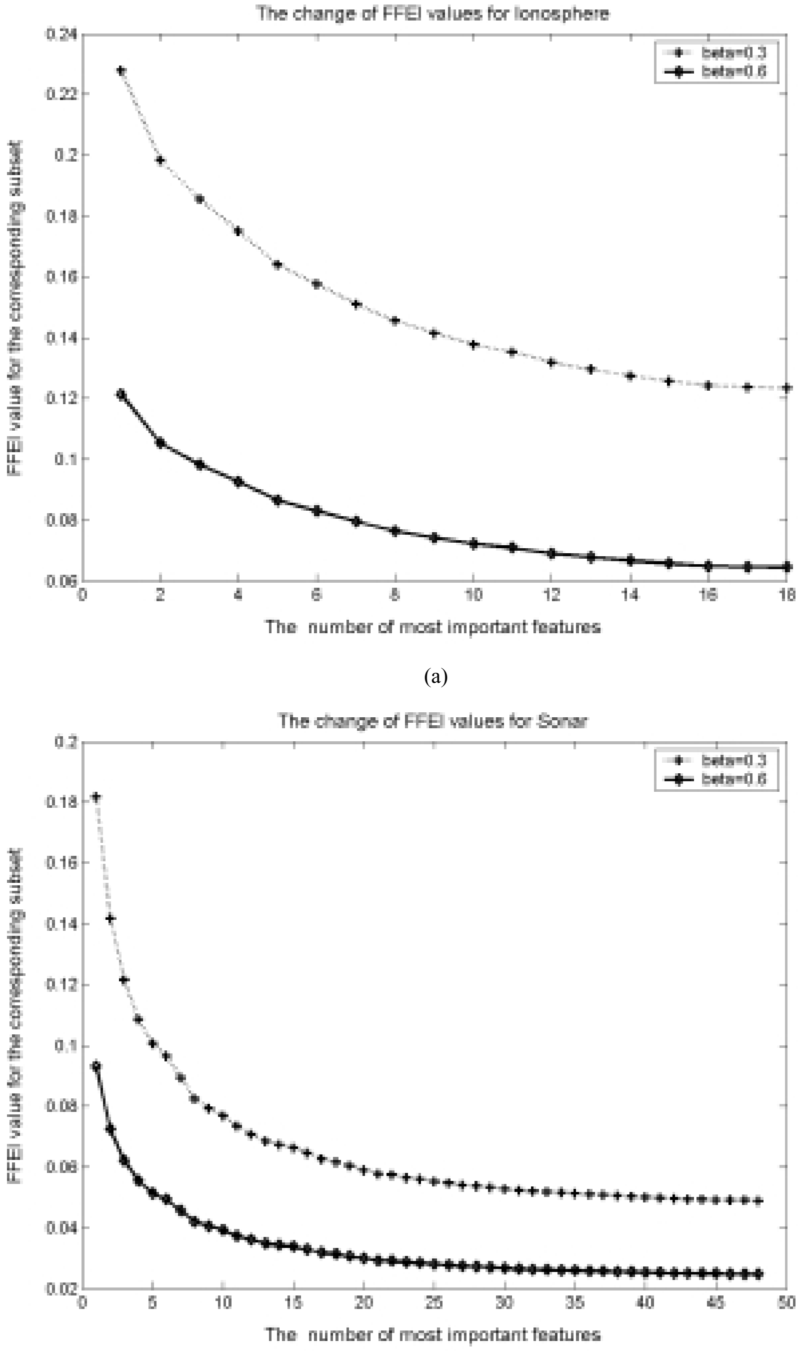
(a)

(b)

Fig. 3. The change of FFEI values for Syndata5_100, the sample size is (a) 0.25%, (b) 1%.

points [18]. This shows our ranking measure favors features that are correlated to the class. Although for Corral this is not desired but for real-world data this may be acceptable. For Parity3 + 3 ranking was correct although the redundant features could not be detected.

The results for ranking of high-dimensional synthetic data set are shown in Table 2. Sample sizes mean the percentage of the training sample for ranking, which is



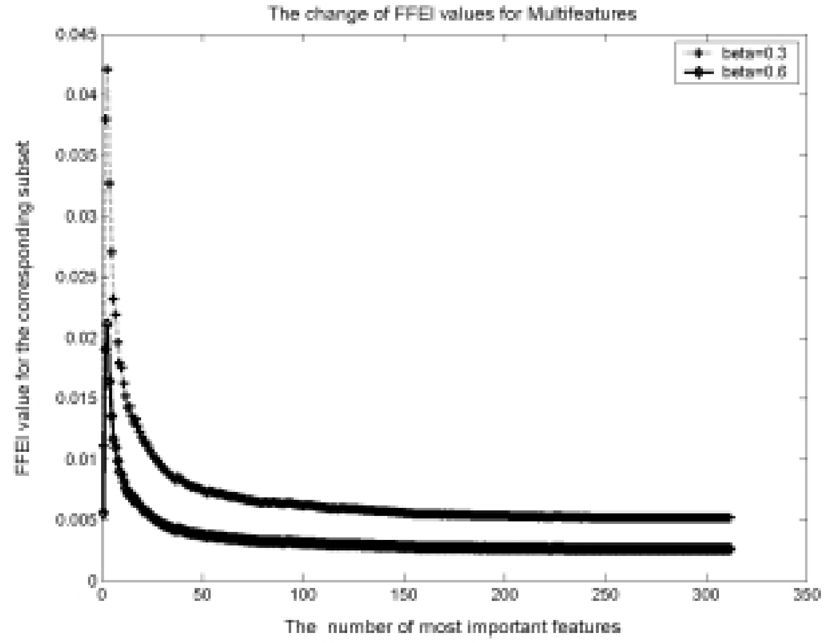
(a)

(b)

Fig. 4. The change of FFEI values for (a) Ionosphere, (b) Sonar, (c) Multi-features.

chosen as 0.25%, 0.50% and 1.0%, respectively. We have shown the ranking results of SRANK for 3 runs in different rows. In all runs the 20 important features are ranked at the top and hence they are ranked at the top in over all ranking as well.

The changes of FFEI value are shown in Fig. 2, which are corresponding to Iris, Monk3, Syndata3_11. For the sake of clarity and briefness, the figures of other data



(c)

Fig. 4. continued.

sets are not listed here, and we only plot two curves with different D values, similar results are obtained in other cases. Figure 3 describes the changes of FFEI value for Syndata5_100 in two runs with different sample size and two curves for different D values. According to the curve of FFEI change, FFEI values decrease with the addition of relevant features in a fast rate but slow down to almost a halt after all the relevant features are added. For a practical application it will not be difficult to notice this trend. So we can determine the selected feature subset by finding the approximate halt point in the curve, and hence selecting a subset of features can be an easy task. The results for selecting a subset of relevant features using AIF are listed in last column of Table 1 and the selection result of Syndata5_100 is shown in the last row of Table 2. For different D values, the plots have similar trend, which also shows that our method is robust against D and is convenient for user.

4.3. Performance of MAIF

We use the MAIF to choose the feature subset for the real-world high-dimensional data sets. We get the k value for Mitra's to obtain the approximate maximum classification accuracy. Since there is not existing a commonly accepted evaluation criterion of clustering performance, we use the classification accuracy of K-NN classifier with the $K = 3$ to evaluate the selected subset of features. Cross-validation is performed in the following manner: we randomly select 10 percent of the data as the training set for Mitra's, then randomly select some data from the remaining data (it is better to keep the structure of cluster) as another training set to rank the output of Mitra's (we suggest the use of at least 35 samples as 35 is often considered the minimum number of samples for large sample procedures [13]) and also select some data as testing set to determine the final feature subset of using FFEI. K-NN classifier will classify the final remaining data. Ten such independent runs are performed and the average classification accuracy is used. The classification accuracy and dimensionality reduction are shown in Table 3 in the sixth and seventh column (from the left), respectively. Figure 4 shows the changes of FFEI for these data sets. It is easy to approximately determine the final feature subset used the method described in Section 3.4. The results listed in Table 3 show our method can get higher dimensionality reduction rate without sacrificing the performance of classification. In some cases, it also can help to improve the classification accuracy. Of course, the time cost of the Mitra's is surely less than that of MAIF because MAIF consists of Mitra's and AIF.

However, note that feature selection can be performed offline in many cases. We should pay more attention to rate of accuracy and dimensionality reduction.

5. Conclusions

We present a two-level filter method for unsupervised feature selection with low time cost, which can remove irrelevant and redundant features and it has been tested on the synthetic and real-world data sets. The Mitra's is selected to remove the redundant features and a new algorithm AIF is presented to eliminate the irrelevant features. The experimental results have shown the better effect for this integration, and also proved the method can handle high dimensional data with noise.

In the AIF, we used random samples to handle large data sets. The method only requires the cluster structure be retained which a reasonably small random sample is expected to maintain. The ranking measure works well consistently for the different runs with different sizes of random samples. For the evaluation index, we consider the Fuzzy Feature Evaluation Index (FFEI) as criterion and propose a simple method to calculate the weight of feature in the FFEI. This breaks down the assumption of hyper-spherical data and has been proved to be very efficient. Our fuzzy evaluation index is robust against the different D values in Eq. (4) and neither domain specialists nor prior knowledge of the problem is required. Therefore, any user can perform the algorithm.

In future, we would like to consider determining the stopping point of the feature selection automatically and reducing the time complexity of the method largely. Using the hyperplane, which is describe in [14], instead of random sampling as the reasonably method to select data subset in SRANK is ongoing work. The hyperplane strategy divides original training set into smaller training sets using a series of hyperplanes and can maintain the structural properties of the smaller data set as those of original data set. The filter method is preferred to select feature subset for high dimensional data, and we will devote ourselves to it. At the same time, the unsupervised feature selection is a challenging issue, which needs much harder work to solve it.

Acknowledgments

We gratefully thank Dr. P Mitra, who is the author of reference [23], for his help. This research was partially supported by the National Natural Science Foundation of China via the grants NSFC 60375022 and NSFC 60473040.

References

[1] A.L. Blum and R.L. Rivest, Training a 3-node neural networks is NP-complete, *Neural Network* **5** (1992), 117–127.

[2] C.J. Merz and P.M. Murphy, UCI repository of machine learning database, http://www.ics.uci.edu/mlearn/MLRepository.html,1996.

[3] D. Koller and M. Sahami, *Towards optimal feature selection*, Proc. of the 13^{th} Int'l Conf. on Machine Learning, 1996, 284–292.

[4] H.C. Mart, A.T. Mario, Figueiredo and A.K. Jain, Simultaneous feature selection and clustering using mixture models, *IEEE Trans. Pattern Analysis and Machine Intelligence* **9**(26) (2004), 1–13.

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

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

[7] I. Kononerko, *Estimating attributes: analysis and extension of RELIEF*, Proc. of European Conf. On Machine Learning, 1994, 171–182.

[8] J. Basak, R.K. De and S.K. Pal, Unsupervised feature selection using a neuro-fuzzy approach, *Pattern Recognition Letters* **19** (1998), 997–1006.

[9] J. Basak, R.K. De and S.K. Pal, Unsupervised neuro-fuzzy feature selection, *Proc. of IEEE Int'l Joint Conf. On Neural Network* **1** (1998), 18–23.

[10] J. Bi, K. Bennett, M. Embrechts, C. Breneman and M. Song, Dimensionality reduction via sparse support vector machines, *Journal of Machine Learning Research* **3** (2003), 1229–1243.

[11] J.G. Dy and C.E. Brodley, *Feature subset selection and order identification for unsupervised learning*, Proc. of seventeenth Int'l Conf. Machine Learning, 2000, 247–254.

[12] J.G. Dy, C.E. Brodley, A. Kak, L.S. Broderick and A.M. Aisen, Unsupervised feature selection applied to content-based retrieval of lung images, *IEEE Trans. Pattern Analysis and Machine Intelligence* **3**(25) (2003), 373–378.

[13] J.L. Devore, *Probability and statistics for engineering and science*, (4th edition), Duxbury Press, 1995, 121–136.

[14] K.A. Wang, H. Zhao and B.L. Lu, Task Decomposition using geometric relation for Min-Max Modular SVMs, *LNCS* **3496** (2005), 887–892.

[15] K. Kira and L. Rendell, *A Practical approach to feature selection*, Proc. of ninth Int'l Workshop Machine Learning, 1992, 249–256.

[16] L. Talavera, Dependency-based feature selection for clustering symbolic data, *Intelligent Data Analysis* **4** (2000), 19–28.

[17] L. Yu and H. Liu, *Feature selection for high-dimensional data: a fast correlation-based filter solution*, Proc. of the 20^{th} Int'l Conf. Machine Learning, 2003, 856–863.

[18] M. Dash and H. Liu, Feature selection for classification, *Intelligent data analysis* **1**(3) (1997), 131–156.

[19] M. Dash and H. Liu, *Feature selection for clustering*, Proc. of Pacific Asia Conf. on Knowledge Discovery and Data Mining, 2000, 110–121.

[20] M. Dash, K. Choi, P. Scheuermann and H. Liu, *Feature selection for clustering-A filter solution*, Proc. of IEEE Int'l Conf. on Data Mining, 2002, 115–122.

[21] M. Devaney and A. Ram, *Efficient feature selection in conceptual clustering*, Proc. of fourteenth Int'l Conf. Machine Learning, 1997, 92–97.

[22] M. Hall, *Correlation-based feature selection for discrete and numeric class machine learning*, Proc. of the 17^{th} Int'l Conf. Machine Learning, 2000, 359–366.

[23] P. Mitra, C.A. Murthy and S.K. Pal, Unsupervised feature selection using feature similarity, *IEEE Trans. Pattern Analysis and Machine Intelligence* **24**(3) (2002), 301–312.

[24] S. Chang, N. Dasgupta and L. Carin. A Bayesian approach to unsupervised feature selection and density estimation using expectation propagation, *Proc. of IEEE Conf. Computer Vision and Pattern Recognition* **7** (2005), 1043–1050.

[25] S.K. Pal, R.K. De and J. Basak, Unsupervised feature evaluation: a neuro-fuzzy approach, *IEEE Trans. Neuro Network* **11**(3) (2000), 366–376.

[26] Y. Kim, W. Street and F. Menczer, *Feature selection in unsupervised learning via evolutionary search*, Proc. of 6^{th} ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, 2000, 365–369.

[27] Y. Li, Z.F. Wu and J.M. Liu, Efficient feature selection for high-dimensional data using two-level filter, *Proc. of the third Int'l Conf. Machine Learning and Cybernetics* **8** (2004), 1711–1716.

Copyright of Journal of Intelligent & Fuzzy Systems is the property of IOS Press and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.