Available online at www.sciencedirect.com

ScienceDirect

Pattern Recognition 41 (2008) 1429–1439

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/pr

Invariant optimal feature selection: A distance discriminant and feature ranking based solution

Jianning Liang[a,*], Su Yang[a,*], Adam Winstanley[b]

[a]Shanghai Key Laboratory of Intelligent Information Processing, Department of Computer Science and Engineering, Fudan University, Shanghai 200433, China
[b]National Centre for Geocomputation, Department of Computer Science, National University of Ireland, Maynooth, Co. Kildare, Ireland

Received 4 February 2007; received in revised form 28 August 2007; accepted 10 October 2007

Abstract

The goal of feature selection is to find the optimal subset consisting of m features chosen from the total n features. One critical problem for many feature selection methods is that an exhaustive search strategy has to be applied to seek the best subset among all the possible $\binom{n}{m}$ feature subsets, which usually results in a considerably high computational complexity. The alternative suboptimal feature selection methods provide more practical solutions in terms of computational complexity but they cannot promise that the finally selected feature subset is globally optimal. We propose a new feature selection algorithm based on a distance discriminant (FSDD), which not only solves the problem of the high computational costs but also overcomes the drawbacks of the suboptimal methods. The proposed method is able to find the optimal feature subset without exhaustive search or Branch and Bound algorithm. The most difficult problem for optimal feature selection, the search problem, is converted into a feature ranking problem following rigorous theoretical proof such that the computational complexity can be greatly reduced. The proposed method is invariant to the linear transformation of data when a diagonal transformation matrix is applied. FSDD was compared with ReliefF and mrmrMID based on mutual information on 8 data sets. The experiment results show that FSDD outperforms the other two methods and is highly efficient.

Keywords: Optimal feature selection; Distance discriminant; Feature ranking

1. Introduction

For pattern recognition applications, feature selection is essential in that it is able to improve the accuracy and efficiency of classification [1–5]. The data can be either labeled or not, leading to the development of supervised and unsupervised feature selection methods. Supervised feature selection determines relevant features by their relations with the corresponding class labels and discards irrelevant and redundant features. Unsupervised feature selection explores data variance and separability to choose relevant features [6–8]. For supervised feature selection, the existing methods fall into two categories: filters and wrappers [9,10]. Wrappers employ a classifier to evaluate feature subsets, based on which they select features, such that the feature selection results are classifier-specific. Filters select the features maximizing some predefined performance indices and are independent of classifiers since no classifier is involved in feature selection. Filters are usually less computationally complex than wrappers. Wrappers often lead to better results in comparison with filters because feature selection is optimized for the particular learning algorithm used. However, wrappers are intractable to large data sets and must be trained again when switching from a classifier to another. In Ref. [11], an approach combining wrappers with filters is proposed to make use of the advantages of both methods.

Provided N samples $Y_1, Y_2, \ldots, Y_N$ belong to c classes and each sample has n features, i.e. $Y_i = [y_i^1, y_i^2, \ldots, y_i^n]$, $1 \leqslant i \leqslant N$, the feature selection problem can be formulated as: choose an optimal or suboptimal subset X consisting of m features of the total n features. Here, the goal of feature selection is to lead

* Corresponding authors. Tel./fax: +86 21 55664412.
E-mail addresses: 052021180@fudan.edu.cn (J. Liang), suyang@fudan.edu.cn (S. Yang).

0031-3203/$30.00 © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2007.10.018

to an as little as possible performance degradation in terms of classification or even a performance improvement for the subsequent classification. Let function $J(X)$ represent the performance index to evaluate any given feature subset X, based on which the feature selection decision is made. Let us assume that a higher value of $J(X)$ indicates a better performance of the feature subset X. If there are total n features, the goal is to select the optimal subset of m $(m \leqslant n)$ features maximizing $J(X)$. One straightforward method is to exhaustively search all $\binom{n}{m}$ combinations of feature subsets. Because the computational complexity increases rapidly with n, the high computational cost makes it impractical to select features in such a manner. So far, the only feature selection method that promises optimal feature subset without exhaustive search is the Branch and Bound algorithm (BB) [12]. However, it is based on such an assumption that the evaluation function is monotonic. Unfortunately, most commonly used evaluation functions do not satisfy this monotonic requirement. Sometimes, BB algorithm is also time-expensive. So, a number of suboptimal feature selection methods [13–16] have been proposed, which provide a tradeoff between the optimization and the computational efficiency. Among those suboptimal feature selection algorithms, one special group is the feature ranking methods, such as ReliefF [17]. Such methods rank each feature according to some criteria and choose the m individually best features. In general, feature ranking is much faster than feature selection.The drawback of such methods is: The m individually best features are not certainly the best combination of m features as a whole.

In this study, we propose a novel feature selection method based on a distance discriminant (FSDD), which belongs to the filter category. The proposed algorithm is an optimal method. It promises the optimal feature selection result like exhaustive search methods. Following a rigorous theoretical proof, in the meantime, it makes use of a feature ranking scheme to approach the globally optimal solution such that the computationally expensive problem of $\binom{n}{m}$ combination can be solved and the computational complexity is far less than that of the exhaustive search methods. The key to convert the feature selection into feature ranking is: a new distance discriminant is proposed in this study, which leads to the theoretical basis to guarantee such an approach. It is argued that since feature selection is done in a off-line manner, the execution time of an algorithm is not as critical as the performance of it. It is true for moderate size samples and features. However, some new applications such as gene expression analysis [18] and document classification involve thousands of features. In such cases, the time cost is crucial. Many methods that work well in low-dimensional spaces cannot be applied to high-dimensional cases due to the computational problem. The computational complexity of FSDD is very low and thus fits well into the high-dimensional problems or on-line feature selection.

This paper is organized as follows: Section 2 presents the theoretical analysis and property of FSDD. In Section 3, we compare the proposed algorithm with ReliefF [17] and mrmrMID [19], both of which are filter methods, and evaluate the 3 methods on 8 data sets from UCI [20] and Statlib [21] with 4 classifiers: KNN, NB (Naive Bayes), DT (Decision Tree), and SVM (Support Vector Machine). The experimental results show that the proposed method outperforms the other two and confirm that FSDD is efficient and effective. Sections 4 and 5 are discussions and conclusions, respectively.

2. Feature selection based on distance discriminant

2.1. The proposed distance discriminant

The basis of the proposed algorithm is to find out the features that promise good class separability among different classes as well as make the samples in the same classes as close as possible. A criterion used for selecting the good features is

$$d_b - \beta d_w, \tag{1}$$

where d_b is the distance between different classes, d_w corresponds to the distance within classes, and β (in the experiments, we set $\beta = 2$) is used to control the impact of d_w. As is proved in the following, Eq. (1) can be transformed into the form

$$d_b - \beta d_w = \sum_{k=1}^{m} \frac{1}{\sigma_k^2}\left[\sigma_k''^2 - \beta \sum_{i=1}^{c} \rho_i \sigma_k'^2(i)\right], \tag{2}$$

where m is the number of selected features, c the number of classes, and ρ_i the prior probability of the ith class.

$$\sigma_k^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i^k - \overline{y_k})^2, \quad \overline{y_k} = \frac{1}{N}\sum_{i=1}^{N} y_i^k$$

are the standard deviation and the mean of all samples in the kth feature, respectively.

$$\sigma_k'^2(i) = \frac{1}{n_i - 1}\sum_{j=1}^{n_i}(y_j^k - \overline{y_j^k})^2, \quad \overline{y_j^k} = \frac{1}{n_i}\sum_{j=1}^{n_i} y_j^k$$

are the standard deviation and the mean of the samples in the ith class (having n_i samples) in the kth feature, respectively.

$$\sigma_k''^2 = \sum_{i=1}^{c} \rho_i (m_i^k - m_k)^2 = \mu_1 - \mu_2^2,$$

$$\mu_1 = \sum_{i=1}^{c} \rho_i (m_i^k)^2, \quad \mu_2 = \sum_{i=1}^{c} \rho_i m_i^k, \quad m_k = \sum_{i=1}^{c} \rho_i m_i^k.$$

Here, $\sigma_k''^2$ is the weighted standard deviation of the class center m_i in the kth feature; m_k is the center of all samples in the kth feature; m_i^k is the center of the samples of the ith class in the kth feature; μ_1, μ_2 are the weighted mean of the squared class center m_i^2 and the class center m_i in the kth feature, respectively.

Before starting to prove Eq. (2), some definitions [22] are recalled as follows:

Definition 1. For two arbitrary points $Y_i = [y_i^1, \ldots, y_i^n]$ and $Y_j = [y_j^1, \ldots, y_j^n]$, the distance between Y_i and Y_j is

$$d(Y_i, Y_j) = \|Y_i - Y_j\| = \sum_{k=1}^{n} \frac{(y_i^k - y_j^k)^2}{\sigma_k^2}.$$

Definition 2. Point-to-set distance

$$d(Y,C)=\frac{1}{m}\sum_{i=1}^{m} d(Y,Y_i),\quad Y_i\in C.$$

Definition 3. Intra-set distance of set C (m points)

$$D(C)=\frac{2}{m(m-1)}\sum_{j=1}^{m}\sum_{i>j}^{m} d(Y_i,Y_j),\quad Y_i,Y_j\in C.$$

Lemma 1. *The intra-set distance of class C is equal to*

$$D(C)=2\sum_{k=1}^{n}\frac{\sigma_k^{'2}}{\sigma_k^2}. \tag{3}$$

Proof.

$$\begin{aligned}
D(C)&=\frac{1}{m}\sum_{j=1}^{m}\frac{1}{m-1}\sum_{i=1}^{m} d(Y_j,Y_i)\\
&=\frac{1}{m(m-1)}\sum_{j=1}^{m}\sum_{i=1}^{m}\sum_{k=1}^{n}\frac{(y_i^k-y_j^k)^2}{\sigma_k^2}\\
&=\frac{m}{m-1}\sum_{k=1}^{n}\frac{1}{m^2\sigma_k^2}\sum_{j=1}^{m}\sum_{i=1}^{m}[(y_i^k)^2-2y_i^k y_j^k+(y_j^k)^2]\\
&=\frac{m}{m-1}\sum_{k=1}^{n}\frac{1}{\sigma_k^2}\left[\frac{1}{m}\sum_{i=1}^{m}(y_i^k)^2\right.\\
&\qquad\left.-\frac{2}{m}\sum_{j=1}^{m}y_j^k\frac{1}{m}\sum_{i}^{m}y_i^k+\frac{1}{m}\sum_{j=1}^{m}(y_j^k)^2\right],
\end{aligned}$$

$$\overline{y_i^k}=\frac{1}{m}\sum_{i=1}^{m}y_i^k,\quad \overline{(y_i^k)^2}=\frac{1}{m}\sum_{i=1}^{m}(y_i^k)^2 \quad \text{and}$$

$$\overline{y_i^k}=\overline{y_j^k},\quad \overline{(y_i^k)^2}=\overline{(y_j^k)^2},$$

$$D(C)=\frac{2m}{m-1}\sum_{k=1}^{n}\frac{1}{\sigma_k^2}[\overline{(y_i^k)^2}-(\overline{y_i^k})^2],$$

$$\begin{aligned}
\sigma_k^{'2}&=\frac{1}{m-1}\sum_{i=1}^{m}(y_i^k-\overline{y_i^k})^2\\
&=\frac{1}{m-1}\sum_{i=1}^{m}[(y_i^k)^2-2y_i^k\overline{y_i^k}+(\overline{y_i^k})^2]\\
&=\frac{m}{m-1}[\overline{(y_i^k)^2}-2(\overline{y_i^k})^2+(\overline{y_i^k})^2]\\
&=\frac{m}{m-1}[\overline{(y_i^k)^2}-(\overline{y_i^k})^2],
\end{aligned}$$

so,

$$D(C)=2\sum_{k=1}^{n}\frac{\sigma_k^{'2}}{\sigma_k^2}. \qquad \square$$

Now, the within-class distance is easy to obtain, that is,

$$d_w=\sum_{i=1}^{c}\rho_i D(C_i)=2\sum_{k=1}^{n}\sum_{i=1}^{c}\rho_i\frac{\sigma_k^{'2}(i)}{\sigma_k^2}. \tag{4}$$

The distance between each pair of classes is

$$d_b=\frac{1}{2}\sum_{i=1}^{c}\rho_i\sum_{j=1}^{c}\rho_j d(m_i,m_j)=\sum_{k=1}^{n}\frac{1}{\sigma_k^2}(\mu_1-\mu_2^2)=\sum_{k=1}^{n}\frac{\sigma_k^{''2}}{\sigma_k^2},$$

$$m_i=\frac{1}{n_i}\sum_{i=1}^{n_i}Y_i,\ Y_i\in C_i, \tag{5}$$

where m_i is the center of the ith class.

Proof.

$$\begin{aligned}
d_b&=\frac{1}{2}\sum_{i=1}^{c}\rho_i\sum_{j=1}^{c}\rho_j\sum_{k=1}^{n}\frac{(m_i^k-m_j^k)^2}{\sigma_k^2}\\
&=\frac{1}{2}\sum_{k=1}^{n}\frac{1}{\sigma_k^2}\left[\sum_{i=1}^{c}\rho_i(m_i^k)^2\sum_{j=1}^{c}\rho_j-2\sum_{i=1}^{c}\rho_i m_i^k\right.\\
&\qquad\left.\times\sum_{j=1}^{c}\rho_j m_j^k+\sum_{i=1}^{c}\rho_i\sum_{j=1}^{c}\rho_j(m_j^k)^2\right],
\end{aligned}$$

$$\sum_{i=1}^{c}\rho_i=1,\quad \sum_{i=1}^{c}\rho_i m_i^k=\sum_{j=1}^{c}\rho_j m_j^k,$$

$$\sum_{i=1}^{c}\rho_i(m_i^k)^2=\sum_{i=1}^{c}\rho_j(m_j^k)^2,$$

$$\begin{aligned}
d_b&=\sum_{k=1}^{n}\frac{1}{\sigma_k^2}\left[\sum_{i=1}^{c}\rho_i(m_i^k)^2-\left(\sum_{i=1}^{c}\rho_i m_i^k\right)^2\right]\\
&=\sum_{k=1}^{n}\frac{1}{\sigma_k^2}(\mu_1-\mu_2^2),
\end{aligned}$$

$$\mu_1=\sum_{i=1}^{c}\rho_i(m_i^k)^2,\quad \mu_2=\sum_{i=1}^{c}\rho_i m_i^k=m_k,$$

$$\begin{aligned}
\sigma_k^{''2}&=\sum_{i=1}^{c}\rho_i(m_i^k-m_k)^2\\
&=\sum_{i=1}^{c}\rho_i(m_i^k)^2-2m_k\sum_{i=1}^{c}\rho_i m_i^k+m_k^2\\
&=\mu_1-\mu_2^2
\end{aligned}$$

so,

$$d_b=\sum_{k=1}^{n}\frac{\sigma_k^{''2}}{\sigma_k^2}.$$

In view of Eqs. (4) and (5), Eq. (2) is proved. □

The proposed criterion in Eq. (1) is similar to that of Ref. [23] applied to feature extraction. In fact, if the Euclidean distance is employed, d_b should be the widely used class separability measure and identical to the counterpart in Ref. [23], In contrast, d_w is different in computing the intra-set distance (see Definition 3) and derived from the book [22]. For different features, the scales of them are usually greatly different. The usual Euclidean distance is highly dependent on the features that have large values. However, the features with small values may also contain useful information in terms of class separability. So, in computing d_w and d_b, a normalized distance measure (Definition 1) is used instead of the Euclidean distance. A parameter β is also introduced to control the effect of d_w in order to select the features that correspond with good class separability but large within-class distances.

2.2. *Optimal feature selection based on feature ranking*

According to Eq. (2), the optimal feature subset can be chosen as follows: First, rank n features in descending order according to the evaluation function $(1/\sigma_k^2)[\sigma_k''^2 - \beta\sum_{i=1}^{c}\rho_i\sigma_k'^2(i)]$. Then, the optimal subset of m features maximizing Eq. (2) is just *the first m features* sorted by the feature ranking. The feature subset is truly identical to the optimal one found by the exhaustive method. As a result, the problem of $\binom{n}{m}$ combination is solved efficiently through the feature ranking.

2.3. *The property of FSDD*

For the convenience of computation, sometimes, some simple preprocessing methods need to be performed prior to feature selection. However, they will affect the subsequent feature selection methods. The proposed algorithm is invariant to linear transformations of features when a diagonal transformation matrix is applied. So, the data preprocessing methods such as z-score (let the mean be 0 and the standard deviation be 1) has no effect on the result of FSDD. The proof is as follows.

For an arbitrary instance Y_i, suppose that its value of the kth feature is y_i^k and the corresponding feature value following a linear transformation is $ay_i^k + b$. According to Eq. (1), in the new space following the linear transformation, we have

$$\underline{d_b} - \beta\underline{d_w} = \sum_{k=1}^{m}\frac{1}{\underline{\sigma_k^2}}\left[\underline{\sigma_k''^2} - \beta\sum_{i=1}^{c}\rho_i\underline{\sigma_k'^2(i)}\right],$$

$$\underline{\sigma_k^2} = \frac{1}{N}\sum_{i=1}^{N}\left[ay_i^k + b - \frac{1}{N}\sum_{i=1}^{N}(ay_i^k + b)\right]^2 = a^2\sigma_k^2,$$

$$\begin{aligned}\underline{\sigma_k''^2} &= \sum_{i=1}^{c}\rho_i(\underline{m_i^k} - \underline{m_k})^2\\ &= \sum_{i=1}^{c}\rho_i\left[\frac{1}{n_i}\sum_{i=1}^{n_i}(ay_i^k + b) - \frac{1}{N}\sum_{i=1}^{N}(ay_i^k + b)\right]^2\\ &= a^2\sigma_k''^2,\end{aligned}$$

$$\underline{\sigma_k'^2(i)} = \frac{1}{n_i - 1}\sum_{i=1}^{n_i}\left[ay_i^k + b - \frac{1}{n_i}\sum_{i=1}^{n_i}(ay_i^k + b)\right]^2 = a^2\sigma_k'^2(i),$$

so,

$$\underline{d_b} - \beta\underline{d_w} = d_b - \beta d_w,$$

where $\underline{d_b}$ has the same meaning as d_b. The difference is just that the former one is the between-class distance following a linear transformation while the latter one is that without any linear transformation. The other underlined notations are defined similarly.

Table 1
Data used in the experiments

Data	#Attributes	#Instances	#Classes	From	Testing method
Mfeat	649	2000	10	UCI	2-Fold CV
Satimage	36	6435	6	UCI	2-Fold CV
Spambase	57	4601	2	UCI	2-Fold CV
Spectrometer	99	509	5	UCI	2-Fold CV
Wine	13	178	3	UCI	10-Fold CV
Analcatdata	70	841	4	Statlib	10-Fold CV
Iris	4	150	3	UCI	10-Fold CV
Vowel	10	990	11	UCI	10-Fold CV

Table 2
Runtime (seconds) of three methods for three data sets

Methods	Data		
	Mfeat	Satimage	Spambase
FSDD	0.375	0.063	0.062
ReliefF	165.13	28.047	31.719
mrmrMID	1824.2	92.015	83.266
#Feature required	50	36	30

3. Experiments

3.1. *Related works*

In the literature, there are many feature selection algorithms. Among these methods, ReliefF [17] is a special one, which is indeed a feature ranking algorithm. It works as follows. First, randomly select several samples from the training samples. Then, find the nearest samples of them from the same class as well as the different classes. Finally, estimate the quality of the attributes according to how well the feature values can distinguish between the instances that are close to each other. In the study, the number of randomly selected samples is 10, which is also the number of the nearest neighbors. ReliefF chooses the m highest-scored features to construct the suboptimal feature subset.

Recently, mutual information based methods [19,24,25] have received much attention. mrmrMID is an effective and efficient one. It requires a discretization preprocessing as follows: Each feature variable is assigned a binary value according to the mean value of this feature. It is set to be 1 if the actual feature value is larger than the mean value of this feature, and −1 otherwise (only for Mfeat data set as Peng [19] did). An alternative preprocessing is to let each feature variable be −1, 1, and 0 if the

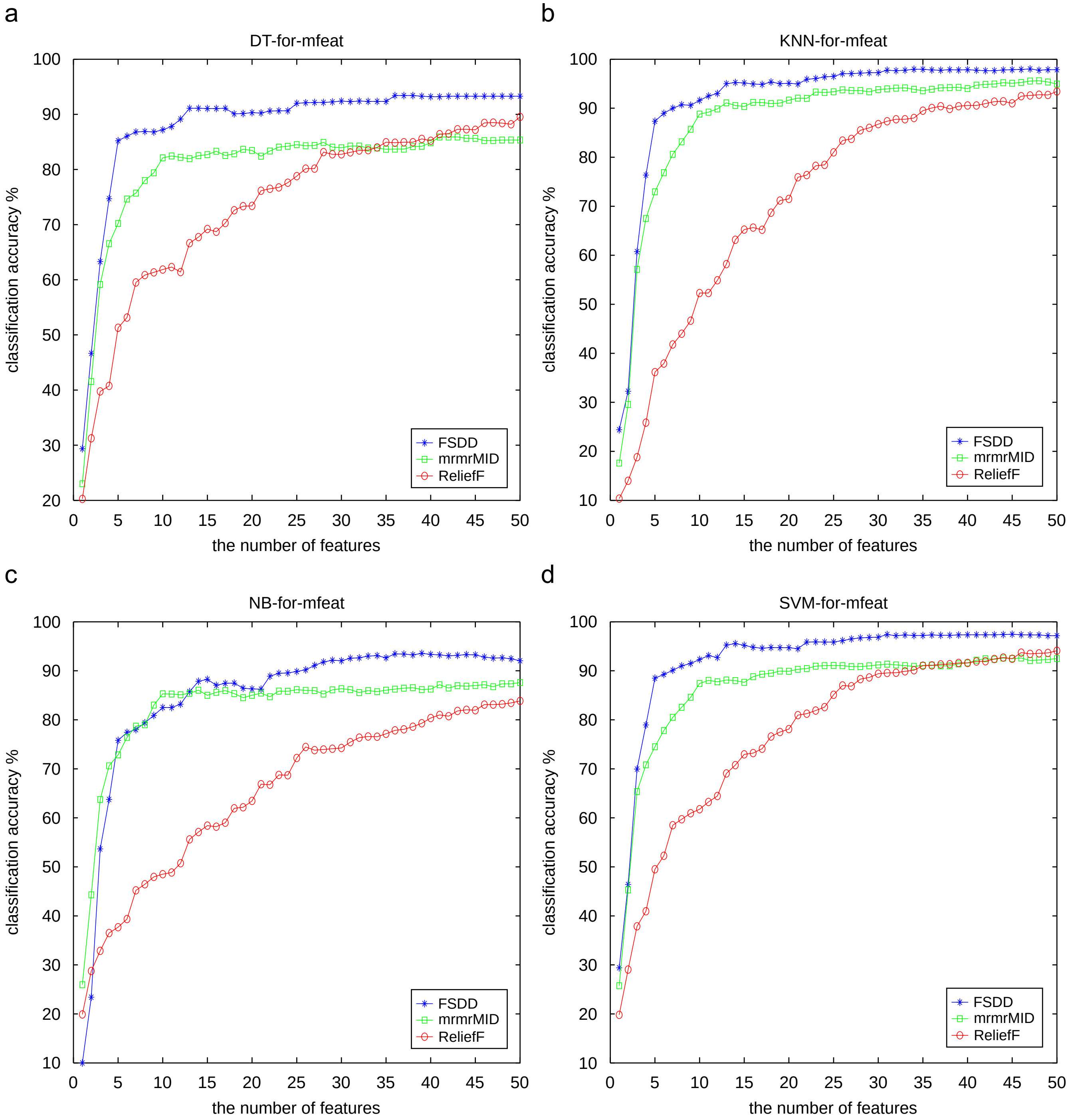

Fig. 1. Two-fold CV classification accuracy for Mfeat with four classifiers.

actual feature value falls within $(-\infty, \mu-\sigma)$, $(\mu+\sigma, +\infty)$, and $[\mu-\sigma, \mu+\sigma]$ respectively (μ is the mean value, σ^2 is the standard deviation). mrmrMID uses an incremental/greedy search to select one feature at each iteration until the predefined number of features is obtained.

3.2. Data sets

Eight data sets are used in the experiments to test the proposed algorithm. Seven of them are from the UCI machine learning databases [20] and the other one is from the Statlib [21]. All the feature values of the above data sets are continuous data. The properties of the data sets are summarized in Table 1 (they differ greatly in the sample size, feature number, data distribution, and class number). In these data sets, the first three are tested by two-fold cross validation (CV) due to the large number of instances and the corresponding high computational costs. Some classes of Spectrometer have less than 10 instances. Thus, it is also tested by two-fold CV. The other four data sets are tested by 10-fold CV.

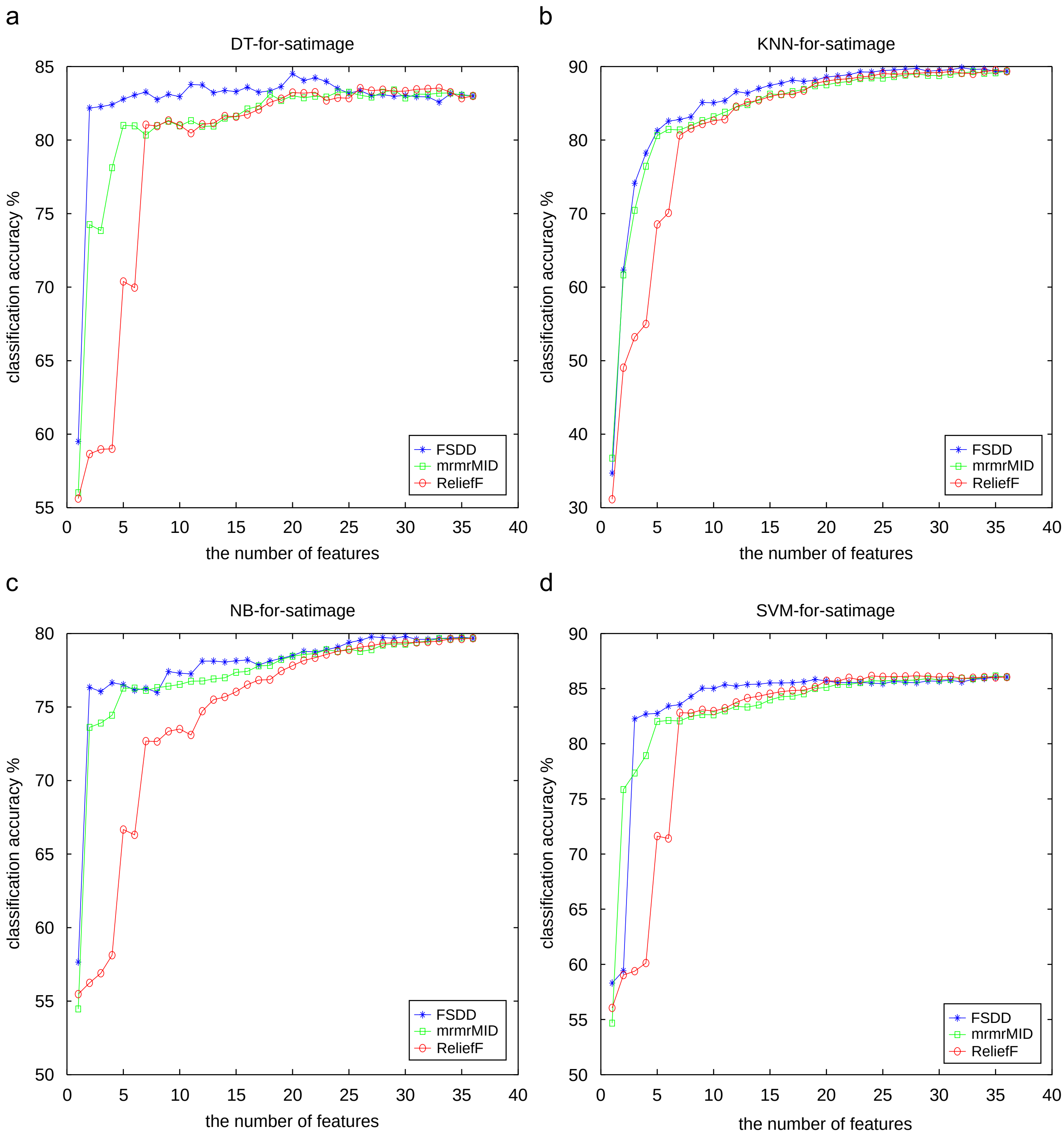

Fig. 2. Two-fold CV classification accuracy for Satimage with four classifiers.

3.3. Classifiers

The proposed algorithm is independent of any classifier. So, four widely used classifiers are considered to estimate the performance of our feature selection method, i.e. KNN ($K = 1$, K Nearest Neighbor), NB [26] (Naive Bayes), DT [26,27] (Decision Tree), and SVM (Support Vector Machine). KNN is an instance-based approach and one of the favorite classifiers due to its effectiveness and efficiency. NB classifier is based on Bayes rule and the assumption that the features are independent of each other given target class and that the conditional probability distribution of any given class satisfies normal distribution. NB classifier has shown good performance compared with some sophisticated classifiers on many real data sets. The implementation of the DT classifier in this study is similar to the C4.5 learning algorithm. We use Libsvm package [28] with default parameters to implement the SVM classifier.

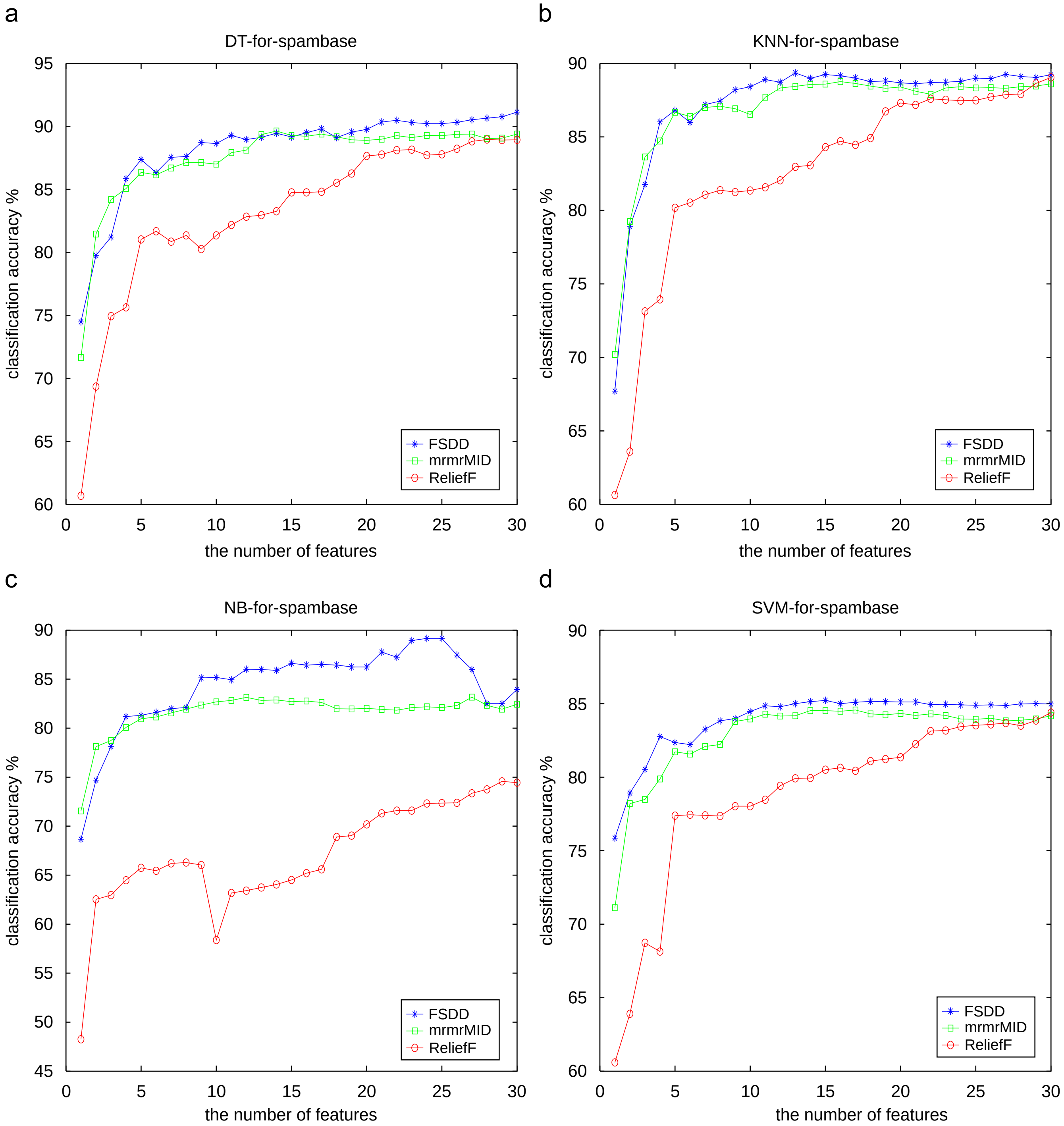

Fig. 3. Two-fold CV classification accuracy for Spambase with four classifiers.

3.4. Computational complexity

Let N denote the total number of the samples, and n the feature number. The computational complexity of the proposed algorithm is roughly $O(N \times n)$. The time cost is very low. The time consumptions (measured by seconds) of the three methods to choose the required number of features for the three data sets, Mfeat, Satimage, and Spambase, are described in Table 2, in which the second to fourth rows are the required time of every method for the data sets and the last row is the number of features selected. The matlab versions of the three methods are implemented on a 1.6 GHz 64 bits AMD CPU. As is shown, FSDD is more efficient than the other two methods.

3.5. Classification accuracy

In Fig. 1, the classification accuracy of four classifiers against the number of features is shown for the Mfeat data with top

Table 3
Ten-fold classification accuracy for Wine

Classifier	mtds	*m*												
		1	2	3	4	5	6	7	8	9	10	11	12	13
KNN	FSDD	70.23	85.39	90.45	92.14	94.94	96.07	96.07	96.07	95.51	96.07	96.07	97.19	95.51
	mrmrMID	38.76	65.17	75.84	75.84	82.58	80.9	85.39	91.01	91.57	94.38	93.82	93.26	95.51
	ReliefF	69.1	82.58	91.57	92.7	93.26	94.94	95.51	95.51	95.51	95.51	94.38	94.94	95.51
NB	FSDD	79.21	88.76	91.57	94.94	94.38	97.19	97.19	96.07	96.63	96.07	96.07	96.07	97.75
	mrmrMID	57.3	73.6	78.09	79.21	84.27	86.52	88.2	94.94	94.38	94.94	95.51	95.51	97.75
	ReliefF	76.97	88.2	91.01	91.57	93.26	94.38	94.38	94.38	94.94	96.63	96.07	97.19	97.75
DT	FSDD	71.91	87.64	92.7	91.57	95.51	95.51	95.51	95.51	95.51	95.51	95.51	94.94	94.94
	mrmrMID	49.44	66.29	73.6	78.65	76.97	78.65	86.52	92.14	92.14	93.82	94.38	93.82	94.94
	ReliefF	70.79	81.46	92.14	92.7	95.51	95.51	95.51	95.51	95.51	95.51	95.51	94.94	94.94
SVM	FSDD	79.21	88.2	92.7	95.51	96.07	97.75	97.19	97.75	97.75	98.32	98.32	97.75	98.32
	mrmrMID	56.18	75.28	82.02	80.9	83.71	85.39	91.57	94.94	95.51	97.19	96.07	97.75	98.32
	ReliefF	75.84	87.64	92.7	92.7	95.51	96.63	97.19	97.19	98.32	98.32	98.32	97.75	98.32

m is the number of features selected. mtds is the abbreviation of the word ‘methods’.

Table 4
Ten-fold classification accuracy for Vowel

Classifier	mtds	*m*									
		1	2	3	4	5	6	7	8	9	10
KNN	FSDD	28.49	62.73	81.31	91.92	94.14	96.16	97.68	98.89	99.09	98.89
	mrmrMID	12.22	27.58	55.35	83.84	93.54	96.87	98.18	98.99	98.49	98.89
	ReliefF	25.35	55.66	80.1	89.8	94.85	97.58	98.79	99.19	98.89	98.89
NB	FSDD	34.95	54.14	58.08	59.8	63.43	62.63	64.14	65.46	67.58	67.58
	mrmrMID	15.35	19.7	27.68	43.43	58.99	62.53	63.54	65.05	65.35	67.58
	ReliefF	32.93	49.6	54.55	58.69	61.92	65.35	65.86	66.87	66.47	67.58
DT	FSDD	31.11	60.4	68.89	74.24	73.94	72.63	75.05	76.97	76.26	75.05
	mrmrMID	11.92	20.51	36.16	58.18	73.43	76.87	78.18	77.07	74.44	75.05
	ReliefF	26.77	53.23	67.68	73.33	74.85	76.87	75.25	75.86	77.88	75.05
SVM	FSDD	35.25	55.46	61.01	65.05	65.76	67.98	69.8	72.12	74.75	74.85
	mrmrMID	16.36	22.02	33.74	48.49	63.84	67.37	68.99	72.02	73.64	74.85
	ReliefF	32.53	50.4	59.29	64.14	65.35	69.8	71.01	72.02	73.03	74.85

Table 5
Ten-fold classification accuracy for Analcatdata

Classifier	mtds	*m*								
		1	5	10	15	20	25	30	35	40
KNN	FSDD	61.12	90.84	96.2	95.96	97.5	98.22	97.86	97.98	98.45
	mrmrMID	47.68	76.69	84.66	85.73	89.3	88.7	89.77	90.96	93.7
	ReliefF	51.37	88.47	96.31	97.27	98.22	98.45	98.34	98.57	99.17
NB	FSDD	64.69	92.87	97.27	97.62	97.86	97.98	97.86	98.22	98.45
	mrmrMID	61.47	82.05	87.87	90.61	91.68	93.1	93.94	95.36	96.67
	ReliefF	56.36	88.82	95.84	97.86	98.45	98.69	99.05	98.81	98.57
DT	FSDD	63.5	88.94	94.53	94.53	94.17	94.17	93.82	94.53	94.41
	mrmrMID	61.59	75.98	76.93	76.81	77.17	77.29	76.22	79.19	83.47
	ReliefF	61	86.8	93.1	94.41	94.41	94.29	94.41	94.77	95.24
SVM	FSDD	64.69	92.75	96.43	98.57	98.93	99.29	99.29	99.29	99.52
	mrmrMID	61.95	82.4	88.82	91.32	92.87	94.06	94.17	96.31	96.91
	ReliefF	58.38	89.54	97.27	98.1	98.93	99.52	99.41	99.52	99.64

Table 6
Two-fold classification accuracy for Spectrometer

Classifier	mtds	*m*								
		1	5	10	15	20	25	30	35	40
KNN	FSDD	68.17	77.21	80.94	85.86	89.59	90.37	89	88.21	88.02
	mrmrMID	50.69	49.71	64.24	69.94	77.01	80.55	83.5	85.86	85.07
	ReliefF	62.28	68.37	83.69	84.87	85.27	87.23	85.27	85.46	86.05
NB	FSDD	75.44	81.14	83.69	83.69	81.73	82.71	84.28	84.09	83.5
	mrmrMID	58.55	54.22	67.39	70.33	74.66	75.44	79.76	82.32	82.91
	ReliefF	69.16	68.37	79.57	79.76	82.52	82.52	82.52	81.93	81.93
DT	FSDD	71.32	78.98	79.76	83.3	83.89	84.09	84.28	83.5	84.09
	mrmrMID	54.81	48.72	67.19	71.91	77.8	81.53	82.71	84.09	81.53
	ReliefF	64.24	67.19	79.57	82.12	82.32	83.89	83.69	83.69	84.09
SVM	FSDD	70.53	78	80.55	83.69	85.46	86.44	87.03	86.84	86.05
	mrmrMID	59.14	54.22	69.35	70.73	79.18	81.93	82.71	86.25	87.03
	ReliefF	69.75	71.51	81.93	84.68	84.48	85.86	87.03	87.03	86.44

Table 7
Ten-fold classification accuracy for Iris

Classifier	mtds	*m*			
		1	2	3	4
KNN	FSDD	88.67	96	95.33	95.33
	mrmrMID	58.67	72	88.67	95.33
	ReliefF	87.33	96	95.33	95.33
NB	FSDD	94	96	95.33	96
	mrmrMID	72.67	78	88	96
	ReliefF	94	96	96	96
DT	FSDD	94.67	96	96	96
	mrmrMID	71.33	68	93.33	96
	ReliefF	95.33	96	96	96
SVM	FSDD	94.67	95.33	95.33	95.33
	mrmrMID	74	79.33	92	95.33
	ReliefF	95.33	95.33	95.33	95.33

50 features selected from all features. It is obvious that FSDD outperform the other methods remarkably. For example, in Fig. 1(a), when the number of features is 10, the classification accuracy of FSDD, mrmrMID, ReliefF is 87.2%, 82.1%, and 61.85%, respectively. FSDD obtains higher accuracy than the other methods with all possible feature numbers for KNN, SVM, and DT. For NB, when the number of features is few, the performance curve of FSDD overlaps with that of mrmrMID. However, FSDD outperforms mrmrMID when the number of features increases. Figs. 2 and 3 are the benchmarks of Satimage and Spambase, respectively. For Satimage, all features are used to compare the three algorithms while just 30 features are chosen for Spambase. In these two data sets, FSDD also outperforms the other two with the four classifiers.The experimental results of the other five data sets are described in Tables 3–7, respectively. Since the number of features is small, we choose all features for Iris, Vowel, and Wine. The top 40 features are selected from the total features for Analcatdata and Spectrometer. For the sake of saving space, only the results of 1, 5, 10, ..., 40 features are listed in the tables. For Wine, FSDD outperforms the other two. For Analcatdata, when the number of features is small, FSDD also outperforms the other two. When the number of features increases, the performance of FSDD is comparable to that of ReliefF and the two methods outperform mrmrMID. It is almost the same case for Vowel. The performance of FSDD is comparable to that of ReliefF and the two outperform mrmrMID for Iris. For Spectrometer, FSDD outperforms the other two methods.

As is shown in the figures and tables, when the number of instances is large (For example, Mfeat, Satimage, and Spambase have thousands of samples), the performance of mrmrMID is better than that in such a case that there are only several hundreds of instances. This is because when 3^n (suppose that there are n features and each feature has three values: -1, 0, 1 after discretization) is comparable to the number of instances, the estimation of mutual information becomes unreliable and mrmrMID prefers to select bad features. ReliefF is a feature ranking algorithm, which suffers from redundant attributes. If the m features are independent, ReliefF promises good performance. However, if the features are relevant to each other, the classification accuracy increases slowly with the number of features. It is known that the m best features are not always the best m features. This is the reason why ReliefF shows worse performance for Mfeat, Spambase, and Satimage than that for the other five data sets.

4. Discussions

The value of β plays an important role in the proposed algorithm and should vary with different classification problems. However, we find that the feature selection order changes very little with various values of β in the experiments. For example, when $\beta = 0.1$, 1, 2, 5, 10, 20, 50, and 100, respectively, the feature selection order for Iris data is the same: 3,4,1,2. The experimental results show that a good performance can be promised with $\beta = 2$.

As is shown in the experiments, FSDD leads to high classification accuracy usually. Sometimes, the curve of the classification accuracy against the number of features is fluctuant. Many factors account for the fluctuation. One cause is that the additional features might be noisy ones, which degrade the classification performance. Another possible reason is that for the use of difference of d_b and d_w, FSDD could not prefer to select the features that correspond with good class separability but large within-class distances. In such a case, a less penalty on d_w would alleviate this problem. The distribution overlapping degrees of the classes are not considered in FSDD. Using the overlapping degree between every two classes instead of a constant β would be a reasonable method, which will be our future work.

For the unbalanced data in which the numbers of samples in each class are greatly different (for example, Spectrometer has five classes and each class has 12, 90, 273, 38, and 96 samples, respectively), FSDD also exhibits good performance because the effect of unbalanced distributions is compensated by the use of the prior probability of each class when calculating d_b and d_w.

FSDD does not provide the criterion to determine the optimal number of features. However, the methods that provide stop criteria do not always reach the highest classification accuracy when the stop criterion is satisfied. We can make a tradeoff between the number of features and the classification accuracy and then determine an appropriate number of features in accordance with the learning tasks.

5. Conclusions

We propose a new feature selection algorithm based on distance discriminant. The goal of the method is to select good features that have good class separability as well as make the instances in the same classes as close as possible. The proposed criterion for selecting features is intuitional and easy to understand. As is proved above, our method can find the optimal feature subset through feature ranking such that it solves the combination problem and overcomes the drawbacks of suboptimal methods. It is also invariant to linear transformations of data when a diagonal transformation matrix is applied. So, the data preprocessing methods such as z-score have no effect on the result of FSDD. The experimental results on 8 data sets with four classifiers confirm that the proposed algorithm is an effective and efficient feature selection method.

Acknowledgment

This work is supported by National Basic Research Program of China under Grant 2006CB705700, Hi-Tech Research and Development Program of China under grant 2006AA010111 and Natural Science Foundation of China under Grant 60305002.

References

[1] A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1) (2000).

[2] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.

[3] M. Dash, H. Liu, Feature selection for classification, Intelligent Data Anal. 1 (1997) 131–156.

[4] R. Gilad-Bachrac, A. Navot, N. Tishby, Margin based feature selection—theory and algorithms, in: Proceedings of the 21st International Conference on Machine Learning, 2004.

[5] K.H. Quah, C. Quek, MCES: a novel Monte Carlo evaluative selection approach for objective feature selections, IEEE Trans. Neural Networks 18 (2) (2007).

[6] J. Dy, C.E. Brodley, Feature selection for unsupervised learning, J. Mach. Learn. Res. 5 (2005) 845–889.

[7] Z. Zhao, H. Liu, Spectral feature selection for supervised and unsupervised learning, in: Proceedings of the 24th International Conference on Machine Learning, ICML2007.

[8] L. Wolf, A. Shashua, Feature selection for unsupervised and supervised inference: the emergence of sparsity in a weight-based approach, J. Mach. Learn. Res. 6 (2005) 1855–1887.

[9] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artif. Intell. 97 (1997) 273–324.

[10] S. Aeberhard, O.Y. De Vel, D.H. Coomans, New fast algorithms for error rate-based stepwise variable selection in discriminant analysis, SIAM J. Sci. Comput. 22 (3) (2000) 1036–1052.

[11] R. Kumar, V.K. Jayaraman, B.D. Kulkarni, An SVM classifier incorporating simultaneous noise reduction and feature selection: illustrative case examples, Pattern Recognition 38 (1) (2005) 41–49.

[12] P. Somol, P. Pudil, J. Kittler, Fast branch & bound algorithms for optimal feature selection, IEEE Trans. Pattern Anal. Mach. Intell. 26 (7) (2004).

[13] A. Jain, D. Zongker, Feature selection: evaluation, application, and small sample performance, IEEE Trans. Pattern Anal. Mach. Intell. 19 (2) (1997).

[14] P. Pudil, F.J. Ferri, J. Novovicova, J. Kittler, Floating search methods for feature selection with nonmonotonic criterion functions, in: Proceedings of the 12th IAPR International Conference on Pattern Recognition, Conference B: Computer Vision & Image Processing, vol. 2, 1994.

[15] M. Kudo, J. Sklanshy, Comparison of algorithm that select features for pattern classifiers, Pattern Recognition 33 (2000) 25–41.

[16] J. Pacheco, S. Casado, L. Nez, O. Gmez, Analysis of new variable selection methods for discriminant analysis, Comput. Stat. Data Anal. 51 (2006) 1463–1478.

[17] M. Robnik-Sikonja, I. Kononenko, Theoretical and empirical analysis of ReliefF and RReliefF, Mach. Learn. 53 (2003) 23–69.

[18] H. Liu, J. Li, L. Wong, A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns, Genome Inf. 13 (2002) 51–60.

[19] H. Peng, F. Long, C. Ding, Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2005) 1226–1238.

[20] Available at ⟨http://www.ics.uci.edu/mlearn/databases/⟩.

[21] Available at ⟨http://lib.stat.cmu.edu/datasets/⟩.

[22] M. Friedman, A. Kandel, Introduction to Pattern Recognition: Statistical, Structural, Neural and Fuzzy Logic Approaches, World Scientific Publishing, Singapore, 1999 pp. 143–147.

[23] H. Li, T. Jiang, K. Zhang, Efficient and robust feature extraction by maximum margin criterion, IEEE Trans. Neural Networks 17 (1) (2006) 157–165.

[24] N. Kawk, C.H. Choi, Input feature selection by mutual information based on Parzen window, IEEE Trans. Pattern Anal. Mach. Intell. 24 (12) (2002) 1667–1671.

[25] T.W.S. Chow, D. Huang, Estimating optimal feature subsets using efficient estimation of high-dimensional mutual information, IEEE Trans. Neural Networks 16 (1) (2005).

[26] I.H. Witten, E. Fank, Data Mining: Practical Machine Learning Tools and Techniques, second ed., Morgan Kaufmann Publishers, Los Altos, CA, 2005.

[27] F. Espostito, D. Malerba, G. Semeraro, A comparative analysis of methods for pruning decision trees, IEEE Trans. Pattern Anal. Mach. Intell. 19 (5) (1997).

[28] Available at ⟨http://www.csie.ntu.edu.tw/cjlin/libsvm⟩.

About the Author—JIANNING LIANG received a Bachelor degree in Microelectronics from Sichuan University, China, in 2003. He is a Doctor degree candidate in the Department of Computer Science and Engineering, Fudan University, China. His current research interests include pattern recognition and machine learning.

About the Author—SU YANG received Doctor and Master degree in Electrical Engineering, and Bachelor degree in Mechanical Engineering from Northwestern Polytechnical University in 1999, 1996, and 1993, respectively. From 2000 to 2001, he was a postdoctoral fellow in the Department of Electronic Science and Engineering at Nanjing University. Since 2002, he joined the Department of Computer Science and Engineering at Fudan University and is currently an associate professor. His main research interests are pattern recognition, machine vision, and signal processing. His recent research works are focused on symbol recognition, feature selection, acoustic signal classification, and nonlinear signal classification.

About the Author—ADAM WINSTANLEY received his B.A. degree in archaeology, with honours, from the University of Cambridge, UK, in 1978. He worked as an archaeologist and cartographer for several years in Northern Ireland. He earned an M.Sc. degree in computer science and applications at Queen's University, Belfast, in 1987. He also did graduate work on program specification and animation at the Queen's University, and completed his Ph.D. in the Department of Computer Science from 1987 to 1991. From 1991 to 1993, he was a research fellow on process-oriented specifications at the University of Ulster, before returning to Queen's University as a lecturer. In 1995 he became a lecturer in computer science at National University of Ireland Maynooth, where he is currently a senior lecturer and Head of Computer Science. His current research interests include shape recognition and classification, graphics recognition, spatial information science, geo-computation and control systems for electric vehicles. He is a member of the IEEE and ACM.