



Feature evaluation and selection based on neighborhood soft margin

Qinghua Hu^{a,b,*}, Xunjian Che^a, Lei Zhang^b, Daren Yu^a

^a Harbin Institute of Technology, Harbin, China

^b Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China

ARTICLE INFO

Article history:

Received 13 June 2009

Received in revised form

11 February 2010

Accepted 17 February 2010

Communicated by J. Yang

Available online 6 March 2010

Keywords:

Feature selection

Feature evaluation

Margin

Neighborhood

ABSTRACT

Feature selection is considered to be an important preprocessing step in machine learning and pattern recognition, and feature evaluation is the key issue for constructing a feature selection algorithm. In this work, we propose a new concept of neighborhood margin and neighborhood soft margin to measure the minimal distance between different classes. We use the criterion of neighborhood soft margin to evaluate the quality of candidate features and construct a forward greedy algorithm for feature selection. We conduct this technique on eight classification learning tasks and some cancer recognition tasks. Compared with the raw data and other feature selection algorithms, the proposed technique is effective in most of the cases.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Feature selection plays an important role in a number of machine learning and pattern recognition tasks [1–3]. A lot of candidate features are usually provided to a learning algorithm for producing a complete characterization of the classification task. However, it is often the case that majority of the candidate features are irrelevant or redundant to the learning task, which will deteriorate the performance of the employed learning algorithm and lead to the problem of overfitting. The learning accuracy and training speed may be significantly deteriorated by these superfluous features [4]. So it is of fundamental importance to select the relevant and necessary features in the preprocessing step.

Some techniques for feature selection have been developed in the last decade [5–7]. The key issue in constructing feature selection algorithms is to evaluate the quality of candidate features [8,9]. An optimal criterion should naturally relate the Bayes error rate of classification. However, computing Bayes error rates requires detailed knowledge of the probability distribution of the task, whereas in practice these probabilities are unknown. Quite commonly, we focus on the design of performance measures to determine the relevance between features and decision. Distance, correlation, mutual information, consistency and dependency are usually considered as feasible alternatives. Mutual information is widely discussed in characterizing

relevance between categorical attributes and classification [10]. Wang introduced an axiomatic framework for feature selection based on mutual information [1]. A dependency-based feature selection algorithm was proposed, where dependency is defined as the ratio of so-called positive region in the rough set theory over the whole set of samples [11]. Positive region is the subset of samples consistently classified into one of the decision classes if their feature values are the same. However, the rate of positive region is not an effective estimate of classification accuracy. According to the Bayes rule, the samples with the same feature values will be classified as belonging to the majority class. Therefore, only the samples in the minority classes are misclassified in this case. Based on this observation, Dash and Liu introduced the measure of consistency and employed it to evaluate the quality of features where consistency is treated as the ratio of the samples which can be recognized with the Bayes rule [12].

From another viewpoint, classification margin was introduced to evaluate features in recent years. Class margin is used to characterize the confidence of classification in statistical learning theory. It was observed that a classifier producing a great class margin will get good generalization ability. In 1998, Bartlett and Shawe-Taylor [23] showed that the fat shattering dimension fat_F of the linear function set F is less than $\min\{R^2/\gamma^2, n+1\}$, where the samples are γ fat shattered and in a ball of n dimensions of radius R about the origin. This conclusion shows the connection between the margin of a classifier and its generalization ability.

A number of algorithms based on margin have been proposed for evaluating features. In 2002, Crammer et al. gave two ways to define the margin of a sample: sample margin and hypothesis

* Corresponding author at: Harbin Institute of Technology, Harbin, China.
E-mail address: huqinghua@hit.edu.cn (Q. Hu).

margin [13]. In 2004, Gilad-Bachrach et al. introduced hypothesis margin to evaluate features and developed two algorithms, called Simba and G-flip [20]. In 2006, Sun and Li [14] showed that the famous feature evaluating algorithm Relief and its variant ReliefF [15] could also be considered as a margin based feature estimator. And then an iterative version of Relief (I-Relief) was introduced. In 2009, Li and Lu introduced a distance learning scheme based on loss-margin of nearest neighbor classification for ranking features [16]. In 2002, Guyon et al. introduced a well known gene selection algorithm based on support vector machines (SVM) [22]. In fact this algorithm can also be considered as a margin based algorithm, where the quality of a feature is measured with the weight of the feature in the SVM classifier and the weight reflects the contribution of the corresponding feature to the classification margin.

In SVM, the margin is defined as the minimal distance between samples and classification hyperplane. This margin is sensitive to noisy samples. Provided a noisy learning task, the theory about soft margin shows that there should be a tradeoff between margin and training error rate, where the training error rate reflects empirical risk, while margin is the measure of confidence of classification. Empirical risk would rise if we enlarge the classification margin. To minimize the expected risk, tradeoff between empirical risk and margin is required. In this viewpoint, we require to compute the training error and classification margin in measuring the quality of classification. As we know, the existing techniques, such as Simba and ReliefF, just reflect the average margin of samples, and do not directly take the classification error into account.

In 2008, Hu, Yu et al. [9] introduced a neighborhood rough set model to measure the classification power of numerical attributes, where dependency approximately reflects the training accuracy with a given neighborhood size. One naturally expects the derived features can get an optimal subset of features which produce high classification accuracy and a large classification margin. However, it is usually the case that we do not know how to set the size of neighborhood. Given a size of neighborhood δ , although we may get a subset of features such that the classification task in the selected subspace is consistent. That is to say, the δ neighborhood of each sample has the same decision as this sample. We can say that the neighborhood margin of the samples is at least δ . We call the classification task is δ neighborhood classifiable. It is notable that a task may be neighborhood classifiable with the size greater than δ if it is δ neighborhood classifiable. Furthermore, neighborhood separability is also sensitive to noisy samples. Here, we will develop a technique to overcome these problems.

In this work, we introduce the idea hidden in soft-margin support vector machines into neighborhood rough sets and propose a neighborhood soft-margin based feature evaluating and selecting technique. This criterion integrates the classification loss (characterized with neighborhood boundary) and neighborhood margin (characterized with the size of neighborhood) to reflect the classification quality in feature subspaces.

The rest of the paper is organized as follows. The basic knowledge about neighborhood rough sets is given in Section 2. The concepts of neighborhood margin and neighborhood soft margin are introduced in Sections 3 and 4. Experimental analysis is presented in Section 5. Finally, the conclusion comes in Section 6.

2. Preliminaries on neighborhood rough sets

Here we give some basic definitions and notations used in the following sections.

Definition 1. Given a set \mathbb{S} of samples described with features F , Δ is a distance function on \mathbb{S} and δ is a positive constant. Then the neighborhood of sample x is defined as $\delta(x) = \{x_i | \Delta(x, x_i) \leq \delta\}$.

The neighborhood of x is a subset of samples which are close to x . We expect the neighborhood of x should be grouped into the same decision class as they take the similar feature values.

The relation \mathcal{N} of neighborhood divides the samples into a collection of subsets $\{\delta(x_i)\}_{i=1}^n$ of samples, where $\mathcal{N}(x, y) = 1$ if $y \in \delta(x)$; otherwise, $\mathcal{N}(x, y) = 0$. We call $(\mathbb{S}, \mathcal{N})$ a neighborhood approximation space.

Definition 2. Given $(\mathbb{S}, \mathcal{N})$ and an arbitrary subset $X \subseteq \mathbb{S}$ of samples, the lower approximation and upper approximation of X in $(\mathbb{S}, \mathcal{N})$ are defined as

$$\underline{\mathcal{N}}X = \{x \in U | \delta(x) \subseteq X\}, \quad \overline{\mathcal{N}}X = \{x \in U | \delta(x) \cap X \neq \emptyset\}$$

Definition 3. Given $(\mathbb{S}, \mathcal{N})$, \mathbb{S} is partitioned into m decision classes d_1, d_2, \dots, d_m with the decision attribute Y . Then the lower and upper approximations of classification in $(\mathbb{S}, \mathcal{N})$ are defined as

$$\underline{\mathcal{N}}Y = \bigcup_{i=1}^m \underline{\mathcal{N}}d_i; \quad \overline{\mathcal{N}}Y = \bigcup_{i=1}^m \overline{\mathcal{N}}d_i,$$

correspondingly, the approximation boundary of classification is defined as

$$BN_{\mathcal{N}}Y = \overline{\mathcal{N}}Y - \underline{\mathcal{N}}Y$$

It is easy to show that $\overline{\mathcal{N}}Y = \mathbb{S}$. So $BN_{\mathcal{N}}Y = \mathbb{S} - \underline{\mathcal{N}}Y$.

The decision boundary is the subset of samples which have some samples with different classes in their neighborhoods. So, these samples are easy to be misclassified. In some literatures, classification boundary is considered as one of the main sources of classification complexity [6,8,9].

Definition 4. Given $(\mathbb{S}, \mathcal{N})$ and the decision attribute Y , the neighborhood dependency of Y on the set of features F is computed with

$$\gamma_{\mathcal{N}}(Y) = \frac{|\underline{\mathcal{N}}Y|}{|\mathbb{S}|},$$

where $|A|$ is the cardinality of A .

We say decision Y completely depends on features F if $\gamma = 1$. We say the classification task is δ neighborhood consistent or δ neighborhood separable if $\gamma_{\mathcal{N}}(Y) = 1$; otherwise, we say decision Y depends on features F with level γ .

$\gamma_{\mathcal{N}}(Y)$ approximately reflects the classification accuracy. It was used to evaluate quality of features [9,24]. However, this measure cannot reflect the margin with respect to the corresponding $\gamma_{\mathcal{N}}(Y)$. Given two subsets of features, if their values of γ are the same, but the sizes of neighborhood used in computing γ are different. it means that the feature subset producing greater dependency is better than the other one. We prefer to the subset which γ is computed with the greater δ . This difference cannot be reflected with neighborhood dependency.

3. Neighborhood margin

The above section gives the basic definition of neighborhood dependency. We also point out its disadvantage that it cannot reflect the size of margin. Now we introduce some new definitions.

Definition 5. Given two sets $A = \{a_1, a_2, \dots, a_m\}$ and $B = \{b_1, b_2, \dots, b_n\}$, the distance between A and B is computed as

$$\Delta(A, B) = \min_{\substack{a \in A \\ b \in B}} \Delta(a, b) \tag{1}$$

Definition 6. Given a classification task, the samples are divided into m classes $D = \{d_1, d_2, \dots, d_m\}$. We say that the neighborhood margin of the task is δ if

$$\min_{d_i, d_j \in D} \Delta(d_i, d_j) = \delta \tag{2}$$

It is easy to see that in essence neighborhood margin is the least inter-class distance according to the above definition.

Now we discuss the connection of neighborhood margin with the margin defined in support vector machines.

Assume that we are given a set of sample points of the form $\mathbb{S} = \{(x_i, y_i) | x_i \in \mathbb{R}^N, y_i \in \{-1, +1\}\}_{i=1}^n$, where y_i is either d_1 or d_2 , denoted by $+1$ and -1 . Each sample is described with a N -dimensional vector. a maximal margin hyperplane gotten with SVM learning algorithms is written as

$$y_i(w^T x_i - b) \geq 1, \quad i = 1, 2, \dots, n. \tag{3}$$

By using geometry, we know the distance between samples and the hyperplane is $2/\|w\|$, so we can maximize margin by minimizing $\|w\|$. This can be transformed to a quadratic programming optimization problem.

As to a linearly separable task, a large margin classifier can be illustrated as Fig. 1. The real margin is $2/\|w\|$ in this case.

Definition 7. If the distance between the hyperplane and a sample is less than δ , we say that the two classes of samples is δ linearly inseparable; otherwise, the task is δ linearly separable [18].

Obviously, as to Fig. 1, the task is δ linearly inseparable if $\delta > 1/\|w\|$.

The classification margin is shown to be effective for evaluating features. However, in filter based feature selection, we do not know the classifying hyperplane. Thus we cannot compute the margins of samples in this case.

Now, we consider samples close to the classification hyperplane. It is easy to see that these samples are close to samples with different labels.

Given a set of samples \mathbb{S} for classification learning, assume \mathbb{S} is divided into two classes $\{d_1, d_2\}$ based on the decision attribute. If the samples are δ neighborhood separable, we have that $\Delta(d_1, d_2) > \delta$, where $\forall x_i \in d_1, \forall x_j \in d_2, \Delta(d_1, d_2) = \min \Delta(x_i, x_j)$. In this case, we say the neighborhood margin of the classification task is no less than δ .

Neighborhood margin can be directly computed from samples. So we can use it to evaluate features.

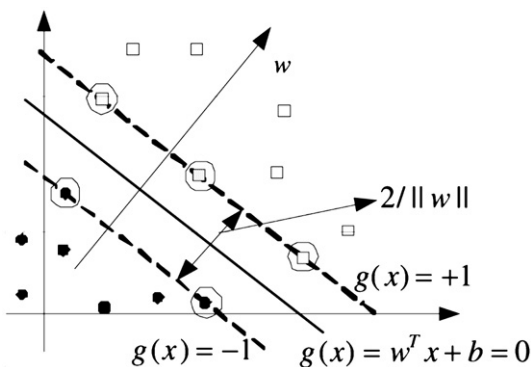


Fig. 1. Linear support vector machine.

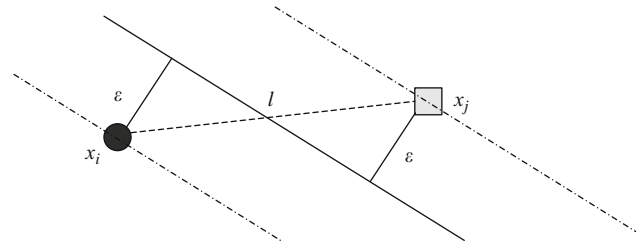


Fig. 2. Relation between δ neighborhood separable and ϵ linearly separable.

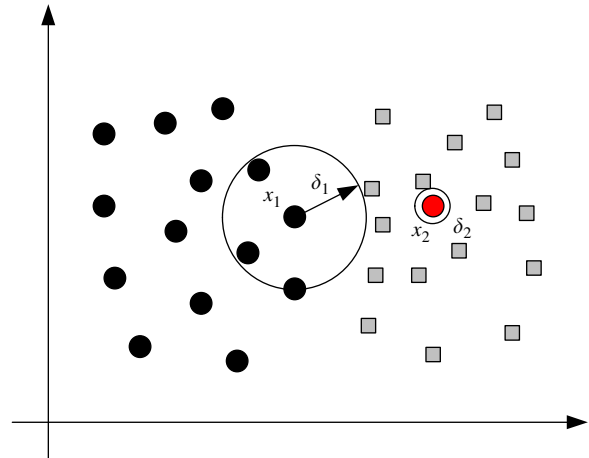


Fig. 3. Classification task with noisy sample.

It is easy to get from Fig. 2 that the task is 2ϵ neighborhood separable if it is ϵ linearly separable with respect to linear SVM because the least distance l between two samples with different classes is not less than 2ϵ .

4. Neighborhood soft margin

One expects the classification task is separable at a large size of neighborhood. In this case, the margin between samples in different classes is large enough for discriminating them. So we can construct a new criterion for evaluating the quality of feature spaces based on neighborhood. However, just as pointed out above that we are usually confronted with tasks which are of little margins due to noisy samples, instead of complexity of tasks.

Fig. 3 shows a task with only one noisy sample, where samples in class 1 are denoted by \bullet and samples in class 2 are marked with \square . We see that x_1 is the closest one to the second class if we do not consider x_2 . In this case the task is δ_1 neighborhood separable. However, if there are some samples like x_2 , the margins between these classes of samples are significantly reduced. The task is δ_2 neighborhood separable if x_2 is considered. As δ_2 is far less than δ_1 , then the quality of the feature subspace gets much worse than the case that x_2 is not considered. The analysis shows that neighborhood margin based feature evaluation is sensitive to noisy samples.

In order to deal with this problem, Cortes and Vapnik suggested a modified maximum margin idea that allows for mislabeled examples in 1995 [17]. If there exists no hyperplane that can split two class of examples, a soft margin method will choose a hyperplane that splits the examples as cleanly as possible, while still maximizing the distance to the nearest cleanly split examples. The method introduces slack variables ζ_i ,

which measures the degree of misclassification of the datum x_i .

$$y_i(w \cdot x_i - b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, n \quad (4)$$

The objective function is then increased by a function which penalizes non-zero ξ_i , and the optimization becomes a trade off between a large margin, and a small error penalty. If the penalty function is linear, the optimization problem can be rewritten as

$$\frac{1}{2} \|w\|^2 + C \sum_i \xi_i, \quad \text{subject to } y_i(w \cdot x_i - b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, n \quad (5)$$

where C is a constant for cost of the constrain violation. This constraint along with the objective of minimizing $\|w\|$ can be solved using Lagrange multipliers.

In order to reduce the influence of noisy samples on neighborhood margin, we introduce the idea of soft-margin support vector machines by trading off between the size of neighborhood and the size of boundary set.

Given a set of samples S for classification learning, and two positive constants δ_1 and δ_2 . B_1 and B_2 are the boundary sets computed with δ_1 and δ_2 , respectively. We have the following conclusion: $B_1 \subseteq B_2$ if $\delta_1 < \delta_2$.

This property shows that the size of boundary set monotonically increases with the size of neighborhood. As we know, the boundary set is easy to be misclassified as they are close to samples from other classes.

On one hand, we expect there is a large neighborhood such that the task is separable. On the other hand, we also desire that the boundary set with respect to the large neighborhood is as small as possible. Based on this idea, we can design a new criterion by combining the sizes of neighborhood and the boundary set.

Following the optimization objective function in soft-margin support vector machine, we give a new measure for evaluating features:

$$NSM = \min_{\delta} \frac{1}{2\delta^2} + \lambda |BN_{\delta} Y| \quad (6)$$

where δ is the size of neighborhood, $|BN_{\delta} Y|$ is the size of the boundary set with respect to δ , and λ is a nonnegative real number to reflect the weight of margin and size of the boundary set. We call this criterion neighborhood soft-margin.

In fact, the size of boundary samples is also sensitive to noisy samples. As shown in Fig. 4, if deleting x_2 , the samples around x_2 belong to the positive region. However, if we consider x_2 , these samples should be divided into the boundary set. As we know, not all these boundary samples are misclassified. According to the class distribution, only samples x_2 are misclassified. We should just compute those misclassified samples in boundary regions. This problem can be overcome with the following definitions.

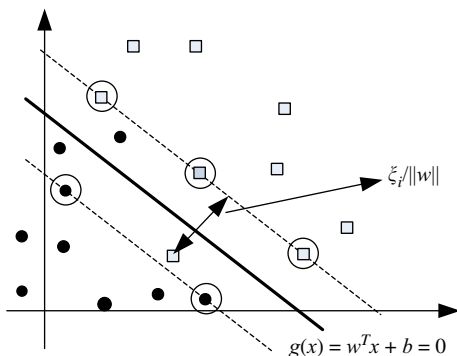


Fig. 4. Soft margin support vector machine.

Definition 8. Given a set of training samples, $\delta(x_i)$ is the neighborhood of sample x_i , $P(\delta(x_i)|\omega_j)$, $j=1,2,\dots,m$, is the class probability of class ω_j , then $ND(x_i) = \omega_l$ if $P(\omega_l|\delta(x_i)) = \max_j P(\omega_j|\delta(x_i))$, where $P(\omega_j|\delta(x_i)) = n_j/N$, N is the number of samples in the neighborhood, n_j is the number of samples in this neighborhood and they belong to decision ω_j .

We introduce 0–1 loss function for misclassified samples

$$\lambda(D(x_i)|ND(x_i)) = \begin{cases} 0 & D(x_i) = ND(x_i) \\ 1 & D(x_i) \neq ND(x_i), \end{cases}$$

where $D(x_i)$ is the real class of x_i .

Definition 9. Neighborhood decision error number (NDEN) is defined as

$$NDEN = \sum_{i=1}^n \lambda(D(x_i)|ND(x_i)).$$

In practice, we replace the size of boundary set with NDEN. Given a set of samples and parameter δ , we can compute the NDEN. Then we get the quality of features. As to a certain subset of features, we can try a series of values for parameter δ , compute the neighborhood soft margins and get the maximal value of neighborhood soft-margin as the final output. Assume $\delta = 0.01$, $\lambda = 1$, and $NDEN = 0$, then $NSM = 5000$. However, in some cases, we change $\delta = 0.1$, we get $NDEN = 20$, then $NSM = 250$. Thus, we should take 0.1 as the soft margin of the classification task although there are 20 samples in the margin. In the experiments, we try δ from 0.01 to $0.25 \times \sqrt{N}$ with step 0.01, where N is the number of features.

It is notable that good subsets of features obtain small neighborhood soft margin according to the above definition. We construct a greedy algorithm to select features based on the proposed measure. We first compute the neighborhood soft margin of each feature and select the feature f_1 with the largest margin. Then we find a feature f_2 to get the largest margin in the subspace of f_1 and f_2 , and so on until all the candidate features are selected. We get a rank of features by this procedure. Then we can check the effectiveness of the first m features, where m is specified by the users or the cross validation technique.

5. Experimental analysis

The proposed technique uses the soft-margin computed with neighborhood to evaluate features. In order to show the disadvantage of neighborhood dependency and neighborhood margin, we conduct some numerical experiments on data set wine. Data wine is a well known linearly separable task. There are 178 samples characterized with 13 numerical features in this data set. The samples are divided into three classes. We normalized each feature into the unit interval $[0,1]$. First we use a randomized feature selection algorithm based on neighborhood rough sets to get 20 subsets of features, as shown in Table 1. In the experiment, we set the size of neighborhood 0.15. That is to say, if the minimal margin between different classes is great than 0.15, the neighborhood dependency is 1 and the task is 0.15 neighborhood separable. In Table 1, the second column shows the selected features with the order that the features are selected in the subsets and the third column gives the variation of neighborhood dependency with the increase of features. Given any of these feature subsets, we can see that the classification task is neighborhood separable. The neighborhood dependency gets its maximal value 1. However, we find that the classification performances of these feature subsets are different. The last column presents the classification accuracies

Table 1
feature subsets, neighborhood dependency, margin, neighborhood soft margin (NSM) and classification accuracies (wine).

ID	Features	Dependency	Margin	NSM	KNN
1	1, 11, 13, 2, 12, 4, 7	0.039, 0.303, 0.685, 0.893, 0.978, 0.989, 1	0.212	23.4603	0.97 ± 0.03
2	13, 11, 12, 9, 4, 1	0.112, 0.354, 0.685, 0.905, 0.989, 1	0.199	38.083	0.95 ± 0.06
3	1, 11, 13, 2, 5, 7	0.039, 0.303, 0.685, 0.893, 0.972, 1	0.233	29.370	0.99 ± 0.02
4	13, 11, 12, 9, 3, 1	0.112, 0.354, 0.685, 0.905, 0.978, 1	0.197	42.083	0.95 ± 0.07
5	1, 12, 10, 4, 2, 7	0.039, 0.365, 0.708, 0.905, 0.989, 1	0.188	43.521	0.92 ± 0.07
6	10, 12, 2, 13, 3, 5, 4	0.034, 0.264, 0.584, 0.888, 0.966, 0.989, 1	0.180	38.446	0.94 ± 0.06
7	13, 10, 12, 7, 4, 1	0.112, 0.449, 0.736, 0.899, 0.966, 1	0.166	35.253	0.95 ± 0.04
8	1, 12, 10, 5, 6, 4	0.039, 0.365, 0.708, 0.910, 0.977, 1	0.175	45.823	0.93 ± 0.05
9	13, 12, 2, 4, 5, 8	0.112, 0.326, 0.691, 0.905, 0.978, 1	0.177	57.482	0.90 ± 0.06
10	13, 10, 9, 6, 2, 8	0.112, 0.449, 0.685, 0.899, 0.966, 1	0.159	55.253	0.94 ± 0.05
11	13, 11, 7, 10, 5, 4	0.112, 0.354, 0.736, 0.887, 0.989, 1	0.163	33.370	0.96 ± 0.05
12	3, 13, 12, 10, 11, 9	0.028, 0.202, 0.573, 0.882, 0.966, 1	0.177	33.942	0.94 ± 0.05
13	13, 11, 7, 5, 10, 3	0.112, 0.354, 0.736, 0.933, 0.989, 1	0.150	29.823	0.96 ± 0.05
14	13, 1, 10, 7, 8, 3	0.112, 0.281, 0.663, 0.893, 0.976, 1	0.164	35.253	0.97 ± 0.04
15	13, 10, 7, 5, 2, 1	0.112, 0.449, 0.753, 0.910, 0.983, 1	0.217	29.370	0.97 ± 0.04
16	3, 1, 12, 11, 10, 6	0.028, 0.129, 0.646, 0.910, 0.983, 1	0.169	45.370	0.93 ± 0.07
17	3, 1, 7, 4, 8, 2	0.028, 0.129, 0.646, 0.882, 0.989, 1	0.155	51.253	0.93 ± 0.06
18	3, 11, 13, 1, 8, 6	0.0281, 0.107, 0.534, 0.860, 0.983, 1	0.156	45.370	0.94 ± 0.06
19	13, 10, 12, 8, 1, 7	0.1124, 0.449, 0.736, 0.910, 0.978, 1	0.155	37.823	0.97 ± 0.04
20	1, 11, 7, 4, 3, 5	0.039, 0.303, 0.736, 0.905, 0.989, 1	0.167	33.370	0.95 ± 0.06

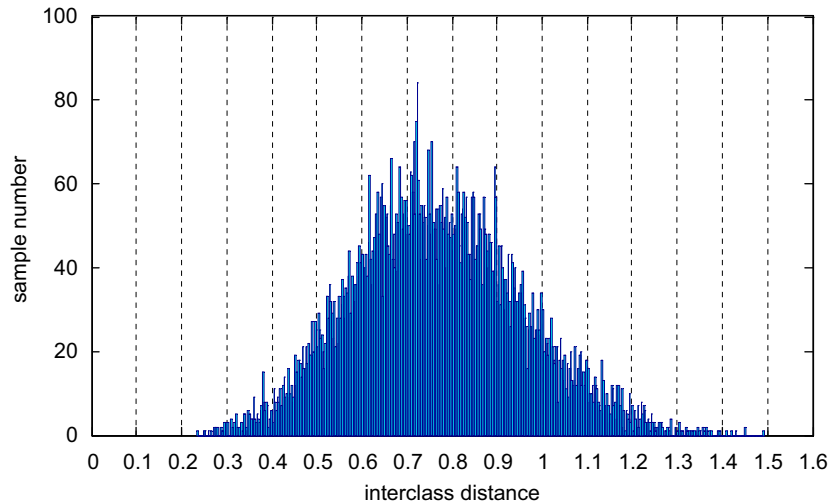


Fig. 5. Distribution of neighborhood margin of samples in different feature subspace (subset 3: 1, 11, 13, 2, 5, 7).

of KNN ($k=5$) in these subspaces. We can see that there are great differences in classification accuracies between these feature subsets.

What is the difference between these feature subsets? We compute the neighborhood margins of the classification task in different feature subspaces, as shown in the fourth column in Table 1 where the neighborhood margin is computed as the minimal distance between classes. The fourth column shows that the real neighborhood margins are usually greater than 0.15 and some feature subsets can yield much greater margin than 0.15. It is easy to see that the feature subsets yielding large margin usually produce good classification performances, such as subsets 1, 3 and 15. However, some subsets of features deriving small margins also produce competent classification performances, such as feature subsets 13, 14 and 19.

Why the feature subset with small neighborhood margin can also generate good performances? We show the distribution of interclass of samples in Figs. 5, 6 and 7, where we compute the distances of samples to all samples coming from different classes.

Comparing Figs. 5, 6 and 7, we can find the minimal interclass distance in feature subset 3 is far greater than those in feature subset 5 and 13. Moreover, we can also see that although the

minimal interclass distance in feature subset 13 is smaller than that in feature subset 5, the samples with interclass distance [0.2, 0.3] in feature subset 13 are much less those in feature subset 5. As the samples with interclass distance [0.2, 0.3] are also easy to be misclassified, the classification accuracies in feature subset 13 are higher than in feature subset 5. This fact shows that the measure of crisp margin cannot reflect the real ability of features. Soft margin should be introduced. The fifth column of Table 1 shows the neighborhood soft margins in the corresponding subsets of features. We compute the correlation coefficient between NSM and KNN accuracy and get the value -0.80 ; while the coefficient between neighborhood margin and KNN accuracy is 0.36. The results show that NSM is much better neighborhood margin in evaluating features.

In Table 1, we observe that features 4 and 8 usually appear at the end of feature subsets, whereas features 10 and 13 are picked up in the beginning. So we estimate the class probability distribution in these features and show them in Fig. 8. We can see that class 2 is overlapped with classes 1 and 3 if we consider feature 4 or feature 8. But class 2 and class 3 are well separated with respect to feature 10; class 2 and class 1 are well separated with respect to feature 13. So regarding features 10 and 13, the

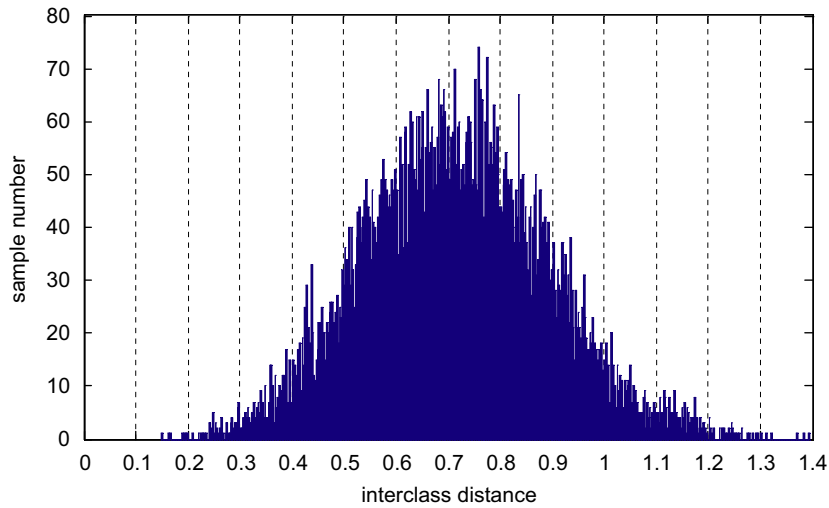


Fig. 6. Distribution of neighborhood margin of samples in different feature subspace (subset 13: 13, 11, 7, 5, 10, 3).

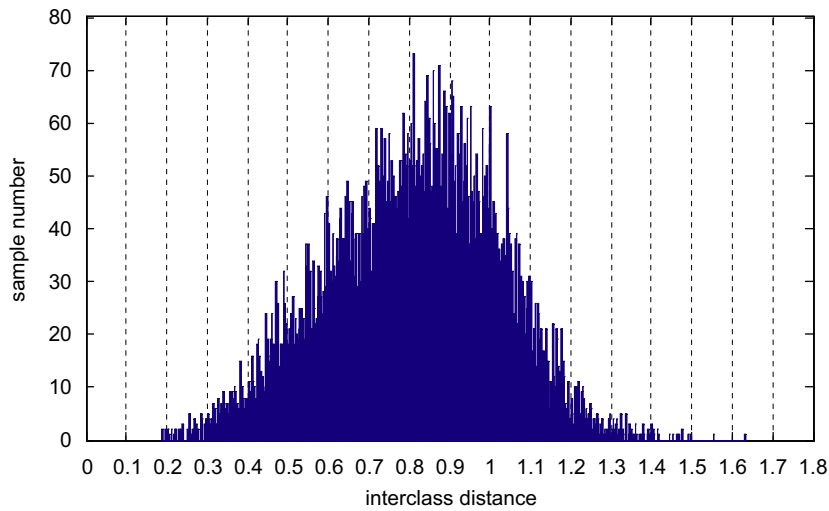


Fig. 7. Distribution of neighborhood margin of samples in different feature subspace (subset5: 1, 12, 10, 4, 2, 7).

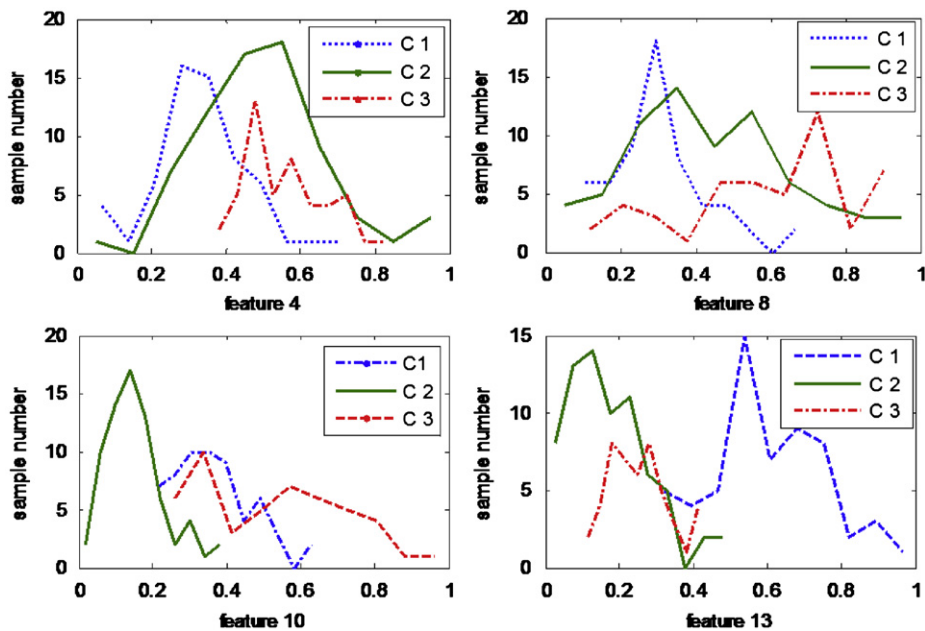


Fig. 8. Distribution of class probability in different feature subspaces (wine).

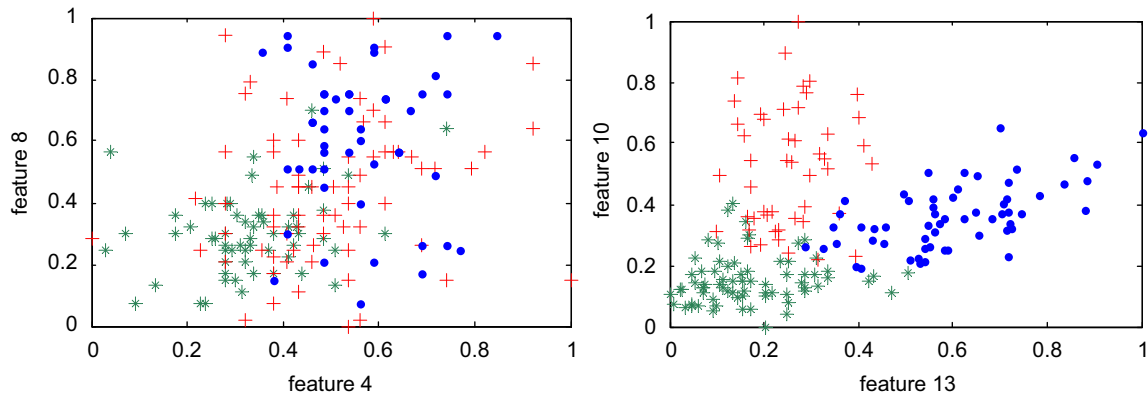


Fig. 9. Sample scatter in different feature subspaces (wine).

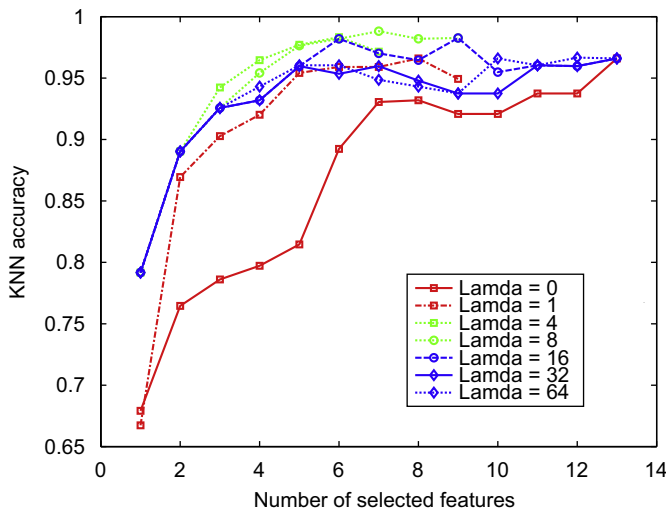


Fig. 10. Comparison of classification performance with different λ (KNN).

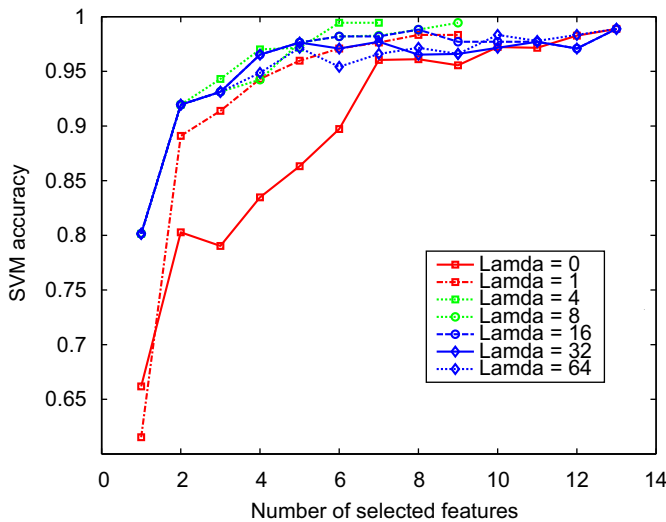


Fig. 11. Comparison of classification performance with different λ (SVM).

Table 2
Data description.

Data	Sample	Feature	Class
German	1000	20	2
Heart	270	13	2
Hepatitis	155	19	2
Horse	368	22	2
Iono	351	34	2
Segmentation	2310	18	7
WDBC	569	30	2
Wine	178	13	3

There is a parameter λ to be set in computing neighborhood soft margin. We try $\lambda = 0, 1, 4, 8, 16, 32, 64$ and compute the classification performance of the selected features with different λ . The results are given in Figs. 10 and 11, where data wine is still used in the experiment.

$\lambda = 0$ means that we just consider neighborhood margin and ignore boundary samples, while $\lambda = 64$ shows that the number of boundary samples is very important. We should reduce the boundary samples in feature selection.

From Figs. 10 and 11, we get that the performance is the best if $\lambda = 4$. We also try other benchmark tasks and find $\lambda = 4$ is a proper value for parameter λ . So we set $\lambda = 4$ in the following experiments.

Eight data were collected from UCI repository of machine learning databases for testing the proposed technique [19], as outlined in Table 2. The numbers of samples vary from 178 to 2310, and the numbers of features are between 13 and 34.

We evaluate the candidate features with neighborhood soft margin and get a rank for each classification task. Then we compute the classification accuracy of the first m features with KNN and RBF-SVM classifiers based on 10-fold cross validation, where $m = 1, 2, 3, \dots$. Then we get a sequence of classification accuracies for each task, as shown in Fig. 12.

In the same time, we also conduct feature selection on these data sets with correlation (CFS) [10], consistency (C) [12] and neighborhood rough sets (NRS) based algorithms [9].

The optimal numbers of features selected with these algorithms are given in Table 3. As a whole, we see that NSM selects the relatively less features. For some data sets, such as German, heart, iono and WDBC, just several features are selected. While correlation and neighborhood rough sets based algorithms get the most features.

Then we compare the classification accuracies of these feature subsets in Tables 4 and 5, where “raw” denotes the classification

three classes of samples are well discriminated. They are useful for classifying different kinds of wines as shown in Fig. 9. Correspondingly, the average margins between classes are greater than other feature pair.

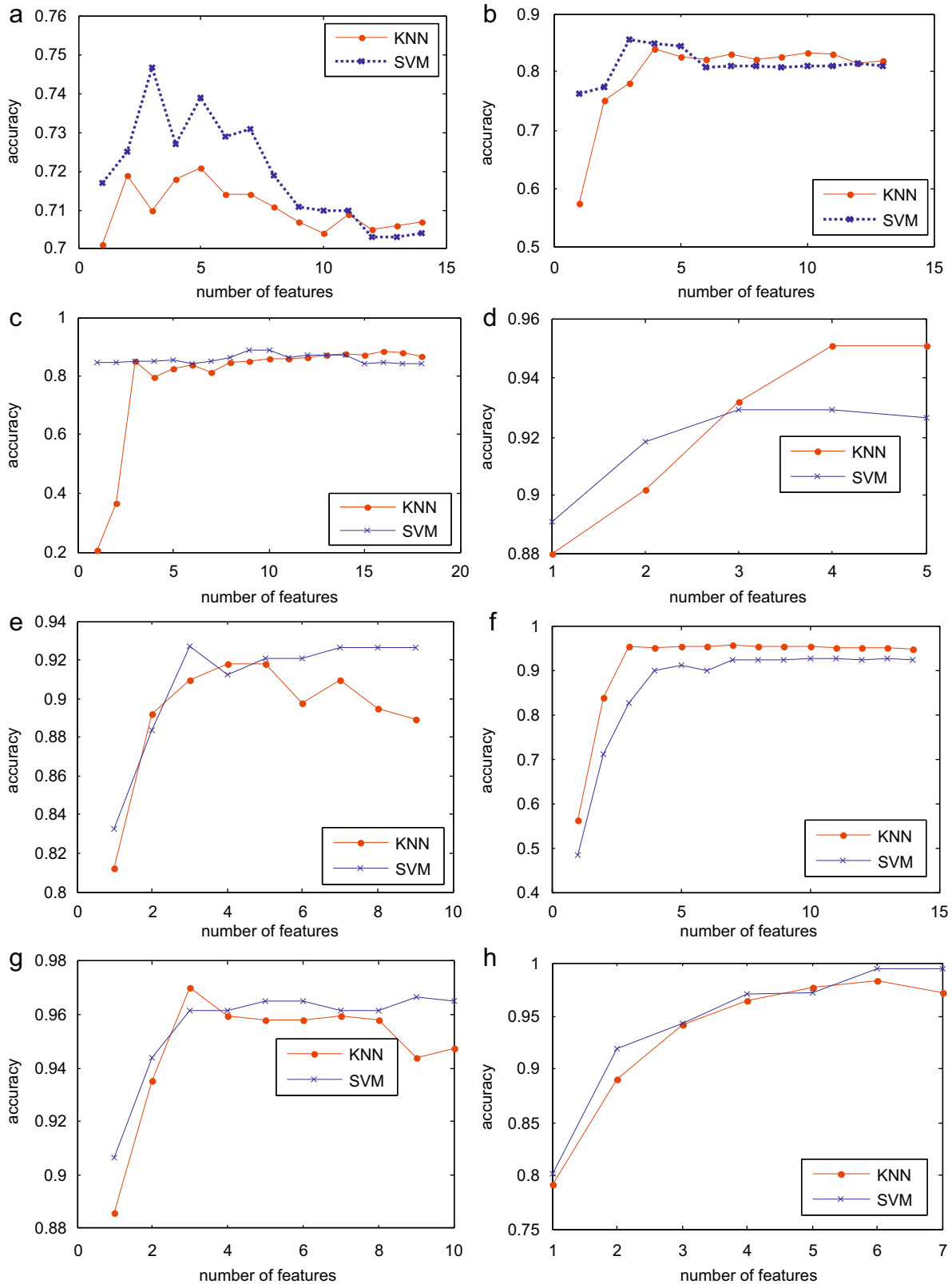


Fig. 12. Variation of classification accuracies with number of features: (a) German; (b) heart; (c) hepatitis; (d) horse; (e) iono; (f) segmentation; (g) WDBC; (h) wine.

accuracy of the raw data sets, NSM is the classification accuracy of features selected with neighborhood soft margin, and CFS, C and NRS are accuracies derived with features selected with correlation, consistency and neighborhood rough sets based algorithms, respectively.

In Tables 4 and 5, markers \uparrow , \downarrow and $-$ means the classification accuracy increase, decrease and keep with respect to the raw data. Compared with KNN accuracies derived from the raw data and NSM, we see that six of the eight tasks are improved and one task keep invariant although more than two thirds of features are removed, as

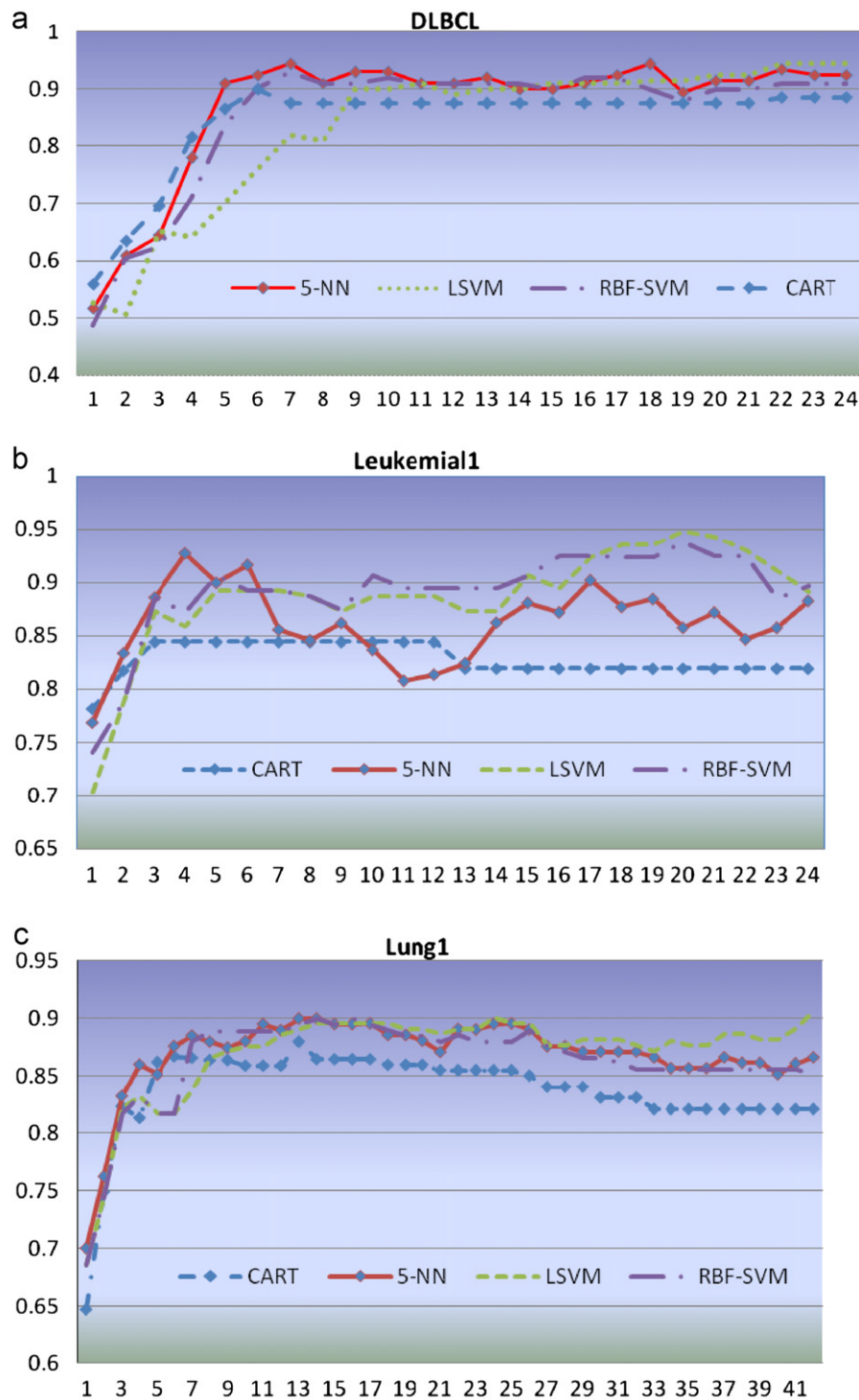


Fig. 13. Variation of classification accuracy with the number of selected features: (a) DLBCL; (b) Leukemia1; (c) Lung1.

shown in Table 3. As to RBF-SVM, five tasks are improved and two tasks are worsened a little. These results show that the proposed technique is effective for feature selection in most of the cases.

Comparing the proposed algorithm with other techniques of feature selection, we also see that NSM is competent. Especially for KNN classifiers, NSM is almost the best one on all the classification tasks except data set horse. As to RBF-SVM, NSM also get good results in majority of samples.

Moreover, we compare the proposed algorithm with some existing large-margin based feature selection techniques including ReliefF [15], SVM [22] and Simba [20] in Tables 6 and 7.

Based on these experiments, we see that the neighborhood soft margin based feature selection algorithm is competent with the classical techniques.

Robustness is one of the advantages of the proposed algorithm. We randomly change the labels of 10% samples in the raw data

Table 3
Number of features selected with different algorithms.

Data	Raw	NSM-KNN	NSM-SVM	CFS	C	NRS
German	20	5	3	5	11	11
Heart	13	4	3	7	10	9
Hepatitis	19	16	14	7	8	7
Horse	22	4	4	8	4	8
Iono	34	4	9	14	7	9
Segmentation	18	11	11	7	5	10
WDBC	30	3	9	11	8	12
Wine	13	6	6	11	5	6
Total	169	53	59	70	58	74

Table 4
KNN accuracy of features selected with different algorithms ($k=3$).

Data	Raw	NSM	CFS	C	NRS
German	0.69	0.73↑	0.70	0.72	0.71
Heart	0.82	0.84↑	0.83	0.84	0.83
Hepatitis	0.87	0.90↑	0.88	0.90	0.90
Horse	0.90	0.95↑	0.92	0.88	0.90
Iono	0.84	0.92↑	0.86	0.88	0.88
Segmentation	0.96	0.95↓	0.95	0.95	0.95
WDBC	0.97	0.97–	0.97	0.97	0.95
Wine	0.95	0.98↑	0.97	0.96	0.97

Table 5
RBF-SVM accuracy of features selected with different algorithms.

Data	Raw	NSM	CFS	C	NRS
German	0.74	0.75↑	0.72	0.72	0.74
Heart	0.83	0.86↑	0.82	0.83	0.83
Hepatitis	0.87	0.89↑	0.90	0.86	0.91
Horse	0.90	0.93↑	0.92	0.83	0.91
Iono	0.95	0.95–	0.97	0.93	0.95
Segmentation	0.90	0.96↑	0.93	0.90	0.89
WDBC	0.98	0.97↓	0.98	0.97	0.97
Wine	0.99	0.99–	0.99	0.98	0.99

Table 6
Comparison of large-margin based algorithms (KNN accuracy/feature number).

Data	ReliefF	SVM	Simba	NSM
German	0.73/7	0.73/5	0.72/15	0.72/5
Heart	0.83/12	0.81/5	0.83/10	0.84/4
Hepatitis	0.89/8	0.87/12	0.87/13	0.89/16
Horse	0.93/3	0.92/3	0.90/8	0.95/4
Iono	0.91/6	0.96/16	0.92/6	0.92/4
Segmentation	0.96/13	0.93/13	0.96/13	0.96/7
WDBC	0.97/30	0.98/23	0.97/26	0.97/3
Wine	0.98/10	0.99/13	0.98/7	0.98/6

Table 7
Comparison of large-margin based algorithms (SVM accuracy/feature number).

Data	ReliefF	SVM	Simba	NSM
German	0.75/16	0.76/4	0.74/8	0.75/3
Heart	0.83/6	0.86/4	0.83/11	0.86/3
Hepatitis	0.88/14	0.92/5	0.88/12	0.89/14
Horse	0.93/3	0.93/7	0.91/5	0.95/4
Iono	0.96/17	0.95/11	0.95/21	0.95/9
Segmentation	0.91/11	0.93/8	0.95/7	0.96/11
WDBC	0.98/22	0.98/11	0.98/16	0.97/9
Wine	0.99/13	0.99/9	0.99/13	0.99/6

Table 8
Performance comparison of NSM on the raw data and noisy data (accuracy/feature number).

Data	SVM	SVM-noise	KNN	KNN-noise
German	0.75/3	0.75/3	0.72/5	0.72/2
Heart	0.86/3	0.86/3	0.84/4	0.84/4
Hepatitis	0.89/14	0.88/7	0.89/16	0.87/7
Horse	0.93/4	0.89/3	0.95/4	0.92/4
Iono	0.95/9	0.96/13	0.92/4	0.92/5
Segmentation	0.96/11	0.95/9	0.96/7	0.96/7
WDBC	0.97/9	0.96/9	0.97/3	0.95/9
Wine	0.99/6	0.98/4	0.98/6	0.98/4

sets. It means we add 10% class noise in the data sets. Then we conduct the algorithms on them again.

Comparing the results in Table 8, although 10% class noise is added in the raw data, we do not observe significant reduction of classification performance. These results show NSM can work on noisy tasks.

In order to test the proposed technique on data with thousands of features, we gathered three cancer classification tasks including DLBCL, Leukemia 1 and Lung1 [21]. DLBCL is a data set recording 88 measurements of diffuse large B-cell lymphoma. This dataset contains 4026 array elements. Leukemia (ALL V.S. AML), shortly Leukemia 1, is a collection of 72 expression measurements. It contains a training set composed of 27 samples of acute lymphoblastic leukemia (ALL) and 11 samples of acute myeloblastic leukemia (AML), and an independent test set composed of 20 ALL and 14 AML samples, where each sample is described with 7129 probes from 6817 human genes. Lung Cancer (Dana-Farber Cancer Institute, Harvard Medical School), A total of 203 snap-frozen lung tumors and normal lung were analyzed. The 203 specimens include 139 samples of lung adenocarcinomas (labelled as ADEN), 21 samples of squamous cell lung carcinomas (labelled as SQUA), 20 samples of pulmonary carcinoids (labelled as COID), 6 samples of small-cell lung carcinomas (labelled as SCLC) and 17 normal lung samples (labelled as NORMAL). Each sample is described by 12 600 genes.

We conduct the neighborhood soft margin based feature selection algorithm on these classification tasks and observe the variation of classification accuracy with the number of features selected with this algorithm. The curves are presented in Figs. 11, 13, where we compute classification accuracies with KNN ($K=5$), linear support vector machine (LSVM), RBF support vector machine (RBF-SVM) and CART based on the 10-fold cross validation technique.

6. Conclusions

Feature selection is considered to be an important preprocessing step in machine learning and pattern recognition. Feature evaluation is a key issue when constructing an algorithm for feature selection. In this work, we propose a new concept of neighborhood margin and neighborhood soft margin, which reflects the minimal distance between different classes.

We use the criterion of neighborhood soft margin to evaluate the quality of candidate features and construct a forward greedy algorithm for feature selection. The connection between neighborhood margin and margin defined in SVM is discussed. It is shown that a task is 2ϵ neighborhood separable if it is ϵ linearly separable with respect to linear SVM. We conduct this technique on some classification learning tasks. When compared to some feature selection algorithms, the proposed technique is shown to be effective in most of the cases.

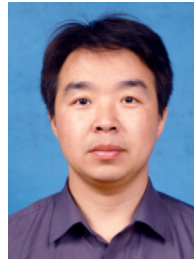
Neighborhood soft margin can be looked as a local approximation strategy for feature evaluation. The connection between neighborhood soft margin, sample margin and hypothesis margin is not discussed in this paper. We will show a systematic analysis on these concepts and give risk estimation on neighborhood soft margin in the future.

Acknowledgments

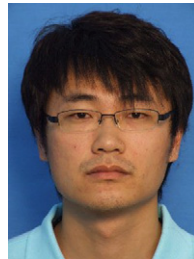
This work is supported by National Natural Science Foundation of China under Grant 60703013 and 10978011 and Prof. Yu is supported by National Science Fund for Distinguished Young Scholars under Grant 50925625. Dr. Hu is supported by The Hong Kong Polytechnic University (G-YX3B).

References

- [1] H. Wang, D. Bell, F. Murtagh, Axiomatic approach to feature subset selection based on relevance, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21 (1999) 271–277.
- [2] L. Yu, H. Liu, Efficient feature selection via analysis of relevance and redundancy, *Journal of Machine Learning Research* 5 (2004) 1205–1224.
- [3] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, *Journal of Machine Learning Research* 3 (2003) 1157–1182.
- [4] H. Liu, L. Yu, Toward integrating feature selection algorithms for classification and clustering, *IEEE Transactions on Knowledge and Data Engineering* 17 (2005) 491–502.
- [5] R. Kohavi, G.H. John, Wrappers for feature subset selection, *Artificial Intelligence* 97 (1997) 273–324.
- [6] S. Abe, R. Thawonmas, Y. Kobayashi, Feature selection by analyzing class regions approximated by ellipsoids, *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews* 28 (1998) 282–287.
- [7] J. Neumann, C. Schnorr, G. Steidl, Combined SVM-based feature selection and classification, *Machine Learning* 61 (2005) 129–150.
- [8] T.K. Ho, M. Basu, Complexity measures of supervised classification problems, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002) 289–300.
- [9] Q.H. Hu, D. Yu, J.F. Liu, C. Wu, Neighborhood rough set based heterogeneous feature subset selection, *Information Sciences* 178 (2008) 3577–3594.
- [10] M.A. Hall, Correlation-based feature selection for discrete and numeric class machine learning, in: *Proceedings of 17th International Conference on Machine Learning*, 2000, pp. 359–366.
- [11] Q.H. Hu, D.R. Yu, Z.X. Xie, Neighborhood classifier, *Expert Systems with Applications* 34 (2008) 866–876.
- [12] M. Dash, H. Liu, Consistency-based search in feature selection, *Artificial Intelligence* 151 (2003) 155–176.
- [13] K. Crammer, R. Gilad-Bachrach, A. Navot, N. Tishby, Margin analysis of the LVQ algorithm, in: *Proceedings of 17th Conference on Neural Information Processing Systems*, 2002.
- [14] Y. Sun, J. Li, Iterative RELIEF for Feature Weighting, in: *Proceedings of 23rd International Conference on Machine Learning*, 2006, pp. 913–920.
- [15] I. Kononenko, Estimating attributes: analysis and extensions of reliEF, in: *Proceedings of European Conference on Machine Learning*, 1994, pp. 171–182.
- [16] Y. Li, B.-L. Lu, Feature selection based on loss-margin of nearest neighbor classification, *Pattern Recognition* 42 (2009) 1914–1921.
- [17] C. Cortes, V. Vapnik, Support-vector networks, *Machine Learning* 20 (1995) 273–297.
- [18] V. Vapnik, *Statistical Learning Theory*, Wiley, NY, 1998.
- [19] C.L. Blake, C.J. Merz, UCI repository of machine learning databases, <<http://www.ics.uci.edu/mllearn/MLRepository.html>> (1998).
- [20] R. Gilad-Bachrach, A. Navot, N. Tishby, Margin based feature selection—theory and algorithm, in: *Proceedings of 21st International Conference on Machine Learning*, Banff, Canada, 2004.
- [21] S. Bandyopadhyay, U. Maulik, D. Roy, Gene identification: classical and computational intelligence approaches, *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 38 (2008) 55–68.
- [22] I. Guyon, J. Weston, M.D. Stephen Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, *Machine Learning* 46 (2002) 389–422.
- [23] P. Bartlett, P. Long, R. Williamson, Fat-shattering and the learnability of real-valued functions, *Journal of Computer and System Sciences* 52 (1996) 434–452.
- [24] W.-Z. Wu, Attribute reduction based on evidence theory in incomplete decision systems, *Information Sciences* 178 (2008) 1355–1371.



Qinghua Hu received the Master Degree and Ph.D. from Harbin Institute of Technology, Harbin, China in 2002 and 2008, respectively. Now he is an Associate Professor with Harbin Institute of Technology and a postdoctoral fellow with the Hong Kong Polytechnic University. His research interests are focused on data mining, knowledge discovery with fuzzy and rough techniques. He is a PC co-chair of RSCTC 2010 and serves as referee for a great number of journals and conferences. He has authored or coauthored more than 60 journal and conference papers in the areas of machine learning, data mining and rough set theory.



Xunjian Che is a master student with Harbin Institute of Technology. His research interests are focused on large-margin learning theory, preference learning.



Lei Zhang received the B.S. degree in 1995 from the Shenyang Institute of Aeronautical Engineering, Shenyang, China, and the M.S. and Ph.D. degrees in Electrical and Engineering from Northwestern Polytechnical University, Xi'an, China, respectively, in 1998 and 2001. From 2001 to 2002, he was a Research Associate with the Department of Computing, The Hong Kong Polytechnic University. From January 2003 to January 2006, he was a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, McMaster University, Canada. Since January 2006, he has been an Assistant Professor in the Department of Computing, The Hong Kong Polytechnic University. His research interests include image and video

processing, biometrics, pattern recognition, multisensor data fusion, machine learning and optimal estimation theory, etc.



Daren Yu received the M.Sc. and D.Sc. degrees from Harbin Institute of Technology, Harbin, China, in 1988 and 1996, respectively. Since 1988, he has been working at the School of Energy Science and Engineering, Harbin Institute of Technology. His main research interests are in modeling, simulation, and control of power systems. He has published more than 100 conference and journal papers on power control and fault diagnosis.