Artif Intell Rev (2011) 36:99–115
DOI 10.1007/s10462-011-9205-2

Feature selection for support vector machines with RBF kernel

Quanzhong Liu · Chihau Chen · Yang Zhang · Zhengguo Hu

Published online: 9 February 2011

Abstract Linear kernel Support Vector Machine Recursive Feature Elimination (SVM-RFE) is known as an excellent feature selection algorithm. Nonlinear SVM is a black box classifier for which we do not know the mapping function Φ explicitly. Thus, the weight vector w cannot be explicitly computed. In this paper, we proposed a feature selection algorithm utilizing Support Vector Machine with RBF kernel based on Recursive Feature Elimination (SVM-RBF-RFE), which expands nonlinear RBF kernel into its Maclaurin series, and then the weight vector w is computed from the series according to the contribution made to classification hyperplane by each feature. Using w_i^2 as ranking criterion, SVM-RBF-RFE starts with all the features, and eliminates one feature with the least squared weight at each step until all the features are ranked. We use SVM and KNN classifiers to evaluate nested subsets of features selected by SVM-RBF-RFE. Experimental results based on 3 UCI and 3 microarray datasets show SVM-RBF-RFE generally performs better than information gain and SVM-RFE.

Keywords Feature selection · RBF kernel · Information gain · SVM-RFE · Recursive Feature Elimination

Q. Liu (✉) · Z. Hu
College of Mechanical and Electric Engineering, Northwest A & F University,
Yangling, 712100 Shaanxi Province, China
e-mail: liuqzhong@nwsuaf.edu.cn

Z. Hu
e-mail: zghu@sina.com

C. Chen
Electrical & Computer Engineering, University of Massachusetts Dartmouth,
Dartmouth, MA 02747-2300, USA
e-mail: cchen@umassd.edu

Y. Zhang
College of Information Engineering, Northwest A & F University,
Yangling, 712100 Shaanxi Province, China
e-mail: zhangyang@nwsuaf.edu.cn

1 Introduction

Feature selection is a key technology which can eliminate irrelevant features, reduce data dimensionality, and increase learning efficiency and improve predictive performance. It has been an active research area in pattern recognition, machine learning and data mining communities (Elalami 2009; Guyon and Elisseeff 2003; Huang et al. 2007; Sun 2007). Gene selection from microarray data and feature selection from text categorization are two typical application domains. In the gene selection domain, the gene expression data usually has thousands or even tens of thousands of genes features but much fewer tissue samples available (usually a few hundred). In order to find gene subset which is strongly correlated with disease (cancer), and improves the classification accuracy as well as to provide helpful information for doctors to diagnose illness, several gene selection algorithms have extensively been developed in related research literature, such as Bontempi et al. (2007), Niijima and Kuhara (2006), Li and Yang (2005), Silva et al. (2005), Zhang et al. (2006) and so on. In text categorization domain, documents contain hundreds of thousands of words, so it is critical for many text data mining problems to select subsets of features that are useful to build a good predictor (Guyon and Elisseeff 2003; Youn and Jeong 2009; Brank et al. 2002).

Feature selection algorithms can be broadly divided into two categories, the filter method and the wrapper model (Kohavi and John 1997). Filter methods select subset of features as a preprocessing step that ignore the effects of the selected feature subset on the performance of learning algorithm (Claeskens et al. 2008). In the machine learning community, information gain has been proved a successful filter algorithm (Lee and Lee 2006), and so is Mutual information algorithm (Estevez et al. 2009). Wrappers use the classification method to score subsets of variables. SVM-RFE is a typical successful wrapper algorithm (Li and Yang 2005; Zhang et al. 2006), and was originally proposed to perform gene selection for cancer classification problems (Guyon et al. 2002), and was further improved and applied to select informative genes from bioinformatics data (Duan and Rajapakse 2004a,b; Duan et al. 2005; Ding and Wilkins 2006; Tang et al. 2007). Wrapper methods are computationally complex but consider the correlation among features and perform better than filter method (Sun 2007).

For SVM-RFE algorithm discussed above, the weight vector w is a normal to the hyperplane that separates two classes of examples with maximum margin, which is the distance from the hyperplane to the closest positive (negative) example. Let there be d training samples $\{\overrightarrow{X_i}, y_i\}, i = 1, 2, \ldots, d$ where y_i denotes the class label of sample $\overrightarrow{X_i}$. The weight vector can be computed by equation $w = \sum_{i=1}^{d} \alpha_i y_i \Phi(\overrightarrow{X_i})$, Here, $\alpha_i (1 \leq i \leq d)$ is knowledge learned by SVM. If a nonlinear kernel is used in SVM, we do not know the mapping function Φ explicitly, and so the weight vector cannot be computed explicitly. To the best of our knowledge, feature selection for Support Vector Machines usually adopts a linear kernel in the research community. In this paper, we propose a wrapper algorithm SVM-RBF-RFE, which expands nonlinear RBF kernel into its Maclaurin series, and compute the weight vector from the series according to the contribution made to classification by each feature. Then, using w_i^2 as ranking criterion, the algorithm starts with all the features, and eliminates one feature with the least squared weight in the weight vector at each step, and repeat this process until all features are ranked.

In order to validate the effectiveness of our algorithm, we compare SVM-RBF-RFE with SVM-RFE and information gain algorithms on three UCI datasets with more features and three microarray data, and use SVM with a linear kernel (SVML), SVM with RBF

kernel (SVM), a Nearest Neighbor with five neighbors (5NN), and a Nearest Neighbor with 10 neighbors (10NN) to evaluate feature subsets selected by the three algorithms. Experimental results show that the proposed algorithm is very encouraging over experimental datasets.

This paper is organized as follows. Section 2 introduces a state-of-the-art feature selection algorithm SVM-RFE. Our algorithm SVM-RBF-RFE is proposed in Sect. 3. Section 4 provides our experimental results on six datasets and Sect. 5 concludes the paper.

2 SVM-RFE algorithm

SVM is a classification algorithm based on statistical learning theory (Vapnik 1998; Burges 1998). The SVM constructs a hyperplane with maximum margin in the feature space, which is mapped from the original input space by the mapping function Φ. Let us use $\overrightarrow{\mathbf{x_i}}$ and $\overrightarrow{\mathbf{z_i}}$ to denote a pair of corresponding vectors in the original input space and in the feature space, respectively, then $\overrightarrow{Z_i} = \Phi(\overrightarrow{X_i})$.

A dataset with d samples could be represented as $\{\overrightarrow{X_i}, y_i\}, i = 1, 2, \ldots, d$, with $\overrightarrow{X_i} \in \{0, 1\}^m$, representing a sample data, and $y_i \in \{+1, -1\}$, representing the class label of this sample. For a testing sample X, the optimal hyperplane constructed by SVM in the feature space is:

$$< w, \Phi(X) > + b = 0 \tag{1}$$

The optimization problem must satisfy the following constraints (Cristianini and Taylor 2000):

$$y_i[< w, \Phi(\overrightarrow{X_i}) > + b] + \xi_i - 1 \geq 0 \quad \xi_i \geq 0 \quad i = 1, 2, \ldots, d \tag{2}$$

$$\min_{w, \xi_i} ||w||^2/2 + C\left(\sum_{i=1}^{d} \xi_i\right) \tag{3}$$

It can be proved that the hyperplane which satisfies the above constraints is an optimal hyperplane. Here, C is a specified constant that controls the trade-off between maximizing the margin and minimizing the training error term.

The optimization problem is usually translated into its dual form by the Lagrangian. For the detailed process, please refer to Duan and Rajapakse (2004b). We can obtain the weight vector and the hyperplane function by the Lagrangian for this problem.

$$w = \sum_{i=1}^{d} \alpha_i y_i \overrightarrow{Z_i} \tag{4}$$

$$f(Z) = b + \sum_{i=1}^{d} \alpha_i y_i < \overrightarrow{Z_i}, Z > \tag{5}$$

Here, $<, >$ denotes the inner product of two vectors. In SVM, kernel function $K(\overrightarrow{X_i}, \overrightarrow{X_j})$ computes the inner product of two vectors in the feature space: $K(\overrightarrow{X_i}, \overrightarrow{X_j}) = < \Phi(\overrightarrow{X_i}), \Phi(\overrightarrow{X_j}) > = < \overrightarrow{Z_i}, \overrightarrow{Z_j} >$, If a nonlinear kernel is applied to SVM, such as RBF, SIGMOID kernel, the weight vector w cannot be computed directly according to equation 4 because we do not know the mapping function Φ. A linear kernel is often used in

the research community: $K(\overrightarrow{X_i}, \overrightarrow{X_j}) =< \overrightarrow{X_i}, \overrightarrow{X_j} >$. For a linear kernel, weight w can be represented as:

$$w = \sum_{i=1}^{d} \alpha_i y_i \overrightarrow{X_i} \tag{6}$$

If a linear kernel is used, SVM-RFE algorithm starts with all the features, and eliminates one feature with the least squared weight at each step until all the features are ranked. In each iteration, w_i^2 is used as the feature ranking criterion.

In SVM-RFE algorithm, the objective function is $J = ||w||^2/2$ as employed in the OBD algorithm (LeCun et al. 1990), which approximates the change of J by removing the ith gene by expanding J in the Taylor series to second order: $\triangle J(i) = \frac{\partial J}{\partial w_i}\triangle w_i + \frac{\partial^2 J}{\partial w_i^2}\triangle w_i^2$. In each iteration, the elimination of the feature with the least squared weight will cause the least effect on J (Guyon et al. 2002; Duan and Rajapakse 2004a; Tang et al. 2007). Therefore, w_i^2 is adopted as ranking criterion, and in order to improve the efficiency of the algorithm, more features can be eliminated at each step.

3 The proposed method

Like many machine learning algorithms, we focus on sample data with discrete attributes (features). Numerical attributes could be discretized into discrete attributes (Fayyad and Irani 1993). For a discrete attribute A, if there are $|A|$ possible values for this attribute, then we can use $|A|$ Boolean literals to represent this attribute, with each Boolean literal representing the occurrence or non-occurrence of the corresponding attribute value.

Three UCI datasets in our experiments are preprocessed according to the above discrete method. The attributes of other three bioinformatics datasets are discrete in original samples. The expression value of each gene is represented as P, M, or A that indicates whether RNA for the gene is present, marginal, or absent, respectively based upon the matched and mismatched probes for the genes.

3.1 RBF Kernel function

Among many kernel functions, RBF kernel is a default and recommended kernel function for SVM classifier. Suppose $U \in R^n$, $V \in R^n$, $g \in R^+$, here, g is a hyper-parameter of RBF kernel. RBF kernel is defined as:

$$K(U, V) = \exp(-g||U - V||^2) \tag{7}$$

After preprocessing in the above discrete way, the input space of sample data becomes $\{0, 1\}^n$. Suppose $U \in \{0, 1\}^n$, $V \in \{0, 1\}^n$, $g \in R^+$, we have:

$$||U - V||^2 = \sum_{i=1}^{n}(U_i - V_i)^2 = \sum_{i=1}^{n} U_i(1 - V_i) + V_i(1 - U_i) \tag{8}$$

The following mapping is used to map vector U from a vector of length n to a vector with length $2n$.

$$\psi(U) = (U_1, U_2, U_3, \ldots, U_n, \overline{U_1}, \overline{U_2}, \overline{U_3}, \ldots, \overline{U_n}) \tag{9}$$

here, $\overline{U_i} = 1 - U_i, 1 \leq i \leq n$. In the rest of the paper, we will use $\psi(U_i)$ to represent the ith element of vector $\psi(U)$.

Similarly, we use the following mapping to map vector V.

$$\tau(V) = (\overline{V_1}, \overline{V_2}, \overline{V_3}, \ldots, \overline{V_n}, V_1, V_2, V_3, \ldots, V_n) \tag{10}$$

In the rest of the paper, we use $\tau(V_i)$ to represent the ith element of the vector $\tau(V)$.

Hence, we get:

$$||U - V||^2 = < \psi(U), \tau(V) > = < \psi(V), \tau(U) > \tag{11}$$

Here, $<, >$ represents the inner product between the two vectors.

Following the above mappings, RBF kernel could be represented as:

$$K(U, V) = \exp(-g < \psi(U), \tau(V) >) \tag{12}$$

3.2 Weight of features

A dataset with d samples could be represented as $\{\overrightarrow{X_i}, y_i\}, i = 1, 2, \ldots, d$, with $\overrightarrow{X_i} \in \{0, 1\}^n$, representing a sample data, and $y_i \in \{+1, -1\}$, representing the class label of this sample. For a testing sample X, the classification function learned by non-linear SVM with RBF kernel could be represented as:

$$F(X) = \text{sgn}\left(b + \sum_{i=1}^{d} \alpha_i y_i K(\overrightarrow{X_i}, X)\right) \tag{13}$$

Here, $\alpha_i (1 \leq i \leq d)$ and b are knowledge learned by SVM; sgn() represents the sign function. $K(\overrightarrow{X_i}, X)$ represent a kernel function, which is a RBF kernel function here. Let's consider the following function:

$$f(X) = b + \sum_{i=1}^{d} \alpha_i y_i K(\overrightarrow{X_i}, X) \tag{14}$$

Suppose $\lambda_i = \alpha_i y_i K(\overrightarrow{X_i}, X) = \alpha_i y_i \exp(-g < \psi(\overrightarrow{X_i}), \tau(X) >)$, here, $K(\overrightarrow{X_i}, X)$ is replaced with equation (12) and expanded into its Maclaurin series (Liu et al. 2007):

$$\begin{aligned}
\lambda_i &= \alpha_i y_i \left(1 + \sum_{k=1}^{\infty} \frac{(-1)^k g^k}{k!} < \psi(\overrightarrow{X_i}), \tau(X) >^k\right) \\
&= \alpha_i y_i \left(1 - \frac{g^1}{1!} < \psi(\overrightarrow{X_i}), \tau(X) > \right. \\
&\quad + \frac{g^2}{2!} < \psi(\overrightarrow{X_i}), \tau(X) >^2 \\
&\quad - \frac{g^3}{3!} < \psi(\overrightarrow{X_i}), \tau(X) >^3 \\
&\quad \left. + \ldots\right)
\end{aligned} \tag{15}$$

For the item $< \psi(\overrightarrow{X_i}), \tau(X) >^m, m \in N$ in equation 15, we use $h(m, p)$ to represent the sum of the coefficients of terms with length $p(p < n)$ in the expansion of $< \psi(\overrightarrow{X_i}), \tau(X) >^m$, and $h(m, p)$ can be computed by the following equation.

$$\begin{cases} h(m,1) = 1 \\ h(m,p) = p^m - \sum_{k=1}^{p-1} \binom{p}{k} h(m,k) & p \le m \\ h(m,p) = 0 & p > m \end{cases} \tag{16}$$

Equation 16 can be proved by mathematical induction, and the detailed proof is omitted here for lack of space.

Then, for λ_i, the contribution of dimension $\tau(X)_{j_1} (1 \le j_1 \le 2n)$ to classification could be represented as:

$$\frac{\partial \lambda_i}{\partial(\tau(X)_{j_1})} = \alpha_i y_i \psi(\overrightarrow{X_i})_{j_1} \sum_{k=1}^{\infty} \frac{(-1)^k g^k}{k!} h(k,1) \tag{17}$$

Here, we borrow the terminology from the research of mining association rules. If we look at the dimension $\tau(X)_{j_1}$ as an item, then equation 17 could be looked as the classification weight of 1-itemset $\{\tau(X)_{j_1}\}$ in λ_i. Therefore, the classification weight of 1-itemset $\{\tau(X)_{j_1}\}$ for classification decision hyperplane could be represented as:

$$\frac{\partial f(x)}{\partial(\tau(X)_{j_1})} = \sum_{i=1}^{d} \alpha_i y_i \psi(\overrightarrow{X_i})_{j_1} \sum_{k=1}^{\infty} \frac{(-1)^k g^k}{k!} h(k,1) \tag{18}$$

Equation 18 is used to compute the weight of each feature in experimental study.

3.3 SVM-RBF-RFE algorithm

Algorithm 1 shows the detail of our algorithm for feature selection by recursive feature elimination, which starts with all the features and remove one feature with the least squared weight at each step. Finally, all the features are ranked. In order to improve the efficiency of our algorithm, the algorithm can remove $percentToEliminate$ percentage rate of attributes each time until the number of remaining attributes up to $percentThreshold$. And then, the algorithm switch to eliminate $numToEliminate$ features at each step. Here, $percentToEliminate$, $percentThreshold$ and $numToEliminate$ are user defined parameters. The detailed computation about the number of elimination features at each step refers to step 7 in Algorithm 1. In our experiments, for 3 microarray data, we set $percentToEliminate = 10\%$, $percentToEliminate = 80$, $numToEliminate = 1$. That is to say, firstly, 10 percentage rates of attributes are eliminated per iteration until the number of remaining features is equal to 80; secondly, the algorithm switches to remove one feature in per iteration. For 3 UCI datasets, the algorithm starts with all the features and eliminates one feature per iteration.

In step 3 of algorithm 1, features with the least squared weight are removed from training samples. In step 5, a SVM with RBF kernel is trained by all the training samples, and obtaining α_i which is the knowledge learned. The weight vector is computed in step 6.

Algorithm 1 SVM-RBF-RFE Algorithm

Require: Training Sample, $X_0 = [X_1, X_2, \ldots, X_k, \ldots, X_l]$,
Class labels, $Y = [Y_1, Y_2, \ldots, Y_k, \ldots, Y_l]$,
Percentage rate of attribute elimination, $percentToEliminate$,
Threshold of percent elimination, $percentThreshold$,
Constant rate of attribute elimination per iteration, $numToEliminate$,
1: Initialize: Subset of surviving features $S = [1, 2, \ldots, n]$, Feature ranked list $r = [\ ]$
2: Repeat until $S = [\ ]$
3: Restrict training samples to good feature indices, $X_r = X_0[:, S]$
4: Expand training samples according to equation 9, and obtain new training samples $\psi(X)$
5: Train SVM classifier with RBF kernel function by all new training samples,
$\alpha = SVM - train(\psi(X), Y)$
6: Compute the weight vector of dimension length(S) according to equation 18
$w = \{w_i | w_i = \sum_{i=1}^{d} \alpha_i y_i \psi(\overrightarrow{X_i})_{j1} \sum_{k=1}^{\infty} \frac{(-1)^k g^k}{k!} h(k, 1)\}$
7: Compute the number of feature $NumElim$ eliminated per iteration
If $(percentToEliminate > 0)$ then
$NumElim = percentToEliminate * lengh(S)$
If $(lengh(S) - NumElim <= percentThreshold)$ then
$NumElim = lengh(S) - percentThreshold$
Else if $(lengh(S) >= numToEliminate)$ then
$NumElim = numToEliminate$
Else $NumElim = lengh(S)$
8: Find the subset f which consist of $NumElim$ features with the least squared weight in the weight vector w
9: Update feature ranked list $r = [S(f_i), r]$, $i = 1, 2, \ldots, length(f)$
10: Eliminate the subset f from subset s: $S = S(1 : (f_i) - 1, (f_i) + 1 : length(S))$, $i = 1, 2, \ldots length(f)$
Ensure: Feature ranked list r

4 Experimental result

In order to validate the effectiveness of our algorithm, we perform experiments on two test-beds, the first test-bed is composed of three UCI Datasets with more features and the second test-bed contains three Microarray Data, and compare our algorithm with information gain and SVM-RFE algorithms. SVML, SVM, 5NN and 10NN classifiers are used to evaluate the feature subsets selected by our algorithm, information gain and SVM-RFE algorithms, respectively. The four classifiers are from WEKA software which is publicly available at the Website (http://www.cs.waikato.ac.nz/ml/weka/) in Machine Learning community.

For datasets without natural training and testing partition, 10-fold cross validation is used and the average results on the 10-folds are reported as the final result.

Our experiments are performed on a PC with Pentium 4, 3.0 GHZ CPU and 512 MB memory. The algorithm was implemented in Java language. In our experiments, we set hyper-parameter of RBF kernel $g = 0.1$, and regulation parameter of SVM $c = 1$.

4.1 Datasets

The experimental data information is summarized in Table 1. Sonar, Hypo and Horse datasets are available from public UCI Machine Learning Repository (http://www.ics.uci.edu/mlearn/MLRepository.html). Column 4 gives the number of features after 3 UCI datasets are discretized.

The first microarray data is Leukemia (Golub et al. 1999). The training data set consist of 38 samples (27 acute lymphoblastic leukemia, ALL, and 11 acute myeloid leukemia, AML) from bone marrow specimens, and the testing data set contain 34 samples (20 ALL and

Table 1 Dataset summary of three UCI data and three microarray data

DataSets	#Training samples	#Testing samples	#Feature	#Class ratio
Sonar	208	/	42	111:97
Hypo	3,163	/	57	3,012:151
Horse	368	/	78	232:136
Leukemia	38	34	7,129	47:25
DLBCL	77	/	7,129	58:19
Prostate	102	/	12,600	52:50

14 AML), which are prepared under different experimental conditions and include 24 bone marrow and 10 blood sample specimens, and all samples contain 7,129 genes.

The second microarray data is diffuse large B-cell lymphomas (DLBCL), which contain 77 samples (Shipp et al. 2002). Fifty eight samples are diffuse large B-cell lymphomas and 19 samples are follicular lymphomas. All samples contain 7,129 genes.

The third microarray data is Prostate cancer (Singh et al. 2002). 102 samples contain 52 tumors and 50 normal prostate specimens. All samples contain 12,600 genes.

Leukemia, DLBCL and Prostate microarray data are publicly available at the Website (http://www.broadinstitute.org/cgi-bin/cancer/datasets.cgi).

4.2 Experiments on UCI data sets

Figure 1, 2 and 3 show experimental results on Horse, Sonar and Hypo datasets, respectively. X axis denotes the size of selected feature subset by information gain, SVM-RFE and SVM-RBF-RFE algorithms, and upper boundary is the feature number of experimental datasets. Each curve denotes the testing error rate on different feature subsets selected by information gain, SVM-RFE and SVM-RBF-RFE algorithms, respectively. Figure(a), Figure(b), Figure(c) and Figure(d) in each figure, respectively show classification error rate of SVML, SVM, 5NN and 10NN classifiers on different feature subsets.

For the Horse dataset with 78 features, In Fig. 1a, the classification error rate of SVML on the original samples without feature selection is 0.1930. The lowest classification error rate of SVML on the feature subsets selected by information gain, SVM-RFE and SVM-RBF-RFE algorithms are 0.1493, 0.1577 and **0.1548**, respectively, and the size of corresponding feature subsets are 53, 38 and **41**. In Fig. 1b, the classification error rate of SVM on the original samples without feature selection is 0.1632. The lowest classification error rate of SVM on the feature subsets selected by information gain, SVM-RFE and SVM-RBF-RFE algorithms are 0.1415, 0.1332 and **0.1386**, respectively, and the size of corresponding feature subsets are 45, 26 and **16**. From Fig. 1a and b, for SVML and SVM classifiers, we can see that the classification error rate is lower on most feature subsets selected by SVM-RBF-RFE than on the same size of feature subsets selected by information gain and SVM-RFE algorithm. In Fig. 1c, the classification error rate of 5NN on the original samples without feature selection is 0.1957, 5NN achieve the best performance on feature subset by SVM-RBF-RFE, and the lowest classification error rate is **0.1251** when the size of selected feature subset by SVM-RBF-RFE is **15**. In Fig. 1d, the classification error rate of 10NN on the original samples without feature selection is 0.1874. When the size of selected feature subset by SVM-RBF-RFE is **13**, the lowest classification error rate on 10NN is **0.1386**. From Fig. 1c and d, it is

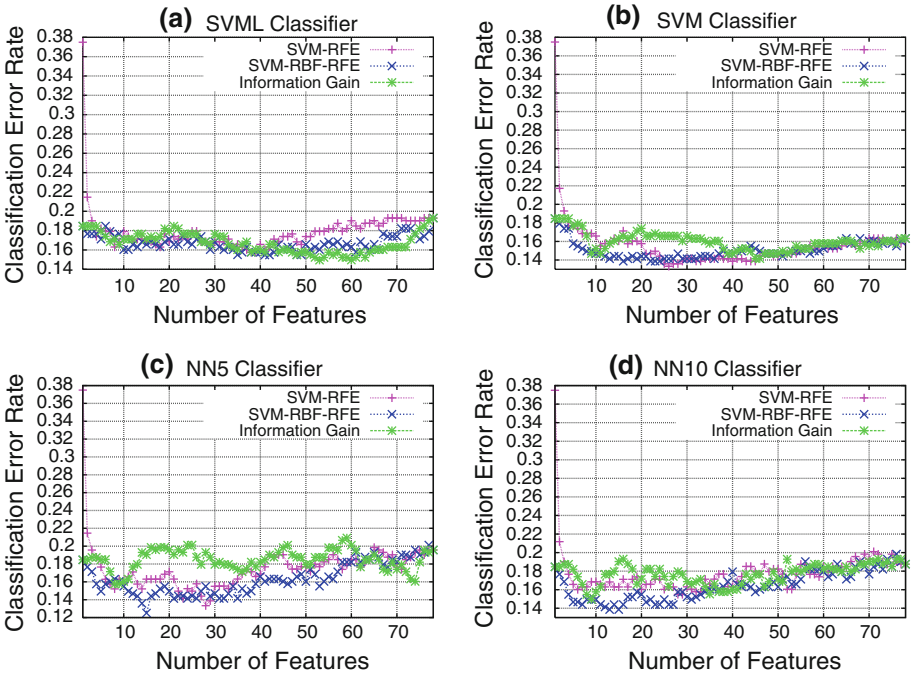

Fig. 1 Compare the effectiveness of information gain, SVM-RFE and SVM-RBF-RFE algorithms on the data set Horse. **a**, **b**, **c**, **d**, respectively, show SVML, SVM, 5NN and 10NN are used to evaluate feature subsets selected by three algorithms, respectively

clear that SVM-RBF-RFE select better feature subsets than information gain and SVM-RFE feature selection algorithms.

For the Sonar dataset with 42 features, in Fig. 2, when no feature selection algorithms is performed on the original samples, the classification error rate on SVML, SVM ,5NN and 10NN are 0.1481, 0.1383, 0.1769, and 0.1626, respectively. From Fig. 2, we can conclude SVML, SVM, 5NN and 10NN achieve the lowest classification error rate on feature subset selected by SVM-RBF-RFE algorithm, the lowest classification error rate are **0.1290**, **0.1240**, **0.1531** and **0.1383**, respectively, and the corresponding size of feature subset are 30, 34, 27 and 35 respectively. Moreover, it is clear that the classification error rate is lower on most feature subsets selected by SVM-RBF-RFE than on the same size of feature subsets selected by information gain and SVM-RFE algorithm.

For the Hypo dataset with 57 features, from Fig. 3, we conclude that information gain, SVM-RFE and SVM-RBF-RFE algorithms cannot improve the performance of SVML and SVM classifiers. In Fig. 3c, the classification error rate of 5NN on the original samples without feature selection is 0.0130, and 5NN achieves the best performance on feature subset selected by SVM-RBF-RFE, and the lowest classification error rate is **0.0073** when 5NN performs on feature subsets with size from 4 to 20. In Fig. 3d, the classification error rate of 10NN on the original samples without feature selection is 0.0133, and 10NN achieve the best performance on feature subset selected by SVM-RBF-RFE, and the lowest classification error rate is **0.0073** when 10NN performs on feature subsets with size from 4 to 5. Moreover, for 5NN and 10NN, it is clear that the classification error rate is lower on most feature subsets

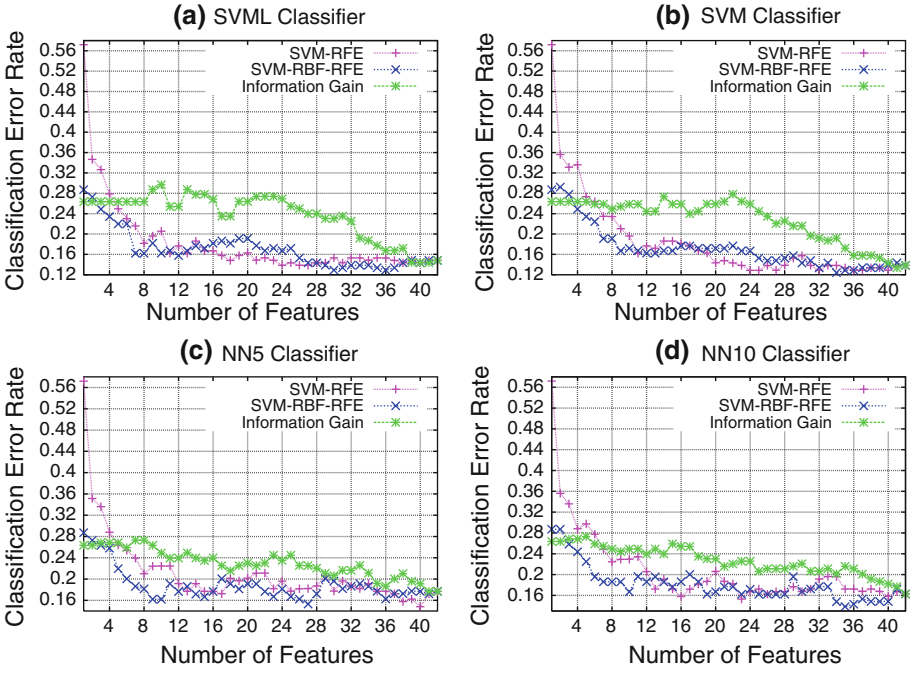

Fig. 2 Compare the effectiveness of information gain, SVM-RFE and SVM-RBF-RFE algorithms on the data set Sonar. (**a**, **b**, **c**, **d**), respectively, show SVML, SVM, 5NN and 10NN are used to evaluate feature subsets selected by three algorithms, respectively

selected by SVM-RBF-RFE than on the same size of feature subsets selected by information gain and SVM-RFE algorithms.

4.3 Experiments on microarray data

For microarray data, we need a threshold s defined by biologist to decide on the size of selected gene subset. The gene subset with the best performance is selected from s gene subsets with sizes from 1 to s. Generally speaking, it is difficult for biologist to decide precise threshold s. A large number of genes are very expensive and meaningless for biologist to research on informative genes which are strongly related with cancer. Too few genes are difficult to construct gene regulatory network which uncovers cancer regulatory mechanisms. In this paper, for Leukemia and DLBCL Microarray data with 7,129 genes in all samples, we set $s = 80$. For Prostate Microarray data with 12,600 genes, we set $s = 100$.

Figure 4 shows the experimental results on selected gene subsets with sizes from 1 to 80 on Leukemia. Figure 4a and b indicate classification results on classifier SVML and SVM, respectively. From Fig. 4a and b, we can conclude that testing error rate can achieve 0 on gene subsets selected by SVM-RFE and SVM-RBF-RFE algorithms, respectively, and the lowest error rate keeps stable on gene subsets with a definite size range. However, for information gain algorithm, the classification error rate achieves 0 just on gene subset with size equal to 22 in Fig. 4b. Figure 4c and d are classification results on 5NN and 10NN classifier, respectively, two classifiers perform better on some gene subsets selected by SVM-RBF-RFE than the same size of gene subsets selected by information gain or SVM-RFE algorithms,

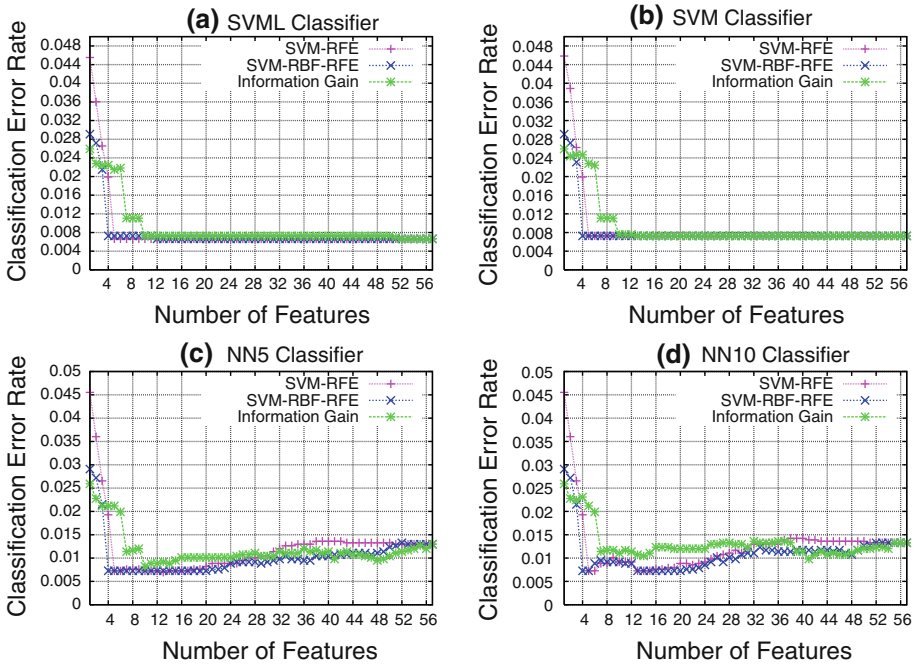

Fig. 3 Compare the effectiveness of information gain, SVM-RFE and SVM-RBF-RFE algorithms on the data set Hypo. **a**, **b**, **c**, **d**, respectively, show SVML, SVM, 5NN and 10NN are used to evaluate feature subsets selected by three algorithms, respectively

and worse than information gain or SVM-RFE algorithms in some gene subsets. Moreover, 5NN and 10NN can achieve the highest performance on gene subsets selected by 3 algorithms.

Figure 5 shows experimental results on gene subsets with sizes from 1 to 80 on DLBCL dataset. Figure 5a, b, c and d show classification error rate on SVML, SVM, 5NN and 10NN classifiers, respectively. From Fig. 5a, c and d, we can see that the classification error rate is lower on almost all of gene subsets selected by SVM-RBF-RFE than the same size of gene subsets selected by information gain and SVM-RFE algorithms. In Fig. 5b, when the size of gene subsets is less than 36, the curve for information gain is lower than the curve for SVM-RBF-RFE algorithms, but when the size of gene subsets is larger than 36, the classification error rate is lower on most gene subsets selected by SVM-RBF-RFE than the same size of selected gene subsets selected by information gain and SVM-RFE. From Fig. 5, we can conclude that four classifiers achieve the lowest classification error rate on gene subset selected by SVM-RBF-RFE algorithm, the lowest classification error rate are 0.0518, 0.0768, 0.0625 and 0.0768, respectively, and the corresponding size of gene subsets are 48, 44, 14 and 14, respectively.

Figure 6 gives the experimental results on gene subsets with sizes from 1 to 100 on Prostate data. Figure 6a and b are classification error rate on SVML and SVM, respectively. From Fig. 6a, we can see that SVML performs worse on gene subsets selected by SVM-RBF-RFE than on the same size of gene subsets selected by information gain and SVM-RFE. However, in Fig. 6b, the classification error rate is lower on almost all of gene subsets selected by SVM-RBF-RFE than on the same size of selected gene subsets by information gain and SVM-RFE. This may be the reason that selected genes overfit the classifier used by gene

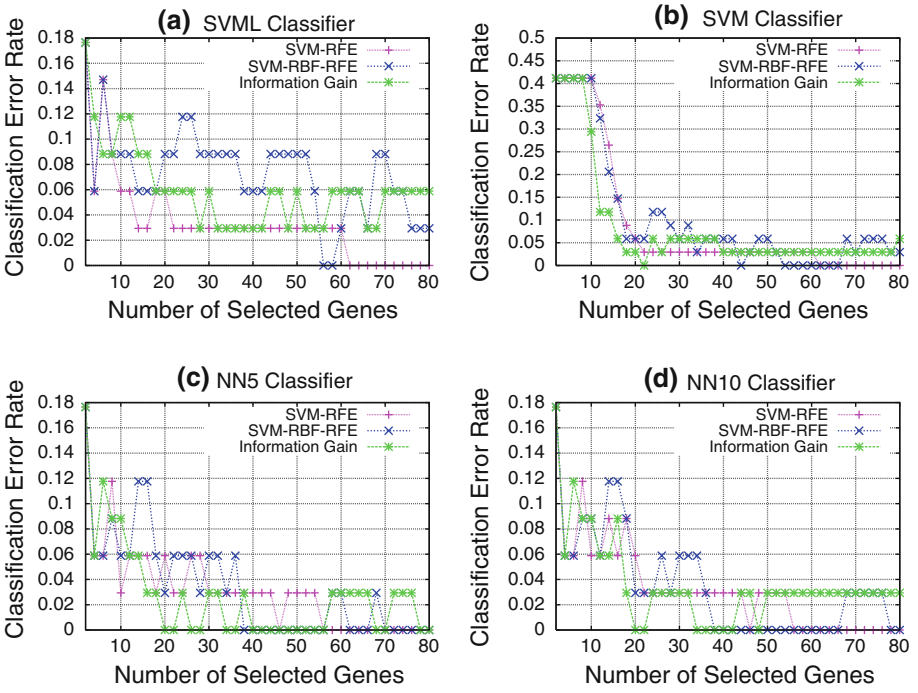

Fig. 4 Compare the effectiveness of information gain, SVM-RFE and SVM-RBF-RFE algorithms on the data set Leukemia. **a**, **b**, **c**, **d**, respectively, show SVML, SVM, 5NN and 10NN are used to evaluate feature subsets selected by three algorithms, respectively

selection algorithm (Niijima and Kuhara 2006; Deng et al. 2004). From Fig. 6c and d, which are classification results performed on 5NN and 10NN classifiers, respectively, we can conclude that classification performance on almost all of selected gene subsets by SVM-RBF-RFE is better than by information gain and SVM-RFE. Moreover, 5NN and 10NN achieve the lowest classification error rate on gene subset by SVM-RBF-RFE algorithm, the lowest classification error rate are 0.1482 and 0.1382, respectively, and the corresponding size of gene subsets are 42 and 32, respectively.

4.4 Discussion on important genes selected by SVM-RBF-RFE

In this section, take Leukemia Data set as example, we show the most important genes selected by SVM-RBF-RFE algorithm. Table 2 list rank, gene accession number(GAN) and description of the top 54 selected genes. Among the top 54 genes, 13 genes most highly correlated with ALL-AML class distinction have been shown by Golub et al. (1999). The 13 genes are U05259, Y12670, M31211, M23197, M92287, L47738, M27891, Y08612, X95735, M84526, M63138, M80254 and M81695. X70297 and D49950 were found to have strong prediction powers, and annotated by Onto-Express with significant biological processes and significant molecular funcitons, moreover, X70297 was annotated by Onto-Express with significant cellular components (Tang et al. 2007). D88422, M89957 and X03934 were identified to have a strong discrimination between ALL and AML (Ando and Iba 2004; Albrecht 2006). Many other genes of 54 have been reported as strong predictors in related literature,

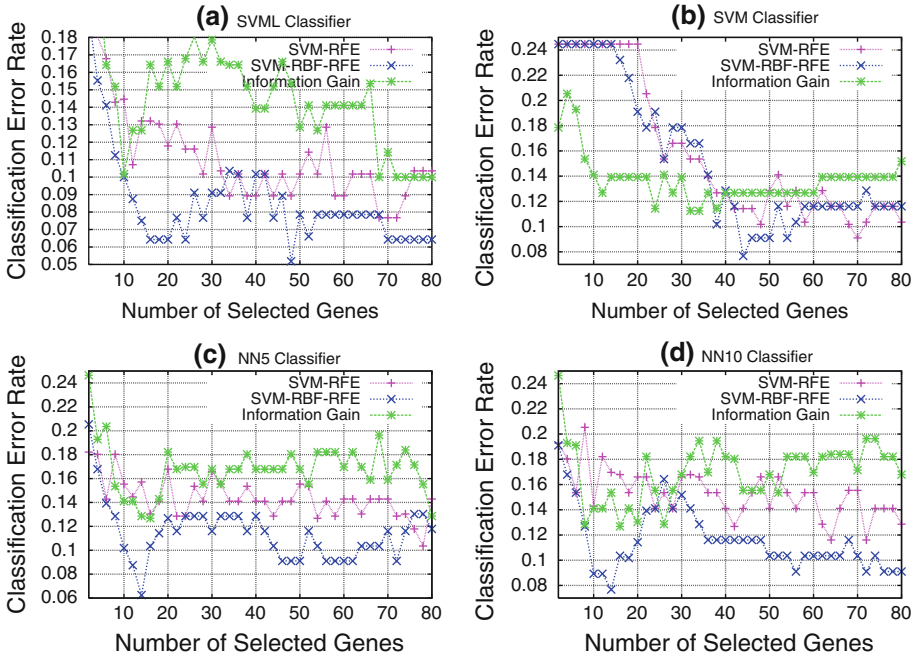

Fig. 5 Compare the effectiveness of information gain, SVM-RFE and SVM-RBF-RFE algorithms on the data set DLBCL. **a**, **b**, **c**, **d**, respectively, show SVML, SVM, 5NN and 10NN are used to evaluate feature subsets selected by three algorithms, respectively

The significant genes such as U46499 (Ding and Peng 2005), L05148 and M11722 (Ho et al. 2006), M77142, M31166, D14874 and M54995 (Draminski et al. 2008), M30703 (Tong et al. 2009), X16665 (Schoch et al. 2002), L41870 (Zhang et al. 2009), X59871 (Wang et al. 2006), J04615 (Tong et al. 2009), Y00339 (Tong et al. 2009).

5 Conclusions

For nonlinear SVM, feature ranking criterion is unknown and the weight vector cannot be computed explicitly. In this paper, we proposed a wrapper feature selection algorithm SVM-RBF-RFE, which expands nonlinear RBF kernel into its Maclaurin series in order to compute the weight of each feature. For 10NN classifier with a weak classification performance, SVM-RBF-RFE can improve the classification performance of 10NN used in our experiment, the error rate is from 0.1874 to 0.1386 on Horse dataset, and the error rate is from 0.1626 to 0.1383 on Sonar dataset, and the error rate is from 0.0133 to 0.0073 on Hypo dataset. In three microarray data, 10NN classifier also gives better performance on gene subset selected by SVM-RBF-RFE. SVM-RBF-RFE is compared with information gain which is a filter feature selection algorithm, and SVM-RFE which is wrapper feature selection algorithm. The results show SVM-RBF-RFE algorithm is very competitive over experimental datasets. Moreover, from Table 2, we can see that SVM-RBF-RFE can identify most significant genes which have been reported in the related research community. So we conclude that SVM-RBF-RFE algorithm is very effective on feature selection for SVM with RBF kernel.

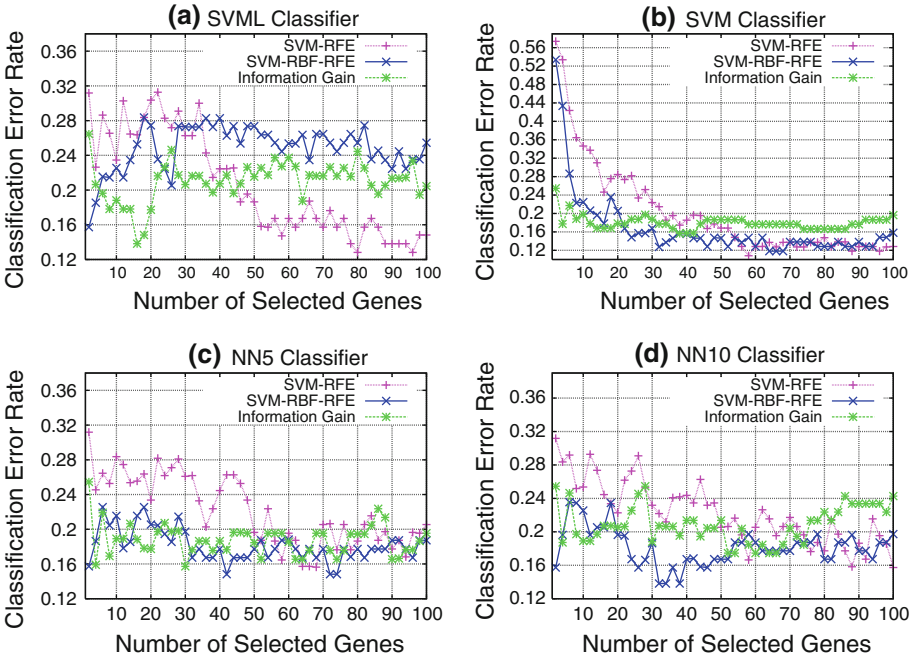

Fig. 6 Compare the effectiveness of information gain, SVM-RFE and SVM-RBF-RFE algorithms on the data set Prostate. **a**, **b**, **c**, **d**, respectively, show SVML, SVM, 5NN and 10NN are used to evaluate feature subsets selected by three algorithms, respectively

Table 2 The most important genes selected on the leukemia data set

Rank	GAN	Description of gene
1	X62535	DAGK1 Diacylglycerol kinase, alpha (80 kD)
2	U05259	MB-1 gene
3	D21262	KIAA0035 gene, partial cds
4	D88422	CYSTATIN A
5	U46499	GLUTATHIONE S-TRANSFERASE, MICROSOMAL
6	M77142	NUCLEOLYSIN TIA-1
7	Y12670	LEPR Leptin receptor
8	M80783	B12 protein mRNA
10	D49950	Liver mRNA for interferon-gamma inducing factor(IGIF)
11	M30703	Amphiregulin (AR) gene
12	M31211	MYL1 Myosin light chain (alkali)
13	X70297	CHRNA7 Cholinergic receptor, nicotinic, alpha polypeptide 7
14	U61836	Putative cyclin G1 interacting protein mRNA, partial sequence
15	M11722	Terminal transferase mRNA
16	X16665	HOXB2 Homeo box B2
17	M23197	CD33 CD33 antigen (differentiation antigen)
18	M92287	CCND3 Cyclin D3
19	L47738	Inducible protein mRNA
20	M27891	CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)

Table 2 continued

Rank	GAN	Description of gene
21	D14811	KIAA0110 gene
22	U41813	HOXA9 Homeo box A9
23	L00058	MYC V-myc avian myelocytomatosis viral oncogene homolog
24	D86983	KIAA0230 gene, partial cds
25	J03589	UBIQUITIN-LIKE PROTEIN GDX
26	S71043	Ig alpha 2=immunoglobulin A heavy chain allotype 2
27	M31166	PTX3 Pentaxin-related gene, rapidly induced by IL-1 beta
28	M83233	TCF12 Transcription factor 12
29	M95678	PLCB2 Phospholipase C, beta 2
30	Y08612	RABAPTIN-5 protein
31	U09578	MAPKAP kinase (3pK) mRNA
32	L41870	RB1 Retinoblastoma 1 (including osteosarcoma)
33	X95735	Zyxin
34	M19508	MPO from Human myeloperoxidase gene
35	D14874	ADM Adrenomedullin
36	M54995	PPBP Connective tissue activation peptide III
37	X03934	GB DEF = T-cell antigen receptor gene T3-delta
38	X59871	TCF7 Transcription factor 7 (T-cell specific)
39	U63825	Hepatitis delta antigen interacting protein A (dipA) mRNA
40	L07758	IEF SSP 9502 mRNA
41	U79285	GLYCYLPEPTIDE N-TETRADECANOYLTRANSFERASE
42	U04840	Onconeural ventral antigen-1 (Nova-1) mRNA
43	M84526	DF D component of complement (adipsin)
44	M63138	CTSD Cathepsin D (lysosomal aspartyl protease)
45	D13633	KIAA0008 gene
46	J04615	SNRPN Small nuclear ribonucleoprotein polypeptide N
47	D43948	KIAA0097 gene
48	M80254	PEPTIDYL-PROLYL CIS-TRANS ISOMERASE
49	D87469	KIAA0279 gene, partial cds
50	M89957	IGB Immunoglobulin-associated beta (B29)
51	L05148	Protein tyrosine kinase related mRNA sequence
52	M81695	ITGAX Integrin, alpha X
53	X82240	TCL1 gene (T cell leukemia)
54	Y00339	CA2 Carbonic anhydrase II

Acknowledgments This research is supported by the National Natural Science Foundation of China (60873196) and Chinese Universities Scientific Fund (QN2009092).

References

Albrecht A (2006) Stochastic local search for the feature set problem, with applications to microarray data. Appl Math Comput 183(2):1148–1164

Ando S, Iba H (2004) Classification of gene expression profile using combinatory method of evolutionary computation and machine learning. Genet Program Evol Mach 5:1573–7632

Bontempi G (2007) A blocking strategy to improve gene selection for classification of gene expression data. IEEE/ACM Trans Comput Biology Bioinform 4:293–300
Brank J, Grobelnik M, Milic-Frayling N, Mladenic D (2002) Feature selection using linear support vector machines. Technical Report, MSR-TR-2002-63, Microsoft Research, Microsoft Corporation
Burges C (1998) A tutorial on support vector machines for pattern recognition. Data Mining Knowl Discovery 2:121–167
Claeskens G, Croux C, Kerckhoven J (2008) An information criterion for variable selection in support vector machines. J Mach Learn Res 9:541–558
Cristianini N, Taylor J (2000) An introduction to support vector machines. Cambridge University Press, Cambridge
Deng L, Pei J, Ma J, Lee D (2004) A rank sum test method for informative gene discovery. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, pp 410–419
Ding C, Peng H (2005) Minimum redundancy feature selection from microarray gene expression data. J Bioinform Comput Biology 3(2):185–205
Ding Y, Wilkins D(2006)Improving the performance of SVM-RFE to select genes in microarray data. BMC Bioinform 7 (Suppl 2):S12. doi:10.1186/1471-2105-7-S2-S12
Draminski M, Rada-Iglesias A, Enroth S, Wadelius C, Koronacki J, Komorowski J (2008) Monte Carlo feature selection for supervised classification. Bioinformatics 24(1):110–117
Duan K, Rajapakse J (2004a) SVM-RFE peak selection for cancer classification with mass spectrometry data. In: Proceedings of the 3rd Asia-pacific bioinformatics conference, pp 191–200
Duan K, Rajapakse J (2004b) A variant of SVM-RFE for gene selection in cancer classification with expression data. In: Proceedings of IEEE symposium computational intelligence in bioinformatics and computational biology, pp 49–55
Duan K, Rajapakse J, Wang H, Azuaje F (2005) Multiple SVM-RFE for gene selection in cancer classification with expression data. IEEE Trans Nanobiosci 4(3):228–234
Elalami M (2009) A filter model for feature subset selection based on genetic algorithm. Knowledge-Based Syst 22:356–362
Estevez P, Tesmer M, Perez C, Zurada J (2009) Normalized mutual information feature selection. IEEE Trans Neural Netw 20:189–201
Fayyad U, Irani K (1993) Multi-interval discretization of continuous-valued attributes for classification learning. In: Proceedings of the 13th international joint conference on artificial intelligence, pp 1022–1027
Golub T et al (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286:531–537
Guyon W, Barnhill V (2002) Gene selection for cancer classification using support vector machines. Mach Learn 46:389–422
Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182
Ho S, Hsieh C, Chen H, Huang H (2006) Interpretable gene expression classifier with an accurate and compact fuzzy rule base for microarray data analysis. BioSystems 85:165–176
Huang J, Cai Y, Xu X (2007) A hybrid genetic algorithm for feature selection wrapper based on mutual information. Pattern Recogn Lett 28:1825–1844
Kohavi R, John G (1997) Wrappers for feature subset selection. Artif Intell 97(1-2):273–324
LeCun Y, Denker J, Solla S (1990) Optimal brain damage. Adv Neural Inform Process Syst II:598–605
Lee C, Lee G (2006) Information gain and divergence-based feature selection for machine learning-based text categorization. Inform Process Manage 42(1):155–165
Li F, Yang Y (2005) Analysis of recursive gene selection approaches from microarray data. Bioinformatics 21(19):3741–3747
Liu Q, Zhang Y, Hu Z (2007) Extracting positive and negative association classification rules from RBF kernel. In: 2007 International conference on convergence information technology. IEEE Computer Society, pp 1285–1291
Niijima S, Kuhara S (2006) Gene subset selection in kernel-induced feature space. Pattern Recogn Lett 27:1884–1892
Schoch C, Kohlmann A, Schnittger S et al (2002) Acute myeloid leukemias with reciprocal rearrangements can be distinguished by specific gene expression profiles. Proc Nat Acad Sci USA 99(15):10008–10013
Shipp M, Ross K, Tamayo P et al (2002) Diffuse large B-Cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nature Med 8(1):68–74
Silva P, Hashimoto R, Kim S et al (2005) Feature selection algorithms to find strong genes. Pattern Recogn Lett 26:1444–1453
Singh D, Febbo P et al (2002) Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 1:203–209

Sun Y (2007) Iterative RELIEF for feature weighting: algorithms, theories, and applications. In: IEEE transactions on pattern analysis and machine intelligence, vol. 29(6):1035–1051

Tang Y, Zhang Y, Huang Z (2007) Development of two-stage SVM-RFE gene selection strategy for microarray expression data analysis. IEEE/ACM Trans Comput Biol Bioinform 4(3):365–381

Tong D, Phalp K, Schierz A, Mintram R (2009) Innovative hybridisation of genetic algorithms and neural networks in detecting marker genes for leukaemia cancer. In: 4th IAPR international conference on pattern recognition in bioinformatics, Sheffield, 7–9 September 2009

Vapnik V (1998) Statistical learning theory. Wiley, New York

Wang Z, Palade V, Xu Y (2006) Neuro-fuzzy ensemble approach for microarray cancer gene expression data analysis. In: Proceedings of the second international symposium on evolving fuzzy system (EFS'06), IEEE Computational Intelligence Society 2006 , pp 241–246

Youn E, Jeong M (2009) Class dependent feature scaling method using naive Bayes classifier for text data mining. Pattern Recogn Lett 30:477–485

Zhang C, Lu X, Zhang X (2006) Significance of gene ranking for classification of microarray samples. IEEE/ACM Trans Comput Biology Bioinform 3(3):312–320

Zhang H, Song X, Wang H, Zhang X (2009) MIClique: an algorithm to identify differentially coexpressed disease gene subset from microarray data. J Biomed Biotechnol 2009. Article No.: 42524, doi:10.1155/2009/642524