Artificial Intelligence in Medicine (2005) **xxx**, xxx–xxx

ELSEVIER

ARTIFICIAL INTELLIGENCE IN MEDICINE

http://www.intl.elsevierhealth.com/journals/aiim

Gene extraction for cancer diagnosis by support vector machines—An improvement

Te Ming Huang, Vojislav Kecman *

School of Engineering, The University of Auckland, 20 Symonds Street, Private Box 92019, Auckland, New Zealand

Received 15 November 2004; received in revised form 4 January 2005; accepted 12 January 2005

KEYWORDS
Cancer diagnosis;
Support vector machines;
Gene selection;
Feature selection

Summary

Objective: To improve the performance of gene extraction for cancer diagnosis by recursive feature elimination with support vector machines (RFE-SVMs): A cancer diagnosis by using the DNA microarray data faces many challenges the most serious one being the presence of thousands of genes and only several dozens (at the best) of patient's samples. Thus, making any kind of classification in high-dimensional spaces from a limited number of data is both an extremely difficult and a prone to an error procedure. The improved RFE-SVMs is introduced and used here for an elimination of less relevant genes and just for a reduction of the overall number of genes used in a medical diagnostic.
Methods: The paper shows why and how the, usually neglected, penalty parameter *C* and some standard data preprocessing techniques (normalizing and scaling) influence classification results and the gene selection of RFE-SVMs. The gene selected by RFE-SVMs is compared with eight other gene selection algorithms implemented in the Rankgene software to investigate whether there is any consensus among the algorithms, so the scope of finding the right set of genes can be reduced.
Results: The improved RFE-SVMs is applied on the two benchmarking colon and lymphoma cancer data sets with various *C* parameters and different standard preprocessing techniques. Here, decreasing *C* leads to the smaller diagnosis error in comparisons to other known methods applied to the benchmarking data sets. With an appropriate parameter *C* and with a proper preprocessing procedure, the reduction in a diagnosis error is as high as 36%.
Conclusions: The results suggest that with a properly chosen parameter *C*, the extracted genes and the constructed classifier will ensure less overfitting of the training data leading to an increased accuracy in selecting relevant genes. Finally, comparison in gene ranking obtained by different algorithms shows that there is a significant consensus among the various algorithms as to which set of genes is relevant.

* Corresponding author. Tel.: +64 9 3737599x88178; fax: +64 9 373 7479.
E-mail address: v.kecman@auckland.ac.nz (V. Kecman).

0933-3657/$ – see front matter
doi:10.1016/j.artmed.2005.01.006

1. Introduction

Recently, huge advances in DNA microarrays have allowed the scientist to test thousands of genes in normal or tumor tissues on a single array and check whether those genes are active, hyperactive or silent. Therefore, there is an increasing interest in changing the criterion of tumor classification from morphologic to molecular [1]. In this perspective, the problem can be regarded as a classification problem in machine learning, in which the class of a tumor tissue with a feature vector $\mathbf{x}$ is determined by a classifier. Each dimension, or a feature, in $\mathbf{x}$ holds the expression value of a particular gene, which is obtained from DNA microarray experiment. The classifier is constructed by inputting l feature vectors of known tumor tissues into machine learning algorithms. To construct an accurate and reliable classifier with every gene included is not a straightforward task due to the fact that in the practice a number of tissue samples available for training is much less (a few dozens) than the number of features (a few thousands). In such a case, the classification space is extremely empty and it is difficult to construct a classifier that generalizes well. Therefore, there is a need to select a handful of most decisive genes in order to shrink the classification space and to improve the performance. Support vector machines (SVMs) are one of the latest developments in statistical learning theory and they have been shown to perform very well in many areas of biological analysis including evaluating microarray expression, detecting remote protein homologies, and recognizing translation initiation sites. More recently, SVMs-based feature selection algorithms dubbed, recursive feature elimination with support vector machines (RFE-SVMs) have been introduced and applied to a gene selection for a cancer classification. In this paper, we improve RFE-SVMs by working on two, often neglected, aspects of the algorithm implementation, which may affect the overall performance of the RFE-SVMs. They are the selection of a proper value for the hyperparameter C and the preprocessing of the microarray data. The C parameter plays an important role for SVMs in preventing an overfitting but its effects on the performance of RFE-SVMs are still unexplored. In terms of the microarray data preprocessing, we will only focus on the part after the gene expressions have been calculated for each array. We also try to investigate whether the gene selection algorithms can assist biologists in finding the right set of genes by comparing the genes selected by different types of algorithms implemented within the Rankgene software [2]. The paper is organized as follows: In Section 2, we review SVM-RFE and some prior work in this area. The results on the influence of the C parameter on a correct selection of relevant features are presented in Section 3. Section 4 shows the results of a genes' selection for two medical data sets (colon cancer and lymphoma data set) and discusses the preprocessing for microarrays. Finally, detailed comparisons between genes selected by RFE-SVMs and by eight other different approaches implemented within the Rankgene software are presented in Section 5.

2. Prior work

2.1. Support vector machines

The support vector machine classifier is based on the idea of margin maximization and it can be found by solving the following optimization problem [3].

$$\min \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\xi_i^2 \tag{1a}$$

$$\text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, l, \xi_i \geq 0 \tag{1b}$$

The decision function for linear SVMs is given as $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$. In this formulation, we have the training data set $\{\mathbf{x}_i, y_i\}$ $i = 1, \ldots, l$, where $\mathbf{x}_i \in \mathbb{R}^n$ are the training data points or the tissue sample vectors, y_i are the class labels, l is the number of samples and n is the number of genes measured in each sample. By solving the optimization problem (1) i.e., by finding the parameters $\mathbf{w}$ and b for a given training set, we are effectively designing a decision hyperplane over an n dimensional input space that produces the maximal margin in the space. Generally, the optimization problem (1) is solved by changing it into the dual problem below:

$$\max L_d(\alpha) = \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j \mathbf{x}_i^T\mathbf{x}_j \tag{2a}$$

$$\text{subject to } 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, l \tag{2b}$$

$$\sum_{i=1}^{l}\alpha_i y_i = 0 \tag{2c}$$

In this setting, one needs to maximize the dual objective function $L_d(\alpha)$ with respect to the dual variables α_i only. The equality constraint (2c) can be eliminated by adding a constant of 1 to all the entries of the kernel matrix as suggested in [4,5]. Hence, the dual objective becomes:

$$\max L_d(\alpha) = \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j (\mathbf{x}_i^T\mathbf{x}_j + \mathbf{1}) \tag{2d}$$

subject only to the box constraints $0 \leq \alpha_i \leq C$. The optimization problem can be solved by various

established techniques for solving general quadratic programming problems with inequality constraints.

2.2. Recursive feature elimination with support vector machines

The idea of using the maximal margin for gene selection was first proposed in [6] and it was achieved by coupling recursive features elimination with linear SVMs to find a subset of genes that maximizes the performance of the classifiers. In a linear SVM, the decision function is given as $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$ or $f(\mathbf{x}) = \sum_{k=1}^{n} w_k x_k + b$. For a given feature x_k, the size of the absolute value of its weight w_k shows how significantly does x_k contribute to the margin of the linear SVMs and to the output of a linear classifier. Hence, it is used as a feature-ranking coefficient in RFE-SVMs. In the original RFE-SVMs, the algorithm first starts constructing a linear SVMs classifier from the microarray data with n number of genes, then the gene with the smallest w_k^2 is removed and another classifier is trained on the remaining $n-1$ genes. This process is repeated until there is only one gene left. A gene ranking is produced at the end from the order of each gene being removed and the most relevant gene will be the one that is left at the end. However, for computational reasons, the algorithm is often implemented in such a way that several features are reduced at a time. In such a case, the method produces a feature subset ranking, as opposed to a feature ranking. Therefore, each feature in a subset may not be very relevant individually, and it is the feature subset that is optimal in some sense [6].

2.3. Selection bias and how to avoid it

As shown in [6], the leave-one-out error rate of RFE-SVMs can reach as low as zero percent with only 16 genes on the well-known colon cancer data set from [7]. However, as it was later pointed out in [1], the simulation results in [6] did not take selection bias into account. The leave-one-out error presented in [6] was measured using the classifier constructed from the subset of genes that were selected by RFE-SVMs using the complete data set. It gives too optimistic an assessment of the true prediction error, because the error is calculated internally. To take the selection bias into account, one needs to apply the gene selection and the learning algorithm on a training set to develop a classifier, and only then to perform an external cross-validation on a test set that had not been seen during the selection stage on a training data set. As shown in [1], the selection bias can be quite significant and the test error that is based on 50% training and 50% test can be as high as 17.5% for the colon cancer data set. Another important observation from [1] is that there are no significant improvements when the number of genes used for constructing the classifier is reduced: the prediction errors are relatively constant until approximately 64 or so genes. These observations indicate that the performance and the usefulness of RFE-SVMs may be in question. However, the influence of the parameter C was neglected in [1] which restricts the results obtained. As a major part of this work, we further investigate the problem by changing (reducing) the parameter C in RFE-SVMs, in order to explore and to show the full potentials of RFE-SVMs.

3. Influence of the parameter C in RFE-SVMs

The formulation in (1) is often referred to as the ''soft'' margin SVMs, because the margin is softened and the softness of the margin is controlled by the C parameter. If C is infinitely large, or larger then the biggest α_i calculated, the margin is basically 'hard' i.e., no points in the training data can be within or on the wrong side of the margin.

If C is smaller than the biggest original α_i, the margin is 'soft' one. As seen from (2b) all the $\alpha_j > C$ will be constrained to $\alpha_j = C$ and corresponding data points will be inside, or on the wrong side of, the margin. In the most of the work related to RFE-SVMs (e.g. [6,8]), the C parameter is set to a number that is sufficiently larger than the maximal α_i, i.e., a hard margin SVM is implemented within such an RFE-SVMs model. Consequently, it has been reported that the performance of RFE-SVMs is insensitive to the parameter C. However, Fig. 1 shows how C may influence the selection of more relevant features in a toy example where the two classes (stars * and pluses +) can be perfectly separated in a feature 2 direction only. In other words, the feature 1 is irrelevant for a perfect classification here. Note in the right hand side plot that a decrease in C, i.e., a constraining of the dual variables $\alpha_j = C$, leads to a moving of some data within the margin. However, at the same time this helps in detecting the more relevant feature, which is an input 2 here.

4. Gene selection for the colon cancer and the lymphoma data sets

4.1. Results for various C parameters

In this section, we present the selection of relevant genes for the two known data set in the gene

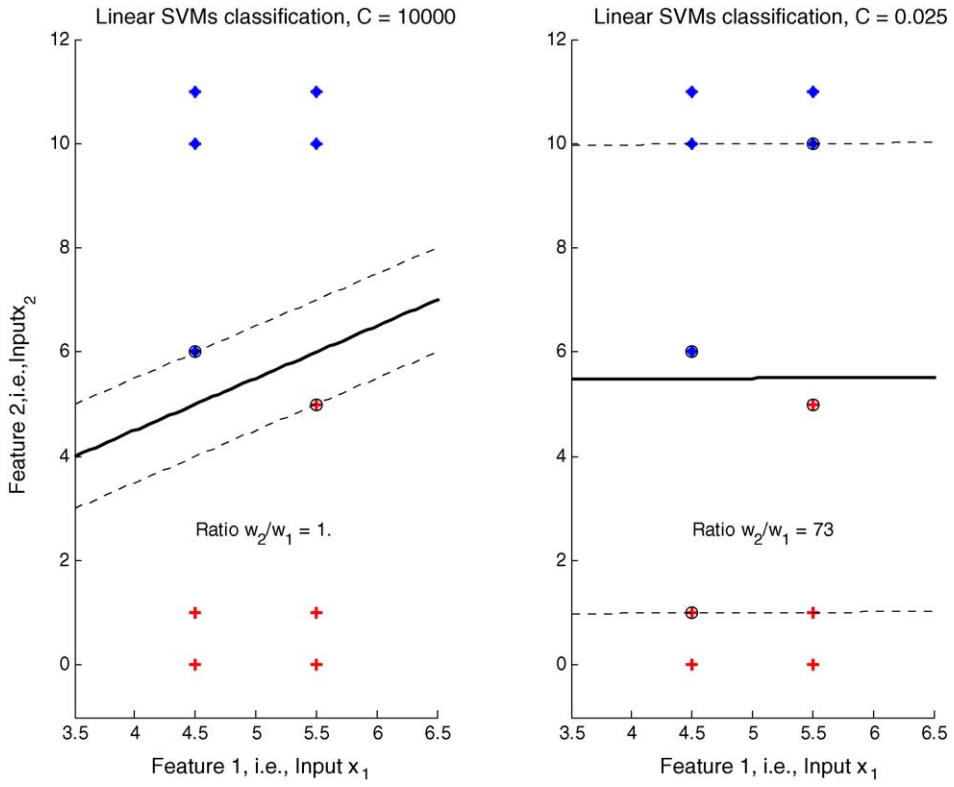

Figure 1 A toy example shows how C may be influential in a feature selection. With C equal to 10,000, both features seem to be equally important according to the feature ranking coefficients (namely, $w_1 = w_2$). With $C = 0.025$, a request for both a maximal and a 'hard' margin is relaxed and the feature 2 becomes more relevant than feature 1, because w_2 is larger than w_1 ($w_2/w_1 = 73$). While the former choice $C = 10{,}000$ enforces the largest margin and all data to be outside it, the later one ($C = 0.025$) enforces the feature 'relevance' and gives better separation boundary because the two classes can be perfectly separated in a feature 2 direction only.

microarray literature. The colon data set was analyzed initially in [7] and the lymphoma data was first analyzed in [9]. The colon data set is composed of 62 samples (22 normal and 40 cancerous) with 2000 genes' expressions in each sample. The training and the test sets are obtained by splitting the dataset into two equal groups of 31 elements, while ensuring each group has 11 normal and 20 cancerous tissues. The RFE-SVMs is only applied on the training set to select relevant genes and to develop classifiers, and then the classifiers are used on the test set to estimate the error rate of the algorithms. Fifty trails were carried out with random split for estimating the test error rate. A simple preprocessing step is performed on the colon data set to make sure each sample is treated equally and to reduce the array effects. Standardization is achieved by normalizing each sample to the one with zero mean and with a standard deviation of one. To speed up the gene selection process, 25% of the genes are removed at each step until less than 100 genes remained still to be ranked. Then the genes are removed one at a time. The simulation results for the colon data set are shown in Fig. 2.

The Ambroise and McLachlan's curve in Fig. 2 is directly taken from [1], and it is unclear what C value is used in this paper. By comparing the error rates for various C parameters, it is clear that changing the parameter C has significant influence on the performance of RFE-SVMs in this data set. The error rate is reduced from previously 17.5% as reported in [1] to 11.16% (a reduction of 36%) when C is equal to 0.005. For $C = 0.01$, the gene selection procedure improves the performance of the classifier: this trend can be observed by looking at the error rate reduction from initially around 15% at 2000 genes to 11.9% with 2^6 genes. Similar trend can be observed when $C = 0.005$, but the error rate reduction is not as significant as in the previous case. This is due to the fact that the error rate of the linear SVMs with $C = 0.005$ is already low, when all the genes are used. This also demonstrates that tuning the C parameter can reduce the amount of over-fitting on the training data even in such a high dimensional space with small number of samples. A preliminary comparison between RFE-SVMs and the well-known nearest shrunken centroid from [10] is made on the colon cancer data set and the lowest

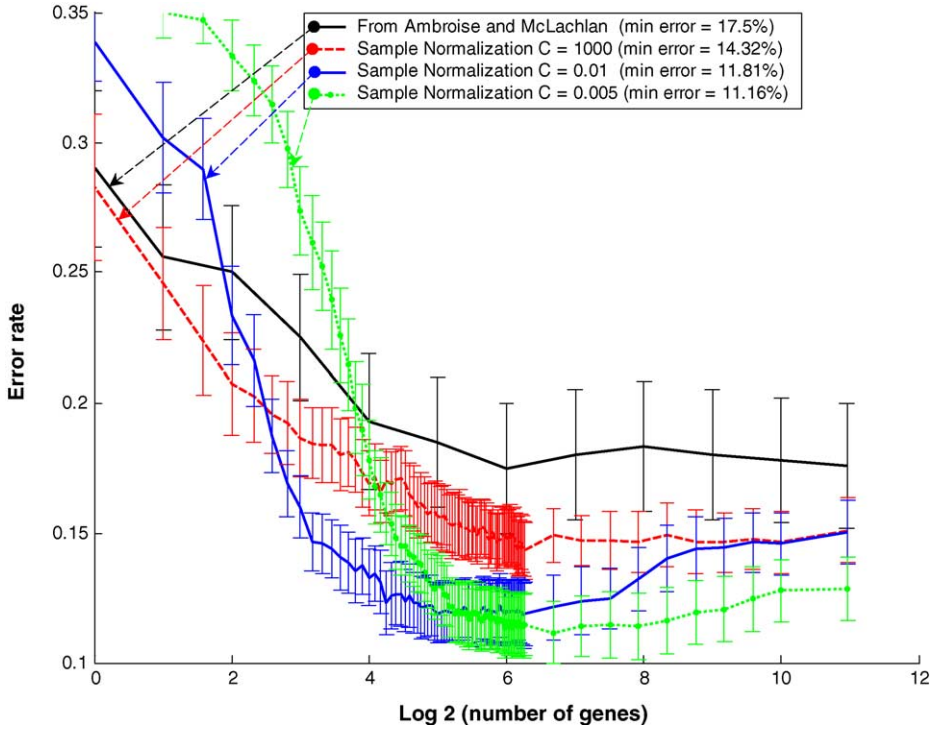

Figure 2 Simulation result on the colon cancer data set with various C parameters. The error bar represents the 95% confident interval.

error rate presented here from RFE-SVMs (11.16% at $C = 0.005$) is lower than the nearest shrunken centroid method (13.45%). Note that the minimal error rate here is 11.16% and this coincides with the suggestion in [1] that there are some wrongly labeled data in the training data set. This makes colon cancer data more difficult to classify than the lymphoma data set presented next.

In terms of the gene selected, comparison can be made between Tables 1 and 4a to show the difference between $C = 0.005$ and 0.01. Except for the first gene in the table, the rank of all the other genes is different. However, seven genes are selected into the top 10 genes by both settings of RFE-SVMs.

The second benchmarking data set is the lymphoma data set first analyzed in [9]. This version of the lymphoma data set is also same as the one used in [11]. It is composed of 62 samples (42 diffuse large B-cell (DLCL), 8 follicular lymphoma (FL) and 12 chronic lymphocytic lymphoma (CL)) with 4026 genes expressions in each sample. This is a multiclass problem and the data set is spilt into 31 data pairs for training and 31 pairs for testing. Each part has 21 samples belonging to DLCL, four belonging to FL and six belonging to CLL. The simulation results

Table 1 Colon cancer data, RFE-SVMs' top 10 genes for $C = 0.005$

Ranking	GAN[a]	Description
1	J02854	Myosin regulatory light chain 2
2	X86693	*Homo sapiens* mRNA for hevin like protein
3	H06524	Gelsolin precursor, plasma (human)
4	M36634	Human vasoactive intestinal peptide (VIP) mRNA, complete cds
5	M76378	Human cysteine-rich protein (CRP) gene, exons 5 and 6
6	M63391	Human desmin gene, complete cds
7	R87126	Myosin heavy chain, nonmuscle (*Gallus gallus*)
8	T92451	Tropomyosin, fibroblast and epithelial muscle-type (human)
9	T47377	S-100p protein (human)
10	Z50753	*H. sapiens* mRNA for GCAP-II/uroguanylin precursor

Genes are ranked in order of decreasing importance.
[a] Gene accession number.

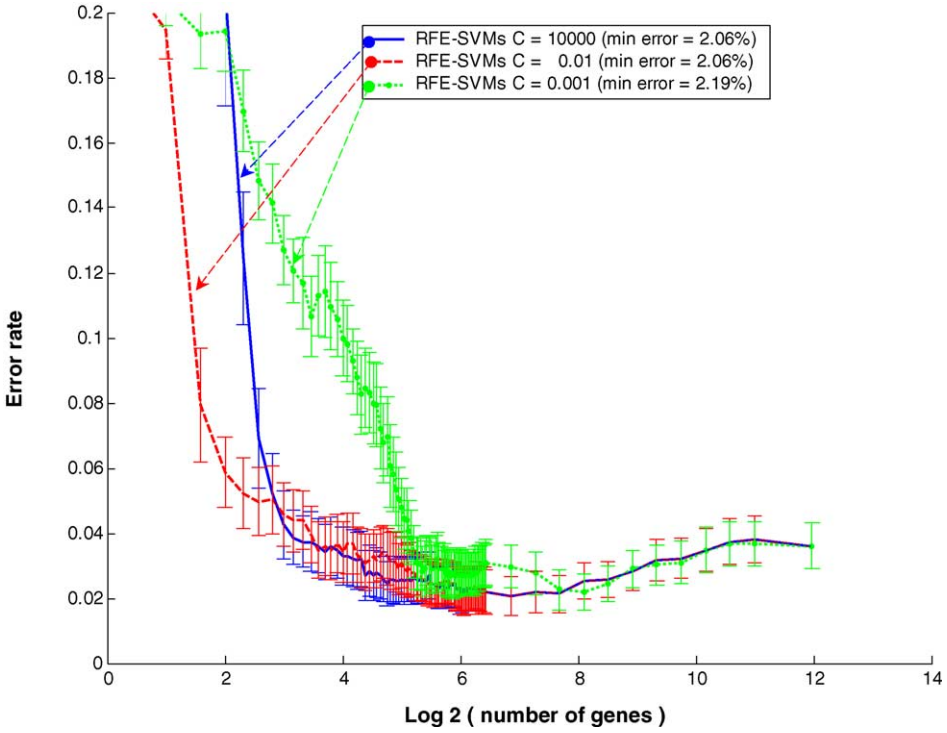

Figure 3 Simulation results on the lymphoma data set with various *C* parameters.

are shown in Fig. 3. As shown in Fig. 3, there is not too much difference in terms of the lowest error rate between the larger *C* values and the smaller ones, and all the models have approximately 2% error rate. In this case, the choice of *C* parameter does not influence the performance very much. This may be due to the fact that this data set is an easy one and that it may be a relatively simple problem to perform the separation between different classes disregarding the true value of parameter *C*. The top 10 genes for the lymphoma data set selected by RFE-SVMs are listed in Table 2.

4.2. Simulation results with different preprocessing procedure

As mentioned previously, we are also interested in the preprocessing of gene expressions after they had been obtained via procedures such as the one implemented in affymetrix microarray suit (MAS) for affymetrix array. A very common preprocessing step before inputting the training data into various machine learning algorithms, or statistical methods, is to normalize each feature vector. In this way each feature (or gene here) has mean of zero and

Table 2 Lymphoma data, RFE-SVMs' top 10 genes for $C = 0.01$

Ranking	GAN[a]	Description
1	GENE1636X	Osteonectin = SPARC = basement membrane protein; clone = 487878
2	GENE1610X	Unknown; clone = 711756
3	GENE1637X	Fibronectin 1; clone = 139009
4	GENE1635X	Fibronectin 1; clone = 139009
5	GENE2328X	FGR tyrosine kinase; clone = 347751
6	GENE263X	Similar to HuEMAP = homolog of echinoderm microtubule associated protein (EMAP); clone = 1354294
7	GENE1648X	Cathepsin B; clone = 261517
8	GENE1609X	Mig = humig = chemokine targeting T cells; clone = 8
9	GENE3320X	Similar to HuEMAP = homolog of echinoderm microtubule associated protein clone = 1354294
10	GENE1641X	Cathepsin B; clone = 261517

Genes are ranked in order of decreasing importance.
[a] Gene accession number.

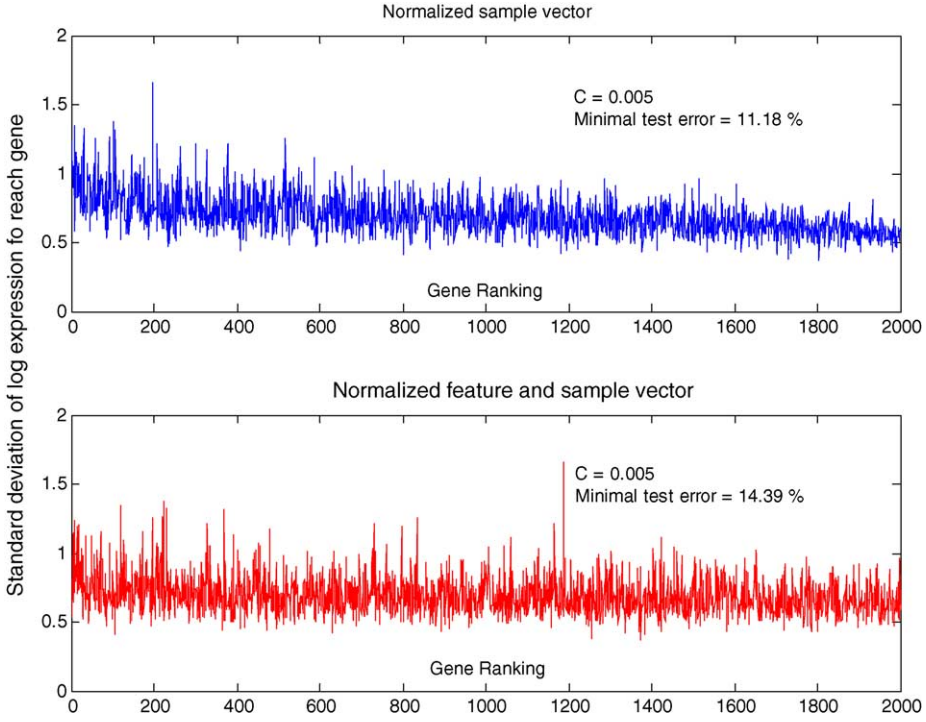

Figure 4 Effect of different preprocessing procedures on the gene ranking for colon cancer data set. The genes are ranked in the order of decreasing importance. The gene with ranking 1 is the most relevant gene.

standard deviation of one in the data set. In a microarray analysis, it is common to normalize the sample vector so that the array effect is minimized i.e., each sample has mean of zero and standard deviation of one. In this section, we investigate two straightforward preprocessing procedures and compare their results using the colon cancer data set. In the first preprocessing procedure, we first take log of all the expressions to obtain the log expressions and then we normalize all the sample vectors in the data set in order to have the zero mean and standard deviation of one. This procedure is referred to as the sample normalization. For the second preprocessing procedure, we perform the same sample normalization to the complete data first and then a feature normalization step to all the features in the data set follows. In order to perform feature normalization without the selection bias for a given feature, we first subtract the mean expression of the feature (calculated from the training data) from all the expression values of the feature in the complete data set. Then we divide all the expression values of the feature in the complete data set by the standard deviation of the feature, which is also calculated from the training data. Consequently, the mean and the standard deviation of the feature within the training data will be zero after the feature normalization, but the mean and the standard deviation of the feature for the complete data set will not equal zero. The second procedure, which will be referred to as the sample and feature normalization, is very similar to the one in [6] except that we did not pass the data through a squashing function. To test these two procedures, we use the same setting as in the previous section i.e., 50 random splits of 50% training data and 50% testing ones.

In Fig. 4, we compare the gene rankings when two different preprocessing procedures are implemented. In the figure, the standard deviation of each gene's log expression (calculated from the complete data set without sample or feature normalization) is plotted on the vertical axis, and their respective gene ranking from RFE-SVMs is plotted on the horizontal axis.

The top graph shows the result with the first preprocessing procedure. Interestingly, the gene with higher standard deviation tends to have higher ranking. This trend suggests that RFE-SVMs with sample normalization will likely pick up genes with expression that vary more across the samples. This fits well with the assumption that a gene is less relevant if its expression does not vary much across the complete data set. Such a general trend cannot be observed in the bottom graph (where a sample and the feature normalization are applied) and there is no connection between the standard deviation of the gene and gene ranking. This phenomenon may be due to the fact that the feature normalization step in the second preprocessing procedure will ensure that each gene has the same standard

Table 3 Colon cancer data, RFE-SVMs' top 10 genes for $C = 0.005$ for both sample and feature vectors normalization

Ranking	GAN[a]	Description
1	R39681	Eukaryotic initiation factor 4 gamma (*H. sapiens*)
2	R87126	Myosin heavy chain, nonmuscle (*G. gallus*)
3	H20709	Myosin light chain alkali, smooth-muscle isoform (human)
4	H06524	Gelsolin precursor, plasma (human)
5	H49870	Mad protein (*H. sapiens*)
6	R88740	ATP synthase coupling factor 6, mitochondrial precursor (human)
7	J02854	Myosin regulatory light chain 2, smooth muscle isoform (human, contains element TAR 1 repetitive element)
8	M63391	Human desmin gene, complete cds
9	H09273	Putative 118.2 kd transcriptional regulatory protein in ACS1-PTA1 intergenic region (*Saccharomyces cerevisiae*)
10	X12369	Tropomyosin alpha chain, smooth muscle (human)

Genes are ranked in order of decreasing importance.
[a] Gene accession number.

deviation. Hence, a gene with higher standard deviation originally will no longer be advantageous over a gene having a smaller standard deviation. In Table 3, the top 10 genes selected by $C = 0.005$ for both sample and feature vectors normalization are presented. By comparing Tables 1 and 3, it is clear that the two preprocessing steps discussed here produced two different rankings and only five of the top 10 genes are selected by both preprocessing steps. This supports the trend that is observed in Fig. 4.

A general practice for producing good results with SVMs is to normalize each input (feature) to the one with mean zero and standard deviation of one as in the feature normalization step. However, in this case, this simple rule does not perform as well as expected: the error rate of applying both sample and feature normalization is higher than when only the sample normalization is performed. This phenomenon may be due to the fact that the feature normalization step in the second preprocessing procedure filters out the information about the spread of the expression for each gene as discussed previously and this information is helpful for selecting the relevant gene and classification.

5. Comparison of gene ranking with different algorithms

Now, we compare the gene ranking from RFE-SVMs with eight different algorithms implemented within the Rankgene software [2]. (Rankgene incorporates eight different methods to produce genes' ranking, including information gain, twoing rule, sum minority, max minority, gini index, sum of variances, *t*-statistic and one-dimensional SVMs.) We combine all the eight different genes' rankings from Rankgene into a single ranking and compared it with an RFE-SVMs ranking. Table 4a shows the top 10 genes from the RFE-SVMs genes' ranking for $C = 0.01$. Genes printed in italics have been selected by the Rankgene too.

Although the genes have been ranked differently, six out of the top 10 genes selected by RFE-SVMs have also been selected within the top 10 genes by the Rankgene package as shown in Table 4a.

Table 4a Colon cancer data, RFE-SVMs' top 10 genes with $C = 0.01$

Ranking	GAN[a]	Description
1	J02854[b]	Myosin regulatory light chain 2
2	H06524	Gelsolin precursor, plasma (human)
3	R87126[b]	Myosin heavy chain, nonmuscle (*G. gallus*)
4	M63391[b]	Human desmin gene, complete cds
5	X86693	H. sapiens mRNA for hevin like protein
6	M76378[b]	Human cysteine-rich protein (CRP) gene, exons 5 and 6
7	T92451[b]	Tropomyosin, fibroblast and epithelial muscle-type (human)
8	Z50753[b]	H.sapiens mRNA for GCAP-II/uroguanylin precursor
9	M31994	Human cytosolic aldehyde dehydrogenase (ALDH1) gene, exon 13
10	M36634	Human vasoactive intestinal peptide (VIP) mRNA, complete cds

Genes are ranked in order of decreasing importance.
[a] Gene accession number.
[b] The genes names have been picked up by the Rankgene software within its top 10 genes.

Table 5 The 10 genes selected by RFE-SVMs and eight other different methods implemented in Rankgene software within their respective top 100 genes

Avg ranking[a]	GAN[b]	Description
2.3	M76378	Human cysteine-rich protein (CRP) gene, exons 5 and 6
13.5	M63391	Human desmin gene, complete cds
27.2	Z50753	*H. sapiens* mRNA for GCAP-II/uroguanylin precursor
28.1	T60155	Actin, aortic smooth muscle (human)
5.7	R87126	Myosin heavy chain, nonmuscle (*G. gallus*)
7.5	M76378	Human cysteine-rich protein (CRP) gene, exons 5 and 6
15.7	T92451	Tropomyosin, fibroblast and epithelial muscle-type (human)
21.3	H43887	Complement factor D precursor (*H. sapiens*)
6.9	J02854	Myosin regulator light chain 2, smooth muscle isoform (human)
31.3	M36634	Human vasoactive intestinal peptide (VIP) mRNA, complete cds

[a] Average ranking of genes in nine different methods.
[b] Gene accession number.

Table 4b Colon cancer data. Top seven genes listed in [6]

Ranking	GAN[a]	Description
1	H08393	Collagen alpha 2(XI) chain (*H. sapiens*)
2	M59040	Human cell adhesion molecule (CD44) mRNA, complete cds
3	T94579	Human chitotriosidase precursor mRNA, complete cds
4	H81558	Procyclic form specific polypeptide B1-alpha precursor (*Trypanosoma brucei brucei*)
5	R88740	ATP synthase coupling factor 6, mitochondrial precursor (human)
6	T62947	60s ribosomal protein L25 (*Arabidopsis thaliana*)
7	H64807	Placental folate transporter (*H. sapiens*)

[a] Gene accession number.

This means that there is still a great deal of consensus on the genes' relevancy obtained by different ranking methods. This may help in narrowing down the scope of the search for the most relevant set of genes.

On the other hand, in Table 4b the top seven genes listed in [6] are shown and there is only one gene overlapped with the top 10 genes from the Rankgene package. Also, only the gene ATP synthase listed in [6] was selected by RFE-SVMs method as shown in Table 3. Furthermore, we found that only 10 genes have been selected by all nine methods (namely by the RFE and by eight different methods implemented in the Rankgene software) within their respective top 100 genes. They are listed in Table 5. The average ranking of these 10 genes shows that only the top ranked genes are overlapped and that they are more likely to be selected by all the different methods. This strongly suggests that the 10 listed genes may be very relevant in an investigation of a colon cancer.

6. Conclusions

We presented the performance of improved RFE-SVMs algorithm for genes extraction in diagnosing two different types of cancers. Why and how is this improvement achieved by using different values for the C parameter was discussed in detail. With a properly chosen parameter C, the extracted genes and the constructed classifier will ensure less overfitting of the training data leading to an increased accuracy in selecting relevant genes. These effects are more remarkable in a more difficult data set such as colon cancer data. The simulation results also suggest that the classifier performs better in the reduced gene spaces selected by RFE-SVMs than in the complete 2000 dimensional gene space. This is a good indication that RFE-SVMs can select relevant genes, which can help in the diagnosis and in the biological analysis of both the genes' relevancy and their function. In terms of the raw data preprocessing, it is clear that the performance of RFE-SVMs can also vary with different preprocessing steps. In the colon cancer data set, we found that normalizing only the sample vector produces better result. The comparison of genes' rankings obtained by the RFE-SVMs and by the Rankgene software package (which implements eight different methods for a gene selection) shows that there is a great deal of consensus on genes' relevancy. This may help in narrowing down the scope of search for the set of 'optimal' genes using machine-learning techniques. Finally, the results in this work are developed from a more machine learning and data

mining perspective, meaning unrelated to any valuable insight from a biology and medicine. Thus, there is a need for a tighter cooperation between the biologists and/or medical experts and data miners in all the future investigations. The basic result of this synergy should be giving the meaning to all the findings presented and in this way ensuring a more reliable guidance for the future research.

References

[1] Ambroise C, McLachlan GJ. Selection bias in gene extraction on the basis of microarray gene-expression data. In: Fienberg SE, editor. Proceedings of the National Academy of Science, USA 2002. 2002. p. 6562–6.
[2] Su Y, Murali TM, Pavlovic V, Schaffer M, Kasif S. Rankgene: a program to rank genes from expression data. Computational Genomics Laboratory, BU, Boston, 2002 (available at: http://genomics10.bu.edu/yangsu/rankgene/) (accessed: 14 January 2005).
[3] Kecman V. Learning and Soft Computing. Cambridge, MA: The MIT Press, 2001.
[4] Huang TM, Kecman V. Bias term b in SVMs again. In: Verleysen M, editor. Proceedings of the ESANN 2004, 12th European Symposium on Artificial Neural Networks. 2004. p. 441–8.
[5] Kecman V, Vogt M, Huang TM. On the equality of kernel adatron and sequential minimal optimization in classification and regression tasks and alike algorithms for kernel machines. In: Verleysen M, editor. Proceedings of the ESANN 2003, Eleventh European Symposium on Artificial Neural Networks, 2003 (available at: http://www.support-vector.ws) (accessed: 14 January 2005).
[6] Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. Mach Learn 2002;46:389–422.
[7] Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, Mack D, et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc Natl Acad Sci USA 1999;96: 6745–50.
[8] Rakotomamonjy A. Variable selection using SVM-based criteria. JMLR Special Issue 2003;1357–70. variable and feature selection.
[9] Alizadeh A, Eisen M, Davis R, Ma C, Lossos I, Rosenwald A, et al. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 2000;403:503–11.
[10] Tibshirani R, Hastie T, Narasimhan B, Chu G. Class prediction by nearest shrunken centroids, with applications to DNA microarrays. Stat Sci 2003;18:104–17.
[11] Chu F, Wang L. Gene expression data analysis using support vector machines. In: Donald C, Wunsch II., editors. Proceedings of the 2003 IEEE International Joint Conference on Neural Networks. 2003. p. 2268–71.