LS Bound based gene selection for DNA microarray data

Xin Zhou\textsuperscript{1,2} and K. Z. Mao\textsuperscript{1,}\textsuperscript{*}

\textsuperscript{1}School of Electrical & Electronic Engineering, \textsuperscript{2}Bioinformatics Research Centre, Nanyang Technological University, Nanyang avenue, Singapore 639798

ABSTRACT
Motivation: One problem with discriminant analysis of DNA microarray data is that each sample is represented by a quite large number of genes, and many of them are irrelevant, insignificant or redundant to the discriminant problem at hand. Methods for selecting important genes are therefore of much significance in microarray data analysis. In the present study, a new criterion, called LS Bound measure, is proposed to address the gene selection problem. The LS Bound measure is derived from leave-one-out procedure of LS-SVMs, and as the upper bound for leave-one-out classification results it reflects to some extent the generalization performance of gene subsets.

Results: We applied this LS Bound measure for gene selection on two benchmark microarray datasets: colon cancer (Alon et al., 1999), and leukaemia (Golub et al., 1999). We also compared the LS Bound measure with other evaluation criteria, including the well-known Fisher’s ratio and Mahalanobis class separability measure, and other published gene selection algorithms, including Weighting factor and SVM RFE. The strength of the LS Bound measure is that it provides gene subsets leading to more accurate classification results than the filter method while its computational complexity is at the level of the filter method.

Availability: A companion website can be accessed at http://www.ntu.edu.sg/home/5/pg02776030/lsbound/. The website contains: 1. the source code of the gene selection algorithm; 2. the complete set of tables and figures regarding the experimental study; 3. proof of the inequality (9).

Contact: ekzmao@ntu.edu.sg

1 INTRODUCTION
In recent years, research focus in molecular biology and genetics has shifted from the study of individual genes to the exploration of the entire genome. For example, the recently developed gene expression microarray technique measures the expression levels of thousands of genes in a single experiment. This large amount of data is something of a gold mine, from which a number of things can be found.

Gene expression microarray data have been explored in a variety of ways including sample discriminant analysis and clustering, gene clustering and gene selection and many others. Among these issues, gene selection for discriminant analysis is of particular interest to us. From the viewpoint of discriminant analysis, we have a few reasons to perform gene selection. Firstly, among the large set of genes many might be irrelevant, insignificant or redundant to a specific discriminant problem. Studies have shown that a small subset of genes might be sufficient for a particular biological problem (Golub et al., 1999). Secondly, gene selection reduces data volume and makes microarray data easier to handle and analyze. Thirdly, reducing the number of genes relieves the demand for a large number of training samples because the performance of a pattern classifier partly depends on the ratio between the number of samples to the number of features. The collection of a quite large number of samples in microarray technique is expensive, time-consuming and even impossible. From the viewpoint of biologists, the importance of gene selection lies in its contribution to understanding diseases and functions of particular genes, and designing microarray experiments for clinical diagnosis and prognosis purpose.

In the context of gene expression data analysis, several gene selection approaches were published. Golub et al. (1999) and Furey et al. (2000) employed an individual gene ranking score, Weighting factor, to perform gene selection prior to classification. Li et al. (2001) proposed a “GA/KNN” method for gene assessment and sample classification. The main idea of GA/KNN is to find a huge number of optimal or near-optimal subsets and to assess the importance of genes for classification by examining the frequency of gene memberships in those subsets. Guyon et al. (2002) introduced a top-down recursive feature elimination (RFE) algorithm, in which features are successively eliminated during training of a sequence of SVM classifiers.
If put in the context of pattern classification, gene selection can be solved as a feature selection problem. In general, a feature selection algorithm mainly consists of two basic components: search procedure and evaluation criterion (Dash and Liu, 1997). The search procedure generates candidate feature subsets for evaluation. In the search procedure, candidate feature subsets can be generated either sequentially or randomly. The well-known sequential forward selection (SFS) starts from an empty set and iteratively adds features, while the sequential backward elimination (SBE) starts from the full feature set and iteratively delete features. The evaluation criterion measures the goodness of the candidate feature subsets generated by the search procedure. At each step of the iterative procedure, the feature that leads to the greatest improvement after addition or the least degradation after deletion is selected.

Generally, feature selection can be performed in two ways: the filter and the wrapper methods (Devijver and Kittler, 1982). The main difference between the two methods lies in the evaluation criterion. The filter method employs intrinsic properties of data, such as Mahalanobis class separability measure (Devijver and Kittler, 1982), as the criterion for feature subset evaluation, while the wrapper method evaluates feature subsets based on the performance of the classifier, such as classification errors. In most pattern recognition applications, the wrapper method outperforms the filter method. Several authors have employed wrapper methods in gene selection of microarray data (Inza et al., 2002; Xiong et al., 2001) and received satisfactory performance. However, the accuracy of wrapper methods is coupled with intensive computations. In contrast, the filter method is computationally efficient, and this makes the filter method very suitable for gene selection in high dimensional gene expression data.

In our work we suggested a new filter-like evaluation criterion, called LS Bound measure, for gene selection. The new criterion has the advantages of both filter and wrapper methods. Firstly, the criterion is derived from the leave-one-out cross validation (LOOCV) procedure of Least Squares Support Vector Machines (LS-SVM), and is closely related to an upper bound of LOOCV classification results. Therefore the criterion provides genes leading to accurate classification. Secondly, the estimation of the upper bound implicitly involves the training of the classifier only once, without repeated use of cross validation. As a result, the computational complexity is significantly reduced compared with the classical wrapper method. Our experiments showed that the computational complexity of our method is at the level of the filter method.

The paper is organized as follows. The Least Square SVM is first briefly reviewed, and a new gene selection criterion, LS Bound measure, based on the upper bound for LOOCV of LS-SVMs is then proposed. The performance of the criterion is finally tested with two benchmark microarray datasets, i.e. the colon cancer dataset (Alon et al., 1999), and the leukaemia dataset (Golub et al., 1999).

2 METHODS

2.1 Least Squares Support Vector Machines

In the past few years support vector machines (Vapnik, 1998) have been introduced for solving pattern recognition problems. When used for classification, SVMs separate one class from the other with a hyperplane that maximizes the distance between the hyperplane and the nearest sample of each class. The determination of the hyperplane involves solving a quadratic programming(QP) problem, which requires expensive computation. To alleviate this problem, Suykens and Vandewalle (1999) proposed a least squares version of SVMs for classification problems. Instead of considering inequality constraints in the classical SVM approach, the LS-SVM employs equality constraints.

Consider \( l \) training data pairs: \( \{x_i, y_i\}, i = 1, \ldots, l \), where \( x_i \) is an \( n \)-dimensional vector representing the \( i \)th sample, and \( y_i \) is the class label of \( x_i \), which is either +1 or -1. The linear decision boundary is described as:

\[
 w^T x + b = 0
\]

where \( w = [w_1, w_2, \ldots, w_n]^T \), and \( b \) is a scalar. In LS-SVMs, the optimization problem is formulated as:

\[
 \min_{w,e} \Phi(w,e) = \frac{1}{2}w^T w + \frac{\gamma}{2}e^T e
\]

subject to equality constraints

\[
 y_i[w^T x_i + b] = 1 - e_i, \quad i = 1, \ldots, l
\]

where \( e_i \) denotes regression error for sample \( x_i \), \( e = [e_1, e_2, \ldots, e_l]^T \), and \( \gamma \) is a given positive value assigned to penalize errors. The role of \( \gamma \), just as that of the \( C \) in classical SVMs, is to adjust the compromise between generalization and training accuracy. The solution to the optimization problem is given by the saddle point of the Lagrangian:

\[
 L(w, b, e, \alpha) = \Phi(w,e) - \sum_{i=1}^{l} \alpha_i[y_i(w^T x_i + b) - 1 + e_i]
\]

with Lagrange multipliers \( \alpha_i \). The conditions for optimality

\[
 \frac{\partial L}{\partial w} = 0, \quad \frac{\partial L}{\partial b} = 0, \quad \frac{\partial L}{\partial \alpha_i} = 0 \quad \text{and} \quad \frac{\partial L}{\partial e_i} = 0
\]
Eq. (5) can be written in matrix form as:

\[
\begin{bmatrix}
0 \\
\mathbf{Y} \\
\mathbf{\Omega} + \gamma^{-1} \mathbf{I}
\end{bmatrix}
\begin{bmatrix}
b \\
\alpha
\end{bmatrix}
= \begin{bmatrix}
0 \\
\mathbf{I}
\end{bmatrix}
\]  

(6)

where \( \mathbf{Y} = [y_1, \ldots, y_l]^T \), \( \mathbf{I} = [1, \ldots, 1]^T \), \( \alpha = [\alpha_1, \ldots, \alpha_l]^T \), and \( \mathbf{\Omega} = \{y_i y_j x_i^T x_j\} \).

Once the classifier is trained, we assign the corresponding class label of the test pattern \( \mathbf{x} \) by the sign of the function \( f(\mathbf{x}) \):

\[
f(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b
\]  

(7)

2.2 LS Bound measure for gene selection

Feature selection approaches can be broadly grouped into filter and wrapper methods based on the evaluation criteria (Deviyier and Kittler, 1982). The filter method evaluates feature subset based on intrinsic properties of data, which are related to the performance of the classifier but are not the direct function of the performance. In contrast the wrapper method evaluates the feature subset based on the performance of the classifier directly. For better generalization, often the leave-one-out cross validation error is employed to guide the selection in the wrapper method.

It is proved that the leave-one-out procedure gives an almost unbiased estimate of the probability of test error (Luntz and Brailovsky, 1969). But to obtain the leave-one-out error of each training data for a particular gene set, it requires repeated training of classifiers in the leave-one-out procedure, which makes the procedure a burdensome task. Several bounds on the expectation of SVMs from the leave-one-out estimator were introduced to reduce the high computational complexity in the leave-one-out procedure (Vapnik and Chapelle, 2000). SVMs are well suited to work with high dimensional data, such as microarray data (Furey et al., 2000). But the determination of the hyperplane involves solving a quadratic programming (QP) problem, which requires expensive computation. To alleviate this problem, our present work is focus on the LS-SVMs (Suykens and Vandewalle, 1999). The advantage of LS-SVMs is that the solution of LS-SVM can be obtained from solving a set of linear equations, which is much easier than the quadratic programming in the classical SVM. Although the performance of LS-SVMs on the classification of gene expression data might not be better than classical SVMs, the simplicity and high computational efficiency of LS-SVMs will be of great benefit to the selection of gene subset from the high dimensional gene expression data.

The decision function of a linear classifier can be written as:

\[
f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b
\]  

(8)

The classification is based on the sign of the decision function \( f(\mathbf{x}) \) in Eq. (8). In the present work, we are concerned with value of \( f(\mathbf{x}) \) rather than the sign of \( f(\mathbf{x}) \), because the value of \( f(\mathbf{x}) \) not only indicates the information of the class label of the sample \( \mathbf{x} \), but also indicates the distance of sample \( \mathbf{x} \) to the decision boundary. This value will change whenever a feature is added to the feature subset or deleted from the feature subset. Even if different feature subsets lead to same classification results, the value \( f(\mathbf{x}) \) also provides a discriminant power to identify which subset is the best.

Next we present a leave-one-out bound of \( f(\mathbf{x}) \) by training once on the entire training set to avoid the repeated training for the leave-one-out procedure. It can be proved (see proof in the companion website) that when LS-SVM classifier is trained on the entire training set, if the corresponding Lagrangian multiplier \( (\alpha_i^0) \) of the training sample \( \mathbf{x}_p \) is positive, the following inequality holds in the leave-one-out procedure.

\[
-y_p f^p(\mathbf{x}_p) \leq \alpha_i^0 [(D_{min}^p)^2 + 2/\gamma] - 1
\]  

(9)

where \( f^p \) is the decision function given by the LS-SVM after the sample \( \mathbf{x}_p \) has been removed, \( \alpha_i^0 \) is the corresponding Lagrangian multiplier of \( \mathbf{x}_p \) when the LS-SVM classifier is trained on the entire training set, and \( D_{min}^p \) is the distance between \( \mathbf{x}_p \) and its nearest neighbor. \( f^p(\mathbf{x}_p) \) is the validation result for the sample \( \mathbf{x}_p \) in the leave-one-out procedure. If \( y_p f^p(\mathbf{x}_p) \) is negative the sample \( \mathbf{x}_p \) is considered as an leave-one-out error, and if \( y_p f^p(\mathbf{x}_p) \) is positive \( \mathbf{x}_p \) is correctly classified in the leave-one-out procedure.

In the LS-SVM, we have \( \alpha_i^0 = \gamma e_i^0 \) and \( y_p f^0(\mathbf{x}_p) = 1 - e_p^0 \) (See Eq. (5) and Eq. (3)). According to the two equations, we group the training data into three categories based on the value of Lagrange multiplier \( \alpha_i^0 \). The first category includes the samples with \( \alpha_i^0 \leq 0 \), which are far away from the decision boundary. They could be correctly classified not only when the decision function is trained on the basis of the entire data set, but also during the leave-one-out cross validation procedure in normal case. The samples with \( 0 < \alpha_i^0 < \gamma \) are in the second category. Although these samples are correctly classified in the training stage, but they might be misclassified during the leave-one-out procedure because they are close to the decision boundary. The third category is comprised of samples with \( \alpha_i^0 \geq \gamma \), which
are misclassified not only in the training procedure but also in the leave-one-out procedure. The bound on the right side of inequality (9) is especially useful for the data in the second category. The upper bound is related to both the corresponding training result and the nearest neighbor. If the bound is negative the sample must be correctly classified in the leave-one-out procedure. If the bound is positive, although we are uncertain whether it will be misclassified in the leave-one-out procedure, the bound indicates to some extent the probability of misclassification, and hence can be used to evaluate the goodness of the feature (gene) set.

Combining the bounds for all training data together, we propose the following measure, called LS Bound measure, as the evaluation criterion for gene selection:

\[
M = \sum_{p=1}^{l} (\alpha_{p}^0 (D_{\min}^p)^2 + 2/\gamma - 1)_+ \tag{10}
\]

where \( (x)_+ = \max(0, x) \). \( \alpha^0 \) can be obtained from solving a set of linear equations in the LS-SVM. The solution of \( \alpha \) also can be simply formulated as:

\[
\alpha = H^{-1}I - (Y^TH^{-1}I)(H^{-1}Y)(Y^TH^{-1}Y) \tag{11}
\]

where \( Y = [y_1, \ldots, y_l]^T \), \( I = [1, \ldots, 1]^T \), and \( H = \Omega + \gamma^{-1}I \). Note that the matrix \( H \) is positive definite and therefore invertible. The measure \( M \) gives a bound on error expectation, and can be considered as an estimate of the generalization performance. The feature (gene) subset which minimizes the measure \( M \) is preferred.

The LS Bound measure \( M \) in Eq. (10) can be combined with any search algorithm, such as the sequential forward selection (SFS), to form a gene selection algorithm. The SFS algorithm is a simple greedy heuristic search algorithm. For better performance, other complex search algorithms, such as sequential floating forward selection (SFFS) (Pudil et al., 1994), can be used but at the cost of increasing the computational complexity. In this paper we only focus on the SFS algorithm, while the implementation and experimental results of combining LS Bound with SFFS algorithm can be found in our website (http://www.ntu.edu.sg/home5/pg02776030/lsbound/).

The pseudo code of the sequential forward gene selection algorithm can be summarized as follows:

The gene selection algorithm combining the LS Bound measure \( M \) with SFS

1. Initialize \( S \) to an empty set;
   /* \( S \) is the set of selected genes */
2. Initialize \( C \) to the full gene set;
   /* \( C \) is the set of candidate genes for selection */
3. For \( i = 1 \) to \( m 
   /* \( m \) genes are expected to be selected */
   \( p = \) number of genes in set \( C \);
   For \( j = 1 \) to \( p \)
   Take gene \( j \) from set \( C \) and temporarily put into set \( S \);
   Calculate the measure \( M \) using all genes in set \( S \);
   End
   Select the gene with the minimal \( M \);
   Put the selected gene into set \( S \);
   End

The stopping criterion in our gene selection algorithm is whether a predefined number of features are selected. In our experiments, we first selected 50 genes from the original gene set. However the 50 genes are not the final selection result because few of them can get the same or satisfactory performance. In our opinion, the LS Bound measure reflects the generalization performance of selected gene subset. The decrease of the LS Bound measure indicates the improvement of the generalization performance. The decision of final gene subset can be made based on whether the LS Bound measure approaches the minimum, or whether adding more genes results in insignificant change. For example on the colon cancer dataset (see section of Results), we selected 15 genes because adding more genes would not reduce the LS Bound measure significantly. For well separated dataset, after selecting several genes the LS Bound measure would be reduced to zero. Under such a case, the gene selection may be terminated when LS Bound measure is very close to zero.

To select the relevant genes using the LS Bound measure proposed in the present study, it is important to find the appropriate value of \( \gamma \) in Eq. (10). This can be done using a simple technique. We employed our algorithm with a sequence of given values of \( \gamma \) to select important genes on the entire dataset. Because the LS Bound measure indicates the generalization performance, the optimal value of \( \gamma \) is chosen to be the one which gives the minimal LS Bound measure during the selection procedure.

2.3 Efficient Computation of LS Bound measure

In the gene selection algorithm described above, the LS Bound measure \( M \) needs to be computed for every candidate gene set. For a high dimensional dataset such as microarray data, the repeated computation of \( M \) involving the inverse of an \( l \)-by-\( l \) matrix \( H \) in Eq. (11) is actually a computationally intensive task. However, in the SFS search procedure we can express the inverse by using the Sherman-Morrison-Woodbury formula (Golub and van Loan, 1996):

\[
(M + BCD^T)^{-1} = M^{-1} - M^{-1}B(C^{-1} + D^TM^{-1}B)^{-1}D^TM^{-1} \tag{12}
\]
Suppose that we have selected a particular gene set \( S \) and the matrix \( H \) for gene set \( S \) is denoted by \( H_S \). During the SFS search procedure, we have to temporarily put each of the candidate genes into gene set \( S \) to test the performance of each candidate gene. When gene \( k \) is added into gene set \( S \) to form a temporary set \( S' = S \cup \{x_k\} \), the matrix \( H \) for gene set \( S' \), denoted by \( H_{S'} \), can be formulated as

\[
H_{S'} = H_S + zz^T
\]

where vector \( z = (y_1x_{1k}, \ldots, y_kx_{kk})^T \). According to the Sherman-Morrison-Woodbury formula, we have:

\[
H_{S'}^{-1} = H_S^{-1} - \frac{H_S^{-1}zz^TH_S^{-1}}{1 + z^TH_S^{-1}z} \tag{14}
\]

Eq. (14) reveals that the inverse of \( H_{S'} \) corresponding to a new subset \( S' = S \cup \{x_k\} \) can be recursively computed from the inverse of \( H_S \) using simple matrix operations. Thus, in the SFS procedure, the complex inverse operation is no longer needed by using the trick of Eq. (14). Even compared with the filter measures, our measure, as well as our algorithm, is quite efficient. For example, 33 seconds are needed to select 50 genes from 1000 candidate genes using our measure, our algorithm, and to compare the performance with other gene selection algorithms. Because there are no consistent approaches for evaluation of gene selection algorithms in the literature, it is not easy to compare the different algorithms only using the published results. Therefore we employed the same technique, external 632+ Bootstrap, to assess the performance of different gene selection algorithms for comparison purpose in our study (the gene selection algorithms used for comparison are described below). The bootstrap samples are generated by resampling with replacement from the original dataset. In the present study, the balanced bootstrap samples with \( K = 200 \) replicates are employed to reduce the variance. Each sample in the original sample set is made to appear exactly \( K \) times in the balanced bootstrap samples. The SVM was employed as the classifier to estimate the error rates of different gene selection algorithms.

In the present study, we compared our LS Bound measure with a few feature selection criteria, including Fisher’s ratio and Mahalanobis class separability measure. Fisher’s ratio is an individual gene ranking criterion, and is used to evaluate how well a single gene is correlated with the separation between classes. For every gene the Fisher’s ratio is defined as

\[
f = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2},
\]

where \( \mu_1, \mu_2, \sigma_1, \sigma_2 \) denote the means and standard deviations of two classes. The Mahalanobis
class separability measure is a well-known feature subset evaluation criterion in the literature of pattern recognition (Devijver and Kittler, 1982). In the experiment, the Mahalanobis class separability measure is combined with SFS algorithm for gene selection. We also compared our gene selection algorithm with other published gene selection algorithms, including the Weighting factor employed both in the Weighted Voting algorithm (Golub et al., 1999) and with the Support vector machine (Furey et al., 2000), and SVM Recursive Feature Elimination (Guyon et al., 2002). The weighting factor \( a = \frac{|\mu_1 - \mu_2|}{\sigma_1 + \sigma_2} \) is a minor variant of Fisher’s ratio, and is also commonly used in the literature of microarray data analysis. Benefiting from the good performance of SVMs in high dimensional gene expression data, SVM RFE is often considered as one of the best gene selection algorithm in the literature.

3.1 Colon cancer dataset
The colon cancer dataset (Alon et al., 1999) contains gene expression levels of 40 tumor and 22 normal colon tissues for 2000 genes. The task is to identify important genes which can distinguish colon cancer from normal tissues.

We assessed the performance of our gene selection criterion using external .632+ bootstrap technique and compared it with Fisher’s ratio and Mahalanobis class separability measure, and other published gene selection algorithms, such as Weighting factor and SVM RFE. As shown in Fig. 1, LS Bound measure is slightly inferior to SVM RFE, but it outperforms other three algorithms. Note that only 1, 2, 4, 8, 16, 32 genes are selected in SVM RFE due to its selection mechanism.

To identify the important genes to classification problem, we first identified 50 genes by employing the gene selection on the entire original dataset. According to the LS Bound measure on the selected gene subset, we finally selected the first 15 genes which can distinguish colon cancer from normal tissues.

3.2 Leukaemia dataset
The leukaemia dataset was first described in Golub et al. (1999). The dataset contains gene expression levels of 72 patients with either acute lymphoblastic leukaemia (ALL, 47 cases) and acute myeloid leukaemia (AML, 25 cases) for 7129 human genes. The raw data are available at http://www-genome.wi.mit.edu/cancer/. The task is to identify important genes which are discriminant between ALL and AML.

Again, we assessed the performance of our gene selection criterion using external .632+ bootstrap technique and compared it with Fisher’s ratio, Mahalanobis class separability measure, Weighting factor and SVM RFE. As shown in Fig. 2, our gene selection algorithm outperforms the other algorithms in this dataset.

By employing the gene selection on the entire dataset we selected 12 most important genes for the classification between ALL and AML. The 12 genes are shown in Tab. 2. Among the selected genes, some of them have been reported to be related to myeloid or lymphoblastic leukaemia. Adipsin (D component of complement) is contained in the locus 19p13.3, whose chromatin reorganization is known to be associated with myeloid cell differentiation (Wong et al., 1999). Cyclin D3 encodes protein critical for controlling the physiological progression from the G1 to the S phase of the cell cycle. In acute lymphoblastic leukaemia (ALL) cells conditional expression of cyclin D3 leads to preventing glucocorticoid-induced cell cycle G1 arrest (Ausserlechner et al., 2004). The gene ALDH1, which encodes the retinoic acid synthesizing enzyme, was found to be ectopically activated by HOX11 in NIH 3T3 cells, while the activation of HOX11 is a feature of some T-cell tumors and in many T-cell ALL cases the HOX11 gene is activated by translocation into either of two T-cell receptor loci (Greene et al., 1998). CD10, the common acute lymphoblastic leukemia antigen (CALLA), is the most common marker for clinical immunophenotypic classification of ALL. It is a membrane-bound neutral endopeptidase that can be expressed in both B- and T-cell ALL (Pui et al., 1993). In the work of Golub et al. (1999) four genes (adipsin, cyclin D3, zyxin and SMARCA4) are also selected. Although Zyxin gene, which encodes protein important for cell adhesion, and SMARCA4, which encodes protein for chromatin remodeling, were not reported to have any role in hematopoisis, Zyxin is highly correlated with acute myelogenous leukaemia (AML) while SMARCA4 is correlated with B-precursor ALL.

4 CONCLUSION
In the present study, we have proposed an LS Bound measure as the evaluate criterion for gene selection. The LS Bound measure can be considered as a hybrid of filter and wrapper methods. On the one hand, the LS Bound
LS Bound based gene selection

Fig. 1. The external B.632+ error for the colon cancer dataset, shown as the number of selected genes. The five curves are obtained from five gene selection algorithms: the LS Bound measure combining with SFS algorithm, Fisher’s ratio, Mahalanobis class separability measure with SFS algorithm, Weighting factor and SVM RFE. The performance of LS Bound measure is quite well.

Table 1. The top 15 selected genes for the colon cancer dataset

<table>
<thead>
<tr>
<th>No.</th>
<th>Access No.</th>
<th>Gene Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>R87126</td>
<td>MYH, nonmuscle(Gallus gallus)</td>
</tr>
<tr>
<td>2</td>
<td>X55715</td>
<td>RPS2</td>
</tr>
<tr>
<td>3</td>
<td>T94579</td>
<td>Chitotriosidase precursor</td>
</tr>
<tr>
<td>4</td>
<td>T61661</td>
<td>PFN1</td>
</tr>
<tr>
<td>5</td>
<td>T57882</td>
<td>MYH9</td>
</tr>
<tr>
<td>6</td>
<td>R88740</td>
<td>ATP5J</td>
</tr>
<tr>
<td>7</td>
<td>T70062</td>
<td>ILF2</td>
</tr>
<tr>
<td>8</td>
<td>L37792</td>
<td>STX1A</td>
</tr>
<tr>
<td>9</td>
<td>M59042</td>
<td>CD44</td>
</tr>
<tr>
<td>10</td>
<td>K03474</td>
<td>MIS</td>
</tr>
<tr>
<td>11</td>
<td>T63508</td>
<td>Ferritin heavy chain</td>
</tr>
<tr>
<td>12</td>
<td>M84349</td>
<td>CD59</td>
</tr>
<tr>
<td>13</td>
<td>H15813</td>
<td>CEBPB</td>
</tr>
<tr>
<td>14</td>
<td>D00762</td>
<td>Proteasome component C8</td>
</tr>
<tr>
<td>15</td>
<td>X53586</td>
<td>ITGA6</td>
</tr>
</tbody>
</table>

The genes and ESTs without indicating the source are all from Homo sapiens.

Fig. 2. The external B.632+ error for the leukaemia dataset, shown as the number of selected genes. The five curves are obtained from five gene selection algorithms: the LS Bound measure combining with SFS algorithm, Fisher’s ratio, Mahalanobis class separability measure with SFS algorithm, Weighting factor and SVM RFE. The LS Bound measure results in better performance than others.

Table 2. The top 12 selected genes for the leukaemia dataset

<table>
<thead>
<tr>
<th>No.</th>
<th>Access No.</th>
<th>Gene Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M84526</td>
<td>Adipsin</td>
</tr>
<tr>
<td>2</td>
<td>L15326</td>
<td>PTGS2(COX2)</td>
</tr>
<tr>
<td>3</td>
<td>M92287</td>
<td>Cyclin D3 (CCND3)</td>
</tr>
<tr>
<td>4</td>
<td>D26308</td>
<td>NADPH-flavin reductase</td>
</tr>
<tr>
<td>5</td>
<td>X95735</td>
<td>Zyxin</td>
</tr>
<tr>
<td>6</td>
<td>U75276</td>
<td>BRF1</td>
</tr>
<tr>
<td>7</td>
<td>U88667</td>
<td>ABCA4</td>
</tr>
<tr>
<td>8</td>
<td>U40343</td>
<td>CDKN2D</td>
</tr>
<tr>
<td>9</td>
<td>M31994</td>
<td>ALDH1</td>
</tr>
<tr>
<td>10</td>
<td>D26156</td>
<td>SMARCA4(SNF2)</td>
</tr>
<tr>
<td>11</td>
<td>J03779</td>
<td>CD10</td>
</tr>
<tr>
<td>12</td>
<td>S57212</td>
<td>MEF2C</td>
</tr>
</tbody>
</table>

All genes are from human mRNA.

On the other hand, unlike the classical wrapper method, the LS Bound measure does not demand repeated trainings for cross validation. The training procedure involved is implicitly expressed in Eq. (11), which has a similar computational complexity to the filter method. The effectiveness of the LS Bound measure has been tested on two benchmark microarray datasets when the LS Bound measure is combined with SFS to form a gene selection algorithm. We also combined the sequential floating forward selection (SFFS) search procedure with
LS Bound measure for gene selection on the same datasets, and the results revealed that the superiority of the LS Bound measure does not depend on a specific search procedure. For details, please refer to the website http://www.ntu.edu.sg/home5/pg02776030/lsbound/.

REFERENCES


