A Genetic Embedded Approach for Gene Selection and Classification of Microarray Data

Jose Crispin Hernandez Hernandez, Béatrice Duval, and Jin-Kao Hao

LERIA, Université d'Angers,
2 Boulevard Lavoisier, 49045 Angers, France
{josehh,bd,hao}@info.univ-angers.fr

Abstract. Classification of microarray data requires the selection of subsets of relevant genes in order to achieve good classification performance. This article presents a genetic embedded approach that performs the selection task for a SVM classifier. The main feature of the proposed approach concerns the highly specialized crossover and mutation operators that take into account gene ranking information provided by the SVM classifier. The effectiveness of our approach is assessed using three well-known benchmark data sets from the literature, showing highly competitive results.

Keywords: Microarray gene expression, Feature selection, Genetic Algorithms, Support vector machines.

1 Introduction

Recent advances in DNA microarray technologies enable to consider molecular cancer diagnosis based on gene expression. Classification of tissue samples from gene expression levels aims to distinguish between normal and tumor samples, or to recognize particular kinds of tumors [9,2]. Gene expression levels are obtained by cDNA microarrays and high density oligonucleotide chips, that allow to monitor and measure simultaneously gene expressions for thousands of genes in a sample. So, data that are currently available in this field concern a very large number of variables (thousands of gene expressions) relative to a small number of observations (typically under one hundred samples). This characteristic, known as the "curse of dimensionality", is a difficult problem for classification methods and requires special techniques to reduce the data dimensionality in order to obtain reliable predictive results.

Feature selection aims at selecting a (small) subset of informative features from the initial data in order to obtain high classification accuracy [11]. In the literature there are two main approaches to solve this problem: the filter approach and the wrapper approach [11]. In the filter approach, feature selection is performed without taking into account the classification algorithm that will be applied to the selected features. So a filter algorithm generally relies on a relevance measure that evaluates the importance of each feature for the classification task. A feasible approach to filter selection is to rank all the features

E. Marchiori, J.H. Moore, and J.C. Rajapakse (Eds.): EvoBIO 2007, LNCS 4447, pp. 90–101, 2007.

according to their interestingness for the classification problem and to select the top ranked features. The feature score can be obtained independently for each feature, as it is done in [9] which relies on correlation coefficients between the class and each feature. The drawback of such a method is to score each feature independently while ignoring the relations between the features.

In contrast, the wrapper approach selects a subset of features that is "optimized" by a given classification algorithm, *e.g.* a SVM classifier [5]. The classification algorithm, that is considered as a black box, is run many times on different candidate subsets, and each time, the quality of the candidate subset is evaluated by the performance of the classification algorithm trained on this subset. The wrapper approach conducts thus a search in the space of candidate subsets. For this search problem, genetic algorithms have been used in a number of studies [15,14,6,4].

More recently, the literature also introduced embedded methods for feature selection. Similar to wrapper methods, embedded methods carry out feature selection as a part of the training process, so the learning algorithm is no more a simple black box. One example of an embedded method is proposed in [10] with recursive feature elimination using SVM (SVM-RFE).

In this paper, we present a novel embedded approach for gene selection and classification which is composed of two main phases. For a given data set, we carry out first a pre-selection of genes based on filtering criteria, leading to a reduced gene subset space. This reduced space is then searched to identify even smaller subsets of predictive genes which are able to classify with high accuracy new samples. This search task is ensured by a specialized Genetic Algorithm which uses (among other things) a SVM classifier to evaluate the fitness of the candidate gene subsets and problem specific genetic operators. Using SVM to evaluate the fitness of the individuals (gene subsets) is not a new idea. Our main contribution consists in the design of semantically meaningful crossover and mutation operators which are fully based on useful ranking information provided by the SVM classifier. As we show in the experimentation section, this approach allows us to obtain highly competitive results on three well-known data sets.

In the next Section, we recall three existing filtering criteria that are used in our pre-selection phase and SVM that is used in our GA. In Section 3, we describe our specialized GA for gene selection and classification. Experimental results and comparisons are presented in Section 4 before conclusions are given in Section 5.

2 Basic Concepts

2.1 Filtering Criteria for Pre-selection

As explained above, microarray data generally concern several thousands of gene expressions. It is thus necessary to pre-select a smaller number of genes before applying other search methods. This pre-selection can be performed by using simply a classical filter method that we recall in this section. The following

filtering or relevance criteria assign to each gene a numerical weight that is used to rank all the genes and then to select top ranked genes.

In the rest of the paper, we shall use the following notations. The matrix of gene expression is denoted by $D = \{(X_i, y_i) \mid i = 1, ..., n\}$, where each (X_i, y_i) is a labeled sample. The labels $y_1, ..., y_n$ are taken from a set of labels Y which represent the different classes (for a two class problem $Y = \{-1, 1\}$). Each $X_i = \{x_{i,1}, ..., x_{i,d}\}$ describes the expression values of the d genes for sample i.

The **BW ratio**, introduced by Dudoit *et al.* [7], is the ratio of between-group to within-group sums of squares. For a gene j, the ratio is formally defined by:

$$BW\,(j) = \frac{\sum_i \sum_k \; I(y_i = k)(\bar{x}_{kj} - \bar{x}_j)^2}{\sum_i \sum_k \; I(y_i = k)(x_{ij} - \bar{x}_{kj})^2} \tag{1}$$

where $I(.)$ denotes the indicator function, equaling 1 if the condition in parentheses is true, and 0 otherwise. $\bar{x}_j$ and $\bar{x}_{kj}$ denote respectively the average expression level of the gene j across all samples and across samples belonging to class k only.

The **Correlation between a gene and a class distinction**, proposed by Golub *et al.* [9], is defined as follows.

$$P\,(j) = \frac{\bar{x}_{1j} - \bar{x}_{2j}}{s_{1j} + s_{2j}} \tag{2}$$

where $\bar{x}_{1j}$,s_{1j} and $\bar{x}_{2j}$,s_{2j} denote the mean and standard deviation of the gene expression values of gene j for the samples in class 1 and class 2. This measure identifies for a two-class problem informative genes based on their correlation with the class distinction and emphasizes the signal-to-noise ratio by using the gene as a predictor. $P(j)$ reflects the difference between the classes relative to the standard deviation within the classes. Large values of $|P(j)|$ indicate a strong correlation between the gene expression and the class distinction.

The **Fisher's discriminant criterion** [8] is defined by:

$$P\,(j) = \frac{(\bar{x}_{1j} - \bar{x}_{2j})^2}{((s_{1j})^2 + (s_{2j})^2)} \tag{3}$$

where $\bar{x}_{1j}$, $(s_{1j})^2$ and $\bar{x}_{2j}$, $(s_{2j})^2$ denote respectively the mean and variance of the gene expression values of gene j across the class 1 and across the class 2. It gives higher values to features whose means differ greatly between the two classes, relative to their variances.

Any of the above criteria can be used to select a subset G_p of p top ranked genes. In our case, we shall compare, in Section 4, these three criteria and retain the best one to be combined with our genetic embedded approach for gene selection and classification.

2.2 Support Vector Machines (SVMs) and Feature Ranking

In our genetic embedded approach, a SVM classifier is used to evaluate the fitness of a given candidate gene subset. Let us recall briefly the basic concept of

SVM. For a given training set of labeled samples, SVM determines an optimal hyperplane that divides the positively and the negative labeled samples with the maximum margin of separation.

Formally, given a training set belonging to two classes, $\{X_i, y_i\}$ where $\{X_i\}$ are the n training samples with their class labels y_i, a soft-margin linear SVM classifier aims at solving the following optimization problem:

$$\min_{w,b,\xi_i} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \tag{4}$$

subject to $y_i (w \cdot X_i + b) \geq 1 - \xi_i$ and $\xi_i \geq 0$, $i = 1, ..., n$.

C is a given penalty term that controls the cost of misclassification errors.

To solve the optimization problem, it is convenient to consider the dual formulation [5]:

$$\min_{\alpha_i} \frac{1}{2} \sum_{i=1}^{n} \sum_{l=1}^{n} \alpha_i \alpha_l y_i y_l X_i \cdot X_l - \sum_{i=1}^{n} \alpha_i \tag{5}$$

subject to $\sum_{i=1}^{n} y_i \alpha_i = 0$ and $0 \leq \alpha_i \leq C$.

The decision function for the linear SVM classifier with input vector X is given by $f(X) = w \cdot X + b$ with $w = \sum_{i=1}^{n} \alpha_i y_i X_i$ and $b = y_i - w \cdot X_i$.

The weight vector w is a linear combination of training samples. Most weights α_i are zero. The training samples with non-zero weights are support vectors.

In order to select informative genes, the orientation of the separating hyperplane found by a linear SVM can be used, see [10]. If the plane is orthogonal to a particular gene dimension, then that gene is informative, and vice versa. Specially, given a SVM with weight vector w the ranking coefficient vector c is given by:

$$\forall i, c_i = (w_i)^2 \tag{6}$$

For our classification task, we will use such a linear SVM classifier with our genetic algorithm which is presented in the next section. Finally, let us mention that SVM has been successfully used for gene selection and classification [16,17,10,13].

3 Gene Selection and Classification by GA

As explained in the introduction, our genetic embedded approach begins with a filtering based pre-selection, leading to a gene subset G_p of p genes (typically with $p < 100$). From this reduced subset, we will determine an even smaller set of the most informative genes (typically < 10) which allows to give the highest classification accuracy. To achieve this goal, we developed a highly specialized Genetic Algorithm which integrates, in its genetic operators, specific knowledges on our gene selection and classification problem and uses a SVM classifier as one key element of its fitness function. In what follows, we present the different elements of this GA, focusing on the most important and original ingredients:

problem encoding, SVM based fitness evaluation, specialized crossover and mutation operators.

3.1 Problem Encoding

An individual $I = < I^x, I^y >$ is composed of two parts I^x and I^y called respectively *gene subset vector* and *ranking coefficient vector*. The first part, I^x, is a binary vector of fixed length p. Each bit I^x_i $(i = 1...p)$ corresponds to a particular gene and indicates whether or not the gene is selected. The second part, I^y, is a positive real vector of fixed length p and corresponds to the ranking coefficient vector c (Equation 6) of the linear SVM classifier. I^y indicates thus for each selected gene the importance of this gene for the SVM classifier.

Therefore, an individual represents a candidate subset of genes with additional information on each selected gene with respect to the SVM classifier. The gene subset vector of an individual will be evaluated by a linear SVM classifier while the ranking coefficients obtained during this evaluation provide useful information for our specialized crossover and mutation operators.

3.2 SVM Based Fitness Evaluation

Given an individual $I = < I^x, I^y >$, the gene subset part I^x, is evaluated by two criteria: the classification accuracy obtained with the linear SVM classifier trained on this subset and the number of genes contained in this subset. More formally, the fitness function is defined as follows:

$$f(I) = \frac{CA_{SVM}(I^x) + \left(1 - \frac{|I^x|}{p}\right)}{2} \tag{7}$$

The first term of the fitness function ($CA_{SVM}(I^x)$) is the classification accuracy measured by the SVM classifier via 10-fold cross-validation. The second term ensures that for two gene subsets having an equal classification accuracy, the smaller one is preferred.

For a given individual I, this fitness function leads to a positive real fitness value $f(I)$ (higher values are better). At the same time, the c vector obtained from the SVM classifier is calculated and copied in I^y which is later used by the crossover and mutation operators.

3.3 Specialized Crossover Operator

Crossover is one of the key evolution operators for any effective GA and needs a particularly careful design. For our search problem, we want to obtain small subsets of selected genes with a high classification accuracy. Going with this goal, we have designed a highly specialized crossover operator which is based on the following two fundamental principles: 1) to conserve the genes shared by both parents and 2) to preserve "high quality" genes from each parent even if they are not shared by both parents. The notion of "quality" of a gene here is

defined by the corresponding ranking coefficient in c. Notice that applying the first principle will have as main effect of getting smaller and smaller gene subsets while applying the second principle allows us to keep up good genes along the search process.

More precisely, let $I =< I^x, I^y >$ and $J =< J^x, J^y >$ be two selected individuals (parents), we combine I and J to obtain a single child $K =< K^x, K^y >$ by carrying out the following steps:

1. We use the boolean logic AND operator ($\otimes$) to extract the subset of genes shared by both parents and arrange them in an intermediary gene subset vector F.
$$F = I^x \otimes J^x$$
2. For the subset of genes obtained from the first step, we extract the maximum coefficients max_I and max_J accordingly from their original ranking vectors I^y and J^y.
$$max_I = max\,\{I_i^y \mid i\ such\ that\ F_i = 1\}$$
and
$$max_J = max\,\{J_i^y \mid i\ such\ that\ F_i = 1\}$$
3. This step aims to transmit high quality genes from each parent I and J which are not retained by the logic AND operator in the first step. These are genes with a ranking coefficient greater than max_I and max_J. The genes selected from I and J are stored in two intermediary vectors AI and AJ
$$AI_i = \begin{cases} 1 & if\ I_i^x = 1\ and\ F_i = 0\ and\ I_i^y > max_I \\ 0 & otherwise \end{cases}$$
and
$$AJ_i = \begin{cases} 1 & if\ J_i^x = 1\ and\ F_i = 0\ and\ J_i^y > max_J \\ 0 & otherwise \end{cases}$$
4. The gene subset vector K^x of the offspring K is then obtained by grouping all the genes of F, AI and AJ using the logical "OR" operator ($\oplus$).
$$K^x = F \oplus AI \oplus AJ$$

The ranking coefficient vector K^y will be filled up when the individual K is evaluated by the SVM based fitness function.

3.4 Specialized Mutation Operator

As for the above crossover operator, we design a mutation operator which is semantically meaningful with respect to our gene selection and classification problem. The basic idea is to eliminate some "mediocre" genes and at the same time introduce randomly other genes to keep some degree of diversity in the GA population.

Given an individual $I =< I^x, I^y >$, applying the mutation operator to I consists in carrying out the following steps.

1. The first step calculates the average ranking coefficient of a gene in the individual I.
$$\bar{c} = \frac{\sum_{k=1}^{p} I_k^y}{p}$$
2. The second step eliminates (with a probability) "mediocre" genes (*i.e.* inferior to the average) and for each deleted gene introduces randomly a new gene. $\forall I_i^x = 1$ and $I_i^y < \bar{c}$ $(i = 1...p)$, mutate I_i^x with probability p^m. If a mutation does occur, take randomly a I_j^x such that I_j^x=0 and set I_j^x to 1.

3.5 The General GA and Its Other Components

An initial population P is randomly generated such that the number of genes by each individual varies between p and $p/2$ genes. From this population, the fitness of each individual I is evaluated using the function defined by the formula 7. The ranking coefficient vector c of the SVM classifier is then copied to I^y.

To obtain a new population, a temporary population P' is used. To fill up P', the top 40% individuals of P are first copied to P' (elitism). The rest of P' is completed with individuals obtained by crossover and mutation. Precisely, Stochastic Universal Selection is applied to P to generate a pool of $|P|$ candidat individuals. From this pool, crossover is applied $0.4 * |P|$ times to pairs of randomly taken individuals, each new resulting individual being inserted in P'. Similarly, mutation is applied $0.2 * |P|$ times to randomly taken individuals to fill up P'. Once P' is filled up, it replaces P to become the current population. The GA stops when a fixed number of generations is reached.

4 Experimental Results

4.1 Data Sets

We applied our approach on three well-known data sets that concern leukemia, colon cancer and lymphoma.

The leukemia data set consists of 72 tissue samples, each with 7129 gene expression values. The samples include 47 acute lymphoblastic leukemia (ALL) and 25 acute myeloid leukemia (AML). The original data are divided into a training set of 38 samples and a test set of 34 samples. The data were produced from Affymetrix gene chips. The data set was first used in [9] and is available at http://www-genome.wi.mit.edu/cancer/.

The colon cancer data set contains 62 tissue samples, each with 2000 gene expression values. The tissue samples include 22 normal and 40 colon cancer cases. The data set is available at http://www.molbio.princeton.edu/colondata and was first studied in [2].

The lymphoma data set is based on 4096 variables describing 96 observations (62 and 34 of which are respectively considered as abnormal and normal). The data set was first analyzed in [1]. This data set has already been used for benchmarking feature selection algorithms, for instance in [17,16]. The data set is available at http://www.kyb.tuebingen.mpg.de/bs/people/spider/.

Prior to running our method, we apply a linear normalization procedure to each data set to transform the gene expressions to mean value 0 and standard deviation 1.

4.2 Experimentation Setup

The following sub-sections present and analyze the different experiments that we carried out in order to compare our approach with other selection methods. We present here the general context that we adopt for our experimentations.

Accuracy evaluation is realized by cross validation, as it is commonly done when few samples are available. To avoid the problem of selection bias that is pointed out in [3] and following the protocol suggested in the same study, we use a cross-validation process that is external to the selection process. At each iteration, the data set is split into two subsets, a training set and a test set. Our method of selection is applied on the training set and the accuracy of the classifier is evaluated on the test set (which is not used in the selection process). 50 independent runs are performed, with a new split of the data into a training set and a test set each time. We report in the following the average results (accuracy, number of genes) obtained on these 50 runs. This experimental setup is used in many other works, even if the number of runs may be different. Let us note that our GA also requires an internal cross-validation to estimate the classifier accuracy during the selection process [11].

For the genetic algorithm, both the population size and the number of generations are fixed at 100 for all the experimentations presented in this section. The crossover and mutation operators are applied as explained in Section 3.5.

4.3 Comparison of Pre-selection Criteria

The first experiment aims to determine the best filtering criterion that will be used in our pre-selection phase to obtain an initial and reduced gene subset G_p. To compare the three criteria presented in Section 2, we apply each criterion to each data set to pre-select the p top ranked genes and then we apply our genetic algorithm to these p genes to seek the most informative ones. We experiment with different values of p (p=50...150) and we observe that large values of p does not affect greatly the classification accuracy, but necessarily increase the computation times. So we decide to pre-select p=50 genes.

In order to compare the three filtering criteria, we report in Table 1 the final number of selected genes, the classification accuracy evaluated on the training set and on the test set. From Table 1, one observes that the best results are obtained with the BW ratio measure. Therefore our following experiments are carried out with the BW ratio criterion and the number p of pre-selected genes is always fixed at 50.

4.4 Comparison with Other Selection Methods

In this section, we show two comparative studies. The first compares our method with two well known SVM based selection approaches reported in [16,17]. We

Table 1. Comparison of three pre-selection criteria. NG is the mean and standard deviation of the number of selected genes, AcTr (resp. AcTe) is the average classification rate (%) on training set (resp. on test set).

	BW ratio criteria			Correlation criteria			Fisher's Criterion		
Dataset	*NG*	*AcTr*	*AcTe*	*NG*	*AcTr*	*AcTe*	*NG*	*AcTr*	*AcTe*
Leukemia	3.93±1.16	98.27	89.05	5.07±1.98	94.40	85.59	4.71±1.44	96.59	86.95
Colon	8.05±1.57	90.62	78.81	10.43±2.77	85.47	76.32	9.17±2.03	87.16	76.59
Lymphoma	5.96±1.31	96.53	88.27	8.01±1.94	92.90	84.47	7.13±1.86	93.82	86.02

also carry out a comparison with two highly effective GA-based gene selection approaches [12,14]. Unlike some other studies, these studies are based on the same experimental protocol as ours that avoids the selection bias problem pointed out in [3].

Comparison with SVM-Based Selection Approaches. In [16], the author reports an experimental evaluation of several SVM-based selection methods. For comparison purpose, we adopt the same experimental methodology. In particular, we fix the number of selected genes and adapt our GA to this constraint (the fitness is then determined directly by the classification accuracy of the classifier, c.f. Equation 7). In Table 2, we report the classification accuracies when the number of genes is fixed at 20 and we compare the best results reported in [16] and our results for the data sets concerning the colon cancer and the lymphoma ([16] does not give information about the leukemia data set).

In [17], the authors propose a method for feature selection using the zero-norm (**A**pproximation of the ze**ro**-norm **M**inimization, AROM), and also gives results concerning the colon and lymphoma data sets that we report in Table 2.

Table 2. A comparison of SVM-based selection methods and our method. The columns indicate: the mean and standard deviation of classification rates on test set (Ac), the number of trials (NT), and the number of samples in the test set (NSa).

	[17]			[16]			Our method		
Dataset	*Ac* (%)	*NT*	*NSa*	*Ac* (%)	*NT*	*NSa*	*Ac* (%)	*NT*	*NSa*
Colon	85.83 ±2.0	30	12	82.33 ±9	100	12	82.52 ±8.68	50	12
Lymphoma	91.57 ±0.9	30	12	92.28 ±4	100	36	93.05 ±2.85	50	36

From Table 2, one observes that for the lymphoma data set, our method obtains a better classification accuracy (higher is better). For the colon data set, our result is between the two reference methods. Notice that in this experiment we restrict our method since the number of selected genes is arbitrarily fixed while our method is able to select dynamically subsets of informative genes. The following comparison provides a more interesting experimentation where the number of genes will be determined by the genetic search.

Comparison with Other Genetic Approaches. In [12], the authors propose a multiobjective evolutionary algorithm (MOEA), where the fitness function evaluates simultaneously the misclassification rate of the classifier, the difference in error rate among classes and the number of selected genes. The classifier used in this work was the weighted voting classifier proposed by [9].

In [14], the authors present a probabilistic model building genetic algorithm (PMBGA) as a gene selection algorithm. The Naive-Bayes classifier and the weighted voting classifier are used to evaluate the selection method in a Leave-One-Out-Cross-Validation process.

Table 3 shows our results on the three data sets together with those reported in [12] and [14]. One can observe that our method gives better results than [12], in the sense that the number of selected genes is smaller and the accuracy is higher. Concerning [14], our results are quite comparable.

Table 3. Comparison of other genetic approaches and our method. The columns indicate: the mean and standard deviation of the number of selected genes (NG), the mean and standard deviation of classification rates on test set (Ac). We also report the number (in two cases) or the percentage of samples that form the test set (NSa) for the experiments.

	[12]			[14]			Our method		
Dataset	*NG*	*Ac* (%)	*NSa*	*NG*	*Ac* (%)	*NSa*	*NG*	*Ac* (%)	*NSa*
Leukemia	15.2±4.54	90 ±7.0	30%	3.16±1.00	90 ±6	34	3.17±1.16	91.5 ±5.9	34
Colon	11.4±4.27	80 ±8.3	30%	4.44±1.74	81 ±8	50%	7.05±1.07	84.6 ±6.6	50%
Lymphoma	12.9±4.40	90 ±3.4	30%	4.42±2.46	93 ±4	50%	5.29±1.31	93.3 ±3.1	50%

We must mention that we report the average results obtained by a 10-fold cross validation, but we observe that in some experiments, our method achieves a perfect classification (100% accuracy). Finally, let us comment that these results are comparable to those reported in [6] and better than those of [15].

5 Conclusions and Future Work

In this paper, we have presented a genetic embedded method for gene selection and classification of Microarray data. The proposed method is composed of a pre-selection phase according to a filtering criterion and a genetic search phase to determine the best gene subset for classification. While the pre-selection phase is conventional, our genetic algorithm is characterized by its highly specialized crossover and mutation operators. Indeed, these genetic operators are designed in such a way that they integrate gene ranking information provided by the SVM classifier during the fitness evaluation process. In particular, the crossover operator not only conserves the genes shared by both parents but also uses SVM ranking information to preserve highly ranked genes even if they are not shared by the parents. Similarly, the gene ranking information is incorporated into the mutation operator to eliminate "mediocre" genes.

Using an experimental protocol that avoids the selection bias problem, our method is experimentally assessed on three well-known data sets (colon, leukemia and lymphoma) and compared with several state of the art gene selection and classification algorithms. The experimental results show that our method competes very favorably with the reference methods in terms of the classification accuracy and the number of selected genes.

This study confirms once again that genetic algorithms constitute a general and valuable approach for gene selection and classification of microarray data. Its effectiveness depends strongly on how semantic information of the given problem is integrated in its genetic operators such as crossover and mutation. The role of an appropriate fitness function should not be underestimated. Finally, it is clear that the genetic approach can favorably be combined with other ranking and classification methods.

Our ongoing works include experimentations of the proposed method on more data sets, studies of alternative fitness functions and searches for other semantic information that can be used in the design of new genetic operators.

Acknowledgments. The authors would like to thank the referees for their helpful suggestions which helped to improve the presentation of this paper. This work is partially supported by the French Ouest Genopole®. The first author of the paper is supported by a Mexicain COSNET scholarship.

References

1. A. Alizadeh, M.B. Eisen, E. Davis, C. Ma, I. Lossos, A. Rosenwald, J. Boldrick, H. Sabet, T. Tran, X. Yu, J.I. Powell, L. Yang, G.E. Marti, J. Hudson Jr, L. Lu, D.B. Lewis, R. Tibshirani, G. Sherlock, W.C. Chan, T.C. Greiner, D.D. Weisenburger, J.O. Armitage, R. Warnke, R. Levy, W. Wilson, M.R. Grever, J.C. Byrd, D. Botstein, P.O. Brown, and L.M. Staudt. Distinct types of diffuse large B–cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511, February 2000.
2. U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci USA*, 96:6745–6750, 1999.
3. C. Ambroise and G.J. McLachlan. Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc Natl Acad Sci USA*, 99(10):6562–6566, 2002.
4. E. Bonilla Huerta, B. Duval, and J.-K. Hao. A hybrid ga/svm approach for gene selection and classification of microarray data. *Lecture Notes in Computer Science*, 3907:34–44, Springer, 2006.
5. B. E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 144–152, ACM Press, 1992.
6. K. Deb and A. R. Reddy. Reliable classification of two-class cancer data using evolutionary algorithms. *Biosystems*, 72(1-2):111–29, Nov 2003.

7. S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457):77–87, 2002.
8. R. O. Duda and P. E. Hart. *Pattern Classification and scene analysis*. Wiley, 1973.
9. T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.
10. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3):389–422, 2002.
11. R. Kohavi and G.H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.
12. J. Liu and H. Iba. Selecting informative genes using a multiobjective evolutionary algorithm. In *Proceedings of the 2002 Congress on Evolutionary Computation*, pages 297–302, IEEE Press, 2002.
13. E. Marchiori, C. R. Jimenez, M. West-Nielsen, and N. H. H. Heegaard. Robust svm-based biomarker selection with noisy mass spectrometric proteomic data. *Lecture Notes in Computer Science*, 3907:79–90, Springer, 2006.
14. T.K. Paul and H. Iba. Selection of the most useful subset of genes for gene expression-based classification. *Proceedings of the 2004 Congress on Evolutionary Computation*, pages 2076–2083, IEEE Press, 2004.
15. S. Peng, Q. Xu, X.B. Ling, X. Peng, W. Du, and L. Chen. Molecular classification of cancer types from microarray data using the combination of genetic algorithms and support vector machines. *FEBS Letters*, 555(2):358–362, 2003.
16. A. Rakotomamonjy. Variable selection using svm-based criteria. *Journal of Machine Learning Research*, 3:1357–1370, 2003.
17. J. Weston, A. Elisseeff, B. Scholkopf, and M. Tipping. The use of zero-norm with linear models and kernel methods. *Journal of Machine Learning Research*, 3(7-8):1439–1461, 2003.