A Study of Crossover Operators for Gene Selection 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 a subset of relevant genes in order to achieve good classification performance. Several genetic algorithms have been devised to perform this search task. In this paper, we carry out a study on the role of crossover operator and in particular investigate the usefulness of a highly specialized crossover operator called GeSeX (GEne SElection crossover) that takes into account gene ranking information provided by a Support Vector Machine classifier. We present experimental evidences about its performance compared with two other conventional crossover operators. Comparisons are also carried out with several recently reported genetic algorithms on four well-known benchmark data sets.

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, the currently available data 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 (gene selection) in order to obtain reliable predictive results.

Gene selection for microarray data is a special kind of feature selection that aims at finding a (small) subset of informative features from the initial data in order to obtain high classification accuracy [13]. Given the particular characteristic of "curse of dimensionality" of microarray data, gene selection for microarray data is known to be particularly difficult.

The literature offers a large number of solution methods for gene selection which are based on genetic algorithms, often combined with other approaches

N. Monmarché et al. (Eds.): EA 2007, LNCS 4926, pp. 243–254, 2008.

[19,6,18,17,8,16,4,22]. For instance, the so-called wrapper approach uses GAs to search over the space of gene subsets, the fitness of each subset being evaluated by its classification performance obtained by a given classifier.

In this paper, we are interested in studying the genetic algorithms for gene selection. In particular, we focus our investigation on the very role of the crossover operator. Indeed, it is now well recognized that among the different components of a GA, the crossover operator may make a difference if it is carefully designed for the targeted problem.

The main contributions of the paper is to present in details a highly specialized crossover operator called GeSeX (GEne SElection crossover) introduced in [12] and to report extensive comparative studies of GeSeX with two other conventional crossover operators (uniform and single point). These results help to understand the behavior of these crossover operators and their relative performance when they are applied with a GA. Comparisons are also carried out with several recently reported genetic algorithms on four well-known benchmark data sets.

The paper is organized as follows; in Section 2, we review the main characteristics of Support Vector Machines (SVM) that are used in our approach. In Section 3, we describe the specialized crossover operator GeSeX and the other components of our GA. Experimental results and comparaisons are presented in Section 4 before conclusions are given in Section 5.

2 SVM Classification and Gene Selection

It is common in wrapper approaches for gene selection to use a classifier to evaluate the quality of a proposed gene subset. SVM classifier can be used for such purposes. The originality of our genetic algorithm is that a SVM classifier is used not only in the fitness evaluation of gene subsets but also in the genetic operators: actually, the characteristics of the SVM classifier are used to propose a specialized crossover operator. This section recalls the main characteristics of SVM and explains how a feature selection process can be guided by the informations provided by a SVM classifier.

2.1 Support Vector Machines

SVMs have been successfully used for gene selection and classification [11,20,15]. SVMs are state-of-the-art classifiers that solve a binary classification problem by searching a decision boundary that has the maximum margin with the examples. SVMs handle complex decision boundaries by using linear machines in a high dimensional feature space, implicitly represented by a kernel function. In this work, we only consider linear SVMs because they are known to be well suited to the datasets that we consider and they offer a clear biological interpretation.

For a given training set of labeled samples, a linear SVM determines an optimal hyperplane that divides the positively and the negatively labeled samples with the maximum margin of separation. A noteworthy property of SVM is that the hyperplane only depends on a small number of training examples called the

support vectors, they are the closest training examples to the decision boundary and they determine the margin.

Formally, we consider a training set of n samples belonging to two classes; each sample is noted $\{X_i, y_i\}$ where $\{X_i\}$ is the vector of attribute values describing the sample and y_i the class label.

A soft-margin linear SVM classifier aims at solving the following optimization problem:

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

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

In this formulation, w is the weight vector that determines the separating hyperplane; C is a given penalty term that controls the cost of misclassification errors. To solve this optimization problem, it is convenient to consider the dual formulation [5]:

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

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 and the training samples with non-zero weights are the support vectors.

2.2 Feature Ranking by SVM

As discussed in [11], the weights of a linear discriminant classifier can be used to rank the features for selection purposes. More precisely, in a backward selection method, the idea is to start with all the features and to iterate the removal of the least informative feature. To determine which feature can be removed, one can consider the feature that has the least influence on the cost function of the classification process. For a linear SVM, the cost function is defined by $\frac{1}{2}||w||^2$. So given a SVM with weight vector w, we can define the ranking coefficient vector c given by:

$$\forall i, c_i = (w_i)^2 \tag{3}$$

Intuitively, that means that in order to select informative genes, the orientation of the separating hyperplane found by a linear SVM can be used. If the plane is orthogonal to a particular gene dimension, then that gene is informative, and vice versa. As we show in the next section, the coefficient vector c can provide a dedicated crossover operator with very useful ranking information.

3 A Dedicated Genetic Algorithm for Gene Selection and Classification

Our genetic algorithm for gene selection begins by a pre-selection step that enables to reduce the gene subset space. For a given microarray dataset, we first

filter the most interesting genes by the BW ratio criterion introduced in [7]; the number p of pre-selected genes is fixed at 50. ¿From this reduced subset, we will determine an even smaller set of genes (typically < 10) which allows to give the highest classification accuracy. To achieve this goal, we propose a dedicated Genetic Algorithm which integrates, in its genetic operators, specific knowledges on our gene selection and classification problem. It relies on a linear SVM classifier to evaluate each individual and the ranking coefficient vector of this SVM enables to propose a highly informed crossover operator. In what follows, we present the main elements of this GA, focusing on the crossover operator. Other characteristics of our approcah can be found in [12].

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 3) 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 the evolutonary process.

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 abily to obtain a good classification in this gene subset representation and the number of genes contained in this subset. More formally, the fitness function is defined as follows:

$$f\,(I) = \frac{CA_{SVM}(I^x) + \left(1 - \frac{|I^x|}{p}\right)}{2} \tag{4}$$

The first term ($CA_{SVM}(I^x)$) is the classification accuracy obtained with a linear SVM classifier trained on this subset and evaluated 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 ranking vector c obtained from the SVM classifier is calculated and copied in I^y.

3.3 Specialized Crossover Operator [12]

Crossover is one of the key evolution operators for any effective GA and needs a particularly careful design. As our goal is to obtain small subsets of selected

genes with a high classification accuracy, we have designed a highly specialized crossover operator 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 stored in I^y. 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.

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 $K =< K^x, K^y >$ using the following steps:

1. Extract the subset of genes shared by both parents by boolean logic AND operator ($\otimes$) 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, 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 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 4. 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', we first copy the two best individuals of P 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 $|P|$ candidate individuals. From this pool, crossover is applied $0.49 * |P|$ times to pairs of randomly taken individuals, each new resulting individual being inserted in P'. Similarly, random mutation is applied $0.49 * |P|$ 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 Comparison

In this section we present two comparative studies. The first compares the crossover operator GeSeX with two well-known crossover operators. In the second study, we carry out a comparison with four highly effective GA-based gene selection approaches [17,22,8,16].

4.1 Data Sets

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

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 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://microarray.princeton.edu/oncology/affydata/index.html and was first studied in [2].

The first lymphoma data set is based on 4026 variables describing 47 samples (24 and 23 of which are respectively considered as GC B-Like samples and activated B-Like samples). The data set was first analyzed in [1]. The data set is available at http://llmpp.nih.gov/lymphoma/data.shtml.

The second lymphoma data set contains 58 patients with DLBCL each with 7129 gene expression values, 32 with cured disease and 26 with fatal or refractory disease. This is available at http://broad.mit.edu/cgi-bin/cancer/datasets.cgi. The data set was reported in [21].

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 Comparison of Crossover Operators

The purpose of the first experiment is to evaluate the performance of two well known crossover operators (single point and uniform crossovers) against our GeSeX crossover operator. The evaluation takes into account two aspects: the capacity to generate new potentially promising individuals and the ability to keep a diversified population. Both characteristics are very important in the whole search process because they represent the classical trade-off between exploration and exploitation.

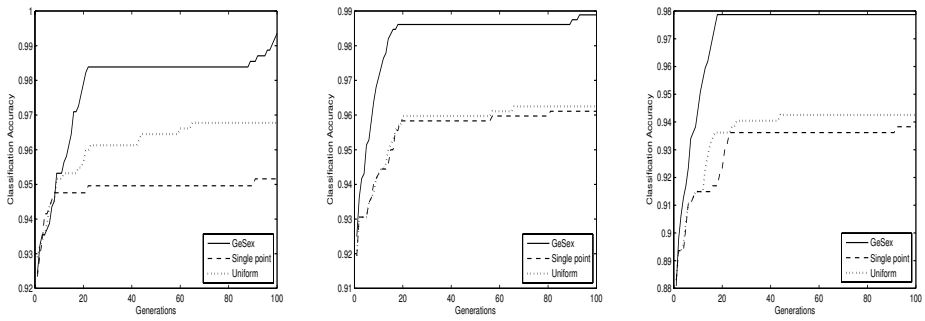
The first criterion is measured by the quality of the best individual of a population. For an individual, that is a gene subset, we measure its quality by the classification accuracy of a SVM classifier built on this gene subset. This accuracy is evaluated via a 10-fold cross validation, so for an individual I, it is exactly $CA_{SVM}(I^x)$ (see Section 3). The population diversity is calculated with the entropy measure proposed in [10] and recalled in Equation 5, where n_{ij} represent the number of times the gene i is set to the value j in the population P. This function takes values in the interval [0, 1]. An entropy of 0 indicates that all the individuals in the population are identical, while an entropy of 1 means that all the individuals are different.

$$Entropy\,(P) = \frac{\sum_{i=1}^{n} \sum_{j=0}^{1} \left(\frac{n_{ij}}{|P|}\right) log\left(\frac{n_{ij}}{|P|}\right)}{nlog2} \tag{5}$$

In order to enable a fair comparison, all the crossover operators were tested under the same conditions on three microarray datasets (Colon, Leukemia and the first lymphoma data set [1]). The following parameters were used in this experiment: a) population size $|P| = 100$, b) maximal number of generations is fixed at 100. We use a classical mutation where each bit of an individual has a mutation probabilty of 0.3. For the single point and uniform crossover operators, we use a crossover probability of 0.5, whereas the general settings for GeSeX operator are explained in subsection 3.4.

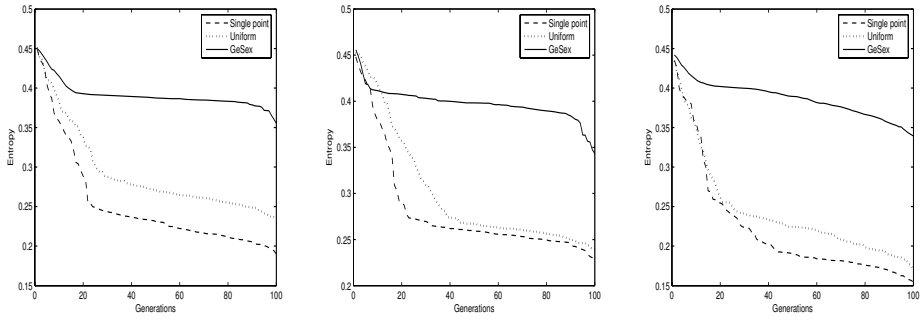
Due to the non deterministic character of GA, 10 independent runs were executed for each dataset/operator combination. The results are shown in figure 1 and in figure 2.

In figure 1 the X axis represents the number of generations, while the Y axis represents the accuracy of the best individual of a population, both averaged over the 10 runs. This figure shows clearly that the GeSeX operator allows us to obtain better results for the three datasets because it constantly reaches a higher classification accuracy. More specifically, let us examine the case of the Leukemia dataset. With the GeSeX operator, an average accuracy of 98.611% is rapidly reached by the best individual within 20 generations, meaning that for each of the 10 experiences, only one sample out of the 72 is misclassified in the cross-validation process. With the two other crossover operators, an average accuracy is onlt around 96% because in most experiences 3 examples out of 72 is misclassified and for one or two experiences, two samples out of 72 are misclassified. We can notice also that after 90 generations, the curve for GeSeX



a) Colon b) Leukemia c) Lymphoma

Fig. 1. Average classification accuracy of the best individuals of populations for Single point, Uniform and GeSeX crossover operators using three microarray datasets



a) Colon b) Leukemia c) Lymphoma

Fig. 2. Population entropy for Single point, Uniform and GeSeX crossover operators

leaves the stage of 98.611% because for one or two experiences among the 10, the best individual reaches the maximal accuracy of 100%.

In figure 2 we show how the population entropy evolves with the number of generations. Each point represents the average population entropy over all runs. Observe that GeSex keeps a higher population entropy that the other crossover operators. Therefore GeSex provides a good balance between quality and diversification of the population.

4.3 Comparison with Other Genetic Algorithms

In this section we carry out a comparison of our GA+GeSeX with four highly effective GA-based gene selection approaches.

Genetic Approaches. In [17] the authors propose a gene selection method that relies on two steps. The first step is a pre-selection that rank the genes according to an original filtering criterion proposed by the authors; the top

genes are selected to construct a reduced search space, that the GA explores in order to minimize the number of selected genes. Their GA is a classical one with a multiple-point crossover. The paper reports the best classification accuracies estimated by LOOCV on the whole set of samples for a single run. Notice that the lymphoma data set is the one analysed in [21] and they find an final subset with 11 selected genes. For the colon cancer, they report a subset with 9 selected genes and for the leukemia data set they select a subset of 8 genes.

In [22] the authors propose a hybrid algorithm using SVM and GA. In the first step of their approach, a gene subset of size p is selected by Least Square Support Vector Machine to construct the search space of the GA. In the second step, they apply a GA to carry out gene selection. The particularity of their GA is that crossover and mutation operators are designed to keep the same number p of genes. So their objective is to explore all the subsets of size p in order to find the best one. The fitness function of a gene subset uses the information entropy of the classes represented on that gene subset. When the GA terminates, they evaluate the quality of the selected gene subset by the accuracy of a SVM classifier. For colon cancer, the test set has 32 samples and the best accuracy (over one run) is obtained with 20 selected genes. For leukemia, the test set has 34 samples and their best result is obtained with 15 selected genes.

In [8] the authors propose a genetic method that is not a wrapper approach: the GA explores the space of subsets and each candidate subset is evaluated by two clustering measures. The idea is to consider the two classes of the data set as two given clusters and to compare the quality of the clusters when the gene subset used to represent the data is changed. Such a GA-Filter approach requires a lower computational burden since the fitness evaluation does not require a classifier training. For each data set, 10 runs of GA-Filter are executed and each time, the gene subset selected by GA-Filter is evaluated by a classification experiment where different classifiers are tried. The paper presents the average and standard deviation of the classification accuracy over these 10 runs. We retain for comparison the best result reported in the paper, for each dataset that we consider. Notice that the lymphoma is the one presented in [1]. The number of selected genes were respectively 15, 17 and 10 together with respectively 34, 22, and 13 testing samples for the leukemia, colon and lymphoma datasets.

In [16] the authors combine SVM and GA in another way. Their SVM uses a kernel function that combines a set of simple kernel functions and they propose a new learning method exploiting Evolutionary Algorithm technique to obtain an optimal decision model. So their genetic search aims to find out the optimal set of features but also the optimal set of parameters for the combined kernel function. The average of the classification accuracy over 10 independent runs is provided for colon and leukemia datasets. The number of selected genes were 15 in both datasets.

Experimental Context and Results of Comparison. In order to compare our approach with each of these four works, we apply our genetic algorithm to each data set with exactly the same experimental conditions as those reported in the corresponding paper. More precisely, we fix the number of genes in

Table 1. Comparison of four GA-based selection approaches and our method. The table gives the number of genes and the classification accuracy reported by each author (*Reported*) and the classification accuracy obtained by our approach (**GeSeX**) when we fix the number of genes to the value used in the corresponding paper.

	[17]			[22]			[8]			[16]		
Data set	*Reported*		**GeSeX**	*Reported*		**GeSeX**	*Reported*		**GeSeX**	*Reported*		**GeSeX**
Leukemia	*8*	*98.6*	**100**	*15*	*97.1*	**100**	*15*	*99.70*	**98.82**	*15*	*77.06*	**98.82**
Colon	*9*	*95.1*	**100**	*20*	*90.6*	**93.75**	*17*	*77.50*	**85.9**	*15*	*75.33*	**86.0**
Lymph.[1]	-		-	-		-	*10*	*96.15*	**96.92**	-		-
Lymph.[21]	*11*	*100*	**100**	-		-	-		-	-		-

our method, that means that for each data set and each previously cited work [17,22,8,16], we determine which classification accuracy can be obtained by our GA for the number of genes reported in this work. Moreover, we evaluate the classifier accuracy with the same number of runs: for [17] and [22], the result is the best accuracy obtained in one run while for [8] and [16], this is the average over 10 runs. We also use the same test samples as the authors for each dataset, this is important because previous studies have shown that the accuracy estimate may be biased and may have an important variance [3]. In this experiment, our genetic algorithm uses also a specialized mutation operator [12] that uses ranking information provided by the SVM and stored in the ranking coefficient vector I^y to eliminate "mediocre" genes.

Table 1 summarizes the comparison: the number of genes and the classification accuracy reported in the papers are in front of the classification accurary obtained by our method. Some cells of the table contain no information because the experiment on the corresponding data set is not available in the papers.

From Table 1, we observe that the results of our GA are better than those published results, except for the result of leukemia reported in [8]. As indicated, in these experiments we restrict our method to consider the same number of selected genes as in the reported works. In fact, our method is able to optimize two criteria: the number of selected genes and the classification accuracy. So, our method is able to select smaller subsets of informatives genes with high classification accuracy. Concretely, we have experimented our method on 50 trials for Leukemia and Colon data sets and we obtain the following results [12]. For Leukemia, the number of selected genes was respectively 3.17±1.16 and the accuracy (evaluated by a 10-fold cross validation) was 91.5±5.9; for Colon, the number of selected genes was 7.05±1.07 and the accuracy was 84.6±6.6. Those numbers cannot be compared with [17] and [22], which do not provide averaged results but they are comparable with those of [16,8] and better in the sense that the number of genes is smaller.

5 Conclusions and Future Work

We have presented a study on the role of the crossover operators for gene selection of microarray data. We have presented a specialized crossover operator

GeSeX that is used in a wrapper genetic algorithm. Contrary to conventional crossover operators, GeSex takes into account the information provided by the SVM classifier used by our fitness function.

Our experimental analysis shows that this crossover operator behaves more efficiently than traditional crossover operators and that it ensures a good trade-off between exploration and exploitation of the search space. We also compare our GA+GeSeX approach to other recently proposed GA devoted to the task of gene selection and classification of microarray data. These experimentations show that GA+GeSex gives globally very competitive results.

We are currently studying alternative fitness functions to provide a more effective guidance of the genetic process. Moreover we are developing a local search based mutation operator in order to intensify the genetic search.

Acknowledgments. This work is partially supported by the French Ouest Genopole® and the "Bioinformatique Ligérienne" projects. The first author is supported by a Mexicain PROMEP-DGEST scholarship. The authors of this paper would like to thank the reviewers for their useful comments.

References

1. Alizadeh, A., Eisen, M.B., Davis, E., Ma, C., Lossos, I., Rosenwald, A., Boldrick, J., Sabet, H., Tran, T., Yu, X., Powell, J.I., Yang, L., Marti, G.E., Hudson Jr, J., et al.: Distinct types of diffuse large B–cell lymphoma identified by gene expression profiling. Nature 403, 503–511 (2000)
2. Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proceedings of the National Academy of Sciences USA 96, 6745–6750 (1999)
3. Ambroise, C., McLachlan, G.J.: Selection bias in gene extraction on the basis of microarray gene-expression data. Proceedings of the National Academy of Sciences USA 99(10), 6562–6566 (2002)
4. Bonilla Huerta, E., Duval, B., Hao, J.-K.: A hybrid ga/svm approach for gene selection and classification of microarray data. In: Rothlauf, F., Branke, J., Cagnoni, S., Costa, E., Cotta, C., Drechsler, R., Lutton, E., Machado, P., Moore, J.H., Romero, J., Smith, G.D., Squillero, G., Takagi, H. (eds.) EvoWorkshops 2006. LNCS, vol. 3907, pp. 34–44. Springer, Heidelberg (2006)
5. Boser, B.E., Guyon, I., Vapnik, V.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144–152. ACM Press, New York (1992)
6. Deb, K., Reddy, A.R.: Reliable classification of two-class cancer data using evolutionary algorithms. Biosystems 72(1-2), 111–129 (2003)
7. Dudoit, S., Fridlyand, J., Speed, T.P.: 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. Feres de Souza, B., de Carvalho, E.C.P.L.F.: Gene Selection Using Genetic Algorithms. In: Barreiro, J.M., Martín-Sánchez, F., Maojo, V., Sanz, F. (eds.) ISBMDA 2004. LNCS, vol. 3337, pp. 479–490. Springer, Heidelberg (2004)

9. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science 286, 531–537 (1999)
10. Grefenstette, J.J.: Incorporating Problem Specific Knowledge into Genetic Algorithms. In: Davis, L. (ed.) Genetic Algorithms and Simulated Annealing, pp. 42–60. Morgan Kaufmann Publishers, London (1987)
11. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine Learning 46(1-3), 389–422 (2002)
12. Hernandez Hernandez, J.C., Duval, B., Hao, J.-K.: A genetic embedded approach for gene selection and classification of microarray data. In: Marchiori, E., Moore, J.H., Rajapakse, J.C. (eds.) EvoBIO 2007. LNCS, vol. 4447, pp. 90–101. Springer, Heidelberg (2007)
13. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artificial Intelligence 97(1-2), 273–324 (1997)
14. Liu, J., Iba, H.: Selecting informative genes using a multiobjective evolutionary algorithm. In: Proceedings of the 2002 Congress on Evolutionary Computation, pp. 297–302. IEEE Press, Los Alamitos (2002)
15. Marchiori, E., Jimenez, C.R., West-Nielsen, M., Heegaard, N.H.H.: Robust svm-based biomarker selection with noisy mass spectrometric proteomic data. In: Rothlauf, F., Branke, J., Cagnoni, S., Costa, E., Cotta, C., Drechsler, R., Lutton, E., Machado, P., Moore, J.H., Romero, J., Smith, G.D., Squillero, G., Takagi, H. (eds.) EvoWorkshops 2006. LNCS, vol. 3907, pp. 79–90. Springer, Heidelberg (2006)
16. Nguyen, H.-N., Ohn, S.-Y., Park, J., Park, K.-S.: Combined Kernel Function Approach in SVM for Diagnosis of Cancer. In: Wang, L., Chen, K., S. Ong, Y. (eds.) ICNC 2005. LNCS, vol. 3610, pp. 1017–1026. Springer, Heidelberg (2005)
17. Ni, B., Liu, J.: A Novel Method of Searching the Microarray Data for the Best Gene Subsets by Using a Genetic Algorithm. In: Yao, X., Burke, E.K., Lozano, J.A., Smith, J., Merelo-Guervós, J.J., Bullinaria, J.A., Rowe, J.E., Tiňo, P., Kabán, A., Schwefel, H.-P. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 1153–1162. Springer, Heidelberg (2004)
18. Paul, T.K., Iba, H.: Selection of the most useful subset of genes for gene expression-based classification. In: Proceedings of the 2004 Congress on Evolutionary Computation, pp. 2076–2083. IEEE Press, Los Alamitos (2004)
19. Peng, S., Xu, Q., Ling, X.B., Peng, X., Du, W., Chen, L.: 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)
20. Rakotomamonjy, A.: Variable selection using svm-based criteria. Journal of Machine Learning Research 3, 1357–1370 (2003)
21. Shipp, M.A., Ross, K.N., Tamayo, P., Weng, A.P., Kutok, J.L., Aguiar, R.C., Gaasenbeek, M., Angelo, M., Reich, M., Pinkus, G.S., Ray, T.S., et al.: Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nature Medecine 8(1), 68–74 (2002)
22. Xiong, W., Zhang, C., Zhou, C., Liang, Y.: Selection for Feature Gene Subset in Microarray Expression Profiles Based on a Hybrid Algorithm Using SVM and GA. In: Min, G., Di Martino, B., Yang, L.T., Guo, M., Rünger, G. (eds.) ISPA Workshops 2006. LNCS, vol. 4331, pp. 637–647. Springer, Heidelberg (2006)