GENE SELECTION FOR CANCER CLASSIFICATION USING WRAPPER APPROACHES

ROSA BLANCO, PEDRO LARRAÑAGA, IÑAKI INZA and BASILIO SIERRA

Computer Science and Artificial Intelligence Department,
University of Basque Country, P.O.Box 649, 20080 San Sebastián

Despite the fact that cancer classification has considerably improved, nowadays a general method that classifies known types of cancer has not yet been developed. In this work, we propose the use of supervised classification techniques, coupled with feature subset selec-tion algorithms, to automatically perform this classification in gene expression datasets. Due to the large number of features of gene expression datasets, the search of a highly accurate combination of features is done by means of the new Estimation of Distribution Algorithms paradigm. In order to assess the accuracy level of the proposed approach, the *naïve-Bayes* classification algorithm is employed in a wrapper form. Promising results are achieved, in addition to a considerable reduction in the number of genes. Stating the optimal selection of genes as a search task, an automatic and robust choice in the genes finally selected is performed, in contrast to previous works that research the same types of problems.

Keywords: Feature subset selection; DNA microarrays; supervised classification; *naïve-Bayes*; estimation of distribution algorithms.

1. Introduction

Cancer classification is based basically on the morphological appearance of the tumor. However, tumors with a similar appearance present different responses to therapy. This fact makes a correct cancer classification very important. The gene expression data can be used to learn classification models to aid cancer classification. Taking into account that one pattern only belongs to one class (or type of cancer), the probabilistic approach to the supervised classification problem is reduced to find c^* such as:

$$c^* = \arg\max_c p(C = c \mid X_1 = x_1, \ldots, X_n = x_n)$$

where C is the cancer class feature and X_i $(i = 1, 2, \ldots, n)$ is the variable related to the ith gene expression data. Nevertheless, depending on the model and the

number of features (and their values) of the data set, the solution of the previous problem might require a large number of instances in order to reliably estimate the parameters needed to learn the joint probability distribution.

The previous approach to cancer classification is known as class prediction, that is, the assignment of tumor cases to already known cancer types. Cancer classification involves another task: class discovery or the finding of the unknown types of cancer in a data set. Class discovery is related to the field known as unsupervised classification or cluster analysis and class prediction is related to supervised classification. In this work, we focus on class prediction and propose techniques connected with supervised classification.

During the last few years, the number of biological data sets has grown spectacularly due to the advances on data acquisition technologies and the advances in digital storage and computing. Genome sequencing is one of the biological techniques with the most improvement. These advances have led to the development of the *DNA microarray*. DNA microarrays have drastically changed biological and medical research. Now, it is possible for the observation and the measurement of the expressions levels of thousands of genes simultaneously in an organism. A systematic and computational analysis of these microarray datasets is a new and interesting way of understanding the underlying biological processes.

DNA microarray examples are obtained by the hybridization of the studied tissues to the microarray, binding them to the complementary probes affixed to the microarray surface. The arrays are then scanned, producing a fluorescent image: this fluorescent intensity at any particular probe location indicates the relative concentration of the complementary DNA sequence in the tissue.

DNA microarray datasets can be an appropriate starting point to carry out systematic and automatic cancer classification[10] as a result of the techniques implied with the analysis of gene expression datasets. This analysis[7] involves class prediction, regression, feature selection, outlier detection, principal component analysis, discovering of gene relationships and cluster analysis.

On the other hand, gene expression data from DNA microarrays are characterized by a large number of genes (or variables or features) on few experiments. The number of genes in a single array are typically in the thousands. Thus, the question is whether all features (or genes) are "useful" to correctly classify new instances. The Feature Subset Selection problem (FSS) tries to answer this question by searching for the best subset of features for a data set and a learning algorithm.[2,17]

Obviously, FSS has several advantages. A number of them being the improvement of the comprehensibility of the final classification model, its faster induction, and an improvement in classification accuracy.

Several classification algorithms can be chosen to solve the supervised classification problem. *Naïve-Bayes*[9] is a paradigm based on the conditional independence of the predictive features given the class. Thus, the number of parameters to estimate the joint probability distribution is considerably reduced.

The aim of this work, i.e. a feature subset selection to maximize the classification model accuracy, can be expressed in the form of a search problem[19] with the objective function being the accuracy of the proposed subset of genes. In our work, the search engine is the novel Estimation of Distribution Algorithms (EDAs).[21] EDAs have been successfully used in similar FSS problems.[12] However, due to the large number of genes involved in the DNA microarray datasets, the initialization of the genes' probabilities is a crucial point: four types of initializations are proposed, three of them based on the results of a classic greedy search algorithm. To guide the search, a wrapper approach over *naïve-Bayes* is used. In previous works[4,10,30] these kinds of problems were not based on a search task, but they perform a somewhat arbitrary choice in the finally selected number of genes. In the actual paper, an automatic and robust choice is performed with the use of a search technique.

Two different well-known gene expression datasets are used to test the proposed approach. The first dataset, related to colon cancer,[3] has 62 instances involving 2,000 predictive features or genes (gene expression length) and the class indicates whether the patient suffers from cancer or not. The second dataset is related to leukemia[10]: 72 instances containing 7,129 predictive features or genes (gene expression length) are presented and the class shows the kind of leukemia suffered: AML or ALL. The experimental results suggest that the accuracy of *naïve-Bayes* classifier is improved (better than 90%) with a significant reduction in the number of features involved in the learning (less than 20 in all runs).

The work is organized as follows: the next section presents the wrapper approach, *naïve-Bayes* supervised paradigm and EDAs. Section 3 presents the integration of these elements to carry out FSS, employing four different initialization methods. Section 4 shows the experimental results. We end up with conclusions and suggested future work.

2. Wrapper, Naïve-Bayes and EDA Paradigms

2.1. *The wrapper approach*

In all problem domains, irrelevant features can degrade the predictive accuracy of learning algorithms. Features, whose information contribution is overlapped or repeated, can act in the same way. Algorithms such as *naïve-Bayes* are robust with respect to irrelevant features but very sensitive to correlated features.

This lack of accuracy can be improved if the learning algorithm only uses adequate features.[19] For this purpose, a feature selection process is required. FSS can be used to find a feature subset that maximizes the predictive accuracy of the classification model built over this subset. From this point of view, FSS can be faced as a search problem where each point of the search space represents a feature subset.[19]

The aim of the search is to maximize the performance of the classifier. A number of evaluation functions carry out this goal by looking only at the intrinsic characteristics of the data and measuring the power to discriminate among the classes of the

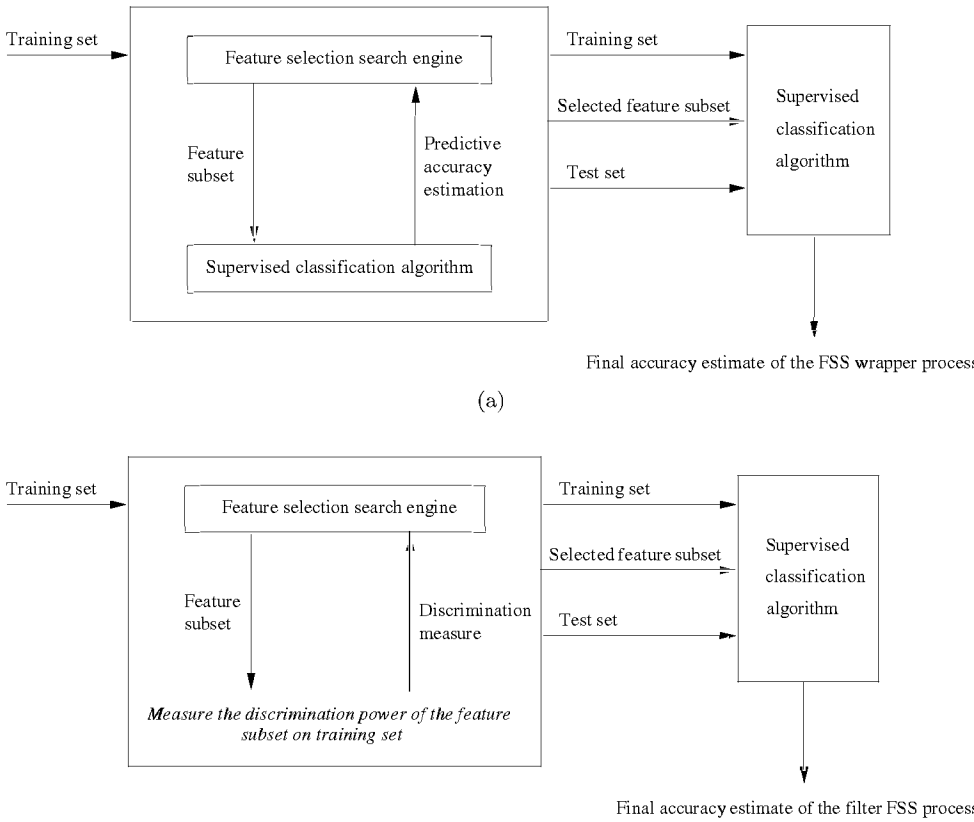


Fig. 1. General schemes for feature subset selection: (a) *Wrapper* and (b) *Filter* approaches.

problem involved. These kinds of evaluation functions are known as *filter* functions. However, Kohavi and John[17] reported that when the goal is to maximize the accuracy of the classification model, FSS should depend not only on the features and the concept to learn, but also on the characteristics of the classifier. This allows for the development of the *wrapper* approach: when a feature subset is selected by the search algorithm, its predictive accuracy is estimated with respect to the supervised classification algorithm proposed to generate the final model. Figure 1 shows the differences between the wrapper and filter approaches.

The wrapper approach is not very popular for DNA microarrays but a few works use it[14,22,30] in order to improve the final accuracy of the classification model.

2.2. *The naïve-Bayes paradigm*

The goal of a supervised classification algorithm is to build a classification model using a data set. This model is used to predict the class of new instances. From a probabilistic perspective, the class chosen, c^*, for a given new instance will be the

class with the highest *a posteriori* probability, given the values of the predictive features:

$$c^* = \arg\max_c p(C = c \mid X_1 = x_1, \ldots, X_n = x_n) .$$

The cost of the estimation of the class depends on the complexity of the model and the assumptions over the data.

Naïve-Bayes is a supervised classification algorithm built over the assumption of conditional independence of the predictive features given the class. Although this assumption is violated in numerous occasions, this fact does not degrade the performance of the paradigm in many situations.[8,11] Under this assumption, the prediction of the class for an unseen instance is simplified.

When the predictive features are discrete the predicted class for an unseen $x = (x_1, x_2, \ldots, x_n)$ test instance is as follows:

$$c^* = \arg\max_c p(C = c) \prod_{i=1}^{n} p_{X_i|C=c}(x_i)$$

where $p_{X_i|C=c}(x_i)$ represents the conditional probability of $X_i = x_i$ given that $C = c$.

In the case that the predictive features are continuous:

$$c^* = \arg\max_c p(C = c) \prod_{i=1}^{n} f_{X_i|C=c}(x_i)$$

where $f_{X_i|C=c}(x_i)$ represents the density function of the ith feature conditioned on $C = c$. In this work, we assume that the previous density conditioned functions follow a normal distribution. That is, for all $i = 1, \ldots, n$ and $c = 0, 1$:

$$f_{X_i|C=c}(x_i) \rightsquigarrow \mathcal{N}(\mu_i^c, \sigma_i^c) .$$

In both cases, with predictive features (either discrete or continuous), the parameters are estimated by means of their maximum likelihood estimates. In the case of discrete features, the parameters are calculated from its relative frecuencies. In the case of continuous features, the parameters are determined using the means and the sample variance of the corresponding feature conditioned to class value.

2.3. *Estimation of distribution algorithms*

A new approach in evolutionary computation to solve optimization problems is Estimation of Distribution Algorithms (EDAs).[20,21,26] This birth is motivated by the difficulty in choosing the optimal parameters in Genetic Algorithms and the impossibility to predict the movements of the populations in the search space.[20]

Although they are based on populations, there are neither crossover nor mutation operators in EDAs. Instead, the new population of individuals is sampled from a probability distribution, which is learnt from a number of selected individuals for each generation.

1. $D_0 \leftarrow$ Generate M individuals randomly (the initial population)

2. Repeat for $l = 1, 2, \ldots$ until the stopping criterion is met:

2.1. $D_{l-1}^{Se} \leftarrow$ Select $N < M$ individuals from D_{l-1} according to the selection method

2.2. $p_l(\mathbf{x}) = p(\mathbf{x}|D_{l-1}^{Se}) \leftarrow$ Estimate the probability distribution of selected individuals

2.3. $D_l \leftarrow$ Sample M individuals from $p_l(\mathbf{x})$ (the new population)

Fig. 2. Pseudo-code for EDA approach.

Figure 2 shows the basic scheme of the EDA paradigm. In the first step, M individuals are generated at random, for example, from a uniform distribution for each feature. These M individuals constitute the initial population D_0, and each is evaluated. In an iterative process until the stopping criterion is met, we repeat the following steps: first, a number N ($N < M$) of individuals are selected usually those with the best objective function values. Second, a n-dimensional probability distribution is learned from the selected individuals. Finally, M new individuals (the new population) are obtained from sampling the probability distribution learnt in the previous step.

The estimation of the joint probability distribution associated to the selected individuals is the bottleneck of EDAs. Different ways in estimating this joint probability exist, with different assumptions on the interrelations among the features of the individuals.

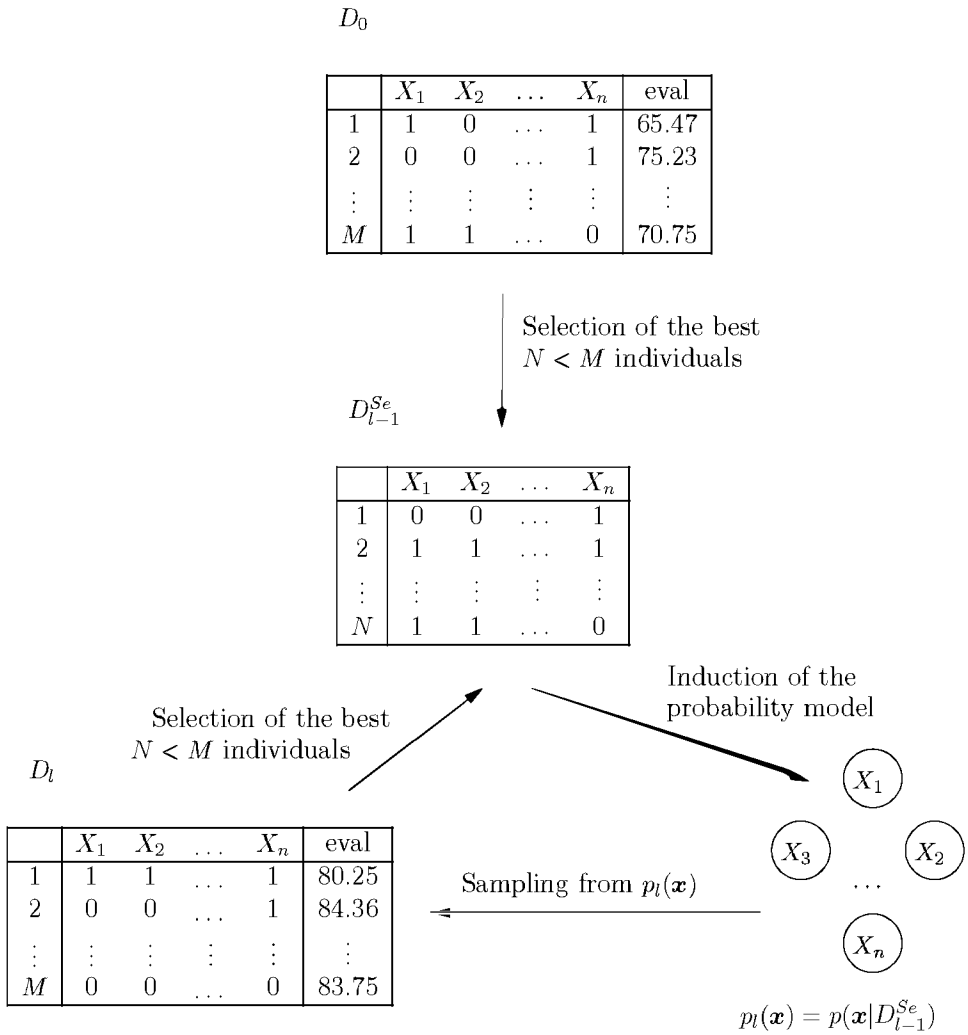
The simplest assumption that can be made over the variables is their independence. In this way, the new individuals can be generated by sampling from the univariate probability distribution of each variable. The Univariate Marginal Distribution Algorithm (UMDA)[25] works in this way. It estimates the joint probability distribution of the selected individuals at each generation, $p_l(\mathbf{x})$ in the following form:

$$p_l(\mathbf{x}) = p(\mathbf{x} \mid D_{l-1}^{Se}) = \prod_{i=1}^{n} p_l(x_i \mid D_{l-1}^{Se}) = \prod_{i=1}^{n} \frac{\sum_{j=1}^{N} \delta_j(X_i = x_i \mid D_{l-1}^{Se})}{N}$$

where

$$\delta_j(X_i = x_i \mid D_{l-1}^{Se}) = \begin{cases} 1 & \text{if in the } j\text{th case of } D_{l-1}^{Se},\ X_i = x_i \\ 0 & \text{otherwise}. \end{cases}$$

That is, the joint probability distribution of the selected individuals at each generation, $p_l(\mathbf{x})$ is factorized as a product of independent univariate marginal distributions. Each univariate marginal distribution is estimated from marginal frequencies.



	X_1	X_2	...	X_n	eval
1	1	0	...	1	65.47
2	0	0	...	1	75.23
⋮	⋮	⋮	⋮	⋮	⋮
M	1	1	...	0	70.75

	X_1	X_2	...	X_n
1	0	0	...	1
2	1	1	...	1
⋮	⋮	⋮	⋮	⋮
N	1	1	...	0

	X_1	X_2	...	X_n	eval
1	1	1	...	1	80.25
2	0	0	...	1	84.36
⋮	⋮	⋮	⋮	⋮	⋮
M	0	0	...	0	83.75

Fig. 3. FSS by the EDA (UMDA) approach.

Due to the wide dimension of our genomic databases, the use of an EDA approach covering the interrelations of order, two or more among the variables of the problem, is discarded.[20] Moreover, the number of parameters needed to estimate these multivariate relations might also be large.

Figure 3 reviews the proposed approach to select features by means of the EDA (UMDA) algorithm.

3. Proposed Approach

Taking the wide dimension of the problem into account, an appropriate initialization of the search can save a great deal of computation time.[1] In this work, the search

initialization is based on the simulation of a probability distribution for each feature or gene. We compare four different initializations, three of them based on the results of a FSS greedy algorithm and *naïve-Bayes*.

Greedy algorithms are deterministic algorithms, that is, over a fixed dataset and with the same initial conditions, always give the same solution. Sequential Forward Selection (SFS)[16] is a classic greedy search algorithm which starts from an empty subset of features and sequentially selects features until no improvement is achieved in the evaluation function value. In this work, the objective function is the estimated accuracy of the classification model built with the selected features.

Based on the feature subset obtained by SFS, three initializations for EDAs are proposed:

- init-A
  This initialization assigns the same probability to all the features of the dataset. This probability is calculated taking the number of selected genes by SFS into account. In the individuals of the first population of EDAs, all the genes have the same probability of being chosen.
- init-B
  In this case, all the genes of the dataset are handled in the same way. The probability assigned to each gene G_i, is determined by means of the estimated accuracy of the classification model built only with the class variable and G_i (this means, that the rest of genes or features are rejected and the assigned probability is proportional to $\mathrm{Acc}(G_i)$). With init-B, the genes with a higher estimated accuracy appear more frequently in the individuals of the first population of EDAs.
- init-C
  Finally, init-C differentiates between the selected features by the SFS and the non-selected features. The selected features are assigned with a probability proportional to the improvement of the estimated accuracy when added to the classification model. That is, if $\mathcal{M}_t$ is the classification model with t features and the feature G_i is added to the classifier then, G_i is assigned with a probability proportional to $\mathrm{Acc}(\mathcal{M}_t \cup G_i) - \mathrm{Acc}(\mathcal{M}_t)$.

  The non-selected features have a probability of being chosen in the first population of EDAs proportional to $1 - \mathrm{Acc}(\mathcal{M})$, where $\mathcal{M}$ is the classification model built with the features selected by SFS.

It must be noted that for three initializations methods, the expected number of selected genes in each individual of the first population of EDAs is the number of features finally selected by SFS.

Apart from these initializations, the init-0 is not dependant on the feature subset obtained by SFS. In this initialization, each feature or gene is chosen with a probability of 0.5. This means that, in the individuals of the first population of EDAs, the expected number of selected genes is the half of the total number of genes.

In the proposed EDA approach, the population size is fixed to 100 individuals, and 50 individuals are selected in order to learn the probability distribution. The search stops when the sum of the scoring function of the previous population is equal to the sum of the scoring function of the current population.

Each solution is evaluated to measure the accuracy of the built model by means of *leave-one-out*.[29] If we denote the number of instances as n_c, this kind of cross-validation builds a model with $n_c - 1$ instances of the dataset and tests it with the remaining instance, leaving out one different instance n_c times as a test set. The accuracy of the classification model built with the n_c instances is estimated by the percentage of correctly classified instances obtained with the n_c models induced with $n_c - 1$ instances.

4. Experimentation in Oncology

The proposed approach has been carried out over two well-known biological data sets. The first was presented by Alon *et al.* in 1999.[a] This data set is composed of 62 instances of colon cancer patients. Each instance is characterized by 2,000 predictive genes, each being related to the numeric expression of a certain gene. The task to be predicted is if patients suffer from colon cancer disease.

The second data set was proposed by Golub *et al.* in 1999.[b] It contains 72 instances of leukemia patients involving 7,129 predictive genes, each being related to the numerical expression of a certain gene. The class to be predicted is the specific type of leukemia: AML or ALL.

The aim of this work is to reach the best accuracy in order to classify the data sets. Due to the fact that nothing is known on the separability of classes in both data sets, we focus our empirical study on the accuracy and the number of generations required.

In order to validate the built model by means of a leave-one-out cross-validation, we merge the train set and the test set used in the literature into one data set.

For the discrete *naïve-Bayes* models, each feature is discretized into two values taking its corresponding median into account.

Although, Colon and Leukemia data sets are well-known sets used previously in the literature, an exhaustive comparison among the results of this work and the results of the literature is not a fair comparison due to the different methodologies applied. However, competitive results are achieved with both datasets. The results for Leukemia can be consulted, for instance, in Refs. 6, 15, 23, 27 and 30. For Colon dataset, the following papers can be seen: Refs. 5, 6 and 15.

Table 1 shows the results of *leave-one-out* with all the features of the problem domain and with the features selected by SFS. The results support the fact that

[a] `http://microarray.princeton.edu/oncology/affydata/index.html`.
[b] `http://www-genome.wi.mit.edu/cgi-bin/cancer/publications/pub_paper.cgi?mode=view&paper_id=43`.

Table 1. Results of *Leave-One-Out* with all the features and with the features selected by SFS.

DATA	TYPE	ALL FEATURES		SFS	
		Accuracy	n. feat.	Accuracy	n. feat.
Colon	Discrete	70.97	2000	91.93	5
	Continuous	53.23	2000	95.83	3
Leukemia	Discrete	63.89	7129	98.61	6
	Continuous	84.72	7129	87.09	2

Table 2. Best estimated accuracy and corresponding number of features.

	Colon				Leukemia			
	Discrete		Continuous		Discrete		Continuous	
INIT.	ACC.	FEA.	ACC.	FEA.	ACC.	FEA.	ACC.	FEA.
init-0	67.74	985	74.19	1069	45.8	3402	76.39	3587
init-A	95.16	13	98.39	6	100	8	100	10
init-B	95.16	13	98.39	10	98.61	15	100	11
init-C	91.93	5	95.16	3	98.61	6	98.61	4

not all the features are relevant in order to learn the classification model or the existence of redundant features.

These results follow the findings of Refs. 10 and 30, relating the low number of features needed to improve the accuracy of the whole feature set.

For each dataset and initialization method, ten EDA independent runs have been executed. Table 2 shows the estimated accuracy of *naïve-Bayes* and the number of selected features for the best run of each initialization method. Table 3 shows the estimated average accuracy, the number of selected features for the ten executions of each initialization method and the average generation where the best solution on the execution is shown.

Although EDAs, in the continuous model, do not report a significant accuracy improvement with respect to SFS in the Colon dataset, the opposite behavior, obtaining a significant accuracy improvement by EDA techniques, is shown in Leukemia domain. However, the use of an extremely low number of features is not recommended in previous works[10]: this is because the use of a very small number of genes (Ref. 10 fix 10) may produce a classification model which depends too heavily on any gene, producing spuriously high prediction strengths.

Previous work in this type of problems[10,30] warn us about their somewhat arbitrary choice in the finally selected number of genes. Thus, stating the problem as a search task and waiting until convergence, a robust and automatic criteria is adopted to carry out this selection, obtaining competitive results with previously cited works.

We carry out the Kruskall–Wallis[18] test over the results of A, B, and C initializations. Table 4 reports the p-values, which indicate the probability that one

Table 3. Average results: estimated accuracy and number of features. Average generation where the best solution of the run appears. Standard-deviation of averages is also reported.

DATA	TYPE	INIT.	ACC.	FEA.	GENERAT.
Colon	Discrete	init-0	64.5 ± 0.2	987 ± 39.1	29.0 ± 6.9
		init-A	91.9 ± 0.1	11.9 ± 4.1	13.0 ± 4.0
		init-B	91.2 ± 0.2	11.8 ± 3.2	11.8 ± 3.2
		init-C	90.9 ± 0.1	6.3 ± 1.6	3.9 ± 1.6
	Continuous	init-0	64.9 ± 10.5	1035 ± 52.4	19.14 ± 8.7
		init-A	95.0 ± 2.3	7.1 ± 2.1	15.2 ± 4.6
		init-B	94.7 ± 2.9	7.2 ± 2.4	12.7 ± 6.9
		init-C	93.4 ± 1.6	6.0 ± 1.9	12.8 ± 5.0
Leukemia	Discrete	init-0	44.0 ± 0.1	3476 ± 57.0	18.2 ± 6.7
		init-A	97.2 ± 0.1	14.6 ± 3.6	14.2 ± 4.2
		init-B	96.9 ± 0.1	14.8 ± 3.6	12.9 ± 4.7
		init-C	98.6 ± 0.0	8.1 ± 1.8	3.3 ± 1.2
	Continuous	init-0	75.9 ± 0.8	3561 ± 35.9	9.3 ± 1.5
		init-A	98.8 ± 1.8	11.0 ± 3.6	18.1 ± 5.7
		init-B	98.8 ± 1.5	11.8 ± 3.2	16.3 ± 3.6
		init-C	96.3 ± 1.1	3.7 ± 1.1	5.9 ± 5.0

Table 4. p-values when comparing A, B and C initializations.

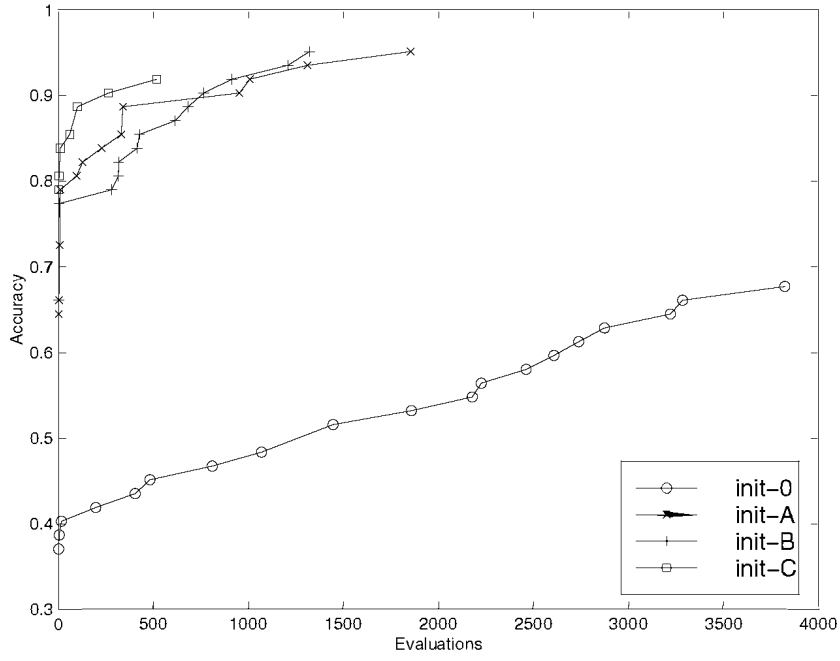
DATA	INIT	ACCURACY	FEATURES	GENERAT.
Colon	Discrete	$p = 0.440$	$p = 0.002$	$p < 0.001$
	Continuous	$p = 0.232$	$p = 0.446$	$p = 0.187$
Leukemia	Discrete	$p = 0.004$	$p = 0.001$	$p < 0.001$
	Continuous	$p = 0.003$	$p < 0.001$	$p < 0.001$

Table 5. p-values when comparing discrete versus continuous models.

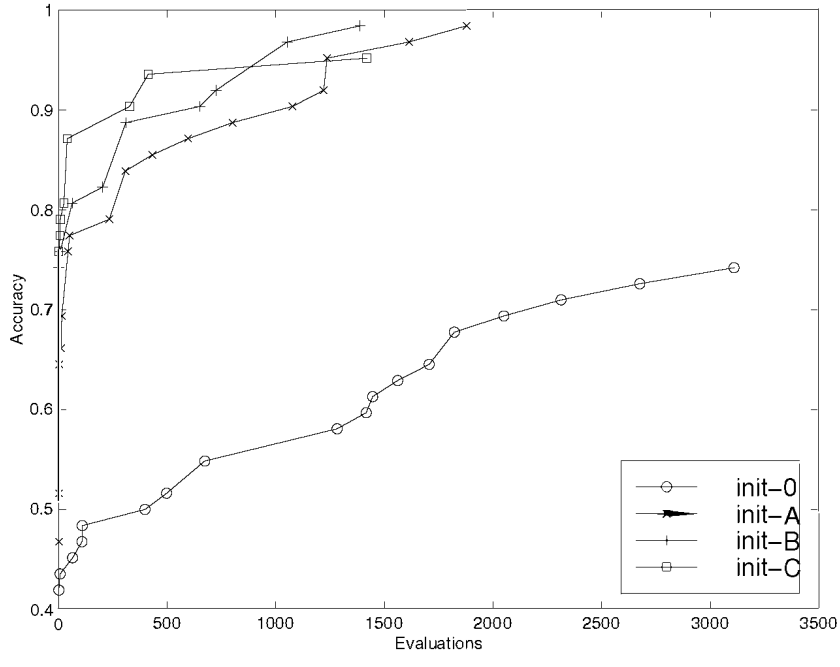
DATA	INIT	DISCRETE versus CONTINUOUS		
		Accuracy	no. var.	no. gen.
Colon	init-0	$p = 0.47$	$p = 0.042$	$p = 0.174$
	init-A	$p = 0.007$	$p = 0.009$	$p = 0.353$
	init-B	$p = 0.043$	$p = 0.004$	$p = 0.631$
	init-C	$p = 0.001$	$p = 0.631$	$p < 0.001$
Leukemia	init-0	$p = 0.017$	$p = 0.017$	$p = 0.067$
	init-A	$p = 0.063$	$p = 0.063$	$p = 0.075$
	init-B	$p = 0.089$	$p = 0.075$	$p = 0.063$
	init-C	$p < 0.001$	$p < 0.001$	$p = 0.218$

initialization is better than the others, where a p-value of 0.05 indicates that the compared initializations are different with a probability of 95%.

In the Colon database, we only obtained significant differences ($p < 0.05$) in the discrete model in relation to the number of features and the number of generations

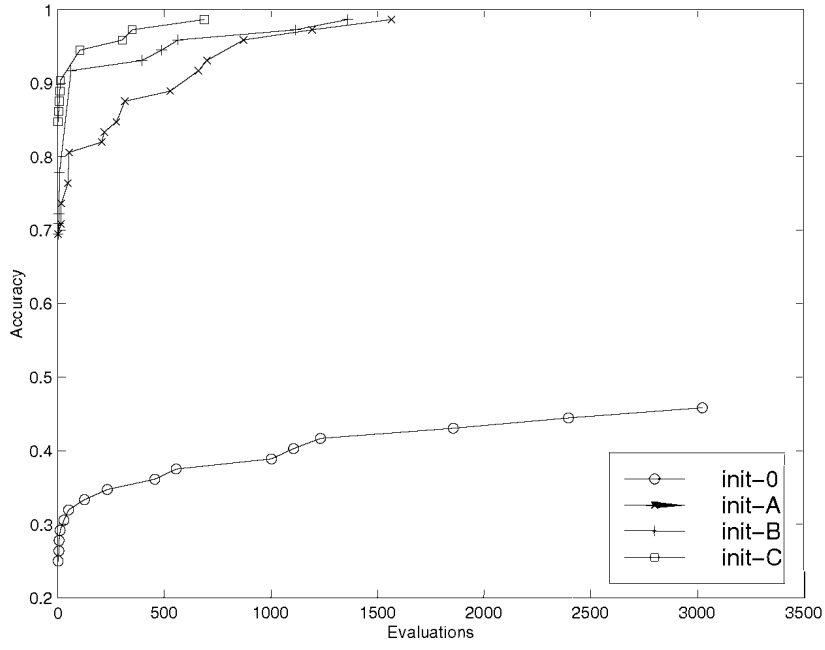


(a)

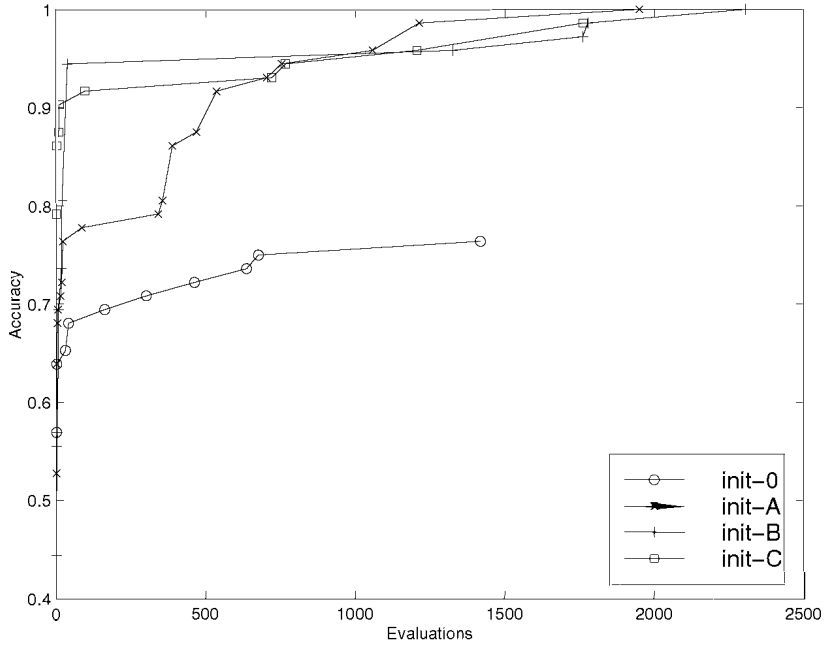


(b)

Fig. 4. The evolution of the best accuracy found in colon dataset: (a) Discrete, (b) Continuous.



(a)



(b)

Fig. 5. The evolution of the best accuracy found in leukemia dataset: (a) Discrete, (b) Continuous.

needed for convergence. In the Leukemia database, the test showed that the differences in all criteria with respect to the three initializations are statistically significant in the two ways of making the classification: when the dataset is discrete and when it is continuous.

Table 5 shows the results obtained when applying the Mann–Whitney[24] test in order to compare the behavior between the discrete and continuous *naïve-Bayes* models.

In the Colon database, we found statistically significant differences in relation to the accuracy of the model for the initializations init-A, init-B, and init-C obtaining the best results in the case of continuous *naïve-Bayes*. With respect to the number of selected features by the EDA, in init-A and init-B initializations, the continuous *naïve-Bayes* model needs significantly more features than its corresponding discrete model. Finally, regarding the number of generations needed until convergence is reached, the differences are statistically significant for initialization init-C where the discrete *naïve-Bayes* needs a larger number of generations.

In the Leukemia database, we find that the differences are only statistically significant in the case of initialization init-C with respect to the accuracy of the model — better result for the discrete *naïve-Bayes* — and the number of features — less features for the continuous *naïve-Bayes*.

Figures 4 and 5 show a typical run of the evolution of the best accuracy found in the search process. In Fig. 4, the evolution of the Colon dataset is depicted. In Fig. 5, the evolution of the Leukemia dataset is drawn. These figures display how the initialization is used to guide the search in the first steps of the process. We can see in all the cases that with init-A, init-B and init-C the best solutions in the first generations are more accurate than with init-0. This demonstrates that more precise solutions can be found during the search processes.

5. Conclusions and Future Work

An application of the EDA approach (by its UMDA algorithm) to select a highly accurate combination of genes in two high-dimensional, well-known gene expression level datasets is carried out. The selection of genes or features is performed within a wrapper approach with respect to the *naïve-Bayes* supervised classification algorithm. Four different approaches, three of them inspired on a sequential selector, are compared in order to initialize the EDA search.

The findings of previous works on the same datasets are confirmed,[10,30] noting that with a low number of genes, the accuracy level of the whole set is significantly improved. In contrast to these works, stating the selection of genes as a search task, an automatic and robust selection of the final number of genes is performed.

This work can be improved in four different and complementary aspects: the classification model, the search engine, the discretization task and the initialization method.

- The classification model
  Obviously, the use of other supervised classifiers that extend the univariate scheme of *naïve-Bayes*, involving relationships among the features, should attain more accurate results.
- The search engine
  Apart from UMDA, other population-based search methods can be proposed to perform the search of the best subset of features. The classical Genetic Algorithms (GA) and other EDA univariate approaches can be applied to this methodology. Other search techniques are able to produce better individuals in a lower number of generations. However, a higher computational cost to produce the offspring may be required.
- The discretization task
  The discretization task should be improved by using a "clever" heuristic approach. The generation of the discrete data set can benefit from the knowledge of the classification model in the continuous domain. In this way, bearing the class variable in mind, a discretization method should provide fitter intervals to discretize.
- The initialization method
  In this work, the proposed initializations are based on a forward sequential greedy wrapper search. Clearly, this search can be extended and improved with other sequential search procedures. Due to the wide dimension of the datasets, a backward sequential wrapper search (that is, starting with the complete set of variables, it removes one feature at each step) is not computationally feasible. Nevertheless, there are algorithms such as floating search[28] that permits a forward sequential wrapper without the inflexibility of a greedy search. Besides, a filter feature subset selection can be used to provide an initialization.

We expect all these improvements to produce more accurate results with a lower number of generations.

Acknowledgments

We thank anonymous referees for their useful suggestions. This work is partially supported by the Department of Education, University and Research of the Basque Government, by the Ministry of Science and Technology, by the Diputación Foral de Guipúzcoa and by the Basque Goverment under grants PI 1999-40, TIC2001-2973-C05-03, OF 760/2003 and ETORTEK-GENMODIS and ETORTEK-BIOLAN projects respectively.

References

1. D. W. Aha and R. L. Bankert, Feature selection for case-based classification of cloud types: an empirical comparison, in *Proc. AAAI-94* (1994), pp. 106–112.
2. H. Almuallim and T. G. Dietterich, Learning with many irrelevant features, in *Proc. Ninth Nat. Conf. Artificial Intelligence* (MIT Press, 1991), pp. 547–552.

3. 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, in *Proc. Nat. Acad. Sci. USA* **96** (1999) 6745–6750.
4. M. Beibel, Selection of informative genes in gene expression based diagnosis: a non-parametric approach, in *Proc. First Int. Symp. Medical Data Analysis* (Springer-Verlang, NY, 2000), pp. 300–306
5. A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer and Z. Yakhini, Tissue classification with gene expression profiles, *J. Comput. Biol.* **7**(3–4) (2000) 559–584.
6. T. H. Bø and I. Jonassen, New features subset selection procedures for classification of expression profiles, *Genome Biol.* **3**(4) (2002).
7. A. Brazma and J. Vilo, Gene expression data analysis, *Fed. Eur. Biochem. Soc. Lett.* **480** (2000) 17–24.
8. P. Domingos and M. Pazzani, On the optimality of the simple Bayesian classifier under zero-one loss, *Mach. Learn.* **29**(2–3) (1997) 103–130.
9. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis* (Wiley, 1973).
10. 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. Caliguri, C. D. Bloomfield and E. S. Lander, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, *Science* **286** (1999) 531–537.
11. D. J. Hand and K. You, Idiot's Bayes — not so stupid after all? *Int. Stat. Rev.* **69** (2001) 385–398.
12. I. Inza, P. Larrañaga, R. Etxeberria and B. Sierra, Feature subset selection by Bayesian network-based optimization, *Artif. Intell.* **123** (2000) 157–184.
13. I. Inza, P. Larrañaga and B. Sierra, Feature subset selection by Bayesian networks: a comparison with genetic and sequential algorithms, *Int. J. Approx. Reason.* **27**(2) (2001) 143–164.
14. I. Inza, B. Sierra, P. Larrañaga and R. Blanco, Gene selection by sequential wrapper approaches in microarray cancer class prediction, *J. Intell. Fuzzy Syst.* **12** (2002) 25–33.
15. A. D. Keller, M. Schummer, L. Hood and W. L. Ruzzo, Bayesian classification of DNA array expression data, Technical Report UW-CSE-2000-08-01, University of Washington (2000).
16. J. Kittler, Feature set search algorithms, in *Pattern Recognition and Signal Processing*, ed. C. H. Chen (Sijthoff and Noordhoff, 1978), pp. 41–60.
17. R. Kohavi and G. John, Wrappers for feature subset selection, *Artif. Intell.* **97**(1–2) (1997) 273–324.
18. W. H. Kruskal and W. A. Wallis, Use of ranks in one-criterion variance analysis, *J. Amer. Statist. Assoc.* **47** (1952) 583–621.
19. P. Langley and S. Sage, Induction of selective Bayesian classifiers, in *Proc. Tenth Conf. Uncertainty in Artificial Intelligence* (Morgan Kaufmann, 1994), pp. 399–406.
20. P. Larrañaga, R. Etxeberria, J. A. Lozano and J. M. Peña, Combinatorial optimization by learning and simulation of Bayesian networks, in *Proc. Sixteenth Conf. Uncertainty in Artificial Intelligence*, eds. C. Boutilier and M. Goldszmidt (Morgan Kaufmann, 2000), pp. 343–352.
21. P. Larrañaga and J. A. Lozano, *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation* (Kluwer Academic Publishers, Boston, 2002).

22. W. Li and Y. Yang, How many genes are needed for a discriminant microarray data analysis? In *Proc. First Conf. Critical Assessment of Microarray Data Analysis, CAMDA* (2000).
23. S. M. Lin and K. F. Johnson, *Methods of Microarray Data Analysis* (Kluwer Academic Publishers, 2000).
24. H. B. Mann and D. R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, *Ann. Math. Statist.* **18** (1947) 50–60.
25. H. Mühlenbein, The equation for response to selection and its use for prediction, *Evolut. Comput.* **5**(3) (1997) 303–346.
26. H. Mühlenbein and G. Paaß, From Recombination of Genes to the Estimation of Distributions I. Binary Parameters, Lecture Notes in Computer Science, Vol. 1411: Parallel Problem Solving from Nature-PPSN IV, 1996, pp. 178–187.
27. O. Pérez, F. J. Marín and O. Trelles, Weighting and selection of variables on gene expression data by the use of genetic algorithms, Technical Report AC-UMA-03ABR02, Universidad de Málaga (2002).
28. P. Pudil, J. Novovicova and J. Kittler, Floating search methods in feature selection, *Patt. Recogn. Lett.* **15**(1) (1994) 1119–1125.
29. M. Stone, Cross-validatory choice and assessment of statistical predictions, *J. Roy. Statist. Soc.* **36** (1974) 111–147.
30. E. P. Xing, M. I. Jordan and R. M. Karp, Feature selection for high-dimensional genomic microarray data, in *Proc. Eighteenth Int. Conf. Machine Learning* (2001), pp. 601–608.