Artificial Intelligence in Medicine (2004) 31, 91–103

ELSEVIER

Artificial Intelligence in Medicine

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

Filter versus wrapper gene selection approaches in DNA microarray domains

Iñaki Inza*, Pedro Larrañaga, Rosa Blanco, Antonio J. Cerrolaza

Department of Computer Science and Artificial Intelligence, University of the Basque Country, P.O. Box 649, E-20080 Donostia-San Sebastián, Basque Country, Spain

Received 24 February 2003; received in revised form 22 July 2003; accepted 16 January 2004

KEYWORDS
DNA microarray;
Bioinformatics;
Supervised classification;
Gene selection;
Wrapper

Summary DNA microarray experiments generating thousands of gene expression measurements, are used to collect information from tissue and cell samples regarding gene expression differences that could be useful for diagnosis disease, distinction of the specific tumor type, etc. One important application of gene expression microarray data is the classification of samples into known categories.

As DNA microarray technology measures the gene expression en masse, this has resulted in data with the number of features (genes) far exceeding the number of samples. As the predictive accuracy of supervised classifiers that try to discriminate between the classes of the problem decays with the existence of irrelevant and redundant features, the necessity of a dimensionality reduction process is essential. We propose the application of a gene selection process, which also enables the biology researcher to focus on promising gene candidates that actively contribute to classification in these large scale microarrays.

Two basic approaches for feature selection appear in machine learning and pattern recognition literature: the filter and wrapper techniques. Filter procedures are used in most of the works in the area of DNA microarrays. In this work, a comparison between a group of different filter metrics and a wrapper sequential search procedure is carried out. The comparison is performed in two well-known DNA microarray datasets by the use of four classic supervised classifiers. The study is carried out over the original-continuous and three-intervals discretized gene expression data. While two well-known filter metrics are proposed for continuous data, four classic filter measures are used over discretized data. The same wrapper approach is used for both continuous and discretized data.

The application of filter and wrapper gene selection procedures leads to considerably better accuracy results in comparison to the non-gene selection approach, coupled with interesting and notable dimensionality reductions. Although the wrapper approach mainly shows a more accurate behavior than filter metrics, this improvement is coupled with considerable computer-load necessities. We note that most of the genes selected by proposed filter and wrapper procedures in discrete and continuous microarray data appear in the lists of relevant-informative genes detected by previous studies over these datasets.

The aim of this work is to make contributions in the field of the gene selection task in DNA microarray datasets. By an extensive comparison with more popular filter techniques, we would like to make contributions in the expansion and study of the wrapper approach in this type of domains.

*Corresponding author. Tel.: +34-943015026; fax: +34-943219306.
E-mail address: inza@si.ehu.es (I. Inza).

0933-3657/$ – see front matter © 2004 Published by Elsevier Ltd.
doi:10.1016/j.artmed.2004.01.007

1. Introduction

The year 2001 marked the emergent bioinformatics discipline with the release of the human genome working draft [1]. This achievement has revolutionized the field of genomics, and within it, producing a spectacular development of the novel DNA microarray technology [2—4]. Instead of investigating one or two specific genes in an organism (as done until nowadays), DNA microarray technology allows for the simultaneous monitoring and measurement of thousands of gene expression activation levels in a single experiment; in this way, researchers can view and study the expression of thousands of genes one at a time.

DNA microarray examples are generated by a hybridization of the mRNA obtained from the studied tissue or blood to the cDNA (in the case of spotted array) and the oligonucleotides of DNA (in the case of Affymetrix chips, on the surface of a chip-array). The arrays are then scanned, producing a fluorescent image: this fluorescent intensity at any particular probe location indicates the relative concentration of the mRNA in the sample (tissue or blood). Microarray data analysis begins with the scanned image of these fluorescent intensities [4].

The DNA microarray technology is providing unprecedent discovery opportunities and reshaping biomedical sciences. Microarray technology opens up the possibility of obtaining answers for old and new biological questions that, before the appearance of this technology, could not be dreamt of. A systematic and computational analysis of microarray datasets is an interesting way to study and understand many aspects of the underlying biological processes. Parallel to these technological advances has been the development of machine learning methods to analyze and understand the data generated by this new kind of experiments [5,6]. The analysis frequently involves class prediction (supervised classification), regression, feature selection (in this case, gene selection), outlier detection, principal component analysis, discovering of gene relationships and cluster analysis (unsupervised classification) [7].

For most biological problems, information about the class (or type) of each cell-line exists: reflecting whether the tissue is diseased or healthy, the distinction of the specific tumor type, etc. By means of this interesting class information, the DNA microarray analysis can be formulated as a classic supervised classification task, classifying samples into categories. Our work is focused on class prediction for DNA microarray problems: starting from a set of cell-lines for which the classification (the class type) is known, we tackle the construction of a predictive model which discriminates among the different categories of the problem. Our objective is to construct accurate and simple classification models. In our study, we use four well-known supervised classification algorithms with completely different approaches to learning and a long tradition in different classification tasks: IB1, naive-Bayes, C4.5 and CN2.

From a pattern recognition point of view, biological samples can be seen as objects, and genes as features to describe each object. In a typical microarray dataset, the number of samples is small (usually less than 100), but the number of genes measured is of magnitude of several thousands, far exceeding the number of samples, with many of the genes being either correlated or irrelevant. Moreover, for diagnostic purposes it is important to find small subsets of genes that are sufficiently informative to distinguish between cells of different types [8]. In this way, the biology researcher can focus attention on a manageable subset of promising gene candidates that actively contribute to classification of cell-lines [9]. All the studies also show that most genes measured in a DNA microarray experiment are not relevant for an accurate distinction among different classes of the problem [3], and scientists in the field are aware that simple classifiers with few genes (less than 15—20) achieve better accuracies [10,11]. To avoid this 'curse of dimensionality' [12], feature selection plays a crucial role in DNA microarray analysis. It is well known that the accuracy of supervised classification methods is not monotonic regarding the inclusion of features [13]: irrelevant or redundant attributes, depending on the specific characteristics of the classifier, may degrade the accuracy of the classification model. In this sense, given the entire set of genes, we aim to find the gene subset with the best predictive accuracy for a certain classifier. This problem is known in the machine learning community as the feature subset selection (FSS) problem (in our case, gene selection) and it has been tackled with success in so many different types of problems [14].

The FSS task has received much attention in the classification literature, where two basic approaches appear to tackle the problem:

- 'Filter' methods evaluate the goodness of the proposed feature subset looking only at the intrinsic characteristics of the data, based on the relation of each single gene with the class label by the calculation of simple statistics computed from the empirical distribution [15]. The most common way is to rank the features in terms of the values of an univariate scoring metric: then, the *d* features with the highest score are chosen

to build the classifier. There is a large variety of different measures such as probabilistic or distance metrics, measures inspired on the information theory, etc. The filter approach is the mostly used FSS method in microarray literature for gene selection [16–18].
- In the 'wrapper' approach [13] a search is conducted in the space of genes, evaluating the goodness of each found gene subset by the estimation of the accuracy percentage of the specific classifier to be used, training the classifier only with the found genes. The wrapper approach, which is very popular in machine learning applications, is not extensively used in DNA microarray tasks, and few works in the field make use of it [19].

In this work, a comparison among a group of different filter metrics and a wrapper sequential search procedure is carried out. The comparison is performed in two well-known DNA microarray datasets involved in the diagnosis of cancer such as *Colon* [8] and *Leukemia* [3]. The study is carried out over the original-continuous and three-intervals discretized gene expression data. While two well-known filter metrics are proposed for continuous data, four classic filter measures are used over discretized data. The same wrapper approach is used for both continuous and discretized data.

The aim of this work is to make contributions in the field of the gene selection task in microarray datasets. We would also like to make contributions in the expansion and study of the wrapper approach in this type of domains, while an extensive and deep comparison with more popular filter techniques is carried out.

The rest of the paper is organized as follows. The supervised classifiers and the gene selection filter and wrapper approaches included in the study are presented in the next section. The studied microarray datasets and experimental results for both continuous and discretized data are shown in Section 3. Section 4 surveys related supervised class prediction works in DNA microarray domains. We finish the work with a brief summary, presenting ways of future research in the field.

2. Learning supervised classifiers by feature subset selection

2.1. Supervised classifiers

In our study, four well-known machine learning supervised classifiers, with completely different approaches to learning, are applied to perform the class prediction in microarray datasets. All the algorithms are selected due to their simplicity and their long standing tradition in classification studies [20]. Apart from the prediction accuracy, the explanation ability of the classifier is also very important. To support the diagnostic process in everyday practice, physicians and biologists need a classifier that is able to explain its decisions, as such transparent decisions are much more acceptable by them. For this reason, other promising techniques, such as neural nets, are not included among our classification models due to their low human-transparency [21,22].

The IB1 [23] is a case-based, nearest-neighbor (K-NN) classifier. To classify a new test sample, all training instances are stored and the nearest training instance regarding the test instance is found: its class is retrieved to predict this as the class of the test instance. To measure the distance between two samples, the Euclidean distance measure is used for continuous values and the overlap metric for discrete ones.

The naive-Bayes (NB) rule [24] uses the Bayes theorem to predict the class for each case, assuming that the predictive genes are independent given the category. To classify a new sample characterized by n genes $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, the NB classifier applies the following rule:

$$c_{NB} = \arg \max_{c_j \in C} p(c_j) \prod_{i=1}^{n} p(x_i|c_j)$$

where c_{NB} denotes the class label predicted by the NB classifier and the possible classes of the problem are grouped in $C = \{c_1, \ldots, c_l\}$. The probability for discrete features is estimated from data using maximum likelihood estimation and applying the Laplace correction. A normal distribution is assumed to estimate the class conditional densities for predictive genes. Despite its simplicity, the NB rule has obtained better results in comparison to more complex algorithms in many domains.

The C4.5 [25] represents a classification model by a decision tree. The tree is constructed in a top-down way, dividing the training set and beginning with the selection of the best variable in the root of the tree. The selection of the best feature is usually based on metrics inspired on the information theory. A descendant of the root node is then created for each possible value of the selected feature, and the training samples are sorted to the appropriate descendant node. The entire process is then recursively repeated using the training cases associated with each descendant node to select the best feature to test at that point in the tree. The process

stops at each node of the tree when all cases in that point of the tree belong to the same category or the best split of the node does not surpass a fixed chi-square significancy threshold. Then, the tree is simplified by a pruning mechanism to avoid over-specialization.

The CN2 [26] algorithm represents a classification model by a set of IF—THEN rules, where the THEN part represents the category predicted for the samples that match the conditions of the IF part. CN2 is also based on the information theory, using a significance metric to improve rule quality and to avoid overspecialization of the results. When a significant rule is found, CN2 removes those samples it covers from the training set and adds the rule to the end of the rule list. To use induced rules to classify test examples, CN2 tries each rule in order until one is found whose conditions are satisfied by the example being classified. If no induced rules are satisfied, the final default rule assigns the most common class in the training set to the test case.

Even though a decision tree can be converted into a set of IF—THEN rules, while CN2 rules are independent to each other, C4.5 rules are dependent on each other. It must be noted that C4.5 and CN2, on their own, can discard some of the presented features to build their classification models. On the other hand, IB1 and NB include all the presented variables in their classification models.

Due to the low number of samples of microarray datasets, the *leave-one-out cross-validation* (LOOCV) procedure [27], a special case of *k*-fold cross-validation, is used in this work to estimate the accuracy of built classifiers. In the LOOCV technique, the supervised algorithm is run N times, where N is the number of examples of the dataset. Each time, $N-1$ examples are used for training and the remaining example is used for testing, where each example is used only once for testing. The LOOCV estimate of accuracy is the overall number of correct classifications during testing, divided by N, the number of examples in the dataset. As the LOOCV estimate is based on the binomial distribution (error versus success), its standard deviation is calculated as

$$\sqrt{\frac{(\text{loocvacc}) \times (100 - \text{loocvacc})}{N-1}}$$

where loocvacc is the LOOCV accuracy estimate percentage and N the number of samples of the dataset. LOOCV estimation is almost unbiased from the real accuracy [27] and it is considered as the most reliable estimator. However, its computational cost can only be assumed, as in our case in microarray datasets, when few samples are presented.

2.2. Selection of genes: the feature subset selection process

The basic problem of supervised classification is concerned with the induction of a model that classifies a given object into one of several known categories. In order to induce the classification model, each object is described by a pattern of n features. Here, the community of researchers has formulated the following question: *Are all of these n descriptive features useful for learning the 'classification rule'?* On trying to respond to this question, we come up with the feature subset selection [14,28] approach which can be reformulated as follows: *given a set of candidate features, select the 'best' subset in a classification problem*. In our case, the 'best' subset will be the one with the best predictive accuracy.

Most of the supervised learning algorithms perform rather poorly when faced with many irrelevant or redundant (depending on the specific characteristics of the classifier) features. In this way, the FSS proposes additional methods to reduce the number of features so as to improve the performance of the supervised classification algorithm.

The FSS problem has been addressed from different research communities such as data mining, pattern recognition, statistics, unsupervised and supervised learning, text learning, etc., and it is an emergent and crucial topic in DNA microarray tasks. Two basic approaches appear in the literature to tackle this problem: 'filter' and 'wrapper' procedures.

2.2.1. FSS: the filter approach

A typical filter procedure assesses the goodness of a single feature looking only at the intrinsic characteristics of the data, measuring the relation of each attribute with the class label of the studied problem. In this way, filter scores try to identify genes that are differentially expressed in the categories of the problem. The first step of the filter procedure is to rank the features in terms of the values of the used univariate scoring metric. In a second step, the d features with the highest scoring metric are chosen to induce the classification model. The literature has plenty of filter metrics of different nature [15]: probabilistic or distance metrics, dependence measures, scores based on the information theory, etc.

In our study, we propose four filter metrics for discretized microarray data. Each metric has a long tradition in FSS and statistics literature [29], and

they calculate the score of each gene X_i in the following way, computing the needed statistics from data:

Shannon-entropy :

$$H(X_i) = -\sum_{j=1}^{v} p(x_j|c_1)\log_2 p(x_j|c_1) + p(x_j|c_2)\log_2 p(x_j|c_2),$$

Euclidean-distance :

$$E(X_i) = \sqrt{\sum_{j=1}^{v}(p(x_j|c_1) - p(x_j|c_2))^2},$$

Kolmogorov-dependence :

$$\mathrm{KO}(X_i) = \sum_{j=1}^{v}(|p(x_j|c_1) - p(x_j|c_2)|)p(x_j),$$

Kullback−Leibler :

$$\mathrm{KL}(X_i) = \sum_{j=1}^{v} p(x_j|c_1)\log\frac{p(x_j|c_1)}{p(x_j)} + \sum_{j=1}^{v} p(x_j|c_2)\log\frac{p(x_j|c_2)}{p(x_j)}$$

where problem classes are $C = \{c_1, c_2\}$ and the gene X_i has v different discrete values.

For microarray continuous data, we propose the following filter metrics to measure the relation between gene X_i and the problem class:

$$P\text{-metric}: \quad P(X_i) = \frac{|\mu_1 - \mu_2|}{\sigma_1 + \sigma_2},$$

$$t\text{-score}: \quad t(X_i) = \frac{|\mu_1 - \mu_2|}{\sqrt{(n_1\sigma_1^2 + n_2\sigma_2^2)/(n_1 + n_2)}}$$

where μ_1 and μ_2 are within-class mean expression levels in class c_1 and class c_2, respectively. σ_1 and σ_2 are standard deviations of expression levels within classes c_1 and c_2, respectively. The t-score is based on a statistical t-test [30] and has a t-distribution with $n_1 + n_2 - 2$ degrees of freedom, where n_1 and n_2 are the number of samples of class c_1 and c_2, respectively.

2.2.2. FSS: the wrapper approach

While the classic filter approach uses a univariate FSS procedure, the wrapper approach carries out a multivariate FSS process. The other crucial difference between both approaches resides in the role that the final classification algorithm plays in the FSS process. Kohavi and John [13] report that when the goal of FSS is the maximization of accuracy, the features selected should depend not only on the features and the target concept to be learned, but also on the classification algorithm. The wrapper concept implies that the FSS algorithm conducts a search for a good subset of features using the induction algorithm itself as a part of the evaluation function, the same algorithm that will be used to induce the final classification model. Once the classification algorithm is fixed, the idea is to train it with the feature subset found by the search algorithm, estimating the accuracy and assigning it as the value of the evaluation function of the feature subset. In this way, representational biases of the induction algorithm which are used to construct the final classifier are included in the FSS process. It is claimed by many authors [13,14] that the wrapper approach obtains better predictive accuracy estimates than the filter approach. In this way, the wrapper approach is becoming the dominant FSS procedure in many scientific areas; however, its computational cost must be taken into account.

In our microarray problems, we propose using sequential forward selection (SFS) [31], a classic and well-known hill-climbing, deterministic search algorithm which starts from an empty subset of genes. It sequentially selects genes, one at a time, until no further improvement is achieved in the evaluation function value. As another advantage with respect to filter approaches, our wrapper approach does not need to fix a specific number of features to train the final classifier, and the number of genes that induce the final classification model is selected by the search component inserted in the own wrapper procedure. As the totality of previous microarray studies note that very few genes are needed to discriminate between different cell classes, we consider that SFS could be an appropriate search engine as it performs the major part of its search near the empty gene subset.

As supervised classifiers when non-gene selection is used, our wrapper approach estimates, by the LOOCV procedure, the goodness of the classifier using only the gene subset found by the search SFS search procedure. Thus, the microarray dataset is projected maintaining the only values of the selected genes and the class variable for all cell samples: the goodness of the proposed gene subset, using the specific classifier, is estimated by the explained LOOCV technique over this projected dataset, which only includes the genes selected by the SFS search procedure and the class of the samples.

3. Experimental results

We test the classification power of proposed gene selection and classification techniques in the following well-known microarray domains:

- The *Colon* dataset of Alon et al. [32] is composed of 62 samples of colon epithelial cell-lines. The samples were collected from colon-cancer patients. The 'tumor' (22 samples) biopsies were collected from tumors, and the 'normal' (40 samples) biopsies were collected from healthy parts of the colons of the same patient. The task is to predict the status of biopsy samples. Of the original ≈6000 genes represented in these arrays, 2000 were selected by the original authors, based on the confidence in their measured expression levels. We work with this dataset of previously selected 2000 genes.
- *Leukemia* dataset of Golub et al. [3]. It contains 72 cell-lines of leukemia patients involving 7129 genes. The class to be predicted is the specific type of acute leukemia of the patient: acute myeloid leukemia-AML (25 patients) or acute lymphoblastic leukemia-ALL (47 patients).

For each filter metric, we construct the classification models with the $d \in \{3, 5, 10, 20\}$ genes of highest scoring value. Thus, for each filter metric, the same subset of genes (selected by the ranking of genes of the filter measure) is used to build our four different classification models (IB1, naive-Bayes, C4.5 and CN2). Experiments are run in an SGI-Origin200 computer using the MLC++ [33] machine learning software library for the presented classifiers.

3.1. Results in continuous data

Tables 1 and 2 show a summary of the results for original-continuous gene expression values in *Colon* and *Leukemia* datasets, respectively. In each table, for each classifier, we first show the LOOCV percentage accuracies for the non-gene selection (no-FSS) and wrapper approaches. Then, these tables show the LOOCV values for each specified gene subset cardinality (3, 5, 10, 20) and filter metric. Tables do not show the standard deviation value of the LOOCV procedure, whose calculation is explained in Section 2.

A deeper analysis of the accuracy results is carried out by using statistical tests. For each domain and specific classifier, a paired *t*-test [30] is performed to determine the statistical significance degree of differences between the accuracy of each approach (without a gene selection and four filter subsets with 3, 5, 10 and 20 genes) and the result of the wrapper procedure. In both tables, the symbol † denotes a statistically significant difference to the wrapper procedure at the $^{*}P < 0.05$ confidence level, and *, denotes significant difference at the $^{*}P < 0.1$ confidence level. This comparison is maintained in the rest of the tables of this work.

With the aid of the wrapper gene selection technique, all classifiers improve their accuracy results in both datasets with respect to the non-gene selection approach. In all cases, except for C4.5 in *Colon*

Table 1 LOOCV accuracy results for each classifier and gene selection technique in *Colon* domain

	IB1		NB		C4.5		CN2	
	P-metric	*t*-score	*P*-metric	*t*-score	*P*-metric	*t*-score	*P*-metric	*t*-score
Three genes	79.03*	75.81†	80.65	82.26	80.65†	74.19†	74.19†	75.81†
Five genes	80.65	72.58†	83.87	87.10	69.35†	82.26†	75.81†	77.42†
Ten genes	82.26	75.81†	83.87	83.87	79.03†	77.42†	77.42†	77.42†
Twenty genes	80.65	69.35†	80.65	83.87	75.81†	77.42†	85.48	80.65*

No-FSS: IB1, 74.19†; NB, 53.23†; C4.5, 87.10; CN2, 77.42†. Wrapper: IB1, 91.94; NB, 87.10; C4.5, 95.16; CN2, 91.94.

Table 2 LOOCV accuracy results for each classifier and gene selection technique in *Leukemia* domain

	IB1		NB		C4.5		CN2	
	P-metric	*t*-score	*P*-metric	*t*-score	*P*-metric	*t*-score	*P*-metric	*t*-score
Three genes	81.94†	81.94†	90.18	90.18	87.50*	84.72†	86.11†	87.50†
Five genes	80.56†	83.33†	90.28	88.89	81.94†	86.11†	83.33†	86.11†
Ten genes	76.39†	84.72†	91.66	91.66	80.56†	72.22†	81.94†	87.50†
Twenty genes	80.56†	81.94†	90.28	90.28	93.06	84.72†	81.94†	83.33†

No-FSS: IB1, 86.11†; NB, 84.72†; C4.5, 84.72†; CN2, 75.00†. Wrapper: IB1, 100.00; NB, 95.83; C4.5, 95.83; CN2, 97.22.

dataset, these accuracy differences between the non-gene selection and the wrapper procedure are statistically significant at the $^{*}P < 0.05$ significance level. Statistically significant differences are also found between the wrapper procedure and most of the gene subsets selected by filter metrics; however, these differences are not noted for NB classifier in both domains.

In the *Colon* dataset, the wrapper procedure selects four, two, three and three genes for IB1, NB, C4.5 and CN2 classifiers, respectively. In the *Leukemia* dataset, the wrapper approach selects three, four, two and three genes for IB1, NB, C4.5 and CN2 classifiers, respectively.

The embedded capacity of C4.5 and CN2 to select features to construct their predictive model should be analyzed. Without a gene selection process, the classification trees built by C4.5 have four and two features for *Colon* and *Leukemia* datasets, respectively. When the CN2 algorithm is faced with the whole set of features, its sets of IF—THEN rules have 10 and 9 genes for *Colon* and *Leukemia* datasets, respectively. In this way, the number of genes of the classification models built by C4.5 and CN2 when non-gene selection is used is similar to the amount of genes selected by the wrapper approach for these classifiers (the dimensionality reduction is larger for CN2).

The own learning process of C4.5 and CN2 prefers genes that are closely correlated (based on information theory metrics) with the class label and does not directly take the predictive accuracy level into account to build the classification model. The filter approach, which applies similar measures to those internally employed by C4.5 and CN2 to select the proper features in the model, is not always able to significantly increase the predictive accuracy. In contrast to filter metrics, the wrapper approach always helps C4.5 and CN2 in the detection of genes that directly build a more accurate model. Thus, by means of the wrapper approach, which focuses its attention on the accuracy level, genes that build a more accurate tree and IF—THEN rules can be found. This capacity of the wrapper approach to improve the accuracy levels of classifiers with an embedded property to select features is largely studied in [34].

However, these accuracy improvements of the wrapper procedure are coupled with demanding computer-load necessities. In the *Colon* dataset, the whole wrapper process needs 10 070, 871, 21 950 and 29 851 CPU seconds for IB1, NB, C4.5 and CN2 classifiers, respectively. In the *Leukemia* dataset, the computer necessities of the whole wrapper procedure are 48 780, 36 006, 115 714 and 203 053 CPU seconds for IB1, NB, C4.5 and CN2 classifiers, respectively. The computer-load necessities of filter procedures can be considered as negligible with respect to wrapper ones.

An analysis of the genes selected by different approaches in the *Colon* dataset reveals interesting questions:

- Among the first 20 genes scored by *P*-metric and *t*-score, the following five genes appear in the top-20-scoring lists of both scores (GenBank number; gene description):
  - R87126; 197371 myosin heavy chain nonmuscle (*Gallus gallus*);
  - M63391; human desmin gene, complete cds.;
  - M76378; human cysteine-rich protein (CRP) gene, exons 5 and 6;
  - Z50753; *H. sapiens* mRNA for GCAP-II/uroguanylin precursor;
  - J02854; myosin regulatory light chain 2, smooth muscle isoform (human), contains element TAR1 repetitive element.
- The only coincidence among the genes selected by the wrapper procedures over four classifiers is that the gene M63391 (human desmin gene, complete cds.) is selected by IB1 and NB inducers.
- Most of the genes selected by proposed filter and wrapper continuous procedures appear in the lists of relevant genes detected by previous studies over this dataset [8,35].

A similar analysis of the genes selected by different approaches in the *Leukemia* dataset reveals interesting questions:

- Among the first 20 genes scored by *P*-metric and *t*-score, the following 15 genes appear in the top-20-scoring lists of both scores (GenBank number; gene description):
  - M23197_at; CD33 CD33 antigen (differentiation antigen);
  - M31211_s_at; MYL1 myosin light chain (alkali);
  - X17042_at; PRG1 proteoglycan 1, secretory granule;
  - X95735_at; zyxin;
  - M22960_at; PPGB protective protein for beta-galactosidase (galactosialidosis);
  - M16038_at; LYN V-yes-1 Yamaguchi sarcoma viral related oncogene homolog;
  - M84526_at; DF D component of complement (adipsin);
  - M62762_at; ATP6C vacuolar H+ ATPase proton channel subunit;
  - M63138_at; CTSD Cathepsin D (lysosomal aspartyl protease);
  - U46499_at; glutathione S-transferase, microsomal;
  - X16546_at; RNS2 ribonuclease 2 (eosinophil-derived neurotoxin, EDN);

- X15414_at; ALDR1 aldehyde reductase 1 (low Km aldose reductase);
- M11147_at; FTL ferritin, light polypeptide;
- Y00787_s_at; interleukin-8 precursor;
- X52056_at; SPI1 spleen focus forming virus (SFFV) proviral integration oncogene spi1.

- The only coincidence among the genes selected by the wrapper procedures over four classifiers is that the gene M23197_at (CD33 CD33 antigen, differentiation antigen) is selected by IB1, NB and CN2 inducers.
- Most of the genes selected by proposed filter and wrapper continuous procedures appear in the lists of relevant genes detected by previous studies over this dataset [5,35].

It must be noted that there are few coincidences in both datasets among the genes selected by the filter and wrapper approaches. It seems that the wrapper approach, by its multivariate selection search procedure, prefers genes which directly cause high accuracy levels in the induced classifiers. On the other hand, the filter approach does not directly take the predictive power of the genes into account, and it univariately selects the genes that are closely related with the class label. Thus, there are no large coincidences between the 'accurate' genes multivariately selected by the wrapper approach and the class-related genes univariately proposed by the filter metrics.

3.2. Results in discrete data

A similar comparison is performed for both datasets when the expression values of each gene are discretized in three states by the procedure proposed in [17], where the author uses the discretization algorithm as a classification model itself. The decision to discretize the gene expression levels in three states is a common approach in the field, assuming that gene expression values can be codified as {under-expressed, baseline, over-expressed} [36].

The applied univariate discretization procedure is supervised by the class label and it is inspired on the information theory. The pair of cutpoints h_1 and h_2 ($h_1 < h_2$) for the gene X_i in a two class problem (class 0, class 1) which maximizes the following measure is chosen:

$$\begin{aligned}\text{cutpoint}(X_i, h_1, h_2) = {} & n_{<0} \log \frac{n_{<0}}{n_<} + n_{<1} \log \frac{n_{<1}}{n_<} \\ & + n_{<>0} \log \frac{n_{<>0}}{n_<>} + n_{<>1} \log \frac{n_{<>1}}{n_<>} \\ & + n_{>0} \log \frac{n_{>0}}{n_>} + n_{>1} \log \frac{n_{>1}}{n_>}\end{aligned}$$

Table 3 LOOCV accuracy results for each classifier and gene selection technique in *Colon* discrete data

	IB1				NB				C4.5				CN2			
	H	*E*	KO	KL	*H*	*E*	KO	KL	*H*	*E*	KO	KL	*H*	*E*	KO	KL
Three genes	85.48*	82.26†	88.71	83.87†	88.71	85.48*	88.81	85.48*	85.48†	82.26†	88.71	83.87†	85.48†	77.42†	83.87†	83.87†
Five genes	85.48*	82.26†	87.10	87.10	87.10	88.71	85.48*	85.48*	83.87†	79.03†	79.03†	79.03†	85.48*	79.03†	83.87†	85.48*
Ten genes	93.55	87.10	83.87†	90.32	80.65†	83.87†	85.48*	85.48*	85.48†	79.03†	82.26†	80.65†	87.10	82.26†	83.87†	83.87†
Twenty genes	88.71	83.87†	87.10	85.48*	69.35†	85.48*	83.87†	82.76†	82.26†	77.42†	77.42†	77.42†	79.03†	87.10	75.81†	87.10

No-FSS: IB1, 95.16; NB, 96.67; C4.5, 62.90†; CN2, 66.13†. Wrapper: IB1, 95.16; NB, 95.16; C4.5, 98.39; CN2, 95.16.

where $n_{<j}$ represents the number of cases in the dataset where $X_i < h_1$ and belong to class j; $n_<$ the number of cases in the dataset where $X_i < h_1$; $n_{<>j}$ the number of cases in the dataset where $h_1 < X_i < h_2$ and belong to class j; $n_{<>}$ the number of cases in the dataset where $h_1 < X_i < h_2$; $n_{>j}$ the number of cases in the dataset where $h_2 < X_i$ and belong to class j; $n_>$ the number of cases in the dataset where $h_2 < x_j$.

To compute the maximum of this formula for each gene X_i, it is sufficient to consider all the

$$\binom{N-1}{2}$$

pairs of the $N-1$ midpoints of X_i's ordered observations.

Tables 3 and 4 show a summary of the results for three-state discretized gene expression values in *Colon* and *Leukemia* datasets, respectively. In each table, for each classifier, we first show the LOOCV percentage accuracies for the non-gene selection (no-FSS) and wrapper approaches. Then, both tables show the LOOCV values for each specified gene subset cardinality (3, 5, 10, 20) and filter metric. The tables do not show the standard deviation value of the LOOCV procedure, whose calculation is explained in Section 2.

In both datasets, the discretization process of the gene expression values helps to considerably improve the predictive accuracy of IB1 and NB classifiers when the whole set of genes is used and for most of the gene subsets selected by filter metrics: for these classifiers, no statistical significant differences are shown between the accuracy of the wrapper procedure and the accuracy of the whole gene set. On the other hand, the advantage of the wrapper procedure with respect to the non-gene selection approach for C4.5 and CN2 classifiers is clear. For all classifiers, statistically significant differences are found between the wrapper approach and most of gene subsets selected by filter metrics.

In the *Colon* dataset, the wrapper procedure selects two, two, three and two genes for IB1, NB, C4.5 and CN2 classifiers, respectively. In the *Leukemia* dataset, the wrapper approach selects three, three, two and two genes for IB1, NB, C4.5 and CN2 classifiers, respectively.

As in the case of continuous data, the dimensionality reduction of the wrapper procedure with respect to the non-gene selection approach for C4.5 and CN2 should be analyzed. When the non-gene selection process is used, the classification trees built by C4.5 have three features for both *Colon* and *Leukemia* datasets, respectively. When

Table 4 LOOCV accuracy results for each classifier and gene selection technique in *Leukemia* discrete data

	IB1				NB				C4.5				CN2			
	H	*E*	KO	KL	*H*	*E*	KO	KL	*H*	*E*	KO	KL	*H*	*E*	KO	KL
Three genes	90.28†	91.67*	90.28†	95.83	90.28*	91.67	93.06	94.44	86.11†	84.72†	88.89	88.89	88.89*	90.28*	91.67	91.67
Five genes	88.89†	95.83	90.28†	93.06	86.11†	93.06	93.06	93.06	86.11†	84.72†	84.72†	86.11	86.11†	88.89*	90.28*	90.28*
Ten genes	87.50†	91.67*	90.28†	88.89†	81.94†	94.44	93.06	91.67	81.94†	84.72†	84.72†	84.72†	84.72†	88.89*	87.50†	87.50†
Twenty genes	91.67*	93.06	90.28†	90.28†	75.00†	88.89*	91.67	87.50	83.33†	83.33†	84.72†	90.28	81.94†	88.89*	83.33†	83.33†

No-FSS: IB1, 100.0; NB, 100.0; C4.5, 86.11†; CN2, 70.83†. Wrapper: IB1, 98.61; NB, 97.22; C4.5, 95.83; CN2, 97.22.

CN2 algorithm is faced with the whole set of features, its sets of IF-THEN rules have 12 and 10 genes for *Colon* and *Leukemia* datasets, respectively. As in the case of continuous microarray data, the number of genes of the classification models built by C4.5 and CN2 when non-gene selection is used, is similar to the amount of genes selected by the wrapper approach for these classifiers for discretized gene expression values; it must be noted that this dimensionality reduction is larger for the CN2 classifier.

As in the case of continuous data, the filter approach, which applies similar measures to those internally employed by C4.5 and CN2 to select the proper features in the model, is not always able to significantly increase the predictive accuracy. In contrast to filter metrics, the wrapper approach always helps C4.5 and CN2 in the detection of genes that directly build a more accurate model. Thus, by means of the wrapper approach, which focuses its attention on the accuracy level, genes that build a more accurate tree and IF—THEN rules can be found.

These accuracy improvements of the wrapper procedure are coupled with demanding computer-load necessities. In the *Colon* dataset, the whole wrapper process needs 4432, 1441, 7849 and 9115 CPU seconds for IB1, NB, C4.5 and CN2 classifiers, respectively. In the *Leukemia* dataset, the computer necessities of the whole wrapper procedure are 47 699, 20 346, 37 483 and 74 741 CPU seconds for IB1, NB, C4.5 and CN2 classifiers, respectively. The computer-load necessities of filter procedures can be considered as negligible with respect to wrapper ones.

An analysis of the genes selected by different approaches in the *Colon* dataset reveals interesting questions:

- Among the first 20 genes scored by four filter metrics, the following 4 genes appear in the top-20-scoring lists of four metrics (GenBank number; gene description):
  - M26383; human monocyte-derived neutrophil-activating protein (MONAP) mRNA, complete cds.
  - H08393; collagen alpha 2(XI) chain (*Homo sapiens*);
  - J05032; human aspartyl-tRNA synthetase alpha-2 subunit mRNA, complete cds.;
  - U09564; human serine kinase mRNA, complete cds.
- Among the first 20 genes scored by four filter scores, other 9 and 8 genes are shared by three and two filter metrics, respectively.
- The gene M26383 (human monocyte-derived neutrophil-activating protein, MONAP, mRNA, complete cds.) is also selected by the wrapper procedures of four classifiers: this is the only coincidence among the genes selected by filter and wrapper approaches.
- Most of the genes selected by proposed filter and wrapper procedures in discrete data appear in the lists of relevant genes detected by previous studies over this dataset [8,35].

A similar analysis of the genes selected by different approaches in the *Leukemia* dataset reveals interesting questions:

- Among the first 20 genes scored by four filter metrics, the following 6 genes appear in the top-20-scoring lists of four metrics (GenBank number; gene description):
  - D88422_at; cystatin A;
  - M23197_at; CD33 CD33 antigen (differentiation antigen);
  - X95735_at; zyxin;
  - M84526_at; DF D component of complement (adipsin);
  - U46499_at; glutathione S-transferase, microsomal;
  - M31211_s_at; MYL1 myosin light chain (alkali).
- Among the first 20 genes scored by four filter scores, another 8 and 6 genes are shared by three and two filter metrics, respectively.
- The gene U46499_at (glutathione S-transferase, microsomal) is also selected by the wrapper procedures of four classifiers: this is the only coincidence among the genes selected by filter and wrapper approaches.
- Most of the genes selected by proposed filter and wrapper procedures in discrete data appear in the lists of relevant genes detected by previous studies over this dataset [5,35].

As in the case of continuous microarray data, few coincidences between the genes selected by filter and wrapper approaches in both datasets appear: there are few coincidences between the 'accurate' genes multivariately selected by the wrapper approach and the class-related genes univariately proposed by filter metrics.

4. Related work in DNA microarray for class prediction

Classic statistical techniques (discriminant analysis, Gaussian and logistic classifiers, etc.) [10,11,37,38], support vector machines [8,18,39], neural networks [40,41] and K-NN [3,11,16,42] are the most broadly used supervised class prediction procedures in microarray domains.

Apart from K-NN, the rest of the classifiers used in our study, which are broadly used in the machine learning field, are not so popular in the DNA microarray area. The decision tree methodology, which is commonly employed in other classification tasks, is not very popular in the DNA microarray domain, but several references can be found in the literature [17,19,43–46]. The NB classifiers only appears in references [19,45,47]. To our knowledge, the CN2 algorithm is only used in our previous work [19] for classification in DNA microarrays. It must be noted that the work of Tobler et al. [45] states the classification problem for choosing the appropriate probes in a microarray study and not for tissue class prediction.

As explained in previous sections, all microarray works are concerned about the necessity of a gene selection procedure to improve the predictive accuracy of their class prediction model. Filter procedures are used in most of the works in the area of DNA microarrays. While the filter metrics that we propose for discretized data are novel in the microarray field, *P*-metric [3,40] and *t*-score [35,48] are broadly used in microarray studies.

Few works use the wrapper approach in microarray domains [19,38,42,47]. Xing et al. [11] propose a hybrid of filter and wrapper approaches.

Before the selection process starts, all the works except [19,47] fix the number of genes for the classifiers to be induced. In this way, we think that the application of the SFS search tool in the space of gene subsets, which does not previously fix a specific number of genes, implies an improvement.

Apart from the FSS approach, other dimensionality reduction procedures appear in supervised microarray literature such as principal component analysis [49] or partial least square [50]. These approaches, considered as feature extraction procedures, combine information from individual features (genes) into components, and describe each sample by these new components instead of individual features.

From the point of view of the employed accuracy estimation technique, most of the works use a hold-out procedure, using a portion of the samples to train a predictive model and using the rest of the instances as a test set to estimate the goodness of the classifier. However, other works use more sophisticated validation techniques, such as *k*-fold cross-validation [43]. LOOCV is also emerging as a reference validation technique in the microarray field [19,35,37,41,42,47]. Although LOOCV can be considered as the most suited estimation procedure for microarray datasets, the employment of the hold-out procedure can be justified in many works as a portion of the samples arise from a specific medical-center or study and other samples arise from a different source. In this way, it is common to use the samples of the first medical-center to train the classifier and the samples of the second source to estimate its accuracy.

5. Summary and future work

A battery of filter and wrapper feature selection algorithms is proposed for the crucial task of accurate gene selection in class prediction problems over DNA microarray datasets. Four classic filter metrics for discretized gene expression data and other two popular filter scores for continuous microarray values are compared with a sequential wrapper technique. The performance of these gene selection techniques is evaluated by four classic supervised classifiers over two well-known DNA microarray datasets involved in the diagnosis of cancer such as *Colon* and *Leukemia*. The benefits, in terms of predictive accuracy improvements, of the proposed gene selection techniques with respect to the non-gene selection approach are shown in most of the cases. Although the wrapper approach mainly shows a more accurate behavior than filter metrics, this improvement is coupled with considerable computer-load necessities. It must be noted the predictive accuracy improvement of IB1 and NB classifiers when gene expression data is discretized.

All gene selection techniques are able to considerably reduce the huge number of genes to small informative and accurate subsets of components: the markers or discriminants of the biological group that actively contribute to the classification of cell-lines [9]. The low number of genes needed by our wrapper approach and previous works over the same datasets show that, with a reduced number of genes (less than 5–10), it is possible to obtain accurate classifiers in these DNA microarray domains.

We note that most of the genes selected by proposed filter and wrapper procedures in discrete and continuous microarray data appear in the lists of relevant-informative genes detected by previous studies over these datasets. Several interesting coincidences between the genes selected by proposed filter and wrapper procedures have also been found.

As future work, we envision to use new filter metrics which, by the use of a statistical hypothesis tests, automatically fix the number of genes to induce the classifier. We also plan to use population-based, randomized search algorithms, such as genetic algorithms or estimation of distribution algorithms for the selection of discriminative genes in

DNA microarray tasks: while our sequential search returns a unique gene subset, the output of a population-based, randomized search algorithm can be interpreted as a group of different gene subsets, from that a 'consensed' final gene subset can be formed [42].

Acknowledgements

This work is supported by the Diputación Foral of Gipuzkoa, the Spanish Carlos III Health Institute, the Spanish Ministry of Science and Technology and the Basque Government under grants OF 146/2002 and OF 760/2003, PI021020, TIC2001-2973-C05-03 and ETORTEK-GENMODIS, respectively.

References

[1] University of California Santa Cruz Genome Bioinformatics, Human Genome Working Draft, 2001. http://genome.ucsc.edu/.
[2] Dopazo J. Microarray data processing and analysis. In: Methods of Microarray Data Analysis II. Proceedings of the Second Conference on Critical Assessment of Microarray Data Analysis, CAMDA'01. Dordrecht: Kluwer Academic Publishers, 2002. p. 43—63.
[3] Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 1999;286:531—37.
[4] McConnell P, Johnson K, Lockhart DJ. An introduction to DNA microarrays. In: Methods of Microarray Data Analysis II. Proceedings of the Second Conference on Critical Assessment of Microarray Data Analysis, CAMDA'01. Dordrecht: Kluwer Academic Publishers, 2002. p. 9—21.
[5] Lin SM, Johnson KF. Methods of microarray data analysis. In: Proceedings of the First Conference on Critical Assessment of Microarray Data Analysis, CAMDA'00. Dordrecht: Kluwer Academic Publishers, 2002.
[6] Lin SM, Johnson KF. Methods of Microarray Data Analysis II. Proceedings of the Second Conference on Critical Assessment of Microarray Data Analysis, CAMDA'01. Dordrecht: Kluwer Academic Publishers, 2002.
[7] Brazma A, Vilo J. Gene expression data analysis. FEBS (Fed Eur Biochem Soc) Lett 2000;480:17—24.
[8] Ben-Dor A, Bruhn L, Friedman N, Nachman I, Schummer M, Yakhini Z. Tissue classification with gene expression profiles. *J Comp Biol* 2000;7(3—4):559—84.
[9] Stefanini FM, Camussi A. The reduction of large molecular profiles to informative components using a genetic algorithm. *Bioinformatics* 2000;16:923—31.
[10] Li W, Yang Y. How many genes are needed for a discriminant microarray data analysis? In: Methods of Microarray Data Analysis. Proceedings of the First Conference on Critical Assessment of Microarray Data Analysis, CAMDA'00. Dordrecht: Kluwer Academic Publishers, 2002. p. 137—50.
[11] Xing EP, Jordan MI, Karp RM. Feature selection for high-dimensional genomic microarray data. In: Proceedings of the 18th International Conference on Machine Learning, ICML'01, 2001. p. 601—8.
[12] Jain A, Zongker D. Feature selection: evaluation, application, and small sample performance. *IEEE Trans Pattern Anal Mach Intell* 1997;19(2):153—8.
[13] Kohavi R, John G. Wrappers for feature subset selection. *Artif Intell* 1997;97(1—2):273—324.
[14] Liu H, Motoda H. Feature selection for knowledge discovery and data mining. Dordrecht: Kluwer Academic Publishers, 1998.
[15] Ben-Bassat M. Pattern recognition and reduction of dimensionality. In: Krishnaiah PR, Kanal LN, editors. Handbook of statistics II. Amsterdam: North-Holland, 1982. p. 773—91.
[16] Aris V, Recce M. A method to improve detection of disease using selectively expressed genes in microarray data. In: Methods of Microarray Data Analysis. Proceedings of the First Conference on Critical Assessment of Microarray Data Analysis, CAMDA'00. Dordrecht: Kluwer Academic Publishers, 2002. p. 69—80.
[17] Beibel M. Selection of informative genes in gene expression based diagnosis: a nonparametric approach. In: Lecture Notes in Computer Sciences. Proceedings of the First International Symposium in Medical Data Analysis, ISMDA'00, vol. 1933. Springer-Verlag, 2000. p. 300—7.
[18] Ding CHQ. Analysis of gene expression profiles: class discovery and leaf ordering. In: Proceedings of the Sixth International Conference on Research in Computational Molecular Biology, 2002. p. 127—36.
[19] Inza I, Sierra B, Blanco R, Larrañaga P. Gene selection by sequential search wrapper approaches in microarray cancer class prediction. *J Intell Fuzzy Syst* 2002;12:25—33.
[20] Mitchell TM. Machine learning. New York: McGraw-Hill, 1997.
[21] Inza I, Merino M, Larrañaga P, Quiroga J, Sierra B, Girala M. Feature subset selection by genetic algorithms and estimation of distribution algorithms. A case study in the survival of cirrhotic patients treated with TIPS. *Artif Intell Med* 2001;23(2):187—205.
[22] Michie D. Personal models of rationality. *J Stat Plan Infer* 1990;25:381—99.
[23] Aha DW, Kibler D, Albert MK. Instance-based learning algorithms. *Mach Learn* 1991;6:37—66.
[24] Cestnik B. Estimating probabilities: a crucial task in machine learning. In: Proceedings of the European Conference on Artificial Intelligence, 1990. p. 147—49.
[25] Quinlan JR. C4.5: Programs for machine learning. Los Altos, CA: Morgan Kaufmann, 1993.
[26] Clark P, Nibblet T. The CN2 induction algorithm. *Mach Learn* 1989;3(4):261—83.
[27] Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of the International Joint Conference on Artificial Intelligence, 1995. p. 1137—45.
[28] Inza I, Larrañaga P, Etxeberria R, Sierra B. Feature subset selection by Bayesian network-based optimization. *Artif Intell* 2000;123(1—2):157—84.
[29] Doak J. An evaluation of feature selection methods and their application to computer security. University of California at Davis Technical Report CSE-92-18, 1992.
[30] Dietterich TG. Approximate statistical tests for comparing supervised learning algorithms. *Neural Comput* 1998;10(7):1895—924.
[31] Kittler J. Feature set search algorithms. In: Chen CH, editor. Pattern recognition and signal processing. Alphen a/d Rijn: Sijthoff & Noordhoff, 1978. p. 41—60.
[32] Alon U, Barkai N, Gish K, Notterman D, Ybarra S, Mack D, et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues

probed by oligonucleotide arrays. In: Proceedings of the National Academy of Sciences, USA, vol. 96, 1999. p. 6745—50.
[33] Kohavi R, Sommerfield D, Dougherty J. Data mining using MLC++, a machine learning library in C++. *Int J Artif Intell Tools* 1997;6:537—66.
[34] Kohavi R. Wrappers for performance enhancement and oblivious decision graphs. PhD Thesis. Stanford University, 1995.
[35] Bø TH, Jonassen I. New feature subset selection procedures for classification of expression profiles. Genome Biol. 2002; 3(4).
[36] Friedman N, Linial M, Nachman I, Pe'er D. Using Bayesian networks to analyze expression data. *J Comp Biol* 2000;7: 601—20.
[37] Grate LR, Bhattacharyya C, Jordan MI, Mian IS. Simultaneous relevant feature identification and classification in high-dimensional spaces. In: Lecture Notes in Computer Sciences. Proceedings of the Workshop on Algorithms in Bioinformatics, WABI'02, vol. 2452. Springer-Verlag, 2002. p. 1—9.
[38] Xiong M, Fang Z, Zhao J. Biomarker identification by feature wrappers. *Genome Res* 2001;11:1878—87.
[39] Furey TS, Cristianini N, Duffy N, Bednarski DW, Schummer M, Haussler D. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics* 2000;16(10):906—14.
[40] Hwang K-B, Cho D-Y, Park S-W, Kim S-D, Zhang B-Y. Applying machine learning techniques to analysis of gene expression data: cancer diagnosis. In: Methods of Microarray Data Analysis. Proceedings of the First Conference on Critical Assessment of Microarray Data Analysis, CAMDA'00. Dordrecht: Kluwer Academic Publishers, 2002. p. 167—82.
[41] Mateos A, Herrero J, Tamames J, Dopazo J. Supervised neural networks for clustering conditions in DNA array data after reducing noise by clustering gene expression profiles. In: Methods of Microarray Data Analysis II. Proceedings of the Second Conference on Critical Assessment of Microarray Data Analysis, CAMDA'01. Dordrecht: Kluwer Academic Publishers, 2002. p. 91—104.
[42] Li L, Pedersen LG, Darden TA, Weinberg CR. Computational analysis of leukemia microarray expression data using the GA/KNN method. In: Methods of Microarray Data Analysis. Proceedings of the First Conference on Critical Assessment of Microarray Data Analysis, CAMDA'00. Dordrecht: Kluwer Academic Publishers, 2002. p. 81—96.
[43] Dubitzky W, Granzow M, Berrar D. Comparing symbolic and subsymbolic machine learning approaches to classification of cancer and gene identification. In: Methods of Microarray Data Analysis. Proceedings of the First Conference on Critical Assessment of Microarray Data Analysis, CAMDA'00. Dordrecht: Kluwer Academic Publishers, 2002. p. 151—66.
[44] Dudoit S, Fridlyand J, Speed TP. Comparison of discriminant methods for the classification of tumors using gene expression data. *J Am Stat Assoc* 2002;97(457):77—87.
[45] Tobler JB, Molla MN, Shavlik JW, Nuwaysir E, Green R, Evaluating machine learning approaches for aiding probe selection for gene-expression arrays. Bioinformatics 2002;18:S164—71.
[46] Zhang H, Yu CY, Singer B, Xiong M. Recursive partitioning for tumor classification with gene expression microarray data. In: Proceedings of the National Academy of Sciences, USA, vol. 98, 1999. p. 6730—35.
[47] Blanco R, Larrañaga P, Inza I, Sierra B. Selection of highly accurate genes for cancer classification by estimation of distribution algorithms. In: Proceedings of the Workshop of Bayesian Models in Medicine, AIME'01, 2001. p. 29—34.
[48] Li Y-J, Zhang L, Speer MC, Martin ER. Evaluation of current methods of testing differential gene expression and beyond. In: Methods of Microarray Data Analysis II. Proceedings of the Second Conference on Critical Assessment of Microarray Data Analysis, CAMDA'01. Dordrecht: Kluwer Academic Publishers, 2002. p. 185—94.
[49] Khan J, Wei JS, Ringér M, Saal M, Ladanyi LH, Westermann F, et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat Med 2001;7(6):673—79.
[50] Nguyen DV, Rocke DM. Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics* 2002;18(1):39—50.