Incremental wrapper–based gene selection from microarray data for cancer classification

Roberto Ruiz *,[a], José C. Riquelme [a], Jesús S. Aguilar-Ruiz [b]

[a] *Department of Computer Science, University of Seville*
Avda. Reina Mercedes s/n. 41012 Seville, Spain

[b] *Polytechnic, Pablo de Olavide University*
Ctra. Utrera, km 1, 41013 Seville, Spain

Abstract

Gene expression microarray is a rapidly maturing technology that provides the opportunity to assay the expression levels of thousands or tens of thousands of genes in a single experiment. We present a new heuristic to select relevant gene subsets in order to further use them for the classification task. Our method is based on the statistical significance of adding a gene from a ranked–list to the final subset. The efficiency and effectiveness of our technique is demonstrated through extensive comparisons with other representative heuristics. Our approach shows an excellent performance, not only at identifying relevant genes, but also with respect to the computational cost.

Key words:
Microarray, gene selection, classification, feature selection

1 Introduction

All cells have a nucleus, and inside nucleus there is DNA, which encodes the program for making future organisms. DNA has coding and non-coding segments, and coding segments, called genes, specify the structure of proteins, which are large molecules, like hemoglobin, that do the essential work in every organism. Practically all cells in the same organism have the same genes, but these genes can be expressed differently at different times and under different conditions. Genes make proteins in two steps. First, DNA is

* Corresponding author. Tel.: +34-954553867; fax: +34-954557139
Email address: rruiz@lsi.us.es (Roberto Ruiz).

transcribed into messenger RNA or mRNA, which in turn is translated into proteins (see Piatetsky-Shapiro and Tamayo (2003) for a review). In recent years there has been an explosion in the rate of acquisition of biomedical data. Advances in molecular genetics technologies, such as DNA microarrays allow us for the first time to obtain a "global" view of the cell.

Analysis of microarray data presents unprecedented opportunities and challenges for data mining in areas such as gene clustering (Alon et al., 1999), sample clustering and class discovery (Alon et al., 1999; Golub et al., 1999), sample classification (Alizadeh et al., 2000; Golub et al., 1999) and gene selection (Ding and Peng, 2003; Inza et al., 2004; Yu and Liu, 2004b; Guyon et al., 2002; Xing et al., 2001). In this work, we address the gene selection issue under a classification framework. The aim is to build a classifier that accurately predicts the classes (diseases or phenotypes) of new unlabelled samples. Three well–known machine learning classifiers (naïve Bayes, instance–based and decision trees), with completely different approaches to learning, are applied to perform the class prediction. A typical data set may contain thousands of genes but only a small number of samples (often less than two hundred). Theoretically, having more genes should give us more discriminating power. However, this can cause several problems: increase computational complexity and cost; too many redundant or irrelevant genes; and estimation degradation in the classification error. In addition to reducing noise and improving the accuracy of classification, the selected subsets of genes may have important biological interpretation and may be used for drug target discovery or identifying future possible research directions.

The problem of feature selection received a thorough treatment in pattern recognition and machine learning. Most of the feature selection algorithms approach the task as a search problem, where each state in the search specifies a distinct subset of the possible attributes (Blum and Langley, 1997). The search procedure is combined with a criterion in order to evaluate the merit of each candidate subset of attributes. There are a lot of possible combinations between each procedure search and each attribute measure (Liu and Yu, 2005). However, search methods can be prohibitively expensive in high–dimensional data sets, especially when a data mining algorithm is applied as evaluation function.

There are various ways in which feature selection algorithms can be grouped according to the attribute evaluation measure: depending on the type (filter or wrapper techniques) or on the way that features are evaluated (individual or subset evaluation). The filter model relies on general characteristics of the data to evaluate and select gene subsets without involving any mining algorithm. The wrapper model requires one predetermined mining algorithm and uses its performance as the evaluation criterion. It searches for features better suited to the mining algorithm, aiming to improve mining performance, but it also is

more computationally expensive (Langley, 1994; Kohavi and John, 1997) than filter models. Feature ranking (FR), also called feature weighting (Blum and Langley, 1997; Guyon and Elisseeff, 2003), assesses individual features and assigns them weights according to their degrees of relevance, while the feature subset selection (FSS) evaluates the goodness of each found feature subset. (Unusually, some search strategies in combination with subset evaluation can provide a ranked list).

In order to compare the effectiveness of gene selection, gene sets chosen by each technique are tested with three well–known learning algorithms: a probabilistic learner (naïve Bayes), an instance–base learner (IB1) and a decision tree learner (C4.5). These three algorithms have been chosen because they represent three quite different approaches to learning, and their long standing tradition in classification studies. The comparison is performed in four DNA microarray data sets involved in diagnosis of cancer (Colon, Leukemia, Lymphoma and Global Cancer Map).

The paper is organized as follows. In the next two Sections, we will review previous work, and notions of feature relevance and redundancy, respectively. In Section 4, we will present our proposed measures of gene relevance and redundancy using a wrapper approach, and our algorithm is described. Experimental results are shown in Section 5, and the most interesting conclusions are summarized in Section 6.

2 Related work

Traditional gene selection methods often select the top–ranked genes according to their individual discriminative power (Golub et al., 1999). This approach is efficient for high–dimensional data due to its linear time complexity in terms of dimensionality. They can only capture the relevance of genes to the target concept, but cannot discover redundancy and basic interactions among genes. In the FSS algorithms category, candidate gene subsets are generated based on a certain search strategy. Different algorithms address these issues distinctively. In (Liu and Yu, 2005), a great number of selection methods are categorized. We found different search strategies, namely *exhaustive*, *heuristic* and *random* search, combined with several types of measures to form different algorithms. The time complexity is exponential in terms of data dimensionality for exhaustive search and quadratic for heuristic search. The complexity can be linear to the number of iterations in a random search, but experiments show that in order to find the best feature subset, the number of iterations required is usually at least quadratic to the number of features (Dash et al., 2000). The most popular search methods in pattern recognition and machine learning can not be applied to these gene expression data sets due to the large number of

genes (sometimes tens of thousands). One of the few used search techniques in these domains is sequential forward (SF, also called hill–climbing or greedy search). Different subset evaluation measures in combination with SF search engine can be found. We are specially interested in wrapper approach (Inza et al., 2004; Xiong et al., 2001).

A key issue of wrapper methods is how to search into the space of subsets of genes. Although several heuristic search strategies such as greedy sequential search, best–first search, and genetic algorithm exist, most of them are still computationally expensive $O(N^2)$ (being N the number of genes of the original data set), which prevents them from scaling well to data sets containing thousands of genes. A rough estimate of the time required by most of these techniques is in the order of thousands of hours, assuming that the method does not get caught in a local minima first and stops prematurely. For example, if we have chosen fifty genes from twenty thousand (0.0025% of the whole set) through a greedy search, the subset evaluator would be run approximately one million times (N times to find the best single gene, then tries each of the remaining genes in conjunction with the best to find the most suited pair of genes $N-1$ times, and so on, more or less 20000×50). Assuming four seconds on average by each evaluation, the results would take more than one thousand hours.

The limitations of both approaches, FR and FSS, clearly suggest that we should pursue a hybrid model. Recently, a new framework of feature (gene) selection has been used, where several above–mentioned approaches are combined. Yu and Liu (2004a) proposed a fast correlation–based filter algorithm (FCBF) which used correlation measure to obtain relevant genes and to remove redundancy. There are other methods based on relevance and redundancy concepts. Recursive Feature Elimination (RFE) is a proposed feature selection algorithm described by Guyon et al. (2002). The method, given that one wishes to find only r dimensions in the final subset, works by trying to choose the r features which lead to the largest margin of class separation, using an SVM classifier. This combinatorial problem is solved in a greedy fashion at each iteration of training by removing the input dimension that decreases the margin the least until only r input dimensions remain (this is known as backward selection). Ding and Peng (2003) have used mutual information for gene selection that has maximum relevance with minimal redundancy by solving a simple two–objective optimization. Xing et al. (2001) proposed a hybrid of filter and wrapper approaches to feature selection.

In (Hall and Holmes, 2003), it is proposed a rank search method to compare feature selection algorithms. Rank search techniques rank all genes, and subsets of increasing size are evaluated from the ranked list (i.e., the first attribute, the two first ones, etc...). The best attribute set is reported. The authors apply the wrapper approach to data sets up to 300 attributes and

state that for the ADS data set (1500 attributes) the estimated time to only generate the ranking in a machine with 1.4GHz processor would be about 140 days and to evaluate the ranked list of attributes would take about 40 days. In contrast, our method can be tested on data sets with 20.000 genes in a similar machine in a few hours.

This paper presents a new gene selection method, based on the hybrid model, and attempts to take advantage of all of the different approaches by exploiting their best performances in two steps: first, a filter or wrapper approach provides a ranked list of genes, and second, ordered genes are added using a wrapper subset evaluation ensuring good performance (the search algorithm is valid for any gene ranked–list). This approach provides the possibility of efficiently applying wrapper model in high–dimensional domains, obtaining better results than the filter model. The final subset is obviously not the optimum, but it is unfeasible to search for every possible subset of genes through the search space. The main goal of our research is to obtain a few features with high predictive power.

3 Preliminary Concepts

3.1 Relevance

The purpose of a feature subset algorithm is to identify relevant features according to a definition of relevance. However, the notion of relevance in machine learning has not yet been rigorously defined in common agreement (Bell and Wang, 2000). Kohavi and John (1997) include three disjointed categories of feature relevance: strong relevance, weak relevance and irrelevance. These groups are important to decide what features should be conserved and which ones can be eliminated. The strongly relevant features are, in theory, important to maintain a structure in the domain, and they should be conserved by any feature selection algorithm in order to avoid the addition of ambiguity to the sample. Weakly relevant features could be important or not, depending on the other features already selected and on the evaluation measure that has been chosen (accuracy, simplicity, consistency, etc.). Irrelevant attributes are not necessary at all. Bell and Wang (2000) make use of Information Theory concepts to define the entropic or variable relevance of a feature with respect to the class. Blum and Langley (1997) collect several relevance definitions. The above notions of relevance are independent of the specific learning algorithm being used. There is no guarantee that just because a feature is relevant, it will necessarily be useful to an algorithm (or vice versa). The definition of incremental relevance by Caruana and Freitag (1994) makes it explicit, being considered especially suited to obtain a predictive feature subset.

Definition 1 (Incremental usefulness) *Given a sample of data D, a learning algorithm L, and a feature subset F, the feature x_i is incrementally useful to L with respect to F if the accuracy of the hypothesis that L produces using the group of features $\{x_i\} \cup F$ is better than the accuracy achieved using just the subset of features F.*

We consider this definition to be especially suited to obtain a predictive feature subset. In the next section, concepts can be applied to avoid a subset which contains attributes with the same information.

3.2 Redundancy

Notions of feature redundancy are normally in terms of feature correlation. It is widely accepted that two features are redundant to each other if their values are completely correlated. There are two widely used types of measures for the correlation between two variables: linear and non–linear. In the first case, the Pearson correlation coefficient is used, and in the second one, many measures are based on the concept of entropy, or measure of the uncertainty of a random variable. Symmetrical uncertainty is frequently used, defined as

$$SU(X,Y) = 2\left[\frac{IG(X|Y)}{H(X)+H(Y)}\right]$$

where $H(X) = -\sum_i P(x_i) log_2(P(x_i))$ is the entropy of a variable X and $IG(X|Y) = H(X) - H(X|Y)$ is the information gain from X provided by Y.

The above–mentioned definitions are between pairs of variables. However, it may not be as straightforward in determining feature redundancy when one is correlated with a set of features. Koller and Sahami (1996) apply a technique based on cross–entropy, named Markov blanket filtering, to eliminate redundant features. This idea is formalized in the following definition.

Definition 2 (Markov blanket) *Given a feature $x_i \in F$ (a set of attributes) and the class C, the subset $M \subseteq F (x_i \notin M)$ is a Markov blanket of x_i if, given M, x_i is conditionally independent of $F - M - \{x_i\}$ and C.*

Two attributes (or set of attributes) X, Y are said to be conditionally independent given a third attribute Z (or set) if, given Z makes X and Y independent, i.e., the distribution of X, known Y and Z, is equal to the distribution X known Z, therefore, Y does not have influence on X ($P(X|Y,Z) = P(X|Z)$).

Theoretically, it can be shown that once we find a Markov blanket M of feature x_i in a feature set F, we can safely remove x_i from F without increasing the divergence from the original distribution. Furthermore, in a sequential

filtering process, in which unnecessary features are removed one by one, a feature tagged as unnecessary based on the existence of a Markov blanket M remains unnecessary in later stages when more features have been removed. The Markov blanket condition requires that M assumes not only the information that x_i has about C, but also about all the other features. In (Koller and Sahami, 1996) it is stated that the cardinality of set M must be small and fixed.

Xing et al. (2001) and Yu and Liu (2004a) are among the most cited works at present following the above–mentioned framework (FR+FSS). Both of them are based on this concept of Markov blanket. In the first one, the number of attributes of M are not provided, but it is a fixed number among the highly correlated features. In the second one, a fast correlation–based filter is implemented (FCBF), where M is formed by only one attribute, and gradually eliminates redundant attributes with respect to M from the first to the final attribute of an ordered list. Other methods based on relevance and redundancy concepts can be found in (Guyon et al., 2002; Ding and Peng, 2003).

4 Incremental Performance of Wrapper–Search over Ranking

In this section, we first introduce our ideas of relevance and redundancy taking into account the aim of applying a wrapper model to gene expression data sets, and then our approach is described.

As previously indicated, the wrapper model makes use of the algorithm that will build the final classifier to select a gene subset. Thus, given a classifier L, and given a set of genes G, a wrapper method searches in the space of G, using cross–validation to compare the performance of the trained classifier L on each tested subset. While the wrapper model is more computationally expensive than the filter model, it also tends to find gene sets better suited to the inductive biases of the learning algorithm and therefore provides superior performance.

In this work, we propose a fast search over a minimal part of the gene space. Beginning with the first gene from the list ordered by some evaluation criterion, genes are added one by one to the subset of selected genes only if such inclusion improves the classifier accuracy. Then, the learning algorithm of the wrapper approach is always run N (number of genes) times, usually with a few genes. A gene ranking algorithm makes use of a scoring function computed from the values of each gene and the class label. By convention, we assume that a high score is indicative of a valuable gene and that we sort genes in decreasing order of this score. We consider ranking criteria defined for individual genes, independently of the context of others.

When a ranking of genes is provided from a high dimensional data set, a large number of genes with similar scores is generated, and a common criticism is that it leads to the selection of redundant subsets. However, according to Guyon and Elisseeff (2003), noise reduction and consequently better class separation may be obtained by adding variables that are presumably redundant. Moreover, a very high attribute correlation (in absolute value) does not mean absence of attribute complementarity. Therefore, our idea of redundancy is not based on correlation measures, but on the learning algorithm target (wrapper approach), in the sense that a gene is chosen if additional information is gained by adding it to the selected subset of genes.

4.1 Incremental Ranked Usefulness

In gene subset selection, it is a fact that two types of genes are generally perceived as being unnecessary: genes that are irrelevant to the target concept, and genes that are redundant given other genes. Our approach is based on the concept of Markov blanket, which is described in (Koller and Sahami, 1996). This idea was formalized using the notion of conditionally independent attribute, which can be defined by several approaches (Xing et al., 2001; Yu and Liu, 2004a). We set this concept by a wrapper model, defining incremental ranked usefulness in order to devise an approach to explicitly identify relevant genes and do not take into account redundant genes.

Let D be a sample of labelled data; G be a subset of genes of D; and L be a learning algorithm; *correct rate* (or accuracy) $\Gamma(D/G, L)$ is named to the ratio between the number of instances correctly classified by L and the total number of instances of D considering only the subset G.

Let $R = \{g_i\}$, $i = 1 \dots N$ be a ranking of all the genes in D sorted in descending order, and G be named the subset of the i first genes of R.

Definition 3 (Incremental Ranked Usefulness) *The gene g_{i+1} in R is incrementally useful to L if it is not conditionally independent of the class C given G, therefore the correct rate of the hypothesis that L produces using the group of genes $\{g_{i+1}\} \cup G$ is significantly better (denoted by $\succcurlyeq$) than the correct rate achieved using just the subset of genes G.*

Therefore, if $\Gamma(D/G \cup \{g_{i+1}\}, L) \not\succcurlyeq \Gamma(D/G, L)$, then g_{i+1} is conditionally independent of class C given the subset G, then we should be able to omit g_{i+1} without compromising the accuracy of class prediction.

A fundamental question in the previous definition is how the *significant* improvement is analyzed. A five–fold cross–validation is used to estimate if the accuracy of the learning scheme for a set of genes is significantly better ($\succcurlyeq$)

```
Input: D training U-measure, L-classifier
Output: BestSubset
1  list R = {}
2  for each gene g_i ∈ D
3      Score = compute(g_i, U, D)
4      append g_i to R according to Score
5  BestClassif = 0
6  BestSubset = ∅
7  for i = 1 to N
8      TempSubset = BestSubset ∪ {g_i} (g_i ∈ R)
9      TempClassif = WrapperClassif(TempSubset, L)
10     if (TempClassif ⩾ BestClassif)
11         BestSubset = TempSubset
12         BestClassif = TempClassif
```

Fig. 1. BIRS Algorithm.

than the accuracy obtained for another set. We conducted a Student's paired two–tailed t–test in order to evaluate the statistical significance (at 0.1 level) of the difference between the previous best subset and the candidate subset. This last definition allows us to select genes from the ranking, but only those that increase the classification rate significantly. Although the size of the sample is small (five folds), our search method uses a t–test. We want to obtain a heuristic, not to do an accurate population study. However, on the one hand it must be noticed that it is a heuristic based on an objective criterion, to determine the statistical significance degree of difference between the accuracies of each subset. On the other hand, the confidence level has been relaxed from 0.05 to 0.1 due to the small size of the sample. Statistically significant differences at the $p < 0.05$ significance level would not allow us to add more features, because it would be difficult for the test to obtain significant differences between the accuracy of each subset. Obviously, if the confidence level is increased, more genes can be selected, and vice versa.

4.2 Algorithm

There are two phases in the algorithm, named BIRS (best incremental ranked subset), shown in Figure 1: firstly, the genes are ranked according to some evaluation measure (line 1–4). In the second phase, we deal with the list of

Table 1
Example of gene selection process by BIRS.

Rank	g_5	g_7	g_4	g_3	g_1	g_8	g_6	g_2	g_9

	Subset Eval.	Acc	P-Val	Acc–Best Sub	
1	g_5	80		**80**	$\mathbf{g_5}$
2	$\mathbf{g_5}$,g_7	82			
3	$\mathbf{g_5}$,g_4	81			
4	$\mathbf{g_5}$,g_3	83			
5	$\mathbf{g_5}$,g_1	84	$< \mathbf{0.1}$	**84**	$\mathbf{g_5}, \mathbf{g_1}$
6	$\mathbf{g_5}, \mathbf{g_1}$,$g_8$	84			
7	$\mathbf{g_5}, \mathbf{g_1}$,$g_6$	86			
8	$\mathbf{g_5}, \mathbf{g_1}$,$g_2$	89	$< \mathbf{0.1}$	**89**	$\mathbf{g_5}, \mathbf{g_1}, \mathbf{g_2}$
9	$\mathbf{g_5}, \mathbf{g_1}, \mathbf{g_2}$,$g_9$	87			

genes once, crossing the ranking from the beginning to the last ranked gene (line 5–12). We obtain the classification accuracy with the first gene in the list (line 9) and it is marked as selected (line 10–12). We obtain the classification rate again with the first and second genes. The second will be marked as selected depending on whether the accuracy obtained is significantly better (line 10). Repeat the process until the last gene on the ranked list is reached. Finally, the algorithm returns the best subset found, and we can state that it will not contain irrelevant or redundant genes.

The first part of the above algorithm is efficient since it requires only the computation of N scores and to sort them, while in the second part, time complexity depends on the learning algorithm chosen. It is worth to note that the learning algorithm is run N (number of genes) times with a small number of genes, only the selected ones. Therefore, the running time of the ranking procedure can be considered as negligible regarding the global process of selection. In fact, the results obtained from a random order of genes (without previous ranking) showed the following drawbacks: 1) the solution was not deterministic; 2) greater number of genes were selected; 3) the computational cost was higher because the classifier used in the evaluation contains more genes since the first iterations.

Consider the situation depicted in Table 1: an example of the gene selection process done by *BIRS*. The first line shows the genes ranked according to some evaluation measure. We obtain the classification accuracy with the first gene in the list (g_5:80%). In the second step, we run the classifier with the first two genes of the ranking (g_5,g_7:82%), and a paired t–test is performed to determine the statistical significance degree of the differences. Since it is greater than 0.1,

g_7 is not selected. The same happens with the next two subsets (g_5,g_4:81%, g_5,g_3:83%). Later, the gene g_1 is added, because the accuracy obtained is significantly better than that with only g_5 (g_5,g_1:84%), and so on. In short, the classifier is run nine times to select, or not, the ranked genes (g_5,g_1,g_2:89%): once with only one gene, four times with two genes, three with three genes and once with four genes. Most of the time, the learning algorithm is run with few genes. In short, this wrapper–based approach needs much less time than others with a broad search engine.

As we can see in the algorithm, the first gene is always selected. This does not mean a great shortcoming in high–dimensional databases, because usually several different sets of genes share similar information. The main disadvantage of *sequential forward generation* is that it is not possible to consider certain basic interactions among genes, i.e., genes that are useless by themselves can be useful together. *Backward generation* remedies some problems although there still will be many hidden interactions (in the sense of being unobtainable), but it demands more computational resources than the forward approach. The computer–load necessities of the forward search might become very inefficient in high–dimensional domains, as it starts with the original set of attributes and removes genes increasingly.

5 Experimental Results

The aim of this section is to evaluate our approach in terms of classification accuracy, degree of dimensionality and speed in selecting genes, in order to see how good *BIRS* is in situations where there is a large number of genes. Four public microarray data sets were used to assess the performance of the algorithm.

Colon cancer data set. This data set is a collection of expression measurements from colon biopsy samples reported by Alon et al. (1999). The data set consists of 62 samples of colon epithelial cells. These samples were collected from colon cancer patients. The "tumor" biopsies were collected from tumors, and the "normal" biopsies were collected from healthy parts of the colons of the same patients. The final assignments of the status of biopsy samples were made by pathological examination. Gene expression levels in these 62 samples were measured using high density oligonucleotide arrays. Of the $\approx$ 6000 genes represented in these arrays, 2000 genes were selected based on the confidence of the measured expression levels.

Leukemia data set. This data set is a collection of expression measurements reported by Golub et al. (1999). The data set contains 72 samples. These samples are divided into two variants of leukemia: 25 samples of acute myeloid

leukemia (AML) and 47 samples of acute lymphoblastic leukemia (ALL). The source of the gene expression measurements was taken from 63 bone marrow samples and 9 peripheral blood samples. Gene expression levels in these 72 samples were measured using high density oligonucleotide microarrays. The expression levels of 7129 genes are reported.

Lymphoma data set. This data set is a collection of expression measurements reported by Alizadeh et al. (2000). The data comprise 96 samples described by the expression values of 4026 genes. There are 9 different subtypes of lymphoma.

GCM data set (Global Cancer Map). This data set is a collection of expression measurements reported by Ramaswamy et al. (2000). The data set contains 190 samples. These samples are divided to 14 variants of tumor. The expression levels of 16063 genes are reported.

Colon and Leukemia are broadly studied, but not Lymphoma and Global Cancer Map, perhaps due to their difficulties at classification tasks.

We chose three different learning algorithms, C4.5, IB1 and Naïve Bayes, to evaluate the accuracy on selected genes for each gene selection algorithm.

As already mentioned, the proposed search was done over a ranking of genes, and any evaluation measure could be used for it. In the experiments, we used two criteria: one belongs to the wrapper model, and one to the filter model. In the wrapper approach, denoted by $BIRS_W$, we order genes according to their individual predictive power, using as criterion the performance of the target classifier built with a single gene. In the filter approach, a ranking is provided using a non–linear correlation measure. We chose symmetrical uncertainty (denoted by $BIRS_F$), based on entropy and information gain concepts.

Due to the high dimensionality of data, we limited our comparison to sequential forward (SF) techniques and fast correlation–based filter (FCBF) algorithm (Yu and Liu, 2004a). We chose three representative subset evaluation measures in combination with SF search engine. One, denoted by SF_W, uses a target learning algorithm to estimate the worth of gene subsets; the other two are subset search algorithms which exploit sequential forward search and use correlation measures (variation of CFS Correlation–based Feature Selection algorithm (Hall, 2000)) or consistency measure (variation of FOCUS (Almuallim and Dietterich, 1994)) to guide the search, denoted by CFS_{SF} and $FOCUS_{SF}$, respectively (both of them used in (Yu and Liu, 2004a)).

The experiments were conducted using the WEKA's implementation of all these existing algorithms and our algorithm is also implemented in the WEKA environment (Witten and Frank, 2005). We must take into account that the proper way to conduct a cross–validation for feature selection is to avoid using

a fixed set of features selected with the whole training data set, because this induces a bias in the results. Instead, one should withhold a pattern, select features, and assess the performance of the classifier with the selected features using the left out examples. The results reported in this section were obtained with a 10–fold cross–validation over each data set, i.e. a feature subset was selected using the 90% of the instances, then, the accuracy of this subset was estimated over the unseen 10% of the data. This was performed 10 times, each time proposing a possible different feature subset. In this way, estimated accuracies, selected attribute numbers and time needed were the result of a mean over ten cross–validation samples. Ambroise and McLachlan (2002) recommends to use 10–fold rather than leave–one–out cross–validation, because the last one can be highly variable. Standard methods have been used for the experimental section (sequential forward; Naïve Bayes, IB1 and C4.5 classifiers; and the t–Student statistical test). There exist other methods following the wrapper approach to extract relevant genes, which involve the selection process into the learning process (neural networks, Bayesian networks, support vector machines), although the source code of these methods is not freely available and therefore the experiments cannot be reproduced. In fact, some of them are designed for specific tasks, so the parameter setting is quite different for the learning algorithm.

Table 2 shows the results obtained with the two BIRS approaches. Tables 3, 4 and 5 report accuracy and number of genes selected by Naïve Bayes, IB1 and C4.5, respectively, by each gene selection algorithm and the original set. We conducted a student's paired two–tailed t–test in order to evaluate the statistical significance of the difference between two averaged accuracy values: one resulted from $BIRS_W$ and the other resulted from one of $BIRS_F$, SF_W, CFS_{SF}, $FOCUS_{SF}$, $FCBF$ and the original set. The symbols "+" and "−", respectively, identify statistic significance, at 0.05 level, wins or losses over $BIRS_W$.

Before comparing our technique to the others, note the similarity in Table 2 between the results obtained with the two approaches of our algorithm, one based on a ranking–wrapper ($BIRS_W$) and the other on a ranking–filter ($BIRS_F$). As we can see in Table 2, in all the cases these accuracy differences are not statistically significant. The number of genes selected are similar too, although $BIRS_F$ is a little bit faster than $BIRS_W$ because of the time needed to build the ranking for the wrapper–ranking approach (see Table 6).

Apart from the previous comparison, we studied the behavior of $BIRS_W$ in three ways in Tables 3, 4 and 5: with respect to a whole set of genes (last row, original); with respect to another wrapper approach (SF_W) ; and to three filter approaches (CFS_{SF}, $FOCUS_{SF}$ and $FCBF$).

Table 6 reports the running time for each gene selection algorithm, showing

Table 2
BIRS accuracy of Naïve Bayes (NB), IB1 (IB) and C4.5 (C4) on selected genes: Acc records 1x10CV classification rate (%) and #g records the number of genes selected by each algorithm. Rank indicates if the classifier is using a wrapper ($_W$) or a filter ranking ($_F$).

Data:		colon		leukemia		lymphoma		gcm	
Classif.	Rank	Acc.	#g	Acc.	#g	Acc.	#g	Acc.	#g
NB	$BIRS_W$	85.48	3.50	93.04	2.50	82.14	10.30	67.37	44.00
	$BIRS_F$	85.48	7.40	93.04	2.80	76.11	9.80	65.26	40.90
IB	$BIRS_W$	79.76	6.30	93.04	3.30	85.56	16.40	58.95	37.00
	$BIRS_F$	79.05	7.60	88.75	4.20	83.44	19.60	60.00	33.10
C4	$BIRS_W$	83.81	2.90	88.57	1.20	80.00	8.80	46.84	9.80
	$BIRS_F$	83.81	2.90	84.64	1.10	81.33	6.60	44.21	24.80

Table 3
Accuracy of Naïve Bayes on selected genes: Acc records 1x10CV accuracy rate (%) and #g records the number of genes selected by each algorithm. N/A–Not available.

Data:	colon		leukemia		lymphoma		gcm	
Algorithm	Acc.	#g	Acc.	#g	Acc.	#g	Acc.	#g
$BIRS_W$	85.48	3.50	93.04	2.50	82.14	10.30	67.37	44.00
SF_W	84.05	5.90	87.32	3.20	83.56	7.10	N/A	
CFS_{SF}	82.62	22.10	91.43	40.30	75.11	153.22	N/A	
$FOCUS_{SF}$	77.14	4.60	84.82^-	2.40	70.07	3.90	56.84	12.20
$FCBF$	77.62	14.60	95.89	45.80	78.22	290.90	68.95	60.90
Original	53.33^-		98.57		75.11		65.79	

three different results for each wrapper approach, depending on the learning algorithm chosen. Obviously, time needed for filter approaches are approximately the same in the three cases, because filters do not depend on the classifier used.

Classification accuracies obtained with our wrapper approach are not statistically different than those obtained with the original set of genes, except for colon data set with NB, where $BIRS_W$ wins (see Table 3). We noticed that the number of selected genes was drastically low with regard to the original set, retaining 0.0018% of the genes on average for the three classifier.

Table 4
Accuracy of IB1 on selected genes: Acc records 1x10CV accuracy rate (%) and #g records the number of genes selected by each algorithm. N/A–Not available.

	colon		leukemia		lymphoma		gcm	
Alg.	Acc.	#g	Acc.	#g	Acc.	#g	Acc.	#g
$BIRS_W$	79.76	6.30	93.04	3.30	85.56	16.40	58.95	37.00
SF_W	66.67^-	4.80	88.93	2.30	80.11	8.40	N/A	
CFS_{SF}	80.71	22.10	90.18	40.30	92.78	153.22	N/A	
$FOCUS_{SF}$	69.29	4.60	81.96^-	2.40	61.22^-	3.90	46.84^-	12.20
$FCBF$	80.71	14.60	94.46	45.80	91.89	290.90	61.05	60.90
Original	77.62		86.25		84.33		57.37	

Table 5
Accuracy of C4.5 on selected genes: Acc records 1x10CV accuracy rate (%) and #g records the number of genes selected by each algorithm. N/A–Not available.

	colon		leukemia		lymphoma		gcm	
Alg.	Acc.	#g	Acc.	#g	Acc.	#g	Acc.	#g
$BIRS_W$	83.81	2.90	88.57	1.20	80.00	8.80	46.84	9.80
SF_W	80.71	3.30	87.32	1.60	73.00	8.20	N/A	
CFS_{SF}	86.90	22.10	84.82	40.30	79.22	153.22	N/A	
$FOCUS_{SF}$	79.05	4.60	88.93	2.40	62.44^-	3.90	49.47	12.20
$FCBF$	88.33	14.60	83.21	45.80	78.22	290.90	52.63	60.90
Original	82.14		82.14		81.44		60.00	

5.1 $BIRS_W$ versus SF_W

No significant statistical differences are shown between the accuracy of our wrapper approach and the accuracy of the sequential forward wrapper procedure (SF_W), except for colon data set and IB classifier, where $BIRS_W$ wins (Table 4).

On the other hand, the advantage of $BIRS_W$ with respect to the SF_W for NB, IB1 and C4.5 is clear. We can observe (see Table 6) that $BIRS_W$ is consistently faster than SF_W, because the wrapper subset evaluation is run less times. For example, for lymphoma data set and C4.5 classifier, $BIRS_W$ and SF_W retain 8.80 and 8.20 genes, respectively, on average. To obtain these subsets the first one evaluated 4026 genes individually (to generate the ranking) and 4026 subsets, while the second one evaluated 32180 subsets (4026 genes

+ 4025 pairs of genes + ... + 4019 sets of eight genes). The time savings of $BIRS_W$ became more obvious when the computer–load necessities of the mining algorithm increased. In many cases the time savings were in degrees of magnitude (leukemia and lymphoma), and we must take into account that SF_W did not report any results on global cancer map data set after one week running.

These results verify the computational efficiency of incremental search applied by $BIRS_W$ over greedy sequential search used by SF_W, with similar number of genes selected and without significant statistical differences on accuracy.

5.2 $BIRS_W$ versus $FOCUS_{SF}$

With respect to the sequential search combined with consistency measure as subset evaluator ($FOCUS_{SF}$), classification accuracies obtained with this filter are lower than those obtained with $BIRS_W$ in five cases for the three classifier. In this sense, $FOCUS_{SF}$ retains very few genes, and in all cases, except for two, accuracies obtained with the original set are greater than those obtained with $FOCUS_{SF}$. We noticed that the computer–load necessities of this filter procedure can be considered as negligible regarding wrapper models (Table 6).

5.3 $BIRS_W$ versus CFS_{SF}

With respect to CFS_{SF} algorithm, where the sequential search is carried out together with a subset evaluation based on correlation measure, the error rate produced by our approach is not significantly different. However, gene subsets and time needed are not similar. Firstly, for the last data set (gcm) no results were provided by CFS_{SF} because the program ran out of memory after a long period of time due to its quadratic space complexity. Secondly, considering that gene subsets provided by filters are not dependent on the classifier used later and those by wrappers are, CFS_{SF} obtained on average 22.10, 40.30 y 153.22 genes for colon, leukemia and lymphoma data sets, respectively, while $BIRS_W$ retained 3.5, 2.5 and 10.3 genes for NB classifier, 6.3, 3.3 and 16.4 genes for IB, and 2.9, 1.2 and 8.8 genes for C4. The time needed to reduce leukemia and lymphoma data sets by CFS_{SF} is almost four times the time used by $BIRS_W$. It is certainly true that CFS_{SF} has problems when data sets have high dimensionality.

5.4 $BIRS_W$ versus $FCBF$

Accuracies obtained with $FCBF$ algorithm are very similar to those obtained with CFS_{SF}, but with higher number of genes and much less time needed. Differences in accuracy between $BIRS_W$ and $FCBF$ are not statistically significant. However, the number of genes selected by $FCBF$ are much higher than that of our algorithm. Gene subsets provided by $FCBF$ are twelve times greater than those provided by $BIRS$, i.e. $FCBF$ retains 0.0224% of the genes on average for the four data sets, while $BIRS_W$ retain only 0.0018% of the genes on average for all data sets and the three classifiers. The computational cost of $FCBF$ is very low (see Table 6).

We used WEKA implementation of the $FCBF$ algorithm with default values. However, if the threshold by which genes can be discarded is modified the results obtained might vary. In Tables 3, 4 and 5, accuracy obtained on average with $FCBF$ for all data sets and all classifiers is 79.27% retaining 0.0224% of the original set of genes. If the threshold is set to 0.25, results obtained are 77.35% of accuracy with 0.0156% of genes, and if 0.50 is fixed as a threshold, results are 59.81% and 0.0011%, respectively. This percentage of genes retained is similar to the obtained with our algorithm, 0.0018%, although $BIRS_W$ provides a higher averaged accuracy, 78.74%.

5.5 Biological interpretation

To complete our study, we reduced each data set by running our gene selection method on the original data sets. BIRS procedure always choose the top gene of each ranking, but generally the rest of the genes are not located at consecutive positions. For colon data set and NB classifier, BIRS chooses the genes M63391 (human desmin gene), H25136 (Inositol 1,4,5-Trisphosphate-binding protein type 2 receptor), M64231 (Human spermidine synthase gene) and R80427 (C4-Dicarbosylate transport sensor protein DCTB) ranked at positions 1, 127, 159 and 160, respectively. For IB, H77597 (H.sapiens mRNA for metallothionein), X12671 (Human gene for heterogeneous nuclear ribonucleoprotein (hnRNP) core protein A1) and H24956 (Proto–oncogene tyrosine–protein kinase receptor ret precursor) at positions 1, 6 and 215, respectively. And J02854 (Myosin regulatory light chain 2), D00860 (Ribose–phosphate pyrophosphskinase I) and M26383 (Human monocyte-derived neutrophil-activating protein mRNA) at positions 1, 6 and 1033, respectively, for the C4 classifier. The first gene of each subset appears in the list of relevant genes detected by previous studies over these data sets (Inza et al., 2004; Hellem and Jonassen, 2002; Ben-Dor et al., 2001).

Table 6
Running time (seconds) for each gene selection algorithm. N/A–Not available.

Data	Clasif.	$BIRS_W$	$BIRS_F$	SF_W	CFS_{SF}	$FOCUS_{SF}$	$FCBF$
colon	NB	27.60	19.89	127.09	13.73	1.40	0.42
	IB	42.90	32.98	120.06	13.71	1.39	0.42
	C4	56.27	55.07	156.92	13.71	1.40	0.42
leuk	NB	105.47	71.30	226.12	498.33	6.65	1.96
	IB	145.86	99.26	228.30	498.46	6.63	1.96
	C4	165.63	118.78	189.74	498.87	6.66	1.98
lymp	NB	309.44	304.58	1357.65	1571.52	5.79	8.71
	IB	235.75	202.38	1032.81	1571.80	5.78	8.60
	C4	579.95	428.43	4540.74	1570.68	5.79	8.68
subtotal		1668.87	1332.68	7979.42	6250.81	41.50	33.14
gcm	NB	11060.86	10626.81	N/A		177.33	26.28
	IB	8669.74	7888.31	N/A		177.39	26.24
	C4	32328.50	25624.29	N/A		177.19	26.28
total		53727.97	45472.08	N/A		573.41	111.95

Similar behaviour appears in the rest of data sets. For leukemia and NB classifier, BIRS chooses the genes M84526, M27891, M31523 and M23197 among the top–20 genes of the ranking and M36652. For IB, M23197, M27891, M31523 and M11722, all of them among the top–20. And only one gene, the first (M27891) for C4 classifier. The first gene of each subset appears in the list of relevant genes detected by previous studies over this data sets (Inza et al., 2004; Hellem and Jonassen, 2002; Ben-Dor et al., 2001). For lymphoma and NB classifier, BIRS chooses three genes among the top–20 genes and five further. For IB, seven and three respectively. And two genes and six for C4 classifier (see Wolowiec et al. (1999)). Finally, for gcm, BIRS chooses three, three and five genes for NB, IB and C4, respectively, among the top–20 genes of the ranking (Yeang et al., 2001; Ramaswamy et al., 2003).

6 Conclusions

The success of many learning schemes, in their attempts to construct data models, hinges on the reliable identification of a small set of highly predictive attributes. Traditional gene selection methods often select the top–ranked

genes according to their individual discriminative power. However, the inclusion of irrelevant, redundant and noisy genes in the model building process phase can result in poor predictive performance and increased computation. The most popular search methods in machine learning can not be applied to microarray expression data sets due to the very high dimensionality, especially when a wrapper approach is used as evaluation function. We use the incremental ranked usefulness definition to decide at the same time whether or not a gene is relevant and non–redundant. The technique extracts the best non–consecutive genes from the ranking, trying to statistically avoid the influence of unnecessary genes in further classifications.

Our approach, named BIRS, uses a very fast search through the attribute space and any classifier can be embedded into it as evaluator. Very highly dimensional datasets take a lot of computational resources when wrappers are chosen. BIRS reduces the search space complexity as it works directly on the ranking, transforming the combinatorial search of sequential forward into a quadratic search. However, the evaluation is much less expensive as only a few genes are selected and therefore the subset evaluation is computationally inexpensive in comparison to other approaches involving wrapper methodologies. Other techniques, faster than BIRS, like FCBF, do not perform very well as they evaluate the relevance over the class individually, and the redundancy between pairs of genes, but they do not consider the interaction among the genes belonging to the final subset.

The analysis has been conducted on four well–known microarray gene expression data sets: lymphoma, leukemia, colon cancer and global cancer map, and all the experiments have been carried out by using an ten–fold cross–validation technique.

In short, our technique $BIRS$ chooses a small subset of genes from the original set (0.0018% on average) with similar predictive performance to others. For very high dimensional datasets, wrapper–based methods might be computationally unfeasible, so BIRS turns out a fast technique that provides good performance in prediction accuracy.

Acknowledgements

The research was supported by the Spanish Research Agency CICYT under grant TIN2004-00159 and TIN2004–06689–C03–03.

References

Alizadeh, A., Eisen, M., Davis, R., Ma, C., Lossos, I., Rosenwald, A., Boldrick, J., Sabet, H., Tran, T., Yu, X., Powell, J., Yang, L., Marti, G., Moore, T., Jr, J. H., Lu, L., Lewis, D., Tibshirani, R., Sherlock, G., Chan, W., Greiner, T., Weisenburger, D., Armitage, J., Warnke, R., Levy, R., Wilson, W., Grever, M., Byrd, J., Botstein, D., Brown, P., Staudt, L., 2000. Distinct types of diffuse large b–cell lymphoma identified by gene expression profiling. Nature 403, 503–11.

Almuallim, H., Dietterich, T., 1994. Learning boolean concepts in the presence of many irrelevant features. Artificial Intelligence 69 (1–2), 279–305.

Alon, U., Barkai, N., Notterman, D., Gish, K., Ybarra, S., Mack, D., Levine, A., 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc. Natl. Acad. Sci. USA 96, 6745–50.

Ambroise, C., McLachlan, G., 2002. Selection bias in gene extraction on the basis of microarray gene-expression data. Proc. Natl. Acad. Sci. USA 99, 6562–6566.

Bell, D., Wang, H., 2000. A formalism for relevance and its application in feature subset selection. Machine Learning 41 (2), 175–195.

Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M., Yakhini, Z., 2001. Tissue classification with gene expression profiles. Proc. Natl. Acad. Sci. USA 98 (26), 15149–54.

Blum, A., Langley, P., 1997. Selection of relevant features and examples in machine learning. In: Greiner, R., Subramanian, D. (Eds.), Artificial Intelligence on Relevance. Vol. 97.

Caruana, R., Freitag, D., 1994. How useful is relevance? In: Working notes of the AAAI fall symp. on relevance. AAAI Press, N. Orleans, LA.

Dash, M., Liu, H., Motoda, H., 2000. Consistency based feature selection. In: Pacific-Asia Conf. on Knowledge Discovery and Data Mining.

Ding, C., Peng, H., 2003. Minimum redundancy feature selection from microarray gene expression data. In: IEEE Computer Society Bioinformatics.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., Lander, E., 1999. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science 286, 531–37.

Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182.

Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene selection for cancer classification using support vector machine. Machine Learning 46 (1-3), 389–422.

Hall, M., 2000. Correlation-based feature selection for discrete and numeric class machine learning. In: 17th Int. Conf. on Machine Learning. Morgan Kaufmann, San Francisco, CA.

Hall, M., Holmes, G., 2003. Benchmarking attribute selection techniques for

discrete class data mining. IEEE Transactions on Knowledge and Data Eng. 15 (3).

Hellem, T., Jonassen, I., 2002. New feature subset selection procedures for classification of expression profiles. Genome Biology 3 (4), 0017.1–0017.11.

Inza, I., naga, P. L., Blanco, R., Cerrolaza, A., 2004. Filter versus wrapper gene selection approaches in dna microarray domains. Artificial Intelligence in Medicine 31, 91–103.

Kohavi, R., John, G., 1997. Wrappers for feature subset selection. Artificial Intelligence 1-2, 273–324.

Koller, D., Sahami, M., 1996. Toward optimal feature selection. In: 13th Int. Conf. on Machine Learning. Morgan Kaufmann, Bari, IT.

Langley, P., 1994. Selection of relevant features in machine learning. In: Procs. Of the AAAI Fall Symposium on Relevance.

Liu, H., Yu, L., 2005. Toward integrating feature selection algorithms for classification and clustering. IEEE Trans. on Knowledge and Data Eng. 17 (3), 1–12.

Piatetsky-Shapiro, G., Tamayo, P., 2003. Microarray data mining: facing the challenges. SIGKDD Explor. Newsl. 5 (2), 1–5.

Ramaswamy, S., Ross, K., Lander, E., Golub, T., 2003. A molecular signature of metastasis in primary tomors. Nature genetics 33, 49–54.

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J., Poggio, T., Gerald, W., Loda, M., Lander, E., Golub, T., 2000. Multiclass cancer diagnosis using tumor gene expression signatures. J Comp Biol 7 (3–4), 559–84.

Witten, I., Frank, E., 2005. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, San Francisco.

Wolowiec, D., Berger, F., Ffrench, P., Bryon, P., Ffrench, M., 1999. Cdk1 and cyclin a expression is linked to cell proliferation and associated with prognosis in non-hodgkin's lymphomas. Leuk Lymphoma 1 (2), 147–57.

Xing, E., Jordan, M., Karp, R., 2001. Feature selection for high-dimensional genomic microarray data. In: Proc. 18th Int. Conf. on Machine Learning. Morgan Kaufmann, San Francisco, CA.

Xiong, M., Fang, X., Zhao, J., 2001. Biomarker identification by feature wrappers. Genome Res 11, 1878–87.

Yeang, C., Ramaswamy, S., Tamayo, P., Mukherjee, S., Rifkin, R., Angelo, M., Reich, M., Lander, E., Mesirov, J., Golub, T., 2001. Molecular classification of multiple tumor types. Bioinformatics 17 (1), S316–22.

Yu, L., Liu, H., 2004a. Efficient feature selection via analysis of relevance and redundancy. Journal of machine learning research 5, 1205–24.

Yu, L., Liu, H., 2004b. Redundancy based feature selection for microarry data. In: 10th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining.