Intelligent Data Analysis 16 (2012) 717–737
DOI 10.3233/IDA-2012-0546
IOS Press

MaskedPainter: Feature selection for microarray data analysis

Daniele Apiletti, Elena Baralis, Giulia Bruno and Alessandro Fiori*
Dipartimento di Automatica e Informatica, Politecnico di Torino, Torino, Italy

Abstract. Selecting a small number of discriminative genes from thousands is a fundamental task in microarray data analysis. An effective feature selection allows biologists to investigate only a subset of genes instead of the entire set, thus avoiding insignificant, noisy, and redundant features. This paper presents the MaskedPainter feature selection method for gene expression data. The proposed method measures the ability of each gene to classify samples belonging to different classes and ranks genes by computing an overlap score. A density based technique is exploited to smooth the effects of outliers in the overlap score computation. Analogously to other approaches, the number of selected genes can be set by the user. However, our algorithm may automatically detect the minimum set of genes that yields the best classification coverage of training set samples. The effectiveness of our approach has been demonstrated through an empirical study on public microarray datasets with different characteristics. Experimental results show that the proposed approach yields a higher classification accuracy with respect to widely used feature selection techniques.

Keywords: Feature selection, microarray analysis, tumor classification, data mining

1. Introduction

The rapid advance of molecular biology techniques provides measures of gene expression levels (concentration of mRNA) of thousands of genes simultaneously. DNA microarray experiments generate thousands of gene expression measurements and provide a simple way for collecting huge amounts of data in a short time. They are used to collect information from tissue and cell samples regarding gene expression differences. Compared with traditional tumor diagnostic methods, based mainly on the morphological appearance of the tumor, methods relying on gene expression profiles are more objective, accurate, and reliable [20]. However, microarray data are highly redundant and noisy. Most genes are uninformative and only a subset of features may present distinct profiles for different classes of samples.

Since not all genes provide useful information, a dimensionality reduction process is needed to identify and remove as much of the redundant and unnecessary information as possible. Dimensionality reduction is the process of reducing the number of variables under consideration, and can be performed by means of feature selection or feature extraction. Feature selection obtains a subset of the original variables (also called features or attributes), while feature extraction applies a transformation of the original space into a space with fewer dimensions. Since from a biological viewpoint it is more important to select real genes than to create artificial features with uncertain biological meaning, we focus our attention on feature

*Corresponding author: Alessandro Fiori, Alessandro Fiori, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy. E-mail: alessandro.fiori@polito.it.

selection. Feature selection allows the identification of the genes that are relevant or mostly associated with a tissue category, disease state or clinical outcome. Furthermore, when a small number of genes is selected, their biological relationship with the target disease is more easily identified, thus providing additional scientific understanding of the problem.

Traditional methods in gene selection are filter-based models and often evaluate genes in isolation without considering correlation among them. They rank genes according to their individual relevance or discriminative power to the target class and select the top-ranked genes. These methods are efficient, but cannot remove redundant genes, because simply selecting two highly ranked genes may not form a better gene set if the two genes are highly correlated. A key challenge in gene selection from microarray data is to provide biologists with an efficient filter method that identifies and selects only relevant genes in an automatic manner [54].

Another critical issue of feature selection is finding the optimal number of features that yield good classification accuracy. Only few works address the problem of defining the appropriate number of genes to select [12]. While an excessively conservative estimate may cause an information loss due to relevant features exclusion, an excessively liberal estimate may leave noise in the resulting gene set. For example, for a two-class cancer subtype classification problem, a few tens of genes are usually selected, even if some studies suggest that one or two genes may be already sufficient [31].

In this paper we present the *MaskedPainter* feature selection method. The MaskedPainter method provides two main contributions: (i) it identifies the minimum number of genes that yield the best coverage of the training data (i.e. that maximize the correct assignment of the training samples to the corresponding class), and (ii) it ranks the genes according to a quality score. A density based technique is exploited in the quality score computation to smooth the effect of outliers. The minimum gene subset and the top ranked genes are then combined to obtain the final feature set.

The name "MaskedPainter" originates from the Painter's algorithm used in computer graphics. The Painter's algorithm assigns a priority to the objects to be painted that is based on their overlaps. Similarly, the MaskedPainter assigns a priority to genes that is based on the overlaps in their expression intervals. The term masked is due to the fact that the information carried by a gene is represented in a particular format, named gene mask.

We validated our method on different microarray datasets. We mainly focused on multi-category datasets, because classification problems with multiple classes are generally more difficult than binary ones and give a more realistic assessment of the proposed methods [22]. We compared the MaskedPainter performance with different feature selection techniques by measuring the classification accuracy provided by a classifier taking as input the selected features. In almost all circumstances, the MaskedPainter yields statistically significant higher accuracy than the other techniques. All experiments have been performed for different gene set cardinalities and with different classifiers. Finally, the biological relevance of the genes selected by the proposed approach has been assessed.

The paper is organized as follows. Section 2 discusses related works. Section 3 describes the Masked-Painter feature selection method, while Section 4 presents the experiments we performed to validate our approach, included a biological discussion. Finally, Section 5 draws conclusions.

2. Related work

Feature selection is a fundamental task in the bioinformatics domain to identify the most relevant genes correlated with the considered sample classes. In literature many studies have been addressed to this issue. However, evaluating feature selection techniques and comparing results among different

works is very difficult due to the lack of both standard experimental designs and benchmark datasets containing a sufficient number of relevant genes known in biological literature [49]. A currently adopted way to evaluate feature selection methods is to use as score the accuracy of a classifier applied after the feature selection. The higher is the accuracy, the more relevant the selected genes are. For example, the authors [24] analyze several feature selection methods available in the RankGene software [40] and show that the choice of feature selection criteria can have a significant impact on classification accuracy.

The feature selection methods can be categorized in three categories: (i) filter, (ii) wrapper and (iii) embedded. In the following we describe the main characteristics of these approaches.

Filter methods use general characteristics of the data to to detect differentially expressed genes. A gene is differentially expressed if it shows a certain distribution of expression levels under one condition and a significantly different distribution under other conditions. Filter methods aim at evaluating the differential expression of genes and rank them according to their ability to distinguish among classes. For example, a P-metric correlation [18] which measures the difference between the samples relative to the standard deviation of samples was exploited. Moreover, these approaches operate independently of any learning algorithm and require less computation. We can identify two main sub-categories of filter methods: (i) univariate and (ii) multivariate.

The univariate approaches are based usually on statistical measures used for detecting differences between two groups (including t-test, Wilcoxon test, and Mann-Whitney test), or among three or more groups (ANOVA, Kruskal-Wallis test, and Friedman test) [27,44]. These methods are easily and very efficiently computed but the disadvantage is that they require some assumptions on the data distribution. For example, the t-test requires expression levels to be normally distributed and homogeneous within groups and may also require equal variances between groups. A Bayesian version of t-test by means of a derivation of point estimates for parameters and hyperparameters was proposed in [14].

Instead, the multivariate approaches are devoted to evaluates the correlations among genes belonging to the selected subset. For example, in [33] a k-means clustering is applied to partition the initial set of genes. A filter approach based on the normalized mutual information metric is conducted in each cluster. A sequential forward search in the space of subsets of genes is performed until a predefined number of genes is achieved. Other approaches are based on searching procedures to identify the best subset of genes. In [37] a multivariate score formulated by the CFS algorithm is integrated as evaluation function of a genetic algorithm.

Wrapper methods evaluate the usefulness of a gene by estimating the accuracy of the learning method applied only to selected genes and not to the entire dataset. The aim of this kind of analysis is to select the features that optimize the performance of the target classifier. It is computationally very expensive for data with a large number of features and the selected subset is dependent on the considered learning algorithm.

Among the feature selection categories, the wrapper methods typically require extensive computation to search the best features and depend on the considered learning algorithm [25]. For example, the authors [32] proposed an approach based on genetic algorithms. The Silhouette statistic is used to assign a score to each subset. In order to reduce the search space, a pre-selection of genes is done by the BSS/WSS univariate filter metric [13]. Moreover, wrapper approaches usually require a further step to avoid the exhaustive search among all the possible solutions, because the number of feature subsets grows exponentially with the number of features, making enumerative search unfeasible. In [35] an univariate rank is first computed for all the genes, and then this ranked list is crossed by a wrapper procedure which incrementally augments the subset of selected genes.

The embedded approaches have the advantage of including the interaction with the classification model, while at the same time being far less computationally intensive than wrapper methods. For example,

in [19] the features are ranked with the magnitude of the weights in the SVM classifier. The relevance of each feature is assessed by the calculation of the resubstitution error during a recursive feature elimination procedure. In [15] a variation of this method was proposed. The entropy values of the SVM weights are discretized to eliminate chunks of irrelevant genes. Differently, in [11] the ensemble nature of the decision trees constructed in the random forest framework is exploited to compute the relevance of each feature.

A particular issue of feature selection task is finding the optimal number of genes which improve classification accuracy and show high correlation with disease outcomes. For example, for a two-class cancer subtype classification problem, few tens of genes are usually sufficient, even if there are studies which suggest that one or two genes may be enough [31]. The authors [45] test the classification capability of all simple combinations of the top genes. Firstly, they classify the data set with only one top gene. If the accuracy is not sufficient, they consider all 2-gene combinations of top genes, and consider an increasing number of genes until a good accuracy is reached. High accuracy values are provided by only three or four genes.

For a comparison of various feature selection methods on different microarray data see [16,22,29,42].

3. The MaskedPainter approach

The microarray data E are in the form of a gene expression matrix Eq. (1), in which each row represents a gene and each column represents a sample. For each sample, the expression levels of all the genes in study are measured.

$$E = \begin{bmatrix} e_{11} & e_{12} & \dots & e_{1M} \\ e_{21} & e_{22} & \dots & e_{2M} \\ \dots & \dots & \dots & \dots \\ e_{N1} & e_{N2} & \dots & e_{NM} \end{bmatrix} \tag{1}$$

Let e_{ij} be the measurement of the expression level of gene i for sample j where $i = 1, \dots, N$ and $j = 1, \dots, M$. Each sample is also characterized by a class label, representing the clinical situation of the patient or tissue being analyzed. The domain of class labels is characterized by C different values and label l_j of sample j takes a single value in this domain.

The MaskedPainter method is based on the following idea. Certain genes can identify samples belonging to a class, because their expression interval in that class is not overlapped with the expression intervals of other classes (i.e., all the samples for which the expression value of the gene is in a given range belong to a single class). For example, Fig. 1(a) shows the expression values of a gene with 12 samples belonging to 3 different classes. The same information can be represented as shown in Fig. 1(b). With the latter representation, it is easy to see that the gene is relevant for class 3, because the expression values of this class are concentrated in a small range, which is different from the range of expression values associated with the other classes. Instead, the same gene is not useful to distinguish between class 1 and 2, because the values for such classes have mostly overlapping ranges.

The MaskedPainter initially characterizes each gene by means of a *gene mask*, which represents the gene's capability of unambiguously assigning training samples to the correct class. Next, the method assigns to each gene two values that are then combined in the ranking phase: the *overlap score* and the *dominant class*. The overlap score is a quality index that describes the overlap degree of the expression intervals for different classes. Genes with less overlapping intervals are more important because they

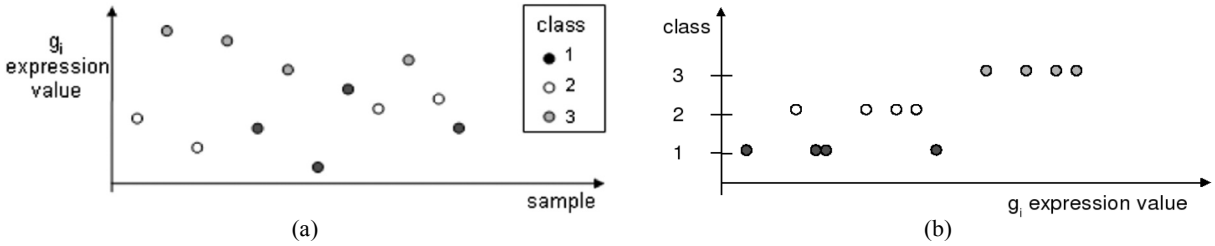

Fig. 1. Two different representations of a gene with 12 samples belonging to 3 classes.

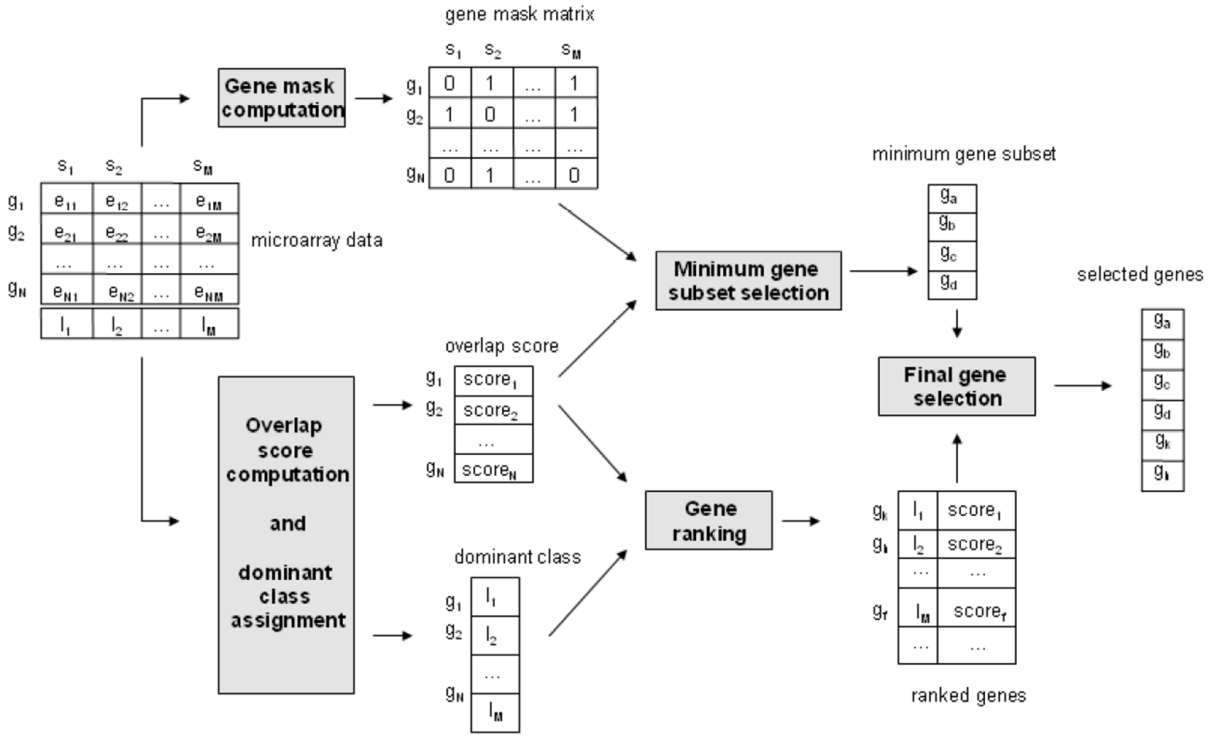

Fig. 2. Building blocks of the MaskedPainter method.

can unambiguously assign the samples to the correct class. The dominant class of a gene is the class to which the majority of samples without overlapping expression intervals belong.

By exploiting these elements, the MaskedPainter defines (i) the minimum set of genes needed to provide the best sample coverage on training data and (ii) a sort of genes by dominant class and increasing value of overlap score. The final gene set is obtained by combining the minimum gene subset and the top ranked genes in the sort.

The building blocks of the MaskedPainter approach are presented in Fig. 2. The approach is based on the following main phases.

Gene mask computation. A gene mask is computed for each gene. The gene mask shows which training samples the gene can unambiguously assign to the correct class. It is a string of 0s and 1s, generated by analyzing the overlaps among the class expression intervals (i.e., the range of expression values of samples belonging to the same class) for each gene.

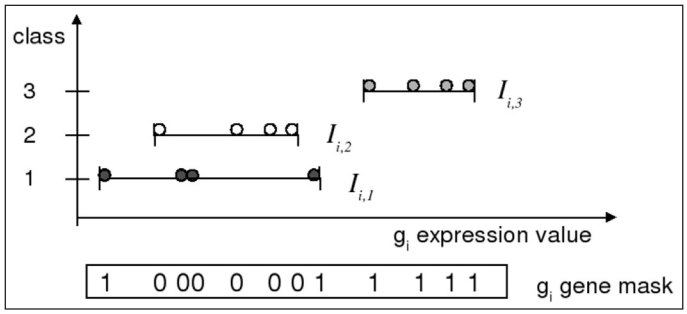

Fig. 3. Class expression intervals and gene mask for a gene with 12 samples belonging to 3 classes.

Overlap score computation and dominant class assignment. An overlap score is assigned to each gene. It assesses the overlap degree among its core expression intervals. The core expression interval of a gene, separately defined for each class, is the expression interval obtained by smoothing the effect of outliers. To compute the core expression intervals, a density based approach is proposed. A dominant class, i.e., the best distinguished class, is also assigned to each gene. This information is exploited to reduce redundancy when genes are selected.

Minimum gene subset selection. The minimum number of genes needed to provide the best training set sample coverage is selected by analyzing the gene masks and exploiting the overlap scores. A greedy approach is exploited to achieve the best sample coverage with a low computational cost.

Gene ranking. Genes that do not belong to the minimum subset are ranked according to increasing values of overlap score, separately for each dominant class. The final gene rank is composed by selecting the topmost gene from each dominant class in a round robin fashion.

Final gene selection. Selected top ranked genes are added to the minimum gene subset, thus providing the final gene set.

All phases are detailed in the following.

3.1. Gene mask computation

We introduced a compact representation, named gene mask, to represent the *discriminative power* of each gene. The discriminative power is the capability of a gene to assign correctly the class label to each sample.

By considering the gene expression matrix in (1), with a class label for each sample, the class expression intervals for each gene are defined. Let i be an arbitrary gene with M samples belonging to C classes. For each gene i we define C class expression intervals (one for each class). The class expression interval for gene i and class k is expressed in the form:

$$I_{i,k} = \left[\min_{i,k}, \max_{i,k}\right] \tag{2}$$

where $\min_{i,k}$ and $\max_{i,k}$ are the minimum and the maximum gene expression values for class k. A graphical example of class expression intervals is presented in Fig. 3.

Since few samples usually belong to each class, the class expression interval contains the entire expression value range for the corresponding class, thus avoiding any assumption on data distribution. The gene mask exploitation aims at reaching the best coverage of the original data of the training set,

thus, no preprocessing (e.g., outlier smoothing) is performed and the min-max expression interval is adopted.

The gene mask is an array of M bits, where M is the number of samples. Consider an arbitrary gene i. Bit j of its mask is set to 1 if the corresponding expression value e_{ij} belongs to the class expression interval of a single class (i.e., it assigns unambiguously the sample to the correct class), otherwise it is set to 0. Formally, given two classes $a, b \in \{1, \ldots, C\}$, bit j of gene mask i is computed as follows.

$$mask_{ij} = \begin{cases} 1 \text{ if } (e_{ij} \in I_{i,a}) \wedge \nexists b \neq a \mid e_{ij} \in I_{i,b} \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

Figure 3 shows the gene mask associated to gene i after the computation of its three class intervals $(I_{i,1}, I_{i,2}, I_{i,3})$.

The gene masks are employed in the minimum gene subset selection step to identify the features which provide the best coverage of the training set samples.

3.2. Overlap score computation and dominant class assignment

An overlap score is assigned to each gene, depending on the amount of overlapping expression intervals among classes. Differently from the gene mask, which is based on the min-max expression intervals (as discussed in Section 3.1), the overlap score aims at modeling the discrimination power of genes and needs to handle noise and outliers to avoid overfitting. Hence, to better model the expected values in an unseen test set, the overlap score computation is based on core expression intervals. Core expression intervals are expression intervals obtained by smoothing the effect of outliers by means of a density-based technique.

The dominant class, i.e., the class with the highest number of samples in non-overlapping intervals, is also assigned to each gene.

This phase can be further divided into three steps, which are detailed in the following: (i) core expression interval definition, (ii) overlap score computation, and (iii) dominant class assignment.

3.2.1. Core expression interval definition

Since microarray data may present noisy values, the influence of values far from the high concentration nucleus (typically outliers) must be reduced. Several techniques have been exploited to reduce the influence of outliers in microarray data [9,52], which are variations of the formula $mean \pm stdev$, such as 3sigma or the Hampel identifier. However, these methods do not consider the density of values. Thus, we propose a density based method to reduce the effect of outliers in the computation of the core expression intervals. The *core expression interval* of a gene in a class is defined by replacing the mean by the weighted mean and the standard deviation by the weighted standard deviation. It models the possible distribution of unseen expression values. A weight is assigned to each data value by considering the number of its neighbors belonging to the same class. In particular, a higher weight is assigned to values with many neighbors and a lower weight to isolated values.

Consider an arbitrary sample j belonging to class k and its expression value e_{ij} for an arbitrary gene i. Let the expression values be independent and identically distributed (i.i.d) random variables and $\sigma_{i,k}$ be the standard deviation for the expression values of gene i in class k. The density weight d_{ij} measures, for a given expression value e_{ij}, the number of neighboring expression values of samples of the same class. A conservative estimation of the interval width is represented by the interval $\pm\sigma_{i,k}$ centered in e_{ij}, because $\sigma_{i,k}$ identifies a neighborhood in which 68% of points should lie when a normal distribution is centered on the considered point.

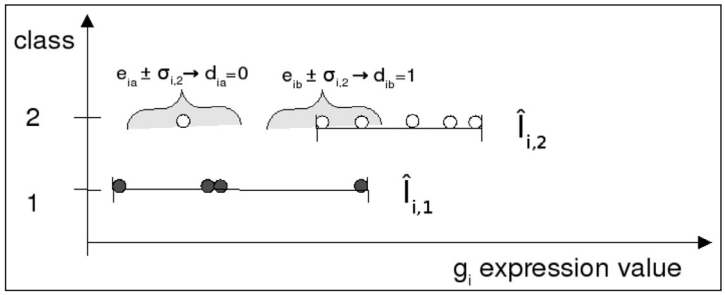

Fig. 4. Density weights to define core expression intervals for a gene with 10 samples belonging to 2 classes.

The density weight for the expression value e_{ij} in class k is defined as

$$d_{ij} = \sum_{m=1, m \neq j}^{M} \delta_{im} \tag{4}$$

where δ_{im} is defined as:

$$\delta_{im} = \begin{cases} 1 \text{ if sample } m \text{ belongs to class } k \ \wedge e_{im} \in [e_{ij} - \sigma_{i,k}; e_{ij} + \sigma_{i,k}] \\ 0 \text{ otherwise} \end{cases} \tag{5}$$

If an expression value is characterized by many neighboring values belonging to the same class, its density is higher. For example, in Fig. 4 an arbitrary gene i with 10 samples belonging to two classes is shown. The first class 2 sample (denoted as e_{ia} in Fig. 4) is characterized by a density weight d_{ia} equal to 0, because there are no other expression values for gene i in the interval $e_{ia} \pm \sigma_{i,2}$ (represented by a curly bracket). The second class 2 sample (e_{ib}) is characterized instead by a density weight d_{ib} equal to 1, because another sample of class 2 belongs to the interval $e_{ib} \pm \sigma_{i,2}$.

The core expression interval of an arbitrary gene i in class k is given by

$$\hat{I}_{i,k} = \hat{\mu}_{i,k} \pm (2 \cdot \hat{\sigma}_{i,k}) \tag{6}$$

where the mean $\hat{\mu}_{i,k}$ and the standard deviation $\hat{\sigma}_{i,k}$ are based on the density weights and are computed as follows.

The mean $\hat{\mu}_{i,k}$ is defined as

$$\hat{\mu}_{i,k} = \frac{1}{D_{i,k}} \sum_{j=1}^{M} \delta_{ij} \cdot d_{ij} \cdot e_{ij} \tag{7}$$

where δ_{ij} is defined as:

$$\delta_{ij} = \begin{cases} 1 \text{ if sample } j \text{ belongs to class } k \\ 0 \text{ otherwise} \end{cases} \tag{8}$$

and $D_{i,k}$ is the sum of density weights for gene i in class k (i.e., $\sum_{j=1}^{M} \delta_{ij} \cdot d_{ij}$).

The standard deviation $\hat{\sigma}_{i,k}$ is given by

$$\hat{\sigma}_{i,k} = \sqrt{\frac{1}{D_{i,k}} \sum_{j=1}^{M} \delta_{ij} \cdot d_{ij} \cdot (e_{ij} - \hat{\mu}_{i,k})^2} \tag{9}$$

The core expression intervals model a normal distribution of expression profiles with a confidence interval equal to 95%. Thus, the resulting intervals are less affected by outlier values, as shown in Fig. 4. Since the first class 2 sample (on the left) has a density weight equal to zero, its value provides no contribution to the core interval computation, and thus it is not included in $\hat{I}_{i,2}$.

The proposed approach has been experimentally compared with alternatives which consider as boundaries $\mu \pm \sigma$ or the minimum and maximum (min-max) values of the expression values. For the min-max technique, interval boundaries are strongly affected by outliers, because no filtering is performed on noisy microarray data. Using $\mu \pm \sigma$ as boundaries, the filtered intervals are excessively reduced and significant values may be lost. A similar problem affects the Hampel identifier [9], also called the median absolute value (MAD). The expression intervals generated by MAD tend to be narrower than the ones obtained by the proposed density weighted method. If intervals are narrower, they tend to be less overlapped. Thus, a gene may be considered more relevant than it actually is, because of a less conservative overlap detection.

3.2.2. Overlap score computation

For each gene we define an overlap score (denoted as os in the following) that measures the degree of overlap among core expression intervals. Since overlapping intervals may lead to misclassifications due to insufficient discriminative power of the considered gene, the overlap score is exploited for ranking genes. The score is higher for less important genes with many overlapping intervals among different classes. On the contrary, lower scores denote genes with higher discriminating power, because they have few overlaps among their intervals.

The os depends on the following characteristics of the gene expression values

A) the number of samples associated to different classes in the same range,
B) the number of overlapping classes,
C) the overlapping interval length.

We compute the overlap score os_i for each gene i. To ease readability, we will omit the i subscript in the following formulas.

We define the total expression interval of a gene as the range given by the minimum and maximum among its core expression interval boundaries. We denote such interval as W, and its amplitude as $|W|$. For example, in Fig. 5, the total expression interval of a gene with samples belonging to three classes (and thus with three core expression intervals) is shown. We divide W in subintervals, where each subinterval is characterized by a different set of overlapping classes with respect to the adjacent subintervals. More specifically, the subinterval w_t is defined as the interval delimited by two consecutive extremes of core expression intervals, as shown in Fig. 5. The amplitude of subinterval w_t is denoted as $|w_t|$.

The idea of the overlap score is to assign higher scores to genes that are characterized by more overlaps among expression intervals of different classes. The score is based on both the number of samples of different classes that belong to the same subinterval and the amplitude of the subinterval itself. Thus, subintervals with larger class overlaps provide a higher contribution to the os. According to this intuition, the overlap score for an arbitrary gene is defined as follows.

$$os = \sum_{t=1}^{T} k(t) \frac{m_t}{M} \frac{|w_t|}{|W|} \tag{10}$$

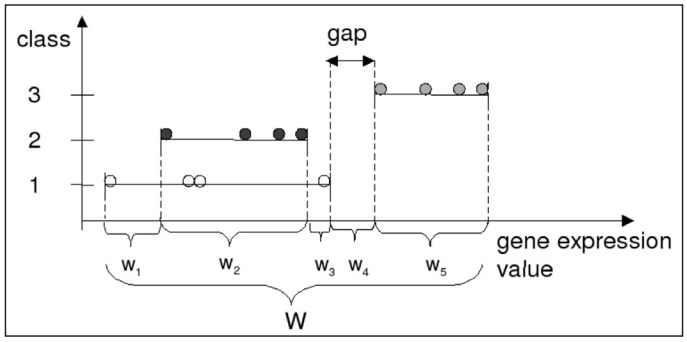

Fig. 5. Subintervals for the computation of the overlap score (os) of a gene.

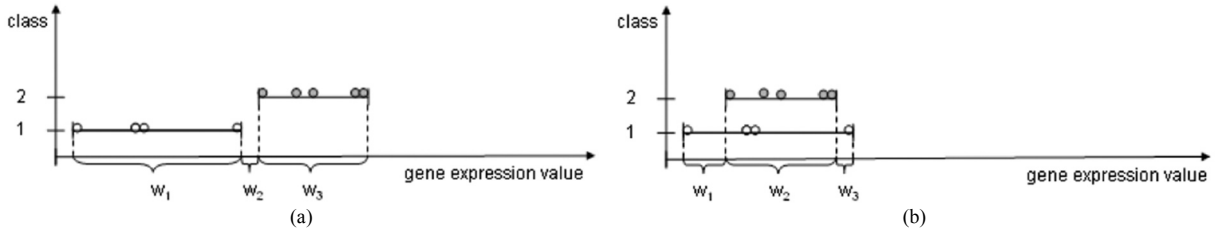

Fig. 6. Overlap score computation for two core expression intervals: (a) a gene with an overlap score equal to 0 and (b) a gene with an overlap score close to 2.

where T is the number of subintervals, m_t is the number of samples expressed in subinterval t, M is the total number of samples. The function $k(t)$ evaluates the set of classes that overlap in the subinterval t and is defined as follows.

$$k(t) = \begin{cases} |C_t| & \text{if } |C_t| \geqslant 2 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

where $|C_t|$ is the number of classes belonging to the subinterval t. Thus, subintervals covered by a single class (e.g., w_1 in Fig. 5) provide no contribution to the overlap score, because the number of overlapping classes is 0. In the case of subintervals without values (i.e., gaps such as w_4 in Fig. 5), the number of overlapping classes is 0. Thus, also in this case, no contribution is added to the overlap score.

Consider the example in Fig. 5. The total expression interval of the gene is divided into five subintervals and the c_t components (number of overlapping classes in each subinterval) take the following values: $c_1 = 0$, $c_2 = 2$, $c_3 = 0$, $c_4 = 0$, $c_5 = 0$.

The os value ranges from 0, when there is no overlap among class intervals, to C (i.e., the number of classes), when all intervals are completely overlapped. For example, in Fig. 6, two illustrative cases for a binary problem are reported. Figure 6(a) shows a gene that correctly distinguishes two classes, because its core expression intervals are not overlapped. The os of this gene is equal to 0, because $c_1 = 0$, $c_2 = 0$, and $c_3 = 0$. Instead, Fig. 6(b) shows a gene unable to distinguish two classes, because the expression intervals associated with the two classes are almost completely overlapped. In this case, the overlap score is close to 2.

3.2.3. Dominant class assignment

Once the overlap score is computed, we associate each gene to the class it distinguishes best, i.e., to its *dominant class*. To this aim, we consider the subintervals where expressed samples belong to a single

class and evaluate the percentage of samples for the considered class in these subintervals. The class with the highest number of samples is the dominant class of the gene. The gene is assigned to the class with the highest number of samples to take into account the a priori probability of the classes. Associating a gene with the class it distinguishes best will allow us to balance the number of selected genes per class (see Section 3.5).

For example, in Fig. 6(b) the gene dominant class is class 1, because its samples for the two non-overlapping subintervals w_1 and w_3 are labeled with class 1. Instead, in Fig. 6(a), since both classes have completely non-overlapping intervals, the gene dominant class is class 2, according to the number of samples.

3.3. Minimum gene subset selection

The information provided by the gene masks is employed to identify the minimum set of genes which classify correctly the maximum set of samples in the training set. It also allows removing redundant information (e.g., genes with low discriminative power or with similar expression profiles). The final objective is the identification of a set of high quality features covering the given sample set.

Let S be a set of genes. We define a *global mask* as the logic OR between all the gene masks belonging to genes in S. The objective is the definition of the minimum set of genes S that holds enough discriminating power to unambiguously assign to the correct class the maximum number of samples in the training set. Thus, given the gene mask of each gene, we search for the global mask with the maximum number of ones. To this aim, we propose a greedy approach.

Greedy approach. The greedy approach identifies at each step the gene with the best complementary gene mask with respect to the current global mask. Thus, it adds at each step the information for classifying most currently uncovered samples.

The pseudo-code of the Greedy approach is reported in Algorithm 1. It takes as input the set of gene masks ($\mathcal{M}$), the set of overlap scores ($\mathcal{OS}$) and produces as output the minimum subset of genes ($\mathcal{G}$). The first step is initializing $\mathcal{G}$ at $\emptyset$ (line 2), the candidate set ($\mathcal{C}$) at $\emptyset$ (line 3), and the global mask with all zeros (line 4). Then the following steps are iteratively performed.

A) The gene mask with the highest number of bits set to 1 is chosen (line 8). If more than one gene mask exists, the one associated to the gene with the lowest overlap score is selected (lines 9–16).
B) The selected gene is added to set $\mathcal{G}$ (line 18) and the global mask is updated by performing the logical OR between the gene mask and the global mask (line 19).
C) The gene masks of the remaining genes (gene mask set $\mathcal{M}$, line 20) are updated by performing the logical AND with the negated global mask (lines 21–24). In this way, only the ones corresponding to the classification of still uncovered samples are considered.
D) If the global mask has no zeros (line 6) or the remaining genes have no ones (line 9), the procedure ends.

At the end of this phase, the minimum set of genes required to provide the best sample coverage of the training set is defined. The genes in the minimum subset are ordered by decreasing number of 1s in the gene mask.

3.4. Gene ranking

The gene rank is defined by considering both the overlap score and the dominant class. Genes that do not belong to the minimum subset are ranked by increasing value of overlap score separately for each

Algorithm 1 Minimum gene subset – Greedy approach

Input: set $\mathcal{M}$ of all the $mask_i$, set $\mathcal{OS}$ of overlap score os_i for each gene i
Output: set $\mathcal{G}$ of genes

```
/*Initialization*/
G = ∅
C = ∅ /*candidate gene set at each iteration*/
global_mask = all_zeros() /*vector with only 0s*/
/*Control if the global mask contains only 1s*/
while not global_mask_all_ones() do
    /*Determine the candidate set of genes with most ones*/
    C = max_ones_genes()
    if C != ∅ then
        /*Select the candidate with minimum overlap score*/
        c = C[1]
        for all j in C[2 :] do
          if OS_j < OS_c then
              c = j
          end if
        end for
        /*Update sets and global_mask*/
        G = G + c
        global_mask = global_mask OR M_c
        M = M − M_c
        /*Update the masks belongs to M*/
        for all M_i in M do
            M_i = M_i AND ¬global_mask
        end for
    else
        break
    end if
end while
return G
```

dominant class. The final rank is composed by selecting the topmost gene from each dominant class rank in a round-robin fashion.

A feature selection method considering only a simplified version of the overlap score was presented in [3]. The overlap score alone ranks high genes with few overlaps, without considering the class they distinguish best. Hence, high-ranked genes may all classify samples belonging to the same class, thus biasing gene selection (typically by disregarding less populated classes). The round-robin gene selection by dominant class allows mitigating this effect.

3.5. Final gene selection

The minimum gene subset includes the minimal number of genes that provide the best sample coverage on the training set, ordered by decreasing number of 1s in the gene mask. However, a larger set of genes may be either beneficial to improve the classification accuracy on unseen (test) data, or directly requested by the user. In this case, the minimum gene subset is extended by including the top k ranked genes in the gene ranking, where k is set by the user. Observe that genes in the minimum subset are inserted in the final gene set independently of their overlap score, because these features allow the classifier to cover the maximum set of training samples. The effect of this choice is discussed in Sections 4.3 and 4.4.

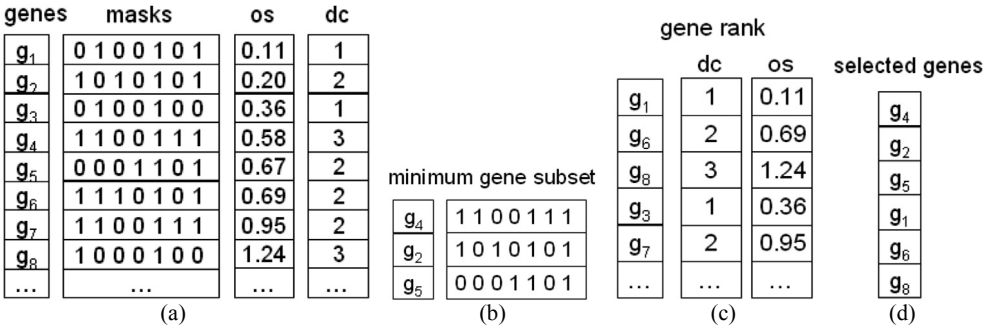

Fig. 7. An example of the MaskedPainter method: (a) genes with their mask, overlap score, and dominant class; (b) minimum gene subset obtained by applying the greedy algorithm; (c) gene ranked by dominant class and overlap score; (d) selected genes at the end of the process.

3.6. Example

As a summarizing example, consider the set of genes represented in Fig. 7(a). Each gene is associated with its overlap score (os), its gene mask (string of 0 and 1), and its dominant class (dc). For example, gene $g1$ has a mask of 0100101 (i.e., it classifies unambiguously the second, the fifth and the seventh samples), an overlap score of 0.11, and its dominant class is class 1. For convenience, genes are pre-sorted by increasing overlap score value.

The first gene selected by the greedy method in the minimum subset is $g4$, because it is characterized by the highest number of bits set to 1 (the same as $g6$ and $g7$) and the lowest overlap score. Then, genes with the best complementary masks are $g2$, $g5$, and $g6$, which all have the same number of bits set to 1. Again, $g2$ is selected because of its lower overlap score. Eventually, the only gene with a complementary mask, which is $g5$, is chosen. In this case the minimum number of genes is three. In Fig. 7(b) the genes in the minimum gene subset are reported.

The remaining genes are divided by dominant class and sorted by ascending overlap score. The gene rank is composed by selecting the topmost gene from each dominant class in a round robin fashion (e.g., g_1 for class 1, g_6 for class 2, g_8 for class 3, g_3 for class 1, etc.) as shown in Fig. 7(c). Suppose that six genes are required by the user for its biological investigation. Then, the three top ranked genes are added to the three genes of the minimum gene subset. The final gene set is shown in Fig. 7(d).

4. Experimental results

We validated the MaskedPainter method by comparison with other feature selection techniques on public gene expression datasets. Classification accuracy is used as the performance metric for evaluation. The biological relevance of the selected genes is discussed in Section 4.5. We performed a large set of experiments addressing the following issues.

A) *Classification accuracy.* The accuracy yielded by MaskedPainter and several other feature selection approaches on seven public datasets is analyzed.
B) *Cardinality of the selected feature set.* The impact on classification accuracy of different numbers of selected genes (from 2 to 20) is analyzed.
C) *Minimum gene subset definition.* The proposed greedy approach is compared with a set covering method by considering both accuracy and computational cost.

Table 1
Dataset characteristics

Dataset	*Samples*	*Genes*	*Classes*
Alon	62	2000	2
Brain1	90	5920	5
Brain2	50	10367	4
Leukemia	72	5327	3
Srbct	83	2308	4
Tumor9	60	5727	9
Welsh	34	7129	2

Table 2
Feature selection techniques

Name	*Abbr*
Information Gain	IG
Twoing Rule	TR
Sum Minority	SM
Max Minority	MM
Gini Index	GI
Sum of Variance	SV

The experiments addressing each issue are reported in the following subsections.

We also analyzed the computational cost of our approach. We measured the time required by each approach to extract a high number of features (i.e., 1000 features) from the considered datasets. The MaskedPainter algorithm proved to be as efficient as the competing feature selection methods. In particular, on a Pentium 4 at 3.2 GHz with 2 GByte of RAM, the time required to extract the top 1000 genes on any complete dataset is in the order of few seconds (e.g., less than 1 second on the Alon dataset, 3 seconds on the Brain2 dataset) and very similar to the time required by the other methods.

4.1. Experimental setting

We validated our feature selection approach on 7 multicategory microarray datasets, publicly available on [2,39,47]. Table 1 summarizes the characteristics of the datasets. Five are characterized by 3 to 9 classes, while two are bi-class datasets. Most contain between 60 and 90 samples, whereas one has 34 samples. The number of features ranges from 2 thousands to more than 10 thousands.

For each dataset we selected the best subset of genes according to the 6 feature selection methods reported in Table 2 available in RankGene [40], besides our approach. These feature selection methods are widely used in machine learning [30]. Furthermore, they are used as comparative methods in many feature selection studies on microarray data [21,36].

The experimental design exploits 50 repetitions of 4-fold stratified cross validation for each parameter set (i.e., given number of features, feature selection algorithm, classifier, and dataset), changing the split seed for each repetition and keeping the same folds (samples) for all methods. The average classification accuracy on the test set over all the 50 repetitions is then computed. Similar experimental designs have been applied in [13,30,55]. Feature selection algorithms have been applied only on the training set to avoid selection bias. The statistical significance of the results has been assessed by computing the Student t-test for each set of repetitions. In the reported tables, statistically relevant values with respect to us (i.e., p-value $\leqslant 0.05$) are followed by a * sign, while best absolute values for each row are in bold.

All experiments have been performed by using small sets of features (from 2 to 20) to focus on the capability of the selected features to improve the classification performance. Using large sets of features allows the classifier to compensate for possible feature selection shortcomings by automatically pruning or giving low weights to the least relevant features.

4.2. Classification accuracy

We computed the classification accuracy provided by the MaskedPainter approach and by the six feature selection techniques reported in Section 4.1 with different cardinalities of the selected feature set. The experiments have been performed on all datasets in Table 1 with the J48 decision tree classifier [50] and the greedy subset search. The decision tree classifier is less capable to compensate for possible

Table 3
Accuracy yielded by the J48 classifier on the Alon dataset

#	MP	IG	TR	SM	MM	GI	SV
2	78.94	74.20*	74.68*	74.84*	75.81*	74.68*	74.68*
4	76.95	73.88*	73.24*	74.01*	75.23*	73.24*	73.24*
6	77.37	73.73*	73.75*	74.01*	75.11*	73.75*	73.75*
8	77.05	73.79*	73.68*	74.38*	74.83*	73.68*	73.68*
10	76.85	73.94*	73.77*	74.15*	74.99*	73.77*	73.77*
12	76.02	74.07*	73.99*	74.22*	74.12*	73.99*	73.99*
14	76.15	73.39*	73.80*	74.24*	74.31*	73.80*	73.80*
16	75.26	73.07*	73.19*	73.75*	74.11	73.19*	73.19*
18	75.65	73.00*	73.05*	73.50*	75.23	73.05*	73.05*
20	75.63	73.13*	73.08*	73.25*	75.29	73.08*	73.08*
avg	76.59	73.62*	73.63*	74.03*	74.90*	73.63*	73.63*
max	78.94	74.20	74.68	74.84	75.81	74.68	74.68
dev	1.03	0.42	0.48	0.43	0.53	0.48	0.48

Table 4
Accuracy yielded by the J48 classifier on the Leukemia dataset

#	MP	IG	TR	SM	MM	GI	SV
2	82.72	81.67	80.00*	77.83*	78.25*	81.89	82.47
4	86.50	84.36*	81.75*	83.72*	82.50*	84.44*	85.78
6	86.69	85.17*	84.06*	85.44*	83.81*	85.42*	85.53
8	86.44	85.53	85.25	85.53	83.78*	86.14	85.06*
10	86.86	85.39*	85.22*	85.89	84.42*	85.75*	84.94*
12	86.83	85.14*	85.14*	85.69*	85.56*	85.56*	85.11*
14	86.72	84.97*	84.92*	85.11*	85.42*	85.25*	85.50*
16	86.58	84.92*	84.89*	85.11*	85.28*	85.11*	85.69
18	86.67	84.69*	84.72*	84.97*	85.03*	84.94*	86.17
20	87.22	84.86*	84.86*	85.03*	85.36*	84.97*	86.44
avg	86.32	84.67*	84.08*	84.43*	83.94*	84.95*	85.27*
max	87.22	85.53	85.25	85.89	85.56	86.14	86.44
dev	1.21	1.05	1.68	2.27	2.11	1.11	1.04

feature selection shortcomings by weighting the most/least relevant features. Hence, it allowed us to focus more effectively on the actual contribution of the feature selection. Different choices for these settings are discussed in the following subsections.

Tables 3, 4 and 5 show the classification accuracy yielded by the 7 feature selection methods on the Alon, Leukemia, and Srbct test sets. The results obtained on the other four datasets are reported in the Supplemental Material. Each row reports the accuracy for a specific cardinality of the selected feature set (reported in Column 1). The average accuracy value, the maximum value and the standard deviation for each method are reported in the last three rows.[1]

The MaskedPainter (MP) approach provides a very good accuracy on all datasets. In particular, on the Alon, Brain1, Brain2, Leukemia, and Welsh datasets, the accuracy on the test set is statistically better than all other feature selection techniques. On the Tumor9 dataset, the MP method shows a performance comparable with the best techniques (IG and SV). Eventually, on the Srbct dataset it is outperformed by the SV technique for larger sets of features (18 and 20). However, its overall average performance is statistically better than all other methods.

[1]The max row never has the * (statistical significance), because the maximum value can be obtained by different methods for different numbers of genes.

Table 5
Accuracy yielded by the J48 classifier on the Srbct dataset

#	MP	IG	TR	SM	MM	GI	SV
2	71.50	65.37*	63.76*	59.51*	62.41*	65.79*	63.63*
4	81.73	75.60*	72.97*	69.23*	69.89*	74.57*	74.00*
6	81.92	78.18*	75.17*	75.06*	72.72*	75.96*	78.05*
8	82.07	78.94*	76.75*	76.63*	75.18*	77.32*	80.61*
10	82.07	79.52*	78.06*	78.21*	77.18*	78.29*	81.02
12	82.09	80.63*	78.99*	78.68*	79.02*	79.64*	80.93*
14	81.48	80.85	79.80*	78.37*	80.75	80.54*	81.23
16	81.07	81.11	80.48	78.10*	81.13	81.20	82.10
18	81.16	81.18	81.01	78.23*	81.71	81.51	82.71*
20	80.61	81.06	81.25	78.28*	82.48*	81.79*	82.78*
avg	80.57	78.24*	76.82*	75.03*	76.25*	77.66*	78.71*
max	82.09	81.18	81.25	78.68	82.48	81.79	82.78
dev	3.06	4.60	5.04	5.85	6.06	4.59	5.60

Table 6
Average accuracy improvement over the second best method on all datasets

features	accuracy improvement
2	+3.08%
4	+2.84%
6	+2.81%
8	+1.87%
10	+1.79%
12	+1.91%
14	+1.79%
16	+1.41%
18	+1.11%
20	+1.08%
average	+2.14%

4.3. Cardinality of the selected feature set

We analyzed the behavior of the MaskedPainter approach when varying the cardinality of the selected feature set. The average improvement of the proposed approach over the second best method is computed across all datasets, separately for cardinalities of the feature set ranging in the interval 2–20. Table 6 shows the obtained results.

The MaskedPainter approach yields the highest improvements for low numbers of selected features, when the quality of the selected features more significantly affects classifier performance. Since the first few selected features typically belong to the minimum gene subset (see Section 4.4), these results highlight the quality of this small subset with respect to the features selected by all other methods. For increasing cardinality of the selected feature set the performance difference decreases, but the MaskedPainter algorithm still yields higher accuracy.

Furthermore, the second best feature selection algorithm is not always the same for all datasets. Hence, our approach can self adapt to the dataset characteristics better than the other methods, whose performance is more affected by the data distribution.

4.4. Minimum gene subset

To evaluate the effectiveness of the minimum gene subset we compared the classification accuracy and execution time of the greedy with the result achieved with a set covering approach. A preliminary study exploiting a set covering based approach on microarray datasets was presented in [5]. To the best of our knowledge, this is the only work that formalizes the minimum subset selection as an optimization problem. In the following, a brief description of this method is introduced.

The set covering approach considers the set of gene masks as a matrix of $N \times M$ bits and performs the following three steps. The first two steps aim at reducing the size of this matrix, to reduce the computational time.

A) *Sample reduction*. Each sample (i.e., column) that contains all 0 or 1 over the N gene masks is removed, because it is uninformative for the searching procedure.
B) *Gene reduction*. Each gene (i.e., row) whose gene mask is a subsequence of another gene mask is removed from the matrix. If two or more genes are characterized by the same gene mask, only the gene with the lowest overlap score is kept in the matrix.

Table 7
Performance of the minimum gene subset selection on all datasets

	Greedy		Set covering	
dataset	#genes	accuracy	#genes	accuracy
Alon	5.09	71.82%	4.68	70.33%
Brain1	6.33	70.23%	5.59	70.11%
Brain2	5.07	57.49%	4.62	56.52%
Leukemia	4.15	87.00%	3.82	85.89%
Srbct	6.51	81.09%	5.95	79.55%
Tumor9	10.11	27.57%	9.08	28.03%
Welsh	1.83	89.00%	1.83	86.06%

C) *Reduced matrix evaluation*. The reduced matrix is evaluated by an optimization procedure that searches the minimum set of rows necessary to cover the binary matrix. Since it is a min-max problem, it can be converted to the following linear programming problem.

$$\min \sum_{i=1}^{N} g_i$$
$$\sum_{i=1}^{N} mask_{ij} \cdot g_i \geqslant 1, j = 1, \ldots, M$$
$$g_i \in \{0, 1\}$$

The branch and bound implementation provided by the Symphony library [34] has been exploited to find the optimum solution.

In Table 7 the average accuracy and the average size[2] of (i) the greedy (Columns 2 and 3) and (ii) the set covering (Columns 5 and 6) minimum subsets are reported for all datasets. Values are averaged over the 50 repetitions of the 4-fold cross validation. The average execution time for one fold, which corresponds to an estimate of the time needed by the final user to perform the feature selection, is also reported for both techniques (greedy in Column 4 and set covering in Column 7).

Independently of the dataset, the greedy minimum subset size is always larger than the set covering size. The greedy approach selects the gene maximizing the number of covered samples at each iteration. The set covering approach, instead, exploits a global optimization procedure to select the minimum number of genes that cover the samples. Hence, the greedy approach may need a larger number of genes to reach the best coverage of the training samples. This larger gene set provides a higher accuracy on most datasets, because it yields a more general model which may be less prone to overfitting. For instance, on the Leukemia dataset the average accuracy is 85.89% for the set covering approach and 87.00% for the greedy approach.

The greedy algorithm is also characterized by a lower execution time with respect to the set covering algorithm. On average, the set covering completed in tens of seconds, whereas the greedy took from 0.1 to 2.6 seconds, considering all datasets.

[2]The average size (i.e., the number of genes) of the minimum subset is not an integer, because it depends on the samples included in the considered fold. To cover different folds (i.e., sample sets), a different number of genes may be needed.

Table 8
Top 20 genes on the Alon dataset (colon cancer) and related references

Rank	Gene ID	Gene Name	References
1	Z50753	GUCA2B	[38,7]
2	H06524	GSN	[6,7]
3	J02854	MYL9	[53,48,28,51,7]
4	K03474	AMH	[46]
5	L07032	PRKCQ	[4]
6	M63391	DES	[53,48,28,51,1,7]
7	M36634	VIP	[43,51,7]
8	R87126	MYH9	[23,51,7]
9	M76378	CSRP1	[53,48,28,51,7]
10	H43887	CFD	[53,48,28,7]
11	M22382	HSPD1	[53,48,28,7]
12	X63629	CDH3	[1,7]
13	H40095	MIF SLC2A11	[1,7]
14	X74295	ITGA7	[28]
15	T71025	MT1G	[7]
16	H77597	MT1G MT1H	[17]
17	J05032	DARS	[51]
18	X86693	SPARCL1	[28,51,7]
19	M26697	NPM1	[1,7]
20	H08393	OVGP1 WDR77	[26,51,1,7]

4.5. Biological discussion

We analyzed the biological information presented in literature for the genes selected by the MaskedPainter technique. In Table 8 we report the first twenty genes selected by our algorithm on the entire Alon dataset, related to colon cancer and commonly used for biological validation [8,15]. Column 4 shows references to published works on colon cancer discussing the genes reported in Column 1. The genes deemed as relevant by the MaskedPainter feature selection technique have been identified and discussed in previous biological studies.

For example, gene Z50753, named GUCA2B, related to uroguanylin precursor, is shown to be relevant in [38]. Lowered levels of the uroguanylin may interfere with renewal and removal of epithelial cells. This could result in the formation of polyps, which can progress to malignant cancers of the colon and rectum [38]. As a second example, the downregulation of H06524, the GSN gene (gelsolin), combined with that of PRKCB1, may concur in decreasing the activation of PKCs involved in phospholipid signalling pathways and inhibit cell proliferation and tumorigenicity [6]. Furthermore, RTPCR experiments could be exploited to deeply validate the biological meaning of selected genes, and other tools, such as David [10] and GSEA [41], could be potentially applied for further biological enrichment.

5. Conclusions

Feature selection is a well-known approach to identify relevant genes for biological investigation (e.g., tumor diseases). Feature selection techniques have proved to be helpful in tumor classification and in identifying the genes related to clinical situations.

In this paper we propose a new method for feature selection on microarray data, the MaskedPainter. It allows (a) defining the minimum set of genes that provides the best coverage of the training samples, and (b) ranking genes by decreasing relevance, thus allowing the user to customize the final size of the

feature set. The MaskedPainter method has been compared with six other feature selection techniques on both binary and multiclass microarray datasets. In most experimental settings it yields the best accuracy, while its computational cost is similar to the other feature selection techniques. The approach has been validated on microarray datasets, but we believe it may be applied to any dataset characterized by noisy and continuously valued features, such as peptide expressions.

On the Alon dataset, the identified relevant genes are consistent with the literature on tumor classification. Hence, the MaskedPainter approach may provide a useful tool both to identify relevant genes for tumor diseases and to improve the classification accuracy of a classifier. The authors will provide the source code or the executable

References

[1] S.M. Alladi, P.S. Santosh, V. Ravi and U.S. Murthy, Colon cancer prediction with genetic profiles using intelligent techniques, *Bioinformation* **3**(3) (2008), 130.

[2] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack and A.J. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, 1999.

[3] D. Apiletti, E. Baralis, G. Bruno and A. Fiori, The painter's feature selection for gene expression data, In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, 2007, pp. 4227–4230.

[4] W.H. Au, K.C. Chan, A.K. Wong and Y. Wang, Attribute clustering for grouping, selection and classification of gene expression data, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2005, pp. 83–101.

[5] E. Baralis, G. Bruno and A. Fiori, Minimum number of genes for microarray feature selection, In *Conference Proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, volume 1, 2008, pp. 5692.

[6] F. Bertucci, S. Salas, S. Eysteries, V. Nasser, P. Finetti, C. Ginestier, E. Charafe-Jauffret, B. Loriod, L. Bachelart and J. Montfort, Gene expression profiling of colon cancer by DNA microarrays and correlation with histoclinical parameters, *Oncogene* **23**(7) (2004), 1377–1391.

[7] J.J. Chen, C.A. Tsai, S.L. Tzeng and C.H. Chen, Gene selection with multiple ordering criteria, *BMC bioinformatics* **8**(1) (2007), 74.

[8] J.L. Chen and B.W. Futscher, Optimal search-based gene subset selection for gene array cancer classification, *IEEE Transactions on Information Technology in Biomedicine* **11**(4) (2007), 398–405.

[9] L. Davies and U. Gather, The identification of multiple outliers, *Journal of the American Statistical Association* (1993), 782–792.

[10] G. Dennis, Jr., B.T. Sherman, D.A. Hosack, J. Yang, W. Gao, H.C. Lane and R.A. Lempicki, David: Database for annotation, visualization and integrated discovery, *Genome Biol* **4**(5) (2003), P3, http://david.abcc. ncifcrf.gov/.

[11] R. Díaz-Uriarte and A. de Andrés, Gene selection and classification of microarray data using random forest, *BMC Bioinformatics* **7**(1) (2006), 3.

[12] C. Ding and H. Peng, Minimum redundancy feature selection from microarray gene expression data, *Journal of Bioinformatics and Computational Biology* **3**(2) (2005), 185–206.

[13] S. Dudoit, J. Fridlyand and T.P. Speed, Comparison of discrimination methods for the classification of tumors using gene expression data, *Journal of the American Statistical Association* **97**(457) (2002), 77–87.

[14] R.J. Fox and M.W. Dimmic, A two-sample Bayesian t-test for microarray data, *BMC Bioinformatics* **7**(1) (2006), 126.

[15] C. Furlanello, M. Serafini, S. Merler and G. Jurman, Entropy-based gene ranking without selection bias for the predictive classification of microarray data, *BMC Bioinformatics* **4**(1) (2003), 54.

[16] C. Furlanello, M. Serafini, S. Merler and G. Jurman, Semisupervised learning for molecular profiling, *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (*TCBB*) **2**(2) (2005), 110–118.

[17] C.P. Giacomini, S.Y. Leung, X. Chen, S.T. Yuen, Y.H. Kim, E. Bair and J.R. Pollack, A gene expression signature of genetic instability in colon cancer, 2005.

[18] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing and M.A. Caligiuri, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, *Science* **286**(5439) (1999), 531.

[19] I. Guyon, J. Weston, S. Barnhill and V. Vapnik, Gene selection for cancer classification using support vector machines, *Machine Learning* **46**(1) (2002), 389–422.

[20] B. Hanczar, M. Courtine, A. Benis, C. Hennegar, K. Clément and J.D. Zucker, Improving classification of microarray data using prototype-based feature selection, *ACM SIGKDD Explorations Newsletter* **5**(2) (2003), 23–30.

[21] J. Hua, W.D. Tembe and E.R. Dougherty, Performance of feature-selection methods in the classification of high-dimension data, *Pattern Recognition* **42**(3) (2009), 409–424.
[22] I.B. Jeffery, D.G. Higgins and A.C. Culhane, Comparison and evaluation of methods for generating differentially expressed gene lists from microarray data, *BMC Bioinformatics* **7**(1) (2006), 359.
[23] W. Jiang, X. Li, S. Rao, L. Wang, L. Du, C. Li, C. Wu, H. Wang, Y. Wang and B. Yang, Constructing disease-specific gene networks using pair-wise relevance metric: application to colon cancer identifies interleukin 8, desmin and enolase 1 as the central elements, *BMC Systems Biology* **2**(1) (2008), 72.
[24] T. Jirapech-Umpai and S. Aitken, Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes, *BMC Bioinformatics* **6**(1) (2005), 148.
[25] T. Juliusdottir, E. Keedwell, D. Corne and A. Narayanan, Two-phase EA/K-NN for feature selection and classification in cancer microarray datasets, In *Computational Intelligence in Bioinformatics and Computational Biology, 2005. CIBCB'05. Proceedings of the 2005 IEEE Symposium on*, 2005, pp. 1–8.
[26] G. Karakiulakis, C. Papanikolaou, S.M. Jankovic, A. Aletras, E. Papakonstantinou, E. Vretou and V. Mirtsou-Fidani, Increased type IV collagen-degrading activity in metastases originating from primary tumors of the human colon, *Invasion and Metastasis* **17**(3) (1997), 158.
[27] M.K. Kerr and G.A. Churchill, Bootstrapping cluster analysis: Assessing the reliability of conclusions from microarray experiments, *Proceedings of the National Academy of Sciences of the United States of America* **98**(16) (2001), 8961.
[28] H. Kishino and P.J. Waddell, Correspondence analysis of genes and tissue types and finding genetic links from microarray data, *Genome Informatics Series*, 2000, pp. 83–95.
[29] Y.Y. Leung, C.Q. Chang, Y.S. Hung and P.C. Fung, Gene selection for brain cancer classification, In *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, 2006.
[30] T. Li, C. Zhang and M. Ogihara, A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression, 2004.
[31] W. Li and Y. Yang, How many genes are needed for a discriminant microarray data analysis, *Methods of Microarray Data Analysis* (2002), 137–150.
[32] T.C. Lin, R.S. Liu, C.Y. Chen, Y.T. Chao and S.Y. Chen, Pattern classification in DNA microarray data of multiple tumor types, *Pattern Recognition* **39**(12) (2006), 2426–2438.
[33] X. Liu, A. Krishnan and A. Mondry, An entropy-based gene selection method for cancer classification using microarray data, *BMC Bioinformatics* **6**(1) (2005), 76.
[34] T. Ralphs and M. Guzelsoy, The SYMPHONY callable library for mixed integer programming, *The Next Wave in Computing, Optimization and Decision Technologies* **29** (2006), 61–76, Software available at http://http://www.coin-or.org/SYMPHONY.
[35] R. Ruiz, J.C. Riquelme and J.S. Aguilar-Ruiz, Incremental wrapper-based gene selection from microarray data for cancer classification, *Pattern Recognition* **39**(12) (2006), 2383–2392.
[36] Y. Saeys, I. Inza and P. Larranaga, A review of feature selection techniques in bioinformatics, *Bioinformatics* **23**(19) (2007), 2507.
[37] S. Shah and A. Kusiak, Cancer gene search with data-mining and genetic algorithms, *Computers in Biology and Medicine* **37**(2) (2007), 251–261.
[38] K. Shailubhai, H.H. Yu, K. Karunanandaa, J.Y. Wang, S.L. Eber, Y. Wang, N.S. Joo, H.D. Kim, B.W. Miedema and S.Z. Abbas, Uroguanylin treatment suppresses polyp formation in the Apc Min/+ mouse and induces apoptosis in human colon adenocarcinoma cells via cyclic GMP, *Cancer Research* **60**(18) (2000), 5151–5157.
[39] A. Statnikov, C.F. Aliferis, I. Tsamardinos, D. Hardin and S. Levy, A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis, *Bioinformatics* **21**(5) (2005), 631–643.
[40] Y. Su, T.M. Murali, V. Pavlovic, M. Schaffer and S. Kasif, RankGene: identification of diagnostic genes based on expression data, 2003.
[41] A. Subramanian, H. Kuehn, J. Gould, P. Tamayo and J.P. Mesirov, Gsea-p: a desktop application for gene set enrichment analysis, *Bioinformatics* **23**(23) (2007), 3251, http://www.broadinstitute.org/gsea/.
[42] Y. Sun, Iterative RELIEF for feature weighting: Algorithms, theories and applications, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **29**(6) (2007), 1035–1051.
[43] Q. Tao, J. Ren and J. Li, Vasoactive intestinal peptide inhibits adhesion molecule expression in activated human colon serosal fibroblasts by preventing NF-κB activation, *Journal of Surgical Research* **140**(1) (2007), 84–89.
[44] J.G. Thomas, J.M. Olson, S.J. Tapscott and L.P. Zhao, An efficient and robust statistical modeling approach to discover differentially expressed genes using genomic expression profiles, 2001.
[45] L. Wang, F. Chu and W. Xie, Accurate cancer classification using expressions of very few genes, *IEEE ACM Transactions on Computational Biology and Bioinformatics* **4**(1) (2007), 40.
[46] S. Wang, H. Chen and S. Li, Gene selection using neighborhood rough set from gene expression profiles, In *Proceedings of the 2007 International Conference on Computational Intelligence and Security*, IEEE Computer Society Washington, DC, USA, 2007, pp. 959–963.

[47] J.B. Welsh, L.M. Sapinoso, A.I. Su, S.G. Kern, J. Wang-Rodriguez, C.A. Moskaluk, H.F. Frierson and G.M. Hampton, Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer, 2001.
[48] L. Wessels, M. Reinders, T. van Welsem and P. Nederlof, Representation and classification for high-throughput data, In *Proceedings of SPIE*, volume 4626, 2002, pp. 226.
[49] M. West, C. Blanchette, H. Dressman, E. Huang, S. Ishida, R. Spang, H. Zuzan, J.A. Olson Jr, J.R. Marks and J.R. Nevins, Predicting the clinical status of human breast cancer by using gene expression profiles, *Proceedings of the National Academy of Sciences* **98**(20) (2001), 11462.
[50] I.H. Witten and E. Frank, Data mining: Practical machine learning tools and techniques, 2005.
[51] M. Xiong, X. Fang and J. Zhao, Biomarker identification by feature wrappers, 2001.
[52] Y.H. Yang, S. Dudoit, P. Luu, D.M. Lin, V. Peng, J. Ngai and T.P. Speed, Normalization for cDNA microarray data: A robust composite method addressing single and multiple slide systematic variation, *Nucleic Acids Research* **30**(4) (2002), e15.
[53] Y.L. Yap, X.W. Zhang, M.T. Ling, X.H. Wang, Y.C. Wong and A. Danchin, Classification between normal and tumor tissues based on the pair-wise gene expression ratio, *BMC Cancer* **4**(1) (2004), 72.
[54] L. Yu, Feature selection for genomic data analysis, *Computational Methods of Feature Selection*, 2007, pp. 337.
[55] X. Zhou and D.P. Tuck, MSVM-RFE: Extensions of SVM-RFE for multiclass gene selection on DNA microarray data, *Bioinformatics* **23**(9) (2007), 1106.