Feature Set Enhancement via Hierarchical Clustering for Microarray Classification

Mattia Bosio*, Pau Bellot Pujalte, Philippe Salenbier Clairon, Albert Oliveras i Verges
*Department of Signal Theory and Communications (TSC)
Universitat Politecnica de Catalunya, 08034 Barcelona, Spain
Email: mattia.bosio@tsc.upc.edu

Abstract—A new method for gene expression data classification is proposed. At first, the original feature set is enriched by including new features, (i.e. metagenes), produced via hierarchical clustering. Then, a reliable classifier is built from a wrapper feature selection process. Inside the process, a new two-level feature ranking criterion based on error rate and a reliability measure is introduced. The expected output is a classifier with good predictive ability, using as few features as possible to reduce the risk of overfitting. This method has been tested on three public datasets: leukemia, lymphoma and colon. The proposed method has obtained interesting classification results, it has confirmed the utility of both metagenes and feature ranking criterion to improve the final classifier.

Keywords—Hierarchical clustering; feature selection; microarray classification;

I. INTRODUCTION

Microarrays are a powerful high-throughput technology which is often used for classification purposes because it allows the simultaneous measurement of thousands of gene expression values. A microarray dataset is typically composed of only tens of observations due to the cost of the experiments. This characteristic of sample scarcity makes necessary a feature selection process to produce reliable classifiers [1].

The aim of this work is to propose a method for microarray classification able to reach small prediction error and using as few features as possible to reduce the risk of overfitting. For this purpose, a two-step process is adopted. At first, the original feature set of gene expression values is enriched with new features that are linear combinations of the original genes. These new features are called metagenes and are produced by hierarchical clustering. In the second step, a wrapper feature selection process [2] is performed; its aim is to find a reduced set of features with which build a classifier.

This paper is organized as follows: in Section 2 the metagenes creation through hierarchical clustering is explained, in Section 3 the feature selection algorithm is detailed, Section 4 contains the adopted experimental protocol, results from experiments are presented in Section 5, and conclusions are included in Section 6.

II. METAGENES CREATION

In this section is explained how the original feature set is expanded by the introduction of metagenes. The clustering process, here, is not used to find a structure for the original feature set, but to generate a new set of features, each of which resumes in itself a cluster of genes. The objective is to join genes with common characteristics into a metagene filtering out noise thanks to a “low-pass” effect.

Many different hierarchical clustering techniques exist, depending on the chosen similarity measure and the metagene generation rule (i.e. how the linear combination of the elements in a metagene is defined). The chosen approach is a bottom up, pairwise hierarchical clustering; a pseudo code is showed in Figure 1.

It is an iterative algorithm joining two features at a time, genes or metagenes, in a cluster. Each cluster is represented by a metagene as a linear combination of the features in it. The created metagene, $m_i$, is then added to the active feature set and used as a feature in subsequent iterations. At the end of the process, the initial feature set of $p$ genes is expanded with $p-1$ metagenes. Key points in the metagene creation are the similarity metric: $d(\cdot, \cdot)$, and the generation rule: $g(\cdot, \cdot)$. Changing either one of those two functions implies the creation of a different metagene set.

Two different metagene generation methods have been applied in this work. The first technique is based on Lee’s work [3], where an adaptive method for multi-scale representation and eigenanalysis of data called Treelets is presented. With Treelets, a clustering tree is built, in which at each level of the tree, the two most similar features are grouped and replaced by a coarse-grained approximation feature and a residual detail feature. The similarity measure in Treelets is correlation $d(f_a, f_b) = \langle f_a, f_b \rangle / (\|f_a\| \cdot \|f_b\|)$, while the two replacing features, approximation and detail, are obtained through a local Principal Component Analysis (i.e. PCA) on two dimensions: the approximation is the first local principal component and the detail is the second. Here, Treelets has been applied on microarray data and the approximation feature in each level has been selected as metagene. The approximation feature has been chosen because a metagene is a synthesis of f its components that should filter out the noise; this is more likely to be obtained with the approximation feature than with the detail feature.

The second technique, called Euclidean clustering adopts a similar iterative procedure but using negative Euclidean distance $d(f_a, f_b) = -\|f_a - f_b\|_2$, as similarity measure, so that the maximum is zero when two features are equal. Euclidean distance has been selected because it measures the punctual closeness rather than the profile shape similarity. This
Original feature set \( F_0 = \{ f_1, \ldots, f_p \} \)
Active feature set \( F \equiv F_0 \)
Metagenec set \( M \equiv \emptyset \)

For \( i = 1 : p-1 \)

1) Calculate pairwise similarity metric \( d(f_2, f_3) \) for all features in \( F_2 \).

2) Find \( a,b : d(f_2, f_3) = \max(d(\cdot, \cdot)) \)

3) New metagene \( m_i = g(f_2, f_3) \) generation:
   \[ m_i = \sum_{i=1}^{p} \alpha_i \cdot f_i \] with \( \alpha \in \mathbb{R}^p \)
   Each metagene is a linear combination of all original features \( f_i \).

4) \( F := F \cup \{ m_i \} : \text{join metagene to active feature set} \)

5) \( F := F \setminus \{ f_2, f_3 \} : \text{remove metagene originating features from active feature set} \)

6) \( M := M \cup \{ m_i \} : \text{join metagene to metagene set} \)

Define new expanded feature set: \( F = F_0 \cup M \) as the union of metagenes and original gene expression profiles.

Fig. 1. General clustering algorithm.

choice implies a modification of the whole clustering process because choosing the first PCA component as metagene implies an increase in its dynamic as the represented cluster grows. An illustrative example comes from Figure 2, in which all features are equal and the clustering process would produce metagenes that are scaled versions of the original feature, with scale factor growing with the number of clustered elements. This phenomenon is irrelevant in the Treelets case, as scaling does not affect correlation, but it is important when Euclidean distance is considered and must be taken care of. To do so, when a metagene \( m_\alpha \) is created, two versions of it are used, one is the same as in the Treelets case, and the second one is a scaled version: \( m_{\alpha, \text{scaled}} = m_\alpha / \| \alpha \|_1 \). The scaled version \( m_{\alpha, \text{scaled}} \) preserves the dynamic range of the original features, thus it is used to calculate the pairwise distance and it will be used as a metagene; the original version, instead, is used when a new metagene is built from \( m_\alpha \) to preserve the energy distribution among the basic components as in Figure 2.

III. Feature selection process

In this section, the adopted feature selection process is presented. The objective is to select a feature subset as small as possible to build a good classifier. Among possible alternatives in the literature, evolutionary algorithms have obtained good results [4], but in [5] is underlined how their performances would decrease when the feature set dimension grows. To avoid those problems, a deterministic approach it has been chosen, which tries to catch the advantages of an evolutionary search including the mutation possibility of previous choices. In Figure 3 the algorithm flowchart is illustrated, it is a modifi-

Feature set \( F_2 = \{ f_1, f_2, f_3 \} \) with \( f_1 = f_2 = f_3 \)
Two metagenes are created

1) metagene \( m_1 \) joining \( f_1 \) and \( f_2 \)
   \[ m_1 = 0.7071 \cdot f_1 + 0.7071 \cdot f_2 \]
   \[ m_{1, \text{scaled}} = 1/2 \cdot f_1 + 1/2 \cdot f_2 \]

2) metagene \( m_2 \) joining \( f_1 \) and \( f_3 \)
   \[ m_2 = 0.5773 \cdot f_1 + 0.5773 \cdot f_2 + 0.5773 \cdot f_3 \]
   \[ m_{2, \text{scaled}} = 1/3 \cdot f_1 + 1/3 \cdot f_2 + 1/3 \cdot f_3 \]

Scaled versions \( m_{1, \text{scaled}} \) and \( m_{2, \text{scaled}} \) are used for Euclidean clustering because they preserve the components dynamics. The scaled versions will expand the feature set.

Non scaled versions \( m_1 \) and \( m_2 \) are used in the construction phase with PCA as they preserve the energy distribution among components.

Fig. 2. Example of metagene creation with Euclidean clustering.

cation of the Sequential Floating Forward Selection algorithm (SFFS) [6] with the introduction of a replacing step when backtracking does not work. It is called Improved sequential Floating Forward Selection (IFFS) [5]. The search process starts with an empty set and ends when a threshold value is reached. The threshold is either the maximum accepted number of features or a maximum number of iterations in case the algorithm has entered in an infinite loop. After the initialization, the selection process enters in a loop of tasks. At first there is the add phase, where all those features not yet selected are added to the current set one at a time. For each one, a classifier is trained and the correspondent classification score \( J(\cdot) \) is calculated. The feature obtaining the best \( J(\cdot) \) is added to the current set.

A. Feature selection algorithm

After the initialization, the selection process enters in a loop of tasks. At first there is the add phase, where all those features not yet selected are added to the current set one at a time. For each one, a classifier is trained and the corresponding classification score \( J(\cdot) \) is calculated. The feature obtaining the best \( J(\cdot) \) is added to the current set. Then, if the threshold has not been reached yet, the algorithm starts a backtracking phase. In this step the weakest feature in the subset (i.e. the feature whose elimination implies the minimum performance loss or the maximum performance gain) can be eliminated. If the elimination improves \( J(\cdot) \), the weak feature is removed and a new backtracking phase is performed. Otherwise, the algorithm looks for substituting one feature in the replacing phase. For each feature in the current set, a substitute is chosen via an analysis like in the add phase. If the best substitution has proven useful, (i.e. the \( J(\cdot) \) value with the substitution is better than without), the current set is updated and a new backtracking phase takes place. Otherwise, the algorithm goes back to the add phase.
Fig. 3. IFFS feature selection algorithm.

B. Feature ranking criterion

The search algorithm is a wrapper feature selection process, so the classifier is applied inside the selection phase. The Linear Discriminant Analysis (LDA) classifier has been used in this study for its simplicity, as recommended to obtain classifiers more robust to overfitting [7]. In Section 3.1 how the criterion \( J(\cdot) \) enters in all parts of the search process can be appreciated; its task is to rank features and decide which is the best. \( J(\cdot) \) is a measure of how good a classifier is to predict new samples. To obtain a reliable value estimation, a 10-fold cross validation process is applied and the \( J(\cdot) \) value is calculated on the test set. Due to the microarray data characteristic of few samples and many dimensions, a \( J(\cdot) \) rule based only on prediction error may not be enough in ranking all the features. It is common to have a group of features with the same prediction error from which only one feature cab be selected. To overcome this limitation, a two level criterion is introduced. In the first level, features are sorted for the selected. To overcome this limitation, a two level criterion is introduced. In the first level, features are sorted for the selected. To overcome this limitation, a two level criterion is introduced. In the first level, features are sorted for the selected. To overcome this limitation, a two level criterion is introduced. In the first level, features are sorted for the selected. To overcome this limitation, a two level criterion is introduced. In the first level, features are sorted for the selected. To overcome this limitation, a two level criterion is introduced. In the first level, features are sorted for the selected. To overcome this limitation, a two level criterion is introduced. In the first level, features are sorted for the selected.

In the second level, a reliability parameter \( r \) is calculated to quantify the estimation goodness as a weighted sum of sample distances from the classification boundary. It is calculated on the test set samples and the final reliability value is the mean of the cross validation iterations. In (1), reliability is calculated inside a cross validation iteration for a two-class problem, which is the one treated in this study. In (1), \( n_{\text{test}} \) is the test set dimension, \( c_l \) is the class of sample \( l \), (it can be 1 or 2) and \( p(c_l) \) is the probability of class \( c_l \) in the test set. The value \( d_l \) is the Euclidean distance of sample \( l \) from the classifier boundary with positive sign in case of correct classification or negative sign otherwise. Finally, \( \tilde{\sigma}_d \) is an estimation of intra class variance of the sample distances from the classification boundary, it is estimated using all training and test samples, \( n_1 \) and \( n_2 \) are the number of samples in class 1 and 2 respectively, to have a more reliable estimation. It is obtained like in the independent two-sample t-test with classes of different size and variance, as it is the most general case for a two-class problem. In detail \( \tilde{\sigma}_1 \) and \( \tilde{\sigma}_2 \) are the estimated variances of sample distance from boundary for all samples of class 1 and 2 respectively.

\[
r = \frac{1}{n_{\text{test}}} \sum_{l=1}^{n_{\text{test}}} \frac{d_l}{p(c_l) \cdot \tilde{\sigma}_d}
\]

where \( \tilde{\sigma}_d = \sqrt{\frac{\tilde{\sigma}_1^2 + \tilde{\sigma}_2^2}{n_1}} \)

Dividing for \( \tilde{\sigma}_d \) makes that scaled versions of the same feature obtain the same reliability value. Dividing for \( (c_1) \), instead, is useful when there are data with almost constant values. In that case, if the test set distribution is not uniform between the two classes, reliability could grow up at wish if the classifier boundary was set sufficiently far from the samples, with an error rate of \( \min(p(c_1), p(c_2)) \). Correcting for \( p(c_l) \) assigns to each class the same relative weight, thus making reliability in the previous case close to zero. Reliability value, \( r \in [\infty, \infty] \), is positively influenced by mean class separation in the perpendicular direction to the classifier boundary, and by small intra class data variance. It is penalized by a factor proportional to error intensity so that greater errors produce greater penalties, allowing discrimination among features with equal error rates.

The final \( J(\cdot) \) value is then composed of the mean error rate value and the mean reliability parameter along the ten iterations of the cross validation process. A feature is ranked to be better than another if its error rate is lower or if its reliability value is higher, in the case they have the same error rate value. This is the ranking criterion used in all phases of the selection process.

IV. EXPERIMENTAL PROTOCOL

In this section the experimental protocol to evaluate the proposed method is presented. Three public, two-class, microarray datasets have been analyzed: Leukemia, Lymphoma and Colon. The Leukemia dataset is a collection of gene expression measurements from 72 Leukemia samples (47 acute lymphoblastic Leukemia (ALL) and 25 acute myeloblastic Leukemia (AML)) as reported in [8]. Each sample contains 7129 probe set values, reduced to 3859 after filtering. The Lymphoma dataset is a collection 96 normal and malignant lymphocyte samples [9], it contains 42 samples of diffused large B-cell Lymphoma (DLBCL) and 54 samples of other types each one including 4026 genes. The Colon dataset is a collection of 62 expression measurements from Colon biopsy samples reported in [10]. It contains 22 normal and 40 Colon cancer samples, each one with 2000 gene expression values.
All dataset pass through a preprocessing phase before the metagen creation. The preprocessing consists in applying a base two, logarithmic transformation to the original data, followed by a mean removal so that all genes have zero mean across the samples. For the Leukemia dataset it has been necessary to include a filtering operation before the preprocessing because there are negative values in the original data, making the logarithmic transformation impossible. The filtering process is the same as in [4], that led to a feature number reduction from 7129 to 3859 genes. It is composed of a threshold operation, in which the minimum value is set to 20 and the maximum to 16000 for all the original data, followed by the exclusion of all genes with a fold change smaller than 5 or a dynamic range smaller than 500.

The feature selection algorithm has been applied on the expanded feature sets produced by Treelets and Euclidean clustering. It has also been applied on the original data only to evaluate the possible classification benefit from the use of metagenes. The experimental results include the error rates obtained applying the best LDA classifier in each case. Error rate is estimated both with 10-fold cross validation, to have a comparable value with results in the literature, and with bolstered resubstitution [11], because it has showed to be the best error estimator for LDA classifier in a small sample context like microarray classification [11]. Estimation via bolstered resubstitution is also included to give a more reliable estimation of the classifier generalization properties.

V. RESULTS

The collected experimental results are presented in Table I. Columns in Table I contains the error rates for the classification of three different datasets. In the first three rows, the obtained error rate estimations with cross validation (10 CV) and bolstered resubstitution (bResub) for the proposed method are presented. In the last rows, results from the literature are included as reference: error rates have all been estimated with cross validation. The number of used features to classify is reported in braces \{-\} when available.

The proposed method is able to reach 0\% error rate for each one of the analyzed datasets if CV is utilized, and obtains very low error rates even with bolstered resubstitution estimator, confirming the reliability of the produced classifier. Furthermore, the introduction of metagenes has proved beneficial for classification: in all cases a classifier including metagenes is better than with original data only. For Colon and Lymphoma datasets the best option is the Euclidean clustering while it is the Treelets clustering for the Leukemia dataset.

Comparing the obtained results with the state of the art, can be observed how the proposed method produces better classifiers for Lymphoma database and Colon database, where it reaches 0\% error rate with fewer features. On the contrary, for Leukemia dataset, the NSGAA II [4] is the best choice as it obtains 0\% error rate with 4 genes, while at least 8 are needed with Treelets to reach the same error rate.

VI. CONCLUSION

It has been developed a microarray classification method able to reach a 0\% classification error in all the considered datasets. Key points are the feature selection process using IFFS with a two-level ranking criterion, introducing the reliability parameter, and the feature set enhancement including metagenes from hierarchical clustering. The feature set expansion with metagenes has proven beneficial for classification since it made possible to obtain 0\% error rates with fewer features than using original features only.

This method has obtained the best classifier for two out of three databases when compared to state of the art alternatives with cross validation error estimation.

Analyzing error estimation with bolstered resubstitution can be observed how the proposed method still obtains very low error rates, thus underlying the good generalization properties of the obtained classifiers that make the proposed method an interesting technique for microarray classification.

REFERENCES