Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data

Fernando Díaz[1], Florentino Fdez-Riverola[2], Daniel Glez-Peña[2], and Juan M. Corchado[3]

[1] University of Valladolid, Plaza Santa Eulalia, 9-11, 40005, Segovia, Spain
fdiaz@infor.uva.es
[2] University of Vigo, Campus Universitario As Lagoas s/n, 32004, Ourense, Spain
{riverola, dgpena}@uvigo.es
[3] University of Salamanca, Plaza de la Merced s/n, 37008, Salamanca, Spain
corchado@usal.es

Abstract. The advent of DNA microarray technology has supplied a large volume of data to many fields like machine learning and data mining. Intelligent support is essential for managing and interpreting this great amount of information. One of the well-known constraints specifically related to microarray data is the large number of genes in comparison with the small number of available experiments. In this context, the ability of design methods capable of overcoming current limitations of state-of-the-art algorithms is crucial to the development of successful applications. In this paper we demonstrate how a supervised fuzzy pattern algorithm can be used to perform DNA microarray data reduction over real data. The benefits of our method can be employed to find biologically significant insights relating to meaningful genes in order to improve previous successful techniques. Experimental results on acute myeloid leukemia diagnosis show the effectiveness of the proposed approach.

1 Introduction

Microarrays are one of the latest high-throughput technologies in experimental molecular biology, which allow monitoring of gene expression for tens of thousands of genes in parallel and are already producing huge amounts of valuable data. Analysis and handling of such data is becoming one of the major bottlenecks in the utilization of this technology.

One of the major uses of DNA microarray experiments is to attempt to infer meaningful relationships among genes. Up to now, the analysis of DNA microarray data has been divided into four main interdependent branches: (*i*) gene identification, gene selection or gene reduction, (*ii*) clustering or class discovery, (*iii*) classification or class prediction and (*iv*) biological discovery. Nevertheless, there are two other parallel research areas in DNA microarray analysis: (*v*) graphical modeling, that allows the rapid interactive exploration of gene relationships and (*vi*) low-level analysis focused on providing better readouts and solving the expression level summarization problem. In addition, the characteristics of the data gathered from DNA microarray experiments determine which machine learning methods will apply, and can drive the extension of existing algorithms.

E. Corchado et al. (Eds.): IDEAL 2006, LNCS 4224, pp. 1087 – 1094, 2006.

The systematic classification of types of tumors is crucial to achieve advances in cancer treatment and research. Although clustering is a popular way of analyzing data, input space reduction is often the key phase in the building of an accurate classifier [1]. We propose the use of a fuzzy prototype-based method able to perform gene selection. In this case, the goal is the identification of a simplified expression profile that can be used to identity relevant genes representing each class of cancer.

This paper describes our initial research in developing a sound method to perform gene selection over real data. Section 2 gives an overview of related work, section 3 explains the proposed algorithm, section 4 introduces the experimental test bed carried out, finally section 5 gives out the results and concluding remarks.

2 Previous Related Work

Classical gene selection methods tend to identify differentially expressed genes from a set of microarray experiments [2]. These genes are expected to be up- or down-regulated between healthy and diseased tissues or between different classes. A differentially expressed gene is a gene which has the same expression pattern for all samples of the same class, but different for samples belonging to different classes. The relevance value of a gene depends on its ability to be differentially expressed. However, a non-differentially expressed gene will be considered irrelevant and will be removed from a classification process even though it might well contain information that would improve classification accuracy. One way or another, the selected method has to pursue two main goals: (*i*) reduce the cost and complexity of the classifier and (*ii*) improve the accuracy of the model.

These methods rank genes depending on their relevance for discrimination. Then by setting a threshold, one can filter the less relevant genes among those considered. As such, these filtering methods may be seen as particular gene selection methods. An important task in microarray data analysis is therefore to identify genes, which are differentially expressed in this way. Statistical analysis of gene expression data relating to complex diseases is of course not really expected to yield accurate results. A realistic goal is to narrow the field for further analysis, to give geneticists a short-list of genes for analysis into which hard-won funds are worth investing.

The area of gene identification has been addressed by [3] through the utilization of information theory. Several methods have been proposed to reduce dimensions in the microarray data domain. These works include the application of genetic algorithms [4], wrapper approaches [5], support vector machines [6, 7], spectral biclustering [8] etc. Other approaches focus their attention on redundancy reduction and feature extraction [9, 10], as well as the identification of similar gene classes making prototypes-genes [11].

3 Gene Selection Using Fuzzy Patterns

This work proposes a method for selecting genes which is based on the notion of fuzzy pattern (see [12, 13] for more details). Briefly, given a set of microarrays which are well classified, for each class it can be constructed a fuzzy pattern (FP) from the

fuzzy microarray descriptor (FMD) associated to each one of the microarrays. The FMD is a comprehensible description for each gene in terms of one from the following linguistic labels: LOW, MEDIUM and HIGH. Therefore, the fuzzy pattern is a prototype of the FMDs belonging to the same class where the membership criterion of each gene to the fuzzy pattern of the class is frequency-based. Obviously, this fact can be of interest, if the set of initial observations are labeled with the same kind of cancer. The pattern's quality of fuzziness is given by the fact that the selected labels come from the linguistic labels defined during the transformation into FMD of an initial observation. Moreover, if a specific label of one feature is very common in all the examples belonging to the same class, this feature is selected to be included in the pattern .

```
procedure DiscriminantFuzzyPatterns (input: ListFP; output: ListDFP)
{
00  begin
01    initialize_DFP: FP ← ∅
02    for each fuzzy pattern FPi ∈ ListFP do
03      Initialize_DFP: DFPi ← ∅
04      for each fuzzy pattern FPj ∈ ListFP and FPi <>FPj do
05        for each gen g ∈GetGenes(FPi) do
06          if  (g ∈ GetGenes(FPj)) AND
                (GetLabel(FPi, g) <> GetLabel(FPj, g)) then
07              AddMember(DFPi, Member(FPi, g))
08      Add_to_List_of_DFP: Add(ListDFP, DFPi)
09  end.
}
```

Fig. 1. Proposed algorithm for selecting genes

3.1 Gene Selection Strategy

The goal of gene selection in this work is to determine a reduced set of genes, which are useful to classify new cases within one of the known classes. For each class it is possible to compute a fuzzy pattern from the available data. Since each pattern is representative of a collection of microarrays belonging to the same class, we can assume that the genes included in a pattern, are significant to the classification of any novel case within the class associated with that pattern. Now we are interested in those genes that allow us to discriminate the new case from one class with regard to the others. Here we introduce the notion of discriminant fuzzy pattern (DFP) with regard to a collection of fuzzy patterns. A DFP version of a FP only includes those genes that can serve to differentiate it from the rest of the patterns. The algorithm used to compute the DFP version of each FP in a collection of fuzzy patterns is shown in Figure 1.

As can be observed from the algorithm, the computed DFP for a specific FP is different depending on what other FPs are compared with it. It's not surprising that the genes used to discern a specific class from others (by mean of its DFP) will be different if the set of rival classes also changes.

4 Case Study: Acute Myeloid Leukemia

Acute myeloid leukemia (AML) is a heterogeneous group of hematological cancers with marked differences in their response to chemotherapy. As in many other human cancers, the diagnosis and classification of AML have been based on morphological, cytochemical and immunophenotipic features. More recently, genetic features have helped to define biologically homogeneous entities within AML as the Acute Promyelocytic Leukemia (APL). The correlation between morphologic characteristics, genetic abnormalities and prognostic features is very consistent within the APL group, whereas is more inconsistent in the remaining AML.

Bone marrow (BM) samples from 62 adult patients with newly de novo diagnosed AML were analyzed. All samples contained more than 80% blast cells. The median age was 36 years (range 14-70 years). Patients were classified according to the WHO classification into 4 subgroups: a) 10 APL with t(15;17), b) 4 AML with inv(16), c) 7 acute monocytic leukemias and d) 41 non-monocytic AML without recurrent cytogenetic translocations. Each case (microarray experiment) stores 22,283 ESTs corresponding to the expression level of thousands of genes. The data consisted of 1,381,546 scanned intensities.

The goal of this study is to characterize the Acute Promyelocytic Leukemia (APL) from the non-APL leukemias in terms of the genetic expression profile. As an additional requirement of this study, the number of selected genes must be the minimum (preserving the accuracy of a binary classifier).

4.1 Methodology

We are interested in determining a list of significant genes following the method described in section 3.1. Firstly, the selected genes can vary widely with the values of the parameters Θ and Π, which must be set up in order to compute the fuzzy patterns. Several configurations of these parameters have been tested. After some initial experiments, the tested values of parameters Θ and Π are the nine configurations of the Cartesian product $\{0.7, 0.8, 0.9\}\times\{0.55, 0.60, 0.65\}$. Each configuration has been used to select significant genes from the whole data set of microarrays. This is the first experiment carried out, herein referred to as EXP#1.

From a different point of view, the selected genes can be sensible to the specific microarrays from they are selected, namely the data sets used to select genes. Therefore, a second experiment is considered, herein referred to as EXP#2. It has been split the original data set in four chunks, following a stratified 4-fold cross validation strategy and then, the nine configurations have been tested.

In order to summarize the results of the tests, for each experiment (EXP#1 and EXP#2), a collection of three lists have been constructed (one list by each one value of parameter Π). Inside each list, the selected genes are ordered by the frequency of

appearance of this gene in the tests carried out with the same Π parameter, but different Θ parameter. Namely, fixed the value of Π, a gene, which appears in the three tests (corresponding to the three possible values of Θ), appears before in the list that other gene which only appears in one test (of the three possible values).

Finally, in order to validate the obtained results we perform two different comparisons. Firstly, the selected genes by the proposed method are compared with the genes selected with the PAM software [14]. Secondly, a classifier is constructed from the data resulting of the projection of the original data within the selected features, and its accuracy is evaluated over the 4 test sets of the 4-fold cross validation. The selected classifier is growing cell structure (GCS) network [15]. Although this ANN is especially suitable for unsupervised learning, its choice is motivated by its use in current work about the same problem in other research tasks. To perform a classification task, the GCS simply responds with the majority class of the node that fires the new case. More detailed information about this network can be found in [12, 13].

Table 1. Selected genes in experiment EXP#1

Π = *0.55 (filter1.1)*		
μA Probeset	***Gene***	***tests***
209960_at	--	XOH
210755_at	--	XOH
210997_at	HGF	XOH
220010_at	KCNE1L	XOH
209560_s_at	DLK1	XOH
203074_at	ANXA8	XOH
207781_s_at	ZNF6	XOH
208894_at	HLA-DRA	XOH
212187_x_at	PTGDS	XOH
222317_at	PDE3B	XOH
209686_at	S100B	XOH
211748_x_at	PTGDS	XOH
212013_at	Q92626_HUMAN	XOH
213385_at	CHN2	XOH
207996_s_at	CS001_HUMAN	XOH
209815_at	PTCH	XOH
213355_at	SIA10_HUMAN	XOH
212012_at	Q92626_HUMAN	XOH
219090_at	SLC24A3	XOH
210998_s_at	HGF	XOH
211474_s_at	SERPINB6	XOH
212590_at	RRAS2	XOH
212590_at	FGF13	-OH

Π = *0.60 (filter1.2)*		
μA Probeset	***Gene***	***tests***
209960_at	--	XOH
210755_at	--	XOH
210997_at	HGF	XOH
220010_at	KCNE1L	XOH
209560_s_at	DLK1	XOH
203074_at	ANXA8	-OH
205110_s_at	FGF13	-OH

Π = *0.65 (filter1.3)*		
μA Probeset	***Gene***	***tests***
209960_at	--	XOH
210755_at	--	XOH
210997_at	HGF	XOH
220010_at	KCNE1L	XOH

Table 2. Accuracy of the GCS network trained with selected genes in EXP#1

Filter	***Set***	***Mean***	***Std. Err.***
filter1.1	training	0.00%	0.00%
	test	0.00%	0.00%
filter1.2	training	0.53%	0.46%
	test	0.00%	0.00%
filter1.3	training	1.62%	0.89%
	test	1.47%	1.27%

Table 3. Selected genes in experiment EXP#2

Π = *0.55 (filter2.1)*		
μA Probeset	*Gene*	*tests*
220010_at	KCNE1L	12/12
210997_at	HGF	12/12
210755_at	--	12/12
209960_at	--	12/12
207996_s_at	CS001_HUMAN	11/12
203074_at	ANXA8	11/12
209560_s_at	DLK1	11/12
211748_x_at	PTGDS	11/12
213355_at	SIA10_HUMAN	10/12
207781_s_at	ZNF6	11/12
212912_at	RPS6KA2	9/12
209686_at	S100B	10/12
220570_at	RETN	9/12
211474_s_at	SERPINB6	11/12
209815_at	PTCH	10/12
205110_s_at	FGF13	8/12
212187_x_at	PTGDS	10/12
208894_at	HLA-DRA	10/12
222317_at	PDE3B	8/12
219090_at	SLC24A3	10/12
213385_at	CHN2	10/12
214617_at	PRF1	7/12

Π = *0.60 (filter2.2)*		
μA Probeset	*Gene*	*tests*
220010_at	KCNE1L	12/12
210997_at	HGF	12/12
210755_at	--	12/12
209960_at	--	12/12
203074_at	ANXA8	11/12
205110_s_at	FGF13	8/12
212187_x_at	PTGDS	10/12

Π = *0.65 (filter2.3)*		
μA Probeset	*Gene*	*tests*
220010_at	KCNE1L	11/12
210997_at	HGF	10/12
210755_at	--	9/12
209960_at	--	9/12

Table 4. Accuracy of the GCS network trained with selected genes in EXP#2

Filter	*Set*	*Mean*	*Std. Err.*
Filter2.1	training	0.00%	0.00%
	test	0.00%	0.00%
Filter2.2	training	0.00%	0.00%
	test	0.00%	0.00%
Filter2.3	training	1.62%	0.89%
	test	1.47%	1.27%

5 Results and Conclusions

Table 1 shows the selected genes in the experiment EXP#1. In this table, the column 'tests' indicates if the gene appears in tests with the same Π value, but different Θ value ('X' stands for Θ=0.7, 'O' for Θ=0.8 and 'H' for Θ=0.9). Analyzing these results, for the value Π=0.55, it has been selected a list with 23 probesets (21 of them corresponding to known genes). The list is reduced to 7 probesets when the parameter Π is 0.60 and only 4 probesets when Π=0.65. The HGF (Hepatocyte growth factor precursor) and KCNE1L (Potassium voltage-gated channel, AMMECR2 protein) genes appear in the first positions of the three lists. The HGF gene has been selected by PAM software, and its significance has been validated by the biological technique qr-PCR, whereas PAM has also detected the KCNE1L gene. The FGF13 gene (Fibroblast growth factor 13) also appears as a significant gene when Π=0.55 and Π=0.60. The relative relevance of the FGF13 increases when the Θ parameter increases (this

gene appears as significant when Π=0.60 for the two higher values of parameter Θ). Moreover, this gene has been detected by PAM software and validated by a qr-PCR analysis. The S100B, ANXA8 and SLCS24A3 genes have been also selected by PAM software, whereas the rest of genes are different. Finally, the three lists of genes (respectively referred to as filter1.1, filter1.2 and filter1.3) have been used to train a GCS network. It has been considered the training and test sets of the 4-fold cross validation used in EXP#2. The accuracy of the different classifiers is shown in Table 2.

Table 3 shows the genes selected in the experiment EXP#2. Now, the column 'tests' of the table indicates the number of appearances of a gene in all the possible test for the same value of Π. Additionally, the appearances of the same gene in tests with a great specificity (a higher value of parameter Θ) weigh more than appearances with a lower specificity, when the genes are ranked. As shown in Table 3, the number of selected genes is quite similar in the two experiments (there is only a difference of one gene among filter1.1 and filter2.1). With regard to the degree of overlapping of selected genes in the two experiments, it is also quite similar. The similarity of filter2.1 with regard to filter1.1 is 19/23=82.6%, the similarity of filter 1.1 with regard to filter2.1 is 19/22=86.4%. The degree of overlapping of filter1.2 and filter2.2 is 6/7=85.7% and, it is a 100% in the case of filter1.3 and filter2.3. Finally, Table 4 shows the accuracy of the GCS network when it is trained with the selected genes (filter2.1, filter2.2 and filter3.3). It is remarkable that filter2.2 has an error of 0% predicting novel cases (both within the training set and the test set).

The experiments carried out, show that the number of genes, which are sufficient to correctly classify novel cases, are 7 genes. The genes selected in filter2.2 are especially remarkable, since with a minimal number of genes (7) it is reached a 100% accuracy of the classifier on both training and test sets. The minimal number of genes selected by PAM software, which reaches also an accuracy of 100%, was 23 genes. Therefore, the proposed method has achieved a reduction of the number of genes about the 70% with regard to the PAM software (preserving the classifier accuracy).

From Tables 2 and 4, it can be observed that errors on tests sets are always lesser than errors on training sets. This can also be interpreted as the selected genes are meaningful genes, since they provide an excellent ability of generalization to the constructed classifier.

Finally, from the comparison of similarity among genes selected in the two experiments, we can claim that the proposed method is robust against slight variations of the data set from where genes are selected. It is a desirable feature of the algorithm in order to select truly meaningful genes.

Summarizing our work, in this article we have presented and tested a successful approach of applying fuzzy logic to the process of gene selection and data reduction in the microarray data domain. Our proposed method of fuzzy pattern construction takes advantage of the ability inherent to fuzzy logic to process uncertain, imprecise and incomplete information. In this sense, we have applied fuzzy logic to discretize the original data within the three linguistic labels. This fact leads to the possibility of clearly identifying those genes with a great capacity of discriminate patients based on the selected genes that compose the discriminant fuzzy patterns.

References

1. Cakmakov, D., Bennani, Y.: Feature selection for pattern recognition, Informa Press (2002)
2. Zheng, G., Olusegun, E. Narasimhan, G.: Neural network classifiers and gene selection methods for microarray data on human lung adenocarcinoma. Proc. of the CAMDA 2003 Conference, (2003) 63-67
3. Fuhrman, S. Cunningham, M.J., Wen, X., Zweiger, G., Seilhamer, J.J., Somogyi, R.: The application of Shannon entropy in the identification of putative drug targets. Biosystems 55 (2000) 5-14
4. Li, L., Darden, T.A., Weinberg, C.R., Levine, A.J., Pedersen, L.G.: Gene assessment and sample classification for gene expression data using a genetic algorithm/k-nearest neighbor method. Combinatorial Chemistry and High Throughput Screening, 4(8) (2001) 727-739
5. Blanco, R., Larrañaga, P., Inza, I., Sierra. B.: Gene selection for cancer classification using wrapper approaches. International Journal of Pattern Recognition and Artificial Intelligence 18(8) (2004) 1373-1390
6. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine Learning 46(1-3) (2002) 389-422
7. Chu, F., Wang, L.: Gene Expression Data Analysis Using Support Vector Machines. Bioinformatics using Computational Intelligence Paradigms. Udo Seiffert and Lakhmi C. Jain (Editors). Springer, Berlin (2005) 167-189
8. Liu, L., Wan, C.R., Wang, L.P.: Unsupervised gene selection via spectral biclustering. Proc. of the International Joint Conference on Neural Networks, (2005) 1681-1686
9. Jaeger, J., Sengupta, R., Ruzzo, W.L.: Improved gene selection for classification of microarrays. Proc. of the PSB 2003 Conference, (2003) 53-64
10. Qi, H.: Feature selection and kNN fusion in molecular classification of multiple tumor types. Proc. of the METMBS 2002 Conference, (2002)
11. Hanczar, B., Courtine, M., Benis, A., Hennegar, C., Clément, K., Zucker, J.D.: Improving classification of microarray data using prototype-based feature selection. ACM SIGKDD Explorations Newsletter 5(2) (2003) 23-30
12. Fdez-Riverola, F., Díaz, F., Corchado, J.M., Hernández, J.M., San Miguel, J.: Improving Gene Selection in Microarray Data Analysis using Fuzzy Patterns inside a CBR System. Proc. of the ICCBR 2005 Conference, (2005) 23-26
13. Díaz, F., Fdez-Riverola, F., Corchado, J. M.: GENE-CBR: a Case-Based Reasoning Tool for Cancer Diagnosis using Microarray Datasets. Computational Intelligence. ISSN 0824-7935. In Press.
14. Tibshirani, R., Hastie, T., Narasimhan, B., Chu, G.: Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proc. of the National Academy of Sciences, Vol. 99:(10) (2002) 6567-6572
15. Fritzke, B.: Growing Cell Structures – A Self-Organizing Network for Unsupervised and Supervised Learning. Neural Networks 7 (1994) 1441-1460