An Evolutionary Approach for Sample-Based Clustering on Microarray Data

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

[1] ESEI: Escuela Superior de Ingeniería Informática, University of Vigo, Edificio Politécnico, Campus Universitario As Lagoas s/n, 32004, Ourense, Spain
{dgpena, moncho.mendez, riverola}@uvigo.es
[2] Dept. Informática, University of Valladolid, Escuela Universitaria de Informática, Plaza Santa Eulalia, 9-11, 40005, Segovia, Spain
fdiaz@infor.uva.es
[3] Dept. Informática y Automática, University of Salamanca, Plaza de la Merced s/n, 37008, Salamanca, Spain
corchado@usal.es

Abstract. Sample-based clustering is one of the most common methods for discovering disease subtypes as well as unknown taxonomies. By revealing hidden structures in microarray data, cluster analysis can potentially lead to more tailored therapies for patients as well as better diagnostic procedures. In this work, we present a novel method for automatically discovering clusters of samples which are coherent from a genetic point of view. Each possible cluster is characterized by a fuzzy pattern which maintains a fuzzy discretization of relevant gene expression values. Noise genes are identified and removed from the fuzzy pattern based on their probability of appearance. Possible clusters are randomly constructed and iteratively refined by following a probabilistic search and an optimization schema. Experimental results over publicly available microarray data show the effectiveness of the proposed method.

Keywords: simulated annealing, sample-based clustering, discriminant fuzzy pattern, microarray data.

1 Introduction and Motivation

Within a gene expression matrix, there are usually several particular macroscopic phenotypes of samples related to some diseases or drug effects, such as diseased samples, normal samples or drug treated samples. The goal of sample-based clustering is to find the phenotype structures or sub-structure of these samples. Many conventional clustering algorithms have been adapted or directly applied to gene expression data where the signal-to-noise ratio may seriously degrade the quality and reliability of clustering results. This has the effect of obscuring clustering in samples that may be evident only when looking at a subset of genes.

In this context, existing sample-based clustering methods can be (*i*) directly applied to cluster samples using all the genes as features (i.e., classical techniques as

S. Omatu et al. (Eds.): IWANN 2009, Part II, LNCS 5518, pp. 971–977, 2009.

K-means, SOM, HC, etc.) or (*ii*) executed after a set of informative genes are identified. The problem with the first approach is the signal-to-noise ratio, which is known to seriously reduce the accuracy of clustering results due to the existence of noise and outliers of the samples [1]. To overcome such difficulty, particular methods can be applied to indentify informative genes and reduce gene dimensionality prior to clustering samples in order to detect their phenotypes. In this context, both supervised and unsupervised informative gene selection techniques have been developed.

While supervised informative gene selection techniques often obtain high clustering accuracy rates, unsupervised informative gene selection methods are more complex because they assume no phenotype information being assigned to any sample [2]. In such a situation, two general strategies have been adopted to address the lack of prior knowledge: (*i*) unsupervised gene selection, that aims to reduce the number of genes before clustering samples by using some statistical models [3-5] and (*ii*) interrelated clustering, that takes benefits of utilizing the relationship between the genes and samples to perform gene selection and sample clustering simultaneously in an iterative paradigm [6-10].

In this contribution we propose a simulated annealing-based algorithm for iterative class discovery that uses a novel fuzzy logic method for informative gene selection. The interrelated clustering process carried out is based on an iterative approach where possible clusters are randomly constructed and evaluated by following a probabilistic search and an optimization schema. The rest of the paper is structured as follows: Section 2 introduces the details of our proposed technique discussing relevant aspects of the whole algorithm. Section 3 presents the experimental setup carried out and the results obtained from a publicly available microarray data set. Finally, Section 4 summarizes the main conclusions extracted from this work.

2 Iterative Class Discovery Algorithm

In this section we introduced the proposed method for automatically discovering clusters of samples which are coherent from a genetic point of view. Each possible cluster is characterized by a fuzzy pattern which maintains a fuzzy discretization of relevant gene expression values. Noise genes are identified and removed from the fuzzy pattern based on their probability of appearance. Possible clusters are randomly constructed and iteratively refined by following a probabilistic search and an optimization schema.

2.1 Selecting Relevant Genes

In order to identify potential valuable genes, we use part of a previous successful gene selection technique called DFP (*Discriminant Fuzzy Pattern*) [11]. Our whole DFP algorithm comprises of three main steps. First, we represent each gene value in terms of one from the following linguistic labels: Low, Medium, High and their intersections LowMedium and MediumHigh. The output is a fuzzy microarray descriptor (FMD) for each existing sample (microarray). The second phase aims to find all genes that best explain each class, constructing a supervised fuzzy pattern (FP) for each pathology. Starting from the previous obtained FPs, our technique discriminates those genes that can provide a substantial discernibility between existing classes, generating a unique discriminant fuzzy pattern (DFP). In our present work, we only use steps one and two of the DFP algorithm.

2.2 Filtering Noisy Genes

In order to discard those genes that belong to a given cluster of samples due only to pure chance, we deal with the concept of 'noisy genes'. As the uncertainty decreases (there are predominance of one expression level over the other ones for all the genes in the available set of microarrays) the number of noise genes trends upward (the amount of information encoded by the data also decreases and then, there are more irrelevant genes). When uncertainty increases, the amount of information also grows and more genes are necessary to distinguish arrays in absence of other information.

2.3 Assessing the Value of a Cluster

Our cost function for evaluating each cluster combines two factors: (*i*) the number of genes in the fuzzy pattern associated to each cluster of the partition and (*ii*) the size of such cluster. The first factor in the cost function models the genetic coherence of a cluster. Assuming this hypothesis, it is expected that for clusters with equal sizes, the number of genes in a fuzzy pattern will be greater if the genetic coherence of the cluster is higher. The second factor is relevant since it has been experimentally observed that meaningful genes in great clusters (after noisy genes have been filtered) are several orders of magnitude inferior to meaningful genes computed in small clusters. This fact is reasonable because it will be more probable when the number of possibilities is also more reduced. Therefore, the size factor in the cost function is needed for doing comparable clusters of different size.

2.4 Algorithm

The application of our simulated annealing approach to cluster microarrays is as follows. First of all, we consider a pool which contains the set of m microarrays that must be clusterized into k different and unknown groups. In the final solution, some microarrays can stay in the pool without being associated to any cluster. Initially, a first solution to the problem (a partition of microarrays) is constructed randomly. All the microarrays of the pool are distributed randomly among k classes, where k is the desired number of clusters of the partition (the whole microarrays are spread proportionally among the k clusters and the pool). Figure 1 shows the pseudo-code of the general algorithm.

On every step a neighbour solution is determined by choosing one from the following alternatives: (*i*) either moving a randomly chosen microarray from the pool to a cluster (perhaps empty), (*ii*) or by moving a randomly chosen microarray from a cluster to the pool, (*iii*) or by exchanging randomly chosen microarrays among clusters, (*iv*) or by exchanging randomly chosen microarrays among a cluster and the pool, and (*v*) or by moving a randomly chosen microarray from one cluster to another cluster. The neighbour solutions of lower cost obtained in this way are always accepted, whereas the solutions with a higher cost are accepted with a given probability

The algorithm stops if equilibrium is encountered. We define that equilibrium is reached if after 50 stages of temperature reduction the best solution can not be improved. Opposed to the classical approach in which a solution to the problem is taken

```
01 Create an initial, old_solution as an initial  partition
in k clusters of the microarrays in pool
02 best_solution ← old_solution;
03 equilibrium_conter ← 0;
04 T ← cost(old_solution);
05 repeat
06    old_solution ← best_solution;
07    for iteration_counter← 1 to n do
08       annealing_step(old_solution, best_solution);
09    end for;
10    T ← T · α;
11    equilibrium_counter ← equilibrium_counter + 1;
12 until equilibrium_counter > 50;
```

Fig. 1. General pseudo-code of the simulated annealing-based clustering algorithm

as the last solution obtained in the annealing process, we memorize the best solution found during the whole annealing process. Moreover, at the beginning of each temperature epoch, the search is restarted from the best solution reached for the moment (Cf. line 6 of the *main* procedure presented in Figure1).

Summing up, the annealing algorithm performs the local search by sampling the neighbourhood randomly. It attempts to avoid becoming prematurely trapped in a local optimum by sometimes accepting a low-grade solution. The acceptance level depends on the magnitude of the increment of the solution cost and on the spent search time.

3 Experimental Setup and Results

Dealing with unsupervised classification, it is very difficult to test the ability of a method to perform the clustering since there is no supervision of the process. In [12] the authors proposed that lymphoblastic leukemias with MLL translocations (mixed-lineage leukemia) constitute a distinct disease, denoted as MLL, and show that the differences in gene expression are robust enough to classify leukemias correctly as MLL, acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML). The public dataset of this work has been used to test our proposal. The complete group of samples consists of 24 patients with B-Precursor ALL (ALL), 20 patients with MLL rearranged B-precursor ALL (MLL) and 28 patients with acute myeloid leukemia (AML). All the samples were analyzed using the Affymetrix GeneChip U95a which contains 12600 known genes.

In this sense, the classification into different groups proposed by [12] is assumed to be the reference partition of samples in our work. The results of the proposed clustering algorithm working with this dataset are shown in Table 1. Table 1 presents the percentage of the times that each available microarray has been grouped together with other microarrays belonging to the reference groups (ALL, AML and MLL) in the 10 executions of the algorithm.

Table 1. Clustering carried out by the proposed algorithm using the dataset presented in Armstrong *et al*. [12]

Id . Array	ALL-mll	all-mll	AML-mll	MLL-{all|aml}
ALL-03		0.17	0.50	0.33
ALL-61		0.25		0.75
ALL-06	1.00			
ALL-08	1.00			
ALL-60	1.00			
ALL-11	0.90	0.10		
ALL-19	0.90	0.10		
ALL-07	0.89	0.11		
ALL-58	0.89			0.11
ALL-59	0.89	0.11		
ALL-05	0.88			0.13
ALL-13	0.86			0.14
ALL-02	0.80	0.10		0.10
ALL-20	0.78			0.22
ALL-16	0.75	0.25		
ALL-10	0.70	0.10		0.20
ALL-14	0.70	0.10		0.20
ALL-09	0.67	0.22		0.11
ALL-15	0.57	0.14		0.29
ALL-01	0.50	0.25		0.25
ALL-17	0.50	0.10	0.10	0.30
ALL-18	0.50	0.50		
ALL-12	0.38	0.50		0.13
ALL-04	0.25	0.63		0.13
AML-38			1.00	
AML-39			1.00	
AML-41			1.00	
AML-42			1.00	
AML-43			1.00	
AML-44			1.00	
AML-46			1.00	
AML-49			1.00	
AML-50			1.00	
AML-51			1.00	
AML-52			1.00	
AML-53			1.00	
AML-54			1.00	
AML-57			1.00	
AML-66			1.00	
AML-68			1.00	
AML-69			1.00	
AML-70			1.00	
AML-71			1.00	
AML-72			1.00	
AML-40			0.90	0.10
AML-56			0.90	0.10
AML-67			0.90	0.10
AML-65		0.11	0.89	
AML-55	0.11		0.78	0.11
AML-47			0.70	0.30
AML-48			0.57	0.43
AML-45		0.83	0.17	

Table 1. (*continued*)

Id . Array	ALL-mll	all-mll	AML-mll	MLL-{all|aml}
MLL-33	0.11		0.11	0.78
MLL-29		0.25		0.75
MLL-31		0.25		0.75
MLL-26		0.29		0.71
MLL-23	0.33			0.67
MLL-36	0.11		0.22	0.67
MLL-22	0.13	0.25		0.63
MLL-64	0.30	0.10		0.60
MLL-35	0.11		0.33	0.56
MLL-21	0.25	0.25		0.50
MLL-27	0.33	0.17		0.50
MLL-30	0.40	0.10		0.50
MLL-63	0.20	0.10	0.20	0.50
MLL-32	0.56			0.44
MLL-24	0.29	0.14	0.14	0.43
MLL-37	0.10	0.10	0.40	0.40
MLL-62	0.60	0.10		0.30
MLL-28	0.25	0.38	0.13	0.25
MLL-25	0.14	0.71		0.14
MLL-34	0.25	0.63		0.13

From Table 1 it can be viewed that the AML samples form a group whose samples are clearly distinguished from the rest (only sample AML-45 is mixed with other samples of ALL or MLL clusters, and sample ALL-03 is grouped in a 50% of the executions with other samples of the AML cluster). The confusion is greater between groups ALL and MLL since several samples of type MLL are grouped majorly with samples of ALL group (for example, samples MLL-32 and MLL-62), others are also grouped in a balanced way with samples of ALL/MLL group (MLL-25, MLL-28, MLL-34, ALL-04, ALL-12, and ALL-18), and the sample ALL-61 is grouped majorly with samples of MLL group. These results are reasonable since AML (Acute Myeloid Leukemia) are a different family from the Lymphoblastic Leukemias (ALL and MLL), and the set of MLL samples is speculated to be a potential subtype of the class of ALL.

4 Conclusion

The iterative class discovery method takes advantage of the properties of fuzzy logic and the theory of fuzzy sets for dealing with gene expression unsharp boundaries in which membership is a matter of degree. This method can be used to discover partitions in which biological significance is guaranteed by the similitude between the fuzzy labels assigned to the samples belonging to the cluster. The clustering algorithm can be easily extended to applications different from clustering microarray data.

Acknowledgments. This work is supported in part by the projects *Research on Translational Bioinformatics* (08VIB6) from University of Vigo and *Development of computational tools for the classification and clustering of gene expression data in order to discover meaningful biological information in cancer diagnosis* (ref.

VA100A08) from JCyL (Spain). The work of D. Glez-Peña is supported by a María Barbeito contract from Xunta de Galicia.

References

1. Xing, E.P., Karp, R.M.: Cliff: Clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts. Bioinformatics 17(1), 306–315 (2001)
2. Jiang, D., Tang, C., Zhang, A.: Cluster Analysis for Gene Expression Data: A Survey. IEEE Transactions on Knowledge and Data Engineering 16(11), 1370–1386 (2004)
3. Alter, O., Brown, P.O., Bostein, D.: Singular value decomposition for genome-wide expression data processing and modeling. Proceedings of the National Academy of Sciences of the United States of America 97(18), 10101–10106 (2000)
4. Ding, C.: Analysis of gene expression profiles: class discovery and leaf ordering. In: Proceedings of the Six Annual International Conference on Computational Molecular Biology, pp. 127–136 (2002)
5. Yeung, K.Y., Ruzzo, W.L.: Principal component analysis for clustering gene expression data. Oxford Bioinformatics 17(9), 763–774 (2000)
6. Ben-Dor, A., Friedman, N., Yakhini, Z.: Class discovery in gene expression data. In: Proceedings of the fifth Annual International Conference on Computational Biology, pp. 31–38 (2001)
7. Xing, E.P., Karp, R.M.: Cliff: Clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts. Oxford Bioinformatics 17(1), 306–315 (2001)
8. von Heydebreck, A., Huber, W., Poustka, A., Vingron, M.: Identifying splits with clear separation: a new class discovery method for gene expression data. Oxford Bioinformatics 17, 107–114 (2001)
9. Tang, C., Zhang, A., Ramanathan, M.: ESPD: a pattern detection model underlying gene expression profiles. Oxford Bioinformatics 20(6), 829–838 (2004)
10. Varma, S., Simon, R.: Iterative class discovery and feature selection using Minimal Spanning Trees. BMC Bioinformatics 5, 126 (2004)
11. Glez-Peña, D., Álvarez, R., Díaz, F., Fdez-Riverola, F.: DFP: A Bioconductor package for fuzzy profile identification and gene reduction of microarray data. BMC Bioinformatics 10, 37 (2009)
12. Armstrong, S.A., Stauton, J.E., Silverman, L.B., Pieters, R., den Boer, M.L., Minden, M.D., Sallan, S.E., Lander, E.S., Golub, T.R., Korsmeyer, S.J.: MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. Nature Genetics 20, 41–47 (2002)