IADIS International Journal on Computer Science and Information Systems
Vol. 1, No. 2, pp. 88-99
ISSN: 1646-3692

EVOLUTIONARY ALGORITHMS FOR FINDING INTERPRETABLE PATTERNS IN GENE EXPRESSION DATA

Carlos Cano, Armando Blanco, Fernando García and Francisco J. López *Ciencias de la Computación e I.A. E.T.S. Ingeniería Informática. Universidad de Granada. Periodista Daniel Saucedo Aranda s/n. CP 18071. Granada. España.*

ABSTRACT

Microarray Technology allows us to measure the expression of thousands of genes simultaneously, and under specific conditions. Clustering is the main tool used to analyze gene expression data obtained from microarray experiments. By grouping together genes with the same behavior across samples, resultant clusters suggest new functions for some of the genes. Non-exclusive clustering algorithms are required, as a gene may have more than one biological function. *Gene Shaving* (Hastie et al. 2000) is a clustering algorithm which looks for coherent clusters with high variance across samples, allowing clusters to overlap. In this paper we present two Evolutionary Algorithm approaches, based on Genetics Algorithms (GA) and Estimation of Distribution Algorithms (EDA), whose aim is to find clusters of similar genes with large between-sample variance. We apply our methods *GA-Shaving* and *EDA-Shaving* to *S. cerevisiae* cell cycle dataset outperforming *Gene-Shaving* results in terms of quality and size of obtained clusters. Furthermore, we use GO Term Finder (Boyle et al. 2004) to evaluate the biological interpretation of the results. It computes the most statistically significant biological processes associated to every cluster by means of the annotations of the Gene Ontology (Gene Ontology Consortium 2004).

KEYWORDS

Microarrays, Cluster, Estimation of Distribution Algorithms.

1. INTRODUCTION

1.1 Microarray Technology

Microarray technology makes use of the sequence resources created by the genome projects and other sequencing efforts to monitor the expression of genes in a particular cell type of an organism, at a particular time and under particular conditions.

A microarray is a glass slide on which single-stranded DNA molecules are attached at fixed locations called spots. There can be thousands of spots on a single microarray, each one containing a huge number of identical DNA molecules which identify one gene. Hybridization experiments with two samples consist on the following steps: firstly, the total mRNA from cells in two different conditions (for example, healthy and cancerous cells) is dyed with two different fluorescent labels. Secondly, the labeled mRNA is washed over the microarray. These labeled gene products hybridize to their complementary sequences in the spots. The fluorescence emitted from each spot when the microarray is excited by a laser, allows the measurement of the amount of sample from each condition bounded to the DNA of the spot.

Measuring the fluorescence intensities of various microarrays, each one considering samples at a particular time or under particular conditions, we can characterize the dynamic behavior of a large number of genes in the genome. These gene expression profiles are usually presented in a matrix A_{n*m} where the *n* rows represent genes, the *m* columns the *m* different cell situations we have studied, and every element a_{ij} of the matrix *A* indicates the expression level of gene *i* under condition *j*.

This way, DNA microarray data provide us with a global picture of the cell's activities and open the way to a high-level understanding of its behavior. By analyzing these matrices we can learn more about cellular operation in organisms, but it turns out to be very complex due to the large amount of genes which even the simplest organisms have and to the unlimited number of conditions under we can study them (Berrar et al. 2003).

1.2 Related Work

The potential of clustering to reveal biologically meaningful patterns in microarray data, considering no other knowledge, was demonstrated by Eisen et al. (1998), who applied hierarchical clustering to identify functional groups of genes. After that, other approaches have been proposed to cluster microarray data: k-means (Smet et al. 2002), kohonen maps (Tamayo et al. 1999), etc. (for a review see Jiang et al. 2004). By applying clustering, genes are grouped in sets which have similar expression profiles along conditions (coexpressed genes). This means that genes in the same set respond similarly in different circumstances, so they are likely to share a common function. In addition, clustering can also be used to group conditions with similar gene profiles, so we can also draw conclusions about them. Gene discovery, functional annotation of genes, disease diagnosis, drug discovery and tumor subtypes detection are some of the goals we pursue in this kind of analysis.

However, the mentioned approaches group genes into mutually exclusive clusters, whereas in the real biological system a gene may play multiple roles in different biological processes. To address this, different methods have been proposed: Hastie et al. (2000), Dembele et al. (2003), etc. Our focus is on *Gene Shaving* (Hastie et al. 2000) whose aim is to identify coherent clusters with high between-sample variance, allowing genes to belong to more than one cluster. The algorithm finds a series of nested clusters on the basis of correlation with the leading principal component, so each one of them has the maximum variance of its mean gene, given the cluster size. The nested cluster with more similar genes and higher variance along samples is selected. Then the expression matrix is orthogonalized with respect to the mean of the selected cluster in order to search for a further cluster, which can share genes with the previous one. As they look for high between-sample variance clusters, these obtained

clusters will reveal genes with very different behavior along samples (thus ignoring genes involved in constantly activated processes as well as those involved in none of the active processes), so they become very useful for identifying distinct types of samples and the biological processes which may produce these differences.

The aim of this paper is to present two new approaches based on Evolutionary Algorithms (EAs) for identifying clusters of genes with similar expression patterns and high variance along samples. The first one uses Genetic Algorithms (GAs) and the second one Estimation of Distribution Algorithms (EDAs). We will apply all these methods on yeast cell cycle microarray data by Cho et al. (1998), showing that our algorithms *GA-shaving* and *EDA-shaving* produce larger clusters of more-similar genes with higher between-sample variance than those obtained by *Gene Shaving*. We will also evaluate the biological significance of obtained clusters. For this end, we use the *Gene Ontology* (Gene Ontology Consortium, 2004) and *GO Term Finder* (Boyle et al. 2004) to retrieve the most significant *biological process* term associated to each cluster, extracting relevant and significant insights from yeast expression data.

The rest of the paper is organized as follows. In section 2 we summarize the proposal of Hastie et al. (2000) to obtain clusters with maximum between-sample variance. In section 3.1 we present an alternative solution which uses Genetic Algorithms. In section 3.2 we present another solution employing Estimation of Distribution Algorithms. Section 4 contains experimental results on *S. cerevisiae* cell cycle expression data, and the biological interpretation of obtained clusters using *GO Term Finder*. Finally, section 5 presents conclusions and future work.

2. INITIAL APPROACH: GENE-SHAVING

Rather than simply looking for genes with similar expression patterns, the *Gene Shaving* approach searches for coherent clusters with high variance across samples. The algorithm takes the expression matrix A_{n*m} and the number of desired clusters *M* as input. Let S_k be a cluster of *k* genes and $\bar{a}_{S_k} = \left(\frac{1}{k}\sum_{i \in S_k} a_{i1}, \frac{1}{k}\sum_{i \in S_k} a_{i2}, ..., \frac{1}{k}\sum_{i \in S_k} a_{im} \right)$ be the collection of *m* column averages of the expression values for this cluster. Then, for each cluster size *k*, the algorithm seeks a cluster S_k having the highest variance of the column averages, i.e., S_k which maximizes $\mathrm{var}(\bar{a}_{S_k})$.

For obtaining this cluster, *Gene Shaving* generates a sequence of nested clusters:

$$S_n \supset ... \supset S_{k_i} \supset S_{k_j} ... \supset S_1$$

of decreasing size, starting with *k=n*, the total number of genes, and finishing with *k=1* gene. At each stage the largest principal component of each cluster of genes is computed. This *eigen-gene* is the normalized linear combination of genes with the largest variance across the samples. Then we discard a fraction ($\alpha \in [0,1]$) of the genes having lowest correlation (lowest absolute inner-product) with this *eigen-gene*, obtaining the next nested cluster. The process is repeated until we get a cluster with one gene.

Once the nested sequence of clusters has been completed, the algorithm selects one cluster from the sequence. This selection is made by calculating, by analogy with ANOVA (Analysis of Variance), the following measures of variance for each cluster S_k :

- *Within Variance*: $V_W = \frac{1}{m}\sum_{j=1}^{m}\left[\frac{1}{k}\sum_{i\in S_k}\left(a_{ij}-\bar{a}_j\right)^2\right]$
- *Between Variance*: $V_B = \frac{1}{m}\sum_{j=1}^{m}\left(\bar{a}_j-\bar{a}\right)^2$
- *Total Variance*: $V_T = \frac{1}{k*m}\sum_{i\in S_k}\sum_{j=1}^{m}\left[\left(a_{ij}-\bar{a}\right)^2\right] = V_W + V_B$

  Where $\bar{a}_j = (1/k)\sum_{i\in S_k} a_{ij}$ in all expressions above.

The *Within Variance* measures the variability between the genes of the cluster (cohesion of the cluster), so minimizing this measure we will obtain clusters with similar gene profiles. The *Between Variance* is the variance of the mean gene of the cluster (variance across samples), so we want to maximize this measure to get a cluster with high variability over the samples.

To take these two measures into account, the *percent of variance explained* (R^2) is computed:

$$R^2 = 100\frac{V_B}{V_T} = \frac{\frac{V_B}{V_W}}{1+\frac{V_B}{V_W}}$$

So large R^2 values imply high values of V_B and low values for V_W .

But we also need to know whether a value of R^2 for a given cluster S_k is larger than we would expect by chance, if the rows and columns of *A* were independent. To overcome this problem, Hastie et al. (2000) proposed the following measure.

Let D_k be the R^2 measure for S_k , and A^{*b} a permuted data matrix, obtained by randomly permuting the elements of each row of *A*. If we form *B* such matrices, we define *GAP* function as:

$$GAP(S_k) = D_k - \overline{D}_k^{\,*}$$

Where $\overline{D}_k^{\,*}$ is the mean R^2 value for S_k in the *B* randomly permuted matrices: $A^{*1},...,A^{*B}$

This way, a large *GAP* value for S_k will reveal a relevant (non-spurious) pattern.

After selecting one cluster from the sequence, *A* is orthogonalized with respect to the mean of the selected cluster, promoting new patterns to be revealed in further iterations.

The whole process is shown in Figure 1.

1. Start with the entire expression matrix X, each row centered to have zero mean.
2. Compute the leading principal component of the rows of X.
3. Shave off the proportion α (typically 10%) of the genes having smallest absolute inner-product with the leading principal component.
4. Repeat steps 2 and 3 until only one gene remains.
5. This produces a sequence of nested clusters $S_N \supset S_{k_1} \supset S_{k_2} \supset \ldots \supset S_k \supset \ldots \supset S_1$ where S_k denotes a cluster of k genes. Estimate the optimal cluster $S_{\hat{k}}$ using the GAP statistic.
6. Orthogonalize each row of X with respect to the average gene in $S_{\hat{k}}$.
7. Repeat steps 1-5 above with the orthogonalized data to find the second optimal cluster. This process is continued until a maximum of M clusters are found, where M is chosen *a priori*.

Figure 1. The *Gene Shaving* process (Hastie et al. 2000).

3. PROPOSED METHODS

As can be noted from Figure 1, the generation of the sequence of nested clusters is strongly driven by the variance of the genes along conditions, as genes are *shaved-off* depending on their correlation with the leading principal component. However, we are not only interested in clusters with high between-sample variance, but also in high-coherence clusters, and this criterion is not used for obtaining the clusters sequence. It is only considered at the end, when we compute the *GAP* statistic for every cluster of the sequence in order to select one of them.

The *shaving* process can be seen as a multiple-step *Feature Subset Selection (FSS)* problem in which, given a set of genes S_k with $k \in [2, n]$, we want to select a subset with $k(1-\alpha)$ genes: $S_{k*(1-\alpha)} \subset S_k$ which maximizes a given criterion. As we have mentioned, in *Gene Shaving* the optimization criterion is the variance of the cluster mean. We consider that maximizing *GAP* function instead of between-sample variance will provide overall better results. In this work, we address the *FSS* problem of finding clusters with high values for the *GAP* function with Evolutionary Algorithms (EA) and in particular, with Genetic Algorithms (GA) and Estimation of Distribution Algorithms (EDA), which have been proven to have an excellent performance on highly complex optimization problems.

3.1 Genetic Algorithms approach: GA-Shaving.

Genetic Algorithms (GAs), initially introduced by Holland (1975), are stochastic global search heuristics optimization methods based on the mechanics of natural selection and genetic recombination. Genetic algorithms typically maintain a constant-sized population of individuals which represent samples of the space to be searched. Each individual is evaluated on the basis of its fitness with respect to the given application domain. New individuals are

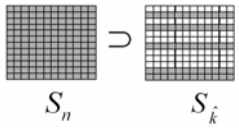
produced by selecting high performing individuals to recombine their "genetic material", so they will retain many of the features of their "parents". This eventually leads to a population that has improved fitness with respect to the given goal.

For addressing the above FSS problem, we have implemented a GA with elitism. Its main characteristics are:

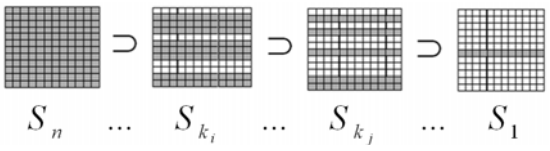
- Representation for solutions: each individual is a binary string of length *k* representing whether each gene is selected in the cluster or not.
- Selection: Baker's stochastic universal sampling. This is a roulette wheel method with slots which are sized according to the fitness of each individual.
- Crossover operator: given two parents, the offspring maintain the common values to both parents.
- Mutation operator: *BitFlip* operator. Given an individual in the population, a fraction of its bit values are changed for their complementary values.
- Fitness: *GAP* function.
- Restart. For the restart strategy, we have chosen to move the best individual to the new population. In addition, 20% of the new population individuals will be obtained from mutating the best individual of current population. This restart will be applied when 10% of the generations to be made have taken place with no change in the best element of the population.

There are two ways we can apply GAs to our optimization problem:

1.-Single-step FSS: only one execution of a GA which takes the whole matrix as input and directly generates a resulting cluster maximizing *GAP*. We have implemented this approach but the search space is so vast that the GA could not converge to good solutions in a reasonable time.



2.-Multiple-step FSS: we can generate a nested sequence of clusters: $S_n \supset ... \supset S_{k_i} \supset S_{k_j} ... \supset S_1$ by selecting/discarding, step by step, a fraction of the remaining genes with a GA guided with the *GAP* function.



This approach provides good results as we will show in section 4.

3.2 EDA approach: EDA-Shaving.

Estimation of Distribution Algorithms (EDAs) are a set of Evolutionary Algorithms mainly characterized by the use of explicit probability models to recover the information of the selected individuals and to sample new solutions (Larrañaga & Lozano 2001). In EDAs, there are neither crossover nor mutation operators. Instead, a probabilistic model is inferred from selected individuals of the current generation, and the new population of individuals is sampled from the estimated distribution (see Figure 2).

The simplest way to compute the probability distribution consists in considering all the variables of a problem to be independent. The joint probability distribution is therefore converted into the product of the marginal probabilities of *n* variables:

$$p_l(\vec{x}) = \prod_{i=1}^{n} p_l(x_i)$$

where $p_l(\vec{x})$ represents the joint probability distribution of selected individuals in generation *l*, $\vec{x} = (x_1,...,x_n)$ represents the *n* variables, and $p_l(x_i)$ are the independent univariate marginal distributions, which are estimated from marginal frequencies:

$$p_l(x_i) = p\left(x_i \mid D_{l-1}^{Se}\right)$$

where D_{l-1}^{Se} is the set of selected individuals from previous generation ($l-1$).

As we are addressing a FSS problem, the univariate distribution $p_l(x_i)$ can be computed as the fraction of individuals from D_{l-1}^{Se} for which the feature x_i is selected.

This algorithm was introduced by Muhlenbein (1998) and is called Univariate Marginal Distribution Algorithm (UMDA) (See Figure 3).

We implemented the UMDA algorithm and we applied it to solve the FSS problem following the two schemes proposed in the previous section: single-step FSS and multiple-step FSS, obtaining good results with both approaches (see next section).

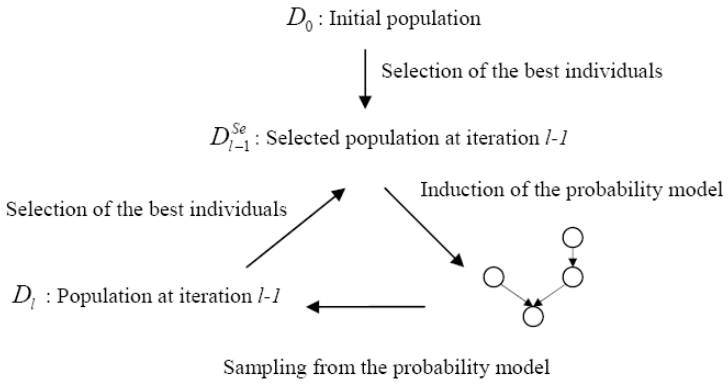


Figure 2. Scheme of an EDA Algorithm.

UMDA Algorithm

---

1. $D_o \leftarrow$ Generate M individuals (Initial Population) at random
2. Repeat for $l=1,2,\ldots$ until the stopping criterion is met

   $D_{l-1}^{Se} \leftarrow$ Select best $N \leq M$ individuals from D_{l-1}

   Estimate the joint probability distribution: $p_l(x_i) = p\left(x_i \mid D_{l-1}^{Se}\right)$

   $D_l \leftarrow$ Sample M individuals from $p_l(\vec{x})$

Figure 3. Pseudocode for UMDA (Larrañaga & Lozano 2001).

4. EXPERIMENTS

We have tested *Gene Shaving*, *GA-Shaving* and *EDA-Shaving* against yeast *S. cerevisiae* cell cycle expression data from Cho et al. (1998). This dataset contains the expression levels of 2879 yeast genes under 17 cell cycle conditions, covering approximately two full cell cycles.

We focus this comparison on the *GAP* value and size of obtained clusters. Furthermore, we have also considered the biological interpretation of the results. We use *GO Term Finder* (Boyle et al. 2004) to find out the most significantly enriched *Gene Ontology* terms (Gene Ontology Consortium 2004) associated with the genes belonging to every obtained cluster. This tool allows us to determine whether any GO term annotates a specified list of genes at a frequency greater than would be expected by chance, by calculating the associated p-value.

Once we have, for a given set of genes, the p-values for all GO terms, our attention will focus on those from the *biological process* ontology, and we will select the one with lower p-value for representing this set of genes. We focus on this ontology, and not on *cellular component* or *molecular function*, because genes belonging to a cluster of co-expressed genes respond similarly along the samples, so they are likely to participate in the same biological process.

Associating a biological process to each obtained cluster, with a p-value representing the statistical significance of this association, is a way of validating our clustering method. As we will see, a large fraction of our clusters can be assigned to GO biological processes with high reliability, so we can state that our algorithms describe accurately the known classification (in this case, the one given by *the Gene Ontology*) and, in this way, are reliable for extracting new biological insights. The association of biological processes to clusters can also be used to annotate genes with unknown function: if one of these genes is highly co-expressed with a group of genes which has a significant biological function, it will probably play the same role.

4.1 Results Comparison.

Table 1 shows average *GAP* and size (number of genes) for 30 clusters obtained in three executions of each algorithm: *Gene Shaving*, *GA-Shaving* and *EDA-Shaving* (with multiple-step and single-step *shaving*).

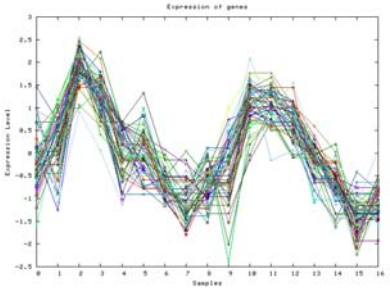
Table 1. Averages (and standard deviations, in parenthesis) of *GAP* values and cluster sizes for 30 clusters

Algorithm	*GAP*	Size
Gene Shaving	62.04 (23.8)	14.86 (10.3)
GA-Shaving	80.78 (3.7)	15.43 (4.3)
EDA-Shaving (multiple-step shaving)	82.02 (3.4)	16.9 (6.2)
EDA-Shaving (single-step shaving)	76.32 (6.1)	34.3 (8.6)

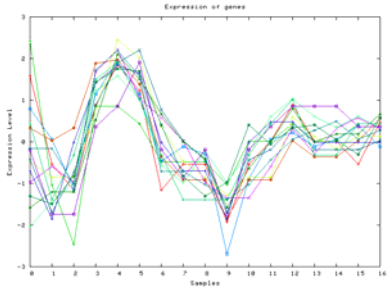
GA-Shaving and *EDA-Shaving* (both *single-step and multiple-step shavings*) outperform *Gene Shaving* in average *GAP*. The clusters identified by the proposed algorithms also show lower variability in their *GAP* values than those obtained by *Gene Shaving*. Therefore, the proposed algorithms find clusters with higher coherence and higher between-sample variance than *Gene-Shaving*. Moreover, when we apply an *EDA* to solve the FSS problem in only one step, we obtain clusters with a little lower *GAP* than our multiple-step approaches (*GA-Shaving* and *multiple-step EDA Shaving*), but with higher *GAP* than *Gene Shaving* and higher cluster size than any other method.

4.2 Biological interpretation of obtained clusters.

As we have mentioned above, by using *GO Term Finder* we can assign the GO term with lowest p-value to every obtained cluster. Significant biological signals are revealed when we consider high-*GAP* and low-p-value clusters (Figure 4). This way we can validate our algorithms and interpret the results to extract new and reliable biological knowledge. For example, looking at the first plot in Figure 4 we can confirm the correspondence between the biological process *DNA metabolism*, which is the one with lowest p-value for this cluster, and the expression behavior of the genes belonging to the cluster, which are over-expressed in samples 2-3 and 10-12. These samples are associated to the *S phase* of cell cycle, in which DNA replication takes place.



DNA metabolism. P-value: 1,8e-13. GAP:83,38. size:50 genes.
Single-step EDA-Shaving

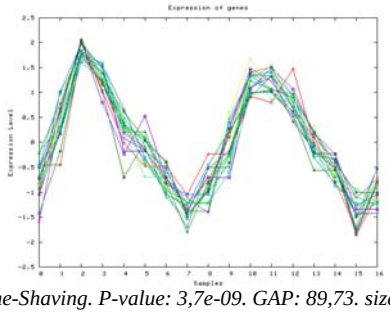


Sulfur metabolism. P-value: 7,2e-15. GAP:83,4. size:14 genes.
Multiple-step EDA-Shaving

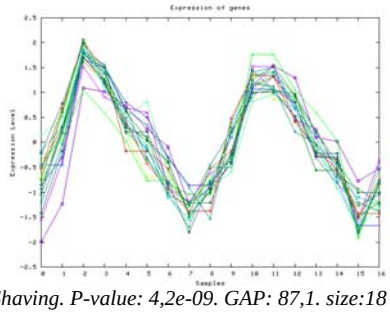
Figure 4. Expression profiles for some biologically significant clusters obtained with *EDA-Shaving*.

An example of how *EDA-Shaving (single-step scheme)* outperforms all the other methods in cluster size can be seen in Figure 5. All the algorithms have found a cluster significantly associated to *DNA replication*. It can be observed that, unless *Gene-Shaving*, *AG-Shaving* and *multiple-step EDA-Shaving* present higher *GAP* values for their clusters, *single-step EDA-Shaving* groups together much more genes (18 for *Gene-Shaving* against 38 for *EDA-Shaving*)

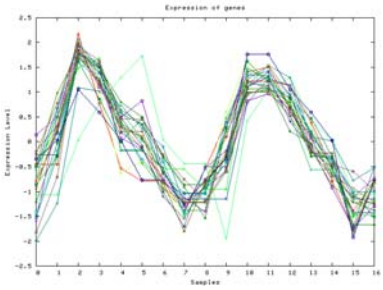
with highly similar patterns, good *GAP* value and very low p-value. So the cluster found by *single-step EDA-Shaving* seems to be the most useful and valuable for extracting new biological insights.



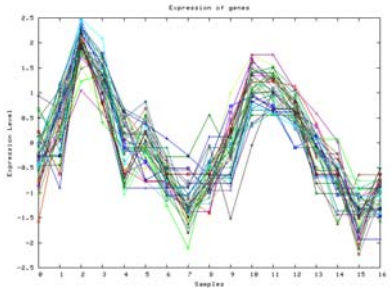
Gene-Shaving. P-value: 3,7e-09. GAP: 89,73. size:18 genes



AG-Shaving. P-value: 4,2e-09. GAP: 87,1. size:18 genes



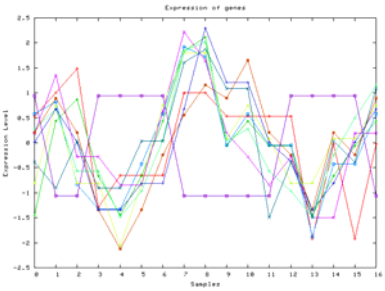
Multiple-step EDA-Shaving. P-value: 1,5e-10. GAP: 89,27. size:24 genes



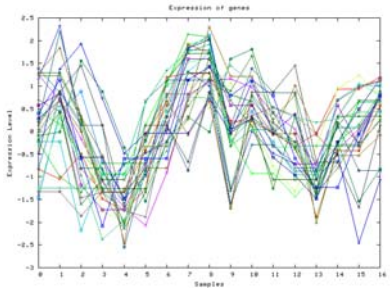
Single-step EDA-Shaving. P-value:9,3e-14. GAP:85,2. size:38 genes

Figure 5. Gene expression profiles for significant clusters representing '*DNA replication*', obtained with *Gene-Shaving*, *AG-Shaving*, *multiple-step EDA-Shaving* and *single-stepEDA-Shaving*.

Figure 6 shows another example. Both clusters are significantly associated with *DNA unwinding* but the one obtained by *EDA-Shaving* has higher *GAP* and size than the one obtained by *Gene-Shaving*.



Gene-Shaving. P-value: 4,6e-11. GAP:42,58. size:10 genes



Single-step EDA-Shaving. P-value: 1,64e-05. GAP: 73. size:29 genes

Figure 6. Gene expression profiles for significant clusters representing '*DNA unwinding*', obtained with *Gene-Shaving* and *single-step EDA-Shaving*.

For comparing *Gene-Shaving* and *EDA-Shaving* in terms of biological significance of the obtained clusters, we have used *correspondence plots* (Tanay et al. 2002). These plots depict the distribution of p-values of the clusters using a given gene annotation (in our case, *Gene Ontology* annotations made by *Saccharomyces Genome Database –SGD-*). The plot represents, for each value of *p*, the fraction of clusters whose p-value is at most *p* out of the, in our case, 30 clusters obtained with each method in total. Figure 7 shows correspondence plots for *Gene-Shaving*¸ *EDA-Shaving* (*single step* scheme) and random clusters of the same size as *EDA-Shaving*. We can see that *Gene-Shaving* and *EDA-Shaving* present a very similar distribution of p-values, different from the one associated to poor significant random clusters. Therefore, *Gene-Shaving* and *EDA-Shaving* group together genes with highly related biological functions, but as previous results have shown, *EDA-Shaving* finds clusters with *more* genes than *Gene-Shaving*, so we consider that *EDA-Shaving* results are more informative from a biological point of view.

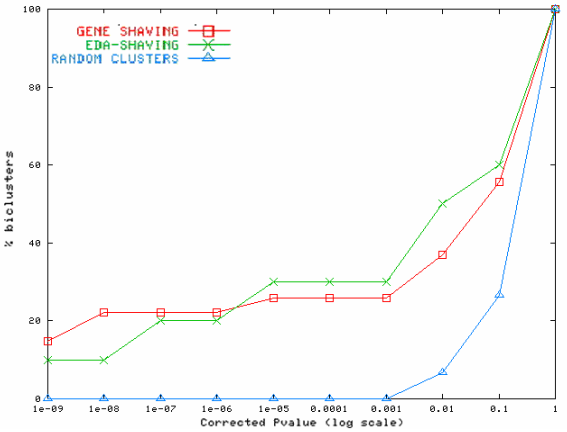


Figure 7. Correspondence plots for clusters generated with *Gene-Shaving*, *EDA-Shaving* and random clusters of fixed size (35 genes). These plots depict the distributions of p-values of the GO terms representing each obtained cluster.

5. CONCLUSION

We have presented two new clustering algorithms: *GA-Shaving* and *EDA-Shaving* which look for coherent clusters of high between-samples variance. Experimental results demonstrate that the proposed algorithms outperform *Gene-Shaving* in terms of *GAP* value and, in the case of *single-step EDA-Shaving*, also in the size of obtained clusters. Moreover, the resulting clusters are biologically significant. The paper shows the methodology we use for validating and interpreting the results from a biological point of view with *GO Term Finder*.

The main drawback of all studied methods comes from the fact that the original matrix is orthogonalized with respect to the mean of the last obtained cluster in order to search for a further cluster. This limits the total number of clusters that can be obtained.

Gene discovery, functional annotation of genes, disease diagnosis, and tumor subtypes detection are some of the goals we pursue with the analysis of gene expression matrices by using such clustering algorithms.

ACKNOWLEDGEMENT

This work has been carried out as part of projects TIC-640 of J.A. Sevilla and TIC-2003-09331-C02-01 of DGICIT. Madrid.

REFERENCES

Berrar, P. et al. 2003. *A Practical Approach to Microarray Data Analysis.* Kluwer Academic Publishers, USA.

Boyle, E.I. et al. 2004. GO::TermFinder – open source software for accesing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics*, Vol.20, 973-980.

Cho, R.J. et al 1998. A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell.* Vol. 2, No. 1, 65-73.

Dembele,D. and Kastner,P.2003. Fuzzy C-means method for clustering microarray data. *Bioinformatics.*Vol.19, 973-980.

Eisen, M. et al 1998. Cluster analysis and display of genome-wide expression patterns. *PNAS. USA.* 95. 14863-14868.

Gene Ontology Consortium. 2004. *Nucleic Acids Research.* Vol. 32, D258-D261. http://www.geneontology.org

Hastie, T. et al, 2000. 'Gene Shaving' as a method for identifying distinct set of genes with similar expression patterns. *Genome Biology*, Vol. 1, No. 2, 1-21.

Holland, J.H. 1975. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor.

Jiang D. et al., 2004. Cluster Analysis for gene expression data: A survey. *IEEE Transactions on Knowledge and Data Engineering*. Vol. 16, No. 11, 1370-1386.

Muhlenbein, H. 1998. The equation for response to selection and its use for prediction. *Evol.Comp.* Vol. 5, 303-346.

Larrañaga, P. and Lozano, J.A. 2001. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Kluwer Academic Publishers.

Smet et al. 2002. Adaptive quality-based clustering of gene expression profiles. *Bioinformatics*. Vol. 18. 735-746

Tamayo, P. et al 1999. Interpreting patterns of gene expression with self-organizing maps. *PNAS. USA* 96. 2907-2912.

Tanay, A. et al 2002. *Discovering statistically significant biclusters in gene expression data. Bioinformatics.* 18. 136-144.