On Evolutionary Algorithms for Biclustering of Gene Expression Data

A. Carballido Jessica[*,1], A. Gallo Cristian[1], S. Dussaut Julieta[1] and Ponzoni Ignacio[1,2]

[1]*Laboratorio de Investigación y Desarrollo en Computación Científica (LIDeCC), Dpto. de Cs. e Ing. de la Computación, UNS, Av. Alem 1253, 8000, Bahía Blanca, Argentina*

[2]*Planta Piloto de Ingeniería Química, UNS – CONICET, Complejo CRIBABB, Co. La Carrindanga km.7, CC 717, Bahía Blanca, Argentina*

A. Carballido Jessica

Abstract: Past decades have seen the rapid development of microarray technologies making available large amounts of gene expression data. Hence, it has become increasingly important to have reliable methods to interpret this information in order to discover new biological knowledge. In this review paper we aim to describe the main existing evolutionary methods that analyze microarray gene expression data by means of biclustering techniques. Strategies will be classified according to the evaluation metric used to quantify the quality of the biclusters. In this context, the main evaluation measures, namely mean squared residue, virtual error and transposed virtual error, are first presented. Then, the main evolutionary algorithms, which find biclusters in gene expression data matrices using those metrics, are described and compared.

Keywords: Biclustering, evaluation metrics, evolutionary algorithms, gene expression data, microarray analysis, regulatory networks.

1. INTRODUCTION

Functional genomics is a branch of genomics that aims at discovering the biological function of genes and their products. In other words, this research area uses the emerging knowledge about different genomes in order to comprehend the gene and their product functions and interactions, and on the whole, how all this affects the organisms and their functioning. For this purpose, several methods are constantly being developed for the analysis of the enormous amounts of information on the genome.

Particularly, a common problem being faced in the field of functional genomics consists in the analysis of data that corresponds to the expression level of some genes under different conditions. The main objective in this sense is to determine whether those genes are related in some manner, thus regulating each other [1, 2]. The information about the expression levels is generally obtained through microarray experiments. DNA microarray technology provides huge amounts of data; therefore, in order to process that kind of information, metaheuristics constitute a valuable tool. In the remaining of this section, some basic concepts of the microarray experiments and the resulting information are introduced, and also the core of evolutionary algorithms is described. Next, some definitions for the analysis of gene expression data are given, including a definition of the "bicluster" concept, types, and evaluation metrics. Finally, the review of the main evolutionary methods used to analyze gene expression data will be presented, culminating the article with a section of conclusions.

1.1. Gene Expression Data

The task of grouping genes that present a related behavior can be performed according to the genes' expression levels [3, 4]. As it was aforementioned, the success in this labor helps infer the biological role of genes. In this context, the microarray technology arose as a central tool that provides information about the behavior of thousands of genes under several experimental conditions. The information provided by a microarray experiment corresponds to the relative abundance of the mRNA of genes under a given condition. The abundance of the mRNA is a metric that can be associated with the expression level of the gene. This information can be arranged in a matrix, where rows and columns correspond to genes and experiments respectively. The mission of finding groups of related genes and conditions in the gene expression data matrix is called biclustering [5-9].

1.2. Evolutionary Algorithms (EAs)

Evolutionary algorithms are meta-heuristic methods inspired by biological evolution features such as mutation, recombination, reproduction and natural selection. Usually, the problem search space is described by a genotype, and operators like mutation or reproduction are applied to create new candidate solutions. Finally, a cost function (the so-called fitness function) determines which solution to preserve. These operations are repeated several times and due to natural selection, candidate solutions improve over time. According to some implementation details, evolutionary algorithms are categorized into several types, such as genetic algorithms and genetic programming among others [10].

*Address correspondence to this author at the Laboratorio de Investigación y Desarrollo en Computación Científica (LIDeCC), Dpto. de Cs. e Ing. de la Computación, UNS, Av. Alem 1253, 8000, Bahía Blanca, Argentina;
Tel: (54)(291)4595135; Fax: (54)(291)4595136;
E-mails: jessicarballido@gmail.com, jac@cs.uns.edu.ar

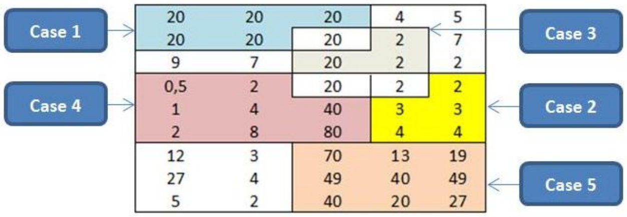
1574-8936/15 $58.00+.00

2. BICLUSTERING

As it was afore-said, gene expression data can be viewed as a matrix that contains expression values, where rows correspond to genes and columns to samples or conditions. A matrix element e_{ij} contains the measured expression value for the corresponding gene i in sample (or under condition) j. In this context, a bicluster can be defined as a pair (G, C) where G is a subset of genes (rows) and C is a subset of conditions (columns). In general, the main goal is to find the largest bicluster that does not exceed a certain homogeneity constrain [5].

2.1. Bicluster Types

An important issue to be considered when analyzing the strategies for biclustering consists in the types of biclusters a particular method is able to find. Main biclusters' classes are those with constant values (case 1 in Fig. **1**), constant values in rows or columns (cases 2 and 3, respectively, in Fig. **1**), coherent values (case 4 in Fig. **1**) and coherent evolution of values (case 5 in Fig. **1**).



20	20	20	4	5
20	20	20	2	7
9	7	20	2	2
0,5	2	20	2	2
1	4	40	3	3
2	8	80	4	4
12	3	70	13	19
27	4	49	40	49
5	2	40	20	27

Fig. (1). Bicluster types.

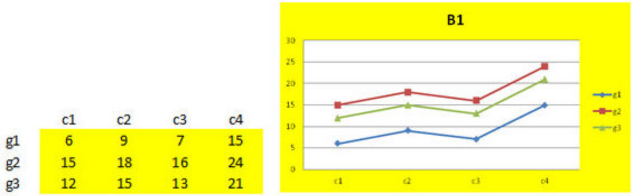
Clearly, constant valued biclusters are more easily identified by biclustering algorithms. In gene expression data matrices, those types of biclusters represent groups of genes that during the experimentation exhibited identical expression values under some subset of conditions. As to the second case, biclusters with constant values in rows indicate a subset of genes with unchanging expression levels across a subgroup of conditions, regardless of the actual expression levels of the individual genes. In relation to biclusters with constant values in columns, they isolate a subset of conditions for which a subgroup of genes have constant expression values that might change from one condition to another. Coherent valued biclusters detect subsets of genes up-regulated or down-regulated coherently across subgroups of conditions, exhibiting in this manner same magnitudes and same directions of the values; they will be further discussed in the following subsection as they constitute two special patterns of interest. Finally, coherent evolution of the values in a bicluster identifies the subsets of genes that are up-regulated or down-regulated coherently across subgroups of conditions notwithstanding of their real values, this means, in this case, same directions with different magnitudes [11].

2.2. Bicluster Patterns

The main feature of biclusters with coherent values is that the genes follow a similar behavior often called "pattern". Two different patterns can be defined: shifting and scaling patterns [12, 13]. A bicluster B follows an additive pattern when every value w_{ij} can be obtained by adding a given value B_i (which remains constant all along the i^{th} condition) to a typical value π_j for the j^{th} gene. Formally, a bicluster exhibits a shifting pattern when its values can be described by the following expression:

$$w_{ij} = \pi_j + B_i + \xi_{ij}$$

where w_{ij} denotes the value for gene j under the i^{th} condition; B_i is the shifting value for the i^{th} condition and ξ_{ij} represents an error. In Fig. (**2**), a bicluster that presents an additive pattern is shown and illustrated.



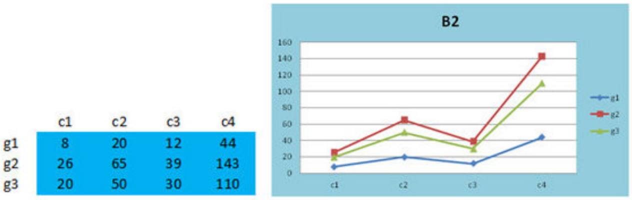
	c1	c2	c3	c4
g1	6	9	7	15
g2	15	18	16	24
g3	12	15	13	21

Fig. (2). Shifting pattern.

Likewise, the definition of a scaling pattern is analogous to the previous one one, but a substitution of the additive value B_i for a multiplicative a_i is needed, as shown in the expression:

$$w_{ij} = \pi_j * \alpha_i + \xi_{ij}$$

where w_{ij} denotes the value for gene j under the i^{th} condition; α_i is the scaling value for the i^{th} condition and ξ_{ij} represents an error. In Fig. (**3**), a bicluster that presents a multiplicative pattern is represented.



	c1	c2	c3	c4
g1	8	20	12	44
g2	26	65	39	143
g3	20	50	30	110

Fig. (3). Scaling pattern.

Under these definitions, when ξ_{ij} is 0 for all i and j, the bicluster is said to be a perfect bicluster. As it can be seen in the graphics of Figs. (**2**, **3**), in the case of a shifting pattern, the lines that represent the genes have the same shape, presenting identical slope in all the stretches, only changing the range in which they are situated. On the other hand, for scaling patterns, even the shape is shared among the lines, the gradients differ.

2.3. Bicluster Evaluation Metrics

2.3.1. Variance

The variance determines the manner in which the data extends around a central value, such as the average of the values in the bicluster B. It can be defined as:

$$\mathrm{VAR}(\,B\,) = \sum_{i\in I, j\in J}(b_{ij} - b_{IJ})^2$$

where b_{ij} is the element in the i^{th} row and j^{th} column and b_{IJ} is the mean of all the values in the bicluster. The value of this equation is 0 whenever B is a perfect bicluster. The variance constitutes a homogeneity parameter that is usually combined with the analysis of the size of the bicluster in almost all of the simplest biclustering methods. In particular, row variance is generally used as a part of the objective function in many of them.

2.3.2. Mean Squared Residue (MSR)

The MSR [14] quantifies the numeric coherence of the values in the bicluster B; the lower the value of residue, the stronger the coherence, and the better the quality of B. It can be calculated with the following formula:

$$\text{MSR(B)} = \frac{1}{I.J}\sum_{i=1}^{I}\sum_{j=1}^{J}(b_{ij} - b_{iJ} - b_{Ij} + b_{IJ})^2$$

where b_{ij} is the element in the i^{th} row and j^{th} column, b_{iJ} is the mean of the j^{th} column, b_{Ij} is the mean of the i^{th} row and b_{IJ} is the mean of all the values in the bicluster. Then, a small value of MSR means a great coherence among the values in the bicluster. If all the genes present an identical behavior under all the conditions, the value of this residue is 0.

2.3.3. Virtual Error

The virtual error [15] aims at creating a pattern for the bicluster so as to represent the general trend of all the genes. The idea is to obtain a pattern that properly identifies the behavior of the genes throughout all the experimental conditions, independently of their numeric values. Some previous calculations must be performed before obtaining the virtual error.

First, given a bicluster B with I conditions and J genes, the *virtual gene* or behavioral pattern P is defined as the collection of I elements named P_i with:

$$\text{P}_\text{i} = \frac{\sum_{j\in J} b_{ij}}{J}$$

where $b_{ij} \in B$, $1 \leq i \leq I$, $1 \leq j \leq J$. Then, each point in P represents a meaningful value for all the genes under a given condition. After the pattern is created, it is necessary to quantify the manner in which all the genes can be adjusted to P. In order to do that, both, the pattern and the bicluster, must be standardized, thus obtaining P' and B' in the following manner:

$$b'_{ij} = \frac{b_{ij} - b_{Ij}}{\sigma_{g_j}}$$

where $b_{ij} \in B$, $1 \leq i \leq I$, $1 \leq j \leq J$, b_{Ij} is the mean of i^{th} row, and σ_{g_j} is the standard deviation of all the expression values for gene j.

$$\text{P}_\text{i}\text{'} = \frac{P_i - \bar{P}}{\sigma_P}$$

where P_i is the i^{th} value in the pattern (value for condition i), $\bar{P}$ and $\sigma_P$stand for the mean and standard deviation of the pattern respectively.

Then, given a bicluster B with I conditions and J genes, and a pattern P with I values, the virtual error (VE) of the bicluster B is defined as the mean of the numeric differences among each standardized gene and each standardized pattern value, for every condition:

$$\text{VE(B)}= \frac{1}{I.J}\sum_{i=1}^{i=I}\sum_{j=1}^{j=J}(b'_{ij} - P'_i)$$

A bicluster with a low value of VE is considered better than those with a high value of VE, since the value of the VE diminishes whenever the values of the genes are more alike.

2.3.4. Transposed Virtual Error (VE^t)

VE^t [16] constitutes an improvement of its ancestor VE. Conceptually, this new measure initially creates a *virtual condition* instead of a *virtual gene* structure. Then, given a bicluster B with I conditions and J genes, the *virtual condition* P is defined as the collection of J elements named P_j with:

$$\text{P}_\text{j} = \frac{\sum_{i\in I} b_{ij}}{I}$$

where $b_{ij} \in B$, $1 \leq i \leq I$, $1 \leq j \leq J$. Then, each point in P represents a meaningful value of all the conditions for a given gene. After this step, the following stages are the same as those for obtaining the value of the original virtual error. B and P are standardized, and then VE^t is calculated analogously to VE for B (only in this case the vector P represents a virtual condition).

In Fig. (**4**), a brief overview of the relationships between biclusters' patterns and the evaluation measures that are able to detect each one of them are depicted. All in all, the main idea is that MSR can only detect shifting patterns, VE detects whether the bicluster presents a shifting or scaling pattern, but separately; and VE^t is the only metric that can identifies both patterns simultaneously.

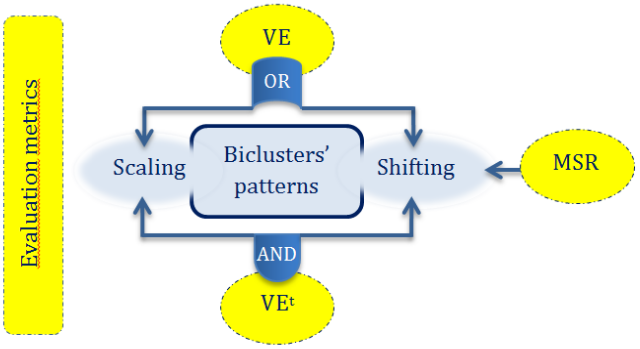


Fig. (4). Patterns and evaluation metrics.

3. EVOLUTIONARY ALGORITHMS FOR MICROARRAY BICLUSTERING

All along this section, several evolutionary methods will be chronologically presented. In all the algorithms, the individuals represent biclusters, and they all use some of the aforementioned measures in order to compute the fitness value.

3.1. MSR-Based Methods

3.1.1. Bleuler et al. [17]

In this article, the authors present the first informed method that tackles microarray biclustering by means of an evolutionary algorithm. A binary representation for the individuals is adopted, and independent bit mutation and uniform crossover are used. Each individual stands for a given bicluster *B*, represented by a binary string of length I+J (I and J denoting the number of genes and conditions respectively). A 1-value in the string means that the corresponding value is selected for the bicluster. The fitness function F is minimized and it is defined in cases as follows:

$$\text{F(B)} = \begin{cases} \frac{1}{|I||J|} \; if \; MSR(B) \leq \delta \\ \frac{MSR(B)}{\delta} \; otherwise. \end{cases}$$

For the first situation a better fitness value, calculated only by using the size of the bicluster, is assigned to those individuals that comply with the residue restriction. If the bicluster has a residue from a given threshold, then a value greater than 1 is set. For the second case, as the residue constraint is considered by the LS strategy, they only look at the size of the biclusters for the fitness assignment.

Several variants were presented in this work. The use of a single-objective EA, an EA combined with a local search (LS) strategy [14] and the local search strategy alone are analyzed. In the case of the EA, one novelty of the strategy consists in a form of diversity maintenance that can be applied during the selection procedure. For the case of the EA hybridized with a LS strategy, the authors consider whether the new individual yielded by the LS procedure should replace the original individual (*Lamarckian* approach) or not (*Baldwinian* approach). As regards the LS as a standalone strategy, they propose a non-deterministic method, where the decision on the course of execution is made according to some probability.

For the experiments, two datasets were used: *Yeast* [18] and *Arabidopsis thaliana* [19, 20]. The study of the results is organized considering whether the aim is to get a unique bicluster or a set of biclusters. For the analysis of a single bicluster, the evaluation is focused on the size of the biclusters, and the algorithm that performed better was the EA combined with the LS method by means of an updating policy. For the second case of analysis, a comparison of the results as regards the level of covering of the matrix is performed, and the hybridized EA with diversity maintenance combined with LS performed better.

3.1.2. SEBI [21]

Another approach, called SEBI for Sequential Evolutionary BIclustering, was later proposed by Divina and Aguilar-Ruiz [21]. In this work, an EA is presented where the individuals also represent biclusters by means of binary strings. The main idea is that the EA is sequentially run several times. From each run, the EA yields the best bicluster according to its size, row variance and overlapping factors. If its MSR is lower than a given threshold, then the bicluster is added into an archive called Results. Whenever this is the case, the method keeps track of the elements of the bicluster so as to use this information to minimize overlapping during the next run of the EA. Tournament selection is chosen and several options for the recombination operators were implemented. The fitness function combines the aforementioned objectives by means of a non-Pareto aggregative function to be minimized as follows:

$$\text{F(B)} = \frac{MSR(B)}{\delta} + \frac{1}{rowVariance(B)} + w_d + penalty$$

where $w_d = w_v \left(w_r \frac{\delta}{|I|} + w_c \frac{\delta}{|J|} \right)$, having w_v, w_r and w_c as weights for the volume, number of rows and number of columns in the bicluster *B*, respectively.

Also, $penalty = \sum_{i \in I, j \in J} w_p(m_{ij})$, where $w_p(m_{ij})$ is a weight associated with each element m_{ij} of the bicluster and is defined as:

$$w_p(m_{ij}) = \begin{cases} 0 \; if \; |COV(m_{ij})| = 0 \\ \frac{\sum_{k \in I, l \in J} e^{-|COV(m_{kl})|}}{e^{-|COV(m_{ij})|}} \; if \; |COV(m_{ij})| > 0 \end{cases}$$

Here, $|COV(m_{ij})|$ denotes the number of biclusters containing m_{ij}. It is important to note that the weight $w_p(m_{ij})$ is used to control the overlapping level among the biclusters.

For the experimental studies, the EA was executed for two datasets: *Yeast* [18] and *Human B-cells* [22]. The comparison is performed against the biclusters found by Chung and Church as regards the covering of the whole gene expression matrix **E**. For the *Yeast* dataset, SEBI manages to cover 38% of **E**, while Chung and Church's covers 81%. Regarding the *Human* dataset, SEBI covers 34% while Chung and Church's biclusters cover 37%. The authors consider that these results can be explained as a consequence of the overlapping factor, since the consideration of this objective naturally goes in detriment of the other goals.

3.1.3. Mitra and Banka [23]

Later, Mitra and Banka [23] presented a MOEA combined with a LS [14] strategy. This method constitutes the first approach that implements a MOEA based on Pareto dominancy for this problem. The authors base their work on the NSGA-II, and look for biclusters with maximum size and homogeneity. The individual representation is the same as in the previously introduced methods; and uniform single-point crossover, single-bit mutation and crowded tournament selection are implemented. The LS strategy is applied to every individual under a *Lamarkian* approach, at the beginning of every generational loop.

The method is tested on microarray data consisting of two benchmark gene expression datasets, *Yeast* [18] and *Human B-cell Lymphoma* [22]. For the analysis of the results, a new measure called Coherence Index (CI) is introduced. CI is defined as "the ratio of mean squared residue (MSR) score to the size of the formed bicluster". The biclusters are compared to those reported by Chung and Church and, in all the cases, Mitra and Banka's results indicate a better performance in terms of the bicluster size, while satisfying the homogeneity criterion.

3.1.4. BiHEA [24]

Afterwards, Gallo *et al.* [24] implemented an EA combined with a LS technique based on Chung and Church's procedure, thus orienting the exploration and speeding up the convergence of the evolutionary algorithm by refining the chromosomes. The novelty of Gallo's method is that two additional mechanisms were incorporated in the evolutionary process in order to avoid the loss of good solutions: an elitism procedure that maintains the best biclusters as well as the diversity in the genotypic space through the generations, and a recovery process that extracts the best solutions of each generation and then copies these individuals into an archive. This archive is actually the set of biclusters returned by the algorithm. Although these two mechanisms appear to be similar to each other, there are several differences between them. The elitism procedure selects the best biclusters that do not overlap in a certain threshold, passing them to the next generation. These solutions can be part of the selection process of further generations thus allowing the production of new solutions by means of the recombination operator. However, due to imperfections in the selection process and in the fitness function, some good solutions can be misplaced through generations. To deal with this issue, the archive is incorporated as it keeps the best generated biclusters through the entire evolutionary process. It is important to remark that this "meta" population is not part of the selection process, i.e., the evolution of the population after each generation is monitored by the recovery process without interfering in the evolutionary process. As regards the fitness function, it optimizes the following objectives:

maximize

$$g(G,C)=|G||C|$$

$$k(G,C)=\frac{\sum_{g\in G,c\in C}\left(e_{gc}-e_{gC}\right)^2}{|G|\cdot|C|}$$

subject to

$$h(G,C)\leq\delta$$

with $(G,C)\in X$, $X=2^{\{1,\ldots,m\}}\times 2^{\{1,\ldots,n\}}$ being the set of all biclusters, where

$$h(G,C)=\frac{1}{|G|\cdot|C|}\sum_{g\in G,c\in C}\left(e_{gc}-e_{gC}-e_{Gc}+e_{GC}\right)^2$$

is the mean squared residue score (MSR), and

$$e_{gC}=\frac{1}{|C|}\sum_{c\in C}e_{gc},\quad e_{Gc}=\frac{1}{|G|}\sum_{g\in G}e_{gc}$$

are the mean column and row expression values of (G,C), and

$$e_{GC}=\frac{1}{|G|\cdot|C|}\sum_{g\in G,c\in C}e_{gc}$$

is the mean expression value in all the cells that are contained in the bicluster (G,C). The user-defined threshold $\delta>0$ represents the maximum allowable dissimilarity within the cells of a bicluster. In other words, the residue quantifies the difference between the actual value of an element and its expected value as predicted for the corresponding row mean, column mean, and bicluster mean.

As for the experimentation, in a first phase on synthetic datasets, the results obtained from this method outperform the outcomes of several biclustering approaches of the literature, especially in the case of coherent biclusters with high overlap degrees. Nonetheless, this should not be considered as a drawback because, in general, the regulatory complexity of an organism is far from the model of non-overlapped biclusters. Furthermore, an analysis of a real dataset (colon cancer data [41]) was performed and, in terms of measure in the paper, the quality of the outcomes of BiHEA is clearly better than the results of the reference methods. In fact, this shows the correctness of the model designed to build the biclusters, i.e., coherent biclusters following an additive model.

3.1.5. GABI [25]

Another evolutionary approach, called GABI (GA based Biclustering), was introduced in Mukhopadhyay *et al.* [25]. The main difference with the rest of the algorithms lays in the representation of the biclusters. Here, each string has two parts, one for clustering the genes and the other one for clustering the conditions. As in the other methods, the fitness function uses the MSR. In this case, the calculus is performed in the following manner:

$$\mathrm{F(B)}=\frac{\mathrm{MSR(B)}}{\delta.(1+VAR(B))}$$

GABI was also tested with the same datasets as the ones used in the other methods: *Yeast* [18] and *Lymphoma* [22]. Regarding the *Yeast* dataset, the method was able to discover interesting biclusters with high row variance. It is important to remark that the authors preferred the achievement of non-trivial and highly coherent biclusters instead of biclusters with high volume. As for the second dataset, the results were compared to those of Cheng and Church's algorithm [14] and the RWB (Random Walk Biclustering) algorithm [26]. According to the experimentation, GABI provides better average row variance and average MSR in comparison to the aforementioned methods. In addition, a biological significance test based on Gene Ontology was performed for one of the biclusters, showing that the method is also able to identify biological significant biclusters.

3.1.6. PCOBA [27]

Recently, Joung *et al.* [27] presented a new probabilistic evolutionary algorithm, called PCOBA (Probabilistic Coevolutionary Biclustering Algorithm). The novelty of this method consists in the use of the global statistical information of two cooperative populations, so that the ability to search biclusters is more effective. The main idea is that the strategy coevolves the two populations of biclusters for a gene set and a condition set, as one is adapted cooperatively to the other. The fitness function aims at minimizing the MSR, whilst maximizing the variance and the volume of the bicluster. Also, the fitness of an individual is determined by the degree of cooperation between the selected one and the individuals of the other population. For the experiments, synthetic datasets and *Yeast* [18] were used. The comparison was made with a general GA and two other implementations called CGA (Coevolutionary Genetic Algorithm) [28] and EDA (Estimation of the Distribution Algorithm) [29], obtaining coherent biclusters with higher quality as regards all of the objectives.

3.2. Virtual Error-Based Method

As it can be seen, all the aforementioned methods use MSR as the main measure of the objective function. However, it was also said that even though MSR allows the finding of interesting biclusters, some others are left behind, such as those exhibiting scaling patterns. As a result, the virtual error (VE) arises, which aims at finding both types of patterns (shifting and scaling patterns).

3.2.1. Pontes et al. [15]

In this regard, in Pontes *et al.* [15] a novel method that improves the performance of the SEBI by varying the fitness function is presented. The strategy implements the objective function using the virtual error metric as follows:

$$\mathrm{F}(B) = \mathrm{VE}(B) + \mathrm{w_d} + penalty$$

In this case, w_d and the penalty are the ones defined before by the SEBI algorithm. This fitness function also has to be minimized. The rest of the algorithm is implemented similarly.

In order to demonstrate the quality of the new approach, two well-known datasets were used: *Yeast* and *human B-cells*. In general terms, the authors show that the new implementation could find very interesting biclusters with special shapes that were difficult to find using the MSR. They also prove that the new version finds biclusters which the algorithm that uses MSR would have rejected. It is also important to note that VE is not sensitive to the scale or magnitude difference in the expression values of the genes, as long as they present the same behavior.

3.3. Transposed Virtual Error-Based Method

Still, even though VE succeeds in finding biclusters that present shifting or scaling patterns, it is not able to recognize both patterns at the same time. Then, as it was aforementioned, $\mathrm{VE^t}$ arises as a new metric that properly identifies those features all together.

3.3.1. Pontes et al. [30]

The most innovative evolutionary algorithm using this measure in the fitness function is described in Pontes *et al.* [30]. In this article the authors present the Evo-Bexpa (Evolutionary Biclustering based in Expression Patterns) that constitutes the first biclustering method in which several biclusters features can be particularized in terms of different objectives, and it is also possible to find biclusters presenting both patterns simultaneously.

Four different objectives were individualized in this approach, attending to the extent to which a bicluster follows a perfect correlation pattern, to its size, to the overlapping level among different solutions and to the mean gene variance. The objectives are considered by means of constructing an aggregative objective function. Then, it is possible to specify the relative influence of each one during the evaluation process, thus allowing the algorithm to be configurable. Concerning the first objective, the $\mathrm{VE^t}$ is calculated as it was explained in section 2. In the next lines, the other three terms will be described.

As regards the bicluster volume, it is defined as follows:

$$\mathrm{Vol(B)} = \left(\frac{-\ln(|I|)}{\ln(|I|)+w_g}\right)+\left(\frac{-\ln(|J|)}{\ln(|J|)+w_c}\right)$$

where $|I|$, $|J|$, $\mathrm{w_g}$ and $\mathrm{w_c}$ are the number of genes, number of conditions, and configurable parameters for both dimensions, respectively. The main idea of this equation is that it uses logarithmic scales so that little changes in the number of rows or columns do not have a significant effect, and it separates the terms for the number of genes and conditions in order to avoid too unbalanced biclusters and to be able to configure each dimension size independently.

Overlapping is controlled with the next term:

$$\mathrm{Overlap(B)} = \frac{\sum_{i\in I, j\in J} W(b_{ij})}{|I|.|J|.(n_b-1)}$$

Here, W is a matrix of weights (similar to the one presented in [31]), whose size is the same as the size of the microarray, initialized with zero values at the beginning of the algorithm. Every time a bicluster is found, W is updated increasing by 1 those elements that are contained in the bicluster. I and J refer to the sets of rows and columns in B, respectively, and $W(b_{\mathrm{ij}})$ corresponds to the weight of b_{ij} in W. Also, n_{b} is the order of the solution bicluster. Broadly, this term computes how many times the elements of B have appeared in any former biclusters, and divides this value by the size of B and the order of the solution.

As for the gene variance, it is generally used to avoid trivial biclusters, preferring those solutions in which genes exhibit high fluctuating trends. In accordance with this idea, the corresponding term is designed as follows:

$$\mathrm{GeneVar(B)} = \frac{1}{|I|.|J|}\sum_{i=1}^{I}\sum_{j=1}^{J}(b_{ij}-\mu_{g_i})^2$$

As it can be observed, gene variance of a bicluster is given by the mean of the variances of all the genes in it. Having defined all the terms that are aggregated by means of a single objective function, the remaining general features of the algorithm will be described.

In this approach, a sequential covering strategy is followed, where a single bicluster is obtained each time the algorithm is executed. Then, if *n* biclusters are desired, the evolutionary algorithm has to be run *n* times. Experimentation was conducted over both synthetic and real data sets. Results were compared with those obtained using OPSM [32], ISA [33, 34], xMotifs [35] and Bimax [36]; all

Table 1. Summary of the revised methods' main features.

	Metrics for the Fitness Function						Multi-Objective Approach		Hibridized with LS	Bicluster Type		Additional Considerations
	Size	rowVAR	MSR	VE	VE^T	Overlap	Agregative	Pareto		Shifting Pattern	Scaling Pattern	
Bleuler *et al.* [16]												
EA alone	X		X				X			X		
Lamarckian	X		X				X		X	X		
Baldwinian	X		X				X		X	X		
SEBI [20]	X	X	X			X	X			X		Sequential covering strategy
Mitra and Banka [22]	X		X					X	X	X		
BiHEA [23]	X	X	X				X		X	X		Elitism to avoid overlaping
GABI [24]		X	X				X			X		
PCOBA [26]	X	X	X				X			X		Two cooperative populations
Pontes *et al.* [14]	X			X		X	X			X	X	Sequential covering strategy
Evo-Bexpa [29]	X	X			X	X	X			X	X	Sequential covering strategy

of them were executed using the BicAT (Biclustering Analysis Toolbox) [37]. The new technique outperformed all the other methods regarding the match scores' indexes that were used on synthetic data. As for the real datasets, *Yeast* [18], *Embryonal* [38], *Leukemia* [39] and *Steminal* [40] were analyzed. Some troublesomeness was found when trying to apply the BicAT with certain methods to some of these datasets. For example, xMotifs could not be tested for *Yeast* and *Steminal* datasets, due to unexpected runtime errors. xMotifs could not be executed for *Leukemia* neither, as it does not support more than 64 samples. Only OPSM, CC and Evo-Bexpa were able to produce results for all of the datasets. As a general result, it was proven that VE^t values for the biclusters found by Evo-Bexpa are smaller than 0.1 for all of the datasets, whereas no other algorithm finds biclusters with such a low level of VE^t. Moreover, with the new method it is possible to adjust the result characteristics to user defined parameters. Regarding the biological validation of the results, they were validated using different levels in the Gene Ontology hierarchy, exhibiting that significant biclusters were obtained by Evo-Bexpa which correspond to neither too general nor specific GO terms. Therefore, it can be concluded that Evo-Bexpa succeeds in finding biclusters whose significant terms have an intermediate level of specificity.

In Table **1**, an overview of all the methods that were described in this section is presented. The table summarizes information about the metrics that are combined in the fitness function, and it also reviews whether a Pareto-based approach is used or not, when the method is hybridized with a local search strategy, what types of biclusters are found by each approach, and some other distinctive considerations for the algorithms.

CONCLUSION

Microarray technology arose in the last decades as a tool to provide information about the behavior of thousands of genes. The data corresponds to the relative amount of the mRNA of a gene under a given experimental condition, which can be associated to the expression level of the gene. This information is arranged into the gene expression data matrix, where rows and columns correspond to genes and conditions respectively. Each matrix entry is a real number that represents the expression level of a given gene under a given condition. Several methods have been increasingly arisen which analyze this matrix in order to obtain significant biological information.

In general, those methods intend to recognize relations among some genes under certain conditions, thus building the so-called biclusters. There are great amounts of strategies that aim at finding biclusters in gene expression data matrices. In this work we have focused on describing the main existing measures for defining the quality of a bicluster. We have also presented the most representative evolutionary techniques that use those measures in the objective function.

As it can be observed from the examples found in the literature, MSR (Mean Squared Residue) remains as the most commonly used measure. However, as for the trends that bicluster values follow, it is known that this measure is only able to identify shifting patterns. In that context, the VE (Virtual Error) that also detects patterns of scaling arises. However, although the behavior of VE ensures its value is near 0 for biclusters with shifting and scaling patterns, it has not been demonstrated analytically that VE allows the recognition of both patterns simultaneously. Then, as an improvement, the VE^t (transposed Virtual Error) is presented, which allows finding both patterns simultaneously after carrying out several transformations to the data. The most recent work which bases the fitness function on the VE^t quality measure is the Evo-Bexpa. This method not only uses the most up-to-date bicluster quality measurement, but also allows the user to particularize the preferred objectives in terms of other bicluster features. Regarding the experiments, meaningful biclusters were found by the Evo-Bexpa along both synthetic and real datasets. As for real datasets, the method also succeeds in the quality of the results regarding the biological significance of the biclusters, according to GO terms.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Authors acknowledge CONICET for Grant PIP N° 112-2009-0100322, the ANPCyT for Grant N°11-1652 and SeCyT (UNS) for Grants PGI 24/N032 and PGI 24/ZN26.

REFERENCES

[1] Lesk A. Introduction to Bioinformatics. 1st ed. Oxford University Press 2008.

[2] Baldi P, Hatfield GW. DNA Microarrays and Gene Expression: From Experiments to Data Analysis and Modeling. 1st ed. Cambridge University Press 2002.

[3] Asyali MH, Colak D, Demirakaya O, Inan MS. Gene expression profile classification: a review. Curr Bioinform 2006; 1: 55-73.

[4] Buness A, Ruschahaupt M, Kuner R, Tresch A. Classification across gene expression microarray studies. BMC Bioinformatics 2009; 10: 453.

[5] Gallo CA, Carballido JA, Ponzoni I. Microarray Biclustering: A Novel Memetic Approach Based on the PISA Platform. Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics Lecture Notes in Computer Science 2009; 5483: 44-55.

[6] Madeira S, Oliveira, AL. Biclustering Algorithms for Biological Data Analysis: A Survey. IEEE ACM Trans Comput Biol Bioinform 2004; 1: 24-45.

[7] Tanay A, Sharan R, Shamir R. Biclustering algorithms: a survey. Handb Comput Mol Biol 2004; 9: 1-26.

[8] Busygin S, Prokopyev OA, Pardalos PM. Biclustering in data mining. Comput OR 2008; 35(9): 2964-87.

[9] Mukhopadhyay A, Maulik U, Bandyopadhyay S. On Biclustering of Gene Expression Data. Curr Bioinform 2010; 5: 204-16.

[10] de Jong KA. Evolutionary Computation: A unified approach. Massachusetts Institute of Technology: Massachusetts 2006.

[11] Irma International. Bioinformatics: Concepts, Methodologies, Tools, and Applications (Hardback). Medical Information Science Reference. Hershey 2013.

[12] Aguilar-Ruiz JS. Shifting and scaling patterns from gene expression data. Bioinformatics 2005; 21: 3840-45.

[13] Zhao H, Liew AWC, Wang DZ, Yan H. Biclustering Analysis for Pattern Discovery: Current Techniques, Comparative Studies and Applications. Curr Bioinform 2012; 7(1): 43-55.

[14] Cheng Y, Church, GM. Biclustering of Expression Data. Proc Int Conf Intell Syst Mol Biol 2000; 8: 93-103

[15] Pontes B, Divina F, Giráldez R, Aguilar-Ruiz JS. Virtual Error: A New Measure for Evolutionary Biclustering. Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics Lecture Notes in Computer Science 2007; 4447: 217-26.

[16] Pontes B, Divina F, Giráldez R, Aguilar-Ruiz JS. Measuring the Quality of Shifting and Scaling Patterns in Biclusters. Pattern Recognition in Bioinformatics Lecture Notes in Computer Science 2010; 6282: 242-52.

[17] Bleuler S, Prelic A, Zitzler, E. An EA framework for biclustering of gene expression data. In: Proceedings of the 2004 Congress on Evolutionary Computation 2004; 1: 166-73.

[18] Cho R, Campbell M, Winzeler E, *et al*. A genome-wide transcriptional analysis of the mitotic cell cycle. Mol Cell 1998; 2(1): 65-73.

[19] Menges M, Hennig L, Gruissem W, Murray J. Genome-wide gene expression in an Arabidopsis cell suspension. Plant Mol Biol 2003; 53(4): 423-42.

[20] Laule O, Fürholz A, Chang HS, *et al*. Crosstalk between cytosolic and plastidial pathways of isoprenoid bio systhesis in arabidopsis thaliana. PNAS 2003; 100(11): 6866-71.

[21] Divina F, Aguilar-Ruiz JS. Biclustering of Expression Data with Evolutionary Computation. IEEE Trans Knowl Data Eng 2006; 18(5): 590-602.

[22] Alizadeh AA. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 2000; 403(6769): 503-11.

[23] Mitra S, Banka H. Multi-objective evolutionary biclustering of gene expression data. Pattern Recognit 2006; 39(12): 2464-77.

[24] Gallo CA, Carballido JA, Ponzoni I. BiHEA: A Hybrid Evolutionary Approach for Microarray Biclustering. Lecture Notes in Computer Science 2009; 5676: 36-47.

[25] Mukhopadhyay A, Maulik U, Bandyopadhyay S. Evolving coherent and non-trivial biclusters from gene expression data: An evolutionary approach. Proc IEEE Region 10 Conf 2008.

[26] Angiulli F, Cesario E, Pizzuti C. Gene expression biclustering using random walk strategies. In: Proc 7th Int Conf on Data Warehousing and Knowledge Discovery Copenhagen, Denmark 2005. Berlin Heidelberg: Springer 2005; 509-19.

[27] Joung JG, Kim SJ, Shin SY, Zhang BT. A probabilistic coevolutionary biclustering algorithm for discovering coherent patterns in gene expression dataset. BMC Bioinformatics 2012; 13(17): S12.

[28] Hills DW. Co-evolving parasites improve simulated evolutions in an optimization procedure. Physica D: Nonlinear Phenomena 1990; 42(1): 228-34.

[29] Pelikan M, Goldberg DE, Lobo F. A survey of optimization for building and using probabilistic models. Comput Optim Appl 2002; 21(1): 5-20.

[30] Pontes B, Divina F, Giráldez R, Aguilar-Ruiz JS. Configurable pattern-based evolutionary biclustering of gene expression data. Algorithm Mol Biol 2013; 8: 4.

[31] Gasch AP, Eisen MB. Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. Genome Biol 2002; 3(11): Research 0059.

[32] Ben-Dor A, Chor B, Karp RM, Yakhini Z. Discovering local structure in gene expression data: the order-preserving submatrix problem. J Comput Biol 2003; 10(3-4): 373-84.

[33] Bergmann S, Ihmels J, Barkai N. Iterative signature algorithm for the analysis of large–scale gene expression data. Phys Rev E 2003; 67(3): 031902.

[34] Ihmels J, Bergmann S, Barkai N. Defining transcription modules using large-scale gene expression data. Bioinformatics 2004; 20(13): 1993-2003.

[35] Murali TM, Kasif S. Extracting conserved gene expression motifs from gene expression data. Pac Symp Biocomput 2003: 77-88.

[36] Prelić A, Bleuler S, Zimmermann P, *et al*. A systematic comparison and evaluation of biclustering methods for gene expression data. Bioinformatics 2006; 22(9): 1122-29.

[37] Barkow S, Bleuler S, Prelić A, Zimmermann P, Zitzler E. BicAT: a biclustering analysis toolbox. Bioinformatics 2006; 22(10): 1282-83.

[38] Pomeroy SL, Tamayo P, Gaasenbeek M, *et al*. Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature 2002; 415(6870): 436-42.

[39] Golub TR, Slonim DK, Tamayo P, *et al*. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 1999; 286(5439): 531–37.

[40] Boyer LA, Plath K, Zeitlinger J, *et al*. Polycomb complexes repress developmental regulators in murine embryonic stem cells. Nature 2006; 441(7091): 349–53.

[41] Alon U, Barkai N, Notterman DA, *et al*. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc Natl Acad Sci USA 1999; 96(12): 6745-50.

Received: December 9, 2013 Revised: June 4, 2014 Accepted: August 8, 2014