LSL: A new measure to evaluate triclusters

David Gutiérrez-Avilés
Department of Computer Science, University of Seville
Seville, Spain
davgutavi@alum.us.es

Cristina Rubio-Escudero
Department of Computer Science, University of Seville
Seville, Spain
crubioescudero@us.es

Abstract—**Microarray technology has led to a great advance in biological studies due to its ability to monitorize the RNA levels of a vast amount of genes under certain experimental conditions. The use of computational techniques to mine hidden knowledge from these data is of great interest in research fields such as Data Mining and Bioinformatics. Finding patterns of genetic behavior not only taking into account the experimental conditions but also the time condition is a very challenging task nowadays. Clustering, biclustering and novel triclustering techniques offer a very suitable framework to solve the suggested problem. In this work we present LSL, a measure to evaluate the quality of triclusters found in 3D data.**

Keywords—triclustering, least square line, genetic algorithms, microarray data, behavior patterns

I. Introduction

Microarray technology is highly used in biological research environments due to its capacity to monitor, for a great gene collection, the RNA concentration levels, thus enabling the study of genetic functions of species under study [1]. Bioinformatics and Data Mining have developed a vast amount of computational tools that allows us to analyze data obtained using this technology and to find new knowledge which is hidden for the human's eyesight [2], [3]. One of the most useful and studied approaches is the behavior pattern search in gene expression data. These genes, that exhibit high correlation among their expression levels, could be involved in similar regulatory processes [4]. The relationship between correlation and functionality has been proved in several studies as in [5].

There are many behavior pattern searching techniques, and in this work we focus on the clustering-based ones, which analyze the whole microarray dimensional space grouping genes taking into account all experimental conditions [6]. Classical clustering techniques group genes based on all conditions. Biclustering emerges as an evolution of clustering since it groups genes under some particular experimental conditions [7]. Another evolution is triclustering, that goes one step further by grouping genes under particular conditions and under particular time points [8], thus being capable of managing 3D data. Therefore, it processes a type of microarray experiment where several samples are taken at different time points [9], which has great interest since it allows for a deep analysis of biological processes where temporary development is important.

Both biclustering and triclustering attack NP-hard problems [10], therefore algorithm based on heuristics are well suited to manage this problem. In [11] we present the TriGen algorithm, a triclustering-genetic algorithm based on an evolutionary heuristic, genetic algorithms, which finds pattern of similarity for genes on a three dimensional space, thus taking into account the gene,conditions and time factors.

In this sense, defining an appropriate quality measure for the triclusters is an important and essential challenge to solve the problem [12]. In biclustering, a classic measure is the Mean Squared Residue [13]. A three dimensions adaptation of this measure, MSR_{3D}, has been defined in [14].

In this work we present the LSL measure for triclusters, which measures the quality of a tricluster based on the similarity among the slopes of the angles formed by the least squares lines from each of the profiles formed by the genes, conditions and times of the tricluster. LSL has obtained better results than MSR_{3D} applied to the same datasets along with the TriGen algorithm.

As related work, in [15] we can see one of the first approaches of triclustering in which the authors presented a measure to assess triclusters quality based on the symmetry property, this allows a very efficient cluster mining since clusters are searched over the dimensions with the least cardinality. This proposal was improved in [16] wherein the authors give an extended and generalized version of previous proposal in wich they claimed that the symmetry property is not suitable for all patterns present in biological data and propose the Spearman rank correlation [17] as a more appropriate tricluster evaluation measure. LagMiner was introduced in [18] to find time-lagged 3D clusters, which allows in turn finding regulatory relationships among genes. They evaluated their triclusters on homogeneity,regulation,minimum gene number, sample subspace size, and time periods length. A new strategy to mine 3D clusters in real-valued data was introduced in [19] in wich the authors defined the Correlated 3D Subspace Clusters (CSCs) where the values in each cluster must have high cooccurrences and those cooccurrences are not by chance. They measure the clusters based on the correlation information measure, which takes into account both prerequisites. One of the most recent approaches can be studied in [20].

II. Methodology

In this section we introduce the triclustering concept as well as its graphics views, which are key to present our proposal LSL.

A. Triclustering

Triclustering emerges as an evolution of biclustering taking into account temporary evolution of genes under particular experimental conditions. In this way, from a dataset D which contains genes, conditions and time points, data are organized in file-column-depth form. We define triclustering as a technique that finds groups of triclusters $TRI_1, ..., TRI_n$ from D. A tricluster is a subset of genes, conditions and time points extracted from D having gene expression levels highly correlated. Formally, we can define a tricluster as $TRI = G \times C \times T$ where $G \subseteq G_D$, $C \subseteq C_D$ and $T \subseteq T_D$ [11].

B. Graphic Views

To visually analyze the behavior patterns that a tricluster TRI traces we depict the graphic views from it. A graphic view TRI_{xop} of TRI is the graphical representation of the tricluster so that the x coordinates will be on X axis and o outlines will be represented on as many panels as p indicates, as can be seen in Figure 1. For legibility reasons the three graphical views will always be considered:

- TRI_{gct} $(x = g, o = c, p = t)$: one panel for each time, genes on the X axis, the expression levels on the Y axis and the lines of conditions as the outline.
- TRI_{gtc} $(x = g, o = t, p = c)$: one panel for each condition, genes on the X axis, the expression levels on the Y axis and the time lines as the outline.
- TRI_{tgc} $(x = t, o = g, p = c)$: one panel for each condition, times on the X axis, the expression levels on the Y axis and the genes as the outline.

The graphical views of a tricluster are key to explain our proposal.

C. LSL measure

Our goal in this proposal is providing a quality measure for a tricluster. In general terms, LSL measures the differences between least squared lines slope angles of every series traced on each of the three graphical views of TRI. In Figure 2 we can observe an example of TRI_{gct} view of tricluster $TRI = G \subset \{g_1, g_4, g_7, g_{10}\}, C \subset \{c_2, c_5, c_8\}, T \subset \{t_0, t_2, t_{11}\}$ in which the dashed lines are the least squared lines, α_{rs}, β_{rs} and γ_{rs} with $r \, \epsilon \, C$, $s \, \epsilon \, T$ correspond to least squared lines slope angles for each series c_2, c_5 y c_8 respectivily.

To obtain LSL, we first perform the angular comparison term calculation. The angular comparison operation of a graphic view xop from a tricluster TRI is defined in Equation 1a.

$$AC_{lsl}(TRI_{xop}) = \frac{V_{cmp} + H_{cmp}}{N_{cmp}} \tag{1a}$$

$$\begin{split} ang = \{\alpha_{o_1p_1}, \alpha_{o_2p_1}, \alpha_{o_3p_1}, \ldots, \\ \alpha_{o_1p_2}, \alpha_{o_2p_2}, \ldots, \alpha_{OUTpAN}\} \end{split} \tag{1b}$$

$$V_{cmp} = \sum_{ang} \Delta(\alpha_{op}, \alpha_{next(o)p}) \tag{1c}$$

$$H_{cmp} = \sum_{ang} \Delta(\alpha_{op}, \alpha_{onext(p)}) \tag{1d}$$

$$N_{cmp} = \frac{|o| * |p| * (|o| + |p| - 2)}{2} \tag{1e}$$

$$\Delta(\alpha_A, \alpha_B) = MAX(\alpha_A, \alpha_B) - MIN(\alpha_A, \alpha_B) \tag{1f}$$

We define AC_{lsl} of a tricluster's graphic view TRI_{xop} as the average of differences of least squared slopes angles for every outline o for each panel p (V_{cmp}) and its equivalent for the rest of the panels (H_{cmp}) with the total number of differences being N_{cmp}. Taking into account least squared slope angles order as we can see in 1b, that is, first by panel p_j and after by outline o_i we define V_{cmp} (Equation 1c) and H_{cmp} (Equation 1d) elements as addition of differences Δ between all of angles from same panel and addition of differences Δ of same angle from different panels respectively. We can observe in Equation 1e the calculate of the number of operations for each view. The operation Δ (Equation 1f) to α_A and α_B angles is defined as the difference between maximal and minimal of these angles.

We calculate each angle of outline o for panel p (α_{op}) based on concept of series (Equation 2a). A series S_{op} of a outline o for a panel p is a set of pair of values from the x axis (x_i) and expression levels (el_j) that form the outline. For each series S_{op} the alpha angle $alpha_{op}$ is calculated as the $spin$ of the arctangent of the least squared slope that best adjusts this series (Equation 2b). $Spin$ operation of an angle showed in 2c is the positive equivalent of this angle if it is negative.

Finally, LSL measure of a tricluster TRI (Equation 3) is the average of the angular comparison of the three graphic views of this tricluster.

$$S_{op} = \{< x_0, el_0 >, \ldots, < x_{AX}, el_L >\} \tag{2a}$$

$$\alpha_{op} = spin\left[\arctan\left\{\frac{|S_{op}|(\sum x_i el_i) - (\sum x_i)(\sum el_i)}{|S_{op}|(\sum x_i^2) - (\sum x_i)^2}\right\}\right] \tag{2b}$$

$$spin(\alpha) = if \alpha < 0 \Rightarrow \alpha = \alpha + 2 * \pi \tag{2c}$$

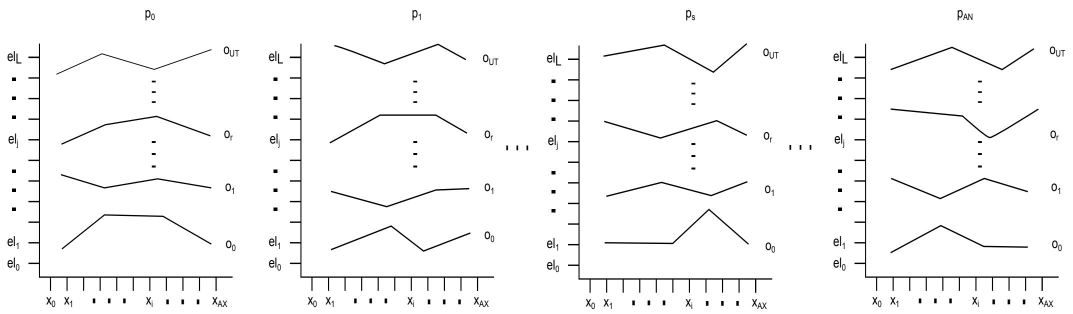


Fig. 1: Graphic view for a tricluster

$$LSL(TRI) = \frac{1}{3}\,[\,AC_{lsl}(TRI_{gct}) + AC_{lsl}(TRI_{gtc}) + AC_{lsl}(TRI_{tgc})\,] \tag{3}$$

D. TriGen Algorithm

The LSL measure has been applied along with the TriGen algorithm presented in [11]. This algorithm is based on the bio-inspirated paradigm called genetic algorithms, and performs a search for a set of triclusters which are selected based on the LSL quality measure. The algorithm creates an initial population where each individual represents a tricluster, and the population evolves modifying the individuals with selection and mutation operators. The outline of the algorithm can be seen in Figure 3.

We now define the most important elements of the algorithm such as inputs, outputs, codification of individuals and genetic operators.

1) Input: The TriGen algoritm has two input arguments:

- D: A dataset containing the gene expression values from an experiment containing genes G, experimental conditions C and times T. Therefore, each cell $[i, j, k]$ from D where $i \in G$, $j \in C$ and $k \in T$, represents the expression level of the gene i under the experimental condition j at time k.
- P: Set of parameters to execute the algorithm, these are $\{N$, G, I, Ale, Sel, Mut, w_g, w_c, w_t, wo_g, wo_c, $wo_t\}$. They control the number of solutions or triclusters to find (N), the number of generations to execute (G), the number of individuals in the population (I) and the randomness factor they are generated with in the initial population (Ale) as well as weights for the selection and mutation operators (Sel and Mut), weights to control the size of the triclusters (w_g, w_c, w_t) and weights to control the overlap among solutions (wo_g, wo_c, wo_t)

2) Output: The *TriGen* algorithm's output will be a set of N triclusters, formally $Sol = \{TRI_1, TRI_2, \ldots, TRI_N\}$. Each $TRI_i \,\epsilon\, Sol$ has the best score in is population when evaluated under the LSL measure.

3) Codification: Each individual in the evolutionary process of the *TriGen* algorithm represents a tricluster which is a potential solution. Therefore, an individual is represented as subset of genes $G =< g_{i_1}, g_{i_2}, \ldots, g_{i_F} >$, a subset of conditions $C =< c_{i_1}, c_{i_2}, \ldots, c_{i_Q} >$ and a subset of time points $T =< t_{i_1}, t_{i_2}, \ldots, t_{i_W} >$. The genes, conditions and times subsets are extracted from the input dataset D. All genetic operators are applied to each individual in the population, in each of these three subsets.

4) Overlapping among solutions: To avoid overlapping among found tricluster solutions, we have designed an overlapping control mechanism in which we maintain the number of occurrences of genes, conditions and time points of dataset D in each tricluster solution. As we will explain in Section II-D5a, the initial population operator uses these structures to initialize population as less overlapped as possible. As we can see in Figure 3, this overlapping mechanism is updated every time that a new tricluster solution is selected.

5) Genetic Operators:

a) Initial Population: In the initial population operator, a percentage of individuals defined by the Ale parameter are created at random and the rest are created taking into account the overlapping mechanism.

b) Fitness: The fitness function is the proposed quality method LSL along with S_{ctrl} and O_{ctrl} factors as we can see in Equation 4. S_{ctrl} factor controls the size of the individuals and depends directly on w_g, w_c and w_t control parameters so that an increase of any weight parameters implies that individuals with a greatest number of elements in the specific subset obtain a better score and therefore have a higher probability of contributing to the next generation. For example, if we increase w_g, the algorithm considers that solutions with a high number of genes are better than those with low number of genes. O_{ctrl} factor controls the overlapping level of individuals and is directly related to wo_g, wo_c and wo_t control parameters so that if we increase. For example, if we increase wo_g, the algorithm considers that solutions with low level of overlap with the genes in previously found solutions are better than those with a high level of overlap.

$$FF(TRI) = LSL(TRI) - S_{ctrl}(TRI) - O_{ctrl}(TRI) \tag{4}$$

c) Selection: This operator is implemented following the roulette wheel selection method [21]. The fitness level is used to associate a probability of selection with each individual of the population. This emulates the behavior of a roulette wheel in a casino. Usually a proportion of the wheel is

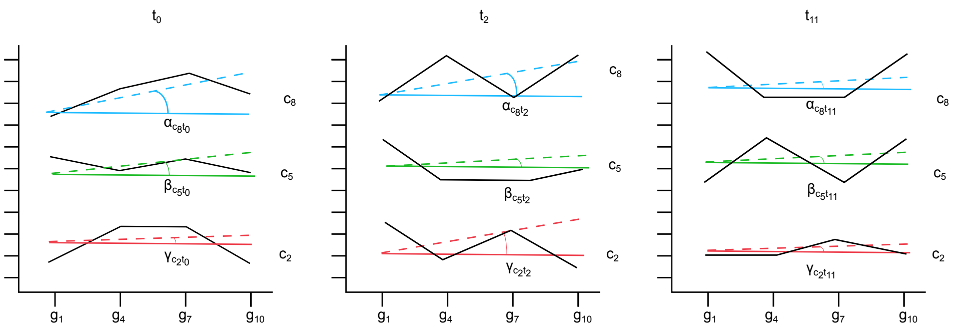


Fig. 2: Angles for TRI_{gct} graphic view

assigned to each of the possible selections based on their fitness value. Then a random selection is made similar to how the roulette wheel is rotated. While candidates with a higher fitness will be less likely to be eliminated, there is still a chance that they might be. There is a chance that some weaker solutions may survive the selection process, which is an advantage, as though a solution may be weak, it may include some component which could prove useful following the recombination process. The Sel parameter indicates how many individuals will pass to the next generation undergoing this method. The rest of the individuals up to complete the next population (I - $\#Selected\ individuals$) will be created based on the crossover operator.

d) Crossover: To complete the next generation, we create new individuals with this operator as follows: two individuals (parents, A and B) are combined to create two new individuals (offsprings, $child1$ and $child2$). The parents are randomly chosen. Their genetic materials are combined by a random one-point cross in the genes G_g, conditions C_c and times T_t and mixing the coordinates in both children.

e) Mutation: An individual can be mutated according to a probability of mutation, Mut. The mutation probability is verified for every individual and if it is satisfied, one out of six possible actions is taken. These actions are: add a new random gene to G_g in TRI, add a new condition to C_c in TRI or add a new time to T_t in TRI, or by removing a random gene, condition or time. The election of these actions is also random. For the case of addition of a new gene, condition or time, the operator checks whether the new member is already in the individual or not.

III. Results

In this section we show the results obtained applying the TriGen algorithm with the LSL measure both to real and synthetic data. We analyze two real datasets, one of them has been acquired from experiments with the yeast cell cycle (Saccharomyces cerevisiae) obtained from the Stanford University [22], and the other one is an mouse experiment [9] that has been retrieved from Gene Expression Omnibus [23], a database repository of high throughput gene expression data.

To examine the quality of the results in experiments with real datasets, we show for each experiment two types of validity measures: analysis of correlation among the genes, conditions, and times in each tricluster and analysis of genes and gene product annotations for the genes in each tricluster based on the Gene Ontology project [24].

```
Input: D, P
Output: Sol

Sol = {}
Repeat N times:
  Pop = Initial Population(Ale)
  Repeat G times:
    Evaluation(Pop,Wg,Wc,Wt,WOg,WOc,WOt)
    Pop = Selection(Pop,Sel)
    Pop = Crossover(Pop)
    Pop = Mutation(Pop,Mut)
  End Repeat
  Sol = Sol + Best(Pop)
  Update Overlapping Mechanism y (Best(Pop))
End Repeat
```

Fig. 3: *TriGen* algorithm

Regarding the correlation analysis, we show a table for each tricluster (in rows) in which we calculate the Pearson and Filon [25] and Spearman [17] correlation coefficients between each combination of condition time and the values series are the expression levels of all genes in the corresponding condition-time combination. For example, for a tricluster with ten genes {1, ..., 10}, three conditions {1, 3 and 5}, and two times {2 and 7}, we provide the Pearsons and Spearmans correlation coefficient for values at the six possible combinations $V_{c=1,t=2}$, $V_{c=1,t=7}$, $V_{c=3,t=2}$, $V_{c=3,t=7}$, $V_{c=5,t=2}$ and $V_{c=5,t=7}$ for each of the ten genes.

In the biological analysis we provide a validation of the triclusters obtained based on the Gene Ontology project (GO) [24]. GO is a major bioinformatics initiative with the aim of standardizing the representation of gene and gene product attributes across species and databases. The project provides an ontology of terms for describing gene product characteristics and gene product annotation data. The ontology covers three domains: cellular component, the parts of a cell or its extracellular environment; molecular function, the elemental activities of a gene product at the molecular level, such as binding or catalysis; and biological process, operations or sets of molecular events with a defined beginning and end, pertinent to the functioning of integrated living units: cells, tissues, organs, and organisms. For legibility reasons, we have presented for one solution of the experiment a GO analysis table in which we include the most representative terms extracted by the Ontologizer software [26]. We have also provided a graphical representation of the triclusters found. For legibility reasons we show graphs for one tricluster for each of the experiments. Each tricluster is represented through three graphical views TRI_{gct}, TRI_{gtc} and TRI_{tgc} described in Section 1 in which we can see the pattern of behavior.

We have compared these results with those obtained using in the measure MSR_{3D} in [14] in terms of correlation, and Gene Ontology analysis in order to emphasize the performance of LSL against MSR_{3D} as well. For each real experiment we have compared the maximum, minimum and mean Pearson's and Spearman's correlation index and the maximum, minimum and mean *p*-value for each solution considered.

A. Synthetic Dataset

Synthetic data are widely used not only for testing the performance of microarray analyzing techniques [9] but also in more general data mining publications [27]. In this work we have executed the TriGen algorithm with the LSL measure over a synthetic dataset composed by 4000 genes, 30 experimental conditions and 20 time points whose expression levels were randomly generated. In this dataset we inserted 10 triclusters composed by 150 genes, 6 experimental conditions and 4 time points whose expression levels form a constant behavior pattern. These triclusters are located in random positions in the dataset. We have varied control parameters of the TriGen algorithm as follows: $N\,\epsilon\,\{200, 300, 50\}$, $G\,\epsilon\,\{50, 100, 1500\}$, $I\,\epsilon\,\{200, 500, 200\}$, $Sel\,\epsilon\,\{0.8, 0.9, 0.4\}$, $Mut\,\epsilon\,\{0.1, 0.1, 0.1\}$, $Ale\,\epsilon\,\{0.9, 0.5, 0.5\}$, $w_g\,\epsilon\,\{0.0, 0.1, 0.1\}$, $w_c\,\epsilon\,\{0.0, 0.4, 0.4\}$, $w_t\,\epsilon\,\{0.0, 0.1, 0.1\}$, $wo_g\,\epsilon\,\{0.0, 0.1, 0.1\}$, $wo_c\,\epsilon\,\{0.2, 0.7, 0.7\}$, $wo_t\,\epsilon\,\{0.0, 0.1, 0.1\}$. The algorithm has been capable of finding between 93% to 97% of the inserted triclusters. The application of the MSR_{3D} along with the TriGen algorithm in [14] was capable of finding 91% to 95% of the triclusters, so we can see in slight improvement when applying the LSL measure.

B. Elutriation Dataset

We have applied the TriGen algorithm to the yeast (*Saccharomyces Cerevisiae*) cell cycle problem [22]. The yeast cell cycle analysis project's goal is to identify all genes whose mRNA levels are regulated by the cell cycle. The resources used are public and available in http://genome-www.stanford.edu/cellcycle/. Here we can find information relative to gene expression values obtained from different experiments using microarrays. In particular, we have created a dataset $Delu_{3D}$ from the elutriation experiment with 7744 genes, 13 experimental conditions and 14 time points. Experimental conditions correspond to different statistical measures of the Cy3 and Cy5 channels while time points represent different moments of taking measures from 0 to 390 minutes. The control parameter's adjustement used for this experiment is $N = 20$, $G = 1400$, $I = 150$, $Sel = 0.6$, $Mut = 0.1$, $Ale = 0.5$, $w_g = 0.1$, $w_c = 0.2$, $w_t = 0.2$, $wo_g = 0.4$, d$wo_c = 0.2$, $wo_t = 0.2$. To analyze the results, we can see the correlations in Table (I). Therefore, when calculating the averages of correlations close to one and correlations close to minus one we get values close to zero.

TRI_{ID}	Pearson	Spearman
TRI_1	0.88	0.61
TRI_2	0.77	0.72
TRI_3	0.78	0.51
TRI_4	0.84	0.66
TRI_5	0.94	0.62
TRI_6	0.78	0.71
TRI_7	0.89	0.58
TRI_8	0.76	0.69
TRI_9	0.95	0.55
TRI_{10}	0.85	0.79
TRI_{11}	0.88	0.54
TRI_{12}	0.79	0.74
TRI_{13}	0.93	0.67
TRI_{14}	0.88	0.67
TRI_{15}	0.91	0.56
TRI_{16}	0.82	0.66
TRI_{17}	0.85	0.73
TRI_{18}	0.82	0.7
TRI_{19}	0.83	0.68
TRI_{20}	0.82	0.66

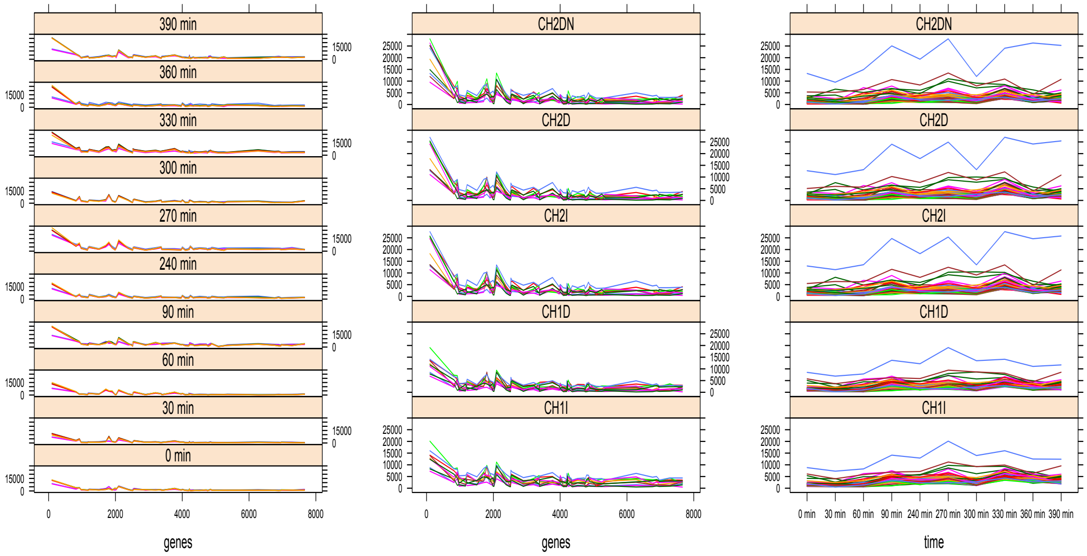
TABLE I: Correlation values of Elutriation results

ID	Name	p-Value
GO:1902222	erythrose 4-phosphate/phosphoenolpyruvate family amino acid catabolic process	4.349×10^{-6}
GO:0006559	L-phenylalanine catabolic process	4.349×10^{-6}
GO:1902221	erythrose 4-phosphate/phosphoenolpyruvate family amino acid metabolic process	2.396×10^{-5}
GO:0006558	L-phenylalanine metabolic process	2.396×10^{-5}
GO:0031305	integral component of mitochondrial inner membrane	4.903×10^{-5}
GO:0006569	tryptophan catabolic process	5.078×10^{-5}
GO:0042436	indole-containing compound catabolic process	5.078×10^{-5}
GO:0046218	indolalkylamine catabolic process	5.078×10^{-5}
GO:0009074	aromatic amino acid family catabolic process	6.945×10^{-5}
GO:0009083	branched-chain amino acid catabolic process	6.945×10^{-5}
GO:0031304	intrinsic component of mitochondrial inner membrane	7.527×10^{-5}

TABLE II: GO analysis of TRI_{14} from Elutriation results

We also show a graphical representation of the genes, conditions and times selected by tricluster TRI_{14} with 50 genes, 5 conditions and 10 time points in Figure 4. In Figure 4a we see TRI_{gct} graphic view where genes at each condition with a graph for each time are represented. Figure 4b shows TRI_{gtc} graphic view which represents the genes at each time with one graph for each condition, and finally in Figure 4c we see TRI_{tgc} graphic view where the times at each gene with a graph for each condition are represented. In Table II we show an analysis of the biological annotations related to the genes selected in our tricluster TC_{14}. In this type of studies, *p*-values are relevant below 0.05 and are better when closer to 0. We show the ten most significant terms with values ranking in the $[4.349 \times 10^{-6}, 7.527 \times 10^{-5}]$ interval. Furthermore, these terms are quiet specific increasing the quality of the tricluster obtained.

We can observe in table III how Pearsons and Spearmans indexes get close to one improving MSR_{3D} index [14] in terms of average, in addition, we can see a clear improvement regarding GO analysis therefore we can conclude that LSL obtains better results in the GO terms.



(a) Sample Curves, TRI_{gct} Graphic View (b) Time-Curves, TRI_{gtc} Graphic View (c) Gene-Curves, TRI_{tgc} Graphic View

Fig. 4: TRI_{14} Graphic views from Elutriation results

	MSR_{3D}	LSL
Max Pearson	0.99	0.95
Min Pearson	-0.02	0.76
Mean Pearson	0.208	0.8485
Max Spearman	0.98	0.79
Min Spearman	-0.02	0.51
Mean Spearman	0.202	0.6525
Max *p*-value	0.010390504	7.53×10^{-5}
Min *p*-value	0.001969545	4.35×10^{-6}
Mean *p*-value	0.005679929	4.29×10^{-5}

TABLE III: Comparison MSR_{3D} and LSL Elutriation results

C. Mouse GDS4510 Dataset

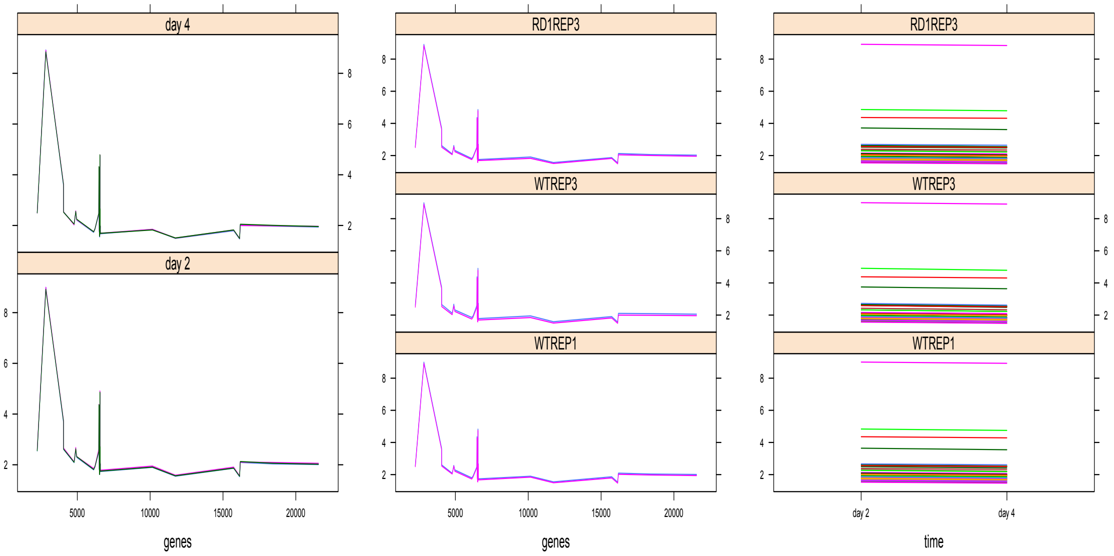
This dataset was obtained from the GEO [23] with accession code GDS4510 and under the title *rd1 model of retinal degeneration: time course* [9]. In this experiment the degeneration of retinal cells in different individuals of home mouse (*Mus musculus*) is analyzed over 4 days just after birth, specifically on days 2, 4, 6 and 8. Our input dataset $DGDS4510_{3D}$ is composed of 22690 genes, 8 experimental conditions (one for each individual involved in the biological experiment) and 4 time points. We have executed the TriGen algorithm with this control parameters: $N = 20$, $G = 500$, $I = 150$, $Sel = 0.6$, $Mut = 0.2$, $Ale = 0.5$, $w_g = 0.8$, $w_c = 0.2$, $w_t = 0.2$, $wo_g = 0.4$, $wo_c = 0.2$, $wo_t = 0.2$. In Table (IV) we see the correlation analysis for the 20 triclusters obtained. The correlation coefficients are very high, and in most cases, perfect with values close to one. This indicates almost perfect homogeneity between the genes, conditions and times of the tricluster.

We show the graphs associated to solution TRI_4 with 40 genes, 2 conditions and 3 time points in Figure 5. We see for the three graphic views, Figures 5a, 5b and 5c how all lines are totally aligned. The biological validity of the solution shown can be found in Table V and yields good results regarding the terms listed and high statistical significance (*p*-values below 0.05). The terms again are very specific and some are related to the dataset under study.

TRI_{ID}	Pearson	Spearman
TRI_1	0.99	0.99
TRI_2	1	0.99
TRI_3	1	1
TRI_4	1	1
TRI_5	1	1
TRI_6	1	1
TRI_7	1	1
TRI_8	1	0.99
TRI_9	1	1
TRI_{10}	1	0.99
TRI_{11}	0.99	0.99
TRI_{12}	1	0.99
TRI_{13}	1	1
TRI_{14}	1	0.99
TRI_{15}	0.99	0.99
TRI_{16}	1	1
TRI_{17}	1	0.99
TRI_{18}	1	0.99
TRI_{19}	1	0.99
TRI_{20}	1	1

TABLE IV: Correlation values of Mouse GDS4510 results

We can observe in Table VI similar values of Pearsons and Spearmans indexes for both quality measures but we can observe a great improvement of LSL against MRS_{3D} in terms of GO analysis. Therefore this reinforces the fact that the LSL measure finds triclusters with higher biological significance.



(a) Sample Curves, TRI_{gct} Graphic view (b) Time-Curves, TRI_{gtc} Graphic view (c) Gene-Curves, TRI_{tgc} Graphic view

Fig. 5: TRI_4 Graphic views from Mouse GDS4510 results

ID	Name	p-Value
GO:0004930	G-protein coupled receptor activity	8.793×10^{-21}
GO:0007606	sensory perception of chemical stimulus	4.312×10^{-19}
GO:0050907	detection of chemical stimulus involved in sensory perception	6.034×10^{-19}
GO:0004984	olfactory receptor activity	1.029×10^{-18}
GO:0050911	detection of chemical stimulus involved in sensory perception of smell	1.029×10^{-18}
GO:0007186	G-protein coupled receptor signaling pathway	8.908×10^{-18}
GO:0009593	detection of chemical stimulus	4.279×10^{-17}
GO:0007608	sensory perception of smell	1.266×10^{-16}
GO:0050906	detection of stimulus involved in sensory perception	5.207×10^{-16}
GO:0004888	transmembrane signaling receptor activity	7.287×10^{-16}
GO:0004872	receptor activity	9.229×10^{-16}
GO:0007600	sensory perception	3.709×10^{-15}
GO:0038023	signaling receptor activity	7.004×10^{-15}
GO:0051606	detection of stimulus	6.489×10^{-13}
GO:0004871	signal transducer activity	1.140×10^{-12}
GO:0060089	molecular transducer activity	1.140×10^{-12}
GO:0007166	cell surface receptor signaling pathway	1.438×10^{-11}
GO:0050877	neurological system process	4.056×10^{-11}
GO:0003008	system process	4.993×10^{-11}
GO:0031224	intrinsic component of membrane	1.235×10^{-8}
GO:0007165	signal transduction	3.107×10^{-8}
GO:0042221	response to chemical	3.585×10^{-8}
GO:0016021	integral component of membrane	3.900×10^{-8}
GO:0007154	cell communication	7.403×10^{-8}

TABLE V: GO analisys of TRI_4 from Mouse GDS4510 results

	MSR_{3D}	LSL
Max Pearson	1	1
Min Pearson	0.99	0.99
Mean Pearson	0.9995	0.9985
Max Spearman	1	1
Min Spearman	0.99	0.99
Mean Spearman	0.9965	0.9945
Max p-value	7.34×10^{-4}	7.40×10^{-8}
Min p-value	1.53×10^{-6}	8.79×10^{-21}
Mean p-value	3.33×10^{-4}	8.02×10^{-9}

TABLE VI: Comparison MSR_{3D} and LSL GDS4510 results

IV. Conclusions and Future Work

In this work we have presented a new evaluation measure for triclusters, LSL, which measures the homogeneity among genes, conditions and times in a tricluster. A detailed formulation of LSL has been provided. In order to assess the quality of the measure, we have applied it along with the TriGen algorithm [11], an evolutionary heuristic to mine triclusters from microarray experiments involving time, to several datasets: synthetically generated data, data from experiments with the yeast cell cycle (*Saccharomyces Cerevisiae*) obtained from the Stanford University [22] and one dataset retrieved from Gene Expression Omnibus [23] which is an experiment for mouse (*Mus Musculus*). All experiments examine the behavior of genes under conditions at certain times. The results obtained have been validated by means of analyzing the correlation among the genes, conditions and times in each tricluster using two different correlation measures: Pearson [25] and Spearman [17]. Besides this, we have provided functional annotations for the genes extracted from the Gene Ontology project [24]. Regarding the synthetic data, we see that LSL combined with TriGen has been capable to extract almost all 10 triclusters artificially inserted in the dataset with a coverage of 93% to 97% . The results for the real datasets are also successful, with correlation values close to one. The GO validation has given good results as well, with high levels of significance for the terms extracted (p-values smaller than 0.05 and very specific terms). Graphical representation of the triclusters has also been provided. LSL is a tricluster evaluation measure created to asses the quality of triclusters extracted from temporal experiments with microarrays, but it can be used in other biologically related fields, for instance combining expression data with gene regulation information by means of substituting the time dimension by ChIP-chip data representing transcription factor-gene interactions what can provide us with regulatory network information. This proposal can also be applied to mine RNA-seq data repositories. Triclustering can also be applied to not biologically related fields, for instance the seismic regionalization of areas at risk of undergoing an earthquake [28]. In this case, the third component does not identify time points but features associated to every pair of geographical coordinates of the area under study.

References

[1] P. O. Brown and D. Botstein, "Exploring the new world of the genome with DNA microarrays," *Nature genetics*, vol. 21, pp. 33–37, 1999.

[2] P. Bajcsy, J. Han, L. Liu, and J. Yang, "Survey of Biodata Analysis from a Data Mining Perspective," *Data Mining in Bioinformatics*, 2005.

[3] J. Quackenbush, "Computational analysis of microarray data." *Nature reviews. Genetics*, vol. 2, no. 6, pp. 418–27, Jun. 2001.

[4] M. P. Tan, E. N. Smith, J. R. Broach, and C. A. Floudas, "Microarray data mining: a novel optimization-based approach to uncover biologically coherent structures." *BMC bioinformatics*, vol. 9, no. 1, p. 268, Jan. 2008.

[5] P. D'haeseleer, S. Liang, and R. Somogyi, "Genetic network inference: from co-expression clustering to reverse engineering." *Bioinformatics (Oxford, England)*, vol. 16, no. 8, pp. 707–26, Aug. 2000.

[6] J. A. Hartigan, "Direct Clustering of a Data Matrix," *Journal of the American Statistical Association*, vol. 67, no. 337, pp. 123–129, Mar. 1972.

[7] B. Pontes, F. Divina, R. Giráldez, and J. S. Aguilar-Ruiz, "Improved biclustering on expression data through overlapping control," *International Journal of Intelligent Computing and Cybernetics*, vol. 3, no. 2, pp. 293–309, 2010.

[8] P. Mahanta, H. A. Ahmed, D. K. Bhattacharyya, and J. K. Kalita, "Triclustering in gene expression data analysis: A selected survey," in *2011 2nd National Conference on Emerging Trends and Applications in Computer Science*. IEEE, Mar. 2011, pp. 1–6.

[9] V. M. Dickison, A. M. Richmond, A. Abu Irqeba, J. G. Martak, S. C. E. Hoge, M. J. Brooks, M. I. Othman, R. Khanna, A. J. Mears, A. Y. Chowdhury, A. Swaroop, and J. M. Ogilvie, "A role for prenylated rab acceptor 1 in vertebrate photoreceptor development." *BMC neuroscience*, vol. 13, p. 152, Jan. 2012.

[10] A. Tanay, R. Sharan, and R. Shamir, "Discovering statistically significant biclusters in gene expression data," *Bioinformatics*, vol. 18, no. Suppl 1, pp. S136–S144, Jul. 2002.

[11] D. Gutiérrez-Avilés, C. Rubio-Escudero, F. Martínez-Álvarez, and J. C. Riquelme, "TriGen: A genetic algorithm to mine triclusters in temporal gene expression data," *Neurocomputing*, vol. 132, no. 0, pp. 42–53, 2014.

[12] F. Divina, B. Pontes, R. Giráldez, and J. S. Aguilar-Ruiz, "An effective measure for assessing the quality of biclusters," *Computers in Biology and Medicine*, vol. 42, no. 2, pp. 245–256, 2012.

[13] Y. Cheng and G. M. Church, "Biclustering of expression data." in *Ismb*, vol. 8, 2000, pp. 93–103.

[14] D. Gutiérrez-Avilés and C. Rubio-Escudero, "Mining 3D Patterns from Gene Expression Temporal Data: A New Tricluster Evaluation Measure," *The Scientific World Journal*, vol. 2014, pp. 1–16, 2014.

[15] L. Zhao and M. J. Zaki, "TRICLUSTER: an effective algorithm for mining coherent clusters in 3D microarray data," in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data - SIGMOD '05*. New York, New York, USA: ACM Press, 2005, p. 694.

[16] H. Jiang, S. Zhou, J. Guan, and Y. Zheng, "gTRICLUSTER : A More General and Effective 3D Clustering Algorithm for Gene-Sample-Time Microarray Data," in *BioDM*, no. 60373019, 2006, pp. 48–59.

[17] C. SPEARMAN, "CORRELATION CALCULATED FROM FAULTY DATA," *British Journal of Psychology, 1904-1920*, vol. 3, no. 3, pp. 271–295, Oct. 1910.

[18] X. Xu, Y. Lu, K.-L. Tan, and A. K. H. Tung, "Finding Time-Lagged 3D Clusters," in *2009 IEEE 25th International Conference on Data Engineering*. IEEE, Mar. 2009, pp. 445–456.

[19] K. Sim, Z. Aung, and V. Gopalkrishnan, "Discovering Correlated Subspace Clusters in 3D Continuous-Valued Data," in *2010 IEEE International Conference on Data Mining*. IEEE, Dec. 2010, pp. 471–480.

[20] A. B. Tchagang, S. Phan, F. Famili, H. Shearer, P. Fobert, Y. Huang, J. Zou, D. Huang, A. Cutler, Z. Liu, and Y. Pan, "Mining biological information from 3D short time-series gene expression data: the OP-Tricluster algorithm." *BMC bioinformatics*, vol. 13, no. 1, p. 54, Jan. 2012.

[21] M. Martínez-Ballesteros, F. Martínez-Álvarez, A. Troncoso, and J. C. Riquelme, "An evolutionary algorithm to discover quantitative association rules in multidimensional time series," *Soft Computing*, vol. 15, no. 10, pp. 2065–2084, Mar. 2011.

[22] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher, "Comprehensive Identification of Cell Cycle-regulated Genes of the Yeast Saccharomyces cerevisiae by Microarray Hybridization," *Molecular Biology of the Cell*, vol. 9, no. 12, pp. 3273–3297, Dec. 1998.

[23] T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, M. Holko, A. Yefanov, H. Lee, N. Zhang, C. L. Robertson, N. Serova, S. Davis, and A. Soboleva, "NCBI GEO: archive for functional genomics data sets–update." *Nucleic acids research*, vol. 41, no. Database issue, pp. D991–5, Jan. 2013.

[24] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock, "Gene ontology: tool for the unification of biology. The Gene Ontology Consortium." *Nature genetics*, vol. 25, no. 1, pp. 25–9, May 2000.

[25] K. Pearson and L. N. G. Filon, "Mathematical contributions to the theory of evolution. IV. On the probable errors of frequency constants and on the influence of random selection on variation and correlation," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, pp. 229–311, 1898.

[26] S. Bauer, S. Grossmann, M. Vingron, and P. N. Robinson, "Ontologizer 2.0–a multifunctional tool for GO term enrichment analysis and data exploration." *Bioinformatics (Oxford, England)*, vol. 24, no. 14, pp. 1650–1, Jul. 2008.

[27] R. P. Pargas, M. J. Harrold, and R. P. Peck, "Test-data generation using genetic algorithms," *Software ...*, vol. 6, 1999.

[28] J. Reyes and V. H. Cárdenas, "A Chilean seismic regionalization through a Kohonen neural network," *Neural Computing and Applications*, vol. 19, no. 7, pp. 1081–1087, May 2010.