A Bi-Objective Clustering Algorithm for Gene Expression Data

Jorge Parraga-Alava, Mario Inostroza-Ponta

Centre for Biotechnology and Bioengineering.

Departamento de Ingeniería Informática,

Universidad de Santiago de Chile

Av. Ecuador #3659, Estación Central, Chile

{jorge.parraga, mario.inostroza}@usach.cl

Abstract

Clustering algorithms are a common method for data analysis in many science field. They have become popular among biologists because of ease to discovery similar cellular functions in gene expression data. Most approaches consider the gene clustering as an optimization problem, where an ad-hoc cluster quality index is optimized which can be defined regarding gene expression data or biological information. However, these approaches may not be sufficient since they cannot guarantee to generate clusters with similar expression patterns and biological coherence. In this paper, we propose a bi-objective clustering algorithm to discover clusters of genes with high levels of co-expression and biological coherence. Our approach uses a multi-objective evolutionary algorithm (MOEA) that optimizes two index based on gene expression level and biological functional classes. The algorithm is tested on three real-life gene expression datasets. Results show that the proposed model yields gene clusters with higher levels of co-expression and biological coherence than traditional approaches.

Keywords: Clustering, Bi-objective, Gene Expression, External Biological Knowledge, Metaheuristic.

1 Introduction

In the field of molecular biology multiple techniques have been applied for measuring the levels of RNA, DNA, proteins or metabolites. Microarray is one of the most popular techniques for global and simultaneous collection of thousands of these levels under different conditions: a time series during a biological process, experiments of different tissue samples, among others [1]. In bioinformatics, a gene expression microarray, can be represented with a matrix $n \times m$, that contains the expression of n probes (for example genes) on m conditions (samples, tests, cases, among others). This matrix has the advantage of allowing the study of problems such as differential gene expression [2, 3], patterns of genes with (dis)similar expression levels [4, 5], prediction of response to treatment [6, 7], detection of gene mutations [8, 9].

Many of these gene expression matrix have a considerable size, so that to study the biological data it is required the use of appropriate computational methods. Clustering techniques have proved to be useful when handling large volumes of data simultaneously while, in the case of genes with (dis)similar expression, identifying groups involved in the same cellular function or that are regulated in the same way. This process, as stated in [10], becomes the goal of clustering since it aims to group n genes or conditions into k subsets in such a way that similar (according to their expression) elements are grouped together, while different elements belong to different groups. The problem of clustering can be seen as an optimization problem where some cluster quality index (objective) is optimized [11]. Quality index may represent different characteristics of the clusters [12], such as compactness, separation among clusters or connectivity within a cluster.

Popular clustering algorithms such as k-Means [13], Hierarchical clustering [14, 15], PAM (Partitional Around Medoids) [16, 17] and SOM (Self Organizing Maps) [18] have been used to find patterns of genes in microarray experiments optimizing one validity index. These approaches are known as Single-Objective Clustering (SOC) [19]. However, they have some problems: one index is rarely equally well applicable to different types of dataset, i.e. one criterion may not fit well with data distribution when regions of the data contains clusters of diverse shapes [20]; moreover, many of these algorithms often fall into local optima

and sometimes they can deliver results that are meaningless for data analysis (e.g. when is optimized only separation validity index it could yield one group with all genes).

In order to overcome these issues, multi-objective optimization (MOO) approach arises as alternative. The works presented in [21, 22, 23, 24] have explored the application of MOO to the clustering problem. In this case, we called Multi-Objective Clustering (MOC), and it corresponds to the simultaneous optimization of several validity index. This approach is a two-step process: (i) discovery of clusters by any clustering algorithm during the optimization of a quality index, and (ii) construction of an “optimal” set of solutions that correspond to various tradeoffs between different objectives [20]. The work of [25] uses a MOC approach incorporating a method called SiMM-TS (Significant Multi-class Membership - Two Stage). In the first stage, through a genetic algorithm (GA) the clustering process is performed based on expression levels and then SiMM points (genes are located in the regions of overlap of two or more clusters) are identified, which are temporarily removed from the data set. In the last stage, it performs a new clustering process, but without the SiMM points, through a genetic algorithm called MOGA (Multi-Objective Genetic Algorithm) which optimizes Xie-Beni (XB) and J_m index. In [23], they also use MOGA but adding a technique of majority vote which identifies genes almost always grouped together in most Pareto optimal solutions. These genes are used as training data for a classifier, whereas the remaining genes are used as test data. Finally, both results are combined to obtain the final clustering of genes. In a more recent work [26], it is proposed a technique for gene clustering which optimizes several validity index (Davies-Bouldin (DB), Xie-Beni (XB), J_m , PBM and S) and adds an interactive component through a decision maker (DM). The DM evaluates clustering solutions obtained based on the biological expertise in the area. If they are biologically related they are labeled as high-quality clustering. Clearly, the effectiveness of the technique will depend on domain knowledge and expertise at the biological level of DM.

In previous studies it is possible to note that clusters are generated using information about expression level and only in one work the biological information is used but not during the step of generation of clusters. If both information were used at the same time it could allow to find gene clusters with stronger relationship in expression levels and common biological properties. In this paper we propose a *Dual-MOC* model that optimizes two index which use information based on gene expression and biological functional classes. The goal is to produce gene clusters with higher levels of co-expression and biological coherence than SOC approaches.

2 Multi-Objective Clustering

A Multi-Objective Clustering (MOC) approach aims to determine the clustering C^* for which:

$$P_t(C^*) = \max_{C \in \Omega} P_t(C), t = 1, \dots, m \quad (1)$$

Where Ω is the set of feasible clustering, C is a clustering of a given dataset E , and $P_t, t = 1, \dots, m$ is a set of m different objective functions (quality index), i.e. that clustering C^* corresponds to a cluster solution that has the best optimized m criteria P [27, 19].

In SOC problems a solution with a higher objective function value is better than (in maximization case) one with a small objective function value. However, in MOC problems, there is no natural ordering in the objective space and often objectives functions tend to be in conflict [28], i.e. improve one involves worsening another. Because of that it is required to reach a “tradeoff” situation where all objective functions are satisfied to an acceptable degree. Such situation implies to find a set of best solutions, which are called non-dominated solutions [19].

2.1 Non-dominated solutions

Given two clustering solutions C_1 and $C_2 \in \Omega$, solution C_1 is said to *dominate* solution C_2 (denoted as $C_1 \succ C_2$) (in maximization) if only if:

$$\forall t : P_t(C_1) \geq P_t(C_2) \wedge \exists t : P_t(C_1) > P_t(C_2) \quad (2)$$

In other words, $C_1 \succ C_2$ if and only if: C_1 is better than C_2 in at least one objective function P_t and there is no worse objective function P_t where C_2 is better than C_1 .

2.2 Pareto optimal set

Pareto optimal set Π is defined as:

$$\Pi = \{C \in \Omega : \nexists C' \in \Omega : C' \succ C\} \quad (3)$$

In other words, a Pareto optimal set Π corresponds to a set of non-dominated solutions, such that there is no other solution C' within or outside the set that dominates any of them.

3 Proposed Model

In this section, we describe in detail our bi-objective gene clustering model called Dual-MOC. Figure 1 shows a basic schema of the model. The proposed model consists of two phases. In the first, called *information integration*, distance matrices of expression and biological functional similarity are combined. In the second, *clustering step*, a popular Multi-Objective Evolutionary Algorithm (MOEA) called NSGA-II (Nondominated Sorting Genetic Algorithm) [29] is used to optimize two objective functions.

Figure 1: Proposed bi-objective clustering model pipeline

3.1 Phase I: Information integration

In this phase, we use a distance matrix that quantifies how genes are correlated regarding expression value profiles. We measure such correlation using *Pearson correlation* (ρ) coefficient [30]. This matrix, is called *distance of expression profiles* (D_{EP}). In addition, we use a similarity matrix that measures the relationship among genes by functional similarity of them considering biological annotations terms of Gene Ontology (GO). We compute such relation through *wang functional similarity* (WS) [31]. This matrix, is called *distance of biological functionality* (D_{BF}).

Both matrices are combined in a *dual similarity matrix* according to [32, 33] using a weighting parameter $\alpha \in [0, 1]$. So that, a gene pair G_1, G_2 have a weighted distance $D_\alpha(G_1, G_2)$ combining their distances of gene expression profiles $D_{EP}(G_1, G_2)$ and biological functionality $D_{BF}(G_1, G_2)$ as follows:

$$D_\alpha(G_1, G_2) = \alpha \times D_{EP}(G_1, G_2) + (1 - \alpha) \times D_{BF}(G_1, G_2) \quad (4)$$

Thus, if the $\alpha = 1$ then be considering only distance of expression profiles, and if $\alpha = 0$, would be considered only distances of biological functionality. Other α values consider to a greater or lesser degree the information of matrices of similarity.

3.2 Phase II: Clustering step

In this second phase, the bi-objective gene clustering process is performed. It is based on NSGA-II algorithm since it is the standard multi-objective algorithm [25] [23] [34] [35] and it has been demonstrated superiority in performance over other MOEAs when the goal is to yield clustering solutions of gene expression data.

Our Dual-MOC algorithm creates the initial population with clustering solutions produced by single objective approaches. Such initial population is used as input of the NSGA-II algorithm which is detailed above.

- An individual in the population is represented as a vector of integer values, where each one denote a cluster medoid. A cluster medoid is the most central gene located in a cluster, i.e. the sum of the distances to other gene of the cluster is minimum. Thereby, each individual has the same length as the number of clusters k , and each position in this chromosome can have an integer value from 1 to n , n being the number of genes of the dataset. Each element in dataset will be allocated to the nearest cluster medoid according to the *dual similarity matrix*.
- As selection method, we use a *binary tournament* along with Pareto ranking and crowding distance (which are characteristic of NSGA-II). To identify the winners individuals of each tournament, the individual belonging to the lower Pareto ranking is chosen and, in the case of a tie, the one with the highest crowding distance is selected.
- As evolutionary operators, we use *single-point crossover* and a *controller-random mutation* [36]. In the crossover, a single crossover point on both individuals (parents) is randomly selected. All cluster medoid beyond that point in either integer vectors are swapped between the two individuals. In controller-random mutation, a random position is selected from the chromosome and the corresponding gene index is replaced by a randomly chosen gene index from the dataset, so that the chromosome does not contain repeated gene index after mutation.

Our Dual-MOC algorithm performs the clustering process optimizing two objective functions. We use the *Co-expression Indicator* (CI) [37] and *Biological Homogeneity Index* (BHI) [38]. They optimise the relationships among genes at expression level and biological functionality, respectively. To compute *CI* is used the gene expression data, meanwhile to *BHI* a reference set of functional classes is required.

4 Computational Experiments

The goal of the experiments is to asses the performance of Dual-MOC to produce gene partitions with better relationships of expression and biologically meaningful clusters.

4.1 Datasets

Data used for experiments corresponds to three real-life microarray gene expression datasets: *yeast cell cycle*, *medulloblastoma metastasis* and *arabidopsis thaliana*. The data were normalized so that each row has mean 0 and variance 1. In Table 1, we show a summary of main features of datasets. Here, column original corresponds to the number of genes available in microarray. Column reduced corresponds to the number of genes available after of a pre-processing stage where duplicated genes and missing values of expression levels are remove from microarray. Such reduced number of genes are finally used in computational experiments.

Table 1: Microarray data used in experiments.

Datasets	# Genes		# Samples
	original	reduced	
Yeast cell cycle	6000	384	17
Medulloblastoma metastasis	1465	585	23
Arabidopsis thaliana	138	133	8

4.1.1 Yeast cell cycle

The yeast cell cycle dataset was collected from <http://faculty.washington.edu/kayee/cluster>. It shows the fluctuation of expression levels of approximately 6000 genes over two cell cycles (17 time points). Out of these, 384 genes are selected to be cell-cycle regulated and to be unique [26].

4.1.2 Medulloblastoma metastasis

The Medulloblastoma metastasis dataset was contributed by [39]. The dataset was collected from the Gene Expression Omnibus (GEO) using the code GSE468. It contains 2059 probes in 23 primary medulloblastomas clinically designated as either metastatic or non-metastatic samples [40]. Out of these probes, 1465 correspond to genes and 585 are selected to be unique. The data are publicly available at <http://www.biolab.si/supp/bi-cancer/projections/info/medulloblastomiGSE468.htm>.

4.1.3 Arabidopsis thaliana

The Arabidopsis thaliana dataset was collected from <http://anirbanmukhopadhyay.50webs.com/data.html> and it shows the gene expression values of 138 genes over eight time points (15 min, 30 min, 60 min, 90 min, 3 h, 6 h, 9 h, and 24 h). Out of these, 133 genes are selected to be unique.

4.2 Distance Metrics

As previously mentioned, we use two measures of distance, *Pearson correlation coefficient* to measure the degree of similarity of genes according to their expression level and *Wang functional similarity* [31] that measures the relationship regarding the biological information using Gene Ontology (GO).

4.2.1 Pearson correlation coefficient

The Pearson correlation coefficient (ρ) is a measure of similarity indicating how and how strongly gene expression of two genes (G_x, G_y) for several different conditions are related. This correlation coefficient is computed as follows:

$$\rho = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{\sigma_x} \right) \left(\frac{y_i - \bar{y}}{\sigma_y} \right), \quad (5)$$

where x_i, y_i are the gene expression of G_x and G_y in the i -th condition; $\bar{x}, \bar{y}$ correspond to the mean of gene expression in all conditions; and, σ_x, σ_y are the standard deviation of the gene expression of x and y .

In our experiments, we compute $1 - \rho_{G_x, G_y}$. Thus, values close to 0 would indicate high similarity between the genes G_x, G_y .

4.2.2 Wang functional similarity

Wang functional similarity (WS) [31] is an information content (IC) based metric which determines the similarity of two GO (Gene Ontology) terms based on both the locations of these terms in the GO graph and their relations with their ancestor terms. First, the measure defines the semantic similarity between one *go* term and a *GO* term set $GO = \{go_1, go_2, \dots, go_k\}$, that is the maximum semantic similarity between term *go* and any of the terms in set *GO*. Calculated as follows:

$$Sim(go, GO) = \max_{1 \leq i \leq k} (S_{GO}(go, go_i)), \quad (6)$$

Where, $S_{GO}(go, go_i)$ is the semantic similarity between *go, go_i* and k is the number of terms in set *GO*. Therefore, given two genes G_x and G_y annotated by GO term sets $GO_1 = \{go_{11}, go_{12}, \dots, go_{1m}\}$ and $GO_2 = \{go_{21}, go_{22}, \dots, go_{2m}\}$ respectively, their Wang functional similarity is defined as:

$$WS(G_x, G_y) = \frac{\sum_{1 \leq i \leq m} Sim(go_{1i}, GO_2) + \sum_{1 \leq j \leq n} Sim(go_{2j}, GO_1)}{m + n} \quad (7)$$

This measure obtains values within the range from 0 to 1. We use $1 - WS(G_1, G_2)$, so that values close to 0 indicates that gene G_1 and G_2 are similar semantically.

4.3 Objective Functions

4.3.1 Objective function (CI)

We use the co-expression indicator (CI) as objective function. This indicator was proposed by [37] who defines it as the average of correlations between the genes of the same cluster C_g based on expression levels. It is calculated as follows:

$$fu_{card} = \frac{2}{(card(C_g))(card(C_g) - 1)}, \quad (8)$$

$$\rho_{G_x, G_y} = \sum_{G_y | G_y \in C_g, G_y > G_x} \frac{1}{M} \sum_{m=1}^M \frac{E_{mG_x} - \bar{E}_{G_x}}{\sigma_{E_{G_x}}} \frac{E_{mG_y} - \bar{E}_{G_y}}{\sigma_{E_{G_y}}}, \quad (9)$$

$$CI(C_g) = fu_{card} \sum_{G_x | G_x \in C_g} (\rho_{G_x, G_y}), \quad (10)$$

where M is the number of samples, E_{mG_x} and E_{mG_y} are the expression for sample m of the genes G_x and G_y respectively. $\bar{E}_{G_x}$ and $\bar{E}_{G_y}$ are the mean of the M expression values of the genes G_x and G_y respectively. $\sigma_{E_{G_x}}$ and $\sigma_{E_{G_y}}$ are the standard deviation of the M expression values of the genes G_x and G_y respectively. In our experiments we consider a variation proposed by [33] as follows:

$$CI(solution) = \sum_{g=1}^k fu_{card} \sum_{G_x | G_x \in C_g} (\rho_{G_x, G_y}), \quad (11)$$

where k is max number of clusters, and therefore we measure the average co-expression of the clustering $CI(solution)$. It variation obtains value within the range from 0 to 1, so that values close to 1 indicates that gene in clusters are co-expressed.

4.3.2 Objective function (BHI)

We use the Biological Homogeneity Index (BHI) as objective function. It measures the biological significance of clusters. This criteria was proposed by [38] and it requires a reference set of functional classes which is created using some functional annotation tool. BHI is calculated as follows:

$$BHI = \frac{1}{K} \sum_{j=1}^K \frac{1}{n_j(n_j - 1)} \sum_{x \neq y \in D_j} I(C(x) = C(y)), \quad (12)$$

where K is the maximum number of partitions of the clustering solution, n_j is the number of annotated genes in cluster D_j . $C(x)$ and $C(y)$ are the functional classes where gen x and gen y are contained. Now, suppose that two annotated genes x and y belong to the same cluster D produced by some algorithm. Let $C(x)$ and $C(y)$ be the functional classes that contain the genes x and y , respectively. The function $I(C(x) = C(y))$ will have the value 1 if $C(x)$ matches with $C(y)$ [26], i.e. genes are listed in the same functional class. In this case at least one match is considered sufficient.

BHI obtains value within the range from 0 to 1, so that values close to 1 indicates that clustering solution has more biologically homogeneous groups.

Functional Classes We create functional classes of three datasets using FatiGO [41]. For *cell cycle dataset* of 384 genes, 243 were annotated and functional classes containing at least 10% genes were selected. It produced 5 overlapping classes corresponding to biological processes. For *medulloblastoma metastasis* of 585 genes, 484 were annotated and functional classes containing at least 5.4% genes were selected. It produced 11 overlapping classes corresponding to biological processes. For *arabidopsis thaliana* of 133 genes, 81 were annotated and functional classes containing at least 12.8% genes were selected. It produced 10 overlapping classes corresponding to biological processes. A summary of the functional classes is shown in Table 2.

Table 2: Functional classes for each dataset.

Dataset	Annotated genes number	Classes number
Cell cycle	243	5
Metastasis	484	11
Thaliana	81	10

4.4 Input parameters

The evolutionary parameters used by Dual-MOC for three datasets are: number of generations = 30, population size = 100, crossover probability = 0.80, mutation probability = 0.01 and tournament selection = 20% of population size. Number of clusters $k = \{5, 6, \dots, 10\}$ and $\alpha \in \{1, 0.9, \dots, 0\}$. Each single objective algorithm has been executed for 100 times meanwhile Dual-MOC 20 times. The parameter have been set after several experiments and considering similar configurations used for MOC studies [25, 23, 26, 42] applied to gene clustering.

4.5 Performance metric

To compare the efficiency of Dual-MOC, we use the metric hypervolume (HV) [43], because according to [44] it is the most widely used metric of performance in multi-objective scenarios. HV is a unary metric that measures the volume (area in our case) in the objective function space covered by members of an Pareto set Q . Here, HV for every $i \in Q$ computes an area a_i regarding a reference point W . Thereby, the union of all areas a_i define the hypervolume value.

$$HV = area \left(\bigcup_{i=1}^{|Q|} a_i \right) \quad (13)$$

4.6 Contestant algorithms

In our experiments, we aim to demonstrate the advantage of bi-objective clustering that simultaneously optimizes objective function based on expression and biology compared with SOC approaches based on optimizing only one objective at a time. For this purpose, we compare our model against three different algorithms which optimized separately expression and biology criteria. The competitive algorithms are:

k-Means: It is a partitioning method that uses a minimum “within-class sum of squares from the centers” criterion to select the clusters [38] i.e., partition the set of n observations into k clusters, so that each observation belongs to the cluster closest to the mean.

PAM: *Partitioning Around Medoids* is another partitioning method that is used to cluster the types of data in which the mean of objects is not defined or available. This algorithm finds the representative object (i.e., medoid, which is the multidimensional version of the median) of each cluster mean while a cost function is minimized [45].

UPGMA: *Unweighted Pair Group Method with Arithmetic mean* is perhaps the most commonly used clustering method with microarray data sets [38]. It is an agglomerative hierarchical method that yield a dendrogram can be cut at a chosen height to produce the desired number of clusters [46]. It uses a dissimilarity matrix in order to decide if two expression or biological profiles are close or not.

4.7 Development Tools

All the algorithms used in the experiment were implemented using libraries of R [47] and computational tests performed on a computer with Intel(R) Core(TM) i7-3930K CPU 3.20GHz of 6 cores, 16GB RAM running Ubuntu OS, kernel version 3.13.0-87-generic. *UPGMA* (hclust) is available in the base distribution of R. *PAM* and *k-means* are available in the “amap” library [48]. Some features of *NSGA-II* and *HV* used were implemented by authors and others are available in the “mco” library [49] and “nsga2R” library [50]. The *distance of expression profiles* was calculated using functions of the base distribution of R. The *distance of biological functionality* using the “GOSemSim” library [51]. The objective function *CI* (Coexpression Indicator) was implemented by authors and the objective function *BHI* (Biological Homogeneity Index) was computed using the “clValid” library [52].

5 Results and Discussion

We first show the α parameter incidence in terms of the size of the objective space covered by Dual-MOC in the 20 runs. To do this, we calculate the hypervolume (HV) indicator using as reference point the worst values of the objective functions, i.e. (0,0). Figure 2 shows the *HV* values obtained for all datasets.

Figure 2: Influence of α parameter in terms of the size of the objective space covered (HV) by Dual-MOC.

In Figure 2, the best values of HV are identified by a black circle. In the case of yeast dataset (Fig. 2a), the best value of HV (0.74897) is reached when $\alpha = 0.2$. In the case of medulloblastoma (Fig. 2b), the best value of HV (0.63658) is reached with $\alpha = 0.8$. Meanwhile, arabidopsis thaliana (Fig. 2c) obtained an HV value of (0.60392) when $\alpha = 0.1$. A summary with the best α values and related k configuration found by Dual-MOC in terms of the size of objective space covered (HV) are shown in Table 3.

Table 3: Best configurations found by Dual-MOC in all datasets

Dataset	Hypervolume	Iteration	α	k
Cell cycle	0.74897	12	0.2	10
Metastasis	0.63658	14	0.8	9
Thaliana	0.60393	3	0.1	8

Based on the best configurations in Table 3, we compare our results against three SOC algorithm: *k-Means*, *PAM* and *UPGMA* under three cases: (a) using only distance of expression profile, (b) only distance of biological functionality and (c) combination of both.

For each SOC algorithm, we report the best value found according to both *CI* and *BHI* index out of a 20 runs. Note that in the case of *k-Means* algorithm, two best values are available. The solution labeled as “*k-Means-exp*” corresponds to the solution with the highest value of *CI*, on which *BHI* value was computed. While the solution labeled as “*k-Means-bio*” corresponds to the solution with the highest value of *BHI*, on which *CI* was computed.

5.1 Results for yeast cell cycle data

Case a: expression Here, we show the results considering $\alpha = 1$, i.e., clustering of genes using only *distance of expression profile*. To facilitate visualization we consider only results for $k = \{5, 7, 9, 10\}$. Table 4 shows the *CI* and *BHI* values for three SOC algorithms and the non-dominated solutions of Dual-MOC model. Here, it is evident that all Dual-MOC solutions achieve better values (bold) than the SOC algorithms at least in one of the objectives under evaluation. In particular, in the case of $k = \{7, 10\}$ values (bold and underline) of *CI* and *BHI* defeat the best (in italic) SOC algorithm in both index.

Table 4: Objective functions values with $\alpha=1$ in yeast cell cycle (run=12). In italic, we show the best solution in SOC. In bold, we show the Dual-MOC solutions that are better in at least one objective than the best SOC solution. In bold and underline, we show the Dual-MOC solutions better than the best SOC solution.

Algorithms	k=5		k=7		k=9		k=10	
	CI	BHI	CI	BHI	CI	BHI	CI	BHI
k-Means-exp	<i>0.7275</i>	0.7988	<i>0.7398</i>	0.7741	<i>0.7470</i>	0.8263	<i>0.7501</i>	0.7500
k-Means-bio	0.7079	<i>0.8730</i>	0.7149	<i>0.8674</i>	0.7299	<i>0.8771</i>	0.7389	<i>0.8701</i>
PAM	0.7022	0.7627	0.7167	0.6822	0.7196	0.6220	0.7222	0.6656
UPGMA	0.6978	0.7202	0.7105	0.6428	0.7144	0.5268	0.7121	0.6001
Dual-MOC-sol-1	0.7329	0.8459	0.7354	0.8985	0.7520	0.8484	0.7543	0.8446
Dual-MOC-sol-2	0.7037	0.8886	0.7462	0.8414	0.7217	0.9104	0.7416	0.9046
Dual-MOC-sol-3	0.7278	0.8488	0.7429	0.8634	0.7237	0.9064	0.7474	0.8872
Dual-MOC-sol-4	0.7238	0.8623	0.7381	0.8769	0.7409	0.9053	0.7510	0.8761
Dual-MOC-sol-5	0.7151	0.8813	0.7376	0.8957	0.7520	0.8580	0.7435	0.9028
Dual-MOC-sol-6	0.7045	0.8865	0.7403	0.8716	0.7431	0.9051	0.7531	0.8623
Dual-MOC-sol-7	0.7109	0.8842	0.7413	0.8716	0.7462	0.8936		
Dual-MOC-sol-8	0.7188	0.8649	0.7355	0.8977	0.7463	0.8866		
Dual-MOC-sol-9	0.7167	0.8704			0.7498	0.8616		
Dual-MOC-sol-10	0.7122	0.8828			0.7488	0.8704		
Dual-MOC-sol-11	0.7170	0.8677			0.7474	0.8793		
Dual-MOC-sol-12	0.7113	0.8836			0.74468	0.8986		
Dual-MOC-sol-13	0.7169	0.8686			0.7446	0.8988		
Dual-MOC-sol-14					0.7498	0.86263		
Dual-MOC-sol-15					0.7482	0.87633		
Dual-MOC-sol-16					0.7479	0.8782		

Case b: biology Similarly, in Table 5, we show the results using only distance of biological functionality ($\alpha = 0$). Here, for instance when the dataset is partitioned into $k=5$ clusters, we have that two of the seven non-dominated solutions achieve better values (in bold and underline) in *BHI* and *CI* index than the SOC algorithms. Similarly when the partition has $k=7$ clusters, we have five (of nine) solutions of Dual-MOC exceed SOC solutions values in *BHI* criteria. For other cases, solutions found by Dual-MOC achieve better values (bold) than the SOC algorithms at least in one of the objectives under evaluation. In particular, when $k = 10$ solutions yield by Dual-MOC achieved higher values of *BHI* than the best (italic) SOC algorithm solution.

5.2 Results for medulloblastoma metastasis data

Case a: expression In Table 6, superiority of Dual-MOC regarding the best (in italic) SOC solution is clearly observed. It achieves better values in most non-dominated solutions (in bold) in *CI* criterion. In the case of $k = 7$, nine out of ten of the non-dominated solutions outperforms the best SOC solution on the mentioned criterion. For the remaining cases of k , there is also at least one non-dominated solution that is better in both criteria. Meanwhile, for $k=10, 11$ (in bold and underline) of the 15 non-dominated solutions dominate all the SOC solutions.

Case b: biology Table 7 shows the results of considering an $\alpha = 0$, i.e., clustering of genes using only distance of biological functionality. Here, Dual-MOC achieves better values (in bold) in *BHI* criterion than

Table 5: Objective functions values with $\alpha=0$ in yeast cell cycle data (run=12). In italic, we show the best solution in SOC. In bold, we show the Dual-MOC solutions that are better in at least one objective than the best SOC solution. In bold and underline, we show the Dual-MOC solutions better than the best SOC solution.

Algorithms	k=5		k=7		k=9		k=10	
	CI	BHI	CI	BHI	CI	BHI	CI	BHI
k-Means-exp	<i>0.6862</i>	0.8137	<i>0.6943</i>	0.9128	<i>0.6945</i>	0.8006	<i>0.6934</i>	0.6583
k-Means-bio	0.6753	<i>0.9550</i>	0.6736	<i>0.9621</i>	0.6707	<i>0.9522</i>	0.6764	<i>0.9170</i>
PAM	0.6728	0.6137	0.6636	0.6556	0.6736	0.5493	0.6699	0.6778
UPGMA	0.6721	0.7012	0.6610	0.6649	0.6707	0.5950	0.6700	0.6439
Dual-MOC-sol-1	0.7058	0.7423	0.7005	0.8011	0.6893	0.9861	0.7054	0.6559
Dual-MOC-sol-2	0.6845	0.9756	0.6798	0.9762	0.7011	0.8395	0.6735	0.9892
Dual-MOC-sol-3	0.6941	0.7635	0.6952	0.9490	<u>0.6957</u>	<u>0.9675</u>	0.7010	0.7478
Dual-MOC-sol-4	0.6929	0.9600	0.6869	0.9637	0.6962	0.8506	0.7007	0.8247
Dual-MOC-sol-5	<u>0.6910</u>	<u>0.9655</u>	0.6927	0.9542	0.6928	0.9779	0.7040	0.6637
Dual-MOC-sol-6	0.7001	0.7614	0.6852	0.9716	0.6977	0.8441	0.7039	0.7370
Dual-MOC-sol-7	0.7011	0.7610	0.6815	0.9718	0.6944	0.9696	0.6931	0.9230
Dual-MOC-sol-8			0.6928	0.9540	0.6971	0.8496	0.6952	0.8713
Dual-MOC-sol-9			0.6812	0.9718	0.6976	0.8444	0.6784	0.9853
Dual-MOC-sol-10							0.6874	0.9573
Dual-MOC-sol-11							0.6980	0.8494
Dual-MOC-sol-12							0.6909	0.9564
Dual-MOC-sol-13							0.6956	0.8563
Dual-MOC-sol-14							0.7038	0.7461
Dual-MOC-sol-15							0.6920	0.9456
Dual-MOC-sol-16							0.6837	0.9800
Dual-MOC-sol-17							0.6863	0.9760
Dual-MOC-sol-18							0.6813	0.9815
Dual-MOC-sol-19							0.6802	0.9823
Dual-MOC-sol-20							0.6869	0.9713
Dual-MOC-sol-21							0.6790	0.9828
Dual-MOC-sol-22							0.6865	0.9739
Dual-MOC-sol-23							0.6789	0.9852

k-Means-bio (best in SOC algorithms) at least twice. Furthermore, complete superiority of non-dominated set yield by Dual-MOC is observed in solutions (bold and underline) where both *CI* and *BHI* have higher values than SOC solutions, in particular the case of $k = 10$ where 10 out of 15 non-dominated solutions outperform SOC solutions in both index.

5.3 Results for arabidopsis thaliana data

Case a: expression In Table 8, we show the results using only distance of expression profiles ($\alpha = 1$). Here, for example in cases $k = \{5, 7, 9\}$, our model found at least one solution with the best values in both objectives (bold and underline). For $k=10$, there are six of twelve non-dominated solutions that are better in *CI* criterion than the best SOC solution.

Case b: biology In Table 9, we observe that for all k cases our proposed model achieved at least one solution with the best values in both objectives function (bold and underline) regarding the best values (italic) the SOC algorithms. When the dataset was partitioned in $k = 10$ groups, Dual-MOC demonstrated superiority since it achieved the highest values (bold and underline) for both *CI* and *BHI* criteria in 6 out of 15 non-dominated solutions.

Table 6: Objective functions values with $\alpha=1$ in medulloblastoma data (run=14). In italic, we show the best solution in SOC. In bold, we show the Dual-MOC solutions that are better in at least one objective than the best SOC solution. In bold and underline, we show the Dual-MOC solutions better than the best SOC solution.

Algorithms	k=5		k=7		k=9		k=10	
	CI	BHI	CI	BHI	CI	BHI	CI	BHI
k-Means-exp	<i>0.7553</i>	0.3591	<i>0.7610</i>	0.2495	<i>0.7436</i>	0.4422	<i>0.7509</i>	0.2643
k-Means-bio	0.7307	<i>0.4981</i>	0.7152	<i>0.5226</i>	0.6862	<i>0.5218</i>	0.6621	<i>0.4627</i>
PAM	0.6390	0.2937	0.6570	0.2541	0.6466	0.2586	0.6535	0.2976
UPGMA	0.6391	0.2540	0.6238	0.2876	0.6496	0.2583	0.6700	0.2774
Dual-MOC-sol-1	0.8522	0.3350	0.7500	0.5994	0.7555	0.6463	0.8774	0.2753
Dual-MOC-sol-2	0.6633	0.6329	0.8440	0.2897	0.8399	0.4107	0.7656	0.6208
Dual-MOC-sol-3	0.7280	0.5753	0.8194	0.5332	0.8365	0.5565	0.8633	0.4182
Dual-MOC-sol-4	0.8467	0.4692	0.7876	0.5454	0.7886	0.6370	0.7853	0.6178
Dual-MOC-sol-5	0.7872	0.5558	0.7769	0.5881	0.8387	0.48618	0.8529	0.4459
Dual-MOC-sol-6	0.6903	0.6227	0.8411	0.2959	0.8102	0.5949	0.8507	0.4774
Dual-MOC-sol-7	0.8352	0.5224	0.8309	0.3998	0.7986	0.5974	0.8352	0.5128
Dual-MOC-sol-8	0.8375	0.4721	0.8236	0.4578			0.7978	0.5828
Dual-MOC-sol-9	0.8178	0.5305	0.8302	0.4371			0.8613	0.4444
Dual-MOC-sol-10	0.7921	0.5549	0.8244	0.4531			0.8140	0.5609
Dual-MOC-sol-11	0.8063	0.5513					0.8332	0.5417
Dual-MOC-sol-12	0.8170	0.5468					0.8430	0.4866
Dual-MOC-sol-13	0.8121	0.5505					0.8262	0.5456
Dual-MOC-sol-14							0.8047	0.5750
Dual-MOC-sol-15							0.8032	0.5779

5.4 Best combination

Here, we show the best α combination according to the HV value achieved. To do this, we use the information in Table 3. Thus, in Figure 3 we show for each dataset, the best set of non-dominated solutions yield by Dual-MOC and the best solutions obtained by competitive algorithms. Here, axes indicate values of both objective functions achieved by the algorithms in their respectively k configuration.

Figure 3: Non-dominated solutions set yield by Dual-MOC and best solutions yield by competitive algorithms. Here, blue circles correspond to non-dominated solutions of our model.

In Figure 3a, we show the results obtained at the run of Dual-MOC where the highest value of HV

Table 7: Objective functions values with $\alpha=0$ in medulloblastoma data (run=14). In italic, we show the best solution in SOC. In bold, we show the Dual-MOC solutions that are better in at least one objective than the best SOC solution. In bold and underline, we show the Dual-MOC solutions better than the best SOC solution.

Algorithms	k=5		k=7		k=9		k=10	
	CI	BHI	CI	BHI	CI	BHI	CI	BHI
k-Means-exp	<i>0.6104</i>	0.3417	<i>0.6071</i>	0.2794	<i>0.6070</i>	0.2880	<i>0.6008</i>	0.1722
k-Means-bio	0.5882	<i>0.5572</i>	0.5918	<i>0.4930</i>	0.5897	<i>0.4800</i>	0.5892	<i>0.4553</i>
PAM	0.5775	0.219	0.5628	0.0878	0.5723	0.1520	0.5673	0.13330
UPGMA	0.5705	0.1769	0.5537	0.0923	0.5797	0.0892	0.5616	0.05228
Dual-MOC-sol-1	0.6194	0.2392	0.5953	0.5784	0.5873	0.6570	0.6442	0.4437
Dual-MOC-sol-2	0.5840	0.7494	0.6183	0.3407	0.6199	0.2837	0.59415	0.7108
Dual-MOC-sol-3	0.5954	0.7492	0.6000	0.4358	0.6069	0.6018	0.6326	0.5211
Dual-MOC-sol-4	0.6006	0.4365	<i>0.6122</i>	0.4223	0.5900	0.6209	0.6178	0.5278
Dual-MOC-sol-5	0.5955	0.6314	0.5991	0.5044	0.6103	0.5623	0.6129	0.6059
Dual-MOC-sol-6	0.5971	0.6022	0.6138	0.4069	0.6130	0.4873	0.6168	0.5426
Dual-MOC-sol-7	0.5992	0.5048			0.6142	0.4365	0.6356	0.4758
Dual-MOC-sol-8	0.6084	0.4338			0.6195	0.3494	0.6401	0.4656
Dual-MOC-sol-9	0.6140	0.4193			0.6172	0.4102	0.6102	0.6307
Dual-MOC-sol-10	0.6140	0.3423			0.6180	0.3579	0.5995	0.6924
Dual-MOC-sol-11	0.6005	0.4961			0.6176	0.3805	0.6064	0.6461
Dual-MOC-sol-12	0.6127	0.4262					0.6023	0.6774
Dual-MOC-sol-13	0.6176	0.2695					0.6052	0.6707
Dual-MOC-sol-14	0.6142	0.3046					0.5955	0.7026
Dual-MOC-sol-15	0.6155	0.2921					0.6053	0.6574
Dual-MOC-sol-16	0.6155	0.3005						
Dual-MOC-sol-17	0.6140	0.3288						

Table 8: Objective functions values with $\alpha=1$ in arabidopsis thaliana data (run=3). In italic, we show the best solution in SOC. In bold, we show the Dual-MOC solutions that are better in at least one objective than the best SOC solution. In bold and underline, we show the Dual-MOC solutions better than the best SOC solution.

Algorithms	k=5		k=7		k=9		k=10	
	CI	BHI	CI	BHI	CI	BHI	CI	BHI
k-Means-exp	<i>0.6854</i>	0.4294	<i>0.7152</i>	0.5050	<i>0.7289</i>	0.4445	<i>0.7380</i>	0.3967
k-Means-bio	0.6445	<i>0.5465</i>	0.6835	<i>0.5612</i>	0.7084	<i>0.5652</i>	0.7229	<i>0.5985</i>
PAM	0.6584	0.3847	0.6692	0.3739	0.6653	0.3244	0.6791	0.3125
UPGMA	0.6584	0.3847	0.6646	0.3244	0.6679	0.2984	0.6823	0.2843
Dual-MOC-sol-1	0.6872	0.6350	0.6962	0.6486	0.7347	0.4677	0.7456	0.3162
Dual-MOC-sol-2	0.6905	0.5222	0.7261	0.3343	0.6991	0.6926	0.7229	0.6735
Dual-MOC-sol-3	0.6877	0.5249	0.7207	0.5529	0.7342	0.5869	0.7447	0.4514
Dual-MOC-sol-4			0.7259	0.4299	0.7274	0.6081	0.7340	0.5711
Dual-MOC-sol-5			0.7258	0.4628	0.7141	0.6682	0.7441	0.5436
Dual-MOC-sol-6			0.7086	0.6232	0.7199	0.6642	0.7427	0.5476
Dual-MOC-sol-7			0.6989	0.6415			0.7447	0.4813
Dual-MOC-sol-8			0.7122	0.6079			0.7239	0.6627
Dual-MOC-sol-9			0.7173	0.5682			0.7329	0.5980
Dual-MOC-sol-10							0.7296	0.6433
Dual-MOC-sol-11							0.7298	0.6414
Dual-MOC-sol-12							0.7447	0.4553

was achieved for *yeast cell cycle* dataset. It can be seen that the solutions in non-dominated set dominates solutions achieved by SOC algorithms. In BHI criterion, all solutions in non-dominated set are better than

Table 9: Objective functions values with $\alpha=0$ in arabidopsis thaliana data (run=3). In italic, we show the best solution in SOC. In bold, we show the Dual-MOC solutions that are better in at least one objective than the best SOC solution. In bold and underline, we show the Dual-MOC solutions better than the best SOC solution.

Algorithms	k=5		k=7		k=9		k=10	
	CI	BHI	CI	BHI	CI	BHI	CI	BHI
k-Means-exp	<i>0.6084</i>	0.4382	<i>0.6174</i>	0.2363	<i>0.6175</i>	0.3603	<i>0.6266</i>	0.3570
k-Means-bio	0.5896	<i>0.7182</i>	0.5995	<i>0.7133</i>	0.6144	<i>0.6922</i>	0.6036	<i>0.6016</i>
PAM	0.5858	0.3506	0.581	0.3178	0.5858	0.3611	0.5910	0.2324
UPGMA	0.5674	0.2805	0.5784	0.3652	0.5767	0.3321	0.5922	0.2313
Dual-MOC-sol-1	0.5996	0.8113	0.6022	0.8226	0.5988	0.7750	0.6491	0.4030
Dual-MOC-sol-2	0.6213	0.6734	0.6299	0.5429	0.6405	0.4661	0.5967	0.7943
Dual-MOC-sol-3	0.6123	0.6983	0.6178	0.7394	0.6362	0.4694	0.6247	0.7843
Dual-MOC-sol-4	0.6086	0.7567	0.6252	0.5768	0.6050	0.7445	0.6019	0.7866
Dual-MOC-sol-5	0.6091	0.7475	0.6101	0.8130	0.6358	0.5663	0.6456	0.4962
Dual-MOC-sol-6			0.6140	0.7566	0.6309	0.6991	0.6416	0.5134
Dual-MOC-sol-7			0.6281	0.5732	0.6161	0.7416	0.6400	0.5918
Dual-MOC-sol-8			0.6113	0.7808	0.6345	0.6257	0.6357	0.6006
Dual-MOC-sol-9					0.6220	0.7183	0.6300	0.7140
Dual-MOC-sol-10					0.6272	0.7166	0.6348	0.6521
Dual-MOC-sol-11					0.6186	0.7411	0.6329	0.6803
Dual-MOC-sol-12					0.6050	0.7445	0.6267	0.7773
Dual-MOC-sol-13							0.6278	0.7425
Dual-MOC-sol-14							0.6393	0.5997
Dual-MOC-sol-15							0.6286	0.7177

the best solutions of *k-means-bio*, *PAM* and *UPGMA*. In CI criterion, most of solutions in non-dominated set dominate to SOC solutions, just two solutions do not (top center) overcome to *k-means-exp*. However, such solutions exceed to this algorithm in BHI criterion. Similarly, in Figure 3b, results to *medulloblastoma metastasis* dataset are shown. Here again, Dual-MOC achieves better results in BHI criterion than SOC algorithms. In fact, all solutions within non-dominated set have the higher values than *k-means-bio*, *PAM* and *UPGMA*. Regarding CI criterion, our model has values in range $[0.75 - 0.85]$ approximately meanwhile SOC algorithm in range $[0.62 - 0.75]$ still evident that solutions in non-dominated set are also better in such case. Finally, in Figure 3c, we show the results for *arabidopsis thaliana* dataset. Here, all solutions in non-dominated set are superior in BHI criterion regarding *PAM* and *UPGMA*. There is only one solution in the approximate Pareto front that does not outperform the solution obtained by *k-means-exp*. Instead, in CI criterion our proposed model has about five solutions that are better than all SOC algorithm.

As it was previously shown, non-dominated sets yield by Dual-MOC consist of a large number of solutions with tradeoffs in both objectives which in many cases are simultaneously better than solutions yield by SOC algorithm in all criteria optimized. Such situation involves that gene clusters discovered by Dual-MOC have higher levels of co-expression and biological coherence than traditional approaches. The selection of a single solution will depend on the instance of the problem and the expertise of the user that is interested on the results. Automatically recommending a single solution in a multi-objective problem remains an open problem and it is an active field of research called Multi-Criteria Decision-Making [53].

5.5 Statistical test

To prove whether Dual-MOC and SOC algorithms perform similarly or not, we have carried out Kruskal-Wallis statistical test [54] followed by Nemenyi post-hoc test [55]. Here, we consider the mean values in each objective function of solutions in non-dominated set with the best combination and SOC algorithms solutions obtained over 20 consecutive runs. In this test, as a null hypothesis, we assume that there is no significant difference between pairwise, whereas the alternative hypothesis is that there is significant difference between them. As *p-value* threshold, we considered an overall 5% significance level. The results in terms of *p-values* are reported in Table 10. Here, we observe that all the *p-values* are lower than 0.0166 (0.05/3). This is strong evidence that Dual-MOC obtains median values of CI and BHI criteria better than SOC algorithms.

Table 10: p-values obtained by Kruskal-Wallis and post-hoc tests comparing algorithms for both objectives.

Pairwise	CI			BHI		
	Cell Cycle	Metastasis	Thaliana	Cell Cycle	Metastasis	Thaliana
Dual-MOC - k-means	1.2e-02	5.8e-05	7.7e-08	1.60e-02	2.5e-03	5.7e-03
Dual-MOC - PAM	1.7e-05	9.8e-03	1.5e-02	1.5e-02	3.0e-03	1.4e-05
Dual-MOC - UPGMA	6.9e-14	8.1e-14	1.5e-02	3.0e-11	1.3e-13	5.2e-14

6 Conclusions and Further Work

This paper proposes a bi-objective algorithm for gene expression data that integrates expression data and external biological knowledge controlled by an α parameter and based on Gene Ontology (GO) annotations. The performance of the clustering proposal has been tested for three real-life gene expression datasets and compared with three widely used single objective clustering algorithms. The highest values of the optimized objective functions are achieved in non-dominated sets with an $\alpha = 0.2$ and $k = 10$ for *yeast cell cycle*, $\alpha = 0.8$ with $k = 9$ for *medulloblastoma metastasis* and $\alpha = 0.1$ and $k = 8$ for *arabidopsis thaliana*.

Results indicate that our bi-objective clustering algorithm achieves higher values of co-expression indicator (*CI*) and Biological Homogeneity Index (*BHI*) than competing algorithms.

The statistical test allows to demonstrate that Dual-MOC outperforms traditional single-objective clustering techniques across a diverse range of data sets. It implies gene clusters found by our model have higher levels of co-expression and biological coherence than benchmark algorithms.

Future work will incorporate local search strategies to improve the solutions found by the algorithm and we will set the α value automatically for each dataset used.

Acknowledgment

JPA thanks the Secretariat for Higher Education, Science, Technology and Innovation (SENESCYT) of Ecuador for its financial support. MIP thanks CEBIB and DICYT project 061619IP, VRIDEI-USACH.

References

- [1] R. Fa, A. Nandi, and L.-Y. Gong, "Clustering analysis for gene expression data: A methodological review," in *Communications Control and Signal Processing (ISCCSP), 2012 5th International Symposium on*, 2012, pp. 1–6. [Online]. Available: <http://dx.doi.org/10.1109/ISCCSP.2012.6217811>
- [2] E. Vaes, M. Khan, and P. Mombaerts, "Statistical analysis of differential gene expression relative to a fold change threshold on nanostring data of mouse odorant receptor genes," *BMC Bioinformatics*, vol. 15, no. 1, p. 39, 2014. [Online]. Available: <http://dx.doi.org/10.1186/1471-2105-15-39>
- [3] T. Lui, N. Tsui, L. Chan, C. Wong, P. Siu, and B. Yung, "Decode: an integrated differential co-expression and differential expression analysis of gene expression data," *BMC Bioinformatics*, vol. 16, no. 1, p. 182, 2015. [Online]. Available: <http://dx.doi.org/10.1186/s12859-015-0582-4>
- [4] Y.-C. Liu, C.-P. Cheng, and V. Tseng, "Mining differential top-k co-expression patterns from time course comparative gene expression datasets," *BMC Bioinformatics*, vol. 14, no. 1, p. 230, 2013. [Online]. Available: <http://dx.doi.org/10.1186/1471-2105-14-230>
- [5] J. Chou, T. Zhou, W. Kaufmann, R. Paules, and P. Bushel, "Extracting gene expression patterns and identifying co-expressed genes from microarray data reveals biologically responsive processes," *BMC Bioinformatics*, vol. 8, no. 1, p. 427, 2007. [Online]. Available: <http://dx.doi.org/10.1186/1471-2105-8-427>
- [6] J. Y. Ban, B. S. Kim, S. C. Kim, D. H. Kim, and J.-H. Chung, "Microarray analysis of gene expression profiles in response to treatment with melatonin in lipopolysaccharide activated raw 264.7 cells," *The Korean journal of physiology & pharmacology : official journal of the Korean Physiological Society and the Korean Society of Pharmacology*, vol. 15, no. 1, pp. 23–29, 2011. [Online]. Available: <http://dx.doi.org/10.4196/kjpp.2011.15.1.23>
- [7] H.-S. Jang, H.-S. Chung, E. Ko, J.-S. Shin, M.-K. Shin, M.-C. Hong, Y. Kim, B.-I. Min, and H. Bae, "Microarray analysis of gene expression profiles in response to treatment with bee venom in

- lipopolysaccharide activated rawx 264.7 cells,” *Journal of Ethnopharmacology*, vol. 121, no. 2, pp. 213–220, 2009. [Online]. Available: <http://dx.doi.org/10.1016/j.jep.2008.09.006>
- [8] A. Igarashi, K. Kobayashi, H. Matsuki, G. Endo, and A. Haga, “Evaluate damage in dna molecules resulting by very-low-frequency magnetic fields using bacterial gene expression system for mutation repairing system,” in *Magnetics Conference, 2005. INTERMAG Asia 2005. Digests of the IEEE International*, 2005, pp. 1145–1146.
 - [9] M. Takahashi and K. Wakabayashi, “Gene mutations and altered gene expression in azoxymethane-induced colon carcinogenesis in rodents,” *Cancer Science*, vol. 95, no. 6, pp. 475–480, 2004. [Online]. Available: <http://dx.doi.org/10.1111/j.1349-7006.2004.tb03235.x>
 - [10] L. Rokach and O. Maimon, “Clustering methods,” in *Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, Eds. Springer US, 2005, pp. 321–352.
 - [11] S. Bandyopadhyay and S. Saha, *Unsupervised classification: similarity measures, classical and meta-heuristic approaches, and applications*. Springer Science & Business Media, 2012.
 - [12] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, “Clustering validity checking methods: Part ii,” 2002. [Online]. Available: <http://doi.acm.org/10.1145/601858.601862>
 - [13] F.-X. Wu, “Genetic weighted k-means algorithm for clustering large-scale gene expression data,” *BMC Bioinformatics*, vol. 9, no. 6, 2008. [Online]. Available: <http://dx.doi.org/10.1186/1471-2105-9-S6-S12>
 - [14] S. Hoque, S. Istyaq, and M. Riaz, “A hierarchical approach for clustering and pattern matching of gene expression data,” in *Genetic and Evolutionary Computing (ICGEC), 2012 Sixth International Conference on*, 2012, pp. 413–416. [Online]. Available: <http://dx.doi.org/10.1109/ICGEC.2012.16>
 - [15] M. Kramer, J. Dutkowski, M. Yu, V. Bafna, and T. Ideker, “Inferring gene ontologies from pairwise similarity data,” *Bioinformatics*, vol. 30, no. 12, pp. i34–i42, 2014. [Online]. Available: <http://dx.doi.org/10.1093/bioinformatics/btu282>
 - [16] S. Muthukalathi, R. Ramanujam, and A. Thalamuthu, “Consensus clustering for microarray gene expression data,” *Bonfring International Journal of Data Mining*, vol. 4, no. 4, p. 26, 2014. [Online]. Available: <http://dx.doi.org/10.1186/gb-2004-5-11-r94>
 - [17] I. Isewon, O. Oyelade, E. Adebisi, and B. Brors, “Ensemble based clustering of plasmodium falciparum genes,” *International Conference on African Development Issues (CU-ICADI)*, 2015.
 - [18] R. Suresh, K. Dinakaran, and P. Valarmathie, “Clustering gene expression data using self-organizing maps,” *Journal of Computer Applications*, vol. 1, no. 4, p. 6, 2008.
 - [19] J. Handl, D. Kell, and J. Knowles, “Multiobjective optimization in bioinformatics and computational biology,” *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, vol. 4, no. 2, pp. 279–292, 2007. [Online]. Available: <http://dx.doi.org/10.1109/TCBB.2007.070203>
 - [20] M. Law, A. Topchy, and A. Jain, “Multiobjective data clustering,” in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2, 2004, pp. II–424–II–430 Vol.2.
 - [21] J. Handl and J. Knowles, “An evolutionary approach to multiobjective clustering,” *Evolutionary Computation, IEEE Transactions on*, vol. 11, no. 1, pp. 56–76, 2007. [Online]. Available: <https://doi.org/10.1109/TEVC.2006.877146>
 - [22] A. Mukhopadhyay and U. Maulik, “Unsupervised pixel classification in satellite imagery using multiobjective fuzzy clustering combined with svm classifier,” *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 47, no. 4, pp. 1132–1138, 2009. [Online]. Available: <http://dx.doi.org/10.1109/TGRS.2008.2008182>
 - [23] U. Maulik, A. Mukhopadhyay, and S. Bandyopadhyay, “Combining pareto-optimal clusters using supervised learning for identifying co-expressed genes,” *BMC Bioinformatics*, vol. 10, no. 1, p. 27, 2009. [Online]. Available: <http://dx.doi.org/10.1186/1471-2105-10-27>
 - [24] A. Mukhopadhyay, U. Maulik, and S. Bandyopadhyay, “Multiobjective genetic algorithm-based fuzzy clustering of categorical attributes,” *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 5, pp. 991–1005, 2009. [Online]. Available: <https://doi.org/10.1109/TEVC.2009.2012163>

- [25] S. Bandyopadhyay, A. Mukhopadhyay, and U. Maulik, "An improved algorithm for clustering gene expression data," *Bioinformatics*, vol. 23, no. 21, pp. 2859–2865, 2007. [Online]. Available: <https://doi.org/10.1093/bioinformatics/btm418>
- [26] A. Mukhopadhyay, U. Maulik, and S. Bandyopadhyay, "An interactive approach to multiobjective clustering of gene expression patterns," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 1, pp. 35–41, 2013. [Online]. Available: <http://dx.doi.org/10.1109/TBME.2012.2220765>
- [27] A. Ferligoj and V. Batagelj, "Direct multicriteria clustering algorithms," *Journal of Classification*, vol. 9, no. 1, pp. 43–61, 1992.
- [28] D. Brockhoff and E. Zitzler, "Objective reduction in evolutionary multiobjective optimization: Theory and applications," *Evol. Comput.*, vol. 17, no. 2, pp. 135–166, 2009. [Online]. Available: <http://dx.doi.org/10.1162/evco.2009.17.2.135>
- [29] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002. [Online]. Available: <http://doi.org/10.1109/4235.996017>
- [30] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proceedings of the National Academy of Sciences*, vol. 95, no. 25, pp. 14 863–14 868, 1998. [Online]. Available: <http://dx.doi.org/10.1073/pnas.96.19.10943-c>
- [31] J. Z. Wang, Z. Du, R. Payattakool, P. S. Yu, and C.-F. Chen, "A new method to measure the semantic similarity of go terms," *Bioinformatics*, vol. 23, no. 10, pp. 1274–1281, 2007. [Online]. Available: <http://doi.org/10.1093/bioinformatics/btm087>
- [32] B.-Y. Kang, S. Ko, and D.-W. Kim, "Sicago: Semi-supervised cluster analysis using semantic distance between gene pairs in gene ontology," *Bioinformatics*, vol. 26, no. 10, pp. 1384–1385, 2010. [Online]. Available: <http://dx.doi.org/10.1093/bioinformatics/btq133>
- [33] D. Pavez-Sandoval, "Incorporación de Anotaciones Génicas en el Algoritmo de Agrupamiento MST-kNN," Master's thesis, Universidad de Santiago de Chile, 2013.
- [34] K. Seridi, L. Jourdan, and E. G. Talbi, "Multi-objective evolutionary algorithm for biclustering in microarrays data," in *2011 IEEE Congress of Evolutionary Computation (CEC)*, 2011, pp. 2593–2599. [Online]. Available: <http://doi.org/10.1109/CEC.2011.5949941>
- [35] F. Luo and J. Liu, "Biclustering of gene expression data with a new hybrid multi-objective evolutionary algorithm of nsga-ii and eda," in *International Conference on Bioinformatics and Biomedical Engineering*, 2008, pp. 1912–1915. [Online]. Available: <http://doi.org/10.1109/ICBBE.2008.807>
- [36] A. Mukhopadhyay and U. Maulik, "Multiobjective approach to categorical data clustering," in *2007 IEEE Congress on Evolutionary Computation*, 2007, pp. 1296–1303. [Online]. Available: <http://doi.org/10.1109/CEC.2007.4424620>
- [37] M. Verbanck, S. Lê, and J. Pagès, "A new unsupervised gene clustering algorithm based on the integration of biological knowledge into expression data," *BMC bioinformatics*, vol. 14, no. 1, p. 42, 2013. [Online]. Available: <http://dx.doi.org/10.1186/1471-2105-14-42>
- [38] S. Datta and S. Datta, "Methods for evaluating clustering algorithms for gene expression data using a reference set of functional classes," *BMC bioinformatics*, vol. 7, no. 1, p. 397, 2006. [Online]. Available: <http://doi.org/10.1186/1471-2105-7-397>
- [39] T. J. MacDonald, K. M. Brown, B. LaFleur, K. Peterson, C. Lawlor, Y. Chen, R. J. Packer, P. Cogen, and D. A. Stephan, "Expression profiling of medulloblastoma: Pdgfra and the ras/mapk pathway as therapeutic targets for metastatic disease," *Nature genetics*, vol. 29, no. 2, pp. 143–152, 2001.
- [40] S. Ghosh, S. Mitra, and R. Dattagupta, "Fuzzy clustering with biological knowledge for gene selection," *Applied Soft Computing*, vol. 16, pp. 102 – 111, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.asoc.2013.11.007>
- [41] R. Alonso, F. Salavert, F. Garcia-Garcia, J. Carbonell-Caballero, M. Bleda, L. Garcia-Alonso, A. Sanchis-Juan, D. Perez-Gil, P. Marin-Garcia, R. Sanchez, C. Cubuk, M. R. Hidalgo, A. Amadoz, R. D. Hernansaiz-Ballesteros, A. Aleman, J. Tarraga, D. Montaner, I. Medina, and J. Dopazo, "Babelomics 5.0: functional interpretation for new generations of genomic data," *Nucleic Acids Research*, 2015. [Online]. Available: <http://doi.org/10.1093/nar/gkv384>

- [42] S. Saha, A. K. Alok, and A. Ekbal, “Use of semisupervised clustering and feature-selection techniques for identification of co-expressed genes,” *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 4, pp. 1171–1177, 2016. [Online]. Available: <http://dx.doi.org/10.1109/JBHI.2015.2451735>
- [43] E. Zitzler and L. Thiele, “Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach,” *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999. [Online]. Available: <http://dx.doi.org/10.1109/4235.797969>
- [44] N. Riquelme, C. V. Lucken, and B. Baran, “Performance metrics in multi-objective optimization,” in *Computing Conference (CLEI), 2015 Latin American*, 2015, pp. 1–11. [Online]. Available: <http://dx.doi.org/10.1109/CLEI.2015.7360024>
- [45] G. Chen, S. A. Jaradat, N. Banerjee, T. S. Tanaka, M. S. Ko, and M. Q. Zhang, “Evaluation and comparison of clustering algorithms in analyzing es cell gene expression data,” *Statistica Sinica*, pp. 241–262, 2002.
- [46] S. Servia-Rodríguez, R. P. Díaz-Redondo, and A. Fernández-Vilas, *Social Computing, Behavioral-Cultural Modeling, and Prediction: 8th International Conference, SBP 2015*. Springer International Publishing, 2015, ch. Are Tweets Biased by Audience? An Analysis from the View of Topic Diversity, pp. 193–202.
- [47] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, 2016.
- [48] A. Lucas, “amap: Another multidimensional analysis package,” *R package*, 2011.
- [49] H. Trautmann, D. Steuer, and O. Mersmann, “mco: Multi criteria optimization algorithms and related functions,” *R Package*, vol. 1, no. 9, 2013.
- [50] C.-S. Tsou and M.-C. Alan, “nsga2r,” *R Package*, 2013.
- [51] G. Yu, F. Li, Y. Qin, X. Bo, Y. Wu, and S. Wang, “Gosemsim: an r package for measuring semantic similarity among go terms and gene products,” *Bioinformatics*, vol. 26, no. 7, pp. 976–978, 2010. [Online]. Available: <http://doi.org/10.1093/bioinformatics/btq064>
- [52] G. Brock, V. Pihur, S. Datta, S. Datta *et al.*, “clvalid, an r package for cluster validation,” *Journal of Statistical Software (Brock et al., March 2008)*, 2011.
- [53] V. Veerappa and E. Letier, “Understanding clusters of optimal solutions in multi-objective decision problems,” in *2011 IEEE 19th International Requirements Engineering Conference*, 2011, pp. 89–98. [Online]. Available: <http://dx.doi.org/10.1109/RE.2011.6051654>
- [54] M. Friedman, “The use of ranks to avoid the assumption of normality implicit in the analysis of variance,” *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.
- [55] P. Nemenyi, “Distribution-free multiple comparisons,” in *Biometrics*, vol. 18, no. 2, 1962, p. 263.