# Differential Biclustering for Gene Expression Analysis

Omar Odibat
Dept. of Computer Science
Wayne State University
Detroit, MI 48202
odibat@wayne.edu

Chandan K. Reddy
Dept. of Computer Science
Wayne State University
Detroit, MI 48202
reddy@cs.wayne.edu

Craig N. Giroux
Karmanos Cancer Institute
Wayne State University
Detroit, MI 48201
cgiroux@wayne.edu

## ABSTRACT

Biclustering algorithms have been successfully used to find subsets of co-expressed genes under subsets of conditions. In some cases, microarray experiments are performed to compare the biological activities of the genes between two classes of cells, such as normal and cancer cells. In this paper, we propose $DiBiCLUS$, a novel **Di**fferential **Biclus**tering algorithm, to identify differential biclusters from the gene expression data where the samples belong to one of the two classes. The genes in these differential biclusters can be positively or negatively co-expressed. We introduce two criteria for any pair of genes to be considered as a differential pair across the two classes. To illustrate the performance of the proposed algorithm, we present the experimental results of applying $DiBiCLUS$ algorithm on synthetic and real-life datasets. These experiments show that the identified differential biclusters are both statistically and biologically significant.

## 1. INTRODUCTION

Measuring the expression level of thousands of genes simultaneously enables several applications such as marker discovery [3], functional annotations and gene networks reconstructions. The availability of such massive data has revolutionized gene expression analysis [17].

The gene expression measurements are organized into two dimensional matrices where rows represent genes and columns represent different conditions. These conditions can be different time points for the same cell, or different types of cells such as tumor and normal cells. The values in any given gene expression data depend on the cell type, the active pathways in the cell and several other factors [7]. Extracting these pathways from the gene expression data is a challenge as different genes are involved in different pathways.

Some of the important goals of gene expression data analysis include clustering the genes, predicting the functions of a gene based on its expression profile, clustering the conditions and classifying a new condition [14]. Differential expression of some genes can cause phenotypic diversity among different conditions [9]. Moreover, the activities of genes are not independent of each other. Therefore, there is a need to study groups of genes rather than performing a single gene analysis. Traditional clustering algorithms, such as $k$-means and hierarchical clustering, have succeeded in clustering gene expression data in many contexts. However, these algorithms fail in some cases when the biological processes are active in subsets of the dimensions of the gene expression data matrix [2]. Missing values and noise add more limitations in applying the traditional clustering techniques on gene expression data [4].

Traditional clustering algorithms assume that related genes should have similar expression profiles in all the conditions [12]. This assumption does not hold in all of the experiments. From the biological perspective, not all the genes are involved in each biological pathway. Furthermore, some of these pathways may be active under a subset of the conditions [14]. Therefore, traditional clustering techniques do not capture such pathways.

## 2. DIFFERENTIAL BICLUSTERING

First proposed by Cheng and Church [4], the goal of biclustering algorithms is to overcome the limitations of traditional clustering algorithms [17]. Biclustering can be used to find a subset of genes that have similar expression profiles under a subset of the conditions [12, 13] as defined in Definition 1. Identifying a subset of genes that are related under a subset of conditions is an important challenge. It was proved that the problem of finding significant biclusters is an NP-hard problem [4].

DEFINITION 1. *Given a gene expression dataset, $D = (G, C)$, where $G$ is the set of genes, and $C$ is the set of conditions, a bicluster $B$ is defined as a subset of genes that are highly related to each other under a subset of conditions i.e., $B = (I, J)$ where $I \subseteq G$ and $J \subseteq C$.*

Biclustering was applied in the classification of cancer subtypes [11]. Since some of the pathways are active only across patients of a certain cancer subtype, the genes in such pathways can be used as markers to classify cancer subtypes. Moreover, biclustering was proposed to identify transcriptional modules from the gene expression data [8].

**Differential Biclustering** aims to find gene sets that are correlated under a subset of conditions in one class of conditions but not in the other class. One of the main applications


*ACM-BCB* 2010, Niagara Falls, NY, USA

of DNA microarray data is to compare the biological activities of the genes in two types of cells, such as normal and disease cells [12]. There are many cases where the conditions belong to two classes, and there is a need to find the set of significant genes for each class. For instance, African American males (AAM) have a higher risk of developing prostate cancer compared to Caucasian American males (CAM) [15]. There are several hypotheses to explain this racial difference. One of them is based on the assumption that genetic factors may play a key role in this difference between the two groups. In this case, differential biclustering can be used to identify the genes that are responsible for the difference between AAM and CAM in developing prostate cancer.

In this paper, we propose $DiBiCLUS$ algorithm to extract differential biclusters from the gene expression data where the samples belong to one of the two classes. The goal of this algorithm is different from the goal of the traditional biclustering algorithms. The genes in the differential biclusters have strong correlation in one class but not in the other, or they may have different types of co-expression among the two classes. Comparing the biological roles of genes in two classes of cells is an important problem to identify the genes are responsible for the phenotypic change. The goal of the proposed algorithm is to discover such genes in the gene expression data. The $DiBiCLUS$ framework has the following novel contributions that distinguish it from existing biclustering algorithms:

1. Incorporating the class labels in the process of identifying the biclusters.
2. Efficient detection and discovery of differential biclusters, which are relevant in comparing two different types of cells.
3. Using clustering algorithms to quantize the expression values of each gene.
4. The ability to find possibly overlapping biclusters. The overlapping can be within the same class or across two classes.

To the best of our knowledge, there is no existing biclustering algorithm that has all of the above properties. However, these properties are necessary for identifying useful patterns in several applications.

# 3. DIBICLUS FRAMEWORK

## 3.1 Preliminaries

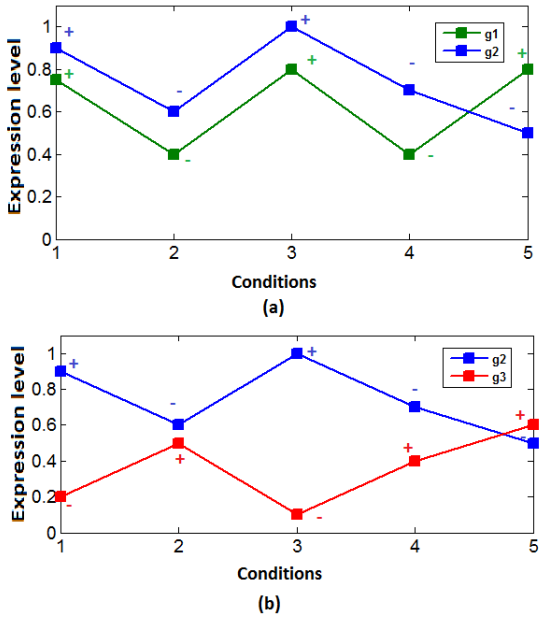
Usually, a gene expression data ($D$) is organized as a $M \times N$ matrix, where rows correspond to genes and columns correspond to conditions. When there are two classes of experimental conditions, gene expression data can be organized in the form of two matrices. Both of the matrices have the same set of genes $G = \{g_1, g_2, ..., g_M\}$. The first matrix, $A$, contains the expression values of the genes under the first class of conditions $C_A = \{c_1, c_2, ..., c_{N_A}\}$, and the second matrix, $B$, contains the expression values of the genes under the second class of conditions $C_B = \{c_1, c_2, ..., c_{N_B}\}$. $N_A$ and $N_B$ are the number of conditions belonging to class $A$ and $B$ respectively ($N = N_A + N_B$). Table 1 shows the structure of such a dataset.

**Table 1: Gene expression data for two classes**

| Genes | Conditions in class A | | | Conditions in class B | | |
|---|---|---|---|---|---|---|
| | $c1_A$ | ... | $c_{N_A}$ | $c1_B$ | ... | $c_{N_B}$ |
| $g_1$ | $d_{1,1}$ | ... | $d_{1,N_A}$ | $d_{1,1}$ | ... | $d_{1,N_B}$ |
| $g_2$ | $d_{2,1}$ | ... | $d_{2,N_A}$ | $d_{2,1}$ | ... | $d_{2,N_B}$ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| $g_M$ | $d_{M,1}$ | ... | $d_{M,N_A}$ | $d_{M,1}$ | ... | $d_{M,N_B}$ |

Similar patterns of gene expression profiles indicate relationships between the genes. Mainly, there are two types of relationships, positive co-expression and negative co-expression. In our algorithm, we implement these relationships using positive and negative numbers as follows: for each gene, over-expression conditions are represented by positive numbers, and under-expression conditions are represented by negative numbers. Two genes are considered positively co-expressed if they have the same signs in a subset of conditions, and they considered negatively co-expressed if they have different signs in a subset of conditions. Figure 1 shows an example of the two types of relationships between different genes.

**Figure 1: An example of different types of gene co-expressions. (a) two gene profiles with positive co-expression in 80% of the conditions (b) two gene profiles with negative co-expression in 80% of the conditions. + (-) sign indicates over-expression (under-expression) in each condition.**

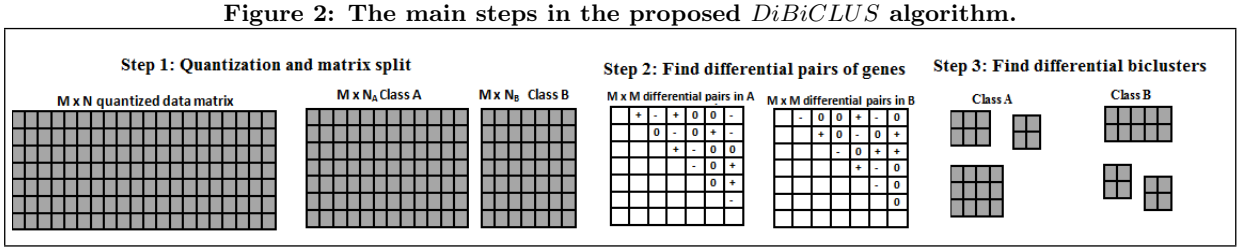



## 3.2 Overview of the DiBiCLUS Algorithm

In this paper, we propose a new approach to identify differential biclusters from gene expression data. The main steps of this approach are shown in Figure 2. These steps are briefly described here:

- **Step 1** *Quantization:* the goal of this step is to create a new representation of the gene expression data. We use a clustering algorithm to create the new representation of the expression values. In this paper, $k$-means

**Figure 2: The main steps in the proposed $DiBiCLUS$ algorithm.**




clustering algorithm is used to cluster the expression values of each gene. Each cluster will be represented by a single value. After clustering, the matrix is split based on the class label of the experimental conditions. The results of this step are two matrices with the same number of genes, but the number of conditions in each matrix can be different.

- **Step 2** *Identifying the differential pairs of genes in each class:* First, the similarity between each pair of genes in each class is computed in a square matrix $M \times M$. The similarity score depends on the number of conditions under which the pair of genes have the same value in the new representation. Second, the type of co-expression between each pair of genes is determined. The co-expression type can be either positive or negative. Finally, for each pair, the ratio of the number of conditions, under which the pair of genes have the same value, to the total number of conditions in class $A$ is compared to the same ratio in class $B$. Based on this comparison and the co-expression type, each pair of genes can be assigned to class $A$ or class $B$.

- **Step 3** *Identifying the differential biclusters:* for a given gene, and from the differential pairs identified in the previous step, the algorithm finds groups of genes that have similar expression values under some conditions. The groups of genes and conditions form the differential biclusters.

## 3.3 Quantization of Gene Expression Data

There are several ways that have been proposed to create a new representation of the gene expression data; such as ranking the expression values of each gene [1] and representing the gene expression data with two values, 1 and 2. All the values that are larger than or equal to the mean of the expression data will be represented by 2, while the other values will be represented by 1 [18]. In this paper, we use a clustering algorithm to create the new representation.

The goal of applying a clustering algorithm is to guarantee that similar expression values will be represented by the same value. Several clustering algorithms can be applied to achieve the above goal. We use $k$-means algorithm to cluster the expression values of each gene. Each cluster is represented by a single integer value. As an illustration of the performance of the new representation, we compare it with the other methods mentioned earlier.

In Table 2, the 10 values of a gene are represented using different methods. R1 is the traditional ranking method, in which the values of each gene are represented by the numbers from 1 to $N$, where $N$ is the number of experimental

**Table 2: Comparison of some gene representation methods (R1, and R2) with the clustering method (R3, R4 and R5) of $DiBiCLUS$ algorithm. The first column has 10 expression values of a gene, and each other column indicates a new representation of the corresponding expression value.**

| Gene values | R1 | R2 | R3 | R4 | R5 |
|---|---|---|---|---|---|
| 0.35 | 1 | 1 | 1 | 1 | -1 |
| 0.36 | 2 | 1 | 1 | 1 | -1 |
| 0.37 | 3 | 1 | 1 | 1 | -1 |
| 0.93 | 4 | 1 | 2 | 2 | 0 |
| 0.99 | 5 | 2 | 2 | 2 | 0 |
| 1.2 | 6 | 2 | 2 | 3 | 1 |
| 1.29 | 7 | 2 | 2 | 3 | 1 |
| 1.3 | 8 | 2 | 2 | 3 | 1 |
| 1.36 | 9 | 2 | 2 | 3 | 1 |
| 1.37 | 10 | 2 | 2 | 3 | 1 |

conditions. The second method, R2, represents the gene expression data with two values: 1 and 2. All the values that are more than or equal to the mean of the expression data will be represented by 2, the other values will be represented by 1 [18]. R3 is the result of $k$-means algorithm with $k = 2$, and R4 is the $k$-means algorithm with $k = 3$. R5 is the same as R4, but the values are shifted so that the middle cluster is represented with zeros.

Using the ranking method, R1, the relative differences between consecutive ranked values are not captured. For instance, 0.37 is represented with 3, and 0.93 is represented with 4. The difference between the two values is $0.93-0.37 = 0.56$. This difference is treated in the same way as the difference between 0.37 and 0.36 which is 0.01. R2 represents the first four values by 1 because the mean of the gene values is 0.95. However, this method assigns 0.37 and 0.93 the same value, 1. Using the proposed quantization method, with proper $k$ value, the above problems are resolved by representing similar gene values with the same value. Using $k = 2$, as shown in R3 column in the table, $0.35, 0.36$ and 0.37 are represented with the same value, 1, and the remaining values are represented by 2. Finally, using $k = 3$, an improved representation is obtained by putting 0.93 and 0.99 in a separate cluster as shown under R4 column.

The values of $k$ can be $1 < k \leq N$ , where $N$ is the total number of conditions in the data (including both classes). When $k = N$, the result of the clustering will be the same as ranking the gene expression data. In our algorithm we use odd values of $k$, so that the middle cluster will be represented with 0's, and half of the remaining clusters will be represented by positive numbers: $1, 2, ..., k/2$, and the sec-

ond half of the remaining clusters will be represented by negative numbers: $-k/2, -k/2+1..., -1$. The mapping between a certain cluster and the new representation is based on the rank of the cluster; hence, the middle cluster is represented by 0. This is shown in the column $R5$ of Table 2.

In $k$-means clustering, the initial centers of the clusters are randomly chosen. Therefore, running the same algorithm several times may produce different representation of the gene values, and different outputs of the algorithm. To minimize the effects of this randomness issue, $k$-means will be run $s$ times for each gene, and then Sum of Squared Error (SSE) will be used as a measure to select the best result for the clustering algorithm. We used $s = 10$ in all the experiments.

## 3.4 Identifying the Differential Pairs of Genes

In order to find the differential pairs of genes in each class, the similarity between each two genes is computed first as in Definition 2. It is worth mentioning that using the new representation of the gene expression values, the zeros are ignored and are not included in computing the similarity between any two genes.

DEFINITION 2. *Given two genes $g_i$ and $g_j$, $common(g_i, g_j)$ is the set of identical non-zero conditions in a given class for the two genes $g_i$ and $g_j$. The similarity between any two genes, $sim(g_i, g_j)$, is defined as:*
$sim(g_i, g_j) = |common(g_i, g_j)|/N_x$
*where $N_x$ is the number of conditions in class $x$.*

To compute the negative co-expression, Definition 2 is slightly modified by taking into account the signs in the quantized gene expression data. In other words, and for negative co-expression, the common set is defined as the set of identical non-zero conditions with opposite signs.

Since it is possible for any two genes to be positively co-expressed in a subset of conditions and negatively co-expressed in another subset of conditions of the same class, special handling is needed. The similarity between each two genes is computed using Definition 2 twice: one for the positive co-expression and one for the negative co-expression. The maximum of the two values is considered as the final value for the similarity measurement.

The goal of differential biclustering is to emphasize the differences between two classes of conditions. In our algorithm, we introduce a differential biclustering algorithm to identify differential co-expressed gene sets. Only a subset of conditions is considered in identifying gene relationships. Two genes are considered as a differential pair if one of the following two criteria is met. The first criterion is based on the relationship type between the two genes in each class. If the two genes are positively co-expressed in one class, and negatively co-expressed in the other class, this pair of genes is considered as a differential pair.

The second criterion is based on the number of conditions under which the two genes are related. A pair of genes is considered as a differential pair if it is significantly related in one of the classes more than in the other class. The level of significance is defined using a user-defined parameter, $\delta$ (a similar criterion was presented in [5]). Definition 3 is used to determine the set of differential pairs of genes in each class.

DEFINITION 3. *Given a gene expression dataset $D$, that has $M$ genes and $N$ conditions, $N_A$ and $N_B$, the number of conditions belong to class $A$ and $B$ respectively; then, two genes $g_i$ and $g_j$ are considered a differential pair if one of the following criteria is met:*

- *The co-expression type of $g_i$ and $g_j$ in class $A$ is not the same as the co-expression type of $g_i$ and $g_j$ in class $B$.*
- *if $sim_A(i,j)$ indicates the similarity between $g_i$ and $g_j$ in class $A$, and $sim_B(i,j)$ indicates the similarity between $g_i$ and $g_j$ in class $B$, then either $\frac{sim_A(i,j)}{N_A} > \frac{sim_B(i,j)}{N_B} + \delta$ or $\frac{sim_B(i,j)}{N_B} > \frac{sim_A(i,j)}{N_A} + \delta$.*

**Algorithm 1** *chkBiclus*

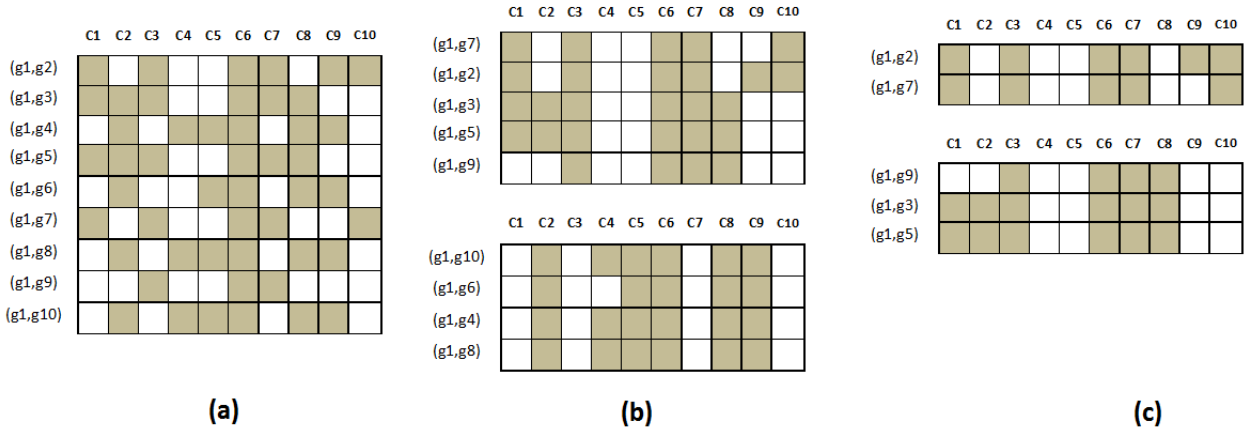
**Input:** $gSet$: a group of genes.
**Output:** $DiffBiclus$: differential bicluster(s) in $gSet$
**Algorithm:**
**if** $|gSet| < minG$ **then**
  **return** {}
**end if**
Using Definition 2, get $cSet$ the common conditions for the genes in $gSet$
**if** $|cSet| \geq minC$ **then**
  $DiffBiclus = \{gSet, cSet\}$
**else**
  Determine the two genes with the lowest similarity
  $[g_k, g_l] = min\{sim(g_i, g_j), i, j \in gSet\}$
  Create two partitions of the gene set
  $p1 = \{g_k\}$
  $p2 = \{g_l\}$
  Divide the genes into one of the partitions
  **for** $i = 1 : |gSet|, i \neq k$ and $i \neq l$ **do**
    **if** $sim(g_k, g_i) > sim(g_l, g_i)$ **then**
      $p1 = p1 \cup \{g_i\}$
    **else**
      $p2 = p2 \cup \{g_i\}$
    **end if**
  **end for**
  **if** $|p1| \geq minG$ **then**
    $DiffBiclus = DiffBiclus \cup chkBiclus(p1)$
  **end if**
  **if** $|p2| \geq minG$ **then**
    $DiffBiclus = DiffBiclus \cup chkBiclus(p2)$
  **end if**
**end if**

In this paper, the $\delta$ parameter, $0 < \delta < 1$, was set to 25% in all of the experiments. Low values of this parameter relax the second criterion in Definition 3, and more pairs of genes will be considered as differential pairs. On the other hand, high values of the $\delta$ parameter has the opposite effect. However, a good approach to set this parameter is by selecting a high value at the beginning; then, keep reducing its value until a certain number of differential biclusters are obtained.

## 3.5 Identifying the Differential Biclusters

The differential pairs of genes are the main building blocks of the differential biclusters as defined in Definition 4. Algorithm 1 is used to discover differential bicluster(s) from a

**Figure 3: Illustration of the divisive approach in the *chkBiclus* algorithm. In this example, $g_1$ is considered as the seed, the goal is to find any possible differential biclusters that contains $g_1$. (a) $2D$ matrix shows 9 differential pairs and the set of common conditions for each pair of genes. (b) and (c) show the results of the first and the second iterations of the algorithm.**




group of genes that are related to a seed gene. If there are no known seed genes, the algorithm iterates over all of the genes, each time considering one of the genes as the seed gene. The size of the conditions and the size of the genes are checked against the user-defined thresholds, $minC$ and $minG$, which are the minimum number of genes and the minimum number of conditions, respectively. If the number of common conditions for a given set of genes is greater than $minC$, then the set of genes and the common conditions are considered as a bicluster. Otherwise, the set of genes are divided into two partitions in which each partition is checked recursively. The partitioning is performed based on the similarity between the genes.

DEFINITION 4. *Given a gene expression dataset with two classes, a differential bicluster is a subset of differential gene pairs that are highly related to each other under a subset of conditions in one class but not in the other class.*

To illustrate the divisive approach followed in the *chkBiclus*, we introduce an example in Figure 3. In this example, $g_1$ is considered as the seed, the goal is to find any possible differential biclusters that contain $g_1$. In Figure 3(a), a $2d$ matrix shows 9 differential pairs and the set of common conditions for each pair of genes. For instance, $g1$ and $g2$ have similar non-zero conditions: $c1, c3, c6, c7, c9$ and $c10$. Assuming that both $minG$ and $minC$ are set to 4. The number of common conditions for all of the 9 pairs is one, which is $c6$. Since this is less than $minC$, and the number of genes is more than $minG$, a split of the genes should be performed. To do the split, the most dissimilar pairs of genes are identified, which are $(g1, g7)$ and $(g1, g8)$. Then, two partitions are created. Initially, the first partition will contain $(g1, g7)$, and the second partition will contain $(g1, g8)$. Each of the remaining 7 pairs will be added to the first partition if it is more similar to $(g1, g7)$ than to $(g1, g8)$, otherwise it will be added to the second partition. The result is shown in Figure 3(b).

The bicluster in the bottom part of Figure 3(b) is composed of $\{g1, g4, g6, g8, g10\}$ and $\{c2, c5, c6, c8, c9\}$. This partition satisfies $minG$ and $minC$ thresholds and thus this partition is not processed further. The top partition is composed of $\{g1, g2, g3, g5, g7, g9\}$ and $\{c3, c6, c7\}$. Since the number of conditions is less than $minC$, this partition will be divided into two new partitions as shown in Figure 3(c). The top partition is composed of $\{g1, g2, g7\}$ and $\{c1, c3, c6, , c7, c10\}$. This partition does not satisfy the $minG$ threshold, and thus will be ignored. The bottom partition satisfies the $minG$ and $minC$ thresholds. Therefore, the final biclusters are $(\{g1, g4, g6, g8, g10\}, \{c2, c5, c6, c8, c9\})$ and $(\{g1, g3, g5, g10\}, \{c3, c6, c7, c8\})$. *DiBiCLUS* is summarized in Algorithm 2. The complexity analysis of this algorithm is $O(M^4 N_{max})$ in the worst case scenario, where $N_{max} = max(N_A, N_B)$.

**Algorithm 2** *DiBiCLUS*

**Input:** $D = M \times N$ : an expression dataset,
$L = 1 \times N$: a binary vector containing the class labels for the conditions in $D$,
$\delta$: significance threshold, $k$: parameter for $k$-means
**Output:** $DiBiCLUS_A$ and $DiBiCLUS_B$ the set of differential biclusters in classes $A$ and $B$ respectively.
**Algorithm:**
Quantize each gene in D using the $k$-means algorithm
$D_Q(i) = Clus\_gene(D(i), k), 1 \leq i \leq M$
Split $D_Q$ into $data_A$ and $data_B$ using $L$
$data_A = D_Q(L = 0)$
$data_B = D_Q(L = 1)$
Determine the set of differential pairs in each class $diff_A$ and $diff_B$ using Definition 3
Using each gene as a seed, check for differential biclusters
**for** $i = 1 : M$ **do**
  $genes_A = diff_A(i)$
  $genes_B = diff_B(i)$
  $DiBiCLUS_A = chkBiclus(genes_A)$
  $DiBiCLUS_B = chkBiclus(genes_B)$
**end for**

# 4. EXPERIMENTAL RESULTS

## 4.1 Synthetic Dataset

To illustrate the performance of the proposed algorithm, we applied it on a synthetic dataset. This dataset is composed of 50 genes, 20 conditions in class $A$ and 20 conditions in class $B$. The datasets and the results are shown in Figure 4. The heat maps were generated using the Heatmap Builder tool [10]. We implanted different types of biclusters based on the following scenarios. The parameters of the algorithm were as follows: $\delta = 25\%, minG = 10, minC = 10$ and $k = 5$.

- Scenario (1): a $10 \times 10$ positively co-expressed bicluster was implanted in class $A$, and no biclusters were implanted in class $B$ as shown in Figure 4(a). In this scenario, the genes are related under a subset of conditions in class A but are not related in any subset of conditions in class $B$. Therefore, the second criterion in Definition 3 is satisfied, and this bicluster is considered to be a differential bicluster. The algorithm outputs this bicluster as shown in Figure 4(e).
- Scenario (2): a $10 \times 10$ positively co-expressed bicluster was implanted in class $A$, and another $10 \times 10$ positively co-expressed bicluster was implanted in class $B$ as shown in Figure 4(b). Both biclusters have the same subset of genes, but different subset of conditions. In this case none of these biclusters is considered a differential bicluster because the same subset of genes are related in both classes with the same type of co-expression and under the same ratio of conditions (50% of the conditions in each class). Therefore, none of the criteria in Definition 3 is satisfied, and the $DiBiCLUS$ algorithm does not output any bicluster.
- Scenario (3): a $10 \times 10$ positively co-expressed bicluster was implanted in class $A$, and another $10 \times 10$ negatively co-expressed bicluster was implanted in class $B$ as shown in Figure 4(c). In this case both of these biclusters are considered differential biclusters because the same subset of genes are related in a different type of co-expression in each class. The first criterion in Definition 3 is satisfied, and the two biclusters are shown in Figure 4(f) and Figure 4(g). The expression profile of the same seed gene is highlighted in both classes. The relationship between this gene and each of the remaining genes is different; thus, this gene formed a differential pair with each of the remaining genes in both classes.
- Scenario (4): a $10 \times 15$ positively co-expressed bicluster was implanted in class $A$, and another $10 \times 5$ positively co-expressed bicluster was implanted in class $B$ as shown in Figure 4(d). This scenario is similar to the scenario 2, with the exception that the subset of conditions, under which the subset of genes is related, is 75% of the conditions in class $A$ and 25% of the conditions in class $B$. The difference between the two ratios, 50%, is larger than the $\delta$ parameter; as a result, the second criterion in Definition 3 is satisfied, and the bicluster in class $A$ is considered to be a differential bicluster as shown in Figure 4(h).

## 4.2 The Prostate Cancer Dataset

In the second experiment, we used a real-life dataset, the prostate cancer dataset. This dataset has two classes of conditions. The first class is an early stage of prostate cancer (class $A$), and the second class is a developed stage of prostate cancer (class $B$). The total number of experimental conditions is 641. 433 of the conditions belong to class $A$, and the remaining 208 conditions belong to class $B$. The set of genes were filtered, and the number of genes was reduced to 529. This set of 529 genes was nominated as a focused set of candidate prostate cancer related genes, based on domain knowledge from the literature and previous laboratory-based studies. Furthermore, all of the genes were known to be expressed in prostate tumors.

Table 3 shows the results of applying $DiBiCLUS$ on the prostate cancer dataset. The parameters were set as follows: $\delta = 25\%, minG = 5$ and $minC = 10$. Several values of $k$ were used. $GSEA$, Gene Set Enrichment Analysis, is a computational application that evaluates the significance concordant differences between two biological states of a given set of genes [16]. This tool was used to identify the enriched gene sets. Low p-values indicates biologically relevant genes. Figure 5 shows four differential biclusters obtained by applying $DiBiCLUS$ on the prostate cancer dataset. The parameters were set as follows:$\delta = 25\%, minG = 10, minC = 10$ and $k = 7$.

**Table 3: The number of statistically significant biclusters obtained by applying $DiBiCLUS$ on the prostate cancer dataset.** $\delta = 25\%, minG = 5, minC = 10$.

| | Class A | | Class B | | Time |
|---|---|---|---|---|---|
| k | $p < 5\%$ | $p < 1\%$ | $p < 5\%$ | $p < 1\%$ | in minutes |
| 3 | 69 | 14 | 34 | 7 | 1.58 |
| 5 | 23 | 4 | 3 | 0 | 1.72 |
| 7 | 2 | 0 | 0 | 0 | 1.95 |

The prostate cancer dataset, used in this experiment, has a special characteristic: the number of conditions is more than the number of genes. Therefore, more time is needed to quantize the gene expression data than to find the differential biclusters in the next steps. The running time is shown in the last column of Table 3. A 2.83 GHz PC with 8 GB RAM was used to perform our experiments. Figure 7 shows the shape of different differential biclusters obtained with different values of $k$. As the value of $k$ increases, the expression profiles of genes get closer to each other and the number of differential biclusters decreases.

As an example of the type of gene relationships captured by $DiBiCLUS$ algorithm, Figure 6 shows the expression profile of three genes ACTA2, MTA1 and DVL3. The expression of the first gene is positively covariant with the other two genes in class $A$, while it is negatively covariant with the two genes in class $B$. To evaluate the relationships between the three genes, we used the IPA Knowledge Base [6].

In Figure 8, the three covariant genes are shown to be mapped to a closely related local sub-network in the IPA biological interaction Knowledge Base. This mapping result suggests that these three genes function in closely related

**Figure 4: The synthetic datasets and the results of the $DiBiCLUS$ algorithm. The first row shows the four scenarios of the synthetic datasets using heat maps. The second row shows the results of $DiBiCLUS$. The result of the algorithm on the dataset in (a) is shown in (e). The algorithm does not produce any result for the dataset in (b). The results of the algorithm on the dataset in (c) are shown in (f) and (g). The result of the algorithm on the dataset in (d) is shown in (h).**

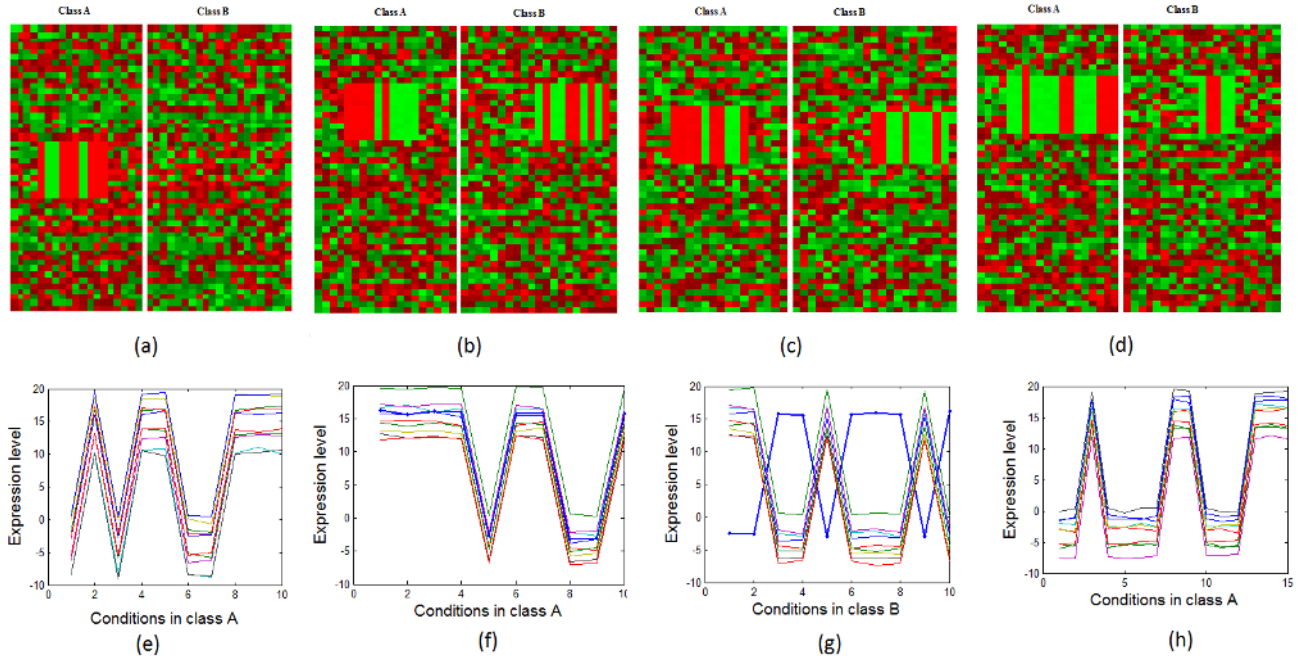



**Figure 5: Some of the differential biclusters obtained by applying $DiBiCLUS$ on the prostate cancer dataset. (a) class = A, number of genes= 13 and number of conditions= 10. (b) class = A, number of genes= 11 and number of conditions= 15. (c) class = B, number of genes= 7 and number of conditions= 15. (d) class = B, number of genes= 7 and number of conditions= 27.**

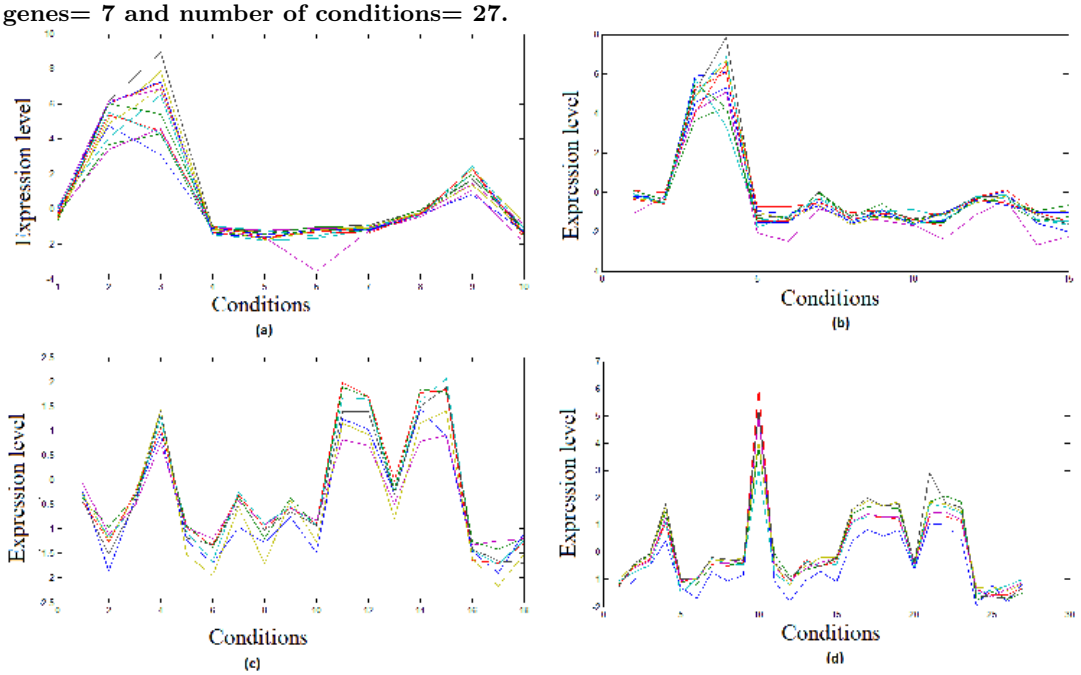


**Figure 7: The effect of changing the parameter $k$ on the shape of the biclusters obtained by applying $DiBiCLUS$ on prostate cancer dataset.**

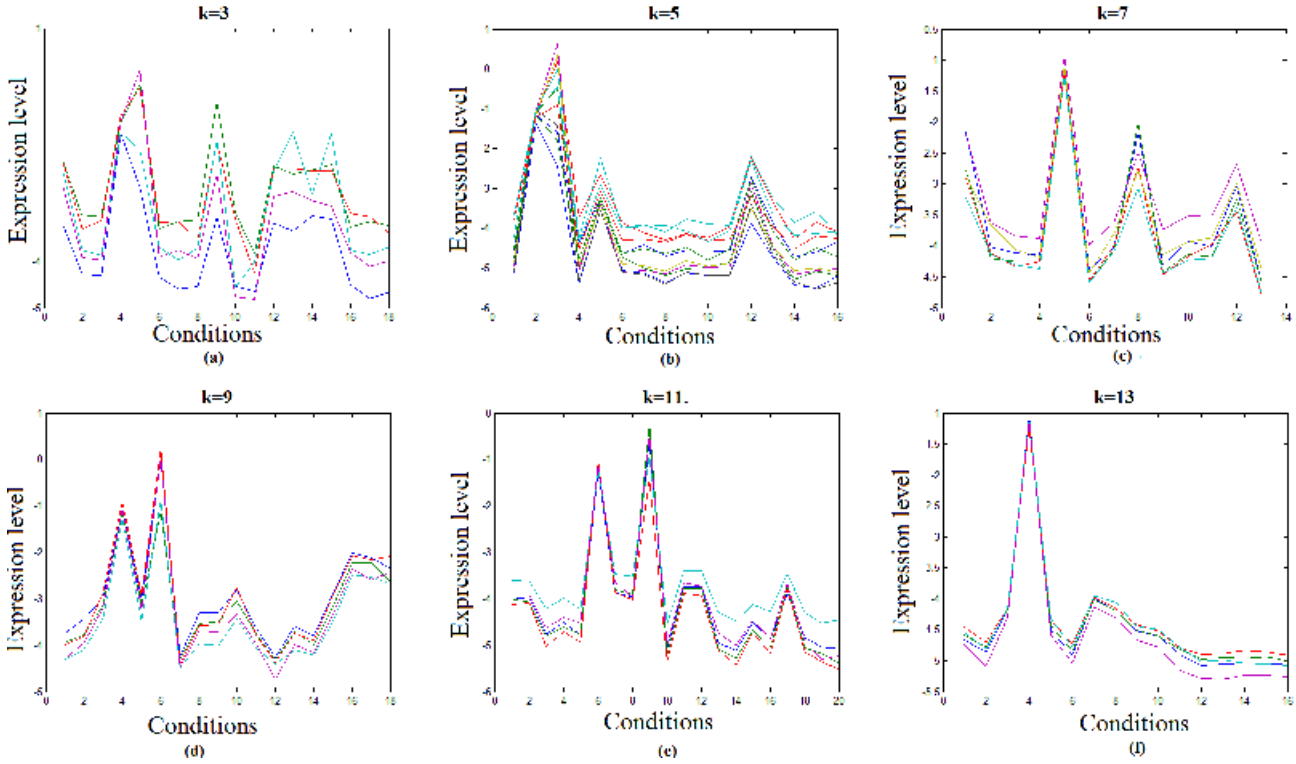
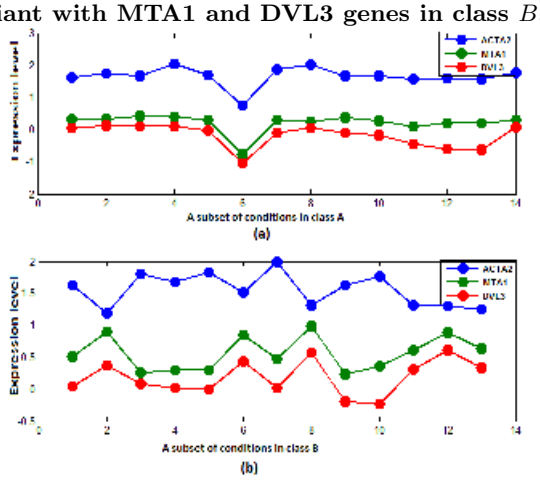



**Figure 6: The expression profile of three genes ACTA2, MTA1 and DVL3. (a) ACTA2 expression is positively covariant with MTA1 and DVL3 genes in class $A$. (b) ACTA2 expression is negatively covariant with MTA1 and DVL3 genes in class $B$.**




biological processes, associated with the aggressive state of prostate cancer. This computational prediction needs to be confirmed experimentally, but it is significant that the covariant gene cluster maps to a biological pathway that has been independently implicated as a determinant of aggressive prostate cancer.

Biological analysis of the bicluster in Figure 5(c) showed that the bicluster is biologically relevant. This bicluster is associated with aggressive tumors, which are driven by androgen (hormone) signals. The associated biological network shows enrichment for gene functions required for steroid hormone metabolism and endocrine system development, which is highly relevant since prostate cancer is a hormonally drive tumor type. This is shown in Figure 9.

## 4.3 The Lung Cancer Dataset

In the third experiment, we applied $DiBiCLUS$ on lung cancer dataset obtained from [5]. This dataset is composed of 1975 genes and 169 samples: 102 are cancer samples (class $A$), and 67 normal samples (class $B$). Table 4 shows the result of applying $DiBiCLUS$ on the lung cancer dataset. The parameters were set as follows: $\delta = 25\%, minG = 10$ and $minC = 10$. Figure 10 shows samples of the differential biclusters obtained by applying $DiBiCLUS$ on the lung cancer dataset. In the first bicluster, one of the genes, displayed in blue in Figure 10 (a), is negatively co-expressed with the other genes in the same bicluster. For a given gene, $DiBiCLUS$ finds the genes that are positively or negatively co-expressed with that gene.

The running time for applying $DiBiCLUS$ on the lung

**Figure 8: Pathway obtained from IPA Knowledge Base for the genes ACTA2, MTA1 and DVL.**

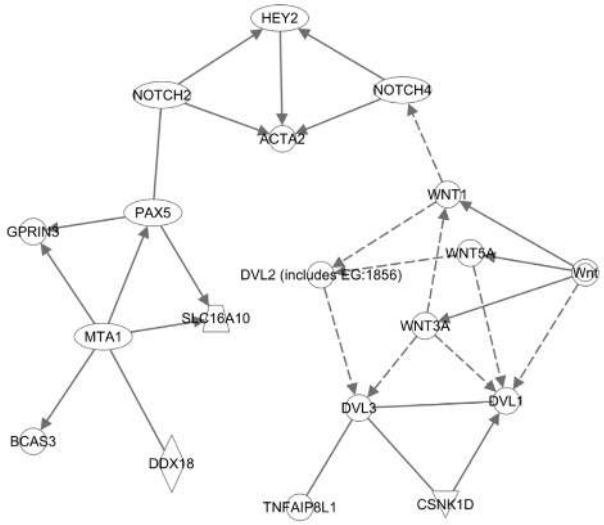
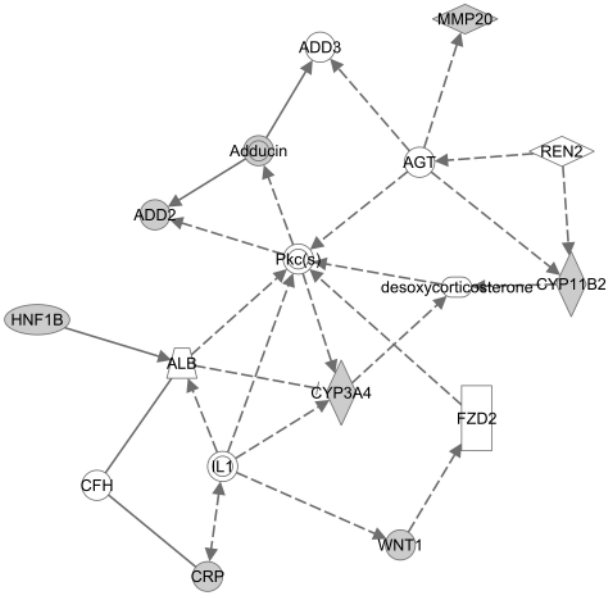



**Figure 9: Pathway obtained from IPA Knowledge Base for the bicluster shown in Figure 5(c). Shaded genes are focus genes from the bicluster, and un-shaded genes are predicted network connected genes.**




cancer dataset with different values of $k$ is shown on the last column of Table 4. Here the case is completely different than the prostate cancer dataset. In the lung cancer dataset, and in most of the other cases, the number of genes is much larger than the number of conditions. As the value of k increases, the number of differential pairs decreases, and the running time decreases.

**Table 4: The number of statistically significant biclusters obtained by applying $DiBiCLUS$ on the lung cancer dataset. $\delta = 25\%, minG = 10$ and $minC = 10$.**

| k | Class A $p < 5\%$ | Class A $p < 1\%$ | Class B $p < 5\%$ | Class B $p < 1\%$ | Time in hours |
|---|---|---|---|---|---|
| 3 | 278 | 107 | 112 | 46 | 6.83 |
| 5 | 14 | 4 | 39 | 18 | 2.04 |
| 7 | 0 | 0 | 3 | 1 | 0.49 |

## 4.4 Discussion

In the proposed algorithm, we identify the differential biclusters. The genes in these differential biclusters have strong correlation in one class but not in the other, or they may have different types of co-expression among the two classes. Furthermore, the genes can be negatively or positively co-expressed. A key feature of the $DiBiCLUS$ algorithm is that it incorporates the class labels in finding the differential biclusters. Therefore, trivial biclusters, that are similarly co-expressed in both classes, are ignored by the algorithm. As a result, biologists can focus on these differential biclusters to understand the differences between two classes of cells.

One of the advantages of the proposed algorithm is that it allows overlapping. The same gene can be a member of more than one bicluster. Biologically, a gene may be involved in more than one biological pathway. In addition, a gene can be a member of biclusters in both classes, $A$ and $B$, which helps in understanding the different roles played by this gene in two different kinds of cells.

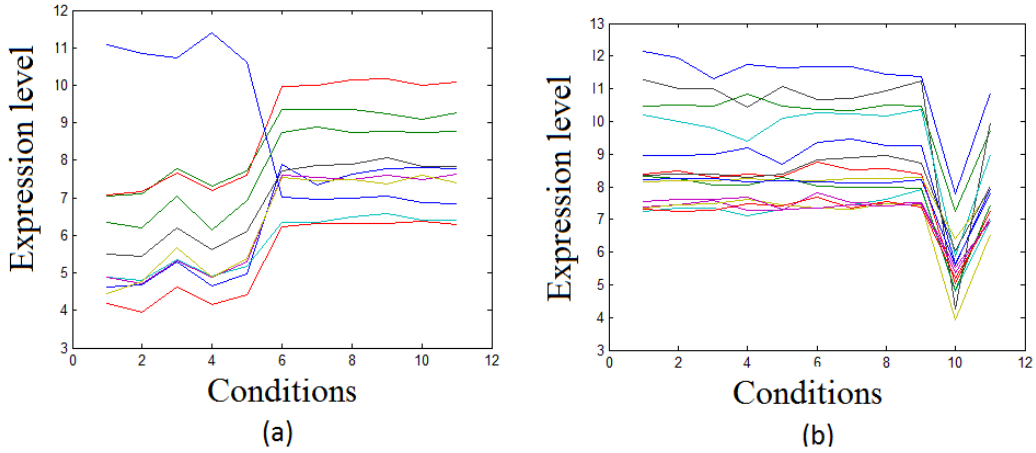
Another advantage of $DiBiCLUS$ is that it can work with seeds or without seeds. If there is a need to study a gene or a specific set of genes, $DiBiCLUS$ can find the differential biclusters for those genes. On the other hand, if there are no seeds, the algorithm iterates over all the genes, considering one gene as a seed at a time.

# 5. CONCLUSION AND FUTURE WORK

In this paper, we presented a novel approach to identify differential biclusters from gene expression data. This new algorithm emphasizes the differences in the biological activities between samples belonging to two different classes. We applied the proposed algorithm on synthetic and real life datasets, and the results showed that $DiBiCLUS$ algorithm is able to identify significant differential biclusters from the gene expression data.

There are some possible extensions for the current model. First, the $DiBiCLUS$ algorithm can be extended so that it works on multi-class gene expression data (more than two classes of conditions). Second, the similarity between any two genes is computed based on the number of identical non-zero conditions. One extension of this model is to consider similar but not exact values in computing the similarity between the genes.

**Figure 10: Samples of the differential biclusters obtained by applying $DiBiCLUS$ on the lung cancer dataset.(a) class = A, number of genes= 10 and number of conditions= 11. (b) class = B, number of genes= 15 and number of conditions= 11.**




## 6. REFERENCES


[1] W. Ayadi, M. Elloumi, and J.-K. Hao. A biclustering algorithm based on a bicluster enumeration tree: application to dna microarray data. *BioData Mining*, 2(1):9, 2009.

[2] A. Ben-Dor, B. Chor, R. Karp, and Z. Yakhini. Discovering local structure in gene expression data: the order-preserving submatrix problem. *Journal of computational biology*, 10(3-4):373–384, 2003.

[3] S. Busygin, O. Prokopyev, and P. M. Pardalos. Biclustering in data mining. *Comput. Oper. Res.*, 35(9):2964–2987, 2008.

[4] Y. Cheng and G. M. Church. Biclustering of expression data. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 93–103. AAAI Press, 2000.

[5] G. Fang, R. Kuang, G. Pandey, M. Steinbach, C. L. Myers, and V. Kumar. Subspace differential coexpression analysis: problem definition and a general approach. *Pacific Symposium on Biocomputing*, pages 145–156, 2010.

[6] B. Ganter and C. Giroux. Emerging Applications of Network and Pathway Analysis in Drug Discovery and Development. *Current Opinion in Drug Discovery and Development*, 11:86–94, 2008.

[7] G. Getz, E. Levine, and E. Domany. Coupled two-way clustering analysis of gene microarray data. *Proc. Natl. Acad. Sci. USA*, 97:12079–12084, 2000.

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

[9] J. Ihmels, S. Bergmann, J. Berman, and N. Barkai. Comparative gene expression analysis by a differential clustering approach: Application to the candida albicanstranscription program. *PLoS Genet*, 1(3):0380–0393, 2005.

[10] J. Y. King, R. Ferrara, R. Tabibiazar, J. M. Spin, M. M. Chen, A. Kuchinsky, A. Vailaya, R. Kincaid, A. Tsalenko, D. X.-F. Deng, A. Connolly, P. Zhang, E. Yang, C. Watt, Z. Yakhini, A. Ben-Dor, A. Adler, L. Bruhn, P. Tsao, T. Quertermous, and E. A. Ashley. Pathway analysis of coronary atherosclerosis. *Physiol. Genomics*, 23(1):103–118, 2005.

[11] G. Li, Q. Ma, H. Tang, A. H. Paterson, and Y. Xu. QUBIC: a qualitative biclustering algorithm for analyses of gene expression data. *Nucl. Acids Res.*, 37(15):e101–, 2009.

[12] J. Liu, Z. Li, X. Hu, and Y. Chen. Biclustering of microarray data with mospo based on crowding distance. *BMC Bioinformatics*, 10(Suppl 4):S9, 2009.

[13] S. Madeira and A. Oliveira. A polynomial time biclustering algorithm for finding approximate expression patterns in gene expression time series. *Algorithms for Molecular Biology*, 4(1):8, 2009.

[14] S. C. Madeira and A. L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE transactions on computational biology and bioinformatics*, 1(1):24–45, 2004.

[15] R. R Renee, A. Deepak, B. D. Melissa, Y. Sean, T. O. Folakemi, K. Nagi, M. H. Joseph, A. Titilola, S. Sandra, and F. S. Karam. Microarray comparison of prostate tumor gene expression in African-American and Caucasian American males: a pilot project study. *Infect Agent Cancer*, 4(1), 2009.

[16] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, and J. P. Mesirov. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, 2005.

[17] A. Tanay, R. Sharan, and R. Shamir. Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, 18(suppl-1):S136–144, 2002.

[18] M. Zou and S. D. Conzen. A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics*, 21(1):71–79, 2005.