ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 1, Issue 10, December 2012

Biclustering for Microarray Data: A Short and Comprehensive Tutorial

[1]Arabinda Panda, [2]Satchidananda Dehuri

[1]Department of Computer Science, Modern Engineering & Management Studies, Balasore
[2]Department of Computer Science, Ajou University, Republic of Korea

Abstract: This paper presents a quick and very comprehensive tutorial on biclustering for the analysis of gene expression data obtained from microarray experiments. The results obtained from the conventional clustering methods to gene expression data are limited by the existence of a number of experimental conditions where the activity of genes is uncorrelated. A similar limitation also exists when clustering of conditions is performed. For this reason, a number of algorithms that perform simultaneous clustering on the row and column dimensions of the gene expression matrix have been proposed to date. This simultaneous clustering, usually called as biclustering, which seeks to find sub-matrices, that is subgroups of genes and subgroups of columns, where the genes exhibit highly correlated activities for every condition. This type of algorithms has also been proposed and used in other fields, such as information retrieval and data mining.
In this comprehensive tutorial, we analyze a number of existing approaches to biclustering, and classify them in accordance with the type of biclusters they can find, the patterns of biclusters that are discovered, the methods used to perform the search and the target applications.

Key Words: Bicluster, Microarray, Gene expression, Data mining.

1. Introduction

DNA chips and other techniques measure the expression level of a large number of genes, perhaps all genes of an organism, within a number of different experimental samples (conditions). The samples may correspond to different time points or different environmental conditions. In other cases, the samples may have come from different organs, from cancerous or healthy tissues, or even from different individuals. Simply visualizing this kind of data, which is widely called *gene expression data* or *microarray data*, is challenging and extracting biologically relevant pattern-after post processing then it became knowledge is still harder [1].
Usually, gene expression data is arranged in a data matrix, where each gene corresponds to one row and each condition to one column. Each element of this matrix represents the expression level of a gene under a specific condition, and is represented by a real number, which is usually the logarithm of the relative abundance of the mRNA of the gene under the specific condition. Gene expression matrices have been extensively analyzed in two dimensions: the gene dimension and the condition dimension. This corresponds to the: i) analysis of expression patterns of genes by comparing rows in the matrix and ii) analysis of expression patterns of samples by comparing columns in the matrix.
The common objectives pursued when analyzing gene expression data include:

1) Grouping of genes according to their expression under multiple conditions.
2) Classification of a new gene, given its expression and the expression of other genes, with known classification.
3) Grouping of conditions based on the expression of a number of genes.
4) Classification of a new sample, given the expression of the genes under that experimental condition.

Clustering is the most popular approach of analyzing gene expression data and has proven successfully in many applications, Such as discovering gene pathway, gene classification, and function prediction. There is a very large body of literature on clustering in general and on applying clustering techniques to gene expression data in particular. Several

All Rights Reserved © 2012 IJARCET

representative algorithmic techniques have been developed and experimented in clustering gene expression data.

Although standard clustering algorithms have been successfully applied in many contexts, they suffer from two well-known limitations that are especially evident in the analysis of large and heterogeneous collections of gene expression data.

i) Grouping genes (or conditions) based on global similarities in their expression profiles. However, a set of coregulated genes might only be co-expressed in a subset of experimental conditions, and show not related, and almost independent expression patterns in the rest. In the same way, related experiments may be characterized by only a small subset of coordinately expressed genes.

ii) Standard clustering algorithms generally assign each gene to a single cluster. Nevertheless, many genes can be involved in different biological processes depending on the cellular requirements and, therefore, they might be coexpressed with different groups of genes under different experimental conditions [2]. Clustering the genes into one and only one group might mask the interrelationships between genes that are assigned to different clusters but show local similarities in their expression patterns.

For this reason, a number of algorithms that perform simultaneous clustering on the row and column dimensions of the gene expression matrix have been proposed to date. This simultaneous clustering, usually called as biclustering, which seeks to find sub-matrices, that is subgroups of genes and subgroups of columns, where the genes exhibit highly correlated activities for every condition.

2. Clustering Vs. Biclustering

The points which can gives us an idea of how clustering vary from biclustering are as follows:

i) clustering methods can be applied to either the rows or the columns of the data matrix in separately, where as in biclustering methods, it performs clustering in the two dimensions simultaneously, ii) clustering methods derive a global model while biclustering methods derive a local model, iii) in clustering algorithms, each gene in given gene cluster is defined using all the conditions, Where as in biclustering algorithms each gene in a given gene cluster is defined using only a subset of conditions and each condition in a bicluster is defined using only a subset of genes.

From the above, we can then conclude that, unlike clustering algorithms, biclustering algorithms identify groups of genes that show similar activity patterns under a specific subset of the experimental conditions. Therefore, biclustering approaches are the key technique to use when one or more of the following situation applies:

- Only a small set of the genes participates in a cellular process of interest.
- An interesting cellular process is active only in a subset of the conditions.
- A single gene may participate in multiple pathways that may or not be co-active under all conditions.

For these reasons, biclustering algorithms should identify groups of genes and conditions, obeying the following restrictions:

- A cluster of genes should be defined with respect to only a subset of the conditions.
- A cluster of conditions should be defined with respect to only a subset of the genes.
- The clusters should not be exclusive and/or exhaustive: a gene or condition should be able to belong to more than one cluster or to no cluster at all and be grouped using a subset of conditions or genes, respectively.

3. Biclusters

To overcome the shortcomings of clustering, we may seek instead a subset of genes that exhibit similar behavior across a subset of conditions. In terms of the expression data matrix, we seek a "homogeneous" sub matrix whose rows and columns correspond to the two subsets [3]. These objects are called biclusters and the process of detecting them is termed as biclustering.

The concept of bicluster was introduced by Cheng and Church in 2000 [7] to capture the coherence of a subset of genes and a subset of conditions. Unlike previous methods that treat similarity as a function of pairs of genes or pairs of conditions, the bicluster model measures coherence within the subset of genes and conditions. Figure 1 demonstrate the unclustered and clustered microarray data.

3.1 The General Model of Bicluster

In this section we present the model of bicluster and a way for accessing the quality of a bicluster. A bicluster is defined on a gene-expression matrix. Let G = $\{g_1, g_2 \ldots\ldots\ldots g_N\}$ be a set of genes and C = $\{c_1, c_2, \ldots\ldots\ldots c_N\}$ be a set of conditions. The data can be viewed as an $N \times M$ expression matrix called EM. EM is a matrix of real numbers, with possible null

All Rights Reserved © 2012 IJARCET

values, where each entry e_{ij} corresponds to the logarithm of the relative abundance of the mRNA of a gene g_i and under a specific condition c_j.

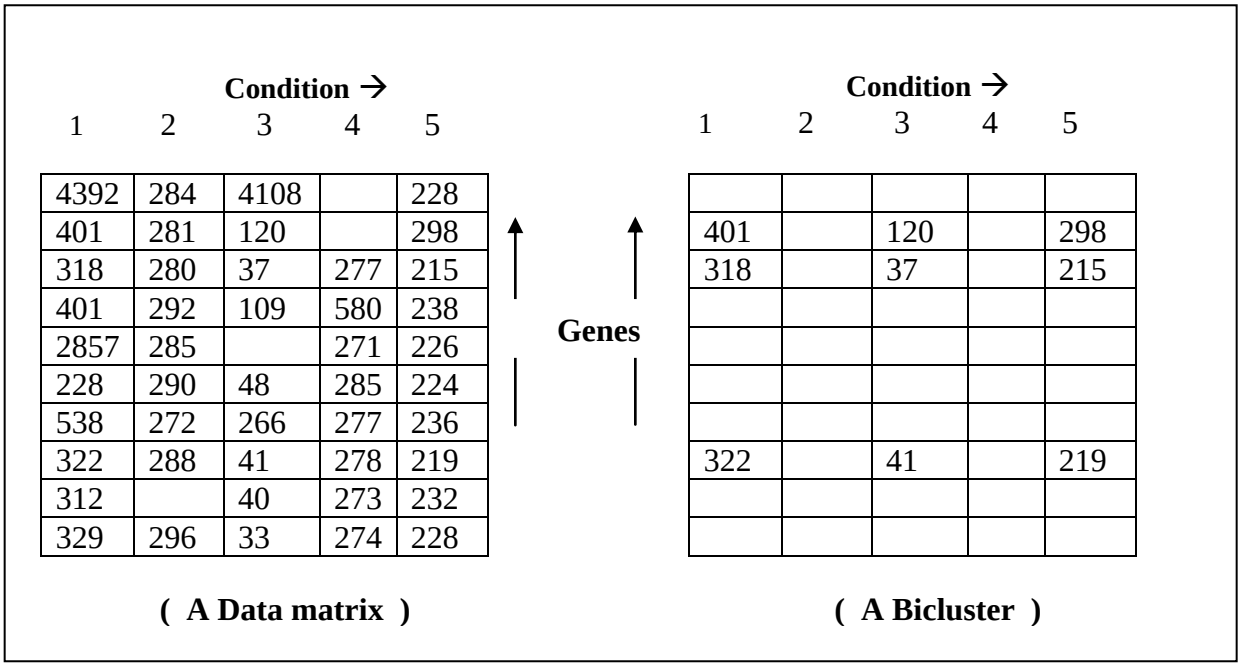
A bicluster essentially correspond to a sub-matrix that exhibits some coherent tendency. Each bicluster can be identified by a unique set of genes and conditions that determine the sub-matrix. Thus a bicluster is a matrix $I \times J$, denoted as (I, J), where I and J are set of genes (rows) and conditions (Columns), respectively, and $|I| \leq |N|$ and $|J| \leq |M|$. We define the volume of a bicluster (I , J) as the number of elements e_{ij} such that $i \in I$ and $j \in J$.

columns (biclusters) with similar values. Since this approach only produces good results when it is performed on non-noise data, which does not correspond to the great majority of available data, more sophisticated approaches can be used to pursue the goal of finding biclusters with constant values. When gene expression data is used, constant biclusters reveal subsets of conditions.

A perfect constant bicluster is a sub-matrix (I, J), where all values within the bicluster are equal for all $i \in I$ and $j \in J$:

$A_{ij} = \mu$, (1)

Where μ is the typical value within the bicluster.



(A Data matrix) **(A Bicluster)**

Biclusters can be classified into four categories and accordingly we can evaluate a biclustering algorithm.

1. Biclusters with constant values.
2. Biclusters with constant values on rows or columns.
3. Biclusters with coherent values.
4. Biclusters with coherent evolutions.

3.2.1 Biclusters with Constant Values

When the goal of a bicluster algorithm is to find a constant bicluster or several constant bicluster, it is natural to consider ways of recording the rows and columns of the data matrix in order to group together similar rows and similar columns, and discover subsets of rows and subsets of

3.2.2 Biclusters with constant values on rows and columns

Many biclustering algorithm aims at finding biclusters with constant value on the rows or columns of the data matrix. In the case of gene expression of data, a bicluster with constant values in the rows identifies a subset of genes with similar expression values across a subset of conditions, allowing the expression levels to differ from gene to gene. The same reasoning can be applied to identify a subset of conditions within which a subset of genes present similar expression values assuming that the expression values may differ from condition to condition [4].

All Rights Reserved © 2012 IJARCET

1.1	1.1	1.1
2.1	2.1	2.1
3.1	3.1	3.1

(Constant Rows)

1.1	2.1	3.1
1.1	2.1	3.1
1.1	2.1	3.1

(Constant Columns)

1.0	2.0	5.0
2.0	3.0	6.0
4.0	5.0	8.0

(Additive Model)

1.0	2.0	0.5
2.0	4.0	1.0
4.0	5.0	8.0

(Mulplicative Model)

A perfect bicluster with constant rows is a sub matrix (I, J), where all the values within the bicluster can be obtained using one of the following expressions:

$A_{ij} = \mu + \alpha_i$ (2)

$A_{ij} = \mu + \alpha_i$ (3)

where μ = The typical value within the bicluster, and α_i = The adjustment for row $i \in I$

1.1	1.1	1.1
1.1	1.1	1.1
1.1	1.1	1.1

(Constant bicluster)

Similarly, a perfect bicluster with constant columns is a sub matrix (I, J), where all the values within the bicluster can be obtained using one of the following expressions:

$A_{ij} = \mu + \beta_j$ (4)

$A_{ij} = \mu + \beta_j$ (5)

where μ = The typical value within the bicluster, β_j = The adjustment for column $j \in J$

This class of biclusters cannot be found simply by computing the variance of the values within the bicluster or similarities between the rows and columns of the data matrix.

3.2.3 Biclusters with Coherent values

Many biclustering algorithm aims at finding biclusters with coherent values on both rows and columns of the data matrix. In the case of gene expression data, we can be interested in identifying more complex biclusters where a subset of genes and a subset of conditions have coherent values on both rows and columns.

Additive Model

$A_{ij} = \mu + \alpha_i + \beta_j$ (6)

where μ = The typical value within the bicluster, α_i = The adjustment for row $i \in I$, β_j = The adjustment for column $j \in J$.

Multiplicative Model

A perfect bicluster with coherent values (I, J), can be defined as subset of rows and a subset of columns and can be represented as:

$A_{ij} = \mu' \times \alpha'_i \times \beta'$ (7)

In this model each element A_{ij} in the data matrix is seen as the product between the typical value within the bicluster, μ', the adjustment for row i, α'_i, and the adjustment for column j, β'_j.

3.2.4 Biclusters with Coherent Evolutions

Some biclustering algorithms address the problem of finding coherent evolutions across the rows and/or columns of the data matrix regardless of their exact values[5]. In the case of gene expression data, we may be interested in looking for evidence that a subset of genes is up-regulated or down-regulated across a subset of conditions without taking into account their actual expression values in the data matrix. The co-evolution property can be observed on both rows and columns of the biclusters.

S1	S1	S1
S2	S2	S2
S3	S3	S3

(Coherent Evolution on Rows)

S1	S2	S3
S1	S2	S3
S1	S2	S3

(Coherent Evolution on Columns)

4. Biclustering Algorithms

Biclustering algorithms may have two different objectives: i) to identify one bicluster and ii) to identify a given number of biclusters.

All Rights Reserved © 2012 IJARCET

- Some approaches attempt to identify *one bicluster at a time*.
- Some approaches attempt to identify *one set of biclusters at a time*.
- Some approaches attempt to identify all *biclusters* simultaneously.

The first two approaches are easier then the third one and to find the complexity of the problem, a number of different heuristic approaches has been used to address this problem [6]. They can be divided into five classes as:

- Iterative Row and Column Clustering Combination
- Divide and Conquer
- Greedy Iterative Search
- Exhaustive Bicluster Enumeration
- Distribution Parameter Identification

In iterative row and column clustering combination, it is a simpler way to perform biclustering using existing techniques is to apply standard clustering methods on the column and row dimensions of the data matrix, and then combine the results to obtain biclusters.

In divide and conquer algorithms, it have the significant advantage of being potentially very fast. However, they have the very significant drawback of being likely to miss good biclusters that may be split before they can be identified.

In greedy iterative search method, it is based on the idea of creating biclusters by adding or removing rows/columns from them, using a criterion that maximizes the *local* gain. It may make wrong decisions and loose good biclusters, but they have the potential to be very fast.

In exhaustive bicluster enumeration method, it is based on the idea that the best biclusters can only be identified using an exhaustive enumeration of all possible biclusters existent in the data matrix. These algorithms certainly find the best biclusters, if they exist, but have a very serious drawback. Due to their high complexity, they can only be executed by assuming restrictions on the size of the biclusters.

In distribution parameter identification, it assumes a given statistical model and tries to identify the distribution parameters used to generate the data by minimizing a certain criterion through an iterative approach.

5. Conclusion

We have presented a comprehensive tutorial of the methods of biclustering for gene expression of data. Many issues in biclustering algorithm design also remain open and should be addressed by the research community belongs to this area.

References

[1] Laura Lazzeroni and Art Owen.Plaid models for gene expression data. Technical report, Standford University, 2000.

[2] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulus, and Prabhakar Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proceedings of the ACM/SIGMOD International Conference on Management of Data*, pages 94–105, 1998.

[3] Amir Ben-Dor, Benny Chor, Richard Karp, and Zohar Yakhini. Discovering local structure in gene expression data: The order–preserving submatrix problem. In *Proceedings of the 6th International Conference on Computacional Biology (RECOMB'02)*, pages 49–57, 2002.

[4] Pavel Berkhin and Jonathan Becher. Learning simple relations: theory and applications. In *Proceedings of the 2nd SIAM International Conference on Data Mining*, pages 420–436, 2002.

[5] Stanislav Busygin, Gerrit Jacobsen, and Ewald Kramer. Double conjugated clustering applied o leukemia microarray data. In *Proceedings of the 2nd SIAM International Conference on Data Mining, Workshop on Clustering High Dimensional Data*, 2002.

[6] Andrea Califano, Gustavo Stolovitzky, and Yunai Tu. Analysis of gene expression microarays for phenotype classification. In *Proceedings of the International Conference on Computacional Molecular Biology*, pages 75–85, 2000.

[7] Yizong Cheng and George M. Church. Biclustering of expression data. In P*roceedings of the 8th International Conference on Intelligent Systems for Molecular Biology (ISMB'00)*, pages 93–103, 2000.

All Rights Reserved © 2012 IJARCET