Recent Patents on Biclustering Algorithms for Gene Expression Data Analysis

Alan Wee-Chung Liew[1*], Ngai-Fong Law[2], Hong Yan[3,4]

[1]*School of Information and Communication Technology,*
Gold Coast Campus, Griffith University, QLD 4222, Australia
Email: a.liew@griffith.edu.au
Tel: +61-7-55528671

[2]*Department of Electronic and Information Engineering,*
Hong Kong Polytechnic University, Hong Kong
Email: ennflaw@polyu.edu.hk

[3] *Department of Electronic Engineering,*
City University of Hong Kong, 83 Tat Chee Avenue, Hong Kong
[4]*School of Electrical and Information Engineering,*
University of Sydney, NSW 2006, Sydney, Australia
Email: h.yan@cityu.edu.hk

Abstract

In DNA microarray experiments, discovering groups of genes that share similar transcriptional characteristics is instrumental in functional annotation, tissue classification and motif identification. However, in many situations a subset of genes only exhibits a consistent pattern over a subset of conditions. Although used extensively in gene expression data analysis, conventional clustering algorithms that consider the entire row or column in an expression matrix can therefore fail to detect useful patterns in the data. Recently, biclustering has been proposed as a powerful computational tool to detect subsets of genes that exhibit consistent pattern over subsets of conditions. In this article, we review several recent patents in bicluster analysis, and in particular, highlight a recent patent from our group about a novel geometric-based biclustering method that handles the class of bicluster patterns with linear coherent variation across the row and/or column dimension. This class of bicluster patterns is of particular importance since it subsumes all constant, additive, and multiplicative bicluster patterns normally used in gene expression data analysis.

Keywords: biclustering, cluster analysis, multidimensional data analysis, gene expression data, geometric-based biclustering, microarray data, pattern discovery.

*corresponding author

Introduction

In DNA microarray experiments, discovering groups of genes that share similar transcriptional characteristics is instrumental in functional annotation, tissue classification and motif identification [1, 2]. Traditional cluster analysis of gene expression data detects groups of genes that share similar expression patterns across all conditions (or time points). However, in many situations, an interesting cellular process is active only under a subset of conditions, or a single gene may participate in multiple pathways that may or may not be co-active under all conditions [3, 4]. In addition, the expression data to be analyzed often include many heterogeneous conditions from many experiments. In these instances, it is often unrealistic to require that related genes behave similarly across all measured conditions or time points. Conventional clustering algorithms, such as the k-means clustering [5], hierarchical clustering [6] and the self-organizing map [7], therefore, cannot produce a satisfactory solution.

By relaxing the constraint that related genes must behave similarly across the entire column, "localized" groupings can be discovered readily. Biclustering allows us to consider only a subset of conditions when looking for similarity between genes. The goal of biclustering is to find submatrices in the dataset, i.e. subsets of genes and subsets of conditions, where the subset of genes exhibits significant homogeneity within the subset of conditions according to some homogeneity criteria. Fig. 1 shows the conceptual difference between traditional clustering and biclustering. Traditional clustering considers the entire set of conditions when clustering similar genes, whereas biclustering considers subset of genes and subset of conditions simultaneously. In Fig.2, we present an example where conventional hierarchical clustering fails but biclustering works. Fig.2a shows a data matrix, which appears random visually even after hierarchical clustering. However, if we permute the rows and columns appropriately as in bicluster analysis, a hidden pattern embedded in the data would be uncovered as shown in Fig.2b.

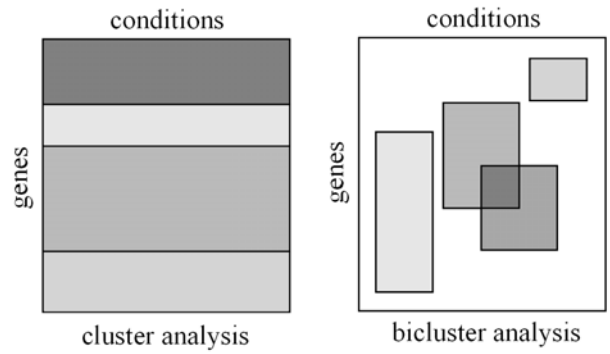
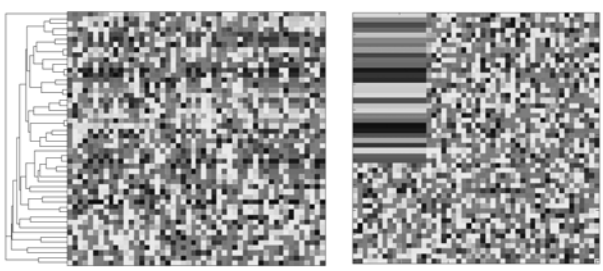


Fig.1 Conceptual difference between cluster analysis (left) and bicluster analysis (right)



(a) (b)

Fig.2 An example where conventional clustering fails but biclustering works: (a) a data matrix, which appears random visually even after hierarchical clustering. (b) A hidden pattern reviewed by appropriate permutation of subset of rows and columns

Bicluster Analysis of Gene Expression Data

In the past decades or so, many algorithms have been proposed for gene expression bicluster analysis [8-28]. The class of distance based biclustering is among the earliest biclustering algorithms proposed for gene expression data analysis [8-10]. In distance based biclustering, iterative search for the subsets of row and column indices that minimizes certain residual sum of squares cost is typically employed and the quality of the biclusters is measured by certain distance based metric. For example, in the δ-biclustering algorithm of Cheng and Church (CC) [8], a bicluster is one that minimizes the mean squared residue *H* score

$$H(X,Y)=\frac{1}{|X||Y|}\sum_{i\in X,\,j\in Y}\left(a_{ij}-a_{iY}-a_{Xj}+a_{XY}\right)^2 \qquad (1)$$

where

$$a_{iY} = \frac{1}{|Y|}\sum_{j\in Y} a_{ij}\ ,\ \ a_{Xj} = \frac{1}{|X|}\sum_{i\in X} a_{ij}\ ,\ \ a_{XY} = \frac{1}{|X||Y|}\sum_{i\in X, j\in Y} a_{ij}$$

are the row and column means and the mean in the submatrix $B=(X, Y)$, respectively. A bicluster is called a δ-bicluster if $H(X, Y)\leq\delta$ for some $\delta>0$.

Another class of biclustering algorithms assumes a probabilistic model of biclusters and applies statistical parameter estimation techniques to search for the biclusters [11, 12, 13]. For example, Gu and Liu [13] proposed a Bayesian biclustering algorithm (BBC) for gene expression data which is based on a generative data model given by

$$a_{ij} = \sum_{k=1}^{K}\left(\left(\mu_k + \alpha_{ik} + \beta_{jk} + \varepsilon_{ijk}\right)\delta_{ik}\kappa_{jk}\right) + e_{ij}\left(1 - \sum_{k=1}^{K}\delta_{ik}\kappa_{jk}\right) \qquad (2)$$

where K is the total number of clusters (unknown), μ_k is the main effect of cluster k, α_{ik} and β_{jk} are the effects of gene i and sample j respectively in cluster k, ε_{ijk} is the noise term for cluster k, and e_{ij} models the data points that do not belong to any cluster. Here, $\delta_{ik}=1$ indicates that gene i belongs to cluster k, and $\delta_{ik}= 0$ otherwise. Similarly, $\kappa_{jk}=1$ indicates that sample j is in cluster k, $\kappa_{jk}= 0$ otherwise. Gibbs sampling method is used for statistical inference in BBC.

Recently, we proposed a class of geometric-based biclustering algorithms based on a spatial interpretation of biclusters [14, 15]. In geometric-based biclustering, the problem of identification of coherent sub-matrices within a data matrix is formulated as the detection of linear geometric patterns (lines, planes, or hyperplanes) in a multidimensional data space [14]. For example, given a matrix $D_{N\times 3}$ represented as N points in a 3D space, a bicluster is given by a plane in the 3D space as shown in Fig.3. Unlike many existing biclustering algorithms which can only handle one specific type of bicluster patterns, i.e. constant, additive, or multiplicative biclusters, the geometric-based biclustering algorithms could detect bicluster patterns with linear coherent variation across the row and/or column dimension. Fig.4 shows the types of linear bicluster patterns that could be detected by geometric-based biclustering. Linear coherent bicluster pattern is of particular importance since it subsumes all constant, additive, and multiplicative biclusters normally used in gene expression data analysis. The ability to detect linear coherent biclusters means that unlike CC and BBC which are unable to detect multiplicative biclusters, geometric-based biclustering can *simultaneously* detect constant, additive, and multiplicative biclusters in the data matrix.

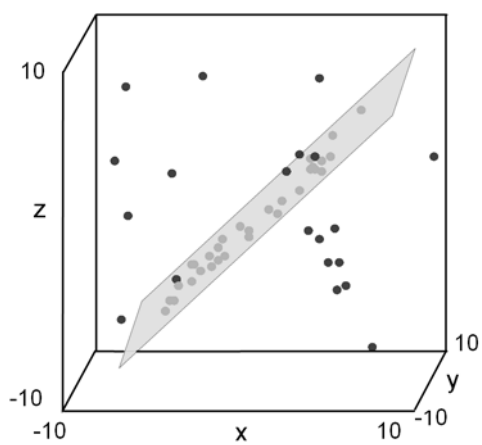


Fig.3 A plane formed by points in a bicluster in the three-dimensional data space where dark grey dots are data located on the plane

x	y	z	w
1.2	1.2	1.2	1.2
1.2	1.2	1.2	1.2
1.2	1.2	1.2	1.2
1.2	1.2	1.2	1.2

(a)

x	y	z	w
1.2	1.2	1.2	1.2
2.0	2.0	2.0	2.0
1.5	1.5	1.5	1.5
3.0	3.0	3.0	3.0

(b)

x	y	z	w
1.2	2.0	1.5	3.0
1.2	2.0	1.5	3.0
1.2	2.0	1.5	3.0
1.2	2.0	1.5	3.0

(c)

x	y	z	w
1.2	2.2	0.2	3.2
2.0	3.0	1.0	4.0
1.4	2.4	0.4	3.4
2.4	3.4	1.4	4.4

(d)

x	y	z	w
1.0	2.0	0.5	1.5
2.0	4.0	1.0	3.0
1.4	2.8	0.7	2.1
2.4	4.8	1.2	3.6

(e)

x	y	z	w
1.0	2.1	0.6	1.7
2.0	4.1	1.1	3.2
1.4	2.9	0.8	2.3
2.4	4.9	1.3	3.8

(f)

Fig. 4 Examples of different linear bicluster patterns: (a) constant values, (b) constant rows, (c) constant columns, (d) additive coherent values, (e) multiplicative coherent values, and (f) linear coherent values

In bicluster analysis, the elements of a data matrix are usually of numerical values but they can also be transformed into symbols that reflect trends in the data. The symbols can be purely nominal, of a given order, or encode positive and negative changes relative to a normal value. In the class of algorithms that attempt to find biclusters of coherent evolution, one is actually not interested in the exact numerical value of the bicluster but is instead interested in finding subsets of genes and conditions that exhibit coherent trend of expression. Fig.5 shows some examples of biclusters with coherent evolution. Several biclustering algorithms have been proposed to find patterns with coherent evolutions [16, 17, 18]. For example, Tanay *et al*. [16] introduced SAMBA (Statistical-Algorithmic Method for Bicluster Analysis) to detect the biclusters of coherent evolution. The data matrix is modeled as a bipartite graph. Discovering the most significant biclusters is equivalent to the selection of the heaviest subgraphs in the bipartite graph. SAMBA assumes that regardless of its true values, a_{ij} can be represented by two symbols: S_0 or S_1, where S_1 means change and S_0 means no-change. As such, the model graph has an edge between a row and a column when the gene is significantly changed with the sample. A large bicluster is, in this case, one with a maximum number of rows whose symbol for a_{ij} is expected to be S_1.

S1	S1	S1	S1
S1	S1	S1	S1
S1	S1	S1	S1
S1	S1	S1	S1

(a)

S1	S1	S1	S1
S2	S2	S2	S2
S3	S3	S3	S3
S4	S4	S4	S4

(b)

S1	S2	S3	S4
S1	S2	S3	S4
S1	S2	S3	S4
S1	S2	S3	S4

(c)

+1	+1	−1	+1
−1	−1	+1	−1
+1	+1	−1	+1
−1	−1	+1	−1

(d)

Fig.5 Types of biclusters with coherent evolution. Considering the entries of a data matrix as symbols, (a) an overall coherent evolution, (b) a coherent evolution on the rows, (c) a coherent evolution on the columns, (d) a coherent sign change across rows.

The intense research effort in biclustering analysis has resulted in many software packages being developed. Below is a list of some of the publicly available biclustering software.

- BBC [13] from http://www.people.fas.harvard.edu/~junliu/BBC/
- FLOC [9] in 'BicARE' from http://www.bioconductor.org/packages/2.8/bioc/html/BicARE.html
- Bimax [26], CC [8], Plaid model biclustering [11], Spectral biclustering [28], Xmotifs [18] in the R package called 'biclust' from https://r-forge.r-project.org/projects/biclust/
- CTWC [10] from http://ctwc.weizmann.ac.il/process.aspx
- bioNMF biclustering [27] from http://bionmf.cnb.csic.es/
- SAMBA [16] from http://acgt.cs.tau.ac.il/expander/
- BiVisu [19, 20] from http://www.eie.polyu.edu.hk/~nflaw/Biclustering/

Evaluating the performance of biclustering algorithms and validating the biclustering results is a challenging problem due to the different bicluster patterns to be detected, the different criteria used, and the different goals in biclustering. Common validation strategies for biclustering results include index based

validation, validation using domain knowledge, and statistical testing. In index based validation, a performance index that measures the compactness of the biclusters (if ground truth is unknown) or a score that measures the agreement between the detected biclusters and the ground truth (which is available in simulation study) is often used. In validation using domain knowledge, the *p*-values can be computed for validation to see whether a detected bicluster is statistically enriched with respect to some known facts about the data. The *p*-value measures the probability of including objects of a given category in a bicluster by chance. Thus, an overrepresented bicluster, for example, a bicluster with a high proportion of genes engage in a particular biological process, is a bicluster of objects which is very unlikely to be obtained by chance. In statistical testing, the statistical significance of a biclustering result is evaluated by comparing the result to a random partitioning of the data matrix. To date, there is a lack of comprehensive evaluation study on the performance of different biclustering algorithms, and no algorithm clearly stands out among the others [26]. This is perhaps due to the difficulty of defining what is meant by the "best" algorithm, similar to the situation in traditional clustering techniques [32]. Further work is clearly needed here.

Recent Patents on Biclustering Algorithm

In the 2006 US patent application by Parida [21], a method that detects fuzzy biclusters in DNA microarray data was presented. The method aims to detect maximal biclusters with column elements that do not deviate from each other by more than a given threshold δ. In this method, two elements x_1 and x_2 in a data matrix M is deemed to be "equal" if their different is smaller than δ. In Parida's method, a bicluster is parameterized by a pattern *m* and a location list L_m as follows. A pattern *m* is a collection of columns of the form $m = \{C_{j1}=X_1, C_{j2}= X_2, \ldots, C_{jl}= X_l\}$ occurring at row *i* if the elements $M(i, j_a)$ are equal to X_a for all $j_a \in \{j_1, j_2, \ldots, j_l\}$, where $M(i, j_a)$ denotes element of the data matrix M at row *i* and column j_a. The value X_k at column C_{jk} are given by the average of all values in that column at row indices given by L_m, where L_m is defined as the collection of rows which satisfy the pattern *m*. In other words, the elements of each column of Parida's bicluster do not deviate from the mean value of that column by more than ±δ/2, or equivalently, the difference between the maximum and minimum value in the column does not exceed δ. One can see that Parida's fuzzy bicluster is a generalization of the noisy constant bicluster (i.e. where every element in the bicluster cannot deviate by more than ±δ/2 from the global mean of the bicluster) in that each column in Parida's bicluster is allowed to have its own mean. Obviously, Parida's biclustering method is capable of discovering the constant and additive bicluster patterns shown in Figs.4a-4d but not the patterns shown in Figs.4e and 4f.

In the 2008 US patent application by Lepre [22], a method that detects biclusters in a data matrix using a three phase process is described. The main idea behind Lepre's method is to find subset of rows that allows partition of columns into clusters using a conventional clustering algorithm. Given a data matrix of, say, genes (row) and samples (column), in phase one of Lepre's method, an estimate of the probability density function (pdf) of each gene is done by fitting a mixture of Gaussians components to the histogram of samples in the gene. To illustrate Lepre's idea, in Fig.6 top panel we show a 1D signal generated by the following model: a values of 10 from samples 1 to 300, a value of 5 from samples 301 to 900, and a value of 15 from samples 901 to 1000, and finally, the sample points are corrupted with Gaussian noise of N(0,1) and randomized in position to give the signal shown in the middle panel. The bottom panel shows the normalized histogram of the randomized signal. From the histogram, we see that its samples formed 3 clusters, with the cluster means at 5, 10, and 15. The histogram could in fact be approximated by a mixture of 3 Gaussians of N(10,1), N(5,1), and N(15,1). After fitting a mixture of Gaussians components to the histogram, each sample in the gene is assigned to the Gaussian cluster (i.e. the Gaussian component) that maximizes the probability. In the second phase, subsets of related genes are group together as follows. Suppose the pdf of a gene *g* consist of 3 Gaussian mixtures, i.e. 3 Gaussian clusters, from phase 1. Three elementary patterns, π_1, π_2, and π_3, that correspond to the 3 Gaussian clusters are generated by setting the entries in the π_k pattern to 1 if the samples of *g* at those positions belong to cluster *k*, or 0 otherwise. Effectively, these elementary patterns are indicator functions (i.e. vectors of 1's and 0's) that indicate whether the samples belong to the particular Gaussian cluster or not. Fig.7 shows the 3 elementary patterns that correspond to the 3 Gaussian clusters detected in the signal of Fig.6. The elementary patterns are shown as images for ease of visualization.

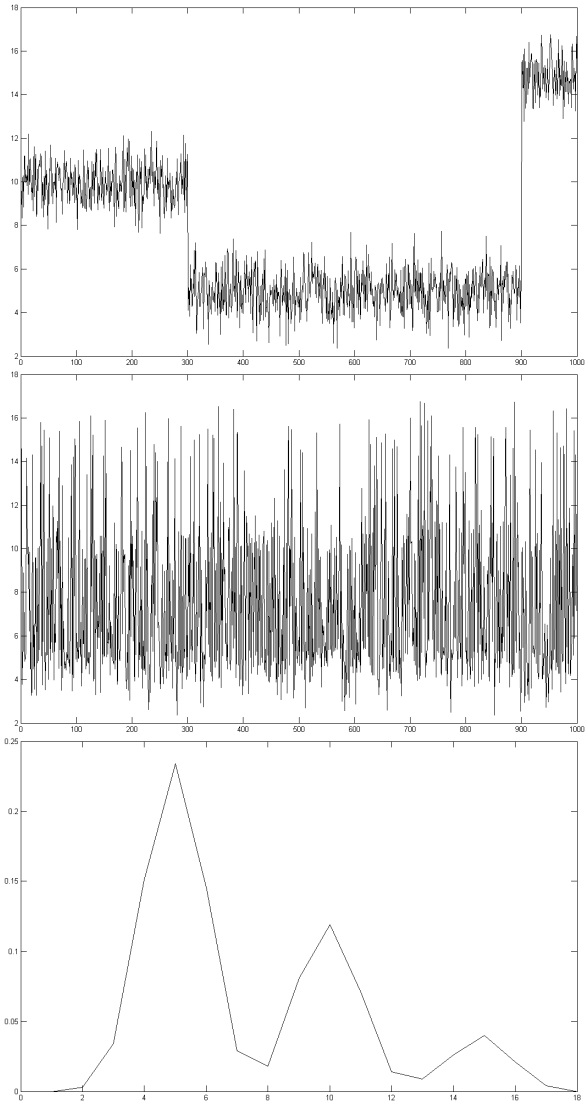


Fig.6 Top panel: A generated 1D signal, middle panel: the randomized signal, bottom panel: the histogram of the randomized signal.

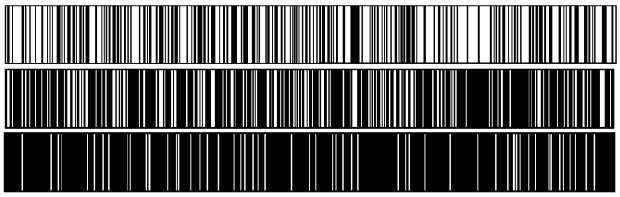


Fig. 7. Three elementary patterns for the 3 Gaussian mixture components of the randomly permuted signal shown in Fig.6. First pattern corresponding to the cluster centered at 5, second pattern corresponding to the cluster centered at 10, and the third pattern corresponding to the cluster centered at 15. White line at a location means that the sample at that location belongs to the cluster of concern while black line means otherwise.

In Lepre's method, only the elementary patterns from Gaussian components that satisfy a tightness criterion, i.e., the standard deviation of the Gaussian component is smaller than a threshold, are retained. Each gene generates M elementary patterns that correspond to the M mixture components, and the elementary patterns from all genes in the data matrix collectively form the pattern space. Hierarchical clustering is then performed on the pattern space to find clusters of elementary patterns that have significant overlap in the positions of 1's and 0's. The genes that correspond to a cluster of elementary patterns then constitute a subspace for phase 3. In phase 3, each subspace (i.e. group of genes) undergoes a conventional multivariate Gaussian mixture analysis. Suppose one of these subspaces contains 5 genes where each gene contains 100 samples, then the subspace is a matrix of 5×100 and the input data to the multivariate Gaussian mixture analysis is a 5D vector obtained from the columns of the 5×100 matrix. The multivariate Gaussian mixture analysis is done using the EM algorithm and the optimal number of mixture components K is determined using the BIC score. After multivariate Gaussian mixture analysis, the samples in the subspace are assigned to the cluster that maximizes the probability. Each of these K clusters in the subspace then defines a bicluster in the original data matrix. It can be seen that Lepre's method is really a two stage clustering process and the biclustering is indirectly induced from performing conventional clustering in the subspace of related rows in the original data matrix. Lepre's method assumes that the pdf of each gene in the data matrix can be approximated by a mixture of Gaussian components. Hence, only constant value biclusters or constant rows biclusters can be detected in the data matrix. Moreover, the gene has to have sufficient number of samples (i.e. the number of columns in the data matrix cannot be too small) in order for the pdf to be estimated adequately by a mixture of Gaussians. Finally, it is easy to see that the detected biclusters in Lepre's method cannot overlap in position in the data matrix. Lepre did not provide any specific example to illustrate the performance of the algorithm in biclustering gene expression data although it was noted that the method can be applied to analyze gene expression data matrix.

In 2009, a US patent was granted to Domany *et al*. on a biclustering algorithm called coupled two-way clustering (CTWC) [23]. CTWC is a framework that can be used to build a biclustering of a 2D data matrix base on iterative application of one-way clustering algorithm [10]. CTWC repeatedly performs one-way clustering on the rows and columns of the data matrix using stable clusters of genes and samples from the previous iteration as attributes. The iterative procedure of CTWC runs as follows. First, clustering is done row-wise and column-wise using the full data matrix, i.e., the full set of genes (g^0) and the full set of samples (s^0), to identify stable clusters of genes (g^1_i) and samples (s^1_j). A cluster is considered stable if it meets certain criteria such as how well it corresponds to a known classification or how well separated the level of expressions of the cluster is to other clusters according to some test statistics such as t test [10]. These stable clusters of genes $\{g^0, g^1_i\}$ and samples $\{s^0, s^1_i\}$ are recorded in the gene cluster list G or the sample cluster list S accordingly. Then at the next iteration, every pair of $(g^m_i, s^n_j) \in (G \times S)$ where $m, n \in (0,1)$, which defines a submatrix of the expression data consisting of subset of genes g^m_i and subset of samples s^n_j, undergoes row-wise clustering to obtain further stable gene clusters (g^2_k) and column-wise clustering to obtain further stable sample clusters (s^2_l). Whenever a clustering operation generates new stable gene clusters or new stable sample clusters, these new clusters are added to the gene cluster list G or the sample cluster list S accordingly. Together with the new clusters, information about the pair of parent clusters (g^m_i, s^n_j) that were used to generate them was recorded as well. These steps are iterated further, using pairs of all previously found clusters until no new clusters that satisfy some criteria of stability and minimum size can be found.

Two examples illustrating the benefit of using CTWC to analyze gene expression data were provided in the patent. In the first example, the leukemia dataset of Golub *et al*. [2] with 72 samples collected from acute leukemia patients at the time of diagnosis was analyzed. Using CTWC, it was possible to reveal the connection between T-cell related genes and allows the sub-classification of the ALL samples into T-ALL and B-ALL in an unsupervised manner. In addition, a stable partition of the AML patients into two groups was also found. The first group contained patients who were treated (with known results) and the second group contained all other patients. In the second example, the gene expression profiles of 40 colon tumor tissues and 22 normal colon tissues from the dataset of Alon *et al*. [29] were analyzed. It was found that the samples readily partition into clusters of tumor and normal tissues. It was also revealed that when the expression profiles were considered over only the tumor tissues, a cluster of cell growth genes was found to be highly correlated with epithelial genes.

In the US patent granted to Fan *et al*. in 2009 [24], a method for detecting biclusters of shift pattern is described. The method attempts to find subsets of columns where the corresponding subsequences from a dataset exhibit similar shift pattern (see Fig.8). To measure the distance between two shifting patterns x and y, the following distance is used

$$dist_{k,S}(x,y) = \max_{i \in S} \left| (x_i - y_i) - (x_k - y_k) \right| \qquad (3)$$

where S is the subset of columns over which the pattern similarity is evaluated, $k \in S$ is the dimension used for normalization. The intuition behind Eqn. (3) is illustrated in Fig.9 as follows. Let $S = \{k, a, b, c\}$. With respect to k, the distance between x and y in S is less than δ if the difference between x and y on any dimension in S is within $\Delta \pm \delta$, where Δ is the difference of x and y in dimension k. Although with a different choice of k, the distance between x and y in S will be different, it is bounded by 2δ. It is clear that Fan's method attempts to find clusters of subsequences in S that are related by a constant shift in value. Hence it is capable of detecting constant and additive bicluster patterns shown in Figs.4a-4d but not the patterns shown in Figs.4e and 4f.

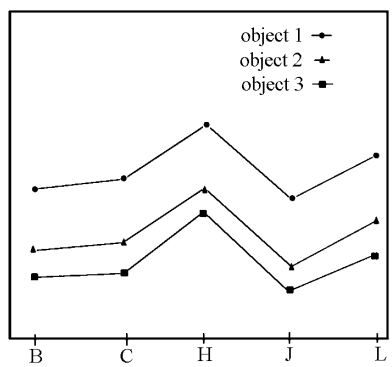


Fig.8 A shifting pattern in subset of columns {B, C, H, J, L}

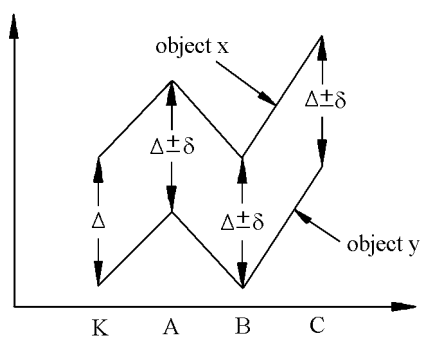


Fig.9 The intuitive meaning of $dist_{k,S}(x,y) \leq \delta$ in Fan's method

Fan's patent proposes a tree structure that provides a compact summary of all of the frequent patterns in a dataset. The tree structure allows efficient determination of the frequency of occurrence of any specified pattern, which is equivalent to finding biclusters of objects sharing similar patterns in a particular subset of columns. The method has been shown to be able to handle very large dataset due to the efficient tree structure proposed. As an example, the method was applied to the yeast microarray dataset as well as the mouse microarray dataset. However, no analysis as to the biological significance of the discovered biclusters was given.

In a 2010 US patent granted to Gan *et al*. [25] from our group, a new geometric framework for biclustering was proposed. In this framework, each gene expression profile (i.e. each row) is viewed as a point in a

high dimensional data space, with the dimension of the space corresponds to the number of columns in the data matrix. In this viewpoint, the problem of biclustering becomes that of detecting dense clusters of points in the high dimensional data space. By restricting the class of bicluster patterns to be of the linear coherent model type, such as in the example shown in Fig.4, the dense clusters of points can be shown to lie on a linear structure, i.e. lines, planes, or hyperplanes, in the high dimensional data space. The problem of biclustering then becomes that of detecting hyperplanes in the data space. Unlike many existing methods, the geometric based biclustering can simultaneously detect constant, additive, and multiplicative biclusters in the data matrix. The geometric framework of the algorithm in [25] can be visualized easily using data matrix with only 3 columns: *x*, *y*, *z*. In a 3D space, if we denote the three measurements as *x*, *y* and *z* respectively, and assume a bicluster covers *x* and *z* only as shown in Fig.10, we can generate 3D geometric views for different patterns as shown in Fig. 11.

(a)

x	y	z
1.2	1.3	1.2
1.2	2.4	1.2
1.2	0.8	1.2
1.2	7.2	1.2

(b)

x	y	z
1.2	1.3	1.2
2.0	2.4	2.0
1.5	0.8	1.5
3.0	7.2	3.0

(c)

x	y	z
1.2	1.3	1.5
1.2	2.4	1.5
1.2	0.8	1.5
1.2	7.2	1.5

(d)

x	y	z
1.2	1.3	0.2
2.0	2.4	1.0
1.4	0.8	0.4
2.4	7.2	1.4

(e)

x	y	z
1.0	1.3	0.5
2.0	2.4	1.0
1.4	0.8	0.7
2.4	7.2	1.2

(f)

x	y	z
1.0	1.3	0.6
2.0	2.4	1.1
1.4	0.8	0.8
2.4	7.2	1.3

Fig.10 Columns *x* and *z* constitute bicluster of: (a) constant values: $x = z = 1.2$, (b) constant rows: $x = z$, (c) constant columns: $x = 1.2$, $y = 1.5$, (d) additive coherent values: $z = x - 1.0$, (e) multiplicative coherent values: $z = 0.5x$, and (f) linear coherent values: $z = 0.5x + 0.1$, in 3D data space

In the geometric-based biclustering algorithm, the Hough transform (HT) is used to detect hyperplanes that correspond to biclusters in the high dimensional data. The HT is a powerful technique for line detection in noisy 2-D images and for plane detection in noisy 3-D signals and is widely used in pattern recognition. In the HT, each feature point in the data space generates "votes" for a set of parameter space points. An area in the parameter space containing many mapped points is suggestive of a hyperplane in the observed data space. The basic idea is as follows. Let $\{F_1, F_2, \ldots, F_M\}$ denotes the coordinates of *M* arrays (samples). For each gene j $\{j = 1, 2, \ldots, N\}$, the expression vector is given as $[F_1(j), F_2(j), \ldots, F_M(j)]$. Suppose that among all the observed data, there exists a target hyperplane with plane equation given by $F_M = \sum_{i=1}^{M-1} \beta_i F_i + \beta_M$, where $\{F_1, F_2, \ldots, F_M\}$ are coordinates of points in observed data space and $\{\beta_1, \beta_2, \ldots, \beta_M\}$ are the *M* parameters of the hyperplane. A set Ω is defined with all the indices of the genes which lie on this hyperplane. Then, for each $j \in \Omega$, $F_M(j) = \sum_{i=1}^{M-1} \beta_i F_i(j) + \beta_M$. It means that all these points on the target hyperplane satisfy

$$\sum_{i=1}^{M-1} F_i(j)\beta_i + \beta_M - F_M(j) = 0 \quad \text{for all} \quad j \in \Omega \tag{4}$$

The parameters $\{\beta_1, \beta_2, \ldots, \beta_M\}$ are given by the intersection of many hyperplanes given by Eqn. (4). Since the HT can be computationally expensive for high dimensional data, the patent in [25] also proposed a divide and stitch strategy to handle very large dataset.

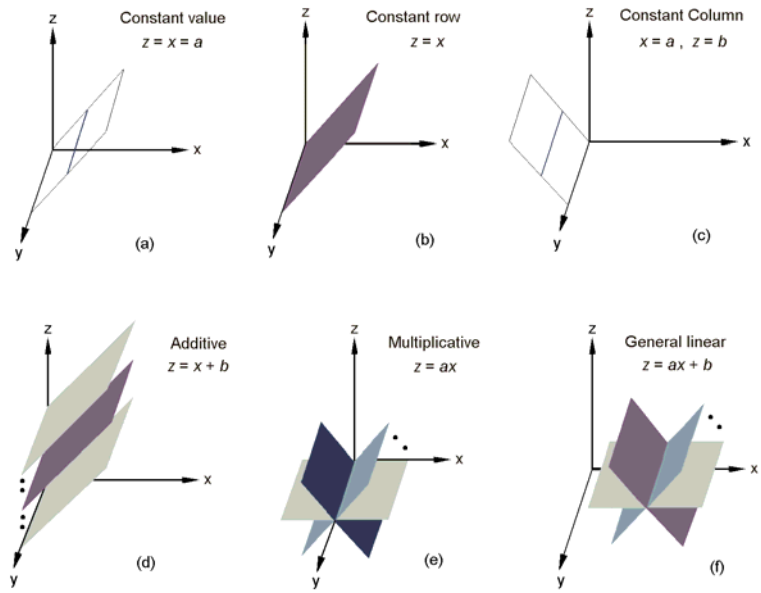


Fig.11 Different geometries (lines or planes) in the 3D data space for corresponding bicluster patterns. (a) A bicluster with constant values: represented by one of the lines that are parallel to the *y*-axis and lie in the plane $x = z$ (the T-plane), (b) a bicluster with constant rows: represented by the T-plane, (c) a bicluster with constant columns: represented by one of the lines parallel to the *y*-axis, (d) a bicluster with additive coherent values: represented by one of the planes parallel to the T-plane, (e) a bicluster with multiplicative coherent values: represented by one of the planes that include the *y*-axis , and (f) a bicluster with linear coherent values: represented by one of the planes that are parallel to the *y*-axis.

Performance of the geometric biclustering algorithm was evaluated in [25] by application to two gene expression datasets: the lymphoma dataset of Alizadeh *et al*. [30] and the human breast cancer dataset of van 't Veer *et al*. [31]. It was shown that the biclusters detected were statistically significant and were biologically relevant in that they have significant enrichment in the relevant GO terms. Further detailed analysis of the biclusters obtained from the lymphoma dataset was recently presented in [14] and interested readers are referred to it for details.

Current & Future Developments

Biclustering analysis is an important but challenging problem in gene expression data analysis. Many algorithms have been proposed and major advances have been made. However, the large varieties of bicluster patterns of interest and the computational complexity of the problem (biclustering analysis is a NP-hard problem) means that more research in this area is still needed. In this article, we presented an overview of the biclustering problem, the different bicluster patterns that are of interest to gene expression data analysis, and the different approaches that have been proposed to date. We surveyed recent patents on biclustering algorithms and discussed the methodologies and the bicluster patterns that could be detected by these algorithms.

The geometric biclustering framework [14, 15, 25] proposed by our group is particularly useful as it unites many different bicluster patterns into a linear coherent model and offers a solution that can simultaneously detect all these bicluster patterns. In principle, any algorithm that could efficiently detect hyperplanes in a high dimensional data space can be used in this framework. Further development in this class of geometric biclustering algorithms would be to search for robust and efficient hyperplane detection algorithms that could handle very large datasets.

Although we focus on gene expression data analysis in this paper, biclustering algorithms have many applications in other fields, such as data mining, market research, and image analysis. We expect that application-dependent strategies will be developed in the future for bicluster pattern discovery and analysis in various real-world systems. Furthermore, efficient computer methods need to be studied to deal with massive amount of multidimensional data.

Conflict of Interest

The authors declare no conflict of interest in the publication of this manuscript.

Acknowledgments

This work is supported by the Hong Kong Research Grant Council (Project CityU123809).

References

[1] Rew DA. DNA microarray technology in cancer research. Eur J Surg Oncol 2001; 27(5):504-508.
[2] Golub TR, Slonim DK, Tamayo P, *et al*. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 1999; 286(5439):531-537.
[3] Laub MT, McAdams HH, Feldblyum T, Fraser CM, Shapiro L. Global analysis of the genetic network controlling a bacterial cell cycle. Science 2000; 290(5499):2144-2148.
[4] DeRisi JL, Iyer VR, Brown PO. Exploring the metabolic and genetic control of gene expression on a genomic scale. Science 1997; 278(5338):680-686.
[5] Tavazoie S, Hughes JD, Campbell MJ, Cho RJ, Church GM. Systematic determination of genetic network architecture. Nat Genet 1999; 22(3):281-285.
[6] Eisen MB, Spellman PT, Brown PO, Botstein D. Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci USA 1998; 95(25):14863-14868.
[7] Tamayo P, Slonim D, Mesirov J, *et al*. Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. Proc Natl Acad Sci USA 1999; 96(6):2907-2912.
[8] Cheng Y, Church G.M. Biclustering of expression data. *Proc. 8th Int. Conf. Intelligent Systems for Molecular Biology* (ISMB '00), 2000, pp. 93-103.
[9] Yang J, Wang W, Wang H, Yu P. δ-clusters: capturing subspace correlation in a large data set. *Proc. 18th IEEE Int. Conf. Data Eng.*, 2002, pp. 517-528.
[10] Getz G, Levine E, Domany E. Coupled two-way clustering analysis of gene microarray data. Proc. Natural Academy of Sciences USA 2000; 97(22): 12079-12084.
[11] Lazzeroni L, Owen A. Plaid models for gene expression data. Technical report, Stanford Univ., 2000.
[12] Sheng Q, Moreau Y, De Moor B. Biclustering microarray data by Gibbs sampling. Bioinformatics 2003; 19: ii196-ii205.
[13] Gu J, Liu JS. Bayesian biclustering of gene expression data. BMC Genomics 2008; 9 (Suppl 1):S4.
[14] Gan X, Liew AWC, Yan H. Discovering biclusters in gene expression data based on high-dimensional linear geometries. BMC Bioinformatics 2008; 9:209.
[15] Zhao H, Liew AWC, Xie X, Yan H. A New geometric biclustering algorithm based on the Hough transform for analysis of large-scale microarray data. J. Theo. Biol. 2008; 251: 264–274.
[16] Tanay A, Sharan R, Shamir R. Discovering statistically significant biclusters in gene expression data. Bioinformatics 2002; 18: S136-S144.
[17] Ben-Dor A, Chor B, Karp R, Yakhini Z. Discovering local structure in gene expression data: the order-preserving submatrix problem. *Proc. 6th Int. Conf. Computational Biology* (RECOMB '02), 2002, pp. 49-57.
[18] Murali TM, Kasif S. Extracting conserved gene expression motifs from gene expression data. *Proc. Pacific Symp. Biocomputing*, 2003; 8: 77-88.
[19] Cheng KO, Law NF, Siu WC, Liew AWC. Identification of coherent patterns in gene expression data using an efficient biclustering algorithm and parallel coordinate visualization. BMC Bioinformatics 2008; 9:210.

[20] Cheng KO, Law NF, Siu WC, Lau TH. BiVisu: software tool for bicluster detection and visualization. Bioinformatics 2007; 23(17): 2342-2344.
[21] Parida LP. Fuzzy biclusters on multi-feature data. US20060184459, 2006.
[22] Lepre JO. Techniques for detection of multi-dimensional clusters in arbitrary subspaces of high-dimensional data. US20080021897, 2008.
[23] Domony E, Getz G, Levine E. Coupled two-way clustering analysis of data. US7599933, 2009.
[24] Fan W, Wang H, Yu PS. System and method for sequence-based subspace pattern clustering. US7565346, 2009.
[25] Gan XC, Liew AWC, Yan H. Representation and extraction of biclusters from data array. US7849088, 2010.
[26] Prelic A, Bleuler S, Zimmermann P, Wille A, Buhlmann P, Gruissem W, Hennig L, Thiele L, Zitzler E. A systematic comparison and evaluation of biclustering methods for gene expression data. Bioinformatics 2006; 22: 1122–1129.
[27] Carmonan-Saez P, Pascual-Marqui RD, Tirado F, Carazo JM, Pascual-Montano A. Biclusteing of gene expression data by non-smooth non-negative matrix factorization. BMC Bioinformatics 2006; 7: 78.
[28] Klugar Y, Basri R, Chang JT, Gerstein M. Spectral biclustering of microarray data: coclustering genes and conditions. Genome Research 2003; 13: 703-716.
[29] Alon U, Barkai N, Notterman DA, *et al*. Broad pattern of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc. Natl. Acad. Sci. USA 1999; 96:6745–6750.
[30] Alizadeh AA, Eisen MB, Davis RE, *et al*. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 2000; 403: 503-11.
[31] van 't Veer LJ, Dai H, van de Vijver MJ, *et al*. Gene expression profiling predicts clinical outcome of breast cancer. Nature 2002; 415: 530-536.
[32] Slonim DK. From patterns to pathways: gene expression data analysis comes of age. Nature genetics supplement, December 2002; 32: 502-508.