Statistical Identification of BiClusters in Gene Expression Data

Anupam Chakraborty[1]

[1]Department Of Computer Science & Engineering, Indian Institute of Technology Kharagpur

Abstract

A bicluster of a gene expression dataset is a subset of genes which exhibit similar expression patterns along a subset of conditions. Given a gene expression matrix, we search for submatrices that are tightly co-regulated according to some scoring criterion. We do not require the identified submatrices to be disjoint or to cover the entire matrix; instead we wish to build a diverse collection of submatrices that will capture all the significant signals in gene expression data. We believe that the size of the bicluster should be small compared to the size of the gene expression data matrix. So our approach finds biclusters by starting from small tightly co-regulated submatrices and adding more rows and columns to them. Our algorithm has three steps. First, we generate a set of high quality bicluster seeds based on a partition based clustering technique. In the second phase, these bicluster seeds are enlarged by adding more genes and conditions. In the third phase, we find the p-values of the biclusters produced for statistical validation.

Keywords: gene expression data, kmeans clustering, biclustering of expression data, p-value.

1. Introduction

The advent of DNA microarray technology has revolutionized the experimental study of gene expression. Clustering is the most popular approach to analyzing gene expression data and has indeed proven to be successful in many applications. However, clustering has its limitations. First, the clustering process builds on the assumption that related genes behave similarly across all measured conditions. Based on a general understanding of cellular processes we expect subsets of genes to be co-regulated and co-expressed under certain experimental conditions, but to behave almost independently under other conditions. Secondly, a clustering solution is often a partition of the genes into disjoint sets, implying an association of each gene with a single biological function or process, which may be an oversimplification of the biological system.

Cheng and Church (2000) [3] were the first to apply biclustering to gene expression data. They defined a biscluster as a uniform submatrix (one having a low mean squared residue score), and used a greedy approach to find biclusters. Yang et al. (2003) [6] generalized the model of bicluster proposed by Cheng and Church to incorporate null values and to remove random interference. They proposed a probabilistic algorithm (FLOC) that can discover a set of k possibly overlapping biclusters simultaneously. Getz et al. (2000) [4] devised a coupled two-way iterative clustering algorithm to identify biclusters. Zhang et al. (2004) [7] presented DBF (Deterministic Biclustering with Frequent pattern mining), that generates a set of good quality biclusters based on frequent pattern mining in the first phase. In the second phase, the biclusters are further iteratively enlarged by adding more genes and/or conditions.

Our method differs from Cheng and Church (2000) [3] approach where rows and columns were deleted from the gene expression data matrix to find a bicluster. Their algorithm is deterministic, repeated runs of them will not discover different biclusters, unless discovered ones are masked. So a discovered bicluster is replaced by random values. These random numbers will interfere the future discovery of biclusters. Our approach avoids random interference by starting from small tightly co-regulated submatrices and adding more rows and columns to them. We propose a simple seed finding technique in which we use gene and condition clusters obtained by a partition based method. We grow each seed separately, adding rows and columns. Our method finds high quality biclusters that we can inspect visually using plots and validate statistically by finding the p-values of the produced biclusters.

2. Definitions

Y. Cheng and G.M. Church (2000) [3] defined the mean squared residue as follows: Let X be the set of genes and Y the set of conditions. Let a_{ij} be the element of the expression matrix A that represents the logarithm of the relative abundance of the mRNA of the i-th gene under the j-th condition. Let $I \subseteq X$ and

$J \subseteq Y$ be subsets of genes and conditions. The pair (I,J) specifies a submatrix A_{IJ}. The residue score of an element a_{ij} in a submatrix A_{IJ} is defined as

$$RS_{IJ}(i,j) = a_{ij} - a_{Ij} - a_{iJ} + a_{IJ}$$

and the Mean Residue Score of the submatrix as

$$H(I,J) = \frac{1}{|I||J|}\sum_{i\in I, j\in J}(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2$$

where

$$a_{iJ} = \frac{1}{|J|}\sum_{j\in J} a_{ij} \quad , \quad a_{Ij} = \frac{1}{|I|}\sum_{i\in I} a_{ij}$$

and

$$a_{IJ} = \frac{1}{|I||J|}\sum_{i\in I, j\in J} a_{ij}$$

A submatrix A_{IJ} is called a δ-bicluster if $H(I,J)\leq\delta$ for some $\delta\geq 0$.

The residue score a_{ij} gives an idea of how the value a_{ij} fits into the data in the surrounding matrix. The mean squared residue score gives an indication of how the data is correlated in the submatrix- whether it has some coherence or is random. A high H value signifies that the data is uncorrelated. A matrix of equally spread random values over the range [a,b] has an expected H score of $(b-a)^2/12$. A low H score means that there is a correlation in the matrix.

Y. Cheng and G.M. Church (2000) [3] proved that the problem of finding the largest square δ-bicluster (|I| = |J|) is NP-hard. We are thus interested in heuristics for finding a large δ-bicluster in reasonable time.

3. Algorithm

There are three steps in our algorithm. The first step is seed finding. The seeds in our approach are tightly co-regulated submatrices. We use a simple seed finding technique in which we use gene and condition clusters obtained by the kmeans partition based method. Secondly, seeds are further enlarged by adding more genes and/or conditions. Thirdly, we calculate the p-values of the biclusters produced.

Seed Finding

The gene expression dataset is partitioned into n gene clusters and m sample clusters using the kmeans algorithm. Each gene cluster is further divided into sets of 10 genes according to the cosine angle distance from the cluster center. Similarly, each sample cluster is further divided into sets of 5 samples according to the cosine angle distance from the cluster center. Suppose, the number of gene clusters, each containing at most 10 genes is x and the number of sample clusters each containing at most 5 samples is y. Thus we have partitioned the initial gene expression data matrix into x*y disjoint sub-matrices each containing at most 10 close genes and at most 5 close conditions. We calculate the HScore for each of the x*y submatrices and select the first 100 submatrices as our initial seed.

Seed Growing

In this phase, the biclusters are enlarged using a greedy algorithm by adding more genes and conditions until the HScore of the BiCluster reaches a given threshold.

Initialize: BiCluster=Seed;
Iteration:
While (BiCluster.HScore<given_threshold)
1. Find the sample C such that addition of C to the BiCluster results in the minimum modified BiCluster.HScore.
2. Add sample C to BiCluster.
3. Find the gene G such that addition of G to the BiCluster results in the minimum modified BiCluster.HScore.
4. Add gene G to BiCluster.

Find the core of the resulting bicluster: This is the simple average of the profiles of the genes that belong to the bicluster. We then remove those genes from the bicluster for which the pearson correlation coefficient from the core is below a threshold (0.70).

Significance Evaluation

Tanay et al. (2002) [5] describes a way of evaluating biclusters using prior biological knowledge. We use the same mathematical framework to measure the significance of the biclusters produced by our algorithm. Existence of biclusters comprising a significant proportion of those samples that are considered similar biologically is proof that a specific biclustering technique produces biologically relevant results. Tanay et al. (2002) [5] used a correspondence plot to evaluate bisclustering results using prior biological knowledge. In a correspondence plot the p-values are on a logarithmic scale; the plot presents the fraction of biclusters whose p-value is at most p out of the 100 best biclusters obtained by our algorithm. P-values of the biclusters are calculated according to the known classification of samples as follows: suppose prior knowledge partitions the m conditions into k classes $C_1,\ldots,C_k$. Let B be a bicluster with b conditions, out of which b_j belong to class C_j. The p-value of B, assuming its most abundant class is C_i, is calculated as

$$p(B) = \sum_{k=b_i}^{b} \binom{|C_i|}{k}\binom{m-|C_i|}{b-k} \Big/ \binom{m}{b}$$

Hence, the p-value measures the probability of obtaining at least b_i elements from the class in a random set of size b. If a majority of conditions in a cluster have the same biological function, then it is unlikely that this happens by chance and the p-value would be close to 0.

Similarly, the probability (p-value) of finding at least k genes, in a bicluster of n genes, belonging to a specific functional category comprising f genes out of total g annotated genes is given by

$$P = 1 - \sum_{i=0}^{k-1} \frac{({}^{f}C_{i})({}^{g-f}C_{n-i})}{({}^{g}C_{n})} = \sum_{i=k}^{\min(n,f)} \frac{({}^{f}C_{i})({}^{g-f}C_{n-i})}{({}^{g}C_{n})}$$

Time Complexity of the Algorithm

This algorithm requires computation of the scores of all the submatrices that may be result from any row or column addition, before each choice of addition can be made. We can calculate the HScore of a submatrix in O(mn) time. Thus our method requires O((n+m)nm) time, where n and m are the row and column sizes of the expression matrix, to find one bicluster.

4. Datasets Used

We have used the same gene expression data sets as used by Y. Cheng and G.M. Church [3] to compare our results with their. The yeast Saccharomyces cerevisiae cell cycle expression dataset contains 2,884 genes and 17 conditions. The dataset is based on Tavazoie et al. [2]. The expression values were transformed by scaling and logarithm $x \rightarrow 100\log(10^5 x)$ and the result was a matrix of integers in the range 0 and 600. Missing values were represented by -1 in the yeast dataset. The human lymphoma dataset contains 4026 genes and 96 conditions. The human lymphoma data was downloaded from the website for supplementary information for the article by Alizadeh et al. (2000) [1]. The expression levels were reported as log ratios and after a scaling by a factor of 100(i.e. $x \rightarrow 100x$), they ended up with a matrix of integers in the range between -750 and 650, with 47,639 missing values (12.3% of the matrix elements). Missing values were represented by 999 in the human lymphoma datasets. The matrices after the above processing, along with the biclustering results of Cheng & Church Algorithm (2000) were obtained from http://arep.med.harvard.edu/biclustering.

5. Results

Algorithm Parameters for 2 datasets

Here we describe the discovery of seeds from the human lymphoma dataset. The lymphoma gene expression data is partitioned into 200 gene clusters and 15 array cluster using the kmeans algorithm. After that we partition the initial gene expression data matrix into 499*24 disjoint sub-matrices each containing at most 10 close genes and at most 5 close conditions. We calculate the HScore for each of the 499*24 submatrices and select the first 100 submatrices as our initial seed.

Now we describe the discovery of seeds from the yeast dataset. The yeast gene expression data is partitioned into 140 gene clusters and 3 array cluster using the kmeans algorithm. Next, we partition the initial gene expression data matrix into 350*5 disjoint sub-matrices each containing at most 10 close genes and at most 5 close conditions. We calculate the HScore for each of the 350*5 submatrices and select the first 100 submatrices as our initial seed.

Seed Growing

From the previous discussion, we know that a completely random submatrix of any size with element values in the range (0 to 800) has a HScore about 53,000. For the yeast dataset, the clusters reported in Tavazoie et al. (1999) [2] have scores in the range 261 (Cluster 3) and 996 (Cluster 7), with a median of 630 (Clusters 8 and 14). We choose the threshold δ as 150, which is slightly greater than the max HScore of the seeds (114.24) used in our algorithm in the seed growing phase. For the human lymphoma expression data we choose the threshold δ as 600 which is also slightly greater than the max HScore of the seed (542.62), used in our algorithm in the seed growing phase. In case of both datasets δ-value used in our approach is half of δ-value used by Cheng & Church.

Coverage of the Biclusters

In the yeast data experiment, the 100 biclusters covered 844, or 29.28% of the genes, 100% of the conditions, and 27.37% of the cells in the matrix. The first 100 biclusters from the human data covered 1541, or 38.27% of the genes, 100% of the conditions, and 15.84 % of the cells in the data matrix.

P-Values of the Human Lymphoma BiClusters

The lymphoma dataset is characterized by well defined expression patterns differentiating three types of lymphoma, DLBCL, CLL and FL from one another (By Alizadeh et al., 2000[1]). Running Cheng & Church (2000) [3] algorithm and our algorithm on the same dataset (the lymphoma data), a collection of biclusters from both algorithms were analyzed using the known classification of conditions to different clinical types (DLBCL, CLL, FL and more). The results clearly indicate that biclusters discovered by our algorithm are aligned more closely with the biological information.

P-Values of the Yeast BiClusters

Tavazoie et al. used an iterative optimization-based partitional clustering (k-means) to group 3000 genes into 30 expression classes which were highly enriched for genes of similar function on Time-series of mRNA abundance, measured over two synchronized Saccharomyces cerevisiae cell cycles. Our result clearly indicates that biclusters discovered by our algorithm are better aligned with the clustering results from Tavazoie et al.

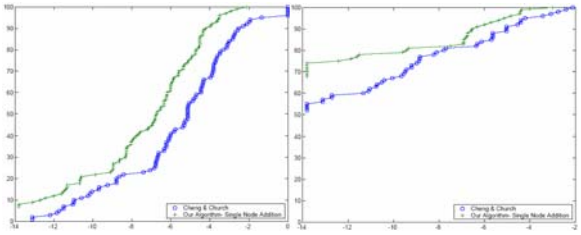

Figure.1: For each value of p on a logarithmic scale, the plot presents the fraction of biclusters whose p-value is at most p out of the (say) 100 best biclusters. a)Correspondence plot for the lymphoma dataset, b)Correspondence plot for the yeast dataset.

BiCluster Plots:

The following plots show the core of biclusters we have found in our biclustering approach.

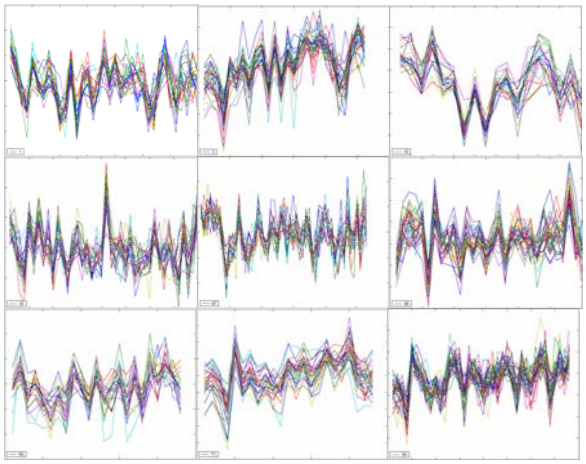

Figure.2: Biclusters discovered from the human lymphoma expression data. The numbers of genes and conditions in each are reported in the following format (bicluster label, number of genes, number of conditions, HScore, pValue) as follows.(1, 23, 35, 497.27, 0.161277), (3, 32, 26, 603.08, 0.000018), (10, 22, 18, 602.56, 0.001832), (35, 22, 40, 429.13, 0.018034), (47, 18, 53, 462.50, 0.034122), (48, 34, 30, 601.42, 0.000087), (65, 21, 23, 487.48, 0.006916), (71, 28, 23, 582.87, 0.000001), (90, 37, 38, 515.37, 0.007974).

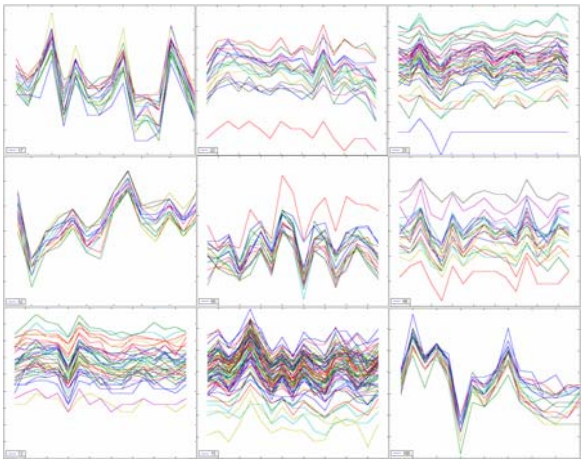

Figure.3: Biclusters discovered from the yeast expression data. The numbers of genes and conditions in each are reported in the following format (bicluster label, number of genes, number of conditions, HScore, pValue) as follows. (17, 16, 16, 128.81,$<10^{-6}$), (22, 19, 17, 125.38, $<10^{-6}$), (33, 50, 17, 124.35, $<10^{-6}$), (52, 14, 14, 87.52, 0.000001), (60, 17, 17, 108.12, $<10^{-6}$), (66, 20, 17, 118.88, $<10^{-6}$), (72, 42, 17, 120.92, $<10^{-6}$), (75, 73, 17, 141.52, $<10^{-6}$), (100, 16, 16, 132.07, $<10^{-6}$).

6. Multiple Node Addition

The Cheng and Church (2000) [3] algorithm is based on multiple row/column deletion that decreases the residue score below a threshold followed by row/column addition that does not increase the score. In our approach the discovered seeds are enlarged by adding more rows and columns to extend their size while keeping their mean squared residue below a certain predetermined threshold. We prove a lemma to give a bound on the increase of residue scores for addition of multiple rows and columns to enlarge the seeds. We do this by building on the framework developed by Cheng and Church in [3] which is reviewed here for completeness.

6.1. Review of Cheng and Church Work

Let Σ be a finite set of points in a space in which a non-negative real-valued function of two arguments, d is defined. Let S be a set of points such that $S \subset \Sigma$. Let m(S) be a point that minimizes the function $f(s) = \sum_{x \in S} d(x,s)$.

Define the measure $E(S) = \frac{1}{|S|} \sum_{x \in S} d(x, m(S))$.

Cheng et al. prove the following lower bound on the decrease of the residue scores for multiple node deletion.

Lemma 1 [3] Suppose the set removed from S is $R \subseteq \{x \in S : d(x, m(S)) > \alpha E(S)\}$ with $\alpha \geq 1$. Then the reduction rate of the score E(S) can be characterized as $\frac{E(S) - E(S-R)}{E(S)} > \frac{\alpha - 1}{|S|/|R| - 1}$.

We wish to obtain an analogous upper bound on the increase of the residue scores for multiple node addition. Cheng et al. do prove a monotonic result for the node addition case which is instrumental in obtaining our bound.

Lemma 2 [3] The addition to S of any non-empty subset $R \subseteq \{x \notin S : d(x, m(S)) \leq E(S)\}$ will not increase the score E: $E(S+R) \leq E(S)$.

6.2. Our Lemma

Lemma 3. Suppose the set added to S is $R \subseteq \{x \notin S : d(x, m(S)) \leq \alpha E(S)\}$ with $\alpha \geq 1$. Then the growth rate of the score E(S) can be characterized as $\frac{E(S+R) - E(S)}{E(S)} < \frac{\alpha - 1}{|S|/|R| + 1}$.

Proof: The condition $E(S+R) \leq E(S)$ of Lemma 2 can be rewritten as: $\frac{A'}{|S+R|} \leq \frac{A-B}{|S|}$,

Where,

$A = \sum_{x \in S+R} d(x, m(S)), A' = \sum_{x \in S+R} d(x, m(S+R)), B = \sum_{x \in R} d(x, m(S))$

The definition of the function m requires that $A' \leq A$.

Thus, a sufficient condition for the inequality is

$\frac{A}{|S+R|} \leq \frac{A-B}{|S|}$

$\Rightarrow |S| A \leq (|S+R|)(A-B)$

Which is equivalent to

$E(S) = \frac{A-B}{|S|} \geq \frac{B}{|R|} = \frac{1}{|R|} \sum_{x \in R} d(x, m(S))$

$\Rightarrow \alpha E(S) = \alpha \frac{A-B}{|S|} \geq \frac{B}{|R|} = \frac{1}{|R|} \sum_{x \in R} d(x, m(S))$

This leads to $\alpha |R| A \geq (|S| + \alpha |R|) B$

Or, equivalently, $|S| A \leq (|S| + \alpha |R|)(A-B)$

This is the same as $\frac{A}{|S-R|} \leq \frac{(|S| + \alpha |R|)}{|S+R|} \frac{A-B}{|S|}$

Using the inequality $A' \leq A$ and the facts that $E(S+R) = A/|S+R|$ and $E(S) = (A-B)/|S|$, this leads to the inequality $E(S+R) \leq \frac{|S| + \alpha |R|}{|S+R|} E(S)$. □

7. Multiple Node Addition Algorithm

The modified algorithm has three steps as the previous one. The seed finding and significance evaluation steps are same. Modification in the seed growing phase gives a time efficient algorithm.

We enlarged the discovered seeds by adding more rows and columns to extend their size while keeping their mean squared residue below a certain predetermined threshold. Lemma 3 states that when we add rows/columns whose relative share to the HScore of a bicluster is less than α*HScore, the increase of HScore is bounded by fraction α. We have used this property of HScore to add more than one row or column in Step 2 and 5 while also keeping the resulting HScore within a bound in our multiple node addition algorithm.

Seed Growing:

Initialize: BiCluster=Seed;

Iteration:

While(BiCluster.HScore < given_threshold)

1. Compute a_{iJ} for all I, a_{Ij} for all J, a_{IJ}, and H(I,J).

2. If α*H(I,J) < given threshold then add columns $j \notin J$ with $\frac{1}{|I|} \sum_{i \in I} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2 \leq \alpha H(I,J)$.

3. If no column is added in step-2 then find the sample C such that addition of C to the BiCluster results in the minimum modified BiCluster.HScore and add sample C to the BiCluster.

4. Recompute a_{iJ} for all I, a_{Ij} for all J, a_{IJ}, and H(I,J).

5. If α*H(I,J) < given threshold then add rows $i \notin I$ with $\frac{1}{|J|} \sum_{j \in J} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2 \leq \alpha H(I,J)$.

6. If no gene is added in step-5 then find the gene G such that addition of G to the BiCluster results in the minimum modified BiCluster.HScore and add gene G to the BiCluster.

8. Multiple Node Addition Results

P-Values of the Human Lymphoma and Yeast BiClusters

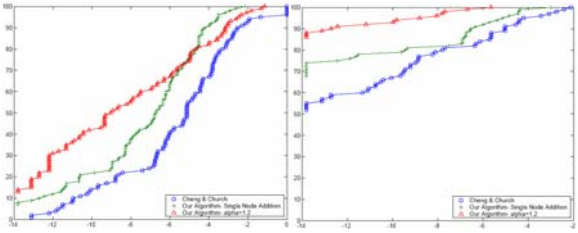

Figure.4: For each value of p on a logarithmic scale, the plot presents the fraction of biclusters whose p-value is at most p out of the (say) 100 best biclusters. a)Correspondence plot

for the lymphoma dataset, b)Correspondence plot for the yeast dataset.

Coverage of the Biclusters

In the yeast data experiment, the 100 biclusters covered 1727, or 59.92% of the genes, 100% of the conditions, and 49.36% of the cells in the matrix. The first 100 biclusters from the human data covered 3623, or 89.99% of the genes, 98.95% of the conditions, i.e., 95 out of 96 conditions and 33.76 % of the cells in the data matrix.

BiCluster Plots:

The following plots show the core of biclusters we have found in our biclustering approach.

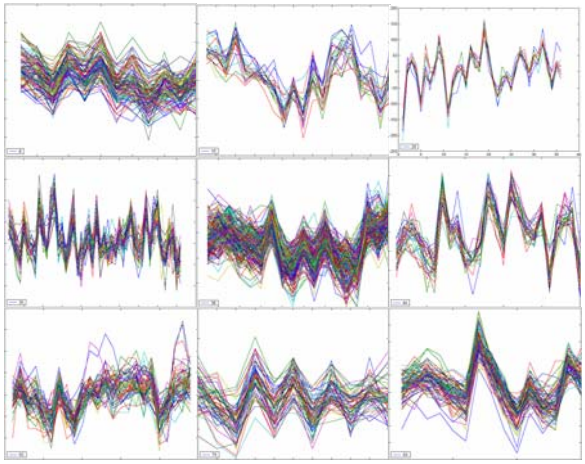

Figure.5: Biclusters discovered from the human lymphoma expression data. The numbers of genes and conditions in each are reported in the following format (bicluster label, number of genes, number of conditions, HScore, pValue) as follows. (8, 102, 12, 445.3551, 0.000165), (10, 22, 20, 623.3054, 0.001832), (28, 10, 36, 308.3003,0.035619), (35, 36, 46, 579.8006, 0.018034), (38, 187, 18, 472.5243, 0.000003), (44, 28, 25, 617.6036, 0.000002), (65, 56, 24, 596.4761, 0.006916), (79, 60, 11, 324.5671, 0.000086), (84, 78, 15, 611.3813, 0.000000).

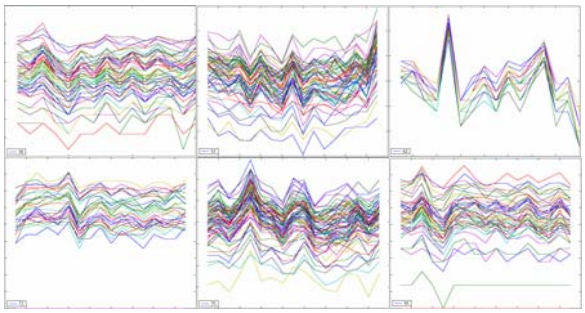

Figure.6: Biclusters discovered from the yeast expression data. The numbers of genes and conditions in each are reported in the following format (bicluster label, number of genes, number of conditions, HScore, pValue) as follows.(30, 56, 15, 128.1779, $<10^{-6}$), (51, 60, 17, 126.4142, $<10^{-6}$), (62, 16, 16, 117.8066, 0.000405), (72, 33, 17, 115.3787, $<10^{-6}$), (75, 71, 17, 139.8978, $<10^{-6}$), (91, 52, 17, 135.7333, $<10^{-6}$)

9. Conclusion

Our work focuses on biclustering of gene expression data, i.e. discovering a subset of genes that exhibit similar expression patterns along a subset of conditions in the gene expression matrix. We have proposed a new approach that exploits a simple technique to find the initial seeds using gene and condition clusters. The seeds generated are then enlarged by adding more rows and columns to extend their size while keeping their mean squared residue below a certain predetermined threshold. We prove a lemma to add more than one row or column to the seed. That makes the algorithm faster and practical for the large gene expression datasets. We have implemented our algorithm, and tested it on the yeast and human lymphoma dataset. For both, we have found biclusters with HScore that are half of the values used in Cheng and Church (2000) [3]. The correspondence plot reveals that our algorithm finds high quality biclusters that are aligned more closely with prior biological knowledge than the Cheng and Church approach.

10.References

[1] Alizadeh, A.A. et al. 2000. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 403:503-510.

[2] Tavazoie S, Hughes JD, Campbell MJ, Cho RJ, and Church GM. Systematic determination of genetic network architecture. Nat Genet, 22:281–5, 1999.

[3] Yizong Cheng and George M. Church, Biclustering of Expression Data, ISMB 2000.

[4] G. Getz, E. Levine and E. Domany, Coupled Two-Way Clustering Analysis of Gene Microarray Data, Proc. Natl. Acad. Sci. USA, 2000.

[5] Amos Tanay, Roded Sharan and Ron Shamir, Discovering Statistically Significant Biclusters in Gene Expression Data, Bioinformatics 2002.

[6] Jiong Yang, Haixun Wang, Wei Wang and Philip Yu, Enhanced Biclustering on Expression Data, BIBE 2003.

[7] Zonghong Zhang, Alvin Teo, Beng Chin Ooi, Kian-Lee Tan, Mining Deterministic Biclusters in Gene Expression Data, BIBE 2004.