CoBi: Pattern Based Co-Regulated Biclustering of Gene Expression Data

Swarup Roy[a,*], Dhruba K Bhattacharyya[b], Jugal K Kalita[c]

[a]*Dept of Information Technology, North Eastern Hill University, Shillong, 793022, Meghalaya, INDIA*
[b]*Dept of Computer Science & Engineering, Tezpur University, Napaam, 784 028, Assam, INDIA*
[c]*Dept of Computer Science, University of Colorado, Colorado Springs, USA*

Abstract

Co-regulation is a common phenomenon in gene expression. Finding positively and negatively co-regulated gene clusters from gene expression data is a real need. Existing techniques based on global similarity are unable to detect true up- and down-regulated gene clusters. This paper presents an expression pattern based biclustering technique, CoBi, for grouping both positively and negatively regulated genes from microarray expression data. Regulation pattern and similarity in degree of fluctuation are accounted for while computing similarity between two genes. Unlike traditional biclustering techniques, which use greedy iterative approaches, it uses a *BiClust* tree that needs single pass over the entire dataset to find a set of biologically relevant biclusters. Biclusters determined from different gene expression datasets by the technique show highly enriched functional categories.

[*]Corresponding author

Email addresses: swarup@nehu.ac.in,contact.swarup@gmail.com (Swarup Roy), dkb@tezu.ernet.in (Dhruba K Bhattacharyya), kalita@eas.uccs.edu (Jugal K Kalita)

Keywords: Biclustering, Co-regulation, Microarray, Expression pattern, Support, One-pass

1. Introduction

Clustering is a popular analysis tool in data mining applications (1, 2) such as scientific data exploration, information retrieval and text mining, spatial database applications, Web analysis, network security, marketing and medical diagnostics. Clustering techniques are also widely used in genomic studies, particularly in the context of microarray gene-expression data analysis (3, 4, 5, 6). Each microarray provides expression measurements for thousands of genes and clustering is a useful exploratory technique for analyzing gene expression data since it groups similar genes together and allows biologists to identify groups of potentially meaningful genes which have related functions or are co-regulated. This, in turn helps find relationships among genes in the form of gene regulatory networks (7). Another common use of cluster analysis is grouping samples (arrays) by similarity in expression patterns, i.e., finding groups of co-expressed genes.

A cluster is a group of objects that are similar to one another within the group but dissimilar to the objects of other groups (8, 9). Clustering is an unsupervised technique to discover hidden patterns. Some well known clustering approaches are partitional (10), hierarchical (11), grid based (12) and density based (9). Traditional clustering techniques are only effective in finding global patterns by maximizing the intra-class similarity and minimizing the

inter-class similarity. This similarity, calculated based on the entire set (space) of attributes, tends to overlook local patterns where different objects are similar based on only a subset (subspace) of attributes. It has frequently been observed that subsets of genes are co-regulated and co-expressed under a subset of environmental conditions or time points (13). However, clustering normally partitions genes into disjoint groups according to the similarity of their expressions across all conditions. Biclustering algorithms tackle the problem of finding a set of submatrices where each submatrix or bicluster meets a given homogeneity criterion. This special sub-class of clustering algorithms was originally introduced by Hartigan (14) and later successfully applied in different application areas such as text mining (15), collaborative filtering (16) and privacy preserving data mining (17).

Biclustering techniques are widely applied in gene expression data clustering. Cheng and Church (18) apply biclustering in expression data to capture the coherence of a subset of genes under a subset of conditions. In Cheng and Church's approach, the degree of coherence is measured using the concept of mean squared residue (MSR) and the algorithm greedily inserts or removes rows and columns to arrive at a certain number of biclusters achieving some predefined residue score. The lower the score, the stronger the coherence exhibited by the biclusters, and better is the quality of the biclusters. Followed by Cheng and Church, a number of biclustering techniques have been proposed (18, 19, 20, 21, 22, 23, 24, 25, 26, 27) to determine quality biclusters.

A greedy iterative search (18, 19) approach finds a local optimal solution with an expectation to finally obtain a globally good solution. A divide and conquer (14) approach divides the whole problem into sub-problems and solves them recursively. Finally, it combines all the solutions to solve the original problem. In exhaustive biclustering (26), the best biclusters are identified using exhaustive enumeration of all possible biclusters extant in the data, in exponential time. A detailed categorization of heuristic approaches is available in (20). A number of techniques based on metaheuristics such as evolutionary and multi-objective evolutionary frameworks have been explored (21) when generating and iteratively refining an optimal set of biclusters. All of them use MSR as the merit function. An MSR based technique is effective in finding optimized maximal biclusters. From a biological point of view, the interest resides in finding biclusters with subsets of genes showing similar behavior and not just similar values. Interesting and relevant patterns from a biological point of view, such as shifting and scaling patterns, may not be detected using this measure as it considers only expression values, not the pattern or tendency of gene expression profiles. It is important to discover this type of patterns because, frequently the genes show similar behavior although their expression levels vary in different ranges or magnitudes. Aguilar-Ruiz (22) has proved that the MSR is not a good measure in discovering patterns in data when the variance of gene values is high, that is, when the genes show scaling and shifting patterns. To detect biologically relevant biclusters with scaling and shifting patterns, a scatter search approach is proposed (23). This method uses a fitness function based on the linear correlation among genes and an improvement method to select just the positively correlated genes. Often, it has been observed that genes share local rather than global similarity in their expression profiles and only under a few conditions or time points. Thus, correlation based technique may not be effective when deciding pair wise similarity between two gene expression profiles. A few frequent itemset mining (1, 2, 28) based biclustering techniques have also been introduced (29, 27, 30). In addition, various pattern-based approaches have also been proposed (24, 25, 31, 32) for discovery of biclusters, where expression levels of genes rise and fall in a subset of conditions or time points.

It has been observed that (33) co-regulated genes also share negative patterns or inverted behaviors, which existing pattern based approaches are unable to detect. In this work, we capture biclusters of both positively and negatively regulated genes as co-regulated genes. A bicluster can be considered a quality bicluster only when participating genes exhibit consistent trends and similar degrees of fluctuation under consecutive conditions (34). We consider both up- and down-regulation trends and similar degrees of fluctuations under consecutive conditions for expression profiles of two genes as a measure of similarity between the genes. Available biclustering techniques are NP-complete (20) in nature requiring either large computational cost or use lossy heuristics approaches to minimize cost. Our approach deterministically finds all biclusters using a non-greedy approach. We use what we call a *BiClust* tree for generating biclusters in polynomial time with a single pass of the dataset.

2. Patterns in Expression Data

Biological processes are regulated in many ways. Examples include the control of gene expression, protein modification or interaction with protein or substrate molecules. Expression patterns with similar tendency or behavior are normally termed positively regulated and inverted behavior as negatively regulated. As described in Amigo[1], negative regulation or down regulation stops, prevents, or reduces the frequency, rate or extent of a biological process and positive regulation or up-regulation does the reverse. To illustrate the fact we consider examples of co-regulated clusters from a real microarray human datset, GDS825, given at the NCBI[2] website. A profile plot is given in Figure 1. In the figure, we easily observe that genes GALNT5 and IDH3B show similar patterns or positive co-expression patterns. On the other hand, IDH3B or GALNT5 show inverted or negative patterns with APOE. As suggested by gene ontology, the three genes are involved in *regulation of plasma lipoprotein particle levels* and *triglyceride-rich lipoprotein particle remodeling*. Pronounced inverted or negative patterns can be observed in Figure 2, taken from NCBI Rat dataset GDS3702. Gene ontology suggests that both are responsible for *regulation of interferon-beta production.* A group of genes may share a combination of both positive and negative co-regulation under a few conditions or at some time points. A majority of existing approaches try to capture genes with similar tendency. In this work, we address the issue of finding both up- and down-regulated gene groups as biclusters of co-regulated genes based on local patterns of gene expression profiles. Un-

[1] *http://amigo.geneontology.org/cgi-bin/amigo/term_details?term=GO:0048519*

[2] *www.ncbi.nlm.nih.gov*

like MSR or correlation based techniques, we use a pattern similarity based approach.

3. Biclustering of co-regulated genes

Let $G = \{G_1, G_2, \cdots G_N\}$ be a set of N genes and $T = \{T_1, T_2, \cdots, T_M\}$ be the set of M conditions or time points of a microarray gene expression dataset. The gene expression dataset D is represented as an $N \times M$ matrix $D_{N \times M}$ where each entry $d_{i,j}$ in the matrix corresponds to the logarithm of the relative abundance of mRNA of a gene.

For a given gene expression dataset D, biclustering finds a set of submatrices $\{(I_1, J_1), \cdots, (I_k, J_k)\}$ of the matrix $D_{N \times M}$ (with $I_i \subseteq N$, $J_i \subseteq M \; \forall i \{1, \cdots, k\}$), where each submatrix (bicluster) meets a given homogeneity criterion. Unlike traditional clustering approaches, biclustering attempts to cluster a set of genes which are similar under a subset of conditions or time points.

Traditional biclustering techniques normally use global similarity measures such as Euclidean distance, Pearson correlation or MSR. These measures sometimes fail to capture the true grouping. In addition, most existing techniques give less emphasis to pattern matching based on local similarity. It has been observed that the genes share local rather than global functional similarity in their gene expression profiles. Moreover, they share co-regulation in terms of up- and down-regulation. When computing similarity, well-known techniques do not consider a positive- or negative-regulation pattern as co-expression or co-regulation, with accompanying having biological significance. We try to capture the pair-wise similarity purely by pattern matching, followed by construction of biclusters by expanding co-regulated gene pairs. We consider both positive- and negative-regulation as co-regulation. In this paper, we develop a pattern similarity based approach to find biclusters among co-regulated genes.

We measure the similarity of two expressions based on the degree of fluctuation between the two and the regulation patterns of gene expression profiles. To capture the pattern of an expression profile, the edge between two consecutive expression values of a gene is considered. Thus, for an expression data with M conditions or time points, there are $(M-1)$ edges. The degree of fluctuation of an edge is the angular deviation of the edge in 180-degree normal plane. The regulation pattern represents the up, down and no regulation of a pattern or edge.

3.1. Terminology

Definition 1. (Pattern Similarity): Given degrees of fluctuation $A = \{a_1, a_2, \cdots, a_{M-1}\}$ and regulation patterns $R = \{r_1, r_2, \cdots, r_{M-1}\}$ of a gene, derived from gene expression profile, two genes' k^{th} expression patterns are similar if the difference in degrees of fluctuation of the two genes' k^{th} edge is less than some given threshold τ. In order to compute the differences in the degrees of fluctuation, we consider two cases: when the regulation patterns are the same (in case of up-regulation) and when the patterns are different (in case of down-regulation) under a particular edge. Mathematically it can be defined as follows:

$$sim(G_{ik}, G_{jk}) = \begin{cases} 1 & \text{if } |G_i(a_k) - G_j(a_k)| < \tau \\ & \text{when } G_i(r_k) = G_j(r_k) \text{ and} \\ & \text{if } |180 - G_i(a_k) + G_j(a_k)| < \tau \\ & \text{when } G_i(r_k) \neq G_j(r_k) \\ 0 & \text{Otherwise.} \end{cases} \quad (1)$$

Definition 2. (Co-regulated Bicluster): Given a gene expression dataset D of N genes and C conditions, a co-regulated bicluster is a sub-matrix of n genes and c conditions where the number of genes n, satisfies a user specified $MinGene$ criterion and the number of edges c, in the bicluster is greater than threshold θ, and all pairs of genes in the bicluster satisfy pattern similarity across all c edges.

$$CorBiClust(D_{N\times C}, MinGene, \theta) = \begin{array}{l} \{D_{n\times c} | \forall G_{i=1\cdots n} \in D_{n\times c}, |n| > MinGene, \\ |c| > \theta \wedge \; sim(G_{ik}, G_{jk}) = 1, \forall k = 1 \cdots (c-1)\}. \end{array} \quad (2)$$

3.2. Preprocessing

To capture patterns of each gene expression, researchers use either angles between the edges for every pair of conditions (30) or regulation patterns in terms of up- or down-regulation (26). Angles or regulation patterns between the edges of the two conditions alone, are ineffective in capturing the true expression pattern of a gene. We compare two gene expressions, both in terms of degrees of fluctuation and regulation patterns between two adjacent conditions (edges), simultaneously. To capture both regulation patterns and degree of fluctuation of each gene, we read rows of original data with M number of expression values or conditions and convert them into another row of $(M-1)$ columns, each column of which contains the degree of fluctuation and the regulation pattern of two adjacent conditions. We consider regulation information as triplet values [1, 0,-1] to represent up-regulation, no changes and down-regulation respectively. The regulation value in the k_{th} edge of a gene G_i , $G_i(r_k)$, based on two consecutive conditions (say, O_{k-1} and O_k), can be calculated as:

$$G_i(r_k) = \begin{cases} 1 & \text{if } O_{k-1} < O_k \\ 0 & \text{if } O_{k-1} = O_k \\ -1 & \text{if } O_{k-1} > O_k. \end{cases} \tag{3}$$

To calculate the degree of fluctuation, we compute the arc tangent between two adjacent expression levels (x, y) as in (30), on the 180 degree plane. For computing arctangent, we use a two-argument $atan2$ function. $atan2(y, x)$ is the angle between the positive x-axis of a plane and the point (x, y) on it, with positive sign for counter-clockwise angles and negative sign for clockwise angles. Next, we convert the angle in the 180 degree plane as follows:

$$DegreeOfFluctuation(x, y) = \begin{cases} 180 - abs(arctan2(y, x)) & \text{if } y < x \\ abs(arctan2(y, x)) & \text{otherwise.} \end{cases} \tag{4}$$

The fact is illustrated in Figure 3 with an example of a gene's expression values $G = \{343, 314, 409\}$ under three conditions. After preprocessing, the value of the expression become $G = \{138, -1;\ 52, 1\}$.

To find co-regulated biclusters based on pattern similarity, we use a BiClust tree based technique. The main advantage of the proposed technique is that it requires only a single scan of the database for finding biclusters.

4. Co-regulated biclustering using BiClust tree

BiClust tree is an m-way tree where each non-leaf node represents an edge or a set of edges and a leaf node represents a gene or a group of genes that are co-regulated or co-expressed under the edge or set of edges. CoBi starts by creating an initial BiClust tree as shown in Figure 4(a).

In the figure, four edges are shown as non-leaf nodes $E1, E2, E3$ and $E4$. We use a dataset D' to construct the initial BiClust tree BT. D' is a transformed dataset generated from the original dataset D to capture degrees of fluctuation and regulation from the expression pattern of each gene. The initial BiClust tree contains $(M-1)$ edges as initial non-leaf nodes for a dataset with M conditions or time points. The leaf nodes are created by forming a k^{th} cluster of genes based on similarity of genes under the k^{th} edge by using Equation (1). For each gene, it tries to form a cluster with other genes belonging to a particular cluster. Otherwise, it creates a new cluster when there are no matching clusters. Thus, multiple clusters or leaf nodes may be formed under a particular edge. The same process is repeated for all edges. $G1, G2$ and $G3$ form a cluster C_1, whereas $G4$ and $G5$ form another cluster C_2 under $E1$. When creating the k^{th}cluster, we transpose the dataset D', so that each row represents the degree of fluctuation and regulation pattern for all genes under each edge. By doing so, we can compare easily all genes' expression patterns under the k^{th} edge. Creating the initial BiClust tree requires a single pass over the dataset. No further consultation of the dataset is required in the following steps. To maintain a moderate number of gene clusters under an edge or a set of edges, it performs a pruning step. Cluster C_i is pruned if the cluster size is less than a user given threshold θ. Next, BT is expanded to produce biclusters using `ExpandCluster` function. The proposed technique, CoBi is shown in Algorithm 1.

In the cluster expansion phase, iteratively tree branches are merged to produce higher order biclusters. When merging two sub-trees, we apply merging in two ways, one at a non-leaf level and the other at the cluster level. Thus, from the initial BiClust tree, edges $E1$ and $E2$ are combined to form a new node $\{ E1, E2 \}$. Next, cluster leaf nodes under both nodes $E1$ and $E2$ are merged to get a new cluster node for $\{E1, E2 \}$. Cluster C_1 is compared with C_3 and C_4. A new cluster node $[G1, G2]$ is formed with all the elements that are common in both C_1 and C_3, or C_1 and C_4. In other words, it performs a intersection operation between the two clusters. Since the number of genes in a dataset is normally high compared to the number of conditions, the cluster list in the subtree is expected to be large. This is more critical especially in the initial stages of the tree. To handle the situation, we use a bit vector for storing gene IDs as a cluster. For merging we use the bitwise AND operation. It is very fast compared to perform normal intersection between two clusters. In order to merge two non-leaf edges, we use the concept of union taken from (35). The BiClust tree thus formed after the expansion of the initial BiClust tree is shown in Figure 4(b). The clusters that do not contain a minimum number of genes are pruned from the tree. During the merging of clusters under a non-leaf node, there may be a chance that a new cluster is formed such that its superset cluster is already present under the same non-leaf. Such subsets are redundant and removed. The process of sub-tree expansion continues until no further expansion is possible and all biclusters are stored in a list with a minimum number of condition θ. After the final expansion of a sub-tree, the biclusters are extracted from the list. The same process is applied to all sub-trees in the BiClust tree. A final BiClust tree is shown in Figure 4(c), where the minimum number of genes is two. The node $\{E1, E2, E4\}$ is pruned from the final tree as it contains a cluster with size one only. Other nodes are not shown in the final tree as they are pruned as well. The biclusters formed are: $\{E1, E2, E3\ \}$ $[G1, G2]$ and $\{E1, E3, E4\ \}$ $[G2, G3]$.

```
input  : $D'$ (Transformed Dataset), MinGene (Minimum number of
         Gene), $\theta$ (Minimum number of edge)
output: BiClust (List of Biclusters)
Construct initial BiClust tree BT;
Prune cluster $C_i$ from BT, if $|C_i| <$ MinGene;
BiClust = ExpandCluster (BT, MinGene,$\theta$) ;
BiClust = RemoveSubCluster (BiClust);
```

Algorithm 1: CoBi: Co-regulated Biclustering

The proposed method is shown in a compact manner in Algorithm 1. At first, CoBi, constructs an initial BiClust tree using the transformed database D'. The initial BiClust tree is pruned based on a user specified threshold `MinGene`. Next, the algorithm iteratively expands the tree to discover all biclusters. The `ExpandCluster` procedure is given in Algorithm 2. Two sub-

```
input  : BT (BiClust tree), MinGene (Minimum number of Gene),
         $\theta$ (Minimum number of edge)
output: BiClust (List of Biclusters)

Create a new BiClust tree BT' ;
foreach non-leaf node $E_i = 1 \rightarrow E_{n-1}$ of BT do
    Create a subtree ST of BT$'$ ;
    foreach non-leaf node $E_j = E_{i+1} \rightarrow E_n$ of BT do
        V = Merge($E_i$, $E_j$, MinGene) ;
        Prune subset of V ;
        Add V to ST;
    end
    Add ST to BT$'$;
end
foreach subtree ST$_i$ of BT$'$ do
    if ST$_i$ can expands further then
        BiClust = BiClust $\cup$ ExpandCluster(ST$_i$, MinGene, $\theta$);
    else
        return GetBiClusters(ST$_i$, $\theta$);
    end
end
```

Algorithm 2: ExpandCluster

trees are merged using `Merge` function and pruned when the number of genes in the merged tree is less than `MinGene`. Once the subtree reaches the end of expansion so that no further merging is possible, it extracts biclusters from the final BiClust subtree using `GetBiClusters` function. The same process is repeated for all subtrees. At the end, the `ExpandCluster` function returns the list of all biclusters generated. The biclusters returned may contain some redundant clusters, where genes in the clusters are the same, although the conditions or time points are a subset of the other. `RemoveSubCluster` function takes the list of biclusters and eliminates such clusters from the final list.

4.1. Complexity analysis

The complexity of the biclustering problem depends on the exact problem formulation, and particularly on the merit function used to evaluate the quality of a given bicluster. However, most interesting variants of this problem are NP-complete requiring either large computational effort or the use of lossy heuristics to short-circuit the calculation (20). Our approach deterministically finds all biclusters using a non-greedy approach in polynomial time. The cost of our algorithm consists of two parts: initial BiClust tree construction from D' (C_{IB}) and the cost for expanding the BiClust tree and extracting biclusters (C_{EX}).

(a) Construction of initial BiClust tree: Let us assume that the pre-processed dataset D' contains N genes and M edges. So, to scan the database, the cost is $(M * N)$. For creating clusters under an edge node, it requires the calculation of pattern similarity among all genes under an edge. Thus, the time requirement for creating clusters is N^2. The total time complexity for construction of the initial BiClust tree is $C_{IB} = O(M * N^2)$.

(b) BiClust tree expansion: Let us assume that the maximum number of iterations for the algorithm is k, which is the number of conditions in the final

bicluster. Let ζ be the number of edges or non-leaf nodes per iteration and the number of clusters under an edge be C. The cost of merging two clusters is $O(C^2)$. We observe that with increase in k, usually C decreases. The reason behind this is that compared to the number of clusters in $(k-1)$ steps, fewer clusters take part in the intersection in the k^{th} step. Thus the worst case complexity for bicluster expansion is no more than $C_{EX} = O(k * \zeta * C^2)$.

Most real microarray datasets contain a larger number of genes compared to the number of conditions. Scanning of the database is a costly activity. Although the complexity of the algorithm is polynomial, compared to the cost of database scanning, it is negligible.

5. Experimental Results

This section provides details of the experiments conducted, the data sets used and biological validation of the results. We use Java 1.6 running on a Windows 7, 2.53 GHz machine for implementation. A software implementation of CoBi as Java executable is available for download [3]. To demonstrate the effectiveness of CoBi in determining co-regulated and functionally enriched clusters, we use nine benchmark gene expression datasets. We analyze the results in terms of biological significance with the help of the GO annotation database. The ability of CoBi to find co-regulated biclusters is demonstrated visually using cluster profile plots. Since it is difficult to present all results, we present some significant find-

[3] https://sites.google.com/site/swarupnehu/publications/resources

ings from each dataset.

5.1. Datasets

Expression datasets are selected from four different organisms for our experiments. We use four different datasets belonging to ***Yeast*** and two from ***Homo Sapiens***. A short description of different gene expression datasets used in analysis is given in Table 1. Normalized expression datasets are used after removing all rows with missing values.

5.2. Input parameters

To obtain moderate sized biclusters, we avoid very small biclusters by setting the parameter $MinGene$ in the range of 3 to 5. During our experiments, we observe that higher number of edge matches in a bicluster gives more biologically significant biclusters. Thus, in most of the experiments, we try to keep the value of θ above 50% of the total number of edges or conditions present in the dataset. In order to calculate similarity between two expression profiles in terms of degree of fluctuation, we achieve good results with τ ranging between 15 to 25.

Below we present few results from our experiments. We first visualize the clusters and next evaluate the results in terms of statistical significance and biological relevance.

5.3. Cluster profile plot

A cluster profile plot shows for each bicluster the normalized expression values with respect to the conditions or time points that

are represented in the bicluster. In Figure 5, we present profile plots of some obtained biclusters. From the figure, we can observe that both positive and negative co-regulations are common in biological data and they are well captured by our technique.

5.4. Statistical significance

We use Gene Ontology (GO) and compute p-values (7) to evaluate the results. To determine the statistical significance of the association of a particular GO term with a group of genes in a cluster, we use online tools from the GO Project[4]. These tools use the hypergeometric distribution to calculate the p-value, which evaluates whether the clusters have significant enrichment in one or more function groups. The p-value is given as follows:

$$p = 1 - \sum_{i=0}^{k} \frac{\binom{f}{i}\binom{g-f}{n-i}}{\binom{g}{n}} \tag{5}$$

The p-value gives the probability of seeing at least k genes out of the total n genes in a cluster annotated with a particular GO term, given the total number of genes in the whole genome g and the number of genes in the whole genome that are annotated with that GO term f. It is important to note that p-value measures whether a cluster is enriched with genes from a particular category to a greater extent than what would be expected by chance. If the majority of genes in a cluster appear in one category, the p-value of the category is small. That is, the closer the p-value to zero,

[4] http://www.geneontology.org

the more the probability that the particular GO term is associated with the group of genes. In our experiments, we use the following tools: FuncAssociate (36), Fatigo (37), GOTermFinder (38) and OntoExpress (39).

Table 2 shows details of selected biclusters from different datasets obtained by applying our biclustering technique. For each bicluster, an identifier of the bicluster, the number of genes, the number of conditions, the volume and MSR score are presented. The MSR score can be used to compare the quality of the biclusters with those obtained by other algorithms. We also report Q value and the associated GO terms for some functionally enriched groups provided by the online tool GeneMANIA (40) in Table 3. The Q-value is the minimal False Discovery Rate (FDR) at which this gene appears significant. Q-values are estimated using the Benjamini Hochberg procedure (41).

5.5. Biological relevance

To evaluate biological significance of the results produced by our technique in terms of associated biological processes, cellular components, and gene function, we apply the Yeast GO term finder[5] to some of the biclusters from the sporulation data. Out of 22 genes from the cluster *Sp1*, the genes {YDR523C, YLR227C, YGR059W, YDR218C, YGL170C, YLR341W, YJL038C, YLR213C} are involved in the process of sporulation, anatomical structure formation involved in morphogenesis and cell differentiation, while

[5] http://www.yeastgenome.org/cgi-bin/GO/goTermFinder.pl

genes {YDR523C, YGL170C, YLR341W, YGR059W, YLR213C, YDR218C} are involved in sexual reproduction and sexual sporulation process resulting in formation of a cellular spore. On the other hand, genes {YCR002c, YGR059W, YDR218C} are involved in GTP binding and guanyl ribonucleotide binding and genes {YGL170C, YCR002c, YLR227C, YGR059W, YDR218C} take part in structural molecular activity. With respect to cellular component ontology, terms associated with genes {YDR523C, YCR002c, YGR059W, YDR218C} are ascospore-type prospore, intracellular immature spore, prospore membrane, septin complex. Similarly, from ***Sp2*** ({YDR523C, YGR225W, YLR227C, YPL027W, YLR343W, YDR516C, YDR218C, YNL204C, YGL170C, YIL099W, YCR002c, YDR260C, YJL038C, YLR213C, YOR242C, YNL225C, YGR059W, YLR054C, YNL128W, YOL132W, YLR308W, YMR017W, YLR341W}), the most significant biological processes are sporulation and anatomical structure formation involved in morphogenesis with a p-value 4.476e-19. GO terms observed in molecular function categories are glucanosyltransferase activity and 1,3-beta-glucanosyl transferase activity. In case of cellular components, genes {YDR523C, YMR017W, YCR002c, YGR059W, YLR314C, YPL027W, YLR054C, YDR218C} are involved in prospore membrane, intracellular immature spore and ascospore-type prospore formation. For the YeastKY dataset, we observe that a majority of the genes are involved in ribosome constituent activity with Q value 1.01e-119.

To verify the biological significance of the results from RatCNS data, we submitted our resulting biclusters to Onto-Express, and obtained a hierarchy of functional annotations in terms of GO for each cluster. An example of the GO tree for a co-regulated gene cluster *RatCNS1* is shown in Figure 6. We further investigated the genes in the clusters for *RatCNS2*. A majority of genes in *RatCNS2* are involved in the protein binding process and the rest of the genes are involved in activities like Calcium ion binding, growth factor activity, and transferase activity. Additional results are available for download[6].

5.6. Performance comparison

To evaluate performance of CoBi in comparison to other algorithms, we consider three popular biclustering techniques: Bimax (42), Cheng and Church (CC) (18) and OPSM (4) for the purpose. We used four Yeast datasets and the BicAT tool (43) for analysis. We compared performance based on functional enrichment of the biclusters. For the purpose of comparison, we set the parameter values of the other algorithms as recommended in the original papers. The functional enrichment of each bicluster is measured based on the Q-value associated with each GO category. For each bicluster, we calculated the average of the percentage of the number of genes from the biclusters with a given function against all genes in the genome with the function. Figure 7 shows the average of the functional enrichments of each bicluster obtained by different biclustering algorithms on four different

[6] https://sites.google.com/site/swarupnehu/publications/resources

datasets.

From the graphs, we observe that CoBi outperforms all three algorithms in obtaining functionally enriched biclusters. However, in case of YeastCho dataset, the Cheng and Church (CC) approach performs better than the other algorithms.

6. Conclusions

In this paper, we present a new biclustering technique, CoBi, that is capable of detecting positively as well as negatively co-regulated genes. Unlike traditional proximity measures such as MSR, Euclidean distance or correlation, it uses a pattern based approach for finding similarities among genes. Unlike available biclustering techniques, which are generally NP-complete in nature, it extracts all biclusters in polynomial time. To generate biclusters, it uses a tree-based algorithm called BiClust. An advantage of BiClust is that it requires a single pass over the database to generate all biclusters. The results establish that co-regulated biclusters are significant from statistical and biological points of view. Work is underway to develop a user friendly tool based on CoBi that may help biologists in finding interesting patterns over a large number of gene expression datasets. In addition, there is an ongoing effort to introduce a similarity measure to effectively handle both shifting and scaling patterns including positive- and negative-regulations with minimum computational cost. We are also working towards exploiting the advantages of BiClust trees to develop a one pass

technique to find all frequent itemsets from market basket data.

Tuning and extension of our biclustering technique to apply to other application domains, including information retrieval, text mining, collaborative filtering, target marketing, market research, database research and data mining is certainly one of the important open issues for future research.

References

[1] J. Han, M. Kamber, Data mining: concepts and techniques, Morgan Kaufmann, 2006.

[2] S. Roy, D. K. Bhattacharyya, Data mining techniques and its application in medical imagery, VDM Verlag Dr. Muller Germany, 2010.

[3] M. B. Eisen, P. T. Spellman, P. O. Brown, D. Botstein, Cluster analysis and display of genome-wide expression patterns, Proc. National Academy of Sciences 95 (25) (1998) 14863–14868.

[4] A. Ben-Dor, R. Shamir, Z. Yakhini, Clustering gene expression patterns, J of computational biology 6 (3-4) (1999) 281–297.

[5] A. Brazma, J. Vilo, et al., Gene expression data analysis, FEBS letters 480 (1) (2000) 17–24.

[6] H. Chipman, T. Hastie, R. Tibshirani, Clustering microarray data, Chapman & Hall/CRC, Boca Raton, Fla, 2003.

[7] S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, G. M. Church, et al., Systematic determination of genetic network architecture, Nature genetics 22 (1999) 281–285.

[8] A. Jain, Data clustering: 50 years beyond k-means, Pattern Recognition Letters 31 (8) (2010) 651–666.

[9] S. Roy, D. K. Bhattacharyya, An approach to find embedded clusters using density based techniques, in: Distributed Computing and Internet Technology, Proc. Intl. Conf. on ICDCIT'05, Springer, 2005, pp. 523–535.

[10] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, Proc. 5th Berkeley symposium on mathematical statistics and probability 1 (281-297) (1967) 14.

[11] S. Guha, R. Rastogi, K. Shim, Cure: an efficient clustering algorithm for large databases, ACM SIGMOD Record 27 (2) (1998) 73–84.

[12] G. Sheikholeslami, S. Chatterjee, A. Zhang, Wavecluster: A multi-resolution clustering approach for very large spatial databases, in: VLDB, Proc. Intl. Conf. on, IEEE, 1998, pp. 428–439.

[13] S. Mitra, H. Banka, Multi-objective evolutionary biclustering of gene expression data, Pattern Recognition 39 (12) (2006) 2464–2477.

[14] J. A. Hartigan, Direct clustering of a data matrix, J. Am. Stat. Assoc 67 (337) (1972) 123–129.

[15] I. S. Dhillon, Co-clustering documents and words using bipartite spectral graph partitioning, in: Knowledge discovery and data mining, Proc. 7th ACM SIGKDD Intl. Conf. on, ACM, 2001, pp. 269–274.

[16] T. George, S. Merugu, A scalable collaborative filtering framework based on co-clustering, in: Data Mining, Fifth IEEE International Conference on, IEEE, 2005, pp. 4–pp.

[17] W. Ahmad, A. Khokhar, Phoenix: Privacy preserving biclustering on horizontally partitioned data, Privacy, Security, and Trust in KDD (2008) 14–32.

[18] Y. Cheng, G. M. Church, Biclustering of gene expression data, in: Proc. Conf. Intelligent Systems for Molecular Biology, 2000, pp. 93–103.

[19] J. Yang, H. Wang, W. Wang, P. Yu, Enhanced biclustering on expression data, in: Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on, IEEE, 2003, pp. 321–327.

[20] S. C. Madeira, A. L. Oliveira, Biclustering algorithms for biological data analysis: a survey, IEEE Trans. Comput. Biol. Bioinformatics 1 (2004) 24–45.

[21] H. Banka, S. Mitra, Evolutionary biclustering of gene expressions, Ubiquity 7 (42) (2006) 1–12.

[22] J. Aguilar-Ruiz, Shifting and scaling patterns from gene expression data, Bioinformatics 21 (20) (2005) 3840–3845.

[23] J. A. Nepomuceno, A. Troncoso, J. Aguilar-Ruiz, Biclustering of gene expression data by correlation-based scatter search, BioData Mining 4 (3).

[24] J. Pei, X. Zhang, M. Cho, H. Wang, P. S. Yu, Maple: A fast algorithm for maximal pattern-based clustering, in: Data Mining, 2003. ICDM 2003. Third IEEE Intl. Conf. on, IEEE, 2003, pp. 259–266.

[25] H. Wang, F. Chu, W. Fan, P. S. Yu, J. Pei, A fast algorithm for subspace clustering by pattern similarity, in: Scientific and Statistical Database Management, 2004. Proc. 16th Intl. Conf. on, IEEE, 2004, pp. 51–60.

[26] A. Tanay, R. Sharan, R. Shamir, Discovering statistically significant biclusters in gene expression data, Bioinformatics 18 (Suppl 1) (2002) S136–S144.

[27] S. Roy, D. K. Bhattacharyya, J. K. Kalita, Deterministic approach for biclustering of co-regulated genes from gene expression data, in: KES12, Knowledge-based and intelligent information & engg. system, Proc. of 16th Int. Conf. on, FAIA, Vol. 243, IOS Press, 2012, pp. 490–499.

[28] S. Roy, D. K. Bhattacharyya, Opam: an efficient one pass association mining technique without candidate generation, J. convergence information technology 3 (3) (2008) 32–38.

[29] G. Pandey, G. Atluri, M. Steinbach, C. L. Myers, V. Kumar, An association analysis approach to biclustering, in: Knowledge discovery and data mining, Proc. 15th ACM SIGKDD international conference on, ACM, 2009, pp. 677–686.

[30] Z. Zhang, A. Teo, B. C. Ooi, K.-L. Tan, Mining deterministic biclusters in gene expression data, in: Bioinformatics and Bioengineering, 2004. BIBE 2004. Proc. Fourth IEEE Symposium on, IEEE, 2004, pp. 283–290.

[31] H. Wang, W. Wang, J. Yang, P. S. Yu, Clustering by pattern similarity in large data sets, in: Proc. 2002 ACM SIGMOD International conference on Management of data, ACM, 2002, pp. 394–405.

[32] Y. Zhao, J. X. Yu, G. Wang, L. Chen, B. Wang, G. Yu, Maximal subspace coregulated gene clustering, Knowledge and Data Engineering, IEEE Transactions on 20 (1) (2008) 83–98.

[33] H. Yu, N. M. Luscombe, J. Qian, M. Gerstein, Genomic analysis of gene expression relationships in transcriptional regulatory networks, TRENDS in Genetics 19 (8) (2003) 422–427.

[34] L. Ji, K.-L. Mock, K.-L. Tan, Quick hierarchical biclustering on microarray gene expression data, in: BioInformatics and BioEngineering, 2006. BIBE 2006. Sixth IEEE Symposium on, IEEE, 2006, pp. 110–120.

[35] R. Agrawal, R. Srikant, et al., Fast algorithms for mining association rules, in: Proc. 20th Int. Conf. Very Large Data Bases, VLDB, Vol. 1215, 1994, pp. 487–499.

[36] G. Berriz, O. King, B. Bryant, C. Sander, F. Roth, Characterizing gene sets with funcassociate, Bioinformatics 19 (18) (2003) 2502–2504.

[37] F. Al-Shahrour, R. Díaz-Uriarte, J. Dopazo, Fatigo: a web tool for

finding significant associations of gene ontology terms with groups of genes, Bioinformatics 20 (4) (2004) 578–580.

[38] E. Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, J. Cherry, G. Sherlock, Go termfinder: open source software for accessing gene ontology information and finding significantly enriched gene ontology terms associated with a list of genes, Bioinformatics 20 (18) (2004) 3710–3715.

[39] P. Khatri, S. Draghici, G. Ostermeier, S. Krawetz, Profiling gene expression using onto-express, Genomics 79 (2) (2002) 266–270.

[40] D. Warde-Farley, et al., The genemania prediction server: biological network integration for gene prioritization and predicting gene function, Nucleic acids research 38 (suppl 2) (2010) W214–W220.

[41] Y. Benjamini, Y. Hochberg, Controlling the false discovery rate: a practical and powerful approach to multiple testing, Journal of the Royal Statistical Society. Series B (Methodological) (1995) 289–300.

[42] A. Prelić, S. Bleuler, et al., A systematic comparison and evaluation of biclustering methods for gene expression data, Bioinformatics 22 (9) (2006) 1122–1129.

[43] S. Barkow, S. Bleuler, A. Prelić, P. Zimmermann, E. Zitzler, Bicat: a biclustering analysis toolbox, Bioinformatics 22 (10) (2006) 1282–1283.

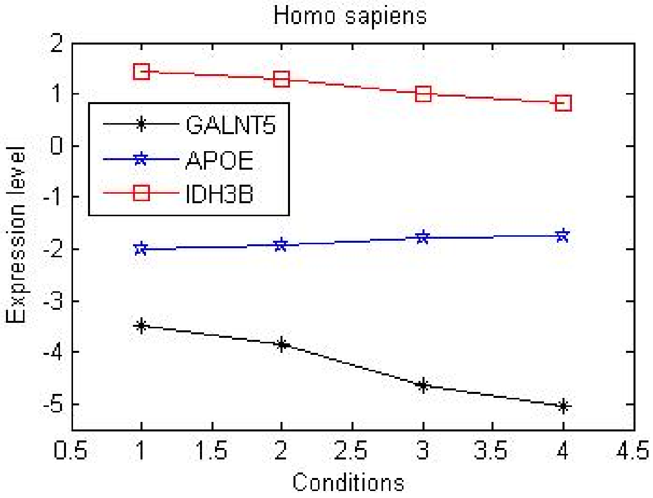


Figure 1: Human genes showing positive- and negative-regulation

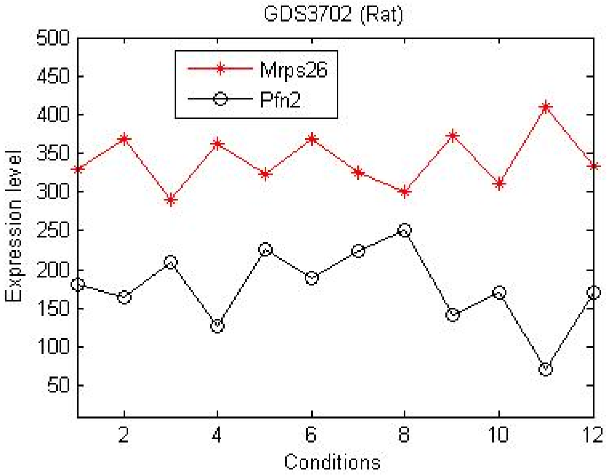


Figure 2: Expression profile of RAT genes showing negative-regulation

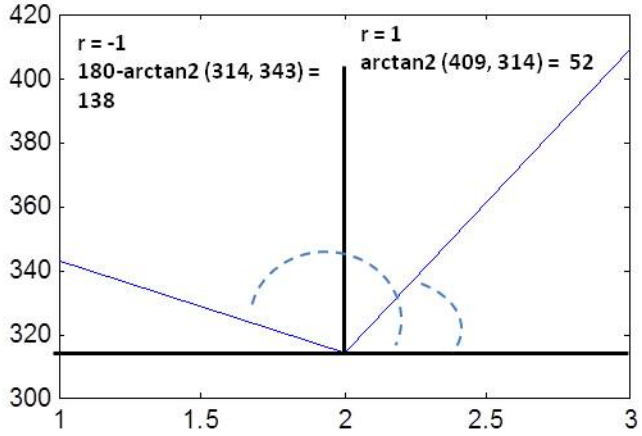
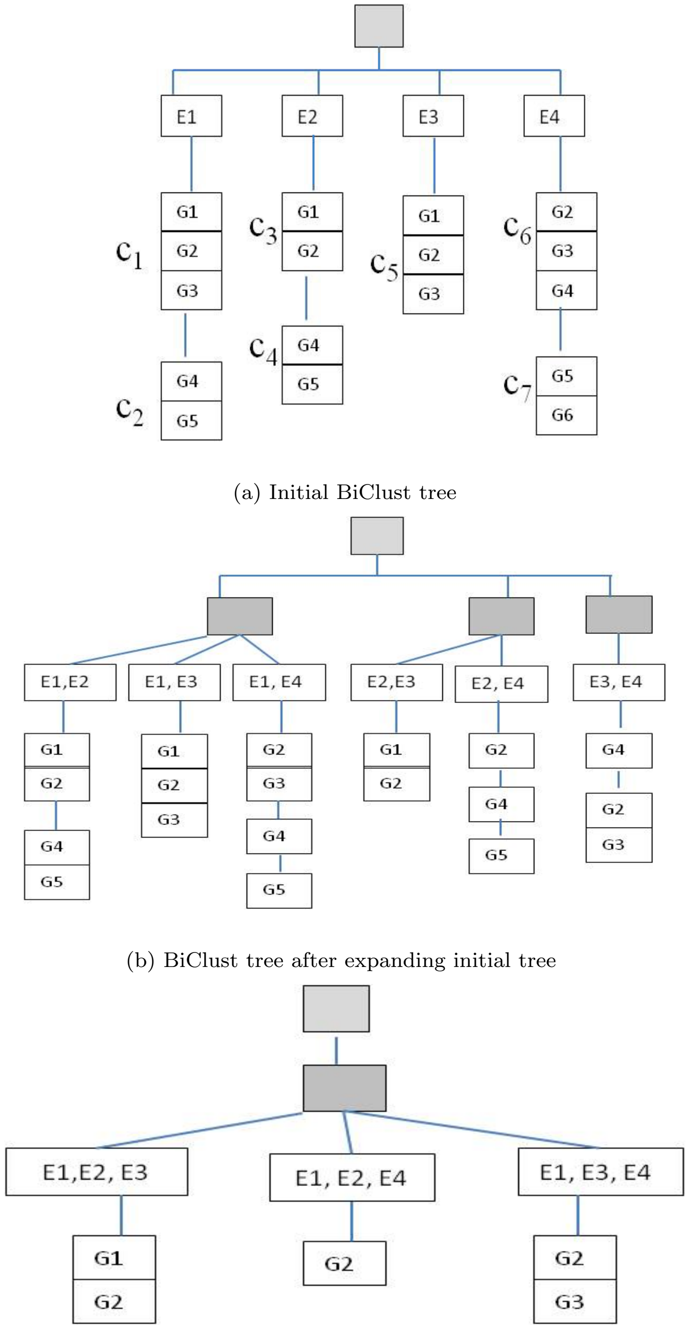


Figure 3: Degree of fluctuation for three expression values of a gene



(a) Initial BiClust tree

(b) BiClust tree after expanding initial tree

(c) Final BiClust tree

Figure 4: Stages of Biclust tree

Table 1: Short description of the datasets

Organism	Dataset	No. of genes	No. of samples	Source
Yeast	YeastDB	2884	17	http://arep.med.harvard.edu/biclustering/yeast.matrix
	Sporulation	474	7	http://cmgm.stanford.edu/pbrown/sporulation
	Yeast_KY	237	17	http://faculty.washington.edu/kayee/cluster/
	YeastCho (cell cycle)	384	17	http://faculty.washington.edu kayee/cluster
Rat	Rat_CNS	112	9	http://faculty.washington.edu/kayee/cluster
Human	GDS3712	325	12	NCBI
	Fibroblast Serum	517	13	http://www.sciencemag.org/feature/data/984559.hsl/
Mouse	GDS958	308	12	NCBI
Rice	Thaliana	138	8	http://homes.esat.kuleuven.be/s̃istawww/bioi/thijs/Work/Clustering.html

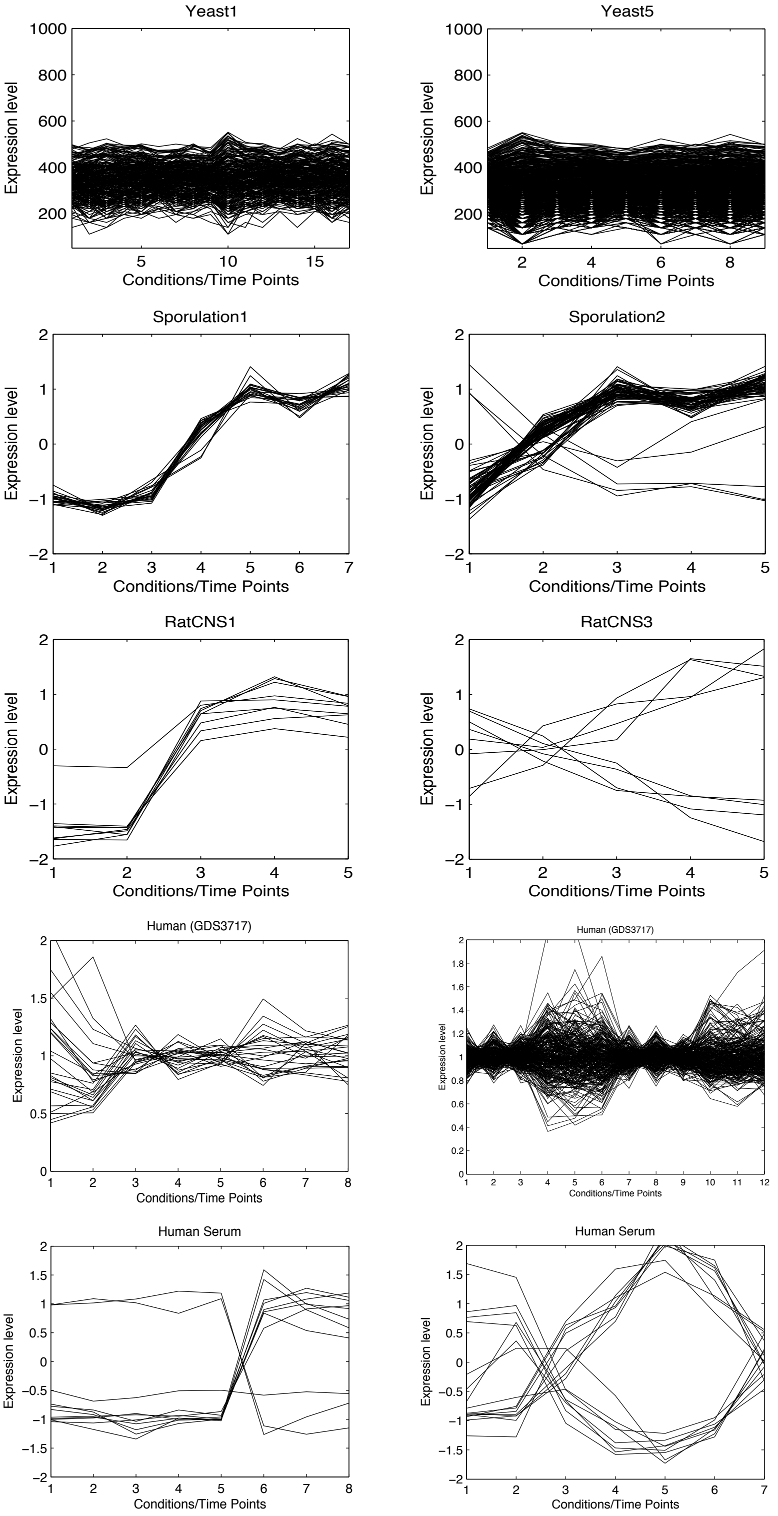


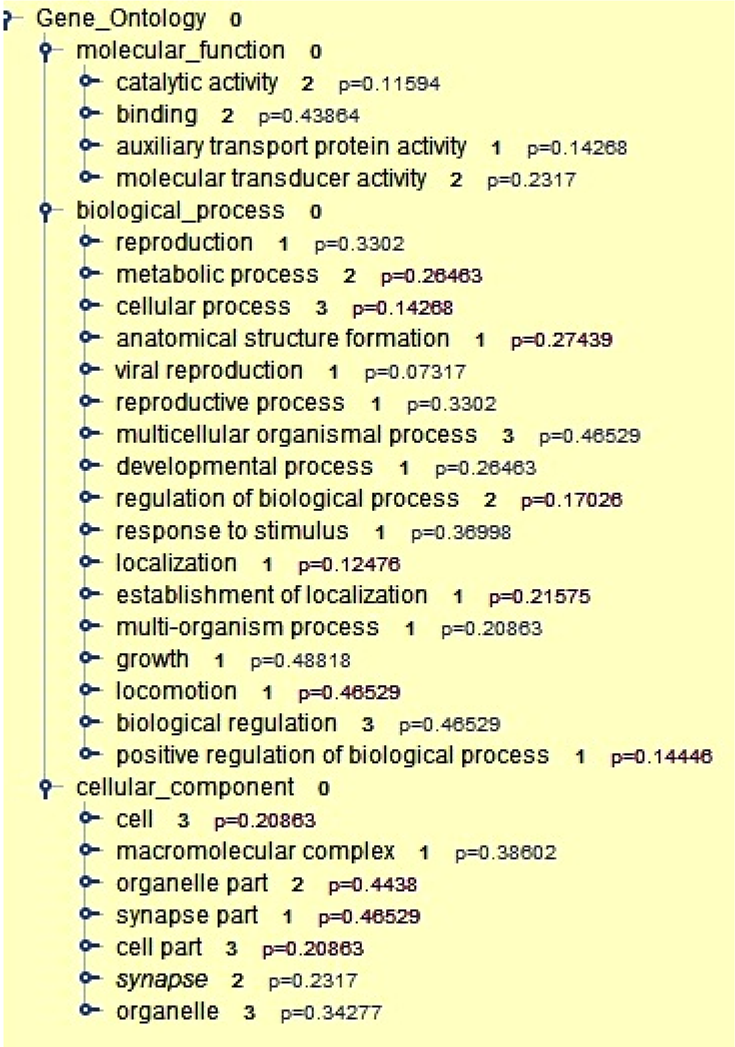
Figure 5: Expression profile plots of biclusters from Yeast, Yeast Sporulation, RatCNS, GDS3717 and Fibroblast Serum data

Table 2: Biclusters results from Yeast, Sporulation and Rat CNS data

Dataset	Bicluster Id	No. of Gene	No. of Cond.	Volume	MSR	p-value	GO attributes
YeastDB	*YDB1*	268	17	4556	654.41	2.075e-9	Cytoplasmic translation
	YDB2	343	15	5145	664.20	3.318e-7	Ribosome
	YDB3	430	13	5590	608.91	8.960e-7	Structural constituent of ribosome
Sporulation	*Sp1*	22	7	154	0.01557	4.543e-9	Cellular development process
	Sp2	69	5	345	0.1285	4.476e-19	Anatomical structure formation for morphogenesis
Rat CNS	*RatCNS1*	9	5	45	0.051	6.81e-4	Male sex determination
	RatCNS2	12	4	48	0.233	4.71e-4	Insulin receptor substrate binding

Table 3: Q-values and GO attributes from different biclusters

Dataset	Bicluster Id	Q-value	GO attributes
	Mouse1	2.18e-12	cytosolic part and ribosomal subunit formation
GDS958	*Mouse2*	5.57e-7	nuclear DNA-direct RNA polymerase complex
	Mouse3	1.76e-6	proteasome complex
	Rat1	1.82e-14	regulation of neuron apoptosis
Rat CNS	*Rat2*	3.59e-14	regulation neurological system process
	Rat3	1.14e-13	positive regulation of glucose import
	Rat4	5.27e-10	growth factor binding
	Cho1	4.03e-10	chromosomal part
YeastCho	*Cho2*	2.38e-10	DNA repair
	Cho3	4.23e-6	protein glycosylation
	SP1	4.48e-19	anatomical structure formation
Sporulation	*SP2*	8.86e-18	cellular component assembly involved in morphogenesis
	SP3	4.54e-9	cellular developmental process
YeastKY	*KY1*	**1.01e-119**	Structural constituents of ribosome
	KY2	**1.83E-110**	ribosome
	Th1	4.19e-13	glutathione transferase activity
Thaliana	*Th2*	6.69e-08	toxin catabolic process, glutathione transferase activity
	Th3	1.32e-6	glutathione transferase activity



- Gene_Ontology 0
  - molecular_function 0
    - catalytic activity 2 p=0.11594
    - binding 2 p=0.43864
    - auxiliary transport protein activity 1 p=0.14268
    - molecular transducer activity 2 p=0.2317
  - biological_process 0
    - reproduction 1 p=0.3302
    - metabolic process 2 p=0.26463
    - cellular process 3 p=0.14268
    - anatomical structure formation 1 p=0.27439
    - viral reproduction 1 p=0.07317
    - reproductive process 1 p=0.3302
    - multicellular organismal process 3 p=0.46529
    - developmental process 1 p=0.26463
    - regulation of biological process 2 p=0.17026
    - response to stimulus 1 p=0.36998
    - localization 1 p=0.12476
    - establishment of localization 1 p=0.21575
    - multi-organism process 1 p=0.20863
    - growth 1 p=0.48818
    - locomotion 1 p=0.46529
    - biological regulation 3 p=0.46529
    - positive regulation of biological process 1 p=0.14446
  - cellular_component 0
    - cell 3 p=0.20863
    - macromolecular complex 1 p=0.38602
    - organelle part 2 p=0.4438
    - synapse part 1 p=0.46529
    - cell part 3 p=0.20863
    - *synapse* 2 p=0.2317
    - organelle 3 p=0.34277

Figure 6: Significant GO terms on molecular function, biological process and cellular component from RatCNS1

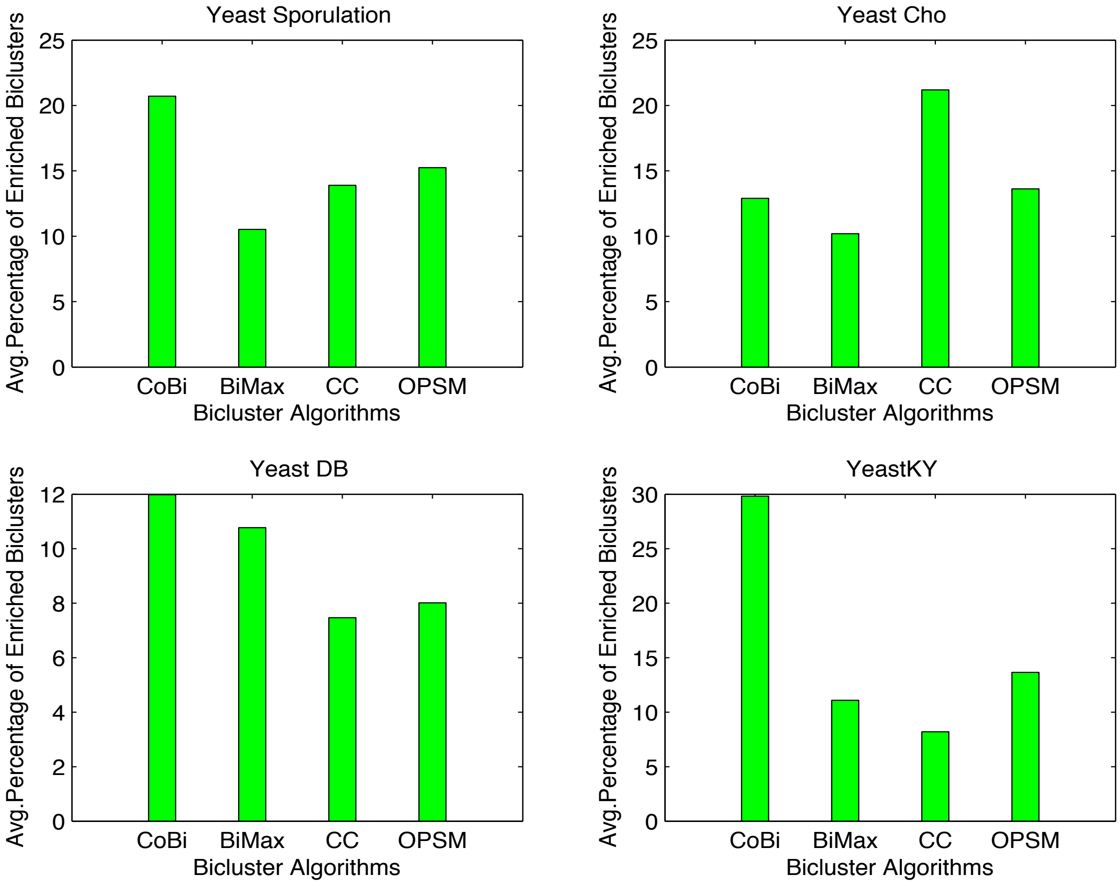


Figure 7: Comparison on functionally enriched biclusters from different biclustering techniques