Serin and Vingron *Algorithms for Molecular Biology* 2011, **6**:18
http://www.almob.org/content/6/1/18

ALGORITHMS FOR MOLECULAR BIOLOGY

RESEARCH Open Access

DeBi: Discovering Differentially Expressed Biclusters using a Frequent Itemset Approach

Akdes Serin[*] and Martin Vingron

Abstract

Background: The analysis of massive high throughput data via clustering algorithms is very important for elucidating gene functions in biological systems. However, traditional clustering methods have several drawbacks. Biclustering overcomes these limitations by grouping genes and samples simultaneously. It discovers subsets of genes that are co-expressed in certain samples. Recent studies showed that biclustering has a great potential in detecting marker genes that are associated with certain tissues or diseases. Several biclustering algorithms have been proposed. However, it is still a challenge to find biclusters that are significant based on biological validation measures. Besides that, there is a need for a biclustering algorithm that is capable of analyzing very large datasets in reasonable time.

Results: Here we present a fast biclustering algorithm called DeBi (Differentially Expressed BIclusters). The algorithm is based on a well known data mining approach called frequent itemset. It discovers maximum size homogeneous biclusters in which each gene is strongly associated with a subset of samples. We evaluate the performance of DeBi on a yeast dataset, on synthetic datasets and on human datasets.

Conclusions: We demonstrate that the DeBi algorithm provides functionally more coherent gene sets compared to standard clustering or biclustering algorithms using biological validation measures such as Gene Ontology term and Transcription Factor Binding Site enrichment. We show that DeBi is a computationally efficient and powerful tool in analyzing large datasets. The method is also applicable on multiple gene expression datasets coming from different labs or platforms.

Background

In recent years, various high throughput technologies such as cDNA microarrays, oligo-microarrays and sequence-based approaches (RNA-Seq) for transcriptome profiling have been developed. The most common approach for detecting functionally related gene sets from such high throughput data is clustering [1]. Traditional clustering methods like hierarchical clustering [2] and k-means [3], have several limitations. Firstly, they are based on the assumption that a cluster of genes behaves similarly in all samples. However, a cellular process may affect a subset of genes, only under certain conditions. Secondly, clustering assigns each gene or sample to a single cluster. However, some genes may not be active in any of the samples and some genes may participate in multiple processes.

Biclustering is a two-way clustering method for detecting local patterns in data. It finds subsets of genes that behave similarly in subsets of samples. Biclustering was initially introduced by Hartigan [4]. However, it was first applied by Cheng and Church [5] on gene expression data. Cheng and Church tried to identify submatrices of low mean residue score which indicates uniform fluctuation in expression profiles. Since the algorithm discovers one bicluster at a time, repeated application of the method on a modified matrix is needed for discovering multiple biclusters. This has the drawback that it results in highly overlapping gene sets. Ben-Dor et al. [6] detected a subset of genes whose expression levels induce the same linear ordering of the experiments. The drawback of this method is that it enforces a strict order of the samples. Bergmann et al. [7] identified biclusters which consist of the set of co-regulated genes and the conditions that induce their co-regulation. Murali and Kasif [8] found subsets of genes that are

[*] Correspondence: serin@molgen.mpg.de
Max Planck Institute for Molecular Genetics, Ihnestrasse 63-73, 14195 Berlin, Germany

simultaneously similarly expressed across a subset of the samples. The algorithm uses prior knowledge about the sample phenotypes. Tanay et al. [9] defined biclustering as a problem of finding bicliques in a bipartite graph. Due to its high complexity, the number of rows the bicluster may have is restricted. Prelic et al. [10] defined the binary inclusion maximal biclustering (BIMAX) using a fast divide and conquer method. However, divide and conquer has the drawback of possibly missing good biclusters by early splits. Li et al. [11] developed an algorithm for discovering statistically significant biclusters from datasets containing tens of thousands of genes and thousands of conditions. Madeira and Oliveira have written a detailed review on different biclustering methods [12].

Here, we propose a novel, fast biclustering algorithm called DeBi that utilizes differential gene expression analysis. In DeBi, a bicluster has the following two main properties. Firstly, a bicluster is a maximum homogenous gene set where each gene in the bicluster should be highly or lowly expressed over all the bicluster samples. Secondly, each gene in the bicluster shows statistical difference in expression between the samples in the bicluster and the samples not in the bicluster. Differentially expressed biclusters lead to functionally more coherent gene sets compared to standard clustering or biclustering algorithms.

There are several advantages of the DeBi algorithm. Firstly, the algorithm is capable of discovering biclusters on very large datasets such as the human connectivity map data with 22283 genes and 6100 samples in reasonable time. Secondly, it is not required to define the number of biclusters a priori [5,7,10].

We evaluated the performance of DeBi on a yeast dataset [13], on synthetic datasets [10], on the connectivity map dataset which is a reference collection of gene expression profiles from human cells that have been treated with a variety of drugs [14], gene expression profiles of 2158 human tumor samples published by expO (Expression Project for Oncology), on diffuse large B-cell lymphoma (DLBCL) dataset [15] and on gene sets from the Molecular Signature Database (MSigDB) C2 category. We show that DeBi compares well with existing biclustering methods such as BIMAX, SAMBA, Cheng and Church's algorithm (CC), Order Preserving Submatrix Algorithm (OPSM), Iterative Signature Algorithm (ISA) and Qualitative Biclustering (QUBIC) [5-7,9,10].

Results

We have evaluated our algorithm on six datasets (a) Prelic's benchmark synthetic datasets with implanted biclusters [10] (b) 300 different experimental perturbations of S. cerevisiae [13] (c) diffuse large B-cell lymphoma (DLBCL) dataset [15] (d) a reference collection of gene-expression profiles from human cells that have been treated with a variety of drugs [14] (e) gene expression profiles of 2158 human tumor samples published by expO (Expression Project for Oncology) (f) gene sets from the Molecular Signature Database (MSigDB) C2 category. The synthetic data is studied to show the performance of our algorithm in recovering implanted biclusters. Additionally, the effect of overlap between biclusters and noise on the performance of the algorithm can be studied using the synthetic data. The yeast and human gene expression datasets are studied to evaluate the biological relevance of the biclusters from several aspects. We used a fold-change of 2 for binarizing the datasets. The set of biclusters generated by all the algorithms are filtered such that the remaining ones have a maximum overlap of 0.5. (unless specified otherwise)

First, for each bicluster we calculated the statistically significantly enriched Gene Ontology (GO) terms using the hypergeometric test. We determined the proportion of GO term enriched biclusters at different levels of significance. Second, Transcription Factor Binding Sites (TFBS) enrichment is calculated by a hypergeometric test using transcription factor binding site data coming from various sources [16-18] at different levels of significance. The GO term and TFBS enrichment analyses are done using Genomica http://genie.weizmann.ac.il.

We have compared our algorithm with BIMAX, SAMBA, Cheng and Churchs algorithm (CC), Order Preserving Submatrix Algorithm (OPSM), Iterative Signature Algorithm (ISA) and Qualitative Biclustering (QUBIC) [5-7,9,10]. We used QUBIC software for QUBIC, BicAT software for OPSM, ISA, BIMAX and Expander software for SAMBA with default settings for each algorithm [10,19,20].

Prelic's Synthetic Data

We applied our algorithm to a synthetic gene expression dataset. In the artificial datasets biclusters have been created on the basis of two scenarios (data available at http://www.tik.ee.ethz.ch/sop/bimax. In the first scenario, non-overlapping biclusters with increasing noise levels are generated. In the second scenario, biclusters with increasing overlap but without noise are produced. In both scenarios, biclusters with constant expression values and biclusters following an additive model where the expression values varying over the conditions are investigated.

In order to assess the performance of different biclustering algorithms, we used two measures from Prelic et al. [10] and Hochreiter et al. [21], respectively. The measure introduced by Prelic et al. calculates a similarity based on the Jaccard index between the computed biclusters and the implanted biclusters. Bicluster recovery score measures the accuracy of the predicted biclusters however it does not consider the number of biclusters in both sets. Hochreiter et al. introduced a consensus score by computing similarities between all pairs of biclusters and then assigning the biclusters of one set to biclusters of the other set. It penalizes different number of biclusters by dividing the sum of similarities by the numbers of biclusters in largest set. A more detailed description of the measures can be found in Additional File 1.

In Figures 1 and 2 the performance of BIMAX, ISA, SAMBA, DeBi, OPSM and QUBIC algorithms on the synthetic data is summarized based on Prelic et al. recovery score and Hochreiter et al. consensus score. The set of biclusters generated by these algorithms are filtered such that the remaining ones have a maximum overlap of 0.25. In the Prelic et al. paper, after the filtering process the largest 10 biclusters are chosen. Since the bicluster number is not known a priori, we have considered all the filtered biclusters. We did not evaluate xMotif and CC algorithms since they have been shown to perform badly in all the scenarios, mostly below 50% of recovery accuracy [10]. The CC and xMotif algorithms produce large biclusters containing genes that are not expressed. ISA and QUBIC give high Prelic et al. recovery score and Hochreiter et al. consensus score in all scenarios. SAMBA has a lower Hochreiter et al. consensus score compared to its Prelic et al. recovery score. The reason is that, Hochreiter et al. consensus score takes into account both gene and condition dimensions and SAMBA is not very accurate in recovering the biclusters in condition dimension. In the absence of noise with an increasing overlap degree, BIMAX has a high performance based on Prelic et al. and Hochreiter et al. scores. However, BIMAX estimates a large number of biclusters upon increasing noise level. The comparision of the estimated number of biclusters given by the algorithms with the true number of biclusters under all the scenarios can be found in Figure S1 in Additional File 1. In the absence of overlap with increasing noise levels, DeBi is able to identify 99% of implanted biclusters both in additive and constant model. High degree of overlap decreases the performance of DeBi because it considers the overlapping part of the biclusters as a seperate bicluster. The DeBi biclustering results can be found in Additional file 2.

Yeast Compendium

We further applied our algorithm to the compendium of gene expression profiles derived from 300 different experimental perturbations of *S. cerevisiae* [13]. We discovered 192 biclusters in the yeast dataset containing 2025 genes and 192 conditions. As a binarization level we used the fold change of 1.58 as recommended in the original paper [13].

Figure 3 (a) illustrates the proportion of GO term and TFBS enriched biclusters for the six selected biclustering methods (ISA, OPSM, BIMAX, QUBIC, SAMBA and DeBi) at different levels of significance. DeBi performs the second best based on biological validation measures. BIMAX discovers a higher proportion of GO term and TFBS enriched biclusters. All the biclusters, the enrichment analysis can be found in Additional file 3.

In the analyzed yeast data, conditions are knocked-out genes. Since biclustering discovers subsets of genes and subsets of conditions we can also examine the biological significance of the clustered conditions. Similar to the previous analysis, we measured GO term enrichment of conditions in each discovered biclusters. DeBi is the second best in discovering high percentage of GO term enriched biclusters.

In the discovered biclusters, the enriched gene functions are related to the enriched sample functions. Bicluster 83, genes are enriched in the ‘conjugation’ GO term and conditions are enriched in ‘regulation of biological quality’. Moreover, there is an enrichment of the TFBS of STE12, which is known to be involved in cell cycle. Bicluster 50, consists of genes and samples that are enriched in ‘ribosome biogenesis and assembly’ GO term. Bicluster 22, consists of genes and samples that are enriched in ‘lipid metabolic process’ GO term, and additionally genes are enriched with TFBS of HAP1. Bicluster 9, consists of down regulated genes and samples that are enriched in ‘cell division’ GO term, and additionally genes are enriched with TFBS of STE12.

DLBCL Data

We also evaluated our DeBi algorithm on ‘diffuse large B-cell lymphoma’ (DLBCL) dataset. DLBCL dataset consists of 661 genes and 180 samples. We applied ISA, OPSM, QUBIC, SAMBA and DeBi algorithms.

Figure 3 (b) illustrates the proportion of GO term and TFBS enriched biclusters for the five biclustering methods at different levels of significance. DeBi discovers the highest proportion of GO term and TFBS enriched biclusters. The up regulated bicluster 16 and down regulated bicluster 4 contains the sample classes identified by [22]. Bicluster 16 is enriched with ‘ribosome’ and ‘cell cycle’ GO Term and Bicluster 4 is enriched with ‘cell cycle’ and ‘death’ GO Terms. The protein interaction networks of this two selected biclusters can be found in Figure S2 and S3 Additional File 1. Protein interaction networks are generated using STRING [23]. All the biclusters and the enrichment analysis can be found in Additional file 4.

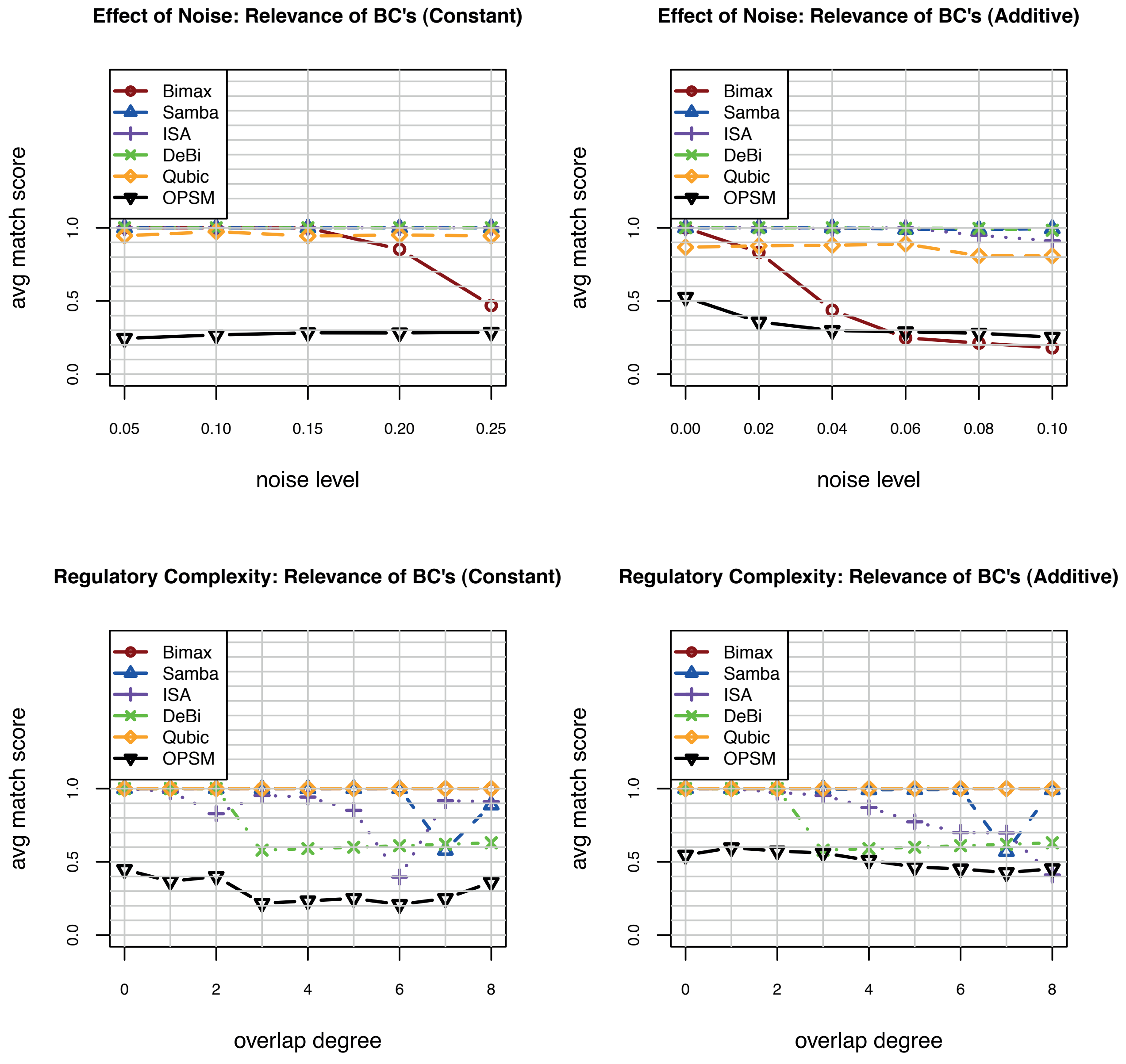

Figure 1 Bicluster recovery accuracy score on synthetic data. The synthetic data have been created based on two scenories (a) and (b) with increasing noise level, constant and additive model respectively. (c) and (d) with increasing degree of overlap, constant and additive model respectively.

Human CMap Data

We also evaluated our DeBi algorithm on the Connectivity Map v0.2 (CMap) [14]. CMap is a reference collection of gene expression profiles from human cells that have been treated with a variety of drugs comprised of 6100 samples and 22283 genes. Figure 3 (c) summarizes the results of DeBi and QUBIC. The proportion of GO term and TFBS enriched biclusters are much more higher in DeBi compared to QUBIC.

The biclusters discovered by DeBi can be used to find drugs with a common mechanism of action and identify new therapeutics. Moreover, we can observe the effect of drugs on different cell lines. Figure 4 shows parallel coordinate plots of some of the identified biclusters. In parallel coordinate plots, the profile of the conditions that are included in a bicluster are shown as black, the other conditions as gray. This aids to visualize the expression difference between the conditions in a bicluster compared to the rest of the conditions. The bicluster 6, contains up regulated ‘heat shock protein binding’ genes and ‘heat shock protein inhibitors’ such as geldanamycin, alvespimycin, tanespimycin, monorden. Heat shock proteins (Hsps) are overexpressed in a wide range of human cancers and are involved in tumor cell

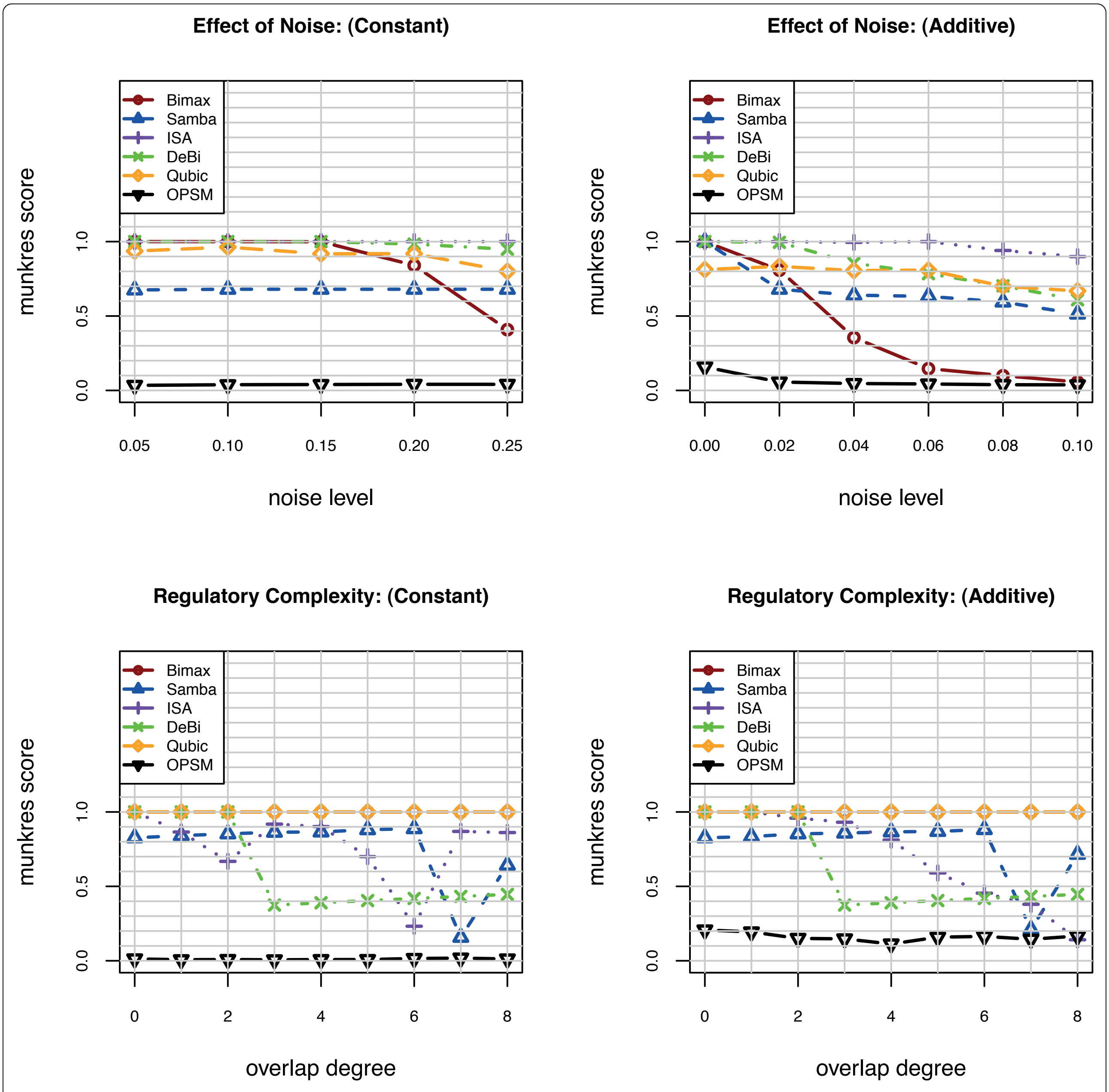

Figure 2 Bicluster consensus score on synthetic data. The synthetic data have been created based on two scenories (a) and (b) with increasing noise level, constant and additive model respectively. (c) and (d) with increasing degree of overlap, constant and additive model respectively.

proliferation [24]. Additionally, genes in the bicluster are enriched with 'P53 binding site', which is known to target heat shock protein binding genes. Bicluster 11, contains up regulated genes enriched with 'cadmium ion binding' GO Term and calcium-binding protein inhibitors, calmidazolium. Bicluster 15, contains up regulated genes enriched with 'transcription corepressor activity' GO Term. Cell lines in this bicluster are all breast cancer. Bicluster 14, contains down regulated genes enriched with 'steroid hormone signalling' GO Term. Additionally, protein interaction networks of the selected biclusters are strikingly connected and they can be found in Figure S4, S5, S6 and S7 in Additional File 1. All the biclusters and the enrichment analysis can be found in Additional file 5.

Human ExpO Data

We applied our DeBi algorithm and QUBIC on Expression Project for Oncology(expO) dataset http://www.intgen.org/. ExpO consists gene expression profiles of 2158

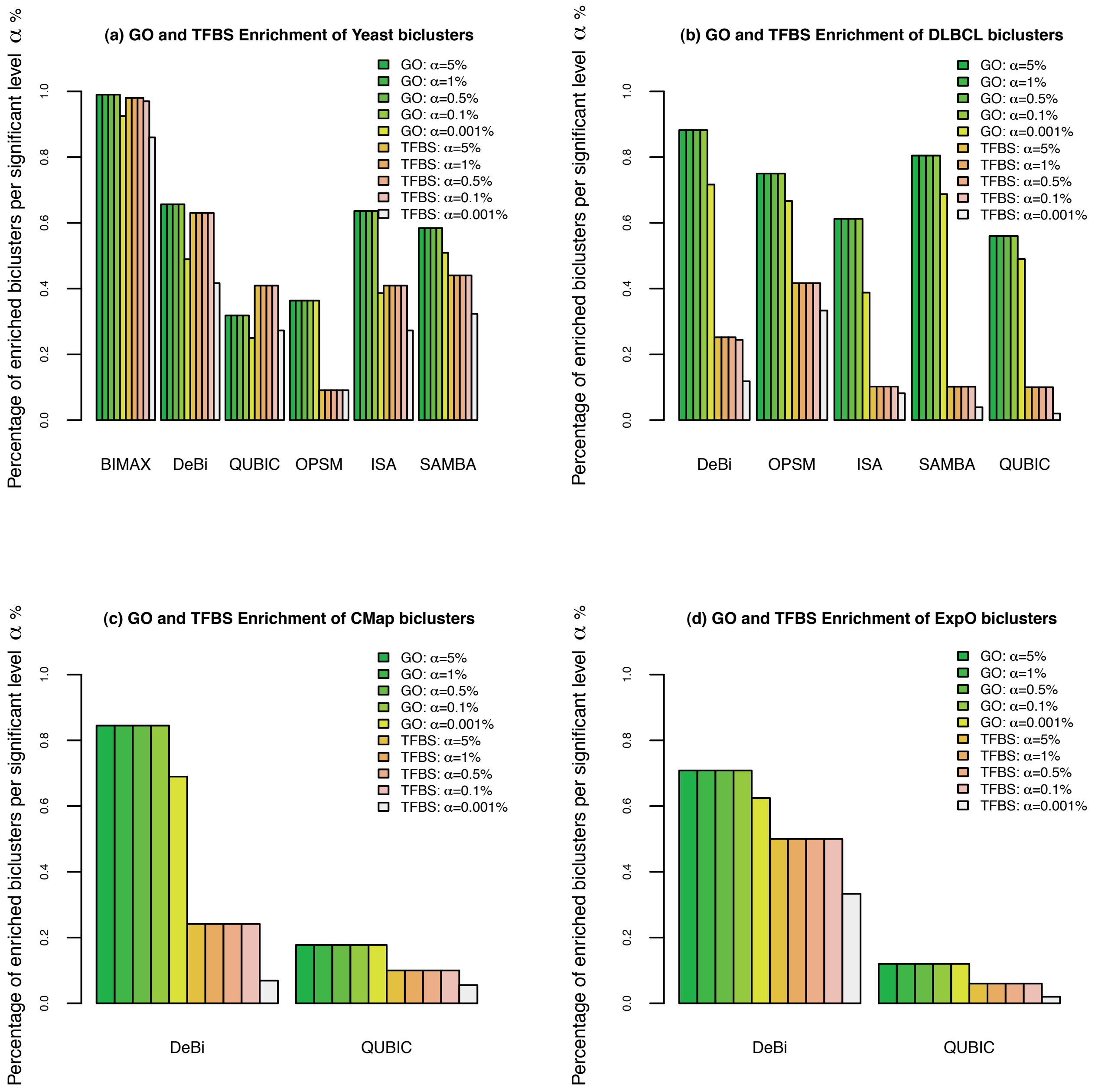

Figure 3 Biological Significance of Yeast, DLBCL, CMap, ExpO Biclusters. GO and TFBS enrichment of yeast, dlbcl, CMap and ExpO biclusters.

human tumor samples coming from diverse tissues with 40223 transcripts.

Figure 3 (d) shows that the proportion of GO term and TFBS enriched biclusters are much more higher in DeBi compared to QUBIC. It illustrates that DeBi performs better than QUBIC in ExpO data. 70% of the DeBi biclusters are enriched with GO Terms with a p-value smaller than 0.05. Moreover biclusters contain tumor samples mostly from similar tissue types. Figure S8 in Additional file 1 shows GO Term enrichment of some of the biclusters. Bicluster 13 contains thyroid tumor samples and genes enriched with 'protein-hormone receptor activity'. Bicluster 3 contains prostate tumor samples and genes enriched with 'tissue kallikrein activity'. Bicluster 22 contains mostly pancreas and colon samples and genes enriched with 'pancreatic elastase activity' GO Term. All the biclusters and the enrichment analysis can be found in Additional file 6.

MSigDB Data

Finally, we applied our algorithm on the manually curated gene sets from the Molecular Signature Database

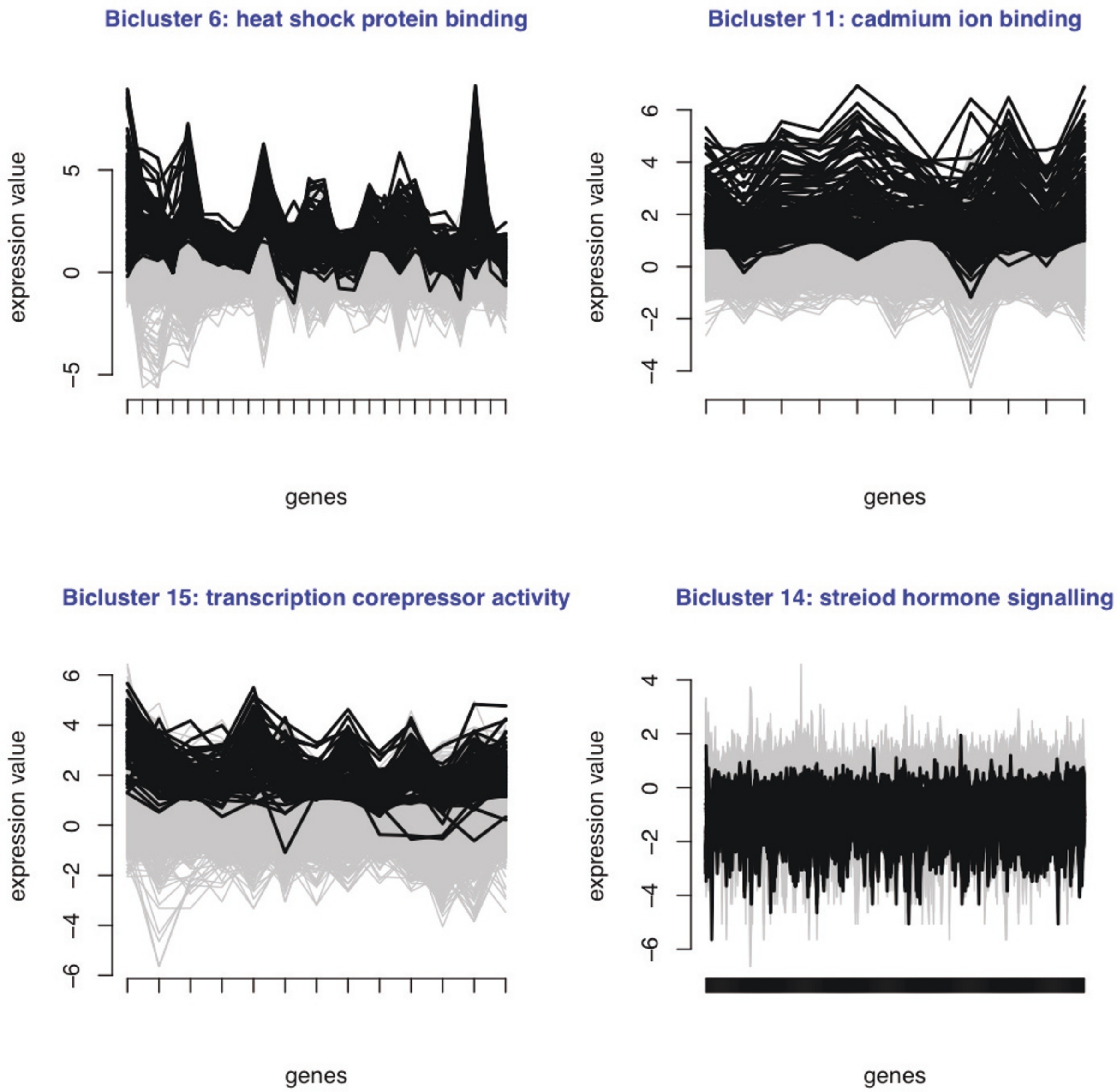

Figure 4 Example CMap biclusters identified using DeBi Algorithm. Parallel coordinate plots of some of the identified CMap biclusters using the DeBi algorithm. In parallel coordinate plots, the profile of the conditions that are included in a bicluster are shown as black, the other conditions as gray.

(MSigDB) C2 category. The C2 category of MSigDB consists of 3272 gene sets in which 2392 gene sets are chemical and genetic pertubations and 880 gene sets are from various pathway databases. The gene sets naturally define a binary matrix where ones indicate the affected gene under certain pertubation/pathway. The binary matrix contains 18205 genes and 3272 samples. This analysis aids us to identify the pathways that are affected by chemical and genetic perturbations. It has not been possible to run QUBIC on this dataset while QUBIC requires a certain amount of overlap between genes.

Figure 5, illustrates all the biclusters using BiVoc algorithm [25]. BiVoc algorithm rearranges rows and conditions in order to represent the biclusters with the minimum space. The output matrix of BiVoc, may have repeated rows and/or columns from the original matrix. In Figure 5, the function of each bicluster is specified based on GO Term enrichment. Bicluster 3, contains the down-regulated gene set from Alzheimer patients and gene set from proteasome pathway. It is known that there is a significant decrease in proteasome activity in Alzheimer patients [26]. Bicluster 3 also contains the

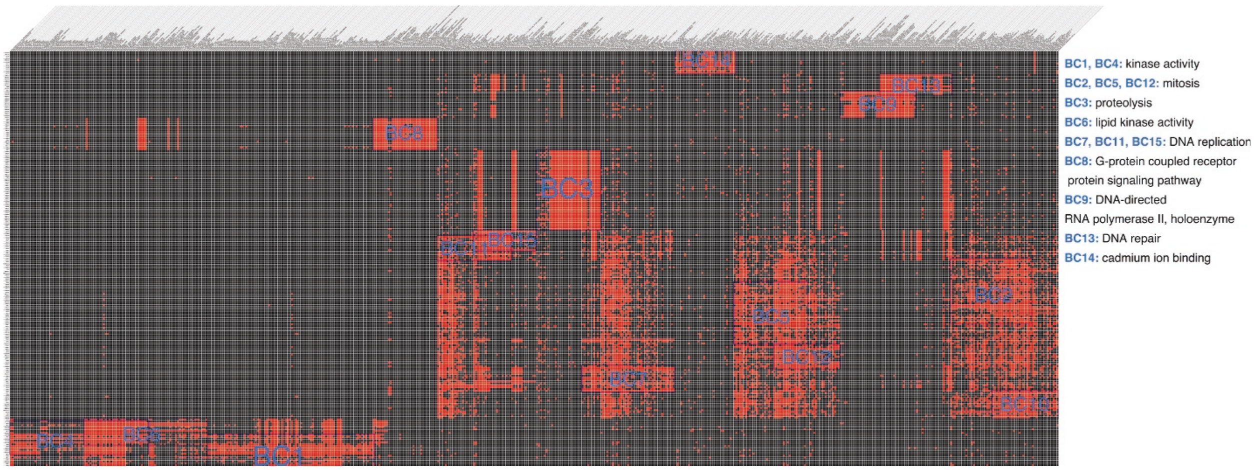

Figure 5 MSigDB biclusters identified using DeBi Algorithm.

up-regulated gene set from pancreatic cancer patients. In previous studies, high activity of ubiquitin-proteasome pathway in pancreatic cancer cell line was detected [27]. Bicluster 8 contains up-regulated gene set from liver cancer patients and gene set from G-protein activation pathway. Dysfunction of G Protein-Coupled Receptor signaling pathways are involved in certain forms of cancer. All the biclusters and the enrichment analysis can be found in Additional file 7.

Running Time

DeBi algorithm is capable of analyzing yeast data(size 6100 × 300) in 6 minutes, ExpO data (size 40223 × 2158) in 12 minutes, MSigDB data (size 18205 × 3272) in 11 minutes, DLBCL data (size 610 × 180) in 11 seconds, CMap data (size 22283 × 6100) in 3 hours 45 minutes. The QUBIC algorithm analyzes CMap data in 2 hours 55 minutes and ExpO data in 3 hours 54 minutes. The running time analysis was done on a 2.13 GHz Intel 2 Dual Core computer with 2GB memory.

Methods

Given an expression matrix E with genes $G = \{g_1, g_2, g_3, ..., g_n\}$ and samples $S = \{s_1, s_2, s_3, ..., s_m\}$ a bicluster is defined as $b = (G', S')$ where $G' \subset G$ is a subset of genes and $S' \subset S$ is a subset of samples. DeBi identifies functionally coherent biclusters $B = \{b_1, b_2, b_3, ..., b_l\}$ in three steps. Below we describe each step in detail. An overview of the DeBi algorithm is shown in Figure 6. The DeBi algorithm is based on a well known data mining approach called Maximal Frequent Item Set [28]. We will refer to this as Maximal Frequent Gene Set, as given by our problem definition. The pseudocode of the algorithm is in Additional file 1.

Preliminaries

The input gene expression data is binarized according to either up or down regulation. Let E^u and E^d denote the up and down regulation binary matrices, respectively. Then the entries e^u_{ij} of E^u are defined as follows:

$$e^u_{ij} = \begin{cases} 1 \text{ if gene } i \text{ is } c \text{ fold up regulated in sample } j \\ 0 \text{ otherwise} \end{cases} \quad (1)$$

and the entries of e^d_{ij} of E^d are defined analogously with a c-fold down-regulation cut-off. The fold change cut-off c will typically be set to 2.

Finding seed biclusters by Maximal Frequent Gene Set Algorithm

The DeBi algorithm, identifies the seed gene sets by iteratively applying the maximal frequent gene set algorithm. We first define the term *support*, which we will later use in the algorithm. The *support* of the gene g_i, $i = 1, ..., n$, is defined as follows:

$$supp(g_i) = \frac{1}{m} \sum_{j=1}^{m} e_{ij} \quad (2)$$

In other words, the *support* is the proportion of samples for which the gene-vector $e_{i.}$ is 1. This is further extended to sets of genes. Let $G'_v = \{g_1, \ldots, g_k\}$ be the v^{th} gene-set. For a set of gene-vectors we define their *phenotype vector* C_v as their element-wise logical AND:

$$C_v = \wedge(e_{1.}, \ldots, e_{k.}) \quad (3)$$

The *support* of the gene set is then defined as the fraction of samples for which the phenotype vector is 1.

A gene set G'_v is (c_1, c_2) - *frequent* iff its support *supp* (G'_v) is larger than c_1 and the cardinality $|G'_v|$ above c_2.

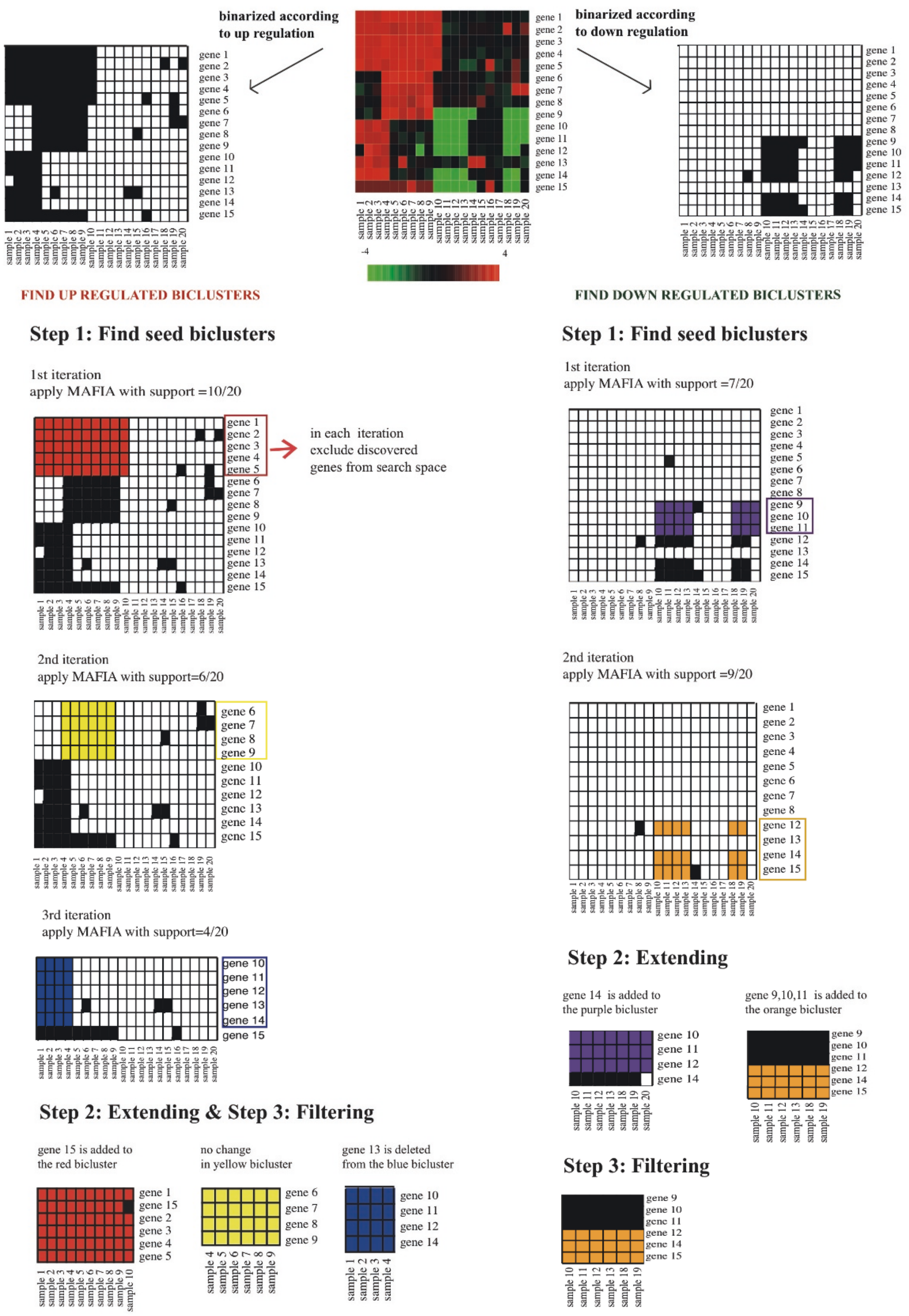
binarized according to up regulation
binarized according to down regulation
gene 1 … gene 15
sample 1 … sample 20
-4
4
FIND UP REGULATED BICLUSTERS
FIND DOWN REGULATED BICLUSTERS
Step 1: Find seed biclusters
1st iteration
apply MAFIA with support =10/20
in each iteration
exclude discovered
genes from search space
2nd iteration
apply MAFIA with support=6/20
3rd iteration
apply MAFIA with support=4/20
Step 2: Extending & Step 3: Filtering
gene 15 is added to the red bicluster
no change in yellow bicluster
gene 13 is deleted from the blue bicluster
gene 1
gene 15
gene 2
gene 3
gene 4
gene 5
gene 6
gene 7
gene 8
gene 9
gene 10
gene 11
gene 12
gene 14
Step 1: Find seed biclusters
1st iteration
apply MAFIA with support =7/20
2nd iteration
apply MAFIA with support =9/20
Step 2: Extending
gene 14 is added to the purple bicluster
gene 9,10,11 is added to the orange bicluster
Step 3: Filtering

Figure 6 Illustration of DeBi algorithm. The algorithm is ran on two different binarized datasets. One is the binarized data based on up regulation and the other is the binarized data based on down regulation. In Step 1, seed biclusters identified within each support value going from high to low. For the binarized data based on up regulation, in the 1st iteration, red gene set with support value 10/20 is detected and excluded from the search space. Similarly, in the second and third iterations yellow and blue clusters with support values, respectively 6/20 and 4/20, are found. In Step 2, seed gene sets are improved based on genes' association strength. Gene 15 is added to the red bicluster because the p-value returned by the Fisher exact test is smaller than α and gene 13 is deleted because the p-value returned by the Fisher exact test is higher than α. None of the discovered biclusters have an overlap of the gene × sample area of more than 50%.

When c_1 and c_2 are not in focus, we will simply speak of a frequent gene set. A gene set is *maximally frequent* iff it is frequent and no superset of it is frequent.

The simplest method for detecting maximally frequent gene sets is a brute force approach in which each possible subset of G ={g_1, g_2, g_3,..., g_n} is a candidate frequent set. To find the frequent sets we count the support of each candidate set. The MAFIA algorithm is an efficient implementation for finding maximally frequent sets with support above a given threshold [28]. The search strategy of MAFIA uses a depth-first traversal of the gene set lattice with effective pruning techniques. It avoids exhaustive enumeration of all candidate gene sets by a monotonicity principle. The monotonicity principle states that every subset of a frequent itemset is frequent. It prunes the candidates which have an infrequent subpattern using this property.

In the first step of the DeBi algorithm, MAFIA is iteratively applied to the binary matrix successively reducing the support threshold. Initially, MAFIA is applied to the full binary matrix E^u (E^d) with support value $(c_1)_0$ equal to support value of the gene with the highest support. In iteration k, MAFIA is applied with support value threshold of $(c_1)_k = (c_1)_{k-1} - \frac{1}{m}$. The identified maximally frequent sets are added to the set of seed gene sets B and the genes in B are deleted from the binary matrix E^u (E^d). In each iteration MAFIA is applied to the modified matrix $E^{u'}(E^{d'})$. The process is repeated until a user defined *minumum support* parameter is reached.

Extending and filtering the biclusters

In the second step of DeBi, the identified seed gene sets $G'_1 = \{G'_1, G'_2 \ldots G'_l\}$ are extended using a local search. For each bicluster $B_v = (G'_v, S'_v)$, $v = 1,...,l$, we have the binary phenotype vector $C_v = \wedge(e_1,...,e_k) = (C_{v1},...,C_{vm})$. The entries of C_v indicate the indices of the bicluster samples. If $C_{vj} = 1 \Rightarrow s_j \in S'_v$, $j = 1,...,m$, i.e. that the sample s_j belongs to the bicluster b_v. The gene g_i, $i = 1,..., n$, is an element of gene set G'_v if $e_{i\cdot}$ is associated with C_v. We evaluate the association strength between the phenotype vector of a bicluster and another gene using Fisher's exact test on a 2 × 2 contingency table. The cells of the contingency table count how often the four possibilities of the phenotype vector containing a 1 or a 0 and the gene-vector containing a 1 or a 0 occur. The Fisher's exact test then tests for independence in the contingency table and thus among the two vectors.

A gene g_i, $i = 1,..., n$ is added, to the gene set G'_v if the pvalue p_{g_i} returned by the Fisher exact test is lower than the parameter α. It gets deleted from b_v if the probability is higher than α and added to b_v if the probability is smaller than α. For this procedure the association probability p_{g_i} with the bicluster needs to be calculated for each gene. However, we reduce the computational effort using the monotonicity property of the hypergeometric distribution. We precompute cut-off values on the contingency table entries that yield a p-value just higher than α. Let $\sigma_{1,\ IN}$ and $\sigma_{1,\ OUT}$ denote the number of 1's a gene-vector has in the bicluster samples and the number of 1's a gene-vector has outside the bicluster samples, respectively. We find the minimal $\sigma_{1,\ IN}$ and maximal $\sigma_{1,\ OUT}$ at this border. Then, we apply Fisher's exact test only to those genes which have $\sigma_{1,\ IN} > min\sigma_{1,\ IN}$ and $\sigma_{1,\ OUT} < max\sigma_{1,\ OUT}$.

In the last step we turn to the sometimes very complicated overlap structure among biclusters. The goal is to filter the set of biclusters such that the remaining ones are large and overlap only little. The size of a bicluster is defined as the number of genes times the number of samples in the bicluster, $|G'_v| \times |S'_v|$. Two biclusters overlap when they share common samples and genes. The size of the overlap is the product of the number of common samples and common genes. To filter out biclusters that are largely contained in a bigger bicluster, we start with the largest bicluster and compare it to the other biclusters. Those biclusters for which the overlap to the largest one exceeds L% (typically 50%) of the size of the smaller one are deleted. This is then repeated starting with the remaining second-largest bicluster and so on.

Choosing the optimum alpha parameter

To formulate an optimality criterion for α one requires an inherent measure of the quality of a set of biclusters. To this end, for a bicluster v, we define its score I_v as the negative sum of the log p-values of the included genes, where the individual p_g is the p-value from the Fisher exact test:

$$I_v = -\sum_{g \varepsilon G'_v} (\log p_g) \qquad (4)$$

However, this bicluster score I_v depends on the size (number of genes × number of conditions) of the bicluster and in order to make it comparable between biclusters one needs to correct for the size. We compute the expected bicluster score through a randomization procedure. A large number, say 500, random phenotype vectors having the same number of 1s as the bicluster has conditions is generated. For these random phenotype vectors a Fisher exact test p-value with respect to each gene in the bicluster is computed. One obtains a random I_v score by adding log p-values over the genes of the bicluster. The mean of these random bicluster scores is the desired estimator. Finally, a normalized NI_v score is definded by dividing I_v by this estimated mean and the total biclustering score CS is defined as the sum of NI_ν normalized scores of all discovered biclusters $CS = \sum_{\nu \varepsilon\ I} (NI_\nu)$. This score serves to distinguish between different choices of α. The program is run under $\alpha = \{10^{-2}, 10^{-3}, \ldots, 10^{-100}\}$ and we choose the α that maximizes CS.

Discussion

We have proposed a novel fast biclustering algorithm especially for analyzing large datasets. Our algorithm aims to find biclusters where each gene in a bicluster should be highly or lowly expressed over all the bicluster samples compared to the rest of the samples. Unlike other algorithms, it is not required to define the number of biclusters a priori. We have compared our method with other biclustering algorithms using synthetic data and biological data. We have shown that the DeBi algorithm provides biologically significant biclusters using GO term and TFBS enrichment. We have also showed the computational efficiency of our algorithm. It is shown that it is a useful and powerful tool in analyzing large datasets.

In spite of efforts by many authors, comparing the performance of biclustering algorithms is still a challenge [29]. Smaller biclusters have a higher chance to yield a coherent GO annotation, while larger biclusters would, of course, be more interesting to observe. Our α threshold influences this behavior. The optimized α threshold is smaller for larger number of samples which limits the number of genes that get accepted into a bicluster.

The binarization of the input data in order to obtain a boolean matrix is another key decision in our approach. In this we go along with many other authors and we think that it helps in applying biclustering to gene expression data coming from different labs or platforms. The hope is that our method will further contribute to establishing biclustering as a general purpose tool for data analysis in functional genomics.

Implementation

The DeBi code is written in c++ programming language for UNIX environment. The MAFIA algorithm c++ code is used for calculating the maximally frequent item sets. The DeBi algorithm is freely available at http://www.molgen.mpg.de/~serin/debi/main.html.

Additional material

Additional file 1: Description of selected biclustering algorithms, description of MAFIA algorithm, protein protein interaction networks.

Additional file 2: DeBi results on synthetic data.

Additional file 3: DeBi, BIMAX, ISA, OPSM, SAMBA and QUBIC biclustering results and GO term, TFBS enrichment analysis of the genes and conditions in biclusters on yeast data.

Additional file 4: DeBi, ISA, OPSM, SAMBA, QUBIC biclustering results and GO term, TFBS enrichment analysis of the genes on DLBCL data.

Additional file 5: DeBi and QUBIC biclustering results and GO term and TFBS enrichment analysis of the biclusters on CMap data.

Additional file 6: DeBi and QUBIC biclustering results and GO term and TFBS enrichment analysis of the biclusters on ExpO data.

Additional file 7: DeBi biclustering results and GO term and TFBS enrichment analysis of the biclusters on MSigDB data.

Authors' contributions

AS developed and implemented the algorithm. AS drafted the versions of the manuscript. MV supervised the work and development of ideas. MV contributed with discussion of the draft versions and critical review. Both authors have read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Received: 14 February 2011 Accepted: 23 June 2011
Published: 23 June 2011

References

1. Andreopoulos B, An A, Wang X, Schroeder M: A roadmap of clustering algorithms: finding a match for a biomedical application. *Brief Bioinformatics* 2008, **10(3)**:297-314.
2. Sokal RR, Michener CD: A statistical method for evaluating systematic relationships. *University of Kansas Scientific Bulletin* 1958, **28**:1409-1438.
3. Hartigan JA, Wong MA: Algorithm AS 136: A k-means clustering algorithm. *Applied Statistics* 1979, **28**:100-108.
4. Hartigan JA: Direct Clustering of a Data Matrix. *Journal of the American Statistical Association* 1972, **67(337)**:123-129.
5. Cheng Y, Church GM: Biclustering of expression data. *Proc Int Conf Intell Syst Mol Biol* 2000, **8**:93-103.
6. Ben-Dor A, Chor B, Karp R, Yakhini Z: Discovering local structure in gene expression data: the order-preserving submatrix problem. *J Comput Biol* 2003, **10(3-4)**:373-384.
7. Bergmann S, Ihmels J, Barkai N: Iterative signature algorithm for the analysis of large-scale gene expression data. *Phys Rev E Stat Nonlin Soft Matter Phys* 2003, **67(3 Pt 1)**:031902.
8. Murali TM, Kasif S: Extracting conserved gene expression motifs from gene expression data. *Pac Symp Biocomput* 2003, 77-88.
9. Tanay A, Sharan R, Shamir R: Discovering statistically significant biclusters in gene expression data. *Bioinformatics* 2002, **18(Suppl 1)**:S136-S144.
10. Prelic A, Bleuler S, Zimmermann P, Wille A, Buehlmann P, Gruissem W, Hennig L, Thiele L, Zitzler E: A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics* 2006, **22(9)**:1122-1129.
11. Li G, Ma Q, Tang H, Paterson AH, Xu Y: QUBIC: a qualitative biclustering algorithm for analyses of gene expression data. *Nucl Acids Res* 2009, **37(15)**:e101[http://nar.oxfordjournals.org/cgi/content/abstract/37/15/e101].
12. Madeira SC, Oliveira AL: Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Trans Comput Biol Bioinform* 2004, **1**:24-45.
13. Hughes TR, Marton MJ, Jones AR, Roberts CJ, Stoughton R, Armour CD, Bennett HA, Coffey E, Dai H, He YD, Kidd MJ, King AM, Meyer MR, Slade D, Lum PY, Stepaniants SB, Shoemaker DD, Gachotte D, Chakraburtty K, Simon J, Bard M, Friend SH: Functional discovery via a compendium of expression profiles. *Cell* 2000, **102**:109-126.
14. Lamb J: The Connectivity Map: a new tool for biomedical research. *Nature reviews Cancer* 2007, **7**:54-60.
15. Rosenwald A, Wright G, Chan WC, Connors JM, Campo E, Fisher RI, Gascoyne RD, Muller-Hermelink HK, Smeland EB, Giltnane JM, Hurt EM, Zhao H, Averett L, Yang L, Wilson WH, Jaffe ES, Simon R, Klausner RD, Powell J, Duffey PL, Longo DL, Greiner TC, Weisenburger DD, Sanger WG,

Dave BJ, Lynch JC, Vose J, Armitage JO, Montserrat E, Lopez-Guillermo A, *et al*: **The Use of Molecular Profiling to Predict Survival after Chemotherapy for Diffuse Large-B-Cell Lymphoma.** *New England Journal of Medicine* 2002, **346(25)**:1937-1947[http://www.nejm.org/doi/full/10.1056/NEJMoa012914].

16. Basehoar AD, Zanton SJ, Pugh BF: **Identification and distinct regulation of yeast TATA box-containing genes.** *Cell* 2004, **116(5)**:699-709[http://www.cell.com/retrieve/pii/S0092867404002053].
17. MacIsaac KD, Wang T, Gordon DB, Gifford DK, Stormo GD, Fraenkel E: **An improved map of conserved regulatory sites for Saccharomyces cerevisiae.** *BMC Bioinformatics* 2006, **7**:113[http://www.biomedcentral.com/1471-2105/7/113].
18. Harbison CT, Gordon DB, Lee TI, Rinaldi NJ, MacIsaac KD, Danford TW, Hannett NM, Tagne JB, Reynolds DB, Yoo J, Jennings EG, Zeitlinger J, Pokholok DK, Kellis M, Rolfe PA, Takusagawa KT, Lander ES, Gifford DK, Fraenkel E, Young RA: **Transcriptional regulatory code of a eukaryotic genome.** *Nature* 2004, **431(7004)**:99-104[http://www.nature.com/nature/journal/v431/n7004/full/nature02800.html].
19. Barkow S, Bleuler S, Prelic A, Zimmermann P, Zitzler E: **BicAT: a biclustering analysis toolbox.** *Bioinformatics* 2006, **22(10)**:1282-1283.
20. Shamir R, Maron-Katz A, Tanay A, Linhart C, Steinfeld I, Sharan R, Shiloh Y, Elkon R: **EXPANDER-an integrative program suite for microarray data analysis.** *BMC Bioinformatics* 2005, **6**:232.
21. Hochreiter S, Bodenhofer U, Heusel M, Mayr A, Mitterecker A, Kasim A, Khamiakova T, Sanden SV, Lin D, Talloen W, Bijnens L, Göhlmann HWH, Shkedy Z, Clevert DA: **FABIA: factor analysis for bicluster acquisition.** *Bioinformatics* 2010, **26(12)**:1520-7[http://bioinformatics.oxfordjournals.org/content/26/12/1520.long].
22. Hoshida Y, Brunet JP, Tamayo P, Golub TR, Mesirov JP: **Subclass Mapping: Identifying Common Subtypes in Independent Disease Data Sets.** *PLoS ONE* 2007, **2(11)**:e1195[http://dx.plos.org/10.1371%2Fjournal.pone.0001195].
23. Jensen LJ, Kuhn M, Stark M, Chaffron S, Creevey C, Muller J, Doerks T, Julien P, Roth A, Simonovic M, Bork P, von Mering C: **STRING 8-a global view on proteins and their functional interactions in 630 organisms.** *Nucleic Acids Res* 2009, , **37 Database:** D412-6.
24. Ciocca DR, Calderwood SK: **Heat shock proteins in cancer: diagnostic, prognostic, predictive, and treatment implications.** *Cell Stress Chaperones* 2005, **10(2)**:86-103.
25. Grothaus GA, Mufti A, Murali TM: **Automatic layout and visualization of biclusters.** *Algorithms for molecular biology : AMB* 2006, **1**:15.
26. Keller JN, Hanni KB, Markesbery WR: **Impaired proteasome function in Alzheimer's disease.** *J Neurochem* 2000, **75**:436-9[http://onlinelibrary.wiley.com/doi/10.1046/j.1471-4159.2000.0750436.x/abstract].
27. Ni XG, Zhou L, Wang GQ, Liu SM, Bai XF, Liu F, Peppelenbosch MP, Zhao P: **The ubiquitin-proteasome pathway mediates gelsolin protein downregulation in pancreatic cancer.** *Mol Med* 2008, **14(9-10)**:582--9.
28. Burdick D, Calimlim M, Gehrke J: **MAFIA: A Maximal Frequent Itemset Algorithm for Transactional Databases.** *Data Engineering, International Conference on* 2001, **0**:0443.
29. Chia BKH, Karuturi RKM: **Differential co-expression framework to quantify goodness of biclusters and compare biclustering algorithms.** *Algorithms for molecular biology : AMB* 2010, **5**:23.

doi:10.1186/1748-7188-6-18
Cite this article as: Serin and Vingron: **DeBi: Discovering Differentially Expressed Biclusters using a Frequent Itemset Approach.** *Algorithms for Molecular Biology* 2011 **6**:18.