BIOINFORMATICS

Vol. 1 no. 1 2002
Pages 1–9

Discovering Statistically Significant Biclusters in Gene Expression Data

Amos Tanay *, *Roded Sharan* * *and Ron Shamir*

School Of Computer Science, Tel-Aviv University, Ramat-Aviv, Tel-Aviv, 69978, Israel

ABSTRACT

In gene expression data, a bicluster is a subset of the genes exhibiting consistent patterns over a subset of the conditions. We propose a new method to detect significant biclusters in large expression datasets. Our approach is graph theoretic coupled with statistical modeling of the data. Under plausible assumptions, our algorithm is polynomial and is guaranteed to find the most significant biclusters. We tested our method on a collection of yeast expression profiles and on a human cancer dataset. Cross validation results show high specificity in assigning function to genes based on their biclusters, and we are able to annotate in this way 196 uncharacterized yeast genes. We also demonstrate how the biclusters lead to detecting new concrete biological associations. In cancer data we are able to detect and relate finer tissue types than was previously possible. We also show that the method outperforms the biclustering algorithm of Cheng and Church (2000).

Contact: {amos,roded,rshamir}@tau.ac.il.

Availability: www.cs.tau.ac.il/∼rshamir/biclust.html.

INTRODUCTION

DNA microarray technology has recently attained a central role in biological and biomedical research. It enables monitoring the transcription levels of many thousands of genes, while the cell undergoes specific conditions or processes. The applications of such technology range from genes functional annotation and genetic networks reconstruction to diagnosis of disease conditions and characterizing effects of medical treatment.

A key step in the analysis of gene expression data is the identification of groups of genes that exhibit similar expression patterns. Clustering gene expression data into homogeneous groups was shown to be instrumental in functional annotation, tissue classification, motif identification and more (for a review see Sharan *et al*. (2002)). However, clustering has its limitations. First, the clustering process builds on the assumption that related genes behave similarly across all measured conditions. This assumption is reasonable when the dataset contains few conditions from a single, focused experiment, but does not hold for larger datasets containing hundreds of heterogeneous conditions from many experiments. Second, 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.

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

Biclustering was introduced in the seventies (Hartigan, 1975). Cheng and Church (2000) were the first to apply it to gene expression data. They defined a bicluster as a uniform submatrix (one having a low mean squared residue score), and used a greedy approach to find biclusters. Getz et al. (2000) devised a coupled two-way iterative clustering algorithm to identify biclusters. Lazzeroni and Owen (2000) introduced the notion of a plaid model, which describes the input matrix as a linear function of variables corresponding to its biclusters. They showed how to estimate a model using an iterative maximization process. Ben-Dor et al. (2002) defined a bicluster as an order preserving submatrix, or equivalently, a group of genes whose expression levels induce some linear order across a subset of the conditions. A greedy heuristic search procedure is employed to detect such biclusters. The work of Segal et al. (2001) described rich probabilistic models for studying relations between expression, regulatory motifs and gene annotations. Its outcome can be interpreted as a collection of disjoint biclusters generated in a supervised manner.

In this paper we develop a novel approach to biclustering, which combines graph theoretic and statistical considerations. The intuitive notion of a bicluster is a subset of genes that exhibit similar expression patterns over a subset of conditions. Following this intuition we define a bicluster as a subset of genes that *jointly respond* across a subset of conditions, where a gene is termed responding in some condition if its expression level changes significantly at

*These authors contributed equally to this work.

that condition w.r.t. its normal level.

We model the input expression data as a bipartite graph whose two parts correspond to conditions and genes, respectively, with edges for significant expression changes. We present two statistical models of the resulting graph. We show how to assign weights to the vertex pairs of the bipartite graph according to each model, so that heavy subgraphs correspond to significant biclusters. Using the first, simpler statistical model, we show how to compute a tight upper-bound on the probability of an observed bicluster. For a more detailed model that takes into account genes and conditions variability, we show how to assign weights so that a maximum weight subgraph corresponds to a maximum likelihood bicluster. We also show how to approximate the p-value of observing a subgraph with weight exceeding a given threshold.

Discovering the most significant biclusters in the data reduces under these weighting schemes to finding the heaviest subgraphs in the model bipartite graph. As explained below, unrestricted versions of this problem are computationally hard, so we assume a degree restriction on the graphs and limit the discussion to graphs in which the gene vertices have degrees not exceeding a fixed constant d. This assumption has several justifications: First, high-degree genes are more likely to participate in heavy subgraphs and, thus, contribute little to the significance of a bicluster containing them. Second, high degree genes are involved in many processes and do not manifest a focused specific effect. In the datasets we studied, filtering high-degree genes resulted in a modest reduction (20% on average) in the number of genes. Still, our practical implementation considers high-degree genes as well, in a heuristic manner.

Requiring a bicluster to be a complete subgraph gives rise to the problem of finding a maximum edge biclique. This problem is NP-complete for weighted bipartite graphs (cf. Hochbaum (1998)). We present a polynomial algorithm for the weighted problem under the degree restriction.

To accommodate noisy data, we search for subgraphs that are not necessarily complete. Let the weight of a subgraph be the sum of the weights of gene-condition pairs in it. We assume that edges are assigned positive weights and non-edges are assigned negative weights. We show that the problem of finding a maximum weight subgraph is NP-complete in this case. In contrast, we give a polynomial time algorithm for the problem on graphs under the degree restriction.

One can argue that in some situations we lose information by just looking for biclusters that manifest changes, without considering if the change was an increase or a decrease in expression (henceforth, the unsigned problem). We therefore study the problem of finding *consistent* biclusters, in which every two conditions must always have the same effect or always have the opposite effect on each of the genes. We show how to solve this problem by a polynomial reduction to the unsigned problem. Hence, our polynomial algorithms apply to consistent biclusters as well, enabling the detection of connections between genes with either similar or complementary patterns.

We implemented a practical heuristic, called SAMBA (Statistical-Algorithmic Method for Bicluster Analysis), which follows the approach of the theoretical algorithm, and is able to analyze large datasets within minutes. We applied our algorithm to a broad class of gene expression datasets, including yeast and human clinical data. In tests on human lymphoma data, when measuring the solution p-value w.r.t. a known tissue classification, our solutions are superior to those of Cheng & Church (2000). Our biclusters also enable differentiating fine tissue types, e.g., germinal center, from DLBCL tissues, although these tissue types were grouped together using standard clustering techniques. We also show the utility of our method to functional annotation, based on a compiled dataset of some 515 yeast expression profiles. Using GO annotations of the yeast genes, we can annotate unknown genes that belong to a bicluster containing many genes with the same known annotation. Our cross validation test proves the soundness of this approach, yielding 81.5% annotation specificity, and we are able to annotate 196 unknown yeast genes in this manner. For example, we discovered a link between a group of unknown subtelomeric Y' genes and DNA repair genes, which was recently discovered experimentally.

The paper is organized as follows: We start by presenting our statistical models for gene expressions data. We then present a combinatorial algorithm for finding maximum weight subgraphs of a bipartite graph, and generalize it to handle consistent biclusters. Finally, we describe our practical implementation and our results on several biological datasets. For lack of space some details are omitted.

STATISTICAL DATA MODELING

Given an input gene expression dataset we form a bipartite graph $G = (U, V, E)$ (see (Golumbic, 1980) for basic graph-theoretic definitions and Figure 1 for an example). In this graph, U is the set of conditions, V is the set of genes, and $(u, v) \in E$ iff v responds in condition u, that is, if the expression level of v changes significantly in u (see the SAMBA Algorithm Section for details). Later we shall refine our graph to include the direction of expression change (up or down regulation). A bicluster corresponds to a subgraph $H = (U', V', E')$ of G, and represents a subset V' of genes that are co-regulated under a subset of conditions U' (see Figure 1). The *weight* of a subgraph (or bicluster) is the sum of the weights of gene-condition pairs in it, including edges and non-edges.

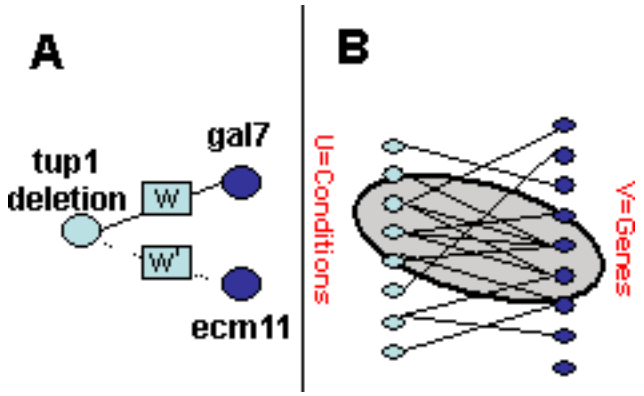


Fig. 1. Gene expression data is modeled using a bipartite graph whose two sides correspond to the set of conditions U and the set of genes V. An edge (u, v) indicates the response of gene v in condition u. A statistical model assigns weights to the edges and non-edges of the graph. A) Part of the graph showing the condition "tup1 deletion" and its effect on the genes "gal7" (response) and "ecm11" (no response). B) A heavy subgraph (shaded) representing a significant bicluster.

In the following we develop statistical models for our bipartite graph representation of expression data. Using these models we derive scoring schemes for assessing the significance of an observed subgraph (corresponding to a bicluster). We shall develop additive scores that can be decomposed across the edges and non edges of the graph. In other words, we shall assign weights to the edges and non-edges of the graph, such that the weight of a subgraph will correspond to its statistical significance. This will allow us to reduce the biclustering problem to that of finding heavy subgraphs in a bipartite graph.

A Simple Model

Let $H = (U', V', E')$ be a subgraph of G. Denote $|U'| = m', |V'| = n'$. Let $p = \frac{|E|}{|U||V|}$, and let $k' = |E'|$. Our first model assumes that edges occur independently and equiprobably with density p. Denote by $BT(k, p, n)$ the binomial tail, i.e., the probability of observing k or more successes in n trials, where each success occurs independently with probability p. Then the probability of observing a graph at least as dense as H according to this model is $p(H) = BT(k', p, n'm')$.

Our goal is to find a subgraph H with lowest $p(H)$. By bounding the terms of the binomial tail using the first one, assuming that $p < 1/2$, we obtain the following upper bound for $p(H)$: $p^*(H) = 2^{n'm'} p^{k'} (1 - p)^{n'm' - k'}$. Seeking a subgraph H minimizing $\log p^*(H)$ is equivalent to finding a maximum weight subgraph of G where each edge has positive weight $(-1 - \log p)$ and each non-edge has negative weight $(-1 - \log(1 - p))$.

Note that $p(H)$ provides a reasonable approximation only if $n'm' \ll nm$, as the calculation of $p(H)$ ignores the total number of edges in G. As we constrain the degree to d, this condition holds.

A Refined Model

We next develop a refined null model that takes into account the variability of the degrees in G, i.e., it incorporates the characteristic behavior of each specific condition and gene.

Let $H = (U', V', E')$ be a subgraph of G and denote $\overline{E'} = (U' \times V') \setminus E'$. For a vertex $w \in U' \cup V'$ let d_w denote its degree in G. Our null model assumes that the occurrence of each edge (u, v) is an independent Bernoulli variable with parameter $p_{u,v}$. The probability $p_{u,v}$ is the fraction of bipartite graphs with degree sequence identical to G that contain the edge (u, v). In practice we estimate $p_{u,v}$ using a Monte-Carlo process. The probability of observing H is thus $p(H) = (\prod_{(u,v)\in E'} p_{u,v}) \cdot (\prod_{(u,v)\in \overline{E'}} (1 - p_{u,v}))$. However, we cannot simply compare subgraphs according to this probability, since it improves (decreases) as the size of H increases.

To overcome this problem, we chose to use a likelihood ratio to capture the significance of biclusters. Our null model is as stated above. For the alternative model we assume that each edge of a bicluster occurs with constant probability $p_c > \max_{(u,v)\in U\times V} p_{u,v}$. The estimation of p_c is described in the SAMBA Algorithm Section.This model reflects our belief that biclusters represent approximately uniform relations between their elements. The log likelihood ratio for H is therefore:

$$\log L(H) = \sum_{(u,v)\in E'} \log \frac{p_c}{p_{u,v}} + \sum_{(u,v)\in \overline{E'}} \log \frac{1 - p_c}{1 - p_{u,v}}$$

Setting the weight of each edge (u, v) to $\log \frac{p_c}{p_{u,v}} > 0$ and the weight of each non-edge (u, v) to $\log \frac{1-p_c}{1-p_{u,v}} < 0$, we conclude that the score of H is simply its weight.

We note that the statistical model is more involved when taking into account the direction of expression change for each edge. Nevertheless, a likelihood score can be computed in essentially the same way as for the unsigned case.

COMBINATORIAL BICLUSTERING

In the previous section we have given an additive scoring scheme assigning weights to edges and non-edges of a model bipartite graph. Discovering the most significant biclusters in the data reduces under this scoring scheme to finding the heaviest subgraphs in the bipartite graph. We now give a polynomial algorithm to solve this problem when the degree of every gene vertex is bounded.

Maximum Bounded Biclique

We start by describing an $O(|V|2^d)$-time algorithm to find a maximum weight biclique in a bipartite graph whose gene vertices have d-bounded degree. This algorithm will be a key component in our more involved algorithms that follow.

Let $G = (U, V, E)$ be a bipartite graph. We say that G has *d-bounded gene side*, if every $v \in V$ has degree at most d. Let $w : U \times V \to \mathcal{R}$ be a weight function. For a pair of subsets $U' \subseteq U, V' \subseteq V$ we denote by $w(U', V')$ the weight of the subgraph induced on $U' \cup V'$, i.e., $w(U', V') = \sum_{u \in U', v \in V'} w((u, v))$. The *neighborhood* of a vertex v, denoted $N(v)$, is the set of vertices adjacent to v in G. We denote $n = |V|$ throughout.

PROBLEM 1 (MAXIMUM BOUNDED BICLIQUE). *Given a weighted bipartite graph G with d-bounded gene side, find a maximum weight complete subgraph of G.*

THEOREM 1. *The maximum bounded biclique problem can be solved in $O(n2^d)$ time and space.*

Proof: Observe that a maximum bounded biclique $H^* = (U^*, V^*, E^*)$ in G must have $|U^*| \leq d$. Figure 2 describes a hash-table based algorithm that for each vertex $v \in V$ scans all $O(2^d)$ subsets of its neighbors, thereby identifying the heaviest biclique. Each hash entry corresponds to a subset of conditions and records the total weight of edges from adjacent gene vertices. The iteration over subsets of $N(v)$ is done by repeatedly changing the current subset S by adding or removing a single element, updating $w(S, \{v\})$ in constant time. Hence, the algorithm spends $O(n2^d)$ time on the hashing and finding U_{best}. Computing V_{best} can be done in $O(nd)$ time, so the total running time is $O(n2^d)$. The space complexity is $O(n2^d)$ due to the hash-table. ■

MaxBoundBiClique(U, V, E, d):
Initialize a hash table $weight$; $weight_{best} \leftarrow 0$
For all $v \in V$ do
  For all $S \subseteq N(v)$ do
    $weight[S] \leftarrow weight[S] + \max\{0, w(S, \{v\})\}$
    If ($weight[S] > weight_{best}$)
      $U_{best} \leftarrow S$
      $weight_{best} \leftarrow weight[S]$
Compute $V_{best} = \cap_{u \in U_{best}} N(u)$
Output (U_{best}, V_{best})

Fig. 2. An algorithm for the maximum bounded biclique problem.

Note that the algorithm can be adapted to give the k condition subsets that induce solutions of highest weight in $O(n2^d \log k)$ time using a priority queue (heap) data structure.

Finding Heavy Subgraphs

We now look for heavy subgraphs which are not necessarily complete. We start by giving weight 1 for an edge and weight -1 for a non-edge. Formally, given a bipartite graph $G = (U, V, E)$ define a weight function $w : U \times V \to \{-1, 1\}$ such that $w((u, v)) = 1$ for $(u, v) \in E$, and $w((u, v)) = -1$ for $(u, v) \in (U \times V) \setminus E$. Consider the following problem:

PROBLEM 2. *(Maximum Bounded Bipartite Subgraph) Given a bipartite graph G with d-bounded gene side, find a maximum weight subgraph of G.*

LEMMA 2. *Let $H^* = (U^*, V^*, E^*)$ be a maximum weight subgraph of G. Then every vertex in H^* is connected to at least half the vertices on the other side of H^*.*

Proof: Follows from the choice of weights, since if a vertex $v \in V^*$ has less than $\lceil |U^*|/2 \rceil$ neighbors, then removing v from H^* will result in a heavier subgraph. The proof for $u \in U^*$ is symmetric. ■

COROLLARY 3. *A maximum weight subgraph of G has at most $2d$ vertices from U.*

LEMMA 4. *Let $H^* = (U^*, V^*, E^*)$ be a maximum weight subgraph of G. For each set $X \subseteq U^*$ there exists a subset $Y \subseteq X$ with $|Y| \geq \lceil |X|/2 \rceil$ such that $Y \subseteq N(v)$ for some $v \in V^*$.*

Proof: Assume there exists $X \subseteq U^*$ such that all subsets $X \cap N(v), v \in V^*$ are of size smaller than $\lceil |X|/2 \rceil$. Then the weight of the subgraph induced on $(U^* \setminus X, V^*)$ exceeds that of H^*, a contradiction. ■

COROLLARY 5. *Let $H^* = (U^*, V^*, E^*)$ be a maximum weight subgraph of G. Then U^* can be covered by at most $\lfloor \log(2d) \rfloor$ sets, each of which is contained in the neighborhood of some vertex in V^*.*

Proof: Denote $|U^*| = t$. By Lemma 4 there exists a subset $Y \subseteq U^*$ with $|Y| \geq \lceil t/2 \rceil$, such that $Y \subseteq N(v)$ for some $v \in V^*$. The same holds for the set $U^* \setminus Y$, and we can continue in this manner until we cover U^*. By construction we have at most $\lfloor \log t \rfloor$ sets in the cover. Since $t \leq 2d$ by Corollary 3, the result follows. ■

Corollary 5 implies an algorithm to find a maximum weight subgraph. The algorithm tests all collections of at most $\lfloor \log(2d) \rfloor$ subsets of neighborhoods of vertices in V. Since there are $O(n2^d)$ such subsets we have:

THEOREM 6. *The maximum bounded bipartite subgraph problem can be solved in $O((n2^d)^{\log(2d)})$ time.*

A *non-redundant* subgraph is one whose weight cannot be increased by removing any vertex from it. Theorem 6 can be generalized to give the k heaviest non-redundant subgraphs in $O((n2^d)^{\log(2d)} \log k)$ time.

We now extend Theorem 6 to graphs with more general weights: Suppose that edges in G have positive weights and non-edges have negative weights. Define $r = \max_{(u,v),(u',v') \in U \times V} |\frac{w(u,v)}{w(u',v')}|$. We call r the *maximum weight ratio* in G. Similarly to Lemma 4 we can show:

LEMMA 7. *Let $H^* = (U^*, V^*, E^*)$ be a maximum weight subgraph of G. For each set $X \subseteq U^*$ there exists a subset $Y \subseteq X$ with $|Y| \geq \lceil |X|/(r+1) \rceil$ such that $Y \subseteq N(v)$ for some $v \in V^*$.*

THEOREM 8. *Let G be a bipartite graph with d-bounded gene side. Suppose a weight function assigns positive and negative weights to edges and non-edges, respectively, such that the maximum weight ratio is r. Then the k heaviest non-redundant subgraphs in G can be found in $O((n2^d)^{\log_{(r+1)/r}(rd)} \log k)$ time.*

We note that the general problem of finding a maximum weight bipartite subgraph of G is NP-hard, as can be shown by a simple reduction from CLIQUE.

THEOREM 9. *For a bipartite weighted graph G and a number k, the problem of determining if G contains a subgraph of weight at least k is NP-complete, even if each edge of G has positive weight and each non-edge has negative weight.*

INCORPORATING THE DIRECTION OF EXPRESSION CHANGES

In our discussion so far, the underlying bipartite graph used for modeling the data contained edges for significantly changed genes, but ignored the type of change (increase or decrease in the expression level). We can integrate additional information into our model by associating a sign of "up" or "down" with each edge. We now have three types of binary relations in our bipartite graphs: An "up" edge, a "down" edge or no edge. It is reasonable to look for a bicluster in which the conditions tend to affect genes in a *consistent* way, i.e., two clustered conditions should either have always the same effect or always the opposite effect on each of the genes. This leads to the definition of a consistent biclique: Given a bipartite graph $G = (U, V, E)$ with edge sign function $c : E \to \{-1, 1\}$, we say that an induced biclique $H = (U', V', E')$ is *consistent* if there exists an assignment $\tau : U' \cup V' \to \{-1, 1\}$ such that for every $v \in V', u \in U'$ we have $c((u,v)) = \tau(u)\tau(v)$. The maximum consistent biclique problem can be solved in polynomial time by reduction to the standard maximum biclique problem:

PROPOSITION 10. *There is an $O(n2^d)$-time algorithm for the maximum consistent bounded biclique problem on graphs with d-bounded gene side.*

Proof: Given G and c, we construct the graph $G' = (U \cup \overline{U}, V \cup \overline{V}, E')$, where $\overline{U}$ and $\overline{V}$ are copies of U and V, respectively, and $E' = \{(u,v), (\overline{u}, \overline{v}) | (u,v) \in E, c((u,v)) = 1\} \cup \{(u, \overline{v}), (\overline{u}, v) | (u,v) \in E, c((u,v)) = -1\}$. Suppose that (U', V') induce a consistent biclique in G of size k with a sign assignment τ. Then $\{v \in U' \cup V' | \tau(v) = 1\} \cup \{\overline{v} | v \in U' \cup V', \tau(v) = -1\}$ induce a biclique of size k in G'. Conversely, if (U', V') induce a biclique in G', then no pair $u, \overline{u}$ is contained in it, so $\{v \in U \cup V | v \in U' \cup V' \text{ or } \overline{v} \in U' \cup V'\}$ induce a consistent biclique in G of the same size, where $\tau(v) = 1$ if $v \in U' \cup V'$ and $\tau(v) = -1$ if $\overline{v} \in U' \cup V'$. The claim thus follows from Theorem 1. ■

We now introduce the maximum consistent subgraph problem and solve it using the algorithm of Theorem 6.

PROBLEM 3. *(Maximum Consistent Bounded Bipartite Subgraph) Given a weighted signed bipartite graph $G = (U, V, E, c, w)$ with d-bounded gene side, find an induced subgraph $H = (U', V', E')$ and an assignment $\tau : U' \cup V' \to \{-1, 1\}$ maximizing the weight function: $w(U', V') = \sum_{(u,v) \in U' \times V'} (-1)^{f(u,v)} w((u,v))$, where $f(u,v) = 0$ if $(u,v) \notin E'$ and $f(u,v) = \frac{1 - \tau(u)\tau(v)c((u,v))}{2}$ otherwise.*

The special properties of the scoring function together with the assignment of positive weights to edges and negative weights to non-edges enable us to apply the techniques for the unsigned case on G'. The crucial observation is that an induced subgraph of maximum weight in G' cannot contain both copies of the same vertex, since the neighborhoods of two copies are disjoint, so one of them must have a negative contribution to the total score. We conclude:

THEOREM 11. *There is an $O((n2^{d+1})^{log_{(r+1)/r}(rd)})$-time algorithm for the maximum consistent bounded bipartite subgraph problem on graphs with maximum weight ratio r.*

Note that the weighting scheme defined above is heuristic in nature and is not a direct outcome of our statistical model. An exact scheme can be obtained using a more detailed statistical model. We omit the details.

SIGNIFICANCE EVALUATION

In this section we develop a method for computing the statistical significance of a bicluster. The method computes a "p-value" for a given bicluster B, i.e., the probability of finding at random a bicluster with at least the weight of B. Let $H = (U', V', E')$ be a subgraph. Suppose at first that U' is fixed, and we wish to compute the probability of observing H, given that its weight is maximum among all subgraphs over the same set of conditions U'. To this end, we note that H is obtained by taking into V' all vertices $v \in V$ whose weight $w(\{v\}, U')$ is positive. Let $f_{U'} : V \to \mathcal{R}$ be a function defined as $f_{U'}(v) = \max\{0, w(\{v\}, U')\}$. For each $v \in V$ we can view $f_{U'}(v)$ as a random variable. The weight of H is just $w(H) = \sum_{v \in V} f_{U'}(v)$, a sum of independent random variables. These variables can be shown to satisfy the requirements of Liapunov's generalization of the Central Limit Theorem (cf. (DeGroot, 1989)), implying that when $|V|$ is sufficiently large, the weight of H is approximately normally distributed. Hence, we can compute the expectation and variance of $w(H)$ and derive a p-value $p(H)$ for observing a subgraph with such weight.

Finally, we have to accommodate for the fact that the subset U' is optimized by the algorithm. For that, we apply Bonferroni's rule and compute an upper bound on the p-value: $p^*(H) = p(H) \sum_{i=1}^{\lceil (r+1)d \rceil} \binom{m}{i}$, since we are trying all subsets of U of size at most $\lceil (r+1)d \rceil$, where r is the maximum weight ratio in the graph. Henceforth we call $\log p^*(H)$ the *significance value* of H.

THE SAMBA ALGORITHM

We used the methods developed above in implementing a novel biclustering algorithm called SAMBA for finding high quality and distinct biclusters. SAMBA works as follows: We first form the bipartite graph and calculate vertex pair weights using one of the weighting methods described above. We consider a gene to be up (down) regulated in a condition if its standardized level with mean 0 and variance 1 is above 1 (below -1). Depending on the data, we may choose to work with signed or unsigned graphs. When using the likelihood weighting scheme we optimize the value of p_c by measuring the significance of the resulting biclusters.

In the second phase of the algorithm we apply the hashing technique of the algorithm in Figure 2 to find the heaviest bicliques in the graph. In fact, we look for the k best bicliques intersecting every given condition or gene. This can be done efficiently using a standard heap data structure. To save on time and space we ignore genes with degree exceeding some threshold D, and hash for each gene only subsets of its neighbors of size ranging from N_1 to N_2.

The third phase of the algorithm performs a local improvement procedure on the biclusters in each heap. The procedure iteratively applies the best modification to the bicluster (addition or deletion of a single vertex) until no score improvement is possible. To avoid similar biclusters whose vertex sets differ only slightly, we greedily filter from the output biclusters whose intersection with a previous solution (number of shared conditions times number of shared genes) is above L%.

The implementation was built on top of the GENESYS platform (Tanay & Shamir, 2001). Typical runs of the algorithm for large datasets (15,000 genes and 500 conditions) use parameter values $D = 40, N_1 = 4, N_2 = 6, k = 20$ and $L = 30$. A complete run of SAMBA on such dataset takes a few minutes on a standard PC with limited memory (256MB).

EXPERIMENTAL RESULTS

We analyzed the performance of our algorithm on several gene expression datasets and compared it to an extant biclustering algorithm. Our main tool in evaluating biclustering results using prior biological knowledge is a *correspondence plot*. The plot depicts the distribution of p-values of the produced biclusters, using for evaluation a known (putatively correct) classification of conditions (e.g., to various cancer types) or a given gene annotation. We describe the plot when a classification is given. 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.

p-values are calculated according to the known classification 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 $p(B) = \sum_{k=b_i}^{b} \binom{|C_i|}{k}\binom{m-|C_i|}{b-k}/\binom{m}{b}$. Hence, the p-value measures the probability of obtaining at least b_i elements from the class in a random set of size b. One should note, that high quality biclusters can also identify phenomena that are not covered by the given classification. Nevertheless, we expect a large fraction of our biclusters to conform to the known classification. Note that our algorithm is unsupervised and does not use the classification in any way.

Performance Evaluation

We first compared the performance of the different weighting schemes for graph edges and non-edges presented in previous sections. To this end we used the dataset of (Alizadeh *et al.*, 2000). It contains the expression levels of 4,026 genes over 96 human tissue samples, which are classified into nine types of lymphoma and normal ones. Figure 3(A) shows the correspondence plots for the three sug-

gested weighting schemes. It is evident that the likelihood-based scoring method with $p_c = 0.9$ outperforms the other schemes. Consequently, all the experiments we report below were performed using likelihood-based weights with $p_c = 0.9$.

Next we compared our performance to that of (Cheng & Church, 2000) on the lymphoma dataset. Correspondence plots for the two biclusterings are shown in Figure 3(B). The plots demonstrate that the biclusters generated by our algorithm describe much more accurately the known classification, and are thus more informative for extracting additional novel biological insights. As a reference we added a correspondence plot calculated on a random annotation of the 96 samples (preserving class sizes). It shows that random p-values are at very low levels and therefore the signal in the biclusters is indeed very strong.

As an additional test, we generated a random expression dataset with the same characteristics as the lymphoma data. This was done by generating a random bipartite graph with the same degree sequence as the original graph for this dataset. We then executed SAMBA on this synthetic data and recorded the resulting biclusters. Figure 3(C) presents a scatter plot of the significance values of biclusters vs. their log likelihood (weight) on each dataset. It can be seen that significance values on the random data are well separated from those computed on the original data and, furthermore, only two random biclusters have significance values below 0. The plot for the real data also demonstrates the quadratic fit between the significance value of a bicluster and its weight. Both observations support our use of weights for detecting biclusters with low significance values.

Functional Annotation in Yeast

We have compiled a data set including 515 conditions for the 6,200 yeast ORFs. The data was collected from five different experiments (Hughes *et al.*, 2000; Gasch *et al.*, 2000, 2001; Spellman *et al.*, 1998; Ideker *et al.*, 2001). Analysis by SAMBA generated 2,406 biclusters ranging over 4,961 genes and 515 conditions. Many of the biclusters contain conditions from several experiments. Hence, the biclustering process truly integrates the data from different experiments.

We utilized our biclustering to perform a naive functional annotation in conjunction with the SGD GO consortium (2000) annotation, as follows: We used the fourth level in the GO annotation as a classification of the genes. We chose those biclusters in which more than 60% of their annotated members had the same class. Out of those, we only used biclusters that were functionally enriched (p-value below 10^{-4}). We then assigned the unannotated genes in those biclusters to this most abundant class. Note that each gene may be annotated more than once, as is the case for the curated GO annotations. For cross validation, we performed 100 runs and in each one we hid 30% of the annotations, and tested our success rate in annotating those hidden genes.

The results of these runs are summarized in Figure 4(A,B). Overall, 81.5% of our test set annotations matched those known from SGD. The results demonstrate that biclusters qualifying as annotators accurately identify biological processes and may be used to extrapolate from known annotations to uncharacterized genes. We thus set out to annotate unknown genes (based on the entire GO annotation). Using the same procedure, we obtained 196 annotations of unknown genes as summarized in Figure 4(C).

Detailed analysis of the results further demonstrates the power of bicluster analysis. For example, one of the biclusters in Figure 5(A) contains DNA repair genes and a large family of Y' DNA helicase genes. The Y' genes are strong paralogs present at the end of the yeast chromosome, and their function is not fully understood. This bicluster raises the conjecture that Y' genes and DNA repair genes are associated. Indeed, a recent study (Yamada *et al.*, 1998) suggested a connection between DNA damage and repair mechanisms to this family. Another bicluster shown in this figure contains several phosphate and glucose related genes grouped with several unknown genes, which may be assigned a putative function according to their expression pattern in this bicluster.

Human Cancer Data

Large datasets of clinical samples are an ideal target for biclustering. We can use biclusters to associate genes with specific clinical classes or for classifying samples. We demonstrate the applicability of our methods for tissue expression analysis in Figure 5(B). The lymphoma dataset is characterized by well defined expression patterns differentiating three types of lymphoma, DLBCL, CLL and FL from one another. However, using hierarchical clustering (see (Alizadeh *et al.*, 2000)) germinal center tissues are interleaved within the DLBCL class. In contrast, SAMBA produced two biclusters associating the two germinal center tissues in the data set with *both* the DLBCL and FL classes, thereby uniquely characterizing them. It is our ability to associate several statistically significant signals with each condition or gene that makes such delicate analysis possible.

DISCUSSION

We have developed a new statistical-algorithmic approach to finding significant biclusters in gene expression data, and demonstrated its utility on diverse datasets. In addition to facilitating novel gene annotation at high specificity and more accurate subclassification of cancer tissues, the

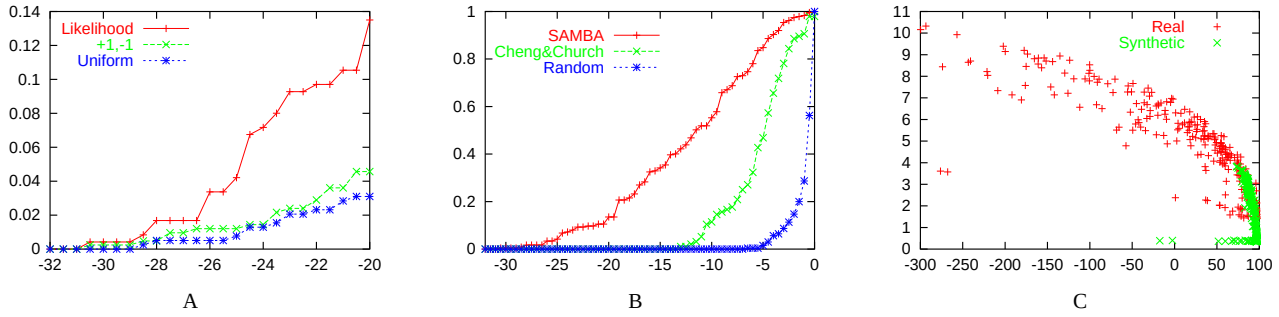


Fig. 3. Performance of different weighting schemes and algorithms. A: Correspondence plots for biclusters generated with different weighting schemes. B: Correspondence plots for SAMBA, the algorithm of Cheng and Church (2000), and random biclusters. Likelihood weights use $p_c = 0.9$. C: Scatter plots of significance values on synthetic and real data. x-axis: significance value, y-axis: bicluster weight.

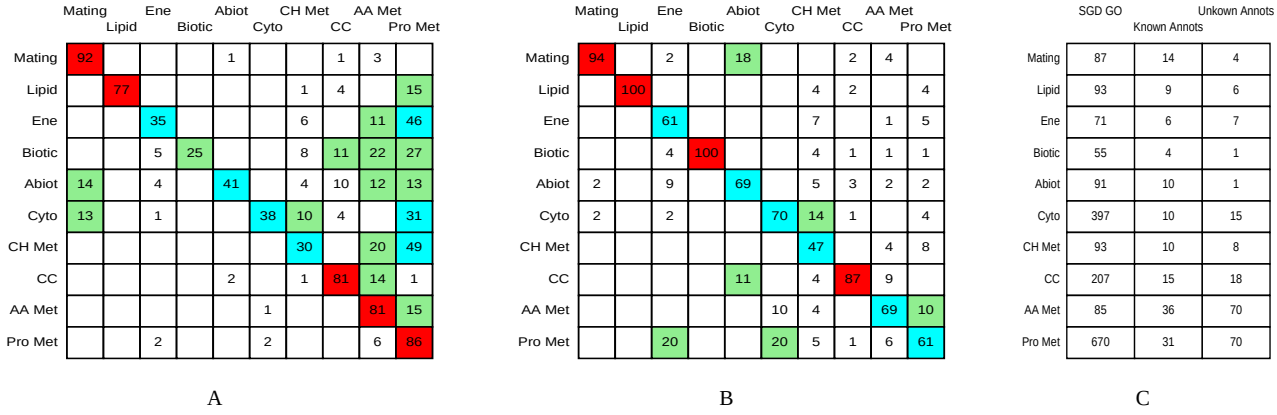


Fig. 4. Yeast functional annotation. A: Annotation specificity. The table depicts the annotation accuracy measured using 70:30 cross-validation. Rows represent classes assigned using our method and columns represent SGD GO classes. Cell (x, y) contains the percentage of genes annotated x that belong to GO class y. Higher percentages are darker. B: Annotation sensitivity calculated w.r.t. annotated genes only. Cell (x, y) contains the percentage of SAMBA annotated genes that belong to GO class y and were annotated x. C: Annotation of unknown genes. The table shows for each functional class its size in the SGD GO annotation, the number of genes that belong to this class and were annotated by SAMBA, and the number of unknown genes assigned to this class by SAMBA. Abbreviations for functional classes: Mating - mating (sensu Saccharomyces, Fungi); Lipid - lipid metabolism; Ene - energy pathway; Biotic - response to biotic stimulus; Abiot - response to abiotic stimulus; Cyb - cytoplasm organization and biogenesis; CH Met - carbohydrate metabolism; CC - mitotic cell cycle; AA Met - amino acid and derivative metabolism; Pro Met - protein metabolism and modification.

method allows performing simultaneously class discovery and feature selection.

Statistically significant biclusters are generated in an unsupervised fashion directly from the dataset by our algorithm, and can be used in many contexts. Each bicluster characterizes some tight biological phenomenon and can be evaluated using existing biological knowledge or provide new hypotheses.

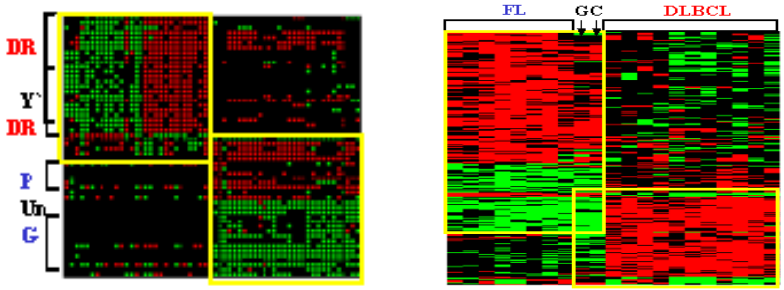
We are currently extending the theoretical and practical study to multiple response levels. A refined version of the software will soon be available on our website.

ACKNOWLEDGMENTS

We thank Rani Elkon, Itsik Pe'er and Martin Kupiec for helpful remarks. R. Shamir was supported by a pilot grant from the McDonnell Foundation, and by a grant from the US-Israel Binational Science Foundation. R. Sharan was supported by an Eshkol fellowship from the Ministry of Science, Israel.

REFERENCES

Alizadeh, A. *et al.* (2000). Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, **403**, 503–511.



A B

Fig. 5. Sample biclusters. Each figure shows the expression patterns in two related biclusters. Rows correspond to genes and columns correspond to conditions or tissues. Expression levels: Dark - up; light - down; black - unchanged. The frames indicate bicluster boundaries. A: Yeast biclusters. A group of unannotated subtelomeric Y' genes is clustered with several DNA repair (DR) genes (upper left corner). This raises the hypothesis of association between DNA repair mechanisms and the Y' genes, which was independently suggested recently. Some of the genes in this bicluster appear also in another presented at the lower right corner, which contains phosphate (P) and glucose (G) related genes. Several unannotated genes (Un) may be assigned a putative function in this way. B: Biclusters in lymphoma data. Germinal center (GC) tissues are biclustered with both DLBCL and FL tissues, thus uniquely characterizing them as a distinct class.

Ben-Dor, A., Chor, B., Karp, R. & Yakhini, Z. (2002). Discovering local structure in gene expression data: The order-preserving submatrix problem. In *Proc. RECOMB'02*. ACM Press. To appear.

Cheng, Y. & Church, G. (2000). Biclustering of expression data. In *Proc. ISMB'00*. AAAI Press, pp. 93–103.

DeGroot, M. (1989). Probability and Statistics. Addison-Wesley.

Gasch, A. *et al.* (2001). Genomic expression responses to DNA-damaging agents and the regulatory role of the yeast ATR homolog mec1p. *Mol. Biol. Cell*, **12(10)**, 2987–3003.

Gasch, A. P. *et al.* (2000). Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell*, **11**, 4241–57.

Gene Ontology Consortium (2000). Gene Ontology: tool for the unification of biology. *Nature Genetics*, **25**, 25–29.

Getz, G., Levine, E. & Domany, E. (2000). Coupled two-way clustering analysis of gene microarray data. *Proc. Natl. Acad. Sci. USA*, **97(22)**, 12079–84.

Golumbic, M. C. (1980). Algorithmic Graph Theory and Perfect Graphs. Academic Press, New York.

Hartigan, J. (1975). Clustering Algorithms. John Wiley and Sons.

Hochbaum, D. S. (1998). Approximating clique and biclique problems. *Journal of Algorithms*, **29**, 174–200.

Hughes, T. *et al.* (2000). Functional discovery via a compendium of expression profiles. *Cell*, **102**, 109–26.

Ideker, T. *et al.* (2001). Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. *Science*, **291**, 929–34.

Lazzeroni, L. & Owen, A. (2000). Plaid models for gene expression data. Technical report, Stanford University.

Segal, E., Taskar, B., Gasch, A., Friedman, N. & Koller, D. (2001). Rich probabilistic models for gene expression. *Bioinformatics*, **17**, S243–52.

Sharan, R., Elkon, R. & Shamir, R. (2002). Cluster analysis and its applications to gene expression data. In *Ernst Schering workshop on Bioinformatics and Genome Analysis*. Springer Verlag. To appear.

Spellman, P. T., Sherlock, G. *et al.* (1998). Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. *Mol. Biol. Cell*, **9**, 3273–3297.

Tanay, A. & Shamir, R. (2001). Computational expansion of genetic networks. *Bioinformatics*, **17**, S270–8.

Yamada, M., Hayatsu, N., Matsuura, A. & Ishikawa, F. (1998). Y'-Help1, a DNA helicase encoded by the yeast subtelomeric Y' element, is induced in survivors defective for telomerase. *J. Biol. Chem.*, **273(50)**, 33360–6.