Proceedings of the 2005 IEEE
Engineering in Medicine and Biology 27th Annual Conference
Shanghai, China, September 1-4, 2005

Finding Co-Clusters of Genes and Clinical Parameters

Sungroh Yoon, *Student Member, IEEE,* Luca Benini, *Senior Member, IEEE,* and
Giovanni De Micheli, *Fellow, IEEE*

Abstract— **For better understanding of genetic mechanisms underlying clinical observations, we often want to determine which genes and clinical traits are interrelated. We introduce a computational method that can find *co-clusters* or groups of genes and clinical parameters that are believed to be closely related to each other based upon given empirical information. The proposed method was tested with data from an *Acute Myelogenous Leukemia* (AML) study and identified statistically significant co-clusters of genes and clinical traits. The validation of our results with *Gene Ontology* (GO) as well as the literature suggest that the proposed method can provide biologically meaningful co-clusters of genes and traits.**

I. Introduction

The invention of DNA microarray technologies has enabled researchers to simultaneously monitor the expression level of virtually all known genes [5], [9]. Thus, for the purpose of finding genes related to a certain clinical trait (or parameter) of interest, it has become feasible to examine all the genes available and then select only those whose expression is consistently correlated with the trait over many samples. Although correlation does not always imply causality, this approach has been successful in many studies as an attempt to understand genetic mechanisms underlying clinical observations [3], [13], [16], [17].

To measure correlation between a gene and a clinical trait, existing approaches obtain a vector of the expression level of the gene over a number of samples and another vector of the value of the clinical trait over the same samples and then calculate statistical correlation between two vectors. By applying this procedure to many genes, we can identify some genes correlated to the clinical trait of interest.

Proceeding one step further from prior methods that can reveal one-to-many relationships between a single trait and multiple genes (or vice versa), we present in this paper a method that can find many-to-many relationships between genes and traits using a clustering technique called *co-clustering*. Here the term co-clustering or *biclustering* [10] refers to an unsupervised learning technique that performs simultaneous clustering of rows and columns in a matrix to find (possibly) overlapping submatrices covering the matrix.

More specifically, given gene expression data and clinical parameter values, we first create a matrix called *correlation matrix* that can collectively represent the degree of correlation between genes and clinical traits. Each row and column of this matrix corresponds to a gene and a clinical trait, respectively. Then, our method searches *co-clusters* or submatrices (with some semantics to be defined) covering the correlation matrix.

The co-cluster search algorithm proposed in this study is an extension of our earlier work [18], which can deterministically find all the co-clusters satisfying specific input parameters in an efficient manner. This prior work possessed clear advantages over heuristic methods that can provide only partial solutions and other exact algorithms that are not scalable to large-scale problems. In addition to this efficiency, our extended algorithm can detect not only positive correlation but also negative correlation, which was neglected in our previous study.

We tested our method with the *Acute Myelogenous Leukemia* (AML) data set [3], which consists of a DNA microarray data matrix and a parameter matrix for 119 patients, 15 parameters, and 6283 genes. We identified 43 co-clusters using the proposed method. To justify the grouping of certain genes and clinical traits by the co-clusters found from the AML data, we present some supporting evidence of co-clustered genes and traits from the literature. In addition, we show that certain *Gene Ontology* terms annotating genes in some co-clusters are significantly over-represented. Taken together, these experimental studies suggest that our method can find biologically meaningful co-clusters.

Section II explains at length our method to find co-clusters. Experimental results and discussions are presented in Section III, followed by concluding remarks in Section IV.

II. Method

Let S represent a set of clinical samples. For each sample in S, gene expression levels are measured by the DNA microarray technology of choice. Let G be the set of genes in the measurement. Clinical traits are recorded for each sample. Let T be the set of the recorded traits.

The input of our method consists of two data matrices constructed from the above experiments. One is a gene expression data matrix denoted by pair $A = (G, S)$. That is, $A \in \mathbb{R}^{|G|\times|S|}$, and the element a_{ik} of the matrix A represents the expression level of gene i for sample k. The other matrix is denoted by pair $B = (T, S)$, where the element b_{jk} of the matrix B corresponds to the value of trait j for sample k. Depending upon the type of trait j, b_{jk} may be quantitative, categorical, or others. We make the columns of A and B arranged in the same order.

This work was supported by a grant of Jerry Yang and Akiko Yamazaki.

S. Yoon is with the Computer Systems Laboratory, Stanford University, CA 94305, USA (corresponding author. phone: +1-650-724-5180; fax: +1-650-725-9802; e-mail: sryoon@stanford.edu).

L. Benini is with DEIS, University of Bologna, Bologna 40136, Italy (e-mail: lbenini@deis.unibo.it).

G. De Micheli is with the Integrated Systems Center, EPFL, Lausanne, Switzerland (e-mail: giovanni.demicheli@epfl.ch).

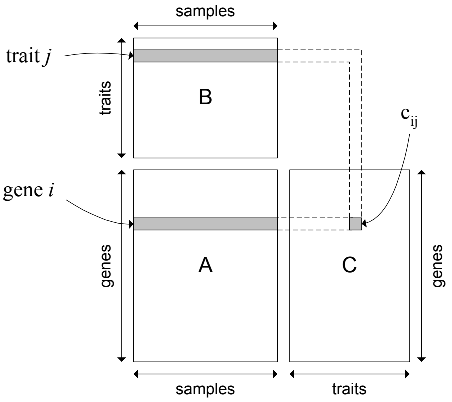


Fig. 1. Construction of the correlation matrix. A co-cluster appears as a submatrix of the correlation matrix C.

The output is a set of co-clusters. A co-cluster is composed of a gene set $I \subseteq G$ and a trait set $J \subseteq T$ and represents a group of genes and traits closely related to each other, given the input matrices A and B. A formal definition of a co-cluster will be presented in Section II-B.

The proposed method consists of the following three steps:

1) An intermediate data matrix called *correlation matrix* is constructed from the input matrices (Section II-A).
2) Special co-clusters called *pairwise co-clusters* are found in the correlation matrix (Section II-C).
3) Co-clusters are derived from the pairwise co-clusters (Section II-D).

A. Correlation matrix computation

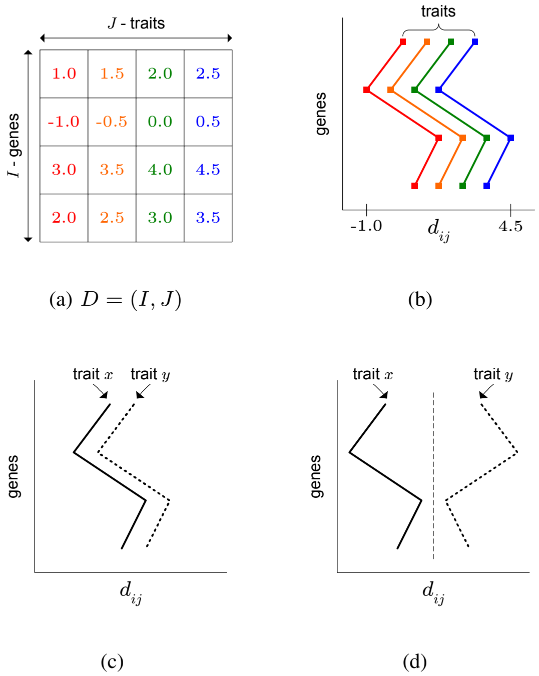
We combine the input matrices A and B and construct a *correlation matrix*. This matrix is denoted by C, and the row set and the column set of C are G and T, respectively. The element c_{ij} indicates the degree of correlation between gene i and trait j, as illustrated in Figure 1.

More precisely, the element c_{ij} is the statistic defined in *significance analysis of microarrays* (SAM) [13], namely,

$$c_{ij} = \frac{r_{ij}}{s_{ij} + s_0}, \qquad (1)$$

where r_{ij} is a score to measure the degree of correlation between the expression level of gene i and the value of clinical trait j, s_{ij} is the "gene-specific scatter" or the standard deviation of repeated expression measurements, and s_0 is a "fudge" factor to prevent the computed statistic from becoming too large when s_{ij} is close to zero [5]. The specific definition of r_{ij} varies depending upon the type of clinical trait j. For example, if clinical trait j has quantitative values then r_{ij} is defined in terms of the Pearson correlation coefficient [12] between the i-th row vector of the matrix A and the j-th row vector of the matrix B.

When calculating c_{ij}, we must follow a procedure for multiple comparisons, thus ensuring that too many falsely significant ones are not declared [5], [12]. To this end, the *false discovery rate* (FDR) is estimated for each c_{ij} by random permutation of the data for gene expression among the different experimental arms. Then, the p-value of the statistic c_{ij} is defined as follows.



(a) $D = (I, J)$

(b)

(c)

(d)

Fig. 2. Defining co-clusters. (a) $D = (I, J)$, an example co-cluster with the gene set I and the trait set J. (b) The column vectors of D show the same trend. (c) Positive correlation between traits x and y. (c) Negative correlation between traits x and y.

Definition 1: The p-value for score c_{ij}, denoted by p-value(c_{ij}), is the lowest FDR at which the score is called significant.

Further details on defining r_{ij}, estimating the FDR, and computing s_0 are beyond the scope of this paper, and the interested readers are directed to [13].

B. Defining co-clusters

We are interested in finding a submatrix (of the correlation matrix) in which the values on all columns exhibit some common behavior. An example is presented in Figures 2(a) and 2(b). In particular, we focus on searching submatrices where every pair of column vectors show positive or negative correlation as seen in Figures 2(c) and 2(d).

To assess the degree of correlation, we introduce a metric called *linear deviation*, which resembles a conventional statistic such as the Pearson correlation coefficient but can be computed more efficiently, especially in the current setup where we want to measure correlation between many sub-vectors of two vectors. The introduction of this metric is not to deny the effectiveness of a conventional statistic but to transform it to a computation-efficient form, minimizing loss in the detection power.

Definition 2: For V, a vector on $\mathbb{R}$, the range of V, denoted by $\textsc{range}(V)$, is the absolute difference between the largest and the smallest elements of V.

Definition 3: Given V and W, two real vectors of the same dimension, the *linear deviation* of V and W, denoted by LIN-DEV(V, W), is defined as

$$min\{\text{RANGE}(V - W), \text{RANGE}(V + W)\}. \tag{2}$$

The example in Figure 3 reveals the relationship between LIN-DEV and the Pearson correlation coefficient: a lower value of LIN-DEV typically corresponds to a higher level of either positive or negative correlation. Using the metric LIN-DEV, a co-cluster is formally defined as follows.

Definition 4: Given the correlation matrix $C = (G, T)$ and thresholds $\tau \geq 0$ and $\pi > 0$, a *co-cluster* is a matrix, denoted by $D = (I, J)$, satisfying the following conditions: (1) $I \subseteq G$ and $J \subseteq T$; (2) for any two column vectors V and W of size $|I|$ in D, LIN-DEV$(V, W) \leq \tau$; (3) $\forall i \in I, \forall j \in J$, p-value$(c_{ij}) < \pi$.

Condition (1) indicates that D is a submatrix of the correlation matrix C. Condition (2) is to require that every pair of $|I|$-dimensional column vectors from D exhibit correlation with respect to the metric LIN-DEV. The last condition is to find co-clusters with statistically significant elements. In this study, we search only *maximal* co-clusters or those that are not contained by others.

C. Discovering pairwise co-clusters

After having computed the correlation matrix, the next step of our method is to find a special type of co-cluster called *pairwise co-cluster*. A pairwise co-cluster is a co-cluster with only two traits and can therefore be represented by a submatrix (of the correlation matrix) with two columns. Pairwise co-clusters are used later in Section II-D as seeds to find (non-pairwise) co-clusters.

To find a pairwise co-cluster in the correlation matrix $C = (G, T)$, we first select two distinct columns $v, w \in T$ and construct from them two $|G|$-dimensional column vectors $V = (c_{1v}, c_{2v}, \ldots, c_{|G|v})$ and $W = (c_{1w}, c_{2w}, \ldots, c_{|G|w})$. Then, we compare V and W to identify I, a set of dimensions over which V and W are correlated ($I \subseteq G$). Finally, we remove all $i \in I$ such that p-value of c_{iv} or c_{iw} is greater than a given threshold. By Definition 4, the matrix denoted by pair $(I, \{v, w\})$ represents a co-cluster, and this co-cluster with only a pair of traits is called *pairwise co-cluster*.

Here we further explain the procedure to compare two vectors V and W and identify the dimension set I. The other details on finding pairwise co-clusters are straightforward and are thus omitted.

Algorithm 1 presents the procedure to find I, a set of dimensions over which two vectors V and W are *positively* correlated. Invoking this algorithm with $-V, W$ or $V, -W$ provides a set of negatively correlated dimensions.

Algorithm 1: Find positively correlated dimensions for two vectors

```
input : $V$ and $W$, two $n$-dimensional vectors
input : $\tau$, a threshold
output: $I \subseteq \{1, 2, \ldots, n\}$, a set of dimensions
1  for $i = 1$ to $n$ do
2      $S[i].score := V_i - W_i$;
3      $S[i].dim := i$;
4  sort $S$ in ascending order with respect to the field $score$;
5  $begin := 1$, $end := 2$;
6  while ($end \leq n$) do
7      if ($S[end].score - S[begin].score \leq \tau$) then
8          $end := end + 1$;
9          if ($end > n$) then
10             Report $\{S[begin].dim, \ldots, S[end-1].dim\}$;
11     else
12         Report $\{S[begin].dim, \ldots, S[end-1].dim\}$;
13         repeat
14             $begin := begin + 1$;
15         until ($begin = end$) or
           ($S[end].score - S[begin].score \leq \tau$);
```

The key idea of Algorithm 1 is simple: when the elements of a vector V are arranged in an ascending or descending order, $range(V)$ is simply the absolute difference between the first and the last elements of V, and no other elements need to be examined. Thus, in Lines 1–3, the vector $S = V - W$ is rearranged in ascending order. Then, in Lines 6–15, the algorithm examines sub-vectors of S and reports those whose range is not greater than the threshold τ. The boundary of a sub-vector under consideration is indicated by two pointers $begin$ and end. The algorithm relies on these pointers to find only maximal subsets and to handle multiple (and possibly overlapping) instances of I.

The worst-case complexity of Algorithm 1 is polynomial in n, the number of dimensions in two vectors.

D. Deriving co-clusters

In the last step of our method, co-clusters are derived from pairwise co-clusters. For the sake of explanation, let pair (I, J) represent a pairwise co-cluster with $J = \{x, y\}$ and assume that we want to expand this co-cluster "horizontally" by adding z, a third column index, to the set J. Let (I', J') denote this new co-cluster. Since we are interested in finding only maximal co-clusters, assume that we are to find the instance of I' with maximal cardinality.

Clearly, the set J' is a superset of J, namely, $J' = J \cup \{z\} = \{x, y, z\}$. In contrast, the set I' is a subset of I by construction[1]. In what follows, we explain more precisely what the set I' should be.

First, $I' \subseteq I$ as previously stated. Second, if the pair $(I', \{x, y, z\})$ represents a co-cluster, then by definition, $(I', \{x, z\})$, $(I', \{y, z\})$, and $(I', \{x, y\})$ should be pairwise co-clusters. Now let I_{xz} and I_{yz} be the row sets of pairwise co-clusters obtained by Algorithm 1 for column pairs $\{x, z\}$ and $\{y, z\}$, respectively. Then, $I' \subseteq I_{xz}$ and $I' \subseteq I_{yz}$, since Algorithm 1 finds only maximal pairwise co-clusters. Therefore, $I' \subseteq I \cap I_{xz} \cap I_{yz}$, and we can obtain the instance of I' with the largest cardinality by setting $I' = I \cap I_{xz} \cap I_{yz}$.

In general, given a co-cluster (I, J), we can add element z to the set J and produce a new maximal co-cluster $(I', J \cup \{z\})$ with

$$I' = I \cap \left(\bigcap_{\forall j \in J} I_{jz} \right), \tag{3}$$

[1] If any two trait vectors show a common trend over $|G|$ dimensions in the correlation matrix, then three traits including the two traits cannot show a common trend over more than $|G|$ dimensions.

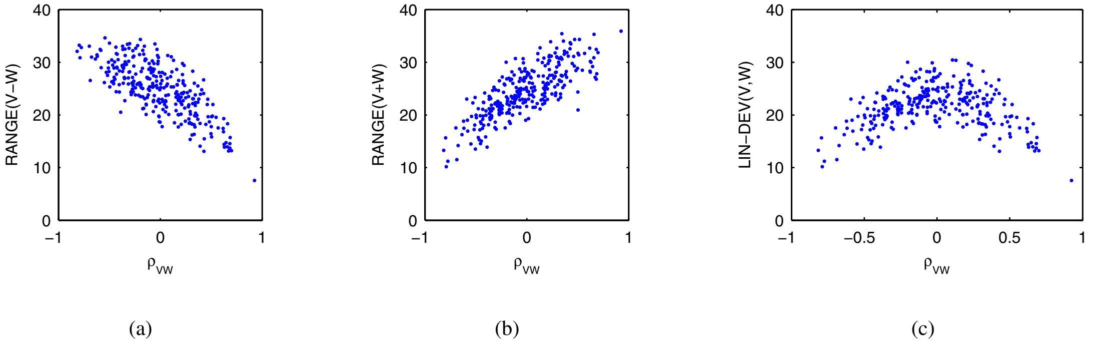


Fig. 3. An empirical study to show the relationship between LIN-DEV and the Pearson correlation coefficient. (a) A pair of 10-dimensional vectors (V, W) were generated in such a way that the Pearson correlation coefficient ρ_{VW} between the two vectors randomly lies in $[-1, 1]$. The elements of the two vectors were ranged in $[-10, +10]$. We also calculated $\text{RANGE}(V - W)$ and then place point $(\rho_{VW}, \text{RANGE}(V - W))$ on a plot. We repeated this procedure 300 times. Two 300-dimensional vectors X and Y were created from the 300 points (x, y) on the plot. We calculated the Pearson correlation coefficient ρ_{XY} between vectors X and Y to obtain $\rho_{XY} = -0.7433$ ($P < 10^{-54}$), indicating significant negative correlation between $\text{RANGE}(V - W)$ and ρ_{VW}. (b) The same procedure repeated for a plot of ρ_{VW} versus $\text{RANGE}(V + W)$: $\rho_{XY} = 0.7934$ ($P < 10^{-66}$). (c) Focusing on a low value of the metric LIN-DEV thus enables us to detect either strongly positive or highly negative correlation between V and W.

```
Algorithm 2: Mining co-clusters
   input : $C = (G, T)$, a correlation matrix
   output: co-clusters
 1 Generate pairwise co-clusters by Algorithm 1;
 2 foreach $\{x, y\} \subseteq T$ do
 3     Create vertex $v$;
 4     $v.J := \{x, y\}$;
 5     $ConstructTrieInPreorder(v)$;
 6     delete $v$;
 7 Remove redundancy and return remaining co-clusters;
 8 procedure $ConstructTrieInPreorder$ (vertex $v$)
 9 begin
10     if $|v.J| = |\{x, y\}| = 2$ then
11         $v.I := I_{xy}$;
12     else
13         vertex $p := v.parent$;
14         $k :=$ the element in $v.J - p.J$;
15         $v.I := p.I \cap (\cap_{\forall j \in p.J} I_{jk})$;
16     if $v.I = \emptyset$ then return;
17     Collect pattern $(v.I, v.J)$;
18     $l :=$ the "largest" element in $v.J$ wrt a total order $\prec$;
19     $J := \{j | j \in T$ and $l \prec j\}$;
20     foreach $j \in J$ do
21         create vertex $w$;
22         $w.J := v.J \cup \{j\}$;
23         $w.parent := v$;
24         $ConstructTrieInPreorder(w)$;
25         delete $w$;
26 end
```

where I_{jz} is a maximal pairwise co-cluster for columns $\{j, z\}$. Our approach to deriving co-clusters from pairwise co-clusters is based upon this idea.

Algorithm 2 provides the details. Recall that T is the set of clinical traits or the set of column indices in the correlation matrix $C = (G, T)$. Algorithm 2 examines elements $J \in 2^T$ in such an order that Equation 3 can be exploited to find a co-cluster (I, J). To this end, a data structure called *prefix tree* or *trie* [1] is employed to systematically represent the elements of the power set 2^T. For the sake of explanation, assume that $T = \{1, 2, 3, 4\}$. The prefix tree representing the power set 2^T is depicted in Figure 4(a). Each vertex v of the prefix tree is associated with two sets $v.I$ and $v.J$ such that $v.I \subseteq G$ and $v.J \subseteq T$. Indicated inside a vertex in Figure 4(a) is $v.J$.

The prefix tree is traversed in preorder by Algorithm 2. In the worst case, the algorithm needs to visit every vertex of the prefix tree. Thus, the worst-case complexity of Algorithm 2 is exponential in $|T|$. However, in most cases, this exhaustive enumeration is avoided, and the running time of our algorithm on typical benchmarks is practical. To see the reason, observe that the subtree rooted at a vertex v with $v.I = \emptyset$ needs not be visited and removed from the prefix tree. The condition $v.I = \emptyset$ means that the matrix represented by the pair $(v.I, v.J)$ cannot be a co-cluster. Thus, any pair (I', J') with $J' \supseteq v.J$ cannot represent a co-cluster, either, regardless of the set I'. For instance, assume that $v.I = \emptyset$ for the top left vertex v with $v.J = \{1, 2\}$. Then, as shown in Figure 4(b), the vertex v and the subtree rooted at v are removed from the tree, producing the reduced tree in Figure 4(c).

Several remarks are in order. First, the algorithm does not maintain the prefix tree in its entirety. Only a part of the subtree is constructed at a time and removed after its use. To emphasize this, the procedure used in Algorithm 2 was termed "*Construct*Trie" rather than "*Traverse*Trie." Second, multiple instances of $v.I$ can be produced in Line 15, since $p.I$ and I_{jk} in this line are not necessarily unique.

E. Remarks

Our method provides both a list of co-clusters found in the correlation matrix and the graphical images of these co-clusters. For example, Figure 6(d) shows some co-clusters discovered from the correlation matrix in Figure 6(c), which was constructed from the data in Figures 6(a) and 6(b). The reader may first refer to Figure 5 for more details on reading the images in Figure 6(d).

Given input matrices A and B, our method can find all co-clusters that satisfy specific input parameters. If desired, the users can also define a criterion to further select co-clusters of specific interest.

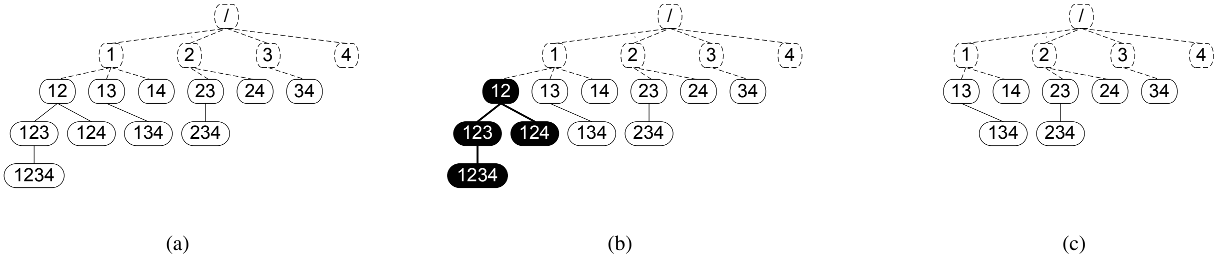


Fig. 4. Prefix tree example. Each vertex v is associated with two sets $v.I$ and $v.J$. Indicated inside each vertex v is $v.J$. Algorithm 2 examines only those vertices with $|v.J| >= 2$. (a) A prefix tree representing the power set 2^T, where $T = \{1, 2, 3, 4\}$. (b) Assuming $v.I = \emptyset$ for the top most vertex with $v.J = \{1, 2\}$, the subtree rooted at the vertex v can be removed. (c) Reduced prefix tree.

The problem of co-clustering is inherently intractable [10], and the worst-case complexity of our method is exponential in $|T|$, the total number of samples. However, the response time of our method was practical in all the cases we tested.

III. Experimental Results

A. Experiment procedure

We tested our method with data from an *Acute Myelogenous Leukemia* (AML) study [3]. The AML data set used includes two matrices. One is a gene expression data matrix with 6283 genes and 119 samples as shown in Figure 6(b). The other is a matrix of 15 clinical parameters measured from the identical samples as seen in Figure 6(a). We used the procedure described in Section II-A to produce the correlation matrix[2] presented in Figure 6(c). The algorithm to find co-clusters in the correlation matrix was implemented in ANSI C++ on a 3.02 GHz Linux machine with 4 GB RAM. The implementation was invoked with the parameters listed in Table I. The running time was in the order of minutes.

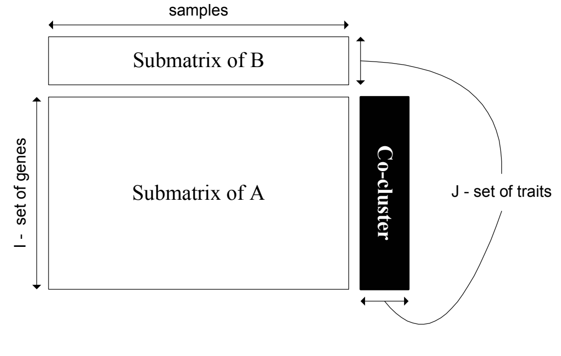


Fig. 5. Composition of each images in Figure 6(d). Each figure is composed of three panels (heat maps). The panel in the middle corresponds to the submatrix (I, S) of the microarray matrix A, where $I \subseteq G$. The panel at the top corresponds to the submatrix (J, S) of the trait matrix B, where $J \subseteq T$. The right panel represents a co-cluster or a submatrix (I, J) of the correlation matrix C. The colored bars at the left of the middle panel indicate those genes from the gene groups C–H labeled in Figure 6(b).

B. Results and discussion

We identified 43 co-clusters from the AML data set. Figure 6(d) shows some of the co-clusters found. Refer to Figure 5 for how to read the images in Figure 6(d). For each co-clusters (I, J) discovered $(I \subseteq G, J \subseteq T)$, it is possible to pose a hypothesis of the form "genes $g \in I$ are correlated with traits $t \in J$", which can then be tested by further experimental studies.

Supporting evidence from the literature: Our data showed that trait "survival" is clustered with genes *TGFB1* or *TGFB2* and *CD1a* multiple times in co-clusters #37, #38, #42 and #43. TGF-β (transforming growth factor-β) is a multifunctional peptide that has both growth-inhibitory and growth-stimulating properties [8]. Its combined effects with other growth factors or inhibitors have been shown to play a central role in the control of growth, differentiation, and morphogenesis of normal and malignant cells. For example, TGF-β is required for efficient *in vitro* generation of *dendritic cells* (DC) from CD34+ progenitor cells [11]. However, it also inhibits cell proliferation and survival mediated by Flt3 (Fms-like tyrosine kinase-3) signaling pathway [7], [14]. Given that a mutated and constitutively active form of Flt3 is detected in 30-35 % of AML cases and the patients with Flt3 mutations tend to have a poor prognosis [15], it is interesting to note that "survival" trait is positively correlated with the expression of *TGFB1* and *TGFB2* which abrogate the effects of Flt3.

In addition, our data indicate that "survival" is associated with the expression of *CD1a*, a cell surface marker for mature DC. Previous studies reported that, when cultured in the presence of GM-CSF (granulocyte-macrophage colony-

TABLE I

The input parameters used for the experiment and some statistics obtained from the output co-clusters.

Parameters/statistic	Value/reference
A: gene expression matrix	[3]
B: trait matrix	[3]
τ: parameter for Algorithm 1	2.5
π: p-value cutoff	0.05
Total number of co-clusters found	43
Average size of co-clusters ($\#genes, \#traits$)	$(143, 3)$

[2] This correlation matrix has 14 columns instead of 15, because the traits "Status" and "Overall survival" were merged into one for the convenience in survival analysis.

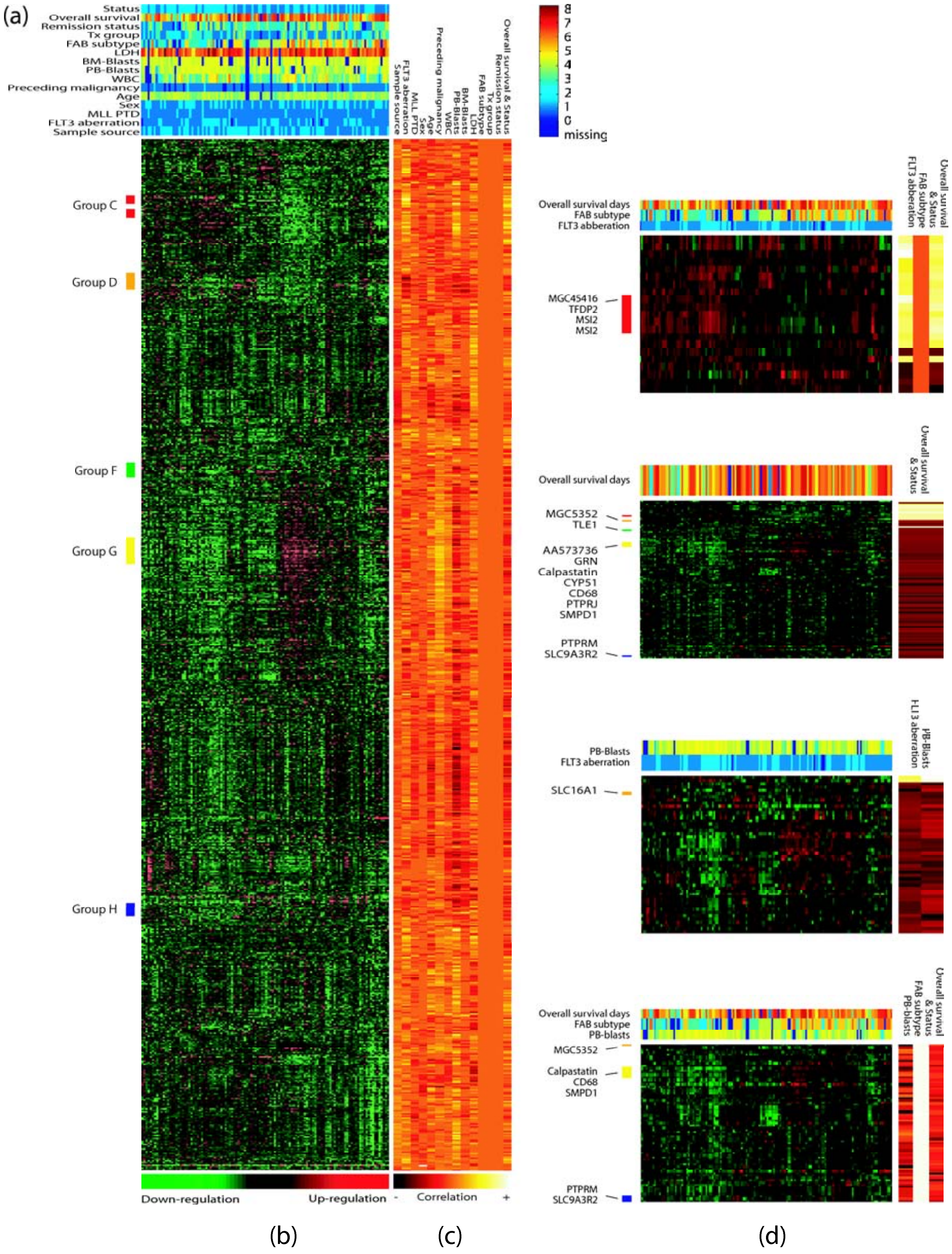


Fig. 6. Data from an adult acute myeloid leukemia (AML) study [3]. (a) The heat map of the clinical trait matrix, in which each row corresponds to a trait and each column a sample. The legend of the heat map can also be found. (b) The heat map of the gene expression matrix with 6283 genes (rows) and 119 samples (columns). The vertical colored bars are to indicate the gene groups C–H used in the original study [3]. (c) The heat map of the correlation matrix. (d) Some co-clusters found by the proposed method. Refer to Figure 5 for further details. [We suggest the reader to print this page in color.]

TABLE II

SOME GENES CONTAINED IN CO-CLUSTER $\#15$ WITH DESCRIPTIONS.

Gene	Description
MALT1	MAL tissue lymphoma translocation gene 1
NFIL3	Nuclear factor, interleukin 3 regulated
APOH	Apolipoprotein H (beta-2-glycoprotein I)
FCGRT	Fc fragment of IgG, receptor, transporter, alpha
SERPINA1	Serine (or cysteine) proteinase inhibitor
C1QA	Complement component 1, q subcomponent, alpha
OAS2	2'-5'-oligoadenylate synthetase 2
ITK	IL2-inducible T-cell kinase
CD3G	CD3G antigen, gamma polypeptide (TiT3 complex)

TABLE III

AN ENRICHED GO TERM OBTAINED BY GO::TERMFINDER [2].

Item	Value
GO term	Defense response
Cluster frequency	9 out of 54 genes (16.7%)
Genome frequency of use	1209 out of 23531 genes (5.1%)
Corrected p-value	0.0359
False Discovery Rate (FDR)	3.00%
False Positives	0.06

stimulating factor), TNF-α (tumor necrosis factor-α), and IL-4 (interleukin-4), AML cells were induced to differentiate into DC and up-regulated the expression of *CD1a* and co-stimulatory molecules such as CD80 and CD86 [4], [6]. Since DC are the most potent *antigen-presenting cells* (APC) in the immune system, the *CD1a*-positive leukemic DC might function as APC bearing leukemic antigen, prime cytotoxic T cells and generate a strong anti-leukemic immune response. This may explain why *CD1a* is often clustered with trait "survival" in our data.

Validation with GO: To determine whether any GO terms annotate genes in a specified co-clusters at a frequency greater than that would be expected by chance, a p-value is calculated in this particular setting using the hypergeometric distribution: $p\text{-value} = 1 - \sum_{i=0}^{k-1} \frac{\binom{M}{i}\binom{N-M}{n-i}}{\binom{N}{i}}$, where N is the total number of genes in the background distribution, M is the number of genes (within that distribution) that are annotated to the node of interest, n is the size of the list of genes in a co-cluster of interest, and k is the number of genes within that list which are annotated to the node. The background distribution is all the genes within a given GO annotation file.

We used the tool GO::TermFinder [2] for the calculation of p-values as well as the multiple hypothesis correction [5] of the calculated p-values. Using this tool, we found from *Process Ontology* the terms that annotate genes in co-cluster $\#15$ with p-values less than a threshold of 0.05. (The descriptions of the genes included in co-cluster $\#15$ are listed in Table II.) These terms include "defense response," "immune response," "acute-phase response," "antigen presentation," and "T-cell activation." Further analysis of each enriched term is also possible, and as an example, Table III shows more statistics for the term "defense response" (GO:0006952).

IV. CONCLUSIONS

We investigated the problem of finding co-clusters of genes and clinical traits using microarray data and clinical parameter information. An intermediate data matrix called correlation matrix was computed by means of a statistical method. We then modeled a co-cluster by a submatrix of the correlation matrix with some semantics and aimed at finding statistically significant co-clusters. We proposed a co-clustering algorithm, tested it with the AML data set and discovered some number of co-clusters. The validation with GO as well as the literature suggests that some co-clusters found be biologically meaningful.

REFERENCES

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Data Structures and Algorithms*. Reading, Massachusetts: Addison-Wesley, 1983.

[2] E. I. Boyle *et al.*, "GO::TermFinder," *Bioinformatics*, vol. 20, no. 18, pp. 3710–3715, December 2004.

[3] L. Bullinger *et al.*, "Use of gene-expression profiling to identify prognostic subclasses in adult acute myeloid leukemia," *N Engl J Med.*, vol. 350, no. 16, pp. 1605–1616, April 2004.

[4] A. Cignetti, E. Bryant, B. Allione, A. Vitale, R. Foa, and M. A. Cheever, "CD34(+) acute myeloid and lymphoid leukemic blasts can be induced to differentiate into dendritic cells," *Blood*, vol. 94, pp. 2048–2055, 1999.

[5] S. Draghici, *Data Analysis Tools for DNA Microarrays*. Florida: Chapman & Hall/CRC, 2003.

[6] B. D. Harrison, J. A. Adams, M. Briggs, M. L. Brereton, and J. A. L. Yin, "Stimulation of autologous proliferative and cytotoxic T-cell responses by "leukemic dendritic cells" derived from blast cells in acute myeloid leukemia," *Blood*, vol. 979, pp. 2764–2771, 2001.

[7] S. E. Jacobsen, O. P. Veiby, J. Myklebust, C. Okkenhaug, and S. D. Lyman, "Ability of Flt3 ligand to stimulate the in vitro growth of primitive murine hematopoietic progenitors is potently and directly inhibited by transforming growth factor-beta and tumor necrosis factor-alpha," *Blood*, vol. 87, pp. 5016–5026, 1996.

[8] J. Keski-Oja, E. B. Leof, R. M. Lyons, R. J. Coffey Jr, and H. L. Moses, "Transforming growth factors and control of neoplastic cell growth," *J Cell Biochem*, vol. 33, pp. 95–107, 1987.

[9] I. S. Kohane, A. T. Kho, and A. J. Butte, *Microarrays for an Integrative Genomics*. Cambridge, Massachusetts: The MIT Press, 2003.

[10] S. C. Madeira and A. L. Oliveira, "Biclustering algorithms for biological data analysis: A survey," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 1, no. 1, pp. 24–45, 2004.

[11] E. Riedl, H. Strobl, O. Majdic, and W. Knapp, "TGF-beta 1 promotes in vitro generation of dendritic cells by protecting progenitor cells from apoptosis," *J Immunol*, vol. 158, pp. 1591–1597, 1997.

[12] B. Rosner, *Fundamentals of Biostatistics*, 5th ed. Pacific Grove, California: Duxbury, 2000.

[13] V. G. Tusher, R. Tibshirani, and G. Chu, "Significance analysis of microarrays applied to the ionizing radiation response," *Proc. Natl. Acad. Sci USA*, vol. 98, no. 9, pp. 5116–5121, April 2001.

[14] O. P. Veiby, F. W. Jacobsen, L. Cui, S. D. Lyman, and S. E. Jacobsen, "The Flt3 ligand promotes the survival of primitive hemopoietic progenitor cells with myeloid as well as b lymphoid potential. Suppression of apoptosis and counteraction by TNF-alpha and TGF-beta," *J Immunol*, vol. 157, pp. 2953–2960, 1996.

[15] E. Weisberg *et al.*, "Inhibition of mutant Flt3 receptors in leukemia cells by the small molecule tyrosine kinase inhibitor PKC412," *Cancer Cell*, vol. 1, pp. 433–443, 2002.

[16] M. West *et al.*, "Predicting the clinical status of human breast cancer by using gene expression profiles," *Proc Natl Acad Sci USA*, vol. 98, no. 20, pp. 11 462–11 467, September 2001.

[17] A. R. Whitney *et al.*, "Individuality and variation in gene expression patterns in human blood," *Proc Natl Acad Sci USA*, vol. 100, no. 4, pp. 1896–1901, February 2003.

[18] S. Yoon, C. Nardini, L. Benini, and G. De Micheli, "Discovering coherent biclusters from gene expression data using zero-suppressed binary decision diagrams," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, to appear.