# Discovery of Error-tolerant Biclusters from Noisy Gene Expression Data

Rohit Gupta
Dept of Comp Sc and Engg
Univ of Minnesota, Twin Cities
Minneapolis, MN USA
rohit@cs.umn.edu

Navneet Rao
Dept of Comp Sc and Engg
Univ of Minnesota, Twin Cities
Minneapolis, MN USA
nrao@cs.umn.edu

Vipin Kumar
Dept of Comp Sc and Engg
Univ of Minnesota, Twin Cities
Minneapolis, MN USA
kumar@cs.umn.edu

## ABSTRACT

An important analysis performed on microarray gene-expression data is to discover biclusters, which denote groups of genes that are coherently expressed for a subset of conditions. Various biclustering algorithms have been proposed to find different types of biclusters from these real-valued gene-expression data sets. However, these algorithms suffer from several limitations such as inability to explicitly handle errors/noise in the data; difficulty in discovering small bicliusters due to their top-down approach; inability of some of the approaches to find overlapping biclusters, which is crucial as many genes participate in multiple biological processes. Association pattern mining also produce biclusters as their result and can naturally address some of these limitations. However, traditional association mining only finds exact biclusters, which limits its applicability in real-life data sets where the biclusters may be fragmented due to random noise/errors. Moreover, as they only work with binary or boolean attributes, their application on gene-expression data require transforming real-valued attributes to binary attributes, which often results in loss of information. Many past approaches have tried to address the issue of noise and handling real-valued attributes independently but there is no systematic approach that addresses both of these issues together. In this paper, we first propose a novel error-tolerant biclustering model, '*ET-bicluster*', and then propose a bottom-up heuristic-based mining algorithm to sequentially discover error-tolerant biclusters directly from real-valued gene-expression data. The efficacy of our proposed approach is illustrated in the context of two biological problems: discovery of functional modules and discovery of biomarkers. For the first problem, we used two real-valued *S.Cerevisiae* microarray gene-expression data sets and evaluate the biclusters obtained in terms of their functional coherence as evaluated using the GO-based functional enrichment analysis. The statistical significance of the discovered error-tolerant biclusters as estimated by using two randomization tests, reveal that they are indeed biologically meaningful and statistically significant. For the second problem of biomarker discovery, we used four real-valued *Breast Cancer* microarray gene-expression data sets and evaluate the biomarkers obtained using MSigDB gene sets. We compare our results obtained from both the problems, with a recent approach *RAP* and clearly demonstrate the importance of incorporating noise/errors in discovering coherent groups of genes from gene-expression data.

## 1. INTRODUCTION

Recent technical advancements in DNA microarray technologies have led to the availability of large-scale gene expression data. These data sets can be represented as a matrix $G$ with genes as rows and different experimental conditions as columns, where $G_{ij}$ denotes the expression value of gene $i$ for an experimental condition $j$. An important research problem of gene-expression analysis is to discover submatrix patterns or biclusters in $G$. These biclusters are essentially subsets of genes that show coherent values across a subset of experimental conditions. However, coherence among the data values can be defined in various ways. For instance, Madeira et al [22] classify biclusters into the following four different categories based on the definition of coherence: (i) biclusters with constant values, (ii) biclusters with constant rows or columns, (iii) biclusters with coherent values, and (iv) biclusters with coherent evolutions.

Many approaches ([4, 9, 12, 22, 32, 3, 27]) have been proposed to discover biclusters from gene-expression data. Different biclustering algorithms have been designed to discover different types of biclusters. For instance, coclustering [12] and SAMBA [32] find constant value biclusters, Cheng and Church (CC) [9] find constant row biclusters and OPSM [3] find coherent evolutions biclusters. Though there are differences in biclustering algorithms in terms of the type of bicluster they discover, there are some common issues with these algorithms in general. First critical issue with all of these biclustering algorithms is that they are oblivious to noise/errors in the data and require all values in the discovered bicluster to be coherent. This limits the discovery of valid biclusters that are fragmented due to random noise in the data. Second issue with at least some of the biclustering algorithms is their inability to find overlapping biclusters. For instance, coclustering is designed to only look for disjoint biclusters and Cheng and Church's approach, which masks the identified bicluster with random values in each iteration, also finds it hard to discover overlapping biclusters. Third, most of the algorithms are top-down greedy schemes that start with all rows and columns, and then iteratively eliminate them to optimize the objective function. This generally results in large biclusters, which although are useful, do not provide information about the small biological functional classes. Finally, all the biclustering algorithms employ heuristics and are unable to search the space of all possible biclusters exhaustively.

Association pattern mining can naturally address some of the issues faced by biclustering algorithms i.e, finding over-

*BIOKDD'10,* July 25, 2010, Washington DC, USA.

lapping biclusters and performing an exhaustive search. However, there are two major drawbacks of traditional association mining algorithms. First, these algorithms use a strict definition of support that requires every item (gene) in a pattern (bicluster) to occur in each supporting transaction (experimental condition). This limits the recovery of patterns from noisy real-life data sets as patterns are fragmented due to random noise and other errors in the data. Second, since traditional association mining was originally developed for market basket data, it only works with binary or boolean attributes. Hence it's application to data sets with continuous or categorical attributes requires transforming them into binary attributes, which can be performed by using discretization [30, 28, 13], binarization [2, 11, 10, 23] or by using rank-based transformation [6]. In each case, there is a loss of information and associations obtained does not reflect relationships among the original real-valued attributes, rather reflect relationships among the binned independent values [16].

Efforts have been made to independently address the two issues mentioned above and to the best of our knowledge, no prior work has addressed both the issues together. For example, various methods [34, 33, 21, 20, 29, 7, 5, 8, 25, 26] have been proposed in the last decade to discover approximate frequent patterns (often called error-tolerant itemsets (ETIs)). These algorithms allow patterns in which a specified fraction of the items can be missing - see [14] for a comprehensive review of many of these algorithms. As the conventional support (i.e the number of transactions supporting the pattern) is not anti-monotonic for error-tolerant patterns, most of these algorithms resort to heuristics to discover these patterns. Moreover, all of these algorithms are developed only for binary data.

Another recent approach [24] addressed the second issue and extended association pattern mining for real-valued data. The extended framework is referred to as *RAP* (Range Support Pattern). A novel *range* and *range support* measures were proposed, which ensure that the values of the items constituting a meaningful pattern are coherent and occurs in a substantial fraction of transactions. This approach reduces the loss of information as incurred by discretization- and binarization-based approaches, as well as enables the exhaustive discovery of patterns. One of the major advantages of using an approach such as *RAP*, which adopts a very different pattern discovery algorithm as compared to more traditional biclustering algorithms such as *CC* or *ISA*, is the ability to find smaller or completely novel biclusters. Several examples shown in [24] illustrated that *RAP* can discover some biologically relevant smaller biclusters, which are either completely missed by biclustering approaches such as *CC* or *ISA*, or are found embedded in larger biclusters. In either case, they are not able to enrich the smaller functional classes as *RAP* biclusters do. Despite these advantages, *RAP* framework does not directly address the issue of noise and errors in the data.

As it has been independently shown that both issues, handling real-valued atributes and noise, are critical and affect the results of the mining process, it is important to address them together. In this paper, we propose a novel extension of association pattern mining to discover error-tolerant biclusters (or patterns) directly from real-valued gene-expression data. We refer to this approach as '*ET-bicluster*' for error-tolerant bicluster. This is a challenging task because the conventional support measure is not anti-monotonic for the error-tolerant patterns and therefore limits the exhaustive search of all possible patterns. Moreover the set of values constituting the pattern in the real-valued data is different than the binary data case. Therefore, instead of using the traditional support measure, we used the *range* and *RangeSupport* measures as proposed in [24] to ensure the coherence of values and for computing the contribution from supporting transactions. *RangeSupport* is anti-monotonic for both dense and error-tolerant patterns, however, *range* is not anti-monotonic for error-tolerant patterns. Due to this, exhaustive search is not guaranteed, however it is important to note that the proposed *ET-bicluster* framework still, by design, finds more number of patterns (biclusters) than it's counterpart *RAP*. Therefore using *range* as a heuristic measure, we describe a bottom-up pattern mining algorithm, which sequentially generates error-tolerant biclusters that satisfy the user-defined constraints, direcly from the real-valued data.

To demonstrate the efficacy of our proposed *ET-bicluster* approach, we compare it's performance with *RAP* in the context of two biological problems: (a) functional module discovery, and (b) biomarker discovery. Since both *ET-bicluster* and *RAP* use same pattern mining framework, comparing them helps to quantify the impact of noise and errors in the data on the discovery of coherent groups of genes in an unbiased way.

For the first problem of functional module discovery, we used real-valued *S.cerevisiae* microarray gene-expression data sets and discovered biclusters using both *ET-bicluster* and *RAP* algorithm. To illustrate the importance of directly incorporating data noise/errors in biclusters, we compared the error-tolerant biclusters and *RAP* biclusters using gene ontology (GO) based biological processes annotation hierarchy [1] as the base biological knowledge. Specifically, for each {bicluster, GO term} pair, we computed a p-value using a hypergeometric distribution, which denotes the random probability of annotating this bicluster with the given GO term. For the second problem of biomarker discovery, we combined four real-valued case-control Breast Cancer gene-expression data sets, and discovered discriminative biclusters (or biomarkers) from the combined data set using both *ET-bicluster* and *RAP*. Again, to illustrate the importance of explicitly incorporating noise/errors in the data, we compared the biomarkers based on their enrichment scores computed using MSiGDB gene sets [31]. MSigDB gene sets are chosen as the base biological knowledge in this case because they include several manually annotated cancer gene sets. The results obtained for both the functional module discovery and biomarker discovery problem clearly demonstrate that error-tolerant biclusters are not only bigger than *RAP* biclusters but are also biologically meaningful. Using randomization tests, we further demonstrated that error-tolerant biclusters are indeed statistically significant and are neither obtained by random chance nor capture random structures in the data. Overall, the results presented for both the biological problems strongly suggest that our proposed *ET-bicluster* approach is a promising method for the analysis of real-valued gene-expression data sets.

**Contributions:**

- We proposed a novel association pattern mining based approach to discover error-tolerant biclusters from noisy real-valued gene-expression data.
- Our work highlights the importance of tolerating error(s) in the biclusters in order to capture the true underlying structure in the data. This is demonstrated

using two case studies: functional module discovery and biomarker discovery. Using various real-valued gene expression data sets, we illustrated that our proposed algorithm *ET-bicluster* can discover additional and bigger biologically relevant biclusters as compared to *RAP*.

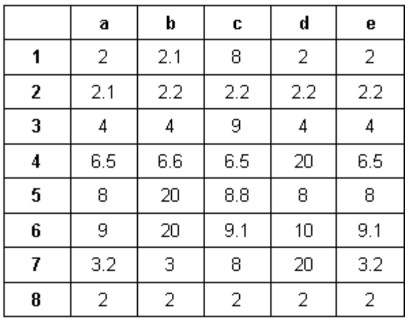
- We used two randomization techniques to compute the empirical p-value of all the discovered error-tolerant biclusters and demonstrated that they are statistically significant and it is highly unlikely to have obtained them by random chance.

**Organization:** The rest of the paper is organized as follows. In Section 2, we discuss our proposed algorithm *ET-bicluster*. Section 3 details the experimental methodology for evaluating the error-tolerant biclusters and their comparison with *RAP* biclusters, and the results obtained. We present a summary of the findings in section 4 followed by a discussion on limitations and future work in section 5.

## 2. ERROR-TOLERANT BICLUSTER MODEL FOR REAL-VALUED DATA

As shown in [22], there can be different types of biclusters one can define on a real-valued data based on different measures of coherence among data values. In this paper, we focus on constant row/column biclusters, as they are well suited for the *ET-bicluster* framework and also considered as one of the promising ways to capture functional coherence from the microarray data sets [9]. However, discovering error-tolerant biclusters directly from real-valued data is a challenging task as several issues arise either due to handling of real-valued attributes or due to relaxing the bicluster requirements to incorporate noise/errors in the data. Specifically, following three issues need to be discussed before we present the algorithm.

(a) **Bicluster Composition:** Unlike the case of binary data where collection of 1$s$ was defined as a bicluster, in the case of real-valued data, similar values across a set of rows constitute a bicluster. These values can be any values in the set $\mathbb{R}$ and athough similar across rows, they can be different for different rows. The errors in the biclusters defined on real-valued attributes are introduced in a way similar to the binary case. However, like binary case in which all non-error entries are same (1$s$), in real-valued case, imposing such a requirement would be very harsh. Therefore, a measure is needed to check the coherence among the gene-expression values. For this purpose, we use the *range* measure, which checks for each transaction if the relative range of the gene-expression values in a bicluster, given as $(max_{val} - min_{val})/min_{val}$, is within a pre-specified threshold $\alpha$. Furthermore, the contribution of each supporting transaction is measured as the minimum of the values taken by any of the genes in the bicluster in that transaction. Overall, to measure the strength of the bicluster, we use the *RangeSupport* (*RS*) measure [24], which sums up the contribution of each supporting transaction. This is similar to the *support* measure that is generally used in association pattern mining for binary data, however unlike binary case, each supporting transaction may not contribute equally for *RangeSupport* of a bicluster in real-valued data. The *range* and *RangeSupport* measures in combination capture the requirement that expression values of the genes in a bicluster are coherent for several transactions, and hence can be used to mine interesting biclusters from the real-valued data. Note that although both measures *range* and *RangeSupport* are anti-monotonic for exact biclusters, *range* is not anti-monotonic for error-tolerant biclusters. Due to this reason, *ET-bicluster* does not exhaustively find all error-tolerant biclusters, but it is noteworthy that it still subsume all biclusters found by *RAP* and can even find biclusters that are fragmented due to noise/errors in the data. One the other hand, as *RAP* is oblivious to errors/noise in the data, it either completely miss these fragmented but valid biclusters or find them as separate parts.



|  | a | b | c | d | e |
|---|---|---|---|---|---|
| **1** | 2 | 2.1 | 8 | 2 | 2 |
| **2** | 2.1 | 2.2 | 2.2 | 2.2 | 2.2 |
| **3** | 4 | 4 | 9 | 4 | 4 |
| **4** | 6.5 | 6.6 | 6.5 | 20 | 6.5 |
| **5** | 8 | 20 | 8.8 | 8 | 8 |
| **6** | 9 | 20 | 9.1 | 10 | 9.1 |
| **7** | 3.2 | 3 | 8 | 20 | 3.2 |
| **8** | 2 | 2 | 2 | 2 | 2 |

**Figure 1: A sample error-tolerant bicluster**

(b) **Positive/Negative Values:** Unlike binary data, real-valued microarray data has both positive and negative values. In this case, it is important to consider the sign of the value to discover meaningful biclusters. Similar to [24], we address this problem by enforcing that a transaction can only be termed as the supporting transaction of a bicluster if for this transaction, the expression values of all the genes in the bicluster are of the same sign. This also help make biological interpretability easier as the sign enforcement would entail finding only those biclusters in which all the genes are either up-regulated or down-regulated for a given experimental condition. However note that the same genes can be up-regulated for one experimental condition and down-regulated for another.

(c) **Error/Non-error Values:** In binary case, 1 is always a non-error value and 0 an error value. This notion is no more valid for the real-valued data case. For example, consider an error-tolerant bicluster shown in figure 1 with 5 genes (a, b, c, d, e) and 8 experimental conditions $(1 \ldots 8)$. For the 1st condition, 8 is an error value, for the 3rd condition 9 is an error value, and for the 5th condition, 20 is an error value. Similarly, non-error values can change for each transaction. Thus, it becomes important to keep track of error and non-error values while mining for biclusters in the real-valued data.

Now, with the understanding of specific challenges and potential ways to address them, we now give the formal definition of error-tolerant biclusters for a real-valued data.

### 2.1 Definition of Error-tolerant Biclusters

Intuitively, a bicluster $B$ is said to be an error-tolerant bicluster if the following two general conditions are satisfied:

- *RangeSupport* of bicluster $B$ should be more than the user-defined threshold, $RS$.
- All supporting transactions of bicluster $B$ should have mostly non-error values i.e. values should be generally coherent (governed by a user-defined parameter $\epsilon$ for maximum number of permissible errors).

*Definition 1.* Let $D$ be a real-valued gene-expression data, $RS$ be the *RangeSupport* threshold, $E$ be a function that takes a set of real values as input and returns the number of errors in them using *range* criteria, and let error threshold be $\epsilon \in (0, 1]$. A bicluster $B$ (with genes $G$) is an error-tolerant bicluster $ET\text{-}bicluster(\epsilon)$ in the real-valued attribute domain, if there exists a set of transactions $T \in D$ such that the following two conditions hold:

$$Range\ Support(B) \geq RS \tag{1}$$

$$\forall t \in T, E(D_{t,G}) \leq \epsilon \cdot |G| \tag{2}$$

Thus according to the definition, fraction of errors in each supporting transaction of the bicluster should not exceed $\epsilon$.

## 2.2 Algorithm to Discover Error-tolerant Biclusters from Real-valued Data

Starting with singletons, the *ET-bicluster* algorithm sequentially generates (k+1)-level biclusters from k-level biclusters. At k = 1, genes that satisfy the *RangeSupport* (computed as the summation of absolute values for all transactions) criterion are valid singletons. Generally speaking, any (k+1)-level bicluster is a valid bicluster if it satisfies the *RangeSupport* criterion and each supporting transaction of the bicluster has at most $\epsilon$ fraction of errors.

*ET-bicluster* algorithm generates (k+1)-level biclusters from k-level biclusters by one of the two steps: error extension or non-error extension. Specifically, if $\lfloor (k+1) \cdot \epsilon \rfloor = \lfloor k \cdot \epsilon \rfloor$, it's a non-error extension step (no more errors values are permitted) or else it will be a error-extension step (one additional error value is permitted). We used two lemmas proved in [20] to efficiently perform these extension steps. In non-error extension step, for each (k+1)-level bicluster, *range* criteria is only checked for the intersection of supporting transactions of all its k-level biclusters. On the other hand, in the error-extension step, *range* criteria is checked for the union of supporting transactions of all its k-level biclusters.

Checking the range criterion to ensure the coherence of values depends on the number of permissible errors at a particular bicluster-level ($k \cdot \epsilon$). For instance, if the permissible number of errors is 1, then *range* criterion for a given transaction is computed as follows. First, for each transaction, all the expression values in a bicluster are sorted and then the range criterion is checked in usual manner by either discarding the minimum value or the maximum value. If the *range* criterion is satisfied in any of the two cases, transaction is classified as the supporting transaction for that bicluster. If for instance, number of permissible errors are 2 at any bicluster-level, we check the *range* criterion for three cases: discarding the 2 minimum values; discarding the 2 maximum values; or discarding 1 minimum value and 1 maximum value. Again, if any of the case satisfies the *range* criterion, transaction is classified as a supporting transaction. Similarly, we exhaustively make all cases when number of permissible errors are more than 2. However, note that with $\epsilon = 0.25$ (value considered in this paper) and itemset size even as big as 12, we only need to make these cases for 3 permissible errors.

## 2.3 An Example

Considering a sample real-valued data with 5 genes (a, b, c, d, and e) and 8 experimental conditions (1 through 8) as shown in figure 1, below we demonstrate the steps of *ET-bicluster* algorithm. Input parameters: Range Support threshold = 5; $\alpha = 0.5; \epsilon = 0.25$

**Step 1:** $k = 1$. As range support for each gene is greater than 5, all the genes are returned as valid singletons.

**Step 2:** $k = 2$. Since $\lfloor k * \epsilon \rfloor = \lfloor k - 1 \rfloor * \epsilon$, this is a non-error extension step. Consider for example bicluster *ab*, for $\alpha = 0.5$, it's supporting transactions are {1,2,3,4,7,8}. To illustrate, while transaction 1 satisfies the range criteria (i.e. $2.1 - 2 \leq 0.5 * 2$) and hence is valid, transaction 5 is not valid since $20 - 8 > 0.5 * 8$. Now, *RangeSupport* of bicluster *ab* is given as the sum of the contributions from each supporting transaction i.e. $RS(ab) = 2 + 2.1 + 4 + 6.5 + 3 + 2 = 19.6$. Since, $RS(ab) > 5$, *ab* is a valid bicluster. Similarly, biclusters *ac*, *ad*, *ae*, *bc*, *bd*, *be*, *cd*, *ce*, *de* are all valid biclusters.

**Step 3:** $k = 3$. Again since $\lfloor k * \epsilon \rfloor = \lfloor k - 1 \rfloor * \epsilon$, this is a non-error extension step. Consider for example, bicluster *abc*, *range* criterion is checked for intersection of supporting transactions of biclusters *ab*, *bc* and *ac* and hence supporting transactions are identified as {2,4,8}. Now, since $RS(abc) = 10.6$, which is greater than the thereshold 5, *abc* is a valid bicluster. Similarly, *abd*, *abe*, *bce*, *bde* and *cde* are all valid biclusters.

**Step 4:** $k = 4$. In this case, since $\lfloor k * \epsilon \rfloor \neq \lfloor k - 1 \rfloor * \epsilon$, this is an error extension step. The number of permissible errors at this level is $k * \epsilon_r = 4 * 0.25 = 1$. Consider for example, bicluster *abcd*, *range* criterion is checked for the union of supporting transactions of all its level-3 biclusters subsets. Hence, we get {1,2,3,4,5,6,8} as the set of supporting transactions. For illustration, take an example of transaction 1. As only one error value is permitted, range criterion is checked as follows: $(((2^{nd}max - min)/min) = (2.1 - 2)/2 = 0.05 < \alpha(0.5))$. Therefore, this is a supporting transaction. On the other hand, transaction 7, even after discarding one error value does not satisfy the range criterion for bicluster *abcd*. Also $RS(abcd) = 33.6$, hence *abcd* is a valid bicluster. Similarly, *abce* is also a valid bicluster.

**Step 5:** $k = 5$. Since, $\lfloor k * \epsilon \rfloor = \lfloor k - 1 \rfloor * \epsilon$, this is a non-error extension step. A bicluster *abcde* will be generated with set of supporting transactions as {1,2,3,4,5,6,8}. Now since $RS(abcde) = 33.6$, abcde is a valid bicluster.

It is important to note that since *RAP* does not explicitly handle errors/noise in the data, it cannot discover the bicluster *abcde*, which is fragmented due to errors.

# 3. EXPERIMENTAL RESULTS AND DISCUSSION

We implemented our proposed association pattern mining approach '*ET-bicluster*' in C++. In this paper, we only compare our proposed approach with *RAP*, as *RAP* has already been shown to outperform biclustering approaches such as ISA and Cheng and Church, especially for finding small biclusters. Also, as mentioned in [24], transformation of data from real-valued attributes to binary attributes leads to loss of distinction between various types of biclusters (or patterns). Therefore, as the focus of this study is to discover constant row biclusters, binarization of real-valued gene-expression data is not meaningful. For this reason, we only show results on real-valued data sets. Further, in order to compare the performance of '*ET-bicluster*' and *RAP* in discovering coherent groups of genes, we considered two biological problems: discovery of functional modules (finding coherent gene groups) and discovery of biomarkers (finding coherent gene groups that are discriminative of the two classes of patients: cases and controls).

**Selecting Top Biclusters** As association mining based approach generally produces a large number of biclusters that often have substantial overlap with each other, this redundancy in biclusters may bias the evaluation. Hence, we used a commonly adopted selection methodology similar to the one proposed by [27] to select upto 500 top biclusters. However, because error-tolerant biclusters generally have a large set of supporting experimental conditions, even biclusters with high overlap in gene dimension may get selected in the top 500 biclusters. To avoid this situation, we computed the size of a bicluster by the number of genes ($|genes|$) in it, not by $|genes| \times |conditions|$ in it. Therefore, starting with the largest bicluster (only in terms of the number of genes in it), we greedily select upto 500 biclusters such that the overlap among any of the selected biclusters is not more than 25%. In case of a tie between the size of biclusters, bicluster with lower Mean Square Error (MSE) value [9] is selected. Please note that MSE of a bicluster is computed by discarding the error values in it, since *ET-bicluster* is meant to look for error-tolerant patterns.

## 3.1 Case Study 1 - Discovery of Functional Modules

We used the following two real-valued *S.cerevisiae* microarray gene-expression data sets for the discovery of functional modules:

- Hughes et al's data set [18]: This data set contains a compendium of expression profiles corresponding to 300 diverse mutations and chemical treatments in S. cerevisiae and was compiled to study the functions of yeast genes on a large scale. The overall dimensions of this data set are 6316 genes x 300 conditions, with values ($log_{10}$ ratio of expression values observed for experimental condition and control condition) in the range [-2,2].
- Mega Yeast data set [19]: This data set contains 501 yeast microarray experiments, including stress responses, cell cycle, sporulation, etc. The overall dimensions of this data set are 6447 genes x 501 conditions, with values in the range [-12,12].

**Functional Enrichment Analysis** Since the discovered biclusters represent groups of genes that are expected to co-express with each other, we evaluated all the biclusters discovered in terms of their functional coherence using the biological processes annotation hierarchy of Gene Ontology [1]. A p-value using a hypergeometric probability distribution is computed for each combination of bicluster and biological process GO term to determine if the discovered biclusters are statistically significant. The p-value computed for a pair of bicluster (denoted by $b$) and GO term (denoted by $t$) denotes the random probability of annotating a bicluster of size same as $b$ with the same GO term $t$.

To compare error-tolerant biclusters and *RAP* biclusters in an unbiased fashion, we used the same 2652 biological processes GO terms (or classes), all of which contain at least 1 and at most 500 genes from *S.cerevisiae*. Furthermore, as only 4684 genes are annotated with either one or more of these 2652 classes, we restricted our analysis to a subset of data sets comprising of 4684 $genes \times 501$ $conditions$ and 4684 $genes \times 300$ $conditions$ for mega yeast and Hughes's et al's gene-expression data sets respectively.

### 3.1.1 Quantitative Analysis of Biclusters

Table 1 provides a general overview of all the biclusters obtained by *ET-bicluster* and *RAP* algorithm on mega yeast and Hughes et al's real-valued gene-expression data sets using various parameter settings. Parameter controlling error-tolerance ($\epsilon$) was set to 0.25 in all the runs for *ET-bicluster*. It is important to note that number of error-tolerant biclusters is substantially larger than the number of *RAP* biclusters. Therefore, for a specific $range(\alpha)$ value and user-defined $RangeSupport$ threshold, if *ET-bicluster* algorithm was not able to finish in a reasonable amount of time and memory with $\epsilon = 0.25$, we first obtain exact biclusters (no error-tolerance) by setting $\epsilon$ to 0 and then increase the $RangeSupport$ to obtain error-tolerant biclusters by setting $\epsilon$ to 0.25. The final resulting set of biclusters is obtained by merging these exact and error-tolerant biclusters. Following are some of the general observations:

**Number of Biclusters:** It can be clearly seen from table 1 that introducing an error-tolerance of 25% substantially increased the total number of biclusters. For example, number of total error-tolerant biclusters obtained on mega yeast data is approximately 5-times (for $\alpha = 0.5$) and 3-times (for $\alpha = 0.3$) the number of *RAP* biclusters for corresponding $\alpha$ values. Similarly, for Hughes et al's data set, number of error-tolerant biclusters is approximately 3-times the number of *RAP* biclusters for both the $\alpha$ values considered ($\alpha = 0.8$ and $\alpha = 0.5$).

**Size of Biclusters:** Another important observation one can make from the results shown in table 1 is that the size of error-tolerant biclusters is more than *RAP* biclusters. This is expected as *RAP* can only find exact biclusters (with no error-tolerance) and hence valid biclusters that are fragmented due to random noise and errors in the data, are either found as separate biclusters or completely missed. On the other hand, because *ET-bicluster* algorithm explicitly handles noise and errors in the data, it can potentially find larger biclusters by stitching together the fragmented parts or can even find new biclusters that were missed by *RAP*. This might have a significant impact on the functional enrichment analysis as *ET-bicluster* algorithm can potentially discover biclusters that have higher overlap with the considered GO biological processes classes. We discuss this further in the next section.

**Coverage of Genes and Relationships Among Them:** As can be noted from table 1, the number of genes covered by *ET-bicluster* and *RAP* algorithm is same at least if we consider all biclusters. This is because the starting set of genes ('singletons') are same for both the algorithms. In fact, if the error-tolerance, $\epsilon$ is 0.25 for example, then singletons, pairs (level-2 bicluster) and even triplets (level-3 bicluster) will be identical for *ET-bicluster* and *RAP*. However note that the number of level-4 biclusters generated by *ET-bicluster* is more than those generated by *RAP*. This is due to the fact that *ET-bicluster* algorithm, owing to its relaxed error-tolerance criterion, can generate more combinations of genes than *RAP*. Therefore in other words, even if the total genes covered by both the algorithms are same, *ET-bicluster* algorithm can find more relationships among them.

As mentioned above and shown in table 1, since *ET-bicluster* algorithm, as compared *RAP*, can potentially find newer and larger biclusters and hence more relationships among genes, an important question to address is: whether these larger and new biclusters are biologically meaningful? One promising way to answer this question is through functional enrichment analysis and below we discuss these results.

### 3.1.2 Functional Enrichment using GO Biological Processes

As mentioned earlier, a p-value for each of the (bicluster, GO term) pair is computed for the selected top 500 biclusters using the 2652 biological processes GO terms considered in this study. To demonstrate how well error-tolerant and *RAP* biclusters are enriched by GO terms, we show the dis-

| **Run ID** | **Parameter Settings** | **# total biclusters** | **# genes covered[1]** | **# top biclusters** | **# genes covered[2]** | **Size distribution[2] # of genes: # of biclusters** | **Time taken (seconds)** |
|---|---|---|---|---|---|---|---|
| **Error-tolerant Biclusters on Mega Yeast Data Set** | | | | | | | |
| *ET-bicluster*$_{M1}$ | $\alpha = 0.5$,<br>$\epsilon = 0\ for\ RS \in [120\ 150)$,<br>$\epsilon = 0.25\ for\ RS \geq 150$ | 153,960 | 361 | 500 | 295 | 2:128, 3:235, 4:8, 5:76, 6:39, 7:7, 8:2, 9:1, 10:2, 11:1, 13:1 | 10,560 |
| *ET-bicluster*$_{M2}$ | $\alpha = 0.3$,<br>$\epsilon = 0\ for\ RS \in [60\ 90)$,<br>$\epsilon = 0.25\ for\ RS \geq 90$ | 271,101 | 792 | 500 | 233 | 3:203, 4:28, 5:177, 6:80, 7:5, 8:3, 9:3, 10:1 | 33,000 |
| **RAP Biclusters on Mega Yeast Data Set** | | | | | | | |
| $RAP_{M1}$ | $\alpha = 0.5,\ RS \geq 120$ | 33,330 | 361 | 500 | 247 | 2:68, 3:379, 4:33, 5:16, 6:4 | 642 |
| $RAP_{M2}$ | $\alpha = 0.3,\ RS \geq 60$ | 94,806 | 792 | 500 | 241 | 3:384, 4:68, 5:43, 6:5 | 7,580 |
| **Error-tolerant Biclusters on Hughes et. al's Data Set** | | | | | | | |
| *ET-bicluster*$_{H1}$ | $\alpha = 0.8$,<br>$\epsilon = 0\ for\ RS \in [10\ 15)$,<br>$\epsilon = 0.25\ for\ RS \geq 15$ | 150,372 | 506 | 496 | 437 | 2:210, 3:187, 4:12, 5:66, 6:14, 7:3, 8:1, 10:1, 11:1, 13:1 | 8,360 |
| *ET-bicluster*$_{H2}$ | $\alpha = 0.5$,<br>$\epsilon = 0\ for\ RS \in [6\ 10)$,<br>$\epsilon = 0.25\ for\ RS \geq 10$ | 234,761 | 1135 | 500 | 443 | 2:115, 3:258, 4:22, 5:69, 6:24, 7:6, 8:1, 9:2, 11:1, 13:1, 14:1 | 21,745 |
| **RAP Biclusters on Hughes et. al's Data Set** | | | | | | | |
| $RAP_{H1}$ | $\alpha = 0.8,\ RS \geq 10$ | 56,009 | 506 | 495 | 438 | 2:212, 3:207, 4:25, 5:40, 6:5, 7:3, 8:2, 11:1 | 2,835 |
| $RAP_{H2}$ | $\alpha = 0.5,\ RS \geq 6$ | 80,335 | 1135 | 500 | 405 | 2:96, 3:303, 4:18, 5:75, 6:2, 7:2, 8:3, 12:1 | 1,505 |

**Table 1: Statistics of biclusters obtained using '*ET-bicluster*' and '*RAP*' from Mega Yeast and Hughes et al's microarray gene-expression data sets. ([1]all biclusters, [2]top biclusters)**

tribution of $-log_{10}(pvalue)$ and size of the biclusters. While figures 2 (a) and (b) show this distribution for mega yeast data set corresponding to two $\alpha$ values of 0.5 and 0.3, figures 2 (c) and (d) show this distribution for Hughes et al's data set corresponding to $\alpha$ values of 0.8 and 0.5 considered in this study. It can be seen from these plots that *ET-bicluster* algorithm not only generates bigger biclusters (in terms of number of genes in them) as discussed before, but also these biclusters have high $-log_{10}(pvalue)$ (or low p-value), which means it is highly unlikely to have discovered these error-tolerant biclusters by random chance. Consider mega yeast data for example, while *ET-bicluster* algorithm can discover biclusters of sizes as big as 13 (for $\alpha = 0.5$) and 10 (for $\alpha = 0.3$), *RAP* algorithm can only discover biclusters of maximum size 6. Moreover, enrichment scores of these larger error-tolerant biclusters (computed using the minimum p-value estimated for these biclusters for 2652 classes) are reasonably high. Therefore, even if the number of unique genes covered and number of enriched GO terms are comparable for *ET-bicluster* and *RAP* algorithm, the degree to which error-tolerant biclusters enrich the GO terms is certainly higher. In other words, *ET-bicluster* algorithm can find more relationships among the genes covered and as shown by functional enrichment analysis, these relationships indeed seem to be biologically relevant and not spurious.

Further, considering various p-value thresholds (from loose – $5\times10^{-2}$ to strict – $1\times10^{-5}$), we collected two more statistics. First, the fraction of biclusters that are enriched by at least one GO term, and second, the fraction of GO terms that enriched at least one bicluster. To illustrate the efficacy of *ET-bicluster* in capturing the functional coherence among genes, and comparing it with *RAP*, the above two statistics are collected for all the runs shown in table 1. For instance, if we compare these statistics for mega yeast data, while 83% of the top 500 error-tolerant biclusters (corresponding to Run ID *ET-bicluster*$_{M2}$) were enriched, only 76% of the top 500 RAP biclusters (corresponding to Run ID $RAP_{M2}$) were enriched by at least one GO term at a reasonable p-value threshold of $1\times10^{-3}$, a gain of 7%. At even more strict p-value threshold of $1 \times 10^{-5}$, the gain is 11%. Similarly, for Hughes et al's data set, though the gain is not significant, biclusters obtained from *ET-bicluster* still outperform those obtained by *RAP* in terms of the fraction of biclusters enriched. As far as the second statistics is concerned i.e. the number of GO terms that enriched at least one bicluster, performance of *ET-bicluster* and *RAP* is comparable, however, as shown in $-log_{10}(pvalue)$ vs. $size$ distribution plots, enrichment scores for error-tolerant biclusters are generally higher than *RAP* biclusters.

### 3.1.3 Statistical Significance of Error-tolerant Biclusters Using Randomization Tests

Motivated by the discussion of randomizaton tests and their importance in validating the results from any data mining approach [17], we further estimate the statistical significance of the error-tolerant biclusters using a data centric randomization approach. More specifically, an empirical p-value is computed for all the error-tolerant biclusters using the two randomization tests.

In the first randomization test, conserving the size of the top 500 error-tolerant biclusters, we generated 1000 random sets of 500 biclusters each and evaluated them by the same functional enrichment analysis using GO biological processes. So effectively, for each actual error-tolerant bicluster, we generated 1000 random biclusters of the same size (in terms of number of genes). The empirical p-value for each actual error-tolerant bicluster is then computed as the fraction of random biclusters (out of total 1000) whose enrichment score ($-log_{10}(pvalue)$) exceeds the enrichment score of the actual error-tolerant bicluster. For instance, if for a error-tolerant bicluster, only 1 out of 1000 random biclusters has higher enrichment score than it's actual value, empirical p-value of this error-tolerant bicluster is given as '1 *in* 1000' or $10^{-3}$. Figure 3 shows the $(-log_{10}(empirical\ p\text{-}values))$ for all the error-tolerant biclusters that were shown in figure 2. To plot these values at the same scale, an empirical p-value of '0 *in* 1000' is set to $10^{-5}$ to ensure that they stand out from the rest. Therefore, all the biclusters showing

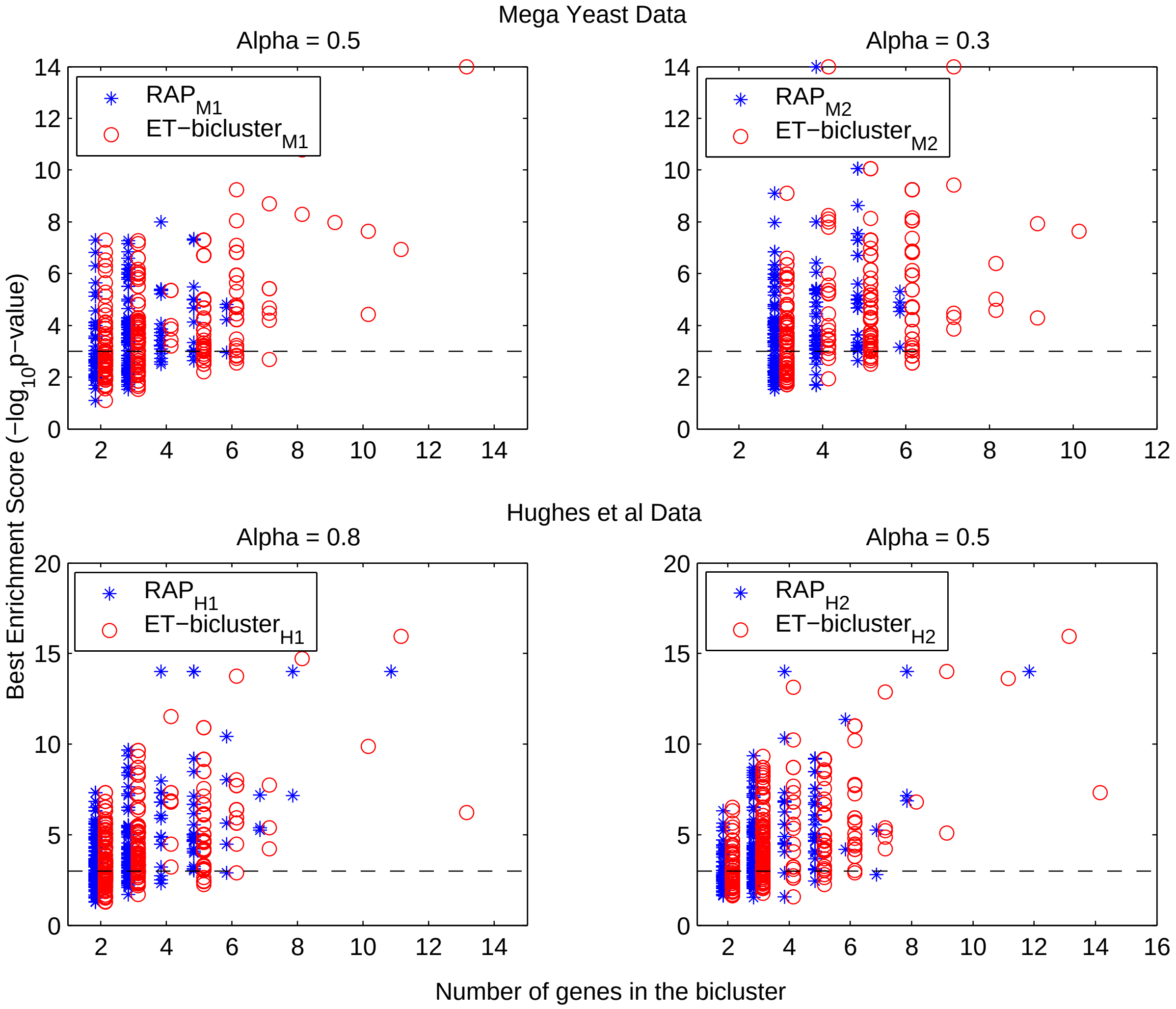


**Figure 2: Bicluster Size vs Enrichment Scores (Computed using Biological Processes) for Mega Yeast and Hughes et al's data sets**

($-log_{10}(empirical\ p\text{-}values)$) as 5 in figure 3 correspond to empirical p-value of '0 *in* 1000'. It can be clearly seen from figure 3 that error-tolerant biclusters that were assigned high enrichment scores from the GO-based evaluation also have high ($-log_{10}(empirical\ p\text{-}values)$). This means higher the enrichment score of a bicluster, less likely it is to obtain this by random chance, which further illustrates that the bigger error-tolerant biclusters discovered by only *ET-bicluster* algorithm but not by *RAP* algorithm are indeed statistically significant.

We also showed in table 2, the summary statistics of the evaluation results on 1000 randomly generated sets of biclusters. More specifically, for a given p-value threshold, we first compute for each of the 1000 random runs, the fraction of biclusters that have a p-value better than the given threshold and then we report how many times it exceeds the same fraction computed for the actual set of biclusters. It can be clearly seen from the table that specially for a stricter p-value threshold, none of the randomly generated biclusters are better than the actual biclusters. For instance, while 83% of the actual 500 biclusters on mega yeast data ('Run ID: $ET\text{-}bicluster_{M2}$') had $-log_{10}(pvalue)$ higher than 3, this percentage for 1000 random runs was substantially lower with mean of around 36% and a maximum of only 42%. The results were very similar for Hughes et al's data set. Both these set of results further confirms the statistical significance of biclusters obtained from *ET-bicluster* algorithm.

In the second randomization test, we randomized the data itself by shuffling the data values among the conditions for each gene. By doing this, we conserved the distribution of each gene profile but broke the correlation among them. We ran our proposed *ET-bicluster* algorithm on randomized mega yeast data set for example, and obtained only 42 biclusters, all of which were pairs. In contrast, application of *ET-bicluster* algorithm on actual non-randomized mega yeast data generated many more biclusters and of size as big as 10.

Both of the above randomization tests suggest that the error-tolerant biclusters obtained from the real-valued gene-expression data sets were indeed biologically meaningful and are neither obtained by random chance nor capture any random structure in the data.

### 3.2 Case Study 2 - Discovery of Biomarkers

We used four real-valued *Breast Cancer* gene-expression data sets, all of which were taken from Affymetrix platform HGU133A and normalized using RMA-normalization ap-

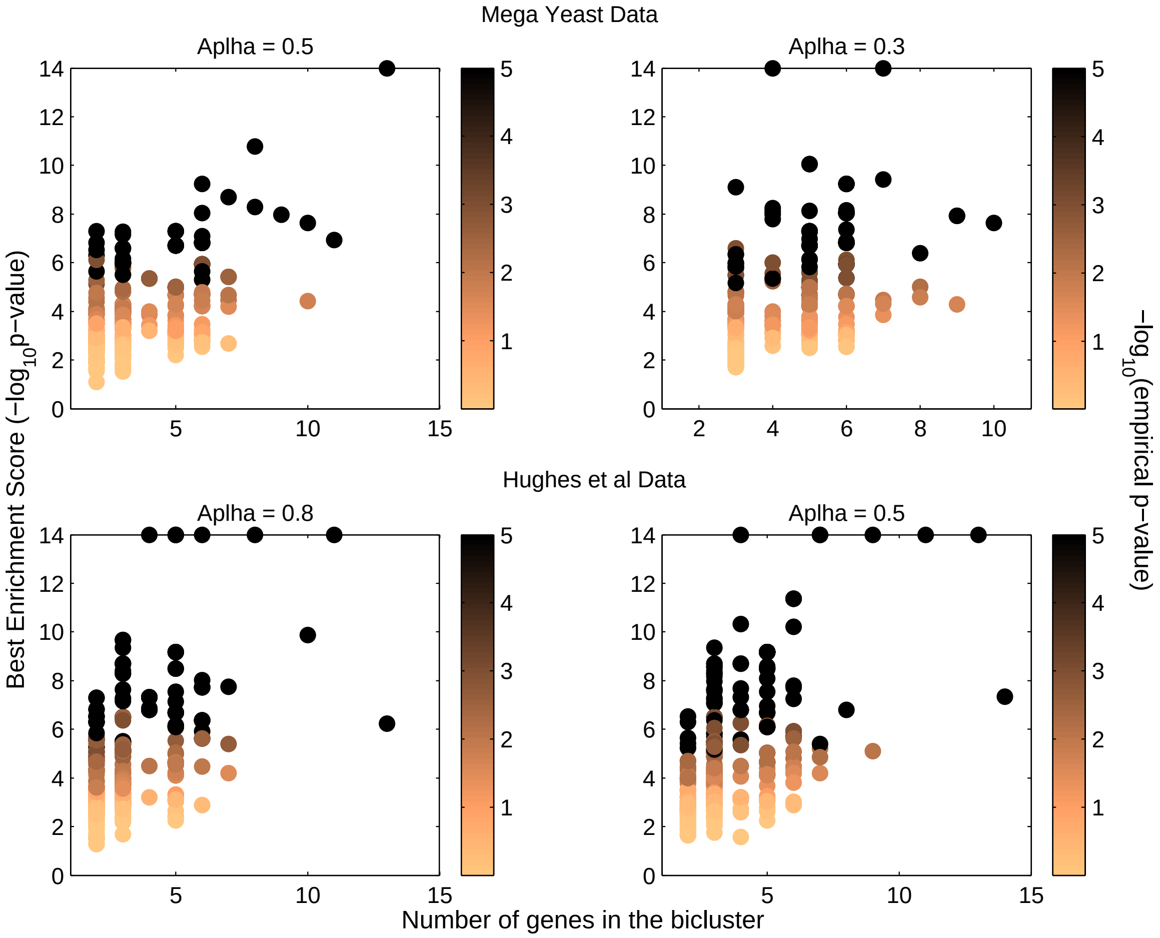


**Figure 3: Biological and Empirical p-value (using** 1000 **random runs) of the Biclusters Obtained from** *ET-bicluster* **Algorithm [figure best viewed in color].**

| Run ID | # of random runs out of 1000<br>in which fraction of biclusters enriched<br>exceeds the fraction for the true run | | | | |
|---|---|---|---|---|---|
| | pval ≤<br>0.05 | pval ≤<br>0.01 | pval ≤<br>0.005 | pval ≤<br>0.001 | pval ≤<br>0.00001 |
| $ET\text{-}bicluster_{M1}$ | 660 | 33 | 0 | 0 | 0 |
| $ET\text{-}bicluster_{M2}$ | 660 | 76 | 4 | 0 | 0 |
| $ET\text{-}bicluster_{H1}$ | 797 | 0 | 0 | 0 | 0 |
| $ET\text{-}bicluster_{H2}$ | 886 | 0 | 0 | 0 | 0 |

**Table 2: Statistical Significance of Biclusters Obtained from** *ET-bicluster*

proach. Please note that these gene-expression data sets are different than those considered for functional module discovery problem, in the sense that experimental conditions are replaced by two groups of patients. All the four breast cancer data sets were downloaded from GEO website: Desmedt (GSE7390), Loi (GSE6532), Miller (GSE3494) and Pawitan (GSE1456). The patients in the four data sets are classified as cases and controls based on their metastasis state. The patients who developed metastasis within 5 years of prognosis were considered as metastasis cases. The patients who were free of metastasis longer than 8 years of survival and follow-up time were considered as controls. The case-control ratio for Desmedt, Loi, Miller and Pawitan data set was 35:136, 51:112, 37:150 and 35:35 respectively. To increase the samle size, we combined these four data sets and used it for the discovery of biomarkers. This combined data set comprises of 8,920 genes and a case-control ratio of 158:433.

We discovered biclusters on combined Breast Cancer gene-expression data set using *ET-bicluster* with parameters, $\alpha = 0.5$, $RS = 80$, and $\epsilon = 0.25$.

**Selecting disriminative biclusters** First we select top biclusters using the approach defined earlier and then amongst the top biclusters, only those are selected as biomarkers that are discriminative of the two groups of patients, cases and controls. To measure the discriminative power, we used two measures, odds ratio and p-value. While odds ratio quantifies how different are cases and controls for a specific bicluster, p-value quantifies the significance of the difference reflected by odds ratio. Only those biclusters are selected that have a p-value of less than 0.05 and odds ratio of more than 2.0 (biclusters more represented in cases) or less than 0.5 (biclusters more represented in controls).

**Functional Enrichment Analysis** We evaluated all the identified biomarkers in terms of their enrichment scores using the MSigDB gene sets [31]. A p-value using a hypergeometric probability distribution, which denotes the random probability of annotating a biomarker with the gene set considered, is computed for all pair combinations of biomarkers and 5452 gene sets from MSigDB database. Enrichment score of each biomarker is then computed as $-log_{10}(p\text{-}value_{min})$ and used as a metric to compare the biomarkers obtained using *ET-bicluster* and *RAP*.

### 3.2.1 *Enrichment Analysis Using MSigDB Gene Sets*

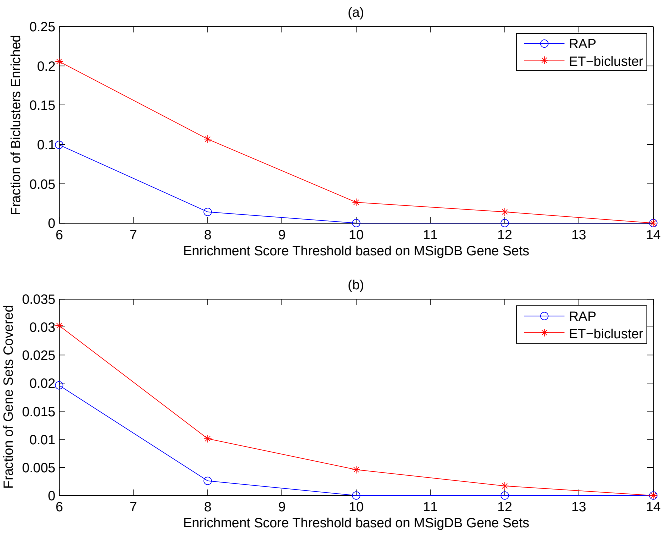


**Figure 4: (a) Fraction of Biomarkers Enriched by at least One Gene Set, (b) Fraction of Gene Sets Enriched by at least One Biomarker**

Considering various p-value thresholds (from $10^{-6}$ to $10^{-14}$), figure 4 shows two statistics: (a) fraction of biomarkers enriched by at least one gene set, and (b) fraction of gene sets that enriched at least one biomarker. These two statistics are collected both for biomarkers obtained from *ET-bicluster* and *RAP* algorithm at various p-value thresholds. As mentioned earlier, biomarkers obtained by *ET-bicluster* are not only bigger than those obtained by *RAP*, as illustrated in figure 4(a), even a higher fraction of them is enriched by at least one gene set. Consider for instance, a strict p-value threshold of $10^{-8}$ (corresponding to $-log_{10}(p\text{-}value)$ of 8 as shown on the x-axis), while 10.5% of the error-tolerant biomarkers are enriched, only 1.5% of the RAP biomarkers are enriched.

Now refer to figure 4(b), gene sets covered by *ET-bicluster* biomarkers are more than those covered by *RAP* biomarkers. The fraction of gene sets covered by biomarkers obtained from both the algorithms seems very low but this is expected because first a large number of gene sets are considered for the analysis and second, these biomarkers are only reflective of breast cancer metastasis. An important point to note however is that even a small change in fraction of gene sets covered would mean covering substantially large number of gene sets. For instance, consider a p-value threshold of $10^{-6}$ (corresponding to $-log_{10}(p\text{-}value)$ of 6 as shown on the x-axis), *ET-bicluster* and *RAP* biomarkers cover 3.03% (165 gene sets) and 1.96% (107 gene sets) respectively. These numbers for a even stricter p-value threshold of $10^{-8}$ are 1.01% (55 gene sets) 0.26% (14 gene sets) respectively.

It is clear that the biomarkers obtained from *ET-bicluster* algorithm are indeed biologically meaningful and because *RAP* algorithm does not explicitly handle noise in the data, it either completely miss some of these biologically relevant biomarkers or find fragmented parts of these, which eventually affect their enrichment score.

## 4. CONCLUSIONS

We proposed a novel error-tolerant biclustering model and presented an heuristic-based algorithm '*ET-bicluster*' to sequentially generate error-tolerant biclusters from real-valued gene-expression data in a bottom-up fashion.
We presented two biological case studies, functional module discovery and biomarker discovery, to demonstrate the importance of incorporating noise and errors in the data for discovering coherent groups of genes. In both the case studies, we found that the biclusters discovered using our proposed *ET-bicluster* algorithm are not only bigger than those obtained by *RAP* algorithm, they were also assigned a higher functional enrichment score using the biological processes GO terms and MSigDB gene sets. These results suggest that the discovered error-tolerant biclusters, not only capture the functional coherence among the genes, it is unlikely to have obtained them by random chance. We further demonstrated that the statistical significance of error-tolerant biclusters is high by computing their empirical p-value using the two randomization tests. The results from both randomization tests (one randomly selects the biclusters and other randomizes the input data itself) suggest the robustness of our proposed approach and clearly illustrate that discovered biclusters were indeed biologically and statistically meaningful and neither obtained by random chance nor capturing any random structure in the data.

## 5. LIMITATIONS AND FUTURE WORK

The work presented in this study has several limitations and can be extended in various ways. Below we discuss some of the limitations of the *ET-bicluster* algorithm and possible ideas to address them.

- Since the *range* criterion that is used to check the coherence of expression values is not anti-monotonic, the proposed *ET-bicluster* approach does not exhaustively search for all error-tolerant biclusters. Therefore, a promising idea is to define a new anti-monotonic measure that measures the coherence among the expression values and enable exhaustive search for error-tolerant biclusters.
- The current implementation of *ET-bicluster* algorithm only impose error-tolerance constraints in the bicluster row. This means that it is possible for a gene in a discovered bicluster to have all error values. To avoid this situation, one can use additional column constraint and find a subset of supporting transactions for which each column in the pattern has no more than some user-defined fraction of errors. For binary data case, this kind of additional column constraint has been used in [20], however, a heuristic-based approach is used to check the column constraint. One potential way to address this is to develop a pattern mining algorithm that checks both the row and column error-tolerance constraints, and exhaustively search for all the error-tolerant biclusters.
- As the error-tolerant pattern mining is computationally more challenging, more efficient data structures and memory management techniques can be used. This would enhance the scalability of the algorithm and enable the discovery of biclusters on a wider range of parameter settings.

We only presented comparison of *ET-bicluster* and *RAP* since comparison with other biclustering approaches such as *CC* and *ISA* is not well suited for quantifying the affect of noise/errors. Moreover *CC* and *ISA* approaches generally finds larger biclusters and follow a different approach based on optimizing an objective function. Nevertheless, it will still be interesting in future to compare *ET-bicluster* with *CC* and *ISA* for potential complementarity among them.

It is also important to note that gene-expression data provides useful but limited view of the genome and therefore biclusters obtained from gene-expression data alone may not elucidate the complete underlying biological mechanism. Hence another promising research direction is to integrate multiple biological data sources for complex problems like discovery of functional modules or biomarkers. For example, protein-protein interaction data can be used as a prior knowledge to guide the discovery of biclusters from the gene-expression data. The biclusters identified in this way are potentially more reliable and biologically plausible than those obtained from individual data sources. We are currently developing error-tolerant pattern mining based approaches for integrated analysis of gene-expression and protein-protein interaction data. One such application for discovering subnetwork based biomarkers for Breast cancer metastasis has been shown in [15], however, these approaches are primitive at this stage and further work is needed in this area.

## 6. ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their constructive and thoughtful comments. This work was supported by NSF grants IIS-0916439, CRI-0551551 and a University of Minnesota Rochester Biomedical Informatics and Computational Biology (BICB) Program Traineeship Award (Rohit Gupta). Access to computing facilities was provided by the Minnesota Supercomputing Institute.

## 7. REFERENCES

[1] M. Ashburner, C.A. Ball, J.A. Blake, D. Botstein, H. Butler, J.M. Cherry, A.P. Davis, K. Dolinski, S.S. Dwight, J.T. Eppig, et al. Gene Ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[2] C. Becquet, S. Blachon, B. Jeudy, J.F. Boulicaut, and O. Gandrillon. Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human SAGE data. *Genome Biology*, 3(12), 2002.

[3] A. Ben-Dor, B. Chor, R. Karp, and Z. Yakhini. Discovering local structure in gene expression data: the order-preserving submatrix problem. *Journal of Computational Biology*, 10(3-4):373–384, 2003.

[4] S. Bergmann, J. Ihmels, and N. Barkai. Iterative signature algorithm for the analysis of large-scale gene expression data. *Physical review E*, 67(3):31902, 2003.

[5] J. Besson, C. Robardet, and J.F. Boulicaut. Mining a new fault-tolerant pattern type as an alternative to formal concept discovery. *LNCS*, 4068:144, 2006.

[6] T. Calders, B. Goethals, and S. Jaroszewicz. Mining rank-correlated sets of numerical attributes. In *ACM SIGKDD*, page 105. ACM, 2006.

[7] H. Cheng, PS Yu, and J. Han. Ac-close: Efficiently mining approximate closed itemsets by core pattern recovery. In *ICDM*, pages 839–844, 2006.

[8] H. Cheng, P.S. Yu, and J. Han. Approximate frequent itemset mining in the presence of random noise. *Soft Computing for Knowledge Discovery and Data Mining*, page 363, 2007.

[9] Y. Cheng and GM Church. Biclustering of gene expression data. In *ISMB 2000*, pages 93–103, 2000.

[10] G. Cong, K.L. Tan, A.K.H. Tung, and F. Pan. Mining frequent closed patterns in microarray data. *ace*, 125:123.

[11] C. Creighton and S. Hanash. Mining gene expression databases for association rules, 2003.

[12] I.S. Dhillon, S. Mallela, and D.S. Modha. Information-theoretic co-clustering. In *ACM SIGKDD*, pages 89–98. ACM New York, NY, USA, 2003.

[13] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama. Mining optimized association rules for numeric attributes. *Journal of Computer and System Sciences*, 58(1):1–12, 1999.

[14] R. Gupta, G. Fang, B. Field, M. Steinbach, and V. Kumar. Quantitative evaluation of approximate frequent pattern mining algorithms. In *ACM SIGKDD*, pages 301–309. ACM, 2008.

[15] Rohit Gupta, Smita Agrawal, Navneet Rao, Ze Tian, Rui Kuang, and Vipin Kumar. Integrative Biomarker Discovery for Breast Cancer Metastasis from Gene Expression and Protein Interaction Data Using Error-tolerant Pattern Mining. In *Proc of the International Conference on Bioinformatics and Computational Biology (BICoB)*, 2010.

[16] A. Gyenesei, R. Schlapbach, E. Stolte, and U. Wagner. Frequent pattern discovery without binarization: Mining attribute profiles. *LNCS*, 4213:528, 2006.

[17] S. Hanhijärvi, M. Ojala, N. Vuokko, K. Puolamäki, N. Tatti, and H. Mannila. Tell me something I don't know: randomization strategies for iterative data mining. In *ACM SIGKDD*, pages 379–388. ACM New York, NY, USA, 2009.

[18] T.R. Hughes, M.J. Marton, A.R. Jones, C.J. Roberts, R. Stoughton, C.D. Armour, H.A. Bennett, E. Coffey, H. Dai, Y.D. He, et al. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–126, 2000.

[19] The Gasch Lab. http://gasch.genetics.wisc.edu/datasets.html.

[20] J. Liu, S. Paulsen, X. Sun, W. Wang, A. Nobel, and J. Prins. Mining approximate frequent itemsets in the presence of noise: Algorithm and analysis. In *SDM*, pages 405–416, 2006.

[21] J. Liu, S. Paulsen, W. Wang, A. Nobel, and J. Prins. Mining approximate frequent itemsets from noisy data. In *IEEE ICDM*, page 4, 2005.

[22] S.C. Madeira and A.L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE Transactions on computational Biology and Bioinformatics*, pages 24–45, 2004.

[23] T. McIntosh and S. Chawla. High-Confidence Rule Mining for Microarray Analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pages 611–623, 2007.

[24] G. Pandey, G. Atluri, M. Steinbach, C.L. Myers, and V. Kumar. An association analysis approach to biclustering. In *ACM SIGKDD*, pages 677–686. ACM New York, NY, USA, 2009.

[25] A.K. Poernomo and V. Gopalkrishnan. Mining statistical information of frequent fault-tolerant patterns in transactional databases. In *ICDM*, pages 272–281. IEEE Computer Society Washington, DC, USA, 2007.

[26] A.K. Poernomo and V. Gopalkrishnan. Towards efficient mining of proportional fault-tolerant frequent itemsets. In *ACM SIGKDD*, pages 697–706. ACM New York, NY, USA, 2009.

[27] A. Prelic, S. Bleuler, P. Zimmermann, A. Wille, P. B "uhlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler. A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics*, 22(9):1122–1129, 2006.

[28] R. Rastogi and K. Shim. Mining optimized association rules with categorical and numeric attributes. *IEEE TKDE*, pages 29–50, 2002.

[29] J.K. Seppänen and H. Mannila. Dense itemsets. In *ACM SIGKDD*, pages 683–688. ACM New York, NY, USA, 2004.

[30] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. *ACM SIGMOD Record*, 25(2):12, 1996.

[31] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, and J. P. Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A*, 102(43):15545–50, 2005.

[32] A. Tanay, R. Sharan, and R. Shamir. Discovering statistically significant biclusters in gene expression data, 2002.

[33] C. Yang, U. Fayyad, and P.S. Bradley. Efficient discovery of error-tolerant frequent itemsets in high dimensions. In *ACM SIGKDD*, pages 194–203. ACM New York, NY, USA, 2001.

[34] M. Zhang, W. Wang, and J. Liu. Mining approximate order preserving clusters in the presence of noise. In *Proc. ICDE*, volume 8, pages 160–168. Citeseer, 2008.