Iterative Search with Incremental MSR Difference Threshold for Biclustering Gene Expression Data

Shyama Das
Department of Computer Science
Cochin University of Science and
Technology, Kochi, Kerala, India

Sumam Mary Idicula
Department of Computer Science,
Cochin University of Science and
Technology, Kochi, Kerala, India,

ABSTRACT

The goal of biclustering in a gene expression data matrix is to find a submatrix such that the genes in the submatrix show highly correlated activities across all conditions in the submatrix. A measure called Mean Squared Residue (MSR) is used to simultaneously evaluate the coherence of rows and columns within a submatrix. In this paper a new method for biclustering gene expression data is developed. In the first step high quality bicluster seeds are generated using K-Means clustering algorithm. Then more genes and conditions (node) are added to the bicluster. Before adding a node the MSR X of the bicluster is calculated. After adding the node again the MSR Y is calculated. The added node is deleted if Y minus X is greater than MSR difference threshold or if Y is greater than δ (MSR threshold) which depends on the dataset. The MSR difference threshold is different for gene list and condition list and it depends on the dataset also. Proper values should be identified through experimentation in order to obtain biclusters of large size. Since it is very difficult to calculate the value of MSR difference threshold, in this algorithm an iterative search is used where MSR difference threshold is initialized with a small value and it is incremented after each iteration. A bicluster is obtained from Yeast dataset with a unique structural appearance. This proves that the newly introduced concept of MSR difference threshold will result in high quality biclusters. The results obtained on bench mark datasets prove that this algorithm is better than many of the existing biclustering algorithms.

Categories and Subject Descriptors

H.2.8 [**Information Systems**]: Data Mining J.3 [**Computer Applications]**: Life and Medical Sciences

General Terms

Algorithms, Experimentation, Measurement

Keywords

Biclustering, gene expression data, K-Means clustering, Mean Squared Residue

1.INTRODUCTION

Microarray technologies simultaneously measure the expression levels of thousands of genes in a single experiment. Microarray data are widely used in medical domain. It is also used genomic research because of the enormous potential in gene expression profiling, facilitating the prognosis and the discovery of subtypes of diseases. Gene expression data is organized in the form of a matrix where rows represent genes and columns represent experimental conditions. Each element in the matrix refers to the expression level of a particular gene under a specific condition.

Clustering is the most popular data mining technique for analyzing gene expression data to group conditions or genes. However clustering has its own limitations. Clustering is based on the assumption that related genes behave similarly across all measured conditions. In the cellular process the subset of genes are co-regulated and co-expressed under certain experimental conditions. But they behave almost independently under some other conditions. Moreover clustering will separate genes into disjoint sets i.e. each gene is associated with a single biological function which is in contradiction to the biological system as a whole.

Biclustering is clustering applied in two dimensions simultaneously. This approach identifies group of genes that show similar expression level under a specific subset of experimental conditions. Hartigan introduced biclustering [1] who called it direct clustering. Cheng and Church were the first to apply biclustering to gene expression data [2]. They introduced a measure known as mean squared residue score to evaluate the coherence of the elements of a bicluster.

Biclusters can generally be classified into four major types. They are: biclusters with constant values, biclusters with constant values on rows or columns, biclusters with coherent values, and biclusters with coherent evolutions. In the case of gene expression data the constant biclusters disclose subsets of genes with similar expression values within a subset of conditions. But a bicluster with constant values in the rows normally identifies a subset of genes with similar expression values across a subset of conditions permitting the expression levels to vary from gene to gene. In this manner a bicluster with constant columns identifies a subset of conditions within which a subset of genes manifest similar expression values presupposing that the expression values might vary from condition to condition. In the case of a bicluster with coherent values, a subset of genes and a subset of conditions with coherent values on both the rows and columns are identified. In this case the similarity among the genes is calculated as the mean squared residue score. If the similarity measure (mean squared residue score) of a matrix satisfies a certain threshold, it is a bicluster. A bicluster with coherent evolutions is a subset of genes which are up-regulated or down-regulated across a subset of conditions without considering their actual expression values [3].

Bicluster model is much more flexible than the row clusters. It is not necessary that the identified submatrices to be disjoint or to cover the entire matrix. But the computation could be costly because one will have to consider all the combinations of columns and rows in order to find out all the biclusters. The search space for the biclustering problem is 2^{m+n} where m and n are size of genes and conditions respectively. Usually m+n is more than 2000. The biclustering problem is Np-hard.

In this work a novel algorithm is developed using the concept of MSR difference threshold. The seeds obtained from K-Means clustering algorithm is enlarged using this algorithm.

2.MATERIALS AND METHODS

2.1 Problem Definition

The gene expression dataset can be viewed as an NxM matrix A of real numbers. A bicluster of a gene expression dataset is a subset of genes which exhibit similar expression patterns along a subset of conditions. Let X={G_1,G_2,....G_N} be the set of genes and Y={C_1,...C_M} be the set of conditions in the gene expression dataset. A bicluster is a subset of rows that shows a coherent behaviour across a subset of columns and vice versa. A bicluster is a submatrix B of A and if the size of B is IxJ, then I is a subset of rows X of A, and J is the subset of the columns Y of A. The rows and columns of the bicluster B need not be contiguous as in the expression matrix A.

Biclusters with coherent values are biologically more relevant than biclusters with constant values. In this work biclusters with coherent values are identified. Thus the problem of biclustering can be formulated as follows: given a data matrix A, find a set of submatrices B1, B2... Bn that satisfy some homogeneity characteristics or coherence. It is not essential that the identified submatrices to be disjoint or to cover the entire matrix. A bicluster with coherent values identifies a subset of genes and a subset of conditions with coherent values on both rows and columns. The degree of coherence is measured by mean squared residue score or hscore. It is the sum of the squared residue score. The residue of an element reveals its degree of coherence with the other elements of the bicluster it belongs to. The residue score of an element *bij* in a submatrix *B* is defined as *RS(bij)=bij-bIj-biJ+bIJ*

The residue score of an element *bij* provides the difference between the actual value and its expected value predicted from its row mean, column mean and bicluster mean. Hence from the value of residue, the quality of the bicluster can be evaluated by computing the mean squared residue. That is Hscore or mean squared residue score of bicluster *B* is

MSR (*B*) $= \frac{1}{|I||J|}\sum i \in I, j \in J\,(\mathrm{RS}(bij))^2$ where

$$biJ = \frac{1}{|j|}\sum_{j\in J}(bij)$$

$$bIj = \frac{1}{|I|}\sum_{i\in I}(bij)$$

$$bIJ = \frac{1}{|I||J|}\sum_{i\in I j\in J}(bij)$$

Here *I* denotes the row set, *J* denotes the column set, *bij* denotes the element in a submatrix, *biJ* denotes the ith row mean, *bIj* denotes the *j*th column mean, and *bIJ* denotes the mean of the whole bicluster. A bicluster *B* is called a δ bicluster if MSR (*B*)< δ for some δ >0 i.e. δ is the MSR threshold. If the MSR value is high it means that the data is uncorrelated. If the MSR value is low then there is correlation in the matrix. The value of δ depends on the dataset and it should be calculated in advance. For Yeast dataset the value of δ is 300 and for Lymphoma dataset the value of δ is 1200. The volume of a bicluster or bicluster size is the product of number of rows and the number of columns in the bicluster. The larger the volume and smaller the MSR or Hscore of the bicluster the better is the quality of the bicluster.

2.2 Encoding of Bicluster

Bicluster is represented by a binary string of fixed length n+m, where *n* and *m* are the number of genes and conditions of the microarray dataset respectively. The first n bits is associated to n genes, the following m bits to m conditions. If a bit is set to 1, it means that the corresponding gene or condition belongs to the bicluster; otherwise it does not. This encoding renders the advantage of having a fixed size [4].

2.3 Seed Finding using K-Means Clustering Algorithm

The gene expression dataset is partitioned into n gene clusters and m sample clusters using the K-Means algorithm. In order to get maximum 10 genes per cluster, it is further divided according to the cosine angle distance from the cluster centre. Similarly each sample cluster is further divided into sets of 5 samples according to cosine angle distance from the cluster centre. If the number of gene clusters, having maximum 10 close genes is p and number of sample clusters having maximum 5 conditions is q. The initial gene expression data matrix is thus partitioned into p*q submatrices and bicluster seeds having Hscore value below a certain limit is selected to initialize the bicluster [5].

2.4 Biclustering using Iterative Search with Incremental MSR Difference Threshold (BISIMT)

The seeds obtained by K-Means algorithm are enlarged by adding more conditions and genes. MSR difference of a gene or condition is the incremental increase in MSR after adding the same to the bicluster. In this method a gene or condition (node) is added to the bicluster. If the incremental value of MSR after adding the node is greater than MSR difference threshold or if the MSR value of the resulting bicluster is greater than δ(MSR threshold), the added node is removed from the bicluster. MSR difference threshold is different for gene list as well as condition list. And it varies depending on the dataset also. The identification of suitable value needs experimentation. In this algorithm MSR difference threshold is initialized with a small value and incremented after each iteration in fixed steps until it reaches a final value. So in this algorithm there are three different parameters such as the initial value of MSR difference threshold, the amount by which it is incremented after each iteration and the final value of MSR difference threshold.

These three parameters apply for both the gene list and condition list. By properly adjusting the MSR difference threshold parameters biclusters of high quality can be obtained. For example consider the bicluster shown in Figure 1. This bicluster is obtained using MSR difference threshold on the Yeast dataset. For Yeast dataset biclusters with such structural appearance are never seen in the literature. But for Lymphoma dataset usually biclusters with such structural appearance are usually identified. This means that the concept of MSR difference threshold is really an innovative idea in the field of biclustering gene expression data. However proper identification of suitable values need more experimentation.

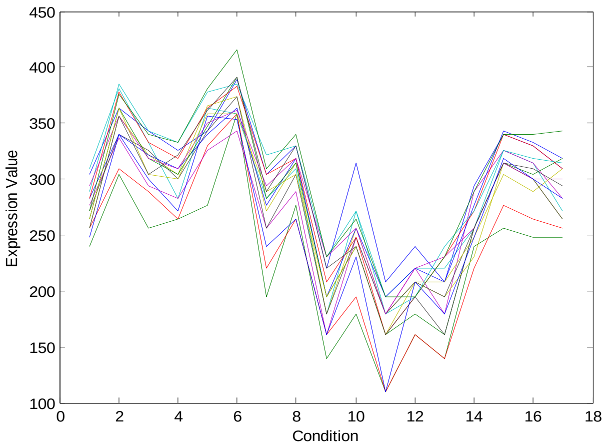


Figure. 1. The unique bicluster obtained for Yeast dataset using the concept of MSR difference threshold

As is shown in Figure 1 a bicluster with a unique structural appearance is obtained for Yeast dataset with size 16*17 and MSR 199.3662 using the concept of MSR difference threshold. This bicluster has strikingly similar up-regulation and down regulation and is with a structural appearance which is hitherto unseen in any of the literature published so far. In the case of novel greedy search algorithm [6] the added node is removed only when the MSR of the bicluster exceeds δ (MSR threshold). But when MSR difference threshold is applied there is more restriction on the incremental value of hscore or MSR which means that the elements in the biclusters are more tightly packed. This will result in biclusters of larger size and low mean squared residue score. Hence this method can produce better biclusters compared with other algorithms like novel greedy search algorithm.

2.5 Iterative Search with Incremental MSR Difference Threshold Algorithm

```
Algorithm IterativeMSRdifference(seed, δ, condthreshinitial, condthreshincrement, condthreshfinal, genethreshinitial, genethreshincrement, genethreshfinal)
bicluster := seed
previous=MSR(seed)
j:= 1;
msrdiffcondthresh=condthreshinitial;
while  (msrdiffcondthresh<condthreshfinal)
          While (j <= total _no_conditions)
     If   condition[ j]  is not included in bicluster
          Changed=1;
           Add all elements of condition[ j]  corresponding to genes
           already included to bicluster
           present= MSR(bicluster)
          if (present> δ) or (present-previous)>msrdiffcondthresh
              remove elements of  condition[ j]  from bicluster
              changed=0;
            endif
           if changed==1
                 previous=present
          endif
   endif
  j:= j+1
  end(while)
  msrdiffcondthresh=msrdiffcondthresh+condthreshincrement
end(while)
 i := 1;
prev=MSR(bicluster)
msrdiffgenethresh=genethreshinitial
```

```
While(msrdiffgenethresh<=genethreshfinal)
   While (i <= total _no_ genes)
      If  gene[i]  is not included in bicluster
         Changed=1;
         Add all elements of gene[i] corresponding to conditions
         already included to bicluster
          present= MSR(bicluster)
         if (present> $\delta$) or (present-previous)>msrdiffgenethresh
                remove elements of  gene[i] from bicluster
               changed=0
         endif
        if changed==1
           previous=present
        endif
    endif
  i:= i+1
 end(while)
  msrdiffgenethresh=msrdiffgenethresh+genethreshincrement
end(while)
return bicluster
end(IterativeMSRdifference)
```

3.EXPERIMENTAL RESULTS

3.1 Datasets used

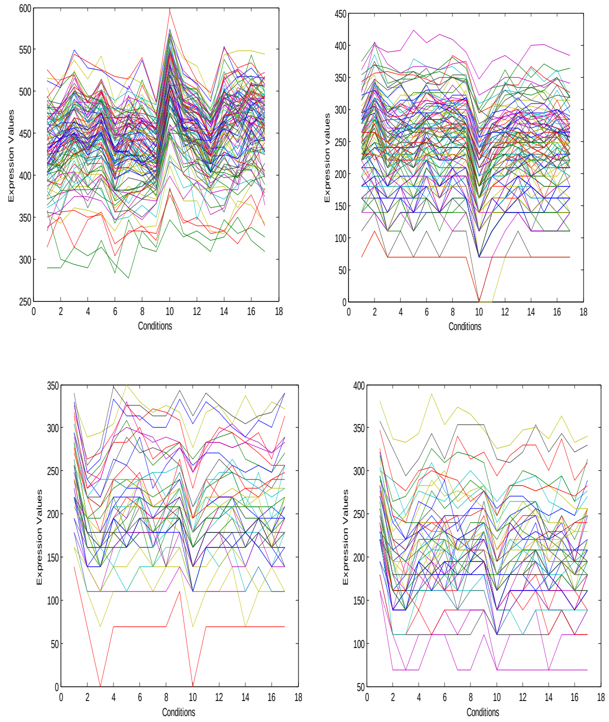
Experiments are conducted on the Yeast Saccharomyces Cerevisiae Cell Cycle expression dataset and Lymphoma dataset in order to evaluate the quality of the proposed algorithm. The algorithm is implemented in Matlab. The Yeast dataset is based on Tavazoie *et al* [7]. Yeast dataset consists of 2884 genes and 17 conditions. The expression values were transformed by scaling and logarithm x→100 log (10^3x). The values in the expression dataset after this transformation are integers in the range 0 to 600. Missing values are represented by -1. Human B-cell Lymphoma expression data contain 4026 genes and 96 conditions. The dataset was downloaded from the website for supplementary information for the article by Alizadeh et al. (2000) [8]. The expression levels were reported as log ratios. After scaling by a factor of 100 the values in the Lymphoma dataset are integers in the range -750 to 650. There are 47,639 (12.3%) missing values in the Lymphoma dataset. Missing values were represented by 999. The datasets after the above preprocessing is obtained from http://arep.med.harvard.edu/biclustering.

3.2 Missing Data Replacement

Missing data in the matrices are replaced with random numbers. It is expected that these random values would not form identifiable patterns. Hence these would be the leading candidates to be removed in node deletion. The random numbers which are used to replace the missing values in the Human Lymphoma dataset are generated so that they are uniformly distributed between -800 and 800. For the Yeast dataset for a set of two genes the entire elements are null values represented by -1. They are then removed from the dataset.

3.3 Bicluster Plots for Yeast Dataset

In Figure 2 eight biclusters obtained using BISIMT algorithm is shown. Out of the eight biclusters shown in Figure 2, six contain all 17 conditions and they differ in appearence. In short the algorithm is ideal for identifying various biclusters with coherent values. From the bicluster plots which show strikingly similar upregulation and down regulation we can conclude that this is an innovative and ideal method for identifying significant biclusters from gene expression data. For Yeast dataset biclusters are found by setting the initial value of MSR difference threshold for condition list as 5. It is incremented by 5 after each iteration and the final value of MSR difference threshold is set to 30. Initial value of MSR difference threshold for gene list is set to 1, and it is incremented by 1and the final value is set to 10.



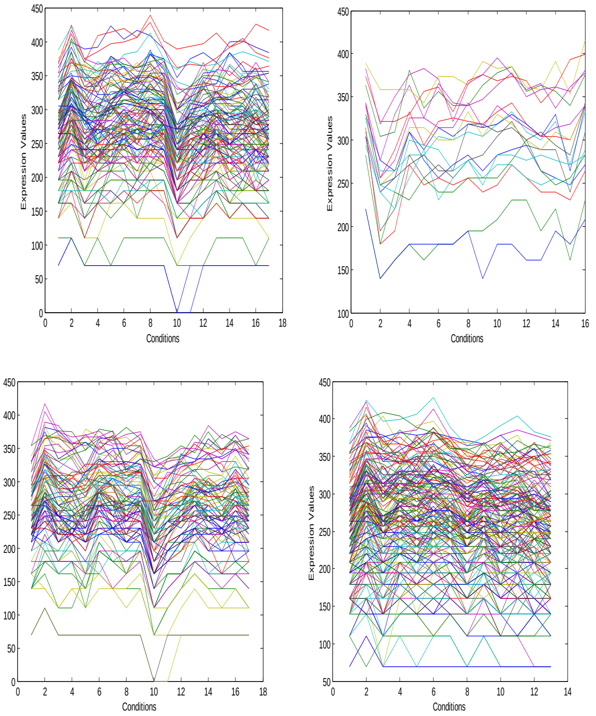


Figure 2. Eight biclusters found for the Yeast dataset.

Bicluster labels are (a), (b), (c), (d), (e), (f), (g) and (h) respectively. In the bicluster plots X axis contains conditions and Y axis contains expression values. The details about biclusters can be obtained from Table I using bicluster label. All the means squared residues are lower than 200. Only biclusters with different shapes are selected.

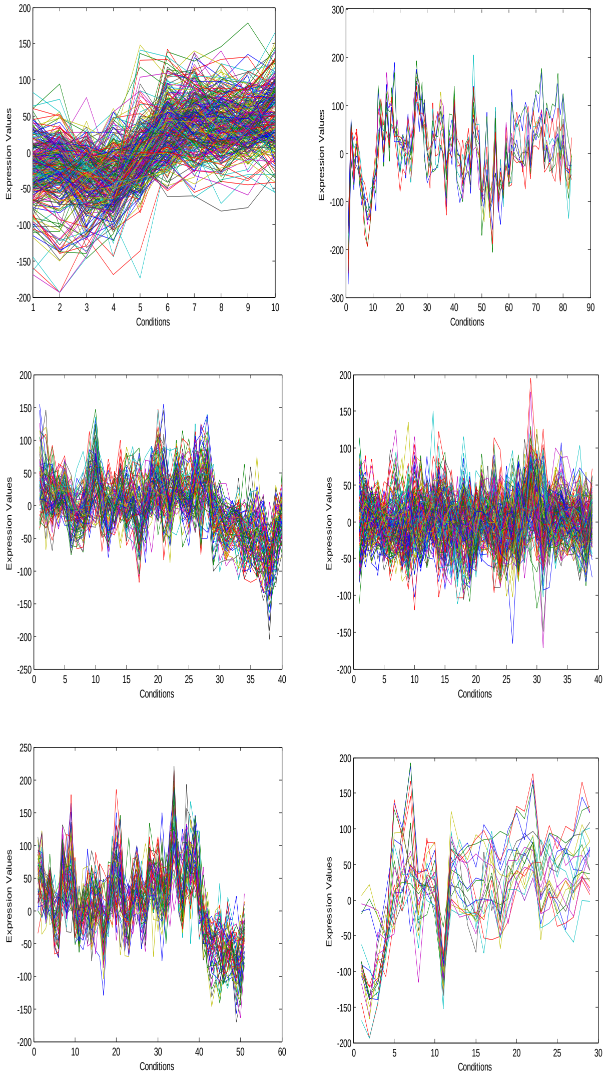
Table 1. Information about Biclusters of Figure 2.

Label	Rows	Columns	Volume	MSR
(a)	98	17	1666	199.9381
(b)	107	17	1819	199.9826
(c)	43	17	731	199.8613
(d)	50	17	850	199.5999
(e)	127	17	2159	199.9656
(f)	19	16	304	199.9141
(g)	99	17	1683	199.9524
(h)	188	13	2444	199.9713

In the above table the first column contains the label of each bicluster. The second and third columns report the number of rows (genes) and number of columns (conditions) of the bicluster respectively. The fourth column reports the volume of the bicluster and the last column contains the mean squared residue or Hscore of the bicluster.

3.4 Bicluster plots for Human Lymphoma Dataset

Figure 3 shows eight bicluster obtained by BISIMT algorithm on Human Lymphoma dataset. The labels of biclusters are (p), (q), (r), (s), (t), (u), (v) and (w) respectively. Here for condition list the initial value of MSR difference threshold is set to 30 and it is incremented by 30 after each iteration and the final value is set to 90. For the gene list the initial value of MSR difference threshold is set to 50 and it is incremented by 50 after each iteration and final value is set to 150. Value of δ is set to 1000 to get biclusters labeled (r), (s), (t) and (u). For all other biclusters the value of δ is set to 1200. All the bicluster plots show strikingly similar upregulation and down regulation.



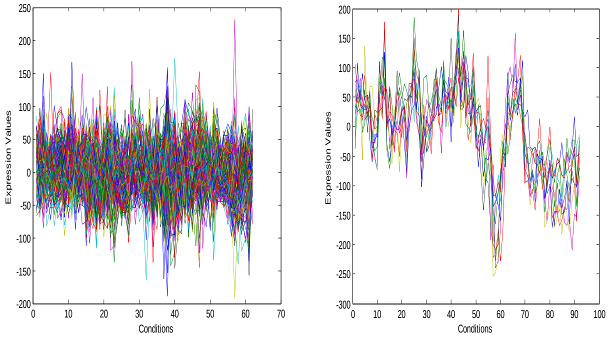


Figure 3. Eight biclusters found for the Lymphoma dataset.

In the bicluster plots X axis contains conditions and Y axis contains expression values. The details about biclusters can be obtained from Table 2 using bicluster label. All the means squared residues are lower than 1200. Only biclusters with different shapes are selected.

Table 2. Information about biclusters of Figure 3.

Label	Rows	Columns	Volume	MSR
(p)	281	10	2810	1001.4000
(q)	10	84	840	1194.9000
(r)	86	40	3440	999.8830
(s)	155	39	6045	999.9171
(t)	50	51	2550	999.9309
(u)	20	29	580	987.8022
(v)	172	62	10664	1199.8000
(w)	10	92	920	1197.4000

In the above table the first column contains the label of each bicluster. The second and third columns report the number of rows (genes) and of columns (conditions) of the bicluster respectively. The fourth column reports the volume of the bicluster and the last column contains the mean squared residue or hscore of the bicluster.

3.5 Significance Evaluation

Biclusters can be evaluated using prior biological knowledge [9]. Biological relevance of biclusters obtained using BISIMT algorithm is verified using a small bicluster of size 12x17. GO annotation database can be used to determine the biological significance of biclusters. In this database genes are assigned to three structured controlled vocabularies. Gene products are described in terms of associated biological process, components and molecular functions in a species-independent manner. To evaluate the statistical significance for the genes in each bicluster p-values are used. P-values indicate the extent to which the genes in the bicluster match with the different GO categories. Smaller p-values indicates better match. Yeast genome gene ontology term finder [10] is such a database available in the Internet which can be used to evaluate the biological significance of biclusters.

In the bicluster selected for testing the biological significance there are 12 genes namely YJR123W, YKL056C, YKL060C, YKL152C, YKR057W, YKR094C, YLR029C, YLR075W, YLR167W, YLR185W, YLR325C, YPR102C. Figure 4 shows the significant GO terms for the set of 12 genes along with their p values. It shows the branching of generalized molecular function into sub-functions like structural molecule activity, binding and protein tag. These activities are clustered using genes to produce the final result. Figure 4 is obtained when gene ontology database is searched by entering the names of genes and by selecting function ontology.

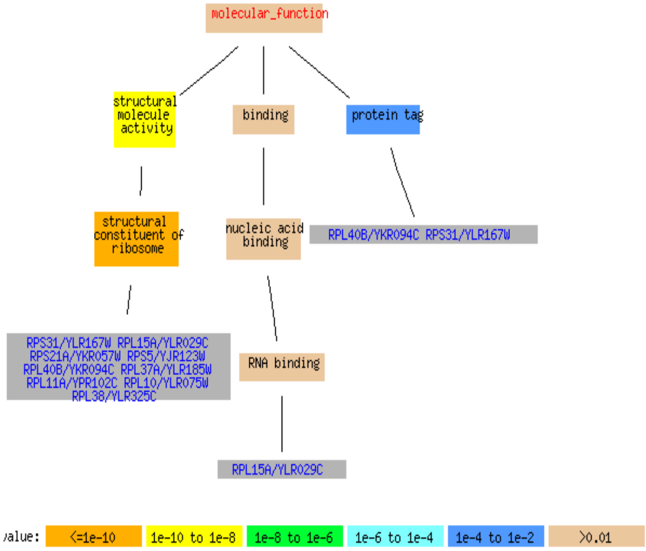

molecular_function
structural molecule activity
binding
protein tag
structural constituent of ribosome
nucleic acid binding
RPL40B/YKR094C RPS31/YLR167W
RPS31/YLR167W RPL15A/YLR029C RPS21A/YKR057W RPS5/YJR123W RPL40B/YKR094C RPL37A/YLR185W RPL11A/YPR102C RPL10/YLR075W RPL38/YLR325C
RNA binding
RPL15A/YLR029C
/alue:
<=1e-10
1e-10 to 1e-8
1e-8 to 1e-6
1e-6 to 1e-4
1e-4 to 1e-2
>0.01

Figure 4. Sample of 12 genes for Yeast data, with corresponding GO terms and their parents for function ontology

Table 3 shows the significant GO terms used to describe the set of 12 genes of the bicluster for the process, function and component ontologies. The common terms are described with increasing order of p-values or decreasing order of significance. In Table 3 the first entry of the second column with the title process contains the tuple Translation (10,2.49e-07) which means that 10 out of the 12 genes of the bicluster are involved in the process of translation and their p-value is 2.49e -07. Second and third entry means that 12 out of 12 genes are involved in cellular biosynthetic process and biosynthetic process. This means that the bicluster contains biologically similar genes and the method used here is capable of identifying biologically significant biclusters.

Table 3. Significant Shared GO terms (process, function, component) of the 12 genes in a small bicluster obtained using BISIMT algorithm

Genes Num.	Process	Function	Component
12	Translation (10,2.49e-07)	structural constituent of ribosome (9, 2.32e-11)	cytosol (12,1.02e-11)
	Cellular biosynthetic process(12,1.03e-05)	structural molecule activity(9, 1.08e-09	cytosolic ribosome (9,1.50e-11)
	Biosynthetic process(12,1.10e-05)	protein tag (2, 0.00037	ribosome (10,1.17e -10)

4. COMPARISON

In the Table 4 given below a comparative summarization of results of Yeast dataset involving the performance of related algorithms is provided. All the algorithms listed in the Table 4 have MSR value more or less equal to 200, even though the maximum limit of δ is 300. Thus the value of δ is set to 200 in this study. The performance of BISIMT algorithm in comparison with that of Novel Greedy [6], SEBI [11], Cheng and Church's algorithm (CC) [3], and the algorithm FLOC by Yang et al. [12] and DBF [13] etc. for the Yeast dataset are given. SEBI (Sequential Evolutionary Biclustering) is based on evolutionary algorithms. In the Cheng and Church approach, rows/columns were deleted from the gene expression data matrix for finding a bicluster. This means that their algorithm is based on greedy row/column removal strategy. Yang et al (2003) generalized the model of bicluster proposed by Cheng and Church for incorporating null values and for removing random interference. They developed a probabilistic algorithm FLOC which can discover a set of possibly overlapping biclusters simultaneously. Zhang et al. presented DBF (Deterministic Biclustering with frequent pattern mining). In the case of DBF a set of good quality bicluster seeds are generated in the first phase based on frequent pattern mining. In the second phase of the algorithm these biclusters are enlarged by adding more genes or conditions. In the case of BISIMT algorithm presented here average number of genes is greater than that of SEBI whereas the average number of conditions is better than that of all other algorithms. Average volume is greater than that of novel greedy, SEBI and CC. Average residue is lower than that of CC and SEBI. The BISIMT algorithm has high value for the largest bicluster size compared to novel greedy, SEBI and FLOC.

Table 4. Performance comparison between BISIMT and other algorithms for Yeast dataset

Algori thm	Avg. Gene num.	Avg Cond. num.	Avg. Volume	Avg. MSR	Largest Bicluster
BISIMT	123.80	16.20	1954.20	199.96	2444
Novel Greedy	94.75	14.75	1422.87	199.78	2112
SEBI	13.61	15.25	209.92	205.18	1394
CC	166.71	12.09	1576.98	204.29	4485
FLOC	195.00	12.80	1825.78	187.54	2000
DBF	188.00	11.00	1627.20	114.70	4000

As is clear from the above table the average mean squared residue, the average number of genes and conditions, average volume and largest bicluster size are compared for various algorithms. For the average mean squared residue field lower values are better where as higher values are better for all other fields.

Table 5 gives a performance comparison for Human B-cell Lymphoma dataset. Value of δ is set to 1200 for Lymphoma dataset. Here the average gene number is greater than SEBI. Average value of condition is better than all other algorithms. Average volume is better than that of SEBI. Average MSR is lower than that of Novel Greedy. Even though theoretically BISIMT is better than that of Novel Greedy, for lymphoma dataset average gene number and average volume is better for novel greedy. One reason for this is low value of average MSR and high value of average condition number for BISIMT compared to novel greedy. Usually multi-objective algorithms will produce biclusters of larger size compared to greedy algorithms. But in the case of multi-objective evolutionary computation [14] the maximum number of conditions obtained is only 11 in the case of Yeast dataset and 40 in the case of Human B-cell Lymphoma dataset. But in this method there are biclusters with all 17 and 92 conditions for Yeast and Lymphoma datasets respectively. For the Yeast dataset the maximum number of genes obtained for this algorithm in all the 17 conditions is 127 with MSR value 199.9656. The maximum available in all the literature published so far is in multi-objective PSO [15]. They obtained 141 genes for 17 conditions with MSR value 203.25. For Lymphoma dataset the maximum number of conditions obtained is only 84 for multi-objective PSO but in this case the maximum number of conditions obtained is 92. Hence this algorithm has a comparative differential advantage over the previous ones.

Table 5. Performance comparison between BISIMT and other algorithms for Human Lymphoma dataset

Algorithm	Avg.Gene Num.	Avg. Cond.Num	Avg. Volume	Avg. MSR
BISIMT	98.00	50.88	3481.13	1072.63
Novel Greedy	741.10	38.50	14455.30	1192.43
SEBI	14.07	43.57	615.84	1028.84
CC	269.22	24.50	4595.98	850.04

In the above table the average mean squared residue, the average number of genes and conditions and average volume and are compared for various algorithms. For the average mean squared residue field lower values are better where as higher values are better for all other fields.

Table 6 gives the difference between the biclusters obtained in terms of the number of genes, number of conditions and mean squared residue score for BISIMT and biclustering using MSR difference threshold, starting with the same seed. In the case of biclustering using MSR difference threshold there is a single MSR difference threshold value for the gene list and condition list. There is no iterative search by incrementing the MSR difference threshold value. The parameters for biclustering using MSR difference threshold are condition difference threshold =30 and gene difference threshold=10 and parameters for BISIMT are initial value of condition difference threshold=5, increment=5, final value of condition difference threshold=30. Similarly initial value of gene difference threshold=1, increment=1 and final value of gene difference threshold=10. It is clear that BISMIT produces large size biclusters compared to biclustering using MSR Difference Threshold. Hence iterative search with incremental MSR difference threshold is better than assigning a single value for MSR difference threshold.

Table 6. Comparison of Iterative Search with Incremental MSR difference Threshold and biclustering using a single value for MSR difference threshold for Yeast dataset starting with same seed

Seed Num.	Details about biclusters From BISIMT			Details about biclusters obtained using a single value for MSR difference Threshold		
	Gene Num.	Col. Num.	MSR	Gene Num.	Col. Num.	MSR
1	110	17	199.95	78	16	199.96
2	93	17	199.79	65	17	198.88
3	99	17	199.95	74	17	199.69
4	96	17	199.69	86	17	198.39
5	125	17	199.91	119	17	199.54

5. CONCLUSION

As a powerful analytical tool for the biologists biclustering has generated considerable interest over the past few decades. Biclustering finds application in the gene expressions of cancerous data for the identification of co-regulated genes, gene functional annotation and sample classification. In this paper a novel algorithm is introduced based on the innovative concept of MSR difference threshold. Here biclusters are produced in two steps. In the first step bicluster seeds are obtained using K-Means clustering algorithm. Then these seeds are enlarged by BISIMT algorithm which uses the concept MSR difference threshold. Here we used iterative search to find out a suitable value for MSR threshold by initializing with a small value and incrementing it after each iteration by a certain amount until it reaches the final value. This algorithm is implemented on both the Yeast dataset and Human Lymphoma dataset. On the basis of the algorithm implementation on both the above mentioned benchmark gene expression datasets, a comparative assessment of the results are given for demonstrating the effectiveness of the proposed method. In terms of the average condition number and the average MSR, the biclusters obtained in this method are far better than many of the metaheuristic algorithms. Moreover this method finds high quality biclusters which show strikingly similar up-regulations and down-regulations under a set of experimental conditions that can be inspected visually by using plots.

A bicluster is obtained using msr difference threshold which is shown in Figure 1 for the Yeast dataset with strikingly similar upregulation and down regulation and with a structural appearance which is hitherto unseen in any of the literature published so far. Hence it is concluded that the newly introduced concept of MSR difference threshold is extremely relevant for biclustering gene expression data. However for identifying suitable values for a particular dataset, gene list and condition list, there is scope for further experimentation.

6. REFERENCES

[1] J. A. Hartigan, “Direct clustering of Data Matrix”, Journal of the American Statistical Association Vol.67, no.337, pp. 123-129, 1972.

[2] Yizong Cheng and George M. Church, “Biclustering of expression data”, Proc. 8th Int. Conf. Intelligent Systems for Molecular Biology, pp. 93-103, 2000.

[3] Madeira S. C. and Oliveira A. L., “Biclustering algorithms for Biological Data analysis: a survey” IEEE Transactions on computational biology and bioinformatics, pp. 24-45, 2004.

[4] Anupam Chakraborty and Hitashyam Maka “Biclustering of Gene Expression Data Using GeneticAlgorithm” Proceedings of Computation Intelligence in Bioinformatics and Computational Biology CIBCB, pp. 1-8, 2005.

[5] Chakraborty A. and Maka H., “Biclustering of gene expression data by simulated annealing”, HPCASIA ’05, pp. 627-632, 2005.

[6] Shyama Das and Sumam Mary Idicula “A Novel Approach in Greedy Search Algorithm for Biclustering Gene Expression Data” International Conference on Bioinformatics, Computational and Systems Biology (ICBCSB), WASET, 2009.

[7] Tavazoie S., Hughes J. D., Campbell M. J., Cho R. J. and Church G. M., “Systematic determination of genetic network architecture”, Nat. Genet., vol.22, no.3 pp. 281-285, 1999.

[8] Alizadeh, A. A. et al., “Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling”, Nature Vol.43,no. 6769, pp. 503-11, 2000.

[9] Amos Tanay, Roded Sharan and Ron Shamir, “Discovering Statistically significant Biclusters in Gene Expression Data,” Bioinformatics; vol.18 Suppl 1, pp.S136-44,2000.

[10] SGD GO Termfinder [http://db.yeastgenome.org/cgi bin/ GO/ goTermFinder]

[11] Federico Divina and Jesus S. Aguilar-Ruize, "Biclustering of Expression Data with Evolutionary computation", IEEE Transactions on Knowledge and Data Engineering, Vol. 18, pp. 590-602, 2006.

[12] J. Yang, H. Wang, W. Wang and P. Yu, "Enhanced Biclustering on Expression Data", Proc. Third IEEE Symp. BioInformatics and BioEng. (BIBE'03), pp. 321-327, 2003.

[13] Z. Zhang, A. Teo, B. C. Ooi, K. L. Tan, "Mining deterministic biclusters in gene expression data", In: Proceedings of the fourth IEEE Symposium on Bioinformatics and Bioengineering(BIBE'04), 2004, pp. 283-292, 2004.

[14] Banka H. and Mitra S., "Multi-objective evolutionary biclustering of gene expression data", Journal of Pattern Recognition, Vol.39 pp. 2464-2477, 2006.

[15] Junwan Liu, Zhoujun Lia and Feifei Liu "Multi-objective Particle Swarm Optimization Biclustering of Microarray Data", IEEE International Conference on Bioinformatics and Biomedicine, pp. 363-366, 2008.