Acta Microbiologica et Immunologica Hungarica, 63 (2), pp. 185–201 (2016)
DOI: 10.1556/030.63.2016.2.4

A ROUGH SET BASED RATIONAL CLUSTERING FRAMEWORK FOR DETERMINING CORRELATED GENES

JEBA EMILYN JEYASWAMIDOSS[1]*, KESAVAN THANGARAJ[1], KADARKARAI RAMAR[2] and MUTHUSAMY CHITRA[3]

[1]Sona College of Technology, Salem, Tamilnadu, India
[2]Einstein College of Engineering, Tirunelveli, Tamilnadu, India
[3]IT Department, Sona College of Technology, Tamilnadu, India

(Received: 5 December 2015; accepted: 12 May 2016)

Cluster analysis plays a foremost role in identifying groups of genes that show similar behavior under a set of experimental conditions. Several clustering algorithms have been proposed for identifying gene behaviors and to understand their significance. The principal aim of this work is to develop an intelligent rough clustering technique, which will efficiently remove the irrelevant dimensions in a high-dimensional space and obtain appropriate meaningful clusters. This paper proposes a novel biclustering technique that is based on rough set theory. The proposed algorithm uses correlation coefficient as a similarity measure to simultaneously cluster both the rows and columns of a gene expression data matrix and mean squared residue to generate the initial biclusters. Furthermore, the biclusters are refined to form the lower and upper boundaries by determining the membership of the genes in the clusters using mean squared residue. The algorithm is illustrated with yeast gene expression data and the experiment proves the effectiveness of the method. The main advantage is that it overcomes the problem of selection of initial clusters and also the restriction of one object belonging to only one cluster by allowing overlapping of biclusters.

Keywords: biclustering algorithm, correlation clustering, gene expression data, overlapping biclusters, rough clustering

Introduction

A lot of techniques have emerged for analyzing microarray gene expression data, but clustering proves to be the primary [1] and the most popular approach for analyzing the expressions of thousands of genes and has been successful in many applications [2]. The process of clustering is the assignment of a set of

*Corresponding author; E-mail: jleenasamsy@yahoo.co.in

observations into subsets called clusters so that observations in the same cluster are similar in some sense. The objects within a cluster are highly similar and objects in different clusters are highly dissimilar. This ultimately increases intraclass similarity but decreases interclass similarity. Clustering is a technique of unsupervised learning that does not have the need of prior knowledge of the groups to which the objects or data members belong to. Varieties of clustering algorithms have been proposed for analyzing gene expression data [3]. The conventional clustering algorithms like *k*-means, hierarchical, SOM, and other density-based methods are very common. The results produced by these methods are consistent for microarray experiments performed on homogeneous conditions. However, when the experimental conditions vary to a great extent, the clusters are no longer correct. This led to a promising alternative prototype of clustering, biclustering.

Biclustering algorithms, also referred to as co-clustering, capture consistency exhibited by subset of genes over subset of conditions. An increasing number of biclustering algorithms have also been proposed for identifying gene patterns [4–9]. Most of the above-mentioned algorithms find exclusive biclusters, but most of these biclusters prove inappropriate in the biological context. Since biological processes are dependent on each other, many genes participate in two or more different processes. Each gene therefore should be grouped to multiple biclusters whenever biclusters are identified.

This problem is addressed in the proposed biclustering algorithm by introducing the framework of generalized rough sets into biclustering. The theory of rough sets is an issue of intense significance in computational intelligence research. The extension of this theory into clustering provides a necessary and potentially useful addition to the range of cluster analysis techniques available to researchers. The concept of rough sets has been introduced into clustering lately and a very few clustering algorithms have also been developed based on rough set theory [10–12]. A technique combining *k*-means and rough set approaches proposed in [13] introduced the concept of upper and lower bounds to the *k*-means centroid. An enhancement to this technique was proposed in [14]. But the main drawback is that these techniques do not address the problem of selection of initial parameters.

This work aims in developing a biclustering algorithm that helps in efficiently identifying all subset of genes that exhibit similar patterns under a subset of experimental conditions. The problem of selection of initial seeds is also addressed here and the quality of the overlapping biclusters is refined based on mean squared residue. Moreover, the proposed approach allows us to profit from the major advantages of rough methods [15], over the crisp techniques. One important aspect of rough sets, bearing significant importance in gene expression clustering, is that it facilitates the identification of overlapping clusters. Hence, by

allowing genes to be members of various clusters, rough methods can more suitably predict the complex relations governing gene regulation.

Rough bi-correlation clustering

The structural framework

In this section, we will present the framework and the methodology we follow to determine the biclusters using Pearson's correlation coefficient and mean squared residue. Subsequently, we provide detailed description of the ROBICOR algorithm and how the algorithm is integrated in the rough clustering process to guide clustering. We explain how the algorithm automatically determines the number of clusters present in a dataset and produces biclusters with upper and lower approximations.

The ROBICOR algorithm is designed to be intelligent and more efficient. It is intelligent as it does not require the number of clusters as input. It is more efficient as it uses Pearson's correlation coefficient and mean squared residue for producing high quality overlapping biclusters. The framework of the proposed model is shown in Figure 1. The proposed algorithm is also robust as it handles noisy data.

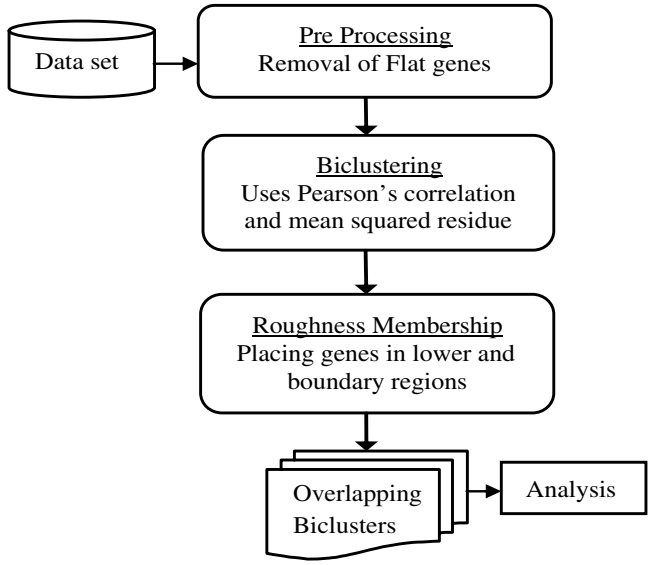

Figure 1. The proposed rough set based model for biclustering

Preprocessing of data

Some genes in the gene expression matrix do not respond much to the experimental conditions and so do not actively participate in the biclustering of the data. These genes are called 'flat genes' and should be removed to provide good quality biclusters. For this, we use the formula proposed by Tang et al. [16]. Each gene vector with j conditions can be represented as $g_i = (e_{i1}, e_{i2}, \ldots e_{ij})$. A vector-cosine can be used to match each gene vector and with a predefined pattern $H = (h_1, h_2, \ldots h_j)$ to determine the deviation in gene intensity values among samples as shown in Equation (1).

$$\cos(\theta) = \sum_{j=1}^{m} e'_{ij} \times h_j \Bigg/ \sqrt{\sum_{j=1}^{m} {e'_{ij}}^2} \times \sqrt{\sum_{j=1}^{m} h_j^2} \tag{1}$$

Both the vectors are said to be more similar if the value of the cosine-vector is close to 1. A threshold value is chosen and the genes which have $\cos(\theta)$ values more than the specified threshold value are removed. This process removes the gene vectors that are more similar to the predefined pattern. The data is now preprocessed and in shape for clustering.

The biclustering algorithm

Usually, gene expression data is arranged in a form of a data matrix. Each row corresponds to one gene and each column to one condition. Each element of this matrix is a real number that represents the expression level of a gene under a specific experimental condition. The value of each element is usually the logarithm of the relative abundance of the mRNA of the gene under the specific condition. Pearson correlation coefficient for measuring similarity between expression patterns of two genes x_i and x_j is defined as

$$\mathrm{Sim}(x_i, x_j) = \sum_{l=1}^{m} (x_{il} - \bar{x}_i)(x_{jl} - \bar{x}_j) \Bigg/ \sqrt{\sum_{l=1}^{m} (x_{il} - \bar{x}_i)^2 \sum_{l=1}^{m} (x_{jl} - \bar{x}_j)2} \tag{2}$$

where x_{il} and x_{jl} are l-th expression values of the i-th and j-th genes, respectively. The terms $\bar{x}_i$ and $\bar{x}_j$ are mean values over m expression values (corresponding to microarray experiments) of the i-th and j-th genes, respectively. The value of m gives the number of conditions or samples under which the genes exhibit the expression patterns.

The proposed ROBICOR algorithm effectively and efficiently approximates a set of overlapping biclusters simultaneously with relative lower mean squared residue. The step 1 of the algorithm produces biclusters for which it uses correlation coefficient metric and mean squared residue. The ROBICOR uses Pearson correlation coefficient for measuring the similarity between expression patterns of two genes x_i and x_j, and is defined in Equation (2). This idea of generating biclusters using correlation coefficient was inferred from the BCCA algorithm proposed by Bhattacharya and De [17]. BCCA uses only Pearson's correlation coefficient to detect the biclusters where as the newly designed ROBICOR uses mean squared residue in addition to Pearson's correlation coefficient to detect biclusters of high quality. The ROBICOR initially starts with a pair of genes and finds the conditions under which they are co-regulated. For any pair of genes (g_i, g_j), the algorithm finds the similarity of the genes under all conditions. $\mathrm{Sim}(x_i,x_j) > \theta$ indicates that x_i and x_j are similarly expressed, i.e., their expression patterns are altering in a similar way. If the similarity is less than θ, then the algorithm finds out the condition, when eliminated gives the maximum increase in the correlation coefficient. That condition is eliminated from the condition set and this step is repeated until the similarity exceeds θ and the number of conditions involved is not less than some specified number of conditions in the condition set. If they are not correlated, it moves on to find the next pair of genes. Otherwise, the algorithm forms a bicluster with the initial two genes and the conditions. The bicluster is further refined by including a new gene based not only on the correlation values with all the other genes in the bicluster, but also on the mean squared residue of the bicluster. When a new gene is added to the bicluster, the mean squared residue of the bicluster is calculated.

Algorithm

```
Step 1: Detect bicluster set Biclust.
Biclust = ∅;
For each pair of genes (gi,gj), i <> j, do:
{
  Set I = (gi,gj) and J = set of all conditions and m = |j|
  While Sim (gi,gj) < θ, gi, gj ∈I and m ≥ r, do:
    {
      From m expression values, find out the elimination of a
      condition y which when eliminated from J will cause
      maximum increase in Sim (xi, xj)
      Remove y from the set J and m = m − 1.
    }
```

```
  If Sim (xi,xj) ≥ θ, for gi, gj ∈ I over m expression values in
  J, where m ≥ r, then
    {
      Remove the set I from X (the set of all genes);
      For each gp ∈ X, do:
      {
        If Sim(gi,gp) ≥ θ, for all gi ∈ I over m expression
        values in J, and If K_gp ≤ δ, then set (I = I ∪ {g_p})
        Remove g_p from the Set X;
      }
      c.Set ccount = ccount+1; Biclust = Biclust ∪ I;
    }
}
Step 2: Detect upper and lower approximations
For each bicluster Bi ∈ Biclust = {B1,B2,...,Bn}
{
  For each object v in bicluster B_j do
  {
    If v ∈ one and only bicluster B_j and If K'_Xj ≤ δ, then
    {v belongs to the lower bound of B_j.}
    Else
    {Compute the difference in the mean squared residue for
      each bicluster (v inserted and removed)
    Let dmin be the minimum mean squared residue.
    Find the ratio between dmin and mean squared residue
      of other clusters
    If the ratio <= ω, add the cluster to set P.
    If P ≠ ∅, insert v to boundary region bicluster with
      dmin and all B̄_j with j ∈ P;
    Else insert v to the lower bound of the bicluster with
      dmin.
    }
  }
}
```

The roughness measure

The mean squared residue of a bicluster (*I*,*J*) as defined by Cheng and Church [18] is

$$K(I,J) = \frac{1}{|I| \cdot |J|} \sum_{i \in I, j \in J} r_{ij}^2, \tag{3}$$

where the residue

$$r_{ij} = d_{ij} - d_{iJ} - d_{Ij} + d_{IJ} \tag{4}$$

is an indicator of the degree of coherence of an entry with remaining entries of a bicluster. Also the base of the gene g_i is defined as

$$d_{iJ} = \sum_{j \in J} d_{ij} / |J| \tag{5}$$

And the base of the condition C_i is defined as

$$d_{Ij} = \sum_{i \in I} d_{ij} / |I| \tag{6}$$

And the base of the bicluster d_{IJ} is defined as

$$d_{IJ} = \frac{\sum_{i \in I, j \in J} d_{ij}}{|I| \cdot |J|} \tag{7}$$

Only if the mean squared residue is less than δ, the gene is placed in the bicluster. Thus, the algorithm reduces the possibility of misplacing a gene in a bicluster. Furthermore, the lower the mean squared residue, the stronger is the coherence exhibited by the bicluster. The mean squared residue well indicates the general coherence of a bicluster. The lower the mean squared residue, the higher is the quality of the bicluster. By the end of step 1, the possible number of high quality biclusters and the objects in each bicluster are obtained.

Based on the concepts of rough sets [16, 19], we can consider each bicluster as a generalized rough set with two approximations, a lower bound and an upper bound. The genes or conditions of the lower approximation belong only to the bicluster, whereas the members of the upper approximation may belong to one or more biclusters. This property leads to overlapping among corresponding biclusters. Given a gene expression data matrix R, for each object (gene or condition), there are three possibilities in the bicluster membership:

- not belonging to any biclusters in R or
- belonging to the lower approximation of the bicluster or
- belonging to the upper approximation of the bicluster in R.

The step 2 of the algorithm gives the procedure to place the genes in the upper and lower approximations for which it uses the ratio of the mean squared residue of the biclusters to which the gene belongs to. To determine the bicluster membership, we follow the following procedure: For each object vector v, an element of $S = \{C_1, C_2, \ldots, C_n\}$, where $C_1, \ldots, C_n$ represents the biclusters generated, find the difference in the mean squared residue before and after the removal of v using Equation 8.

$$\Delta K(v, X_j) = K'(x_j) - K(x_j). \tag{8}$$

Let $K'(x_j)$ and $K(x_j)$ be the mean squared residue of the biclusters after and before v is removed from the bicluster X_j, respectively. Find the minimum of this value dmin.

$$\text{dmin} = \min_{1 \le j \le k} \Delta K(v, X_j) \tag{9}$$

Using Equation (9), the bicluster that has the minimum mean squared residue when gene v is inserted into it is found. Using Equation (10), the ratio of the bicluster (R) with minimum mean squared residue and others is found.

$$R = \Delta K(v, X_j) / \Delta K(v, X_i) \tag{10}$$

Equation (10) helps to resolve the membership of the gene v. Let

$$D = \{j | \Delta K(v, X_i) / \Delta K(v, X_i) \le \omega, i \ne j\} \tag{11}$$

i.e., the set D consists of all biclusters for which the ratio R is less than ω. Furthermore, if $D = \varnothing$, then v is placed in the upper boundary of all biclusters present in the set D. Otherwise, if $D = \varnothing$, then v is placed in the lower boundary of the bicluster which has the minimum mean squared residue.

The parameters ω and δ used in this procedure are predefined thresholds. The parameter δ is to make sure that all biclusters discovered have mean squared residues less than δ to improve cluster quality. The parameter ω determines the degree of overlapping among these biclusters. The set D is calculated using the formula given in Equation (11). The concept of using mean squared residue for rough biclustering was proposed by Wang et al. [13]. In the proposed method, we have used Pearson's correlation and mean squared residue for biclustering, and we

use the ratio of the mean squared residue of the clusters for finding lower and upper approximations.

The initial part of the proposed algorithm ROBICOR generates the number of biclusters. Then the algorithm goes one step further to find the quality of the biclusters generated and also the upper and lower bounds for each bicluster. We have used mean squared residue to determine the bicluster quality and the membership of objects in the lower and upper approximation of the bicluster. The ratio between the mean squared residue of a bicluster and the volume of the bicluster depicts the overall quality of the bicluster. The average of this ratio is also found to decide about the degree of overlapping.

Performance evaluation

The size of the biclusters obtained by ROBICOR depends on the correlation threshold value θ. The optimum correlation threshold value was selected by varying correlation threshold between 0 and 1. This process is very time consuming. The algorithm was experimented for θ values in the range {0.72, 0.74, 0.76, 0.78, 0.80, 0.82, 0.84, 0.86, 0.88}. This variation in the cluster accuracy is depicted using line graph of Figure 2. It has been noted that the relative accuracy (relative accuracy is the accuracy of the algorithm represented in percentage) of the algorithm is 83% when the value of θ is 0.80. As the algorithm yields better results when θ is 0.8, we have chosen the threshold value to be 0.80.

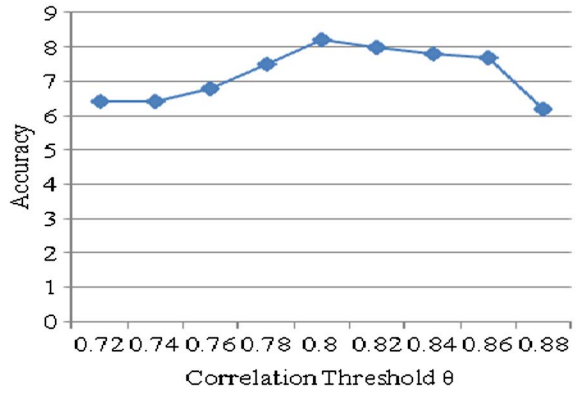

Figure 2. The relative accuracy of ROBICOR for various values of correlation threshold θ

The degree of overlapping between the biclusters is determined by the parameter ω. A range of values {0.25, 0.3, 0.35, 0.4, 0.45, 5, 0.55, 0.6, 0.65, 0.7) were experimented for the datasets. Figure 3 shows the degree of overlapping of the clusters for different values of ω. It has been noted that a value of 0.6 for ω yields an optimal result. Moreover, it is interesting to note that our algorithm delivers meaningful results over the range [0.5, 0.7] of ω, where the overlapping degree increases dramatically and stabilizes as shown in Figure 3. The values for the parameters adopted in ROBICOR are presented in Table I.

The mean squared residue of the biclusters produced by the proposed algorithm is analyzed and evaluated. The lower the mean squared residue, the higher is the quality of the bicluster. It has been found that for most of the biclusters, the mean squared residue value falls below 300. The threshold value 300 is chosen as stated in [19, 20]. Figure 4 shows the mean squared residue for the biclusters produced by ROBICOR for the yeast dataset. It can be observed that the mean squared residue of most of the biclusters falls below 350.

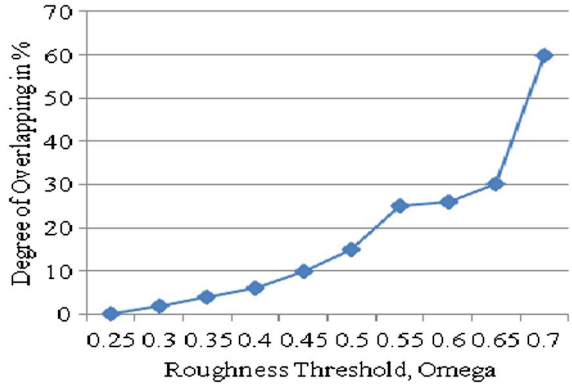

Figure 3. Relative increase in the degree of overlapping when varying ω, the roughness threshold for ROBICOR algorithm

Table I. Optimum values for the parameters used in ROBICOR

Procedure	Parameter	Value
Generating the initial biclusters	Threshold for correlation coefficient θ	0.8
Rough clustering	Overlapping threshold ω	0.6
	Mean squared residue threshold	300

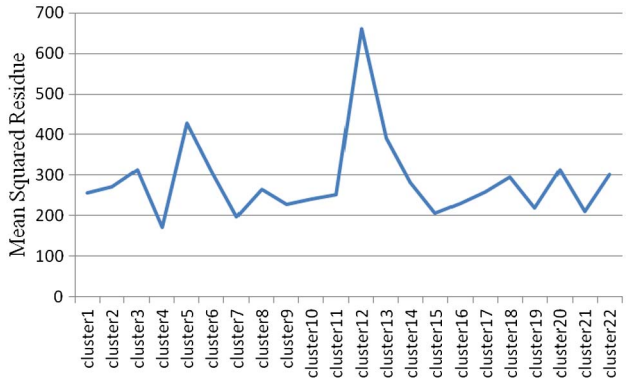

Figure 4. Mean squared residue of the biclusters detected using ROBICOR algorithm

Results

The performance of the proposed algorithm was experimented with two different sets of data. The different data sets, namely yeast gene expression data, colon cancer data, and leukemia dataset were considered for experimentation. The data set is 384×17 matrix. A total of 384 genes were clustered based on 17 experimental conditions. Next, the algorithm was experimented with colon cancer data set which contains expression levels of 2,000 genes taken from 62 different samples out of which 50 genes were chosen across all 62 samples. When applied on 384×17 yeast data matrix, it produced 450 biclusters and when applied on 500×36 colon data matrix, 322 biclusters were produced.

The cluster profile plot of four biclusters depicting the expression level of genes in each bicluster generated by ROBICOR algorithm when applied on yeast and colon expression data is shown in Figures 5 and 6. The profile plot of only four randomly selected clusters is shown to depict the accuracy of the algorithm. Moreover, we can also observe that the genes in the biclusters are highly correlated as their profile patterns are varying similarly. The genes falling in the lower and boundary regions are depicted with the color difference in the profile pattern. The profile pattern of genes in boundary region is depicted with red lines and in the lower approximation is depicted with blue lines. The placement of genes in lower and boundary regions of biclusters based on the mean squared residue is presented in Table II. The membership of a few randomly selected genes in three different biclusters based on their mean squared residue values is clearly presented in Table II. The resulting biclusters with the details of number of genes and conditions, number of genes in the lower approximation and boundary region are shown in Table III. The algorithm depicts and differentiates the members certainly classified as the member of the clusters/biclusters and the members those possibly belong to the clusters/biclusters.

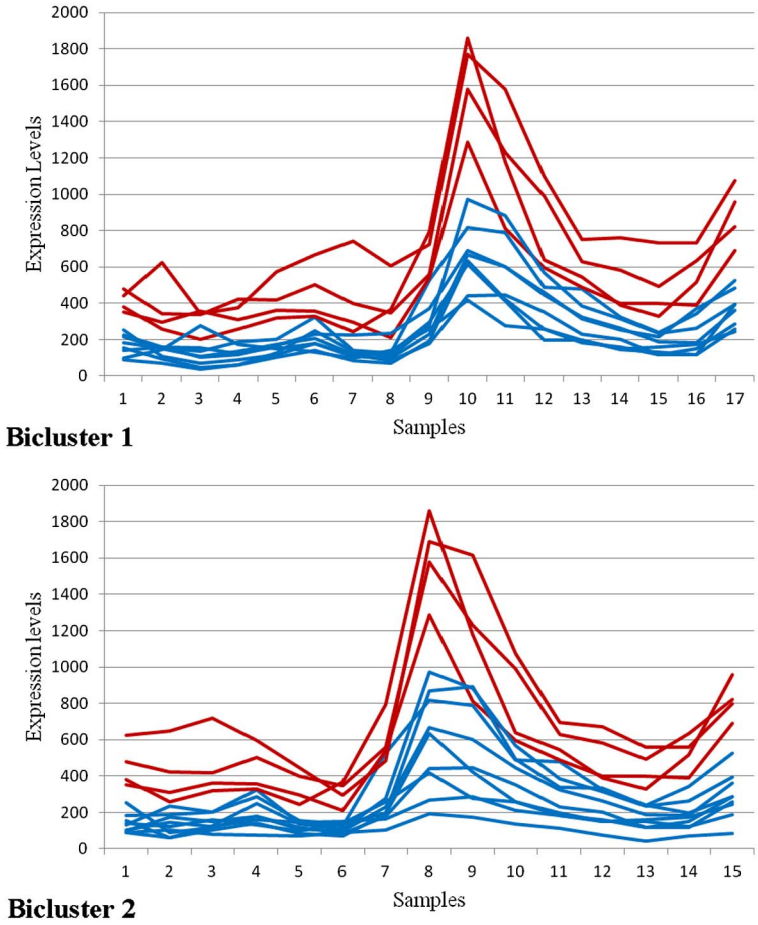

Figure 5. Cluster profile plot of bicluster1 and bicluster2 produced by ROBICOR when applied on colon gene expression

Performance metrics

For evaluating the performance of the biclustering algorithm ROBICOR, Adjusted Rand index (ARI), Silhouettes index (SI), and Davies–Bouldin (DB) index are used. ARI is applied on a 50×10 synthetic data set while SI and DB index are applied on both artificial and real datasets. The average ARI and SI values are reported in Table IV for 10 runs of each algorithm. The value N in the table indicates the number of clusters. The results indicate that for the synthetic dataset, the proposed ROBICOR shows significant improvement in the ARI and SI values when compared with other clustering and biclustering algorithms. The SI and DB index of ROBICOR for the three real datasets is compared with the other algorithms in Table V. The

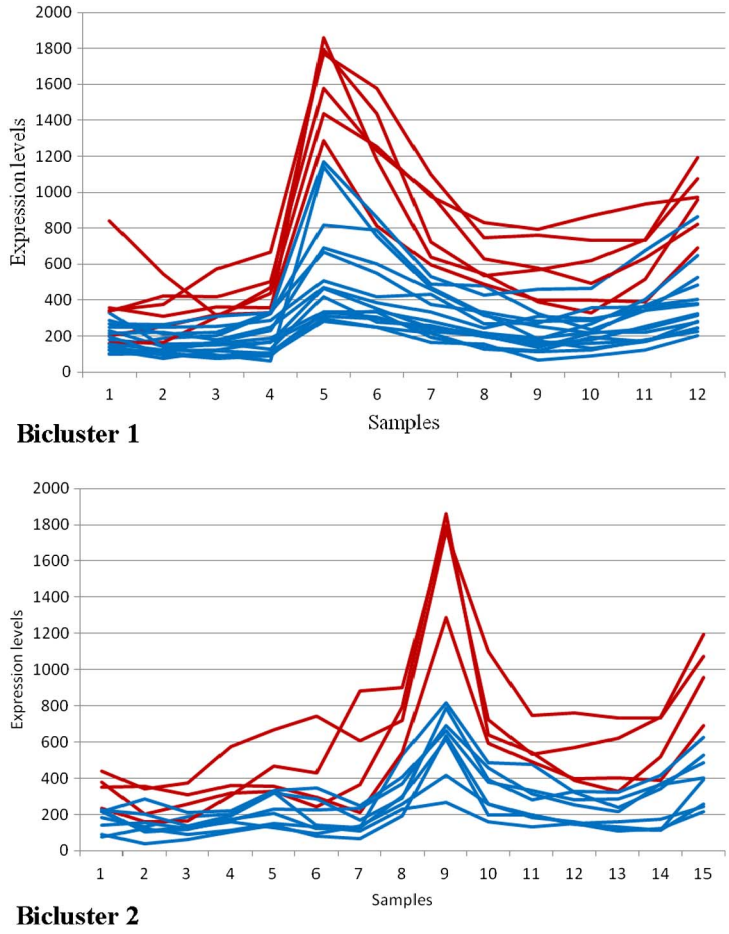

Figure 6. Cluster profile plot of bicluster1 and bicluster2 produced by ROBICOR when applied on yeast gene expression

results show that SI values of ROBICOR algorithm are closer to 1 when applied on the three different real data sets. It has also been observed that the DB index of ROBICOR is minimum when compared with the other biclustering algorithms.

Test for statistical significance

The *p*-values produced by Wilcoxon's rank sum test are calculated for all algorithms participating in the comparison. The ARI scores for the artificial data and SI scores for the real data sets are recorded for 10 consecutive runs of the algorithms. For the null hypothesis, it is assumed that the median values of two groups show no

Table II. Membership of genes in yeast expression data. The ratio between the mean squared residue of the bicluster with minimum mean squared residue and other mean squared residue value determines the membership of a gene

Gene ID	Mean squared residue of biclusters			Membership of the gene
	Cluster 24	Cluster 55	Cluster 86	
G89	240.26	121.12	124.03	Placed in the boundary region of Cluster 24, Cluster 55, and Cluster 86
G107	130.36	290.16	321.00	Placed in the lower approximation of Cluster 24
G260	319.93	155.37	171.83	Placed in the boundary region of Cluster 55 and Cluster 86
G301	146.39	149.63	394.26	Placed in the boundary region of Cluster 24 and Cluster 55

Table III. Number of genes in the lower and boundary region of clusters produced by ROBICOR when applied on yeast expression data

	C4	C15	C26	C48	C75
Total number of genes	17	20	8	5	15
Total number of conditions/attributes	8	15	6	10	4
Number of genes in lower approximation	3	7	5	3	6
Number of genes in boundary region	14	13	3	2	11

Table IV. Comparison of ROBICOR with other algorithms in terms of ARI and SI for synthetic dataset

Algorithm	*N*	ARI	SI
ROBICOR	22	0.6455	0.5799
BCCA	18	0.5548	0.5022
ROB	10	0.5466	0.4099
CC	10	0.5126	0.3716
SCAD	10	0.4713	0.3111
Rough *k*-means	10	0.5492	0.3875

significant difference between them and the alternative hypothesis is that the median values of the two groups show significant difference in them. Table VI reports *p*-values produced by Wilcoxon's rank sum test for comparison of two groups at a time. All the *p*-values reported in the table are *p*-values received when ROBICOR is compared with other algorithms (ROBICOR as group one and the other algorithm as

Table V. Performance comparison of ROBICOR with other algorithms in terms of SI and DB index

Dataset	Clustering algorithm	Number of clusters	Silhouettes index	DB index
Yeast	ROBICOR	350	0.6649	1.7462
	BCCA	326	0.5612	1.8333
	ROB	50	0.5044	2.0666
	CC	50	0.4151	2.1264
	SCAD	50	0.5136	2.0122
	Rough *k*-means	50	0.5632	1.9121
Colon cancer	ROBICOR	239	0.5497	1.8847
	BCCA	209	0.5029	1.8997
	ROB	50	0.4222	2.2744
	CC	50	0.3766	2.8654
	SCAD	50	0.4245	2.3148
	Rough *k*-means	50	0.4144	1.9788
Leukemia	ROBICOR	308	0.5842	1.7113
	BCCA	287	0.5766	1.8411
	ROB	50	0.5111	2.1688
	CC	50	0.4933	2.7613
	SCAD	50	0.4288	2.4254
	Rough *k*-means	50	0.4155	1.9142

group two). It is clearly evident from the values that all the *p*-values are less than 0.05 (5% significance level). It has also been noted that the median values of ROBICOR algorithm are better compared with all other algorithms. The small value of *p*-values (less than 0.05) is a strong proof against the null hypothesis.

Table VI. *p*-values of comparing ROBICOR with other algorithms

Dataset	*p*-value			
	BCCA	ROB	CC	SCAD
Artificial dataset	$4.7 \times E^{-4}$	$4.6 \times E^{-4}$	$4.2 \times E^{-5}$	$5.2 \times E^{-5}$
Yeast	$1.7 \times E^{-4}$	$4.3 \times E^{-4}$	$2.8 \times E^{-4}$	$4.5 \times E^{-4}$
Colon cancer	$2.4 \times E^{-3}$	$3.7 \times E^{-5}$	$1.7 \times E^{-3}$	$2.6 \times E^{-5}$
Leukemia	$3.7 \times E^{-4}$	$5.2 \times E^{-5}$	$3.5 \times E^{-5}$	$3.4 \times E^{-4}$

Conclusion

Here, we have proposed and developed a biclustering algorithm called ROBICOR based on Pearson correlation coefficient and mean squared residue as a similarity measure. The algorithm finds group of genes that show similar pattern in their expression profiles over a subset of conditions. The results clearly demonstrate that the genes in a bicluster obtained by ROBICOR are not only highly correlated but the clusters are also highly coherent. Our method also finds a set of biclusters with a reasonable degree of overlapping associating each bicluster with a lower and an upper approximation. The proposed method is found to be more efficient than many existing biclustering algorithms.

Appendix

The algorithm ROBICOR is implemented in C language. To invoke the algorithm written in C language in java interface, Java Native Interface (JNI) has been used. JNI is a mechanism that allows a Java program to call a function in a C or C++ program. For all the C functions, shared library files are created with .dll extension or .so (linux) extension. The native method has been declared in Java and the shared library files have been loaded before the native method is called. A C header file containing function prototypes for the native methods has also been created. To improve the efficiency of the C programs, the input text file is converted into binary file and every access to the input file is done on the binary file.

Conflict of Interest

The authors declare no conflict of interest.

References

1. Wang, H., Wang, Z., Li, X., Gong, B., Feng, L., Zhou, Y.: A robust approach based on Weibull distribution for clustering gene expression data. Algorithms Mol Bio **6**, 6–14 (2011).
2. Stekel, D: Microarray Bioinform. Cambridge University Press, Cambridge, UK, 2006.
3. Jiang, D., Tang, C., Zhang, A.: Cluster analysis for gene expression data: A survey. IEEE Trans Knowledge Data Eng **16**, 1370–1386 (2004).
4. Madeira, S. C., Oliveira, A. L.: Biclustering algorithms for biological data analysis: A survey. IEEE/ACM Trans Comput Biol Bioinform **1**, 24–45 (2004).

5. Yang, E., Foteinou, P. T., King, K. R., Yarmush, M. L., Androulakis, I. P.: A novel non-overlapping bi-clustering algorithm for network generation using living cell array data. Bioinformatics **23**, 2306 (2007).
6. Pensa, R. G., Boulicaut, J.-F.: Constrained co-clustering of gene expression data. In Proceedings of the 2008 SIAM International Conference on Data Mining, 2008.
7. Tsai, C.-Y., Chiu, C.-C.: A novel microarray biclustering algorithm. Int J Math Comput Phys Elect Comput Eng **4**, 256 (2010).
8. Prelic, A., Bleuler, S., Zimmermann, P., Wille, A., Bühlmann, P., Gruissem, W., Hennig, L., Thiele, L., Zitzler, E.: A systematic comparison and evaluation of biclustering methods for gene expression data. Bioinformatics **22**, 1122–1129 (2006).
9. Frigui, H., Nasraoui, O.: Unsupervised learning of prototypes and attribute weights. Pattern Recogn **37**, 567–581 (2004).
10. Emilyn, J. J., Ramar, K.: An Intelligent mining framework based on rough sets for clustering gene expression data. J Appl Sci **12**, 1932–1938 (2012).
11. Shi, P.: Clustering fuzzy web transactions with rough *k*-means. In AST 09 Proceedings of the 2009 International e-Conference on Advancd Science and Technology, IEEE Computer Society, Washington, DC, 2009, pp. 48–51.
12. Wang, R., Miao, D., Li, G., Zhang, H.: Rough overlapping biclustering of gene expression data. In Proceedings of the 7th IEEE International Conference on Bioinformatics and Bioengineering, Harvard Medical School, Boston, MA, 2007, pp. 828–834.
13. Lingras, P., West, C., Interval set clustering of web users with rough *k*-means. J Intel Inf Syst **23**, 5–16 (2004).
14. Peters, G.: Some refinements of rough *k*-means clustering. Pattern Recogn **39**, 1481–1491 (2006).
15. Lingras, P., Yan, P. R., Hogo, M.: Rough set based clustering: Evolutionary, neural, and statistical approaches. In Proceedings of the First Indian International Conference on Artificial Intelligence, IICAI, Hyderabad, India, 2003, pp. 1074–1087.
16. Tang, C., Zhang, L., Zhang, A., Ranmanathan, M.: Interrelated two-way clustering: An unsupervised approach for gene expression data analysis. In Proceedings of the 2nd IEEE International Symposium on Bioinformatics and Bioengineering, 2001, pp. 41–48.
17. Bhattacharya, A., De, R. K.: Bi-correlation clustering algorithm for determining a set of co-regulated genes. Bioinformatics **25**, 2795–2280 (2009).
18. Cheng, Y., Church, G. M.: Biclustering of expression data. Proc Int Conf Intell Syst Mol Biol **8**, 93–103 (2000).
19. Pawlak, Z.: Rough sets. Int J Comput Inform Sci **2**, 341–356 (1982).
20. Jiong, Y., Haixun, W., Wei, W., Philip, Yu., Uiuc, I., Chapel, U., Hill, I., Watson, T. J.: Enhanced biclustering on expression data. In Proceedings of 3rd IEEE Symposium on BioInformatics and BioEngineering, BIBE, Bethesda, MD, 2003.