Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 12 (18): 1932-1938, 2012
ISSN 1812-5654 / DOI: 10.3923/jas.2012.1932.1938

An Intelligent Mining Framework based on Rough Sets for Clustering Gene Expression Data

[1]J. Jeba Emilyn and [2]K. Ramar
[1]Department of IT, Sona College of Technology, Salem, Tamil Nadu, India
[2]Einstein College of Engineering, Tirunelveli, Tamil Nadu, India

Abstract: The main aim of this study is not only to develop a biclustering algorithm that would successfully identify gene patterns but also to propose an intelligent clustering framework that would improve the cluster quality. Our framework for mining co-regulated genes from gene expression dataset is composed of three important steps: a preprocessing step to refine the data, an intelligent procedure to predict the possible number of biclusters and a procedure based on rough sets to cluster the gene datasets. Our algorithm is said to be intelligent, in the sense that it can predict the possible number of biclusters. Since, the algorithm is based on rough sets, there are high possibilities of placing a gene in more than one bicluster and thus allows overlapping of biclusters. A theoretical understanding of the proposed algorithm is analyzed and results are illustrated with different gene expression data sets. The analysis and the experiment shows that the method is more intelligent and efficient.

Key words: Biclustering algorithm, gene expression data, membership matrix, overlapping biclusters, rough clustering

INTRODUCTION

The gene expression profile is a representation of the complex mechanism behind cancer and the changes in gene expression levels are very common in complex diseases like cancer. Microarray technology is considered as an enhancement to simultaneously observe the expression levels of thousands of genes across collections of related samples. The significant task in analyzing gene expression data is identification of co-expressed genes and the coherent gene expression pattern (Ben-Dor *et al*., 1999; Eisen *et al*., 1998).

Numerous clustering algorithms that can be applied in different fields have been proposed, analyzed and improved (Jiang *et al*., 2004). The conventional algorithms like *k*-means, hierarchical, SOM and other density based methods are very common. These algorithms have their own merits and demerits. Hemalatha and Vivekanandan (2008) have proposed an enhanced version of *k*-means clustering algorithm which is claimed to be parallel and distributed. Garg and Jain (2006) have done a comparison on some of the existing variations of k-mean algorithms. They have used the synthetic sets of high dimensional data as benchmark for evaluating the algorithms and have also proposed some criteria for comparison of these clustering algorithms. Ranjan and Khalil (2007) have worked with the statistical approaches in hierarchical clustering and have also done a comparison on the linkage methods which can assist us in knowing the functionalities of many genes. Vijendra (2011) has presented a detailed review of various subspace and density based clustering algorithms, their efficiencies and inefficiencies on different data sets. Zhou *et al*. (2007) have proposed a Join-Prune algorithm that shows momentous gain in runtime and quality.

Recently a series of pattern based clustering methods have been proposed to capture coherence exhibited by subset of genes over subset of conditions. An increasing number of biclustering algorithms have also been proposed for identifying gene patterns (Madeira and Oliveira, 2004). An iterative co-clustering algorithm that mainly concentrates on user defined constraints and minimizes the sum squared residue was addressed in (Pensa and Boulicaut, 2008). *k*-biclusters clustering (KBC Algorithm) suggested by Tsai and Chiu (2010) aims in minimizing the dissimilarities between genes and bicluster centers thereby minimizing the residue within the clusters. It also tries to involve as many conditions as possible in each iteration of clustering. Of late the concept of Rough sets has also been introduced into clustering and a few clustering algorithms have been developed based on rough set theory (Prelic *et al*., 2006; Emilyn and Ramar, 2010; Shi, 2009). Arora *et al*. (2009) have proposed an integrated approach for filtering all the non-reduct

Corresponding Author: J. Jeba Emilyn, Department of IT, Sona College of Technology, Salem 636005, Tamil Nadu, India

attributes using rough set theory. A Rough Overlapping Biclustering (ROB) algorithm proposed by Wang *et al.* (2007) also works on the framework of generalized rough sets.

The principal aim of this work is in developing an algorithm based on intelligent rough clustering techniques which will efficiently mine co-regulated genes from gene expression dataset by removing the irrelevant dimensions in a high dimensional space and obtain appropriate meaningful clusters.

MATERIALS AND METHODS

In this study, we have proposed a framework for mining co-regulated genes from gene expression dataset. The framework is composed of three sections: 1) a preprocessing step to refine the data so that all the biclusters generated would be meaningful (2) a procedure to determine the possible number of biclusters and (3) a biclustering algorithm based on rough sets. The structure of the proposed framework is shown in Fig. 1.

Preprocessing of data: In gene expression matrix, the intensity values of different genes vary widely. This difference may be due to the influence of different gene dimensions. In order to handle this variation, the values have to be normalized. Normalization is done using Eq. 1:

$$e'_{ij} = \frac{e_{ij} - \mu_i}{\mu_i} \quad (1)$$

Where:

$$\mu_i = \frac{1}{m}\sum_{j=1}^{m} e_{ij}$$

where, e'_{ij} represents normalized intensity value for gene i under condition j and e_{ij} denotes the original intensity value for gene i under condition j, m is the number of conditions and μ_i is the mean of the intensity value.

Some genes in the gene expression matrix do not react much to the experimental conditions and so show very less significance in biclustering the data. These genes named as 'flat genes' should be removed to provide good quality biclusters. For this, we follow the method proposed by Tang *et al.* (2001). After normalization each gene vector with j conditions can be represented as $g_i = (e'_{i1}, e'_{i2},...e'_{ij})$. A vector-cosine can be used between each gene vector and a pre-defined stable pattern $H = (h_1, h_2,....h_j)$ to determine the variation in the gene intensity values among samples:

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

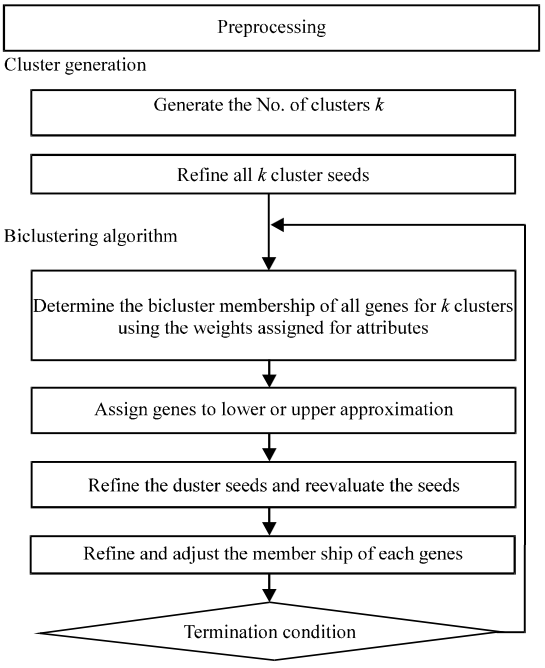

Fig. 1: The structure of rough bi clustering algorithm (ROBICA)

If the value of the cosine-vector is close to 1, then both the vector are more similar. A threshold value can be chosen and genes which have θ values more than the threshold δ (those that are more similar to the pattern) can be removed. The data is now considered to be ready for clustering.

Procedure to detect possible number of biclusters: Biclustering is the problem of finding a subset of the vectors (genes) that express a similar pattern over a subset of the dimensions. The problem requires grouping the vectors and the dimensions at the same time, thus, the name "biclustering". In our algorithm we define a bicluster based on the Pearson's correlation coefficient (Bhattacharya and De, 2009) as the similarity measure. Pearson's correlation coefficient for measuring the similarity between two genes (x_i, x_j) is given as:

$$Sim(x_i, x_j) = \sum_{i=1}^{m} (x_{(j)1} - \bar{x}_i)(x_{jl} - \bar{x}_j / \sqrt{\sum_{1=1}^{m} (x_{il} - \bar{x}_i)^2 \sum_{1=1}^{m} (x_{jl} - \bar{x}j)^2} \quad (2)$$

The procedure for finding the possible number of biclusters is shown in Fig. 2. Initially, all conditions are considered in the condition set for any pair of genes. Then the algorithm finds out the condition which when eliminated gives the maximum correlation. That condition is eliminated from the condition set and repeat this step until you have not less than some specified number of conditions in the condition set. Repeat step B for all genes and add them to this cluster. When there are no more genes to be added increment the cluster count by 1. Repeat the entire process (steps B and C) for the next two pair of genes until there is no more pair to be considered. The variable count will give the possible number of biclusters.

Rough biclustering algorithm (ROBICA): The proposed new algorithm, Rough Biclustering Algorithm (ROBICA), clusters genes based on rough set theory. The main advantage of this method is that it not only places each gene in the corresponding bicluster but also assigns a weight for different conditions depending upon the significance of the condition in the bicluster. Genes can get expressed in two are more clusters i.e. overlapping of genes are possible. The membership matrix, weight matrix and the center matrix are calculated similar to that of SCAD algorithm (Frigui and Nasraoui, 2004).

The equation for finding the membership matrix U given W and O is presented as:

$$U_{ik} = \frac{\left(\sum_{i=1}^{J} (w_{kj})^{\hat{a}} \times (e_{ij} - O_{kj})^2\right)^{\frac{1}{1-\alpha}}}{\sum_{t=1}^{K} \left[\left(\sum_{j=1}^{J} (w_{tj})^{\hat{a}} \times (e_{ij} - O_{tj})^2\right)^{\frac{1}{1-\alpha}}\right]} \quad (3)$$

where, $1 \le i \le I$ and $1 \le k \le K$. The equation for finding the bicluster centroids given W and U can be shown as:

$$O_{kj} = \frac{\sum_{i=1}^{1} (U_{ik})^{\alpha} \times e_{ij}}{\sum_{i=1}^{1} (U_{ik})^{\alpha}} \quad (4)$$

And finally, the equation for calculating and updating the weights for the J conditions can be given as:

$$W_{kj} = \frac{\left(\sum_{i=1}^{I} (U_{ik})^{\alpha} \times (e_{ij} - O_{kj})^2\right)^{\frac{1}{1-\hat{a}}}}{\sum_{t=1}^{J} \left[\left(\sum_{i=1}^{I} (U_{ik})^{\alpha} \times (e_{it} - O_{kt})^2\right)^{\frac{1}{1-\hat{a}}}\right]} \quad (5)$$

```
Input: Gene expression matrix
Procedure:
Step. 1: Initially set X = set of all genes and set count = 0;
Step. 2: For each pair of genes (gi, gj), I<>j, do:
        Set I = (gi, gj) and J = set of all condition 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 (gi, gj);
                Set J =J-{y} and m = m-1;
        If Sim (gi, gj)≥θ, for gi, gj  I over m expression values in J, where m≥r, then
                Set X = X-I;
                For each gp X do:
                        If Sim (gi, gp)≥θ, for all gi ∈ I over m expression values in J, then set {I = I {gp};
                        Set X = X-{gp};
                Set count = count+1;
```

Fig. 2: Algorithm for detecting the number of bicluster, The similarity between any two genes Sim (g_i, g_j) is calculated using correlation coefficient, This algorithm returns the possible number of biclusters as output

Input: Gene expression matrix, count, threshold
Procedure:
Step 1: Initialize W and O
Step 2: Calculate the bicluster membership of all the i genes for all *k* biclusters using the equation (3)
Step 3: Assign each gene to the lower approximation or the upper approximation by using the membership value. i.e.:
If the value uij<λ, gene is in the lower approximation
If the value uij<λ, uij>ω gene is in the lower approximation
If the value uij<ω, then gene does not belong to this bicuster
Step 4: Compute new bicluster centers O for each cluster *k* using (4)
Step 5: Reevaluate the weights of the j condition for the *k* biclusters using (5)
Step 6: Iterate until all bicluster centers in O remain the same without any change

Fig. 3: Pseudo code of ROBICA, The algorithm generates both upper and lower approximations for each cluster using the values of the membership matrix

Initially, the values for W are generated randomly whereas the center values O for the k biclusters are got by finding the mean of all the genes assigned to the biclusters in the above procedure. Afterwards, the ROBICA algorithm repeats to calculate the membership matrix, weight matrix and centers using Eq. 3-5 until all *k* bicluster centers in O remain the same without being changed. The dissimilarity between gene G_i and object center O_k is defined as:

$$f_{sim}(G_i, O_k) = \sum_{j=1}^{J} (W_{kj})^{\alpha} (e_{ij} - O_{kj})^2 \quad (6)$$

where, e_{ij} is the expression level of gene G_i in condition C_j and W_{jk} gives the weight assigned for all j conditions in *k* biclusters. The pseudo-code of the ROBICA is illustrated in Fig. 3. Through the ROBICA algorithm, the genes belonging to the bicluster can be identified based on the elements u_{ik} for I = 1,…, I in U. If the membership values fall above the threshold λ, the gene is put in the lower approximation. If the value is between λ and ù , then the gene is placed in the upper approximation. If the value falls below ù , the gene is not in the bicluster. Similarly, the conditions used for the bicluster can be identified based on the elements w_{jk} for j = 1,… , J in W. If w_{kj} = 1/J in W, then the experimental condition belongs to the bicluster B_k. If w_{kj}<1/J, the condition does not belong to B_k. Since, we decide the placement of a gene in a bicluster based on the membership values, there are high possibilities of placing a gene in more than one bicluster. This is how this algorithm allows overlapping of biclusters. Therefore, the goal of biclustering has been achieved by the ROBICA algorithm.

The algorithm generates the membership matrix based on the rough set theory (Pawlak, 1982). The lower approximation is a subset of the upper approximation. The members (genes or conditions) of the lower approximation belong to and only belong to the bicluster. However, the members of the upper approximation may belong to more than one biclusters. The boundary region between the lower and upper approximation forms an overlapping part among corresponding biclusters. Therefore, it is anticipated that lower and upper approximation resulting directly from expression data would better capture the overlapping feature among the co-regulated genes.

RESULTS

The performance of the proposed ROBICA Algorithm was experimented with two different sets of data. Initially the algorithm was experimented with yeast expression data downloaded from http://faculty.washington.edu /kayee/model. 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 2000 genes taken from 62 different samples out of which 50 genes where chosen across all 62 samples.

As most clustering methods, the proposed ROBICA uses several parameter to approximate the optimal solution. During data pre-processing procedure, we choose the threshold for the vector-cosine δ to be 0.7 (Table 1). We then remove genes with vector-cosine

Table 1: Parameter settings

Procedure	Parameter	Value
Preprocessing	Threshold for the cosine-vector δ	0.70
Generating the initial biclusters	Threshold for correlation coefficient θ	0.84
Rough biclustering algorithm (ROBICA)	Power of condition weights β	1.50
	Roughness of membership α	2.00
	Lower approximation λ	0.80
	Upper approximation ω	0.50

The table gives the values for the parameters used in the different stages of the algorithm. During the preprocessing step, we use the threshold δ for the vector cosine. The second stage uses a correlation coefficient θ. In the third stage we use the parameters α and β for generating the membership matrix and the parameters λ and ω for finding the lower and upper boundaries of the biclusters

higher than that threshold. The intensity values of 802 genes were found to vary little across the conditions and so were removed from 2884 genes.

After preprocessing, we move on to the generation of the initial biclusters where we use a correlation threshold θ. The selection of optimum correlation threshold value by varying correlation threshold between 0 and 1 and judging each biclustering result takes huge amount of time. Based on the study by Allocco *et al.* (2004), which states that two genes having expression profile correlation >0.84 have more than 50% chance of being bounded by the same transcription factor, we have chosen the threshold value to be 0.85. Further, two parameters α and β have to be assigned for the ROBICA algorithm. The values for the pair (α, β) = (2,1.5) is considered for this algorithm.

The algorithm was experimented for different values of λ and ω. The cluster profile plot in Fig. 4 shows the biclusters generated by ROBICA for colon cancer dataset. Figure 4 represents six different biclusters generated on colon cancer data set by the proposed ROBICA algorithm. Each subdivision (Fig. 4a-f) shows the expression levels of genes that are grouped in separate clusters. In all rough clustering algorithms, the number of objects in the boundary region depends on the value of the threshold λ. It has been noted for our algorithm that the number of genes in the boundary region decreases as the value of λ becomes > 0.8 and ù becomes >0.5. When the threshold value becomes smaller, the number of genes in the boundary region also increases. The accuracy of the algorithm for different values of λ and ω is shown in Fig. 5.

Performance comparison: The performance of ROBICA is compared with a few biclustering algorithms like KBC (Tsai and Chiu, 2010), SCAD(Frigui and Nasraoui, 2004), ROB (Wang *et al.*, 2007), CC (Cheng and Church, 2000),

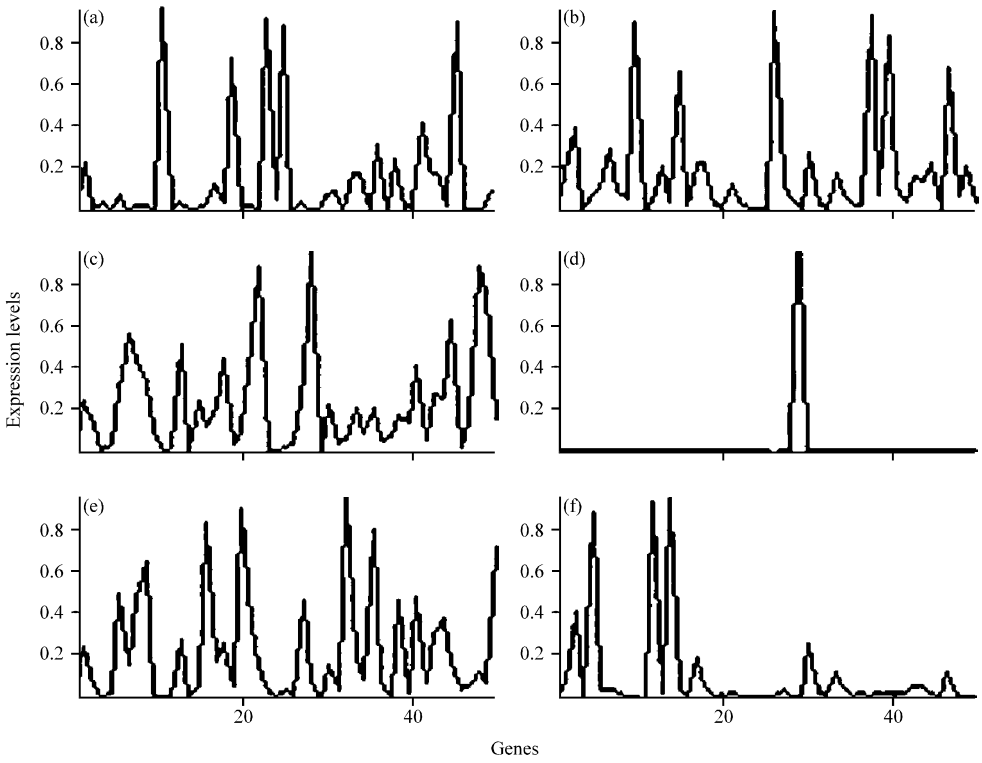

Fig. 4(a-f): Colon cancer data set clustered using ROBICA. Cluster profile plot (with the x-axis representing the genes and the y-axis representing the expression levels) showing the different biclusters generated using ROBICA, (a) Genes grouped into Cluster 1 and their expression levels, (b) Genes grouped into Cluster 2 and their expression levels, (c) Genes grouped into Cluster 3 and their expression levels, (d) Genes grouped into Cluster 4 and their expression levels, (e) Genes grouped into Cluster 5 and their expression levels and (f) Genes grouped into Cluster 6 and their expression levels

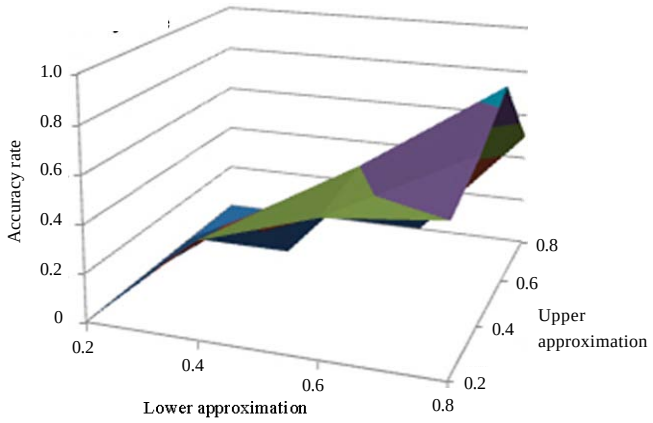

Fig. 5: Accuracy rate of ROBICA for different values of λ and ω figure shows a maximum accuracy for the value (0.8, 0.5)

Table 2: Accuracy comparison for different algorithms for both yeast and colon cancer data sets

		Accuracy rate					
		Yeast data			Colon cancer data		
Algorithm	Initial clusters as input	Maximum	Minimum	Average	Maximum	Minimum	Average
ROBICA	Not required	0.88	0.77	0.82	0.87	0.77	0.820
KBC	Required	0.86	0.76	0.81	0.85	0.75	0.800
SCAD	Required	0.84	0.65	0.74	0.82	0.68	0.750
ROB	Required	0.88	0.77	0.82	0.85	0.75	0.820
CC	Required	0.80	0.66	0.73	0.80	0.61	0.700
Rough *k*-means	Required	0.78	0.74	0.76	0.76	0.75	0.755
RCGED	Required	0.81	0.78	0.795	0.79	0.70	0.745
BCCA	Not required	0.86	0.78	0.82	0.88	0.76	0.820
Bimax	Required	0.84	0.80	0.82	0.82	0.80	0.810

ROBICA: Rough biclustering algorithm, KBC: *k*-biclusters clustering, SCAD: Simultaneous clustering and attribute discrimination algorithm, ROB: Rough overlapping biclustering, CC: Cheng and Church (2000), RCGED: Rough clustering of gene expression data, BCCA: Bi-correlation clustering algorithm, Bimax: Binary inclusion-maximal biclustering algorithm

Rough *k*-means (Pawan and West, 2004), RCGED (Emilyn and Ramar, 2011), BCCA (Bhattacharya and De, 2009), Bimax (Prelic *et al*., 2006) for its accuracy. All of these clustering algorithms except BCCA require the number of clusters to be specified as input. This is not appreciable because small no of cluster centers tend to generate few large clusters and large number of cluster centers generate large number of small clusters. As this may not be accurate, we have come out with a procedure to find out the possible number of biclusters based on the correlation of the genes with one another. These biclusters are refined further in the ROBICA algorithm. Moreover in our method, each and every bicluster is defined with a lower and an upper boundary based on the rough set theory. By doing this, we allow overlapping of biclusters. Gene in the upper boundary of one bicluster can also fall into a boundary of a different bicluster. The maximum and minimum accuracy rate of ROBICA algorithm is compared with the other methods for both the yeast and colon cancer datasets (Table 2).

CONCLUSION

In this article, we propose a novel approach based on rough set theory for clustering gene expression data. It is based on the idea that a group of genes can be clustered together if they exhibit a similar pattern over a subset on experimental conditions. Consequently, we cluster the genes based on the weights assigned to the experimental conditions. A rough set based biclustering algorithm is utilized in the clustering process. The number of clusters in the gene expression data is automatically determined in this biclustering algorithm. We present here, the theoretical understanding, analysis and results of the ROBICA algorithm. Our algorithm proves to be robust as it handles noisy data during the preprocessing step. Then it proves to be intelligent when is automatically detect the possible number of biclusters. It also proves to be a novel method as it allows overlapping of biclusters and also finds the lower and upper boundaries for each bicluster.

REFERENCES

Allocco, D.J., I.S. Kohane and A.J. Butte, 2004. Quantifying the relationship between co-expression, co-regulation and gene function. BMC Bioinformatics. Vol. 8.

Arora, A., S. Upadhyaya and R. Jain, 2009. Integrated approach of reduct and clustering for mining patterns from clusters. Inform. Technol. J., 8: 173-180.

Ben-Dor, A., R. Shamir and Z. Yakhini, 1999. Clustering gene expression patterns. J. Comput. Biol., 6: 281-297.

Bhattacharya, A. and R.K. De, 2009. Bi-correlation clustering algorithm for determining a set of co-regulated genes. Bioinformatics, 25: 2795-2801.

Cheng, Y. and G.M. Church, 2000. Biclustering of expression data. Pcoc. Int. Conf. Intell. Syst. Mol. Biol., 8: 93-103.

Eisen, M.B., P.T. Spellman, P.O. Brown and D. Botstein, 1998. Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Acad. Sci. USA., 95: 14863-14868.

Emilyn, J.J. and K. Ramar, 2010. Rough set based clustering of gene expression data: A survey. Int. J. Eng. Sci. Technol., 2: 7160-7164.

Emilyn, J.J. and K. Ramar, 2011. A rough set based gene expression clustering algorithm. J. Comput. Sci., 7: 986-990.

Frigui, H. and O. Nasraoui, 2004. Unsupervised learning of prototypes and attribute weights. Pattern Recognit., 37: 567-581.

Garg, S. and R.C. Jain, 2006. Variations of K-mean algorithm: A study for high-dimensional large data sets. Inform. Technol. J., 5: 1132-1135.

Hemalatha, M. and K. Vivekanandan, 2008. A semaphore based multiprocessing k-mean algorithm for massive biological data. Asian J. Sci. Res., 1: 444-450.

Jiang, D., C. Tang and A. Zhang, 2004. Cluster analysis for gene expression data: A survey. IEEE Trans. Knowledge Data Eng., 16: 1370-1386.

Madeira, S.C. and A.L. Oliveira, 2004. Biclustering algorithms for biological data analysis: A survey. IEEE/ACM Trans. Comput. Biol. Bioinform., 1: 24-45.

Pawan, L. and C. West, 2004. Interval set clustering of web users with rough K-means. J. Int. Inform. Syst., 23: 5-16.

Pawlak, Z., 1982. Rough sets. Int. J. Comput. Info. Sci., 11: 341-356.

Pensa, R.G. and J.F. Boulicaut, 2008. Constrained co-clustering of gene expression data. Proceedings of the SIAM International Conference on Data Mining, April 24-26, 2008, Villeurbanne, France, pp: 25-36.

Prelic, A., S. Bleuler, P. Zimmermann, A. Wille and P. Buhlmann *et al*., 2006. A systematic comparison and evaluation of biclustering methods for gene expression data. Bioinformatics, 22: 1122-1129.

Ranjan, J. and S. Khalil, 2007. Clustering methods for statistical analysis of genome databases. Inform. Technol. J., 6: 1217-1223.

Shi, P., 2009. Clustering fuzzy web transactions with rough K means. Proceedings of the International e-Conference on Advanced Science and Technology, March 7-9, 2009, Dajeon, pp: 48-51.

Tang, C., L. Zhang, A. Zhang and M. Ranmanathan, 2001. Interrelated two-way clustering: An unsupervised approach for gene expression data analysis. Proceedings of the 2nd IEEE International Symposium on Bioinformatics and Bioengineering, November 4-6, 2001, Bethesda, MD., USA., pp: 41-48.

Tsai, C.Y. and C.C. Chiu, 2010. A novel microarray biclustering algorithm. World Academy Sci. Eng. Technol., 65: 256-262.

Vijendra, S., 2011. Efficient clustering for high dimensional data: Subspace based clustering and density based clustering. Inform. Technol. J., 10: 1092-1105.

Wang, R., D. Miao, G. Li and H. Zhang, 2007. Rough overlapping biclustering of gene expression data. Proceedings of the 7th IEEE International Conference on Bioinformatics and Bioengineering, October 14-17, 2007, Boston, MA., USA., pp: 828-834.

Zhou, H., B. Feng, L. Lv and Y. Hui, 2007. A robust algorithm for subspace clustering of high-dimensional data. Inform. Technol. J., 6: 255-258.