A Granular Self-Organizing Map for Clustering and Gene Selection in Microarray Data

Shubhra Sankar Ray, Avatharam Ganivada, and Sankar K. Pal, *Fellow, IEEE*

***Abstract*— A new granular self-organizing map (GSOM) is developed by integrating the concept of a fuzzy rough set with the SOM. While training the GSOM, the weights of a winning neuron and the neighborhood neurons are updated through a modified learning procedure. The neighborhood is newly defined using the fuzzy rough sets. The clusters (granules) evolved by the GSOM are presented to a decision table as its decision classes. Based on the decision table, a method of gene selection is developed. The effectiveness of the GSOM is shown in both clustering samples and developing an unsupervised fuzzy rough feature selection (UFRFS) method for gene selection in microarray data. While the superior results of the GSOM, as compared with the related clustering methods, are provided in terms of β-index, DB-index, Dunn-index, and fuzzy rough entropy, the genes selected by the UFRFS are not only better in terms of classification accuracy and a feature evaluation index, but also statistically more significant than the related unsupervised methods. The C-codes of the GSOM and UFRFS are available online at http://avatharamg.webs.com/software-code.**

***Index Terms*— Bioinformatics, clustering, feature selection, granular neural network, rough-fuzzy computing.**

I. Introduction

GRANULAR computing (Grc) is a paradigm in which granules are computed by grouping similar type of objects, based on their characteristics like similarity, equality, and proximity. The components of soft computing like fuzzy logic, rough sets, neural networks, their integrations, and many others can be used for Grc. In fuzzy logic, the information granules are characterized by generalized fuzzy constraints [1]. The objects in the granules are assigned with the fuzzy membership values. In rough sets, the crisp information granules are computed for approximating a set. The vagueness of the set is dealt using the lower and upper approximations of the set [2]. In fuzzy rough sets, the fuzzy equivalence classes are based on the tolerance relation or fuzzy reflexive relation. Here, the uncertainty of the set is modeled by the fuzzy lower and upper approximations. In neural networks, the structure of granulation is introduced, typically, by the self-organizing map (SOM) [3]. The present investigation deals with the development of a granular SOM (GSOM) using fuzzy rough sets and a feature selection algorithm based on GSOM for microarray gene expression data analysis. The subsequent paragraphs describe the related works and a brief description of the proposed methods.

In neural networks, the information granules are embedded to develop efficient granular neural networks, which shows high performance and handles uncertainty. While efficiency of the granular neural networks in achieving high performance can be observed by evaluating the clustering solutions or classification results, that in handling uncertainty arising from overlapping class boundaries can be viewed from the data plots in feature space and/or confusion matrix. The existing Grc frameworks [4]–[7] can improve the performance of conventional neural networks by precisely defining the initial connection weights between nodes in different layers through granules, before training. A hierarchical GSOM, involving bidirectional update propagation, is used in [4] for discovering granulation structures. A rough set is mainly characterized by lower and upper approximations and information granules. While the information granulation is exploited for Grc, the lower and upper approximations are used for uncertainty handling. The models in [5]–[7] use the concept of information granules for extracting the domain knowledge. In the process, [5] uses rough sets whereas [6] and [7] use fuzzy rough sets; thereby making the latter models superior in terms of handling uncertainty. The SOM-based models [5] and [7] use Gaussian function for defining the neighborhood. Other rough fuzzy set-based clustering algorithms are rough fuzzy possiblistic c-means (RFPCM) [8], rough possibilistic c-means (RPCM) [8], and robust rough fuzzy c-means algorithm (RRFCM) [9]. Here, the rough fuzzy sets are used to define probabilistic and possibilistic membership values to patterns in a set. In this investigation in designing the GSOM, we exploit the other aspects of rough sets, namely, lower and upper approximations in defining the neighborhood function of SOM based on the concepts of fuzzy rough set for better modeling of the overlapping regions.

Deoxyribonucleic acid (DNA) microarray contains thousands of spots representing genes from different samples. messenger ribonucleic acid are extracted from tumor/normal cells and are converted into cDNAs using reverse transcription. A scanner is used to measure the fluorescence values of tumor cDNA, normal cDNA, and

Manuscript received May 8, 2014; revised September 2, 2014, May 9, 2015, and July 22, 2015; accepted July 23, 2015. Date of publication August 13, 2015; date of current version August 15, 2016.

S. S. Ray and S. K. Pal are with the Machine Intelligence Unit, Center for Soft Computing Research, Indian Statistical Institute, Kolkata 700108, India (e-mail: shubhra@isical.ac.in; sankar@isical.ac.in).

A. Ganivada is with the Center for Soft Computing Research, Indian Statistical Institute, Kolkata 700108, India (e-mail: avatharg@yahoo.co.in).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2015.2460994

2162-237X © 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

complementary sequences of DNA, and used as the expression values for different genes [10]. In general, gene expression data contain higher number of genes and less number of samples (patterns/objects). Some genes may be irrelevant for performing classification, clustering, and feature selection tasks. Hence, selection of relevant genes is an important task for identifying diseases and the underlying biological process.

Gene selection may be viewed as feature selection in pattern recognition framework. The existing methodologies for feature selection involve both classical and modern approaches. A rough set-based method for clustering the genes (features) and selecting the genes in the best cluster, in terms of a cluster validation measure, is developed in [11]. In [12], information measures are developed using fuzzy rough sets. The method involves maximization of relevance and minimization of redundancy among selected genes. In [13], an unsupervised feature selection network based on the minimization of fuzzy feature evaluation index using gradient decent method is developed. Fuzzy rough methodologies for dimensionality reduction are depicted in [14] and [15]. While the method in [14] is a supervised one, the method in [15] works in an unsupervised manner. A popular classical approach for feature selection is reported in [16], where the feature having the farthest distance from its neighboring features in the cluster, evolved by K-nearest neighbor (K-NN) principle, is selected.

In this investigation, a new GSOM and a method for gene selection are introduced. During training of the GSOM for the first iteration, the adjustment of initial weights of the winning neurons and its neighborhood neurons are dependent on the Gaussian neighborhood function as in [3]. For the remaining iterations, the output clusters are presented to a decision table as its decision classes. Based on the decision table, the proposed neighborhood function for updating the connection weights is defined using the lower and upper approximations of fuzzy rough set. The resultant clusters of the GSOM are used in finding the dependence factor for each gene using the concepts of fuzzy rough set, and the genes are ranked according to the increasing values of the dependence factors for selection, say unsupervised fuzzy rough feature selection (UFRFS). Therefore, the novelty of the proposed methods lies in: 1) developing a new neighborhood function using fuzzy rough sets (fuzzy lower and upper approximations) to modify the learning procedure of the SOM for dealing with overlapping regions and 2) ranking genes based on fuzzy rough dependence values derived from the clustering solutions. In the first case, the concept of fuzzy lower and upper approximations is used in granular neural networks for the first time, and its application for designing a neighborhood function for SOM is also unique. In the second case, extracting fuzzy rough dependence values from clustering solutions (in unsupervised manner) is itself a new concept and its application for ranking genes is shown for the first time. Incorporation of fuzziness in the neighborhood function of GSOM handles the uncertainty arising from overlapping regions, and on the top of this, incorporation of lower and upper approximations of rough set theory further helps in determining the exactness in class shapes which is not possible with the conventional SOM for overlapping class boundaries. Beside, the aforesaid conceptwise novelty, the proposed method differs algorithmwise as follows. The methods in [5] and [7] concern with the initialization of the connection weights of SOM using rough rules. On the other hand, the proposed method concerns with the neighborhood function of SOM involving lower and upper approximations of rough sets.

The paper is organized as follows. The preliminaries on fuzzy rough sets are discussed in Section II. The proposed GSOM and the UFRFS for gene selection are described in Sections III and IV, respectively. In Section V, the results of GSOM, in terms of four clustering evaluation measures, and UFRFS, in terms of classification accuracies and feature evaluation measure, are compared with different clustering and feature selection algorithms, respectively. Section VI concludes the present investigation.

II. Preliminaries on Fuzzy Rough Sets

In fuzzy rough sets, the similarity between any two patterns in a set $A \subseteq U$, representing the universe of objects, is modeled by the fuzzy relation R, which is defined as

$$R(x,x) = 1 \quad \text{(reflexive)}$$
$$R(x,y) = R(y,x) \quad \text{(symmetry), and}$$
$$T(R(x,y)R(y,z)) \leq R(x,z) \quad (T-\text{transitivity})$$

for all x, y, and z in A. When the relation R does not satisfy the T-transitivity property with respect to the given t-norm, it is referred as fuzzy similarity relation. The relation R is fuzzy reflexive relation, when it does not satisfy both the symmetry and T-transitivity properties. The fuzzy reflexive relation or fuzzy similarity relation R is used to express the approximation equality between two objects in a set $A \subseteq U$, corresponding to a feature. Different notions to fuzzy rough set, based on the fuzzy similarity relations, are defined in [17] and [18]. A fuzzy rough set, based on fuzzy reflexive relation and fuzzy decision classes, is characterized by rough lower and upper approximations. The fuzzy reflexive relation and fuzzy decision classes are based on the fuzzy decision system. These are defined in [19] and discussed as follows.

A. Decision System

Let $S = (U, \mathcal{F} \cup \{d\})$ denote a decision system. Here, $\mathcal{F}$ represents the features, say $\{a_1, a_2, \ldots, a_n\}$. The crisp decision classes are represented by $\{d\} = \{X_k,\ k = 1, 2, \ldots, c\}$, where c denotes the number of decision classes and X_k is a decision attribute, labeled with c-values (labeled classes). The decision classes are characterized by its decision attribute X_k. The features and decision classes are used to compute the fuzzy reflexive relation and fuzzy decision classes.

B. Fuzzy Reflexive Relation

A fuzzy reflexive relation R_a, between any two patterns x and y in U, with respect to a feature $a \in \mathcal{F}$, is defined

in [7] as

$$R_a(x,y)=\begin{cases}\max\left(\min\left(\dfrac{a(y)-a(x)+\sigma_{a_{k_1}}}{\sigma_{a_{k_1}}},\dfrac{a(x)-a(y)+\sigma_{a_{k_1}}}{\sigma_{a_{k_1}}}\right),0\right)\\ \quad \text{if } a(x),a(y)\in R_d(X_{k_1})\\ \max\left(\min\left(\dfrac{a(y)-a(x)+\sigma_{a_{k_2}}}{\sigma_{a_{k_2}}},\dfrac{a(x)-a(y)+\sigma_{a_{k_2}}}{\sigma_{a_{k_2}}}\right),0\right)\\ \quad \text{if } a(x)\in R_d(X_{k_1}), a(y)\in R_d(X_{k_2})\\ \quad \text{and } \ k_1\neq k_2\end{cases} \tag{1}$$

where k_1 and $k_2 = 1, 2, \ldots, c$, and $\sigma_{a_{k_1}}$ and $\sigma_{a_{k_2}}$ represent the standard deviation for all the patterns in the classes k_1 and k_2, corresponding to the decision attributes X_{k_1} and X_{k_2}.

C. Defining Decision Classes Using Fuzzy Sets

Let O_{kj} and V_{kj}, $j = 1, 2, \ldots, n$, denote mean and standard deviation, respectively, of the patterns belonging to the kth class. The weighted distance Z_{ik} of a pattern $\overrightarrow{x}_i$, $i = 1, 2, \ldots, s$, (where s is the total number of patterns), from the kth decision class is defined as

$$Z_{ik}=\sqrt{\sum_{j=1}^{n}\left[\frac{x_{ij}-O_{kj}}{V_{kj}}\right]^2},\quad \text{for } \ k=1,2,\ldots,c \tag{2}$$

where x_{ij} represents the jth feature of the ith pattern.

For the ith pattern in the kth class, the membership value is defined as

$$\mu_k(\vec{x}_i)=\frac{1}{1+\left(\frac{Z_{ik}}{f_d}\right)^{f_e}} \tag{3}$$

where f_e and f_d are fuzzifiers. The values of f_e and f_d are chosen to be 1 and 5, respectively, as in [20]. When the decision attribute is quantitative, the fuzzy decision classes are defined as follows.

1) The membership values of all the patterns in the kth class to its own class are defined as

$$D_{kk}=\mu_k(\overrightarrow{x}_i),\ \text{if } \ k=l \tag{4}$$

where $\mu_k(\overrightarrow{x}_i)$ represents the membership value of the ith pattern to the kth class.

2) The membership values of all patterns in the kth class to other classes are defined as

$$D_{kl}=1,\quad \text{if } \ k\neq l \tag{5}$$

where k and $l = 1, 2, \ldots, c$. For any two patterns x and $y \in U$, with respect to a feature $a \in \{d\}$, the fuzzy decision classes are defined as

$$R_d(x,y)=\begin{cases}D_{kk}, \text{ if } \ a(x)=a(y)\\ D_{kl}, \text{ otherwise.}\end{cases} \tag{6}$$

D. Lower and Upper Approximations

The lower approximation of a set $A \subseteq U$, denoted by $(R_a \downarrow R_d)$, is defined in [19] as

$$(R_a\downarrow R_d)(x)=\min\{\underline{\gamma}(x),\underline{\gamma}^c(x)\} \tag{7}$$

where

$$\underline{\gamma}(x)=\inf_{y\in A}\{R_a(x,y)\cdot R_d(x,y)\} \tag{8}$$

$$\underline{\gamma}^c(x)=\inf_{y\in U-A}\{\max(1-R_a(x,y),R_d(x,y))\} \tag{9}$$

for all $x \in A$. The upper approximation of a set $A \subseteq U$, denoted by $(R_a \uparrow R_d)$, is defined in [19] as

$$(R_a\uparrow R_d)(x)=\max\{\overline{\gamma}(x),\overline{\gamma}^c(x)\} \tag{10}$$

where

$$\overline{\gamma}(x)=\sup_{y\in A}\{1-R_a(x,y)+(R_a(x,y).R_d(x,y))\} \tag{11}$$

and

$$\overline{\gamma}^c(x)=\sup_{y\in U-A}\{\min(R_a(x,y),R_d(x,y))\} \tag{12}$$

for all x in A.

For any $B \subseteq \mathcal{F}$ and for $x \in U$, the degree of dependence of γ, depending on the set of features $B \subseteq \mathcal{F}$, is defined as

$$\gamma_B=\frac{\sum_{x\in U}(R_B\downarrow R_d)x}{|U|} \tag{13}$$

where $|\cdot|$ denotes the cardinality of a set U, and the value of γ is $0 \leq \gamma \leq 1$. A lower dependence degree for a feature/gene signifies that the gene is the best for selection.

III. Proposed Granular Self-Organizing Map

The concepts of fuzzy rough set and SOM [3] are used to develop the new GSOM. In this regard, every entry in the input data is initially normalized within 0–1 by subtracting the minimum value and dividing that with maximum–minimum value in the entire data. The normalization is performed for scaling the features into the range [0–1] as the concepts of fuzzy rough set, involving the lower and upper approximations of a set, are based on the values of features lying within 0–1.

A. Methodology for Granular Self-Organizing Map

In the proposed GSOM, the number of nodes in the input layer, say n, is set equivalent to the number of features/attributes (genes for microarray data) and in the output layer, it is set equal to be c-number of clusters, chosen by the user. Let $\{x\} \in \mathbf{R}^n$ denote the set of n-dimensional input patterns. The weights, w_{kj}, $k = 1, 2, \ldots, c$ and $j = 1, 2, \ldots, n$, connecting the links between the n nodes in the input layer and the c nodes in the output layer, are initialized with the random numbers within 0–0.5. Let t denote the number of iterations for training GSOM. For the first iteration, $t = 1$, the GSOM uses the conventional SOM. The following steps are involved in the training of the GSOM.

Step 1: Present an input vector, $x_j(t)$, $j = 1, 2, \ldots, n$, at the nodes in the input layer of GSOM.

Step 2: Find the Euclidian distances between the input vector, $x_j(t)$, and weight vector w_{kj} for the kth output node using

$$d_k = \| x_j(t) - w_{kj}(t) \|^2 . \tag{14}$$

Step 3: Find the winner neuron υ using

$$\upsilon = \text{argmin}\{d_k\}, k = 1, 2, \ldots, c. \tag{15}$$

Step 4: Find neighborhood neurons, N_υ, around the winner neuron υ using the Gaussian neighborhood function

$$N_\upsilon(t) = \exp\left(\frac{-\bigwedge_{\upsilon k}^2}{2\sigma(t)^2}\right) \tag{16}$$

where $\bigwedge_{\upsilon k}(t)$ is used as in [3], and σ is defined in [3] as

$$\sigma(t) = \sigma_0 \exp\left(-\left(\frac{t}{\tau_1}\right)\right). \tag{17}$$

Step 5: Update the connection weights of the neighborhood neurons N_υ using

$$w_{kj}(t+1) = \begin{cases} w_{kj}(t) + \alpha(t) N_\upsilon(t)(x_j(t) - w_{kj}(t)) \\ \qquad\qquad \text{if} \quad k \in N_\upsilon(t) \\ w_{kj}(t), \qquad\quad \text{else.} \end{cases} \tag{18}$$

The learning parameter α in (18) is defined in [3] as

$$\alpha(t) = \alpha_0 \exp\left(-\left(\frac{t}{\tau_2}\right)\right) \tag{19}$$

where α_0 is chosen between 0 and 1, and time constant τ_2 is set equal to the total number of iterations.

The aforesaid steps are repeated for all the input patterns during training. After the first iteration, GSOM partitions the data into c clusters which are employed in defining the new neighborhood function using the lower and upper approximations of fuzzy rough set. Note that, Gaussian neighborhood may not be efficient for handling the uncertainty in the overlapping cluster boundaries as there is no concept of fuzzyness.

1) Procedure for Defining Neighborhood Function: The aforesaid c-clusters obtained at c nodes are labeled with positive integers, representing the crisp decision classes. From the crisp decision classes, we formulate the decision system $S = (U, \mathcal{F} \cup \{d\})$ using fuzzy reflexive relations [see (1)] and fuzzy decision classes [see (6)]. Here, $\mathcal{F} = \{a_1, a_2, \ldots, a_n\}$ is represented by n-features (n-genes in microarray data) and $\{d\}$ indicates the crisp decision classes. The steps for defining neighborhood function are as follows.

Step 1: Find fuzzy decision classes (6), corresponding to the crisp decision classes, based on the decision table S.

Step 2: For every feature a_j, $j = 1, 2, \ldots, n$, compute fuzzy reflexive relational matrix, using (1).

Step 3: Compute membership values belonging to lower and upper approximations of the patterns in set A (each output cluster of SOM), using (7) and (10) and denote them by $L_{kj}(A)$ and $U_{kj}(A)$, $k = 1, 2, \ldots, c$, respectively.

Step 4: For every feature a_j, $j = 1, 2, \ldots, n$, find the averages of all membership values in lower approximation and in upper approximation of the set A, and represent by $L_{kj}^{\text{avg}}(A)$ and $U_{kj}^{\text{avg}}(A)$, $k = 1, 2, \ldots, c$, respectively.

Step 5: For every feature a_j, $j = 1, 2, \ldots, n$, find boundary region of the set A, denoted by $\mathcal{B}_{kj}^{\text{avg}}(A)$, by computing $(U_{kj}^{\text{avg}}(A) - L_{kj}^{\text{avg}}(A))$, $k = 1, 2, \ldots, c$.

Step 6: Define the neighborhood function NH_{kj} for the kth node, $k = 1, 2, \ldots, c$, as

$$\text{NH}_{kj} = e^{-\left(L_{kj}^{\text{avg}} + \mathcal{B}_{kj}^{\text{avg}}\right)^2/2}, \quad j = 1, 2, \ldots, n. \tag{20}$$

The significance of the neighborhood function is that the uncertainty arising in overlapping regions between cluster boundaries is handled efficiently.

The sum of the average membership values in the lower approximation and boundary region of all features (n) for neuron υ, denoted by LBS_υ, is then computed for defining its neighborhood neuron, say k, as

$$\text{LBS}_k = \sum_{j=1}^{n} \left(L_{kj}^{\text{avg}} + \mathcal{B}_{kj}^{\text{avg}}\right). \tag{21}$$

Now, the neuron k is considered as the neighborhood neuron of υ, when $\text{LBS}_k \leq \text{LBS}_\upsilon$, where the LBS values are different for each neuron and dependent on the number of patterns assigned to that cluster.

2) Algorithm for Training GSOM: For the second and subsequent iterations, the following steps are repeated for all input patterns used in the training of GSOM.

1) Present an n-dimensional vector $x(t)$ at the input nodes.
2) Compute the Euclidian distances (14) between the input vector $x(t)$ and the weight vector of output nodes.
3) Find the winning neuron υ, using (15).
4) Find neurons which are lying within the neighborhood of υ, using (21).
5) Modify the connection weights of the winning neuron υ and its neighborhood neurons using

$$w_{kj}(t+1) = \begin{cases} w_{kj}(t) + \alpha(t)\text{NH}_{\upsilon j}(t)(x_j(t) - w_{kj}(t)) \\ \qquad\qquad \text{if } k \in \text{NH}_{\upsilon j}(t) \\ w_{kj}(t), \qquad \text{else} \end{cases} \tag{22}$$

where $\alpha(t)$ is the learning rate [see (19)] and $t = 2, 3, \ldots$

The value of $\alpha(t)$ will be monotonically decreased, while the number of iterations, t, is gradually increased. It may be noted that, the neighborhood function is newly defined for every iteration (other than the first iteration), based on the output clusters of GSOM, attained in the previous iteration.

3) Properties of the Proposed Neighborhood Function:

Maximality: The neighborhood function attains maximum value 1 when the value of $(L_{kj}^{\text{avg}} + \mathcal{B}_{kj}^{\text{avg}})$ is zero.

Nonnegativity: $\text{NH}_\upsilon(A) \geq 0$ if the values of patterns in A are greater than or equal to zero.

Continuity: $\text{NH}_\upsilon(A)$ is a continuous function for all the values of patterns in $A \in [0\text{–}1]$.

Sharpness: $\text{NH}_\upsilon(A)$ is close to 0 if the value of $(L_{kj}^{\text{avg}} + \mathcal{B}_{kj}^{\text{avg}})$ is close to 1.

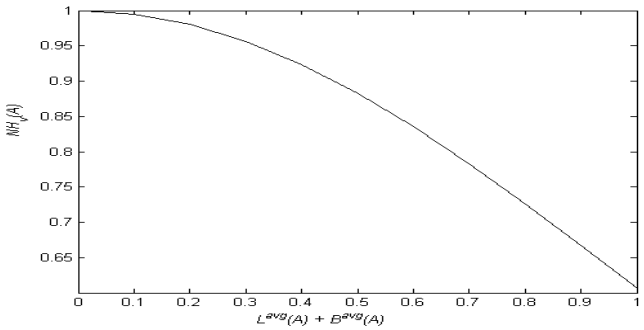

Fig. 1. Variation of the proposed neighborhood function NH_v for different values of $(L^{\mathrm{avg}}(A) + \mathcal{B}^{\mathrm{avg}}(A))$, chosen between 0 and 1.

Convergence: Gauss–Seidel algorithm is used in several cases [8], [21] to establish the convergence of clustering algorithms. The Gauss–Seidel algorithm, representing a set of equations in a matrix form, is assured to converge when every equation in the matrix is diagonally dominant. Based on this principle, the convergence of the GSOM is discussed. After training of GSOM is completed, a confusion matrix $M_{c\times c}$ of the size $c \times c$ is formed by the output clusters of GSOM, where c indicates the number of clusters. Therefore, the matrix M would be

$$M = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1c} \\ A_{21} & A_{22} & \dots & A_{21} \\ A_{c1} & A_{c2} & \dots & A_{cc} \end{pmatrix}$$

where each row in the matrix M can represent an equation. For row l_1 and column $l_2 = 1, 2, \dots, c$, when $l_1 = l_2$ then $A_{l_1 l_2}$ represent the cardinality of the diagonal elements, and when $l_1 \neq l_2$ then $A_{l_1 l_2}$ denote the cardinality of non diagonal elements, in l_1th row of l_1th cluster. When the cardinality of the diagonal element in every row is nonzero and the cardinality of non diagonal elements is zero or less than that of the diagonal element in that row, then the diagonal element in every row of the matrix M is dominant entry and the algorithm is guaranteed to converge.

IV. Proposed Algorithm for Gene Selection in Microarray Data

The selection of the relevant genes is very important in gene expression data for classification or clustering purpose. In this regard, an unsupervised fuzzy rough feature selection algorithm (UFRFS) for feature/gene selection is described. The steps involved in UFRFS are as follows.

Step 1: Partition all samples of gene expression data into c-clusters using GSOM.

Step 2: Present the c-granules with labeled values, representing c-crisp decision classes, to the decision table $S = (U, \mathcal{F} \cup \{d\})$, $\mathcal{F} = \{a_1, a_2, \dots, a_n\}$.

Step 3: Find fuzzy decision classes, using (6), corresponding to the crisp decision classes, based on the decision table S (see Section II-C).

Step 4: Compute fuzzy reflexive relational matrix, using (1), corresponding to each feature/gene a_j, $j = 1, 2, \dots, n$ (see Section II-A).

TABLE I
Characteristics of Microarray Gene Expression Data Sets

Data name	Abbrevi-ation	No. of Samples	No. of Genes	No. of Categories
ALL & AML data [22]	-	38	5000	2
Ethanol (GDS3707)	-	16	18952	2
Cigarette smokers male (GDS3709)	-	40	54675	2
Prostate cancer [23]	-	102	12600	2
DLBCL-A [24]	-	141	661	3
Cigarette smokers female (GDS3709)	-	39	54675	2
Upf1 null mutant (GDS1611)	-	96	9335	2
Diabetic(GDS3715)	-	30	12626	2
Multi-A [25]	-	103	5565	4
Breast Cancer [26]	-	98	1213	3
Lung cancer [27]	-	197	1000	4
Resistance (GDS3715)	-	40	12626	2
Carcinoma data [28]		36	7457	2
Pediatric cerebral palsy (GDS4353)	PCP	20	22277	2
Pediatric development control (GDS4353)	PDC	20	22277	2
Nasal lavage cells (GDS4424)	NLC	32	32408	2
Unresectable colorectal cancer(GDS4393)	URCC	33	54675	2
Quadriceps muscles (GDS4345)	-	24	54675	2
Patient derived colorectal cancer explants (GDS4351)	PDCCE	64	54675	2
Colorectal cancer tumors (GDS4382)	CCT	34	54675	2
J774. A1 macrophage cells (GDS4432)	-	36	45101	2
Atopic dermatitis patients (GDS4444)	ADP	35	54675	2
Peripheral blood mononuclear cells (GDS4407)	PBMCs	30	54675	3
Relaxation response practice effect on blood (GDS3416)	RRPEB	72	54675	3
Cigarette smoke of light flavor effect (GDS3494)	CSLFE	48	54675	3
HepG2 liver cells (GDS3640)	HLC	98	20228	4
Phagocytosis (GDS4438)	PC	18	18950	2
Pancreatic ductal adenocarcinoma (GDS4336)	PDAC	90	28869	2
Primary bovine mammary gland epithelial cells(GDS4437)	PBMEC	52	24128	2
Peripheral white blood cells-A (GDS4395)	PWBC-A	40	54675	4
Peripheral white blood cells-B (GDS4395)	PWBC-B	40	54675	4

Step 5: Calculate membership values of the patterns in the set (cluster) for belonging to lower approximation using (7), corresponding to each feature a_j.

Step 6: Find the dependence value, using (13), for each a_j.

Step 7: Arrange the genes in increasing order according to their dependence values.

Step 8: Select the top-N number of genes with the lower dependence values and, further, use in experiments.

V. Experimental Results

The GSOM and the UFRFS algorithm are coded in c-language using Intel Core i7 CPU 880 at 3.07 GHZ processor and 16 GB RAM. The experiments are carried out on 31 gene expression data whose characteristics are shown in Table I. The accession numbers for data sets, downloaded from Gene Expression Omnibus

(http://www.ncbi.nlm.nih.gov/sites/GDSbrowser), are shown within parenthesis. The data sets have small number of samples ranging from 8 to 197 and large number of genes ranging from 661 to 54 675. The details of the data sets are available at http://avatharamg.webs.com/GSOM-UFRFS.pdf.

A. Algorithms for Comparison and Implementation Issues

The GSOM is compared with seven clustering algorithms like RRFCM [9], RFPCM [8], RPCM [8], FCM, partition around medoids (c-medoids), affinity propagation clustering algorithm (AP method) [29] and SOM. The clustering algorithms used for comparison are either state-of-the-art, or highly cited, or have components similar to the proposed method. While, FCM and SOM fulfill the last two criteria, c-medoids is a highly cited one. The RRFCM, RFPCM, and RPCM are state-of-the-art clustering methods and also use similar components like fuzzy sets and rough sets, in a different manner. The AP method is the state-of-the-art algorithm and uses the concept of numerical methods. The performance of all the clustering algorithms is evaluated with four different cluster validation measures, β-index [30], DB-index [31], Dunn-index [32], and fuzzy rough entropy (FRE) [7]. For a cluster, the lower values of DB-index and FRE and higher values of β-index and Dunn-index signify that the cluster is better. While β-index, DB-index, and Dunn-index are widely studied and they evaluate the compactness of output clusters in terms of low intracluster distances and high intercluster distances, the FRE is a recent one [7] and it is a rough-fuzzy set based measure that evaluates the clusters in terms of minimum roughness (1 − (|lower approxi.|/|upper approxi.|). Further, DB-index and Dunn-index are based on maximum or minimum principle computed over the distance between clusters and variance, taking two clusters together. On the other hand, β-index considers only the variance computed over individual class and the overall feature space.

The proposed UFRFS for gene selection is compared with unsupervised feature selection using feature similarity (UFSFS) [16], unsupervised fuzzy rough dimensionality reduction (UFRDR) [15] and Algorithm 1, which is designed by replacing GSOM with SOM in UFRFS. Moreover, our unsupervised UFRFS is compared with the supervised fuzzy rough mutual information-based method (FRMIM) [12] and correlation-based feature selection (CFS) algorithm [33] to demonstrate its effectiveness as compared with the supervised methods. Among the gene selection methods, the UFSFS and CFS are popular (highly cited) feature selection methods and UFRDR and FRMIM are the state-of-the-art methods based on the concepts of fuzzy rough set as used in ours. The genes selected by these algorithms are evaluated using three different classifiers like fuzzy rough granular neural network (FRGNN) [6], fuzzy multilayer perceptron (FMLP) [20], and K-NN, and a feature evaluation index using entropy measure [16]. The details of all the methods for comparison and evaluation and the way of determining number of hidden nodes in FRGNN and FMLP and the value of K in K-NN are available in supplementary file http://avatharamg.webs.com/GSOM-UFRFS.pdf.

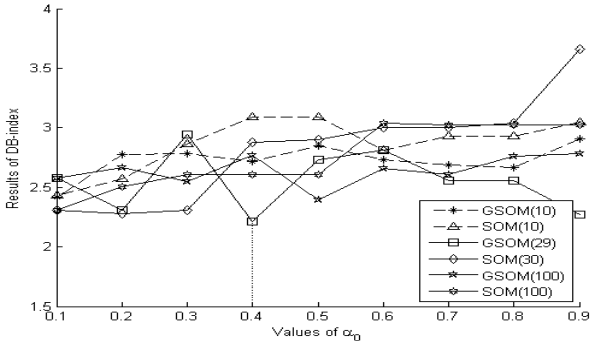

Fig. 2. Variation of DB-index with the values of learning parameter (α_0) for a fixed number of iterations using GSOM and SOM for ALL and AML data.

B. Results of Clustering

Before we explain the results of clustering, the formation of GSOM is explained for ALL and AML data, as an example.

Example: The number of input nodes in GSOM is considered as 5000 as there are 5000 genes in ALL and AML data. The number of output nodes is set equal to 2. The initial connection weights between nodes in the input and hidden layers are initialized by random numbers chosen within 0 to 0.5. Then the GSOM is trained using normalized data within 0 to 1. At first iteration ($t = 1$), the GSOM partitions the data into two granules ($c = 2$), based on the competitive learning of SOM (see Steps S1–S4 in Section III-A).

For the second iteration and thereafter, the training of GSOM is performed using the algorithm in Section III-A2. The samples in the two clusters, obtained from the first iteration, are labeled with integers 1 and 2 indicating two crisp decision classes. The data are then presented to the decision table. The fuzzy decision classes are computed in accordance with the crisp decision classes (see Section II-C). Based on the decision table, the neighborhood function for every neuron is defined using the procedure in Section III-A1, thereby an equation (21) for finding the neighborhood neurons and (22) for updating the connection weights are defined.

1) Selection of Parameters of GSOM and SOM: Five parameters, initial learning α_0, iteration t, time constants τ_1 and τ_2, and initial radius σ_0, are used in the GSOM and SOM. Here, τ_2 is set equal to the total number of iterations, τ_1 (17) is dependent on τ_2, and σ_0 (17) is chosen as the maximum number of neurons in either row or column of the output layer. Considering initial number of clusters $c = 2$ and $\sigma_0 = 2$, we proceed as follows for selecting α_0 for different values of t.

For both GSOM and SOM, the value of α_0 is varied within 0 to 1 in steps of 0.05 for a particular number of iterations (t). Then, we repeat the same process for other values of t ranging from 10 to 100 in steps of 10. Fig. 2 shows the variation of DB-index with α_0 for various values of t. For clarity of presentation, only some curves corresponding to selected α_0 and t are provided. The best results for GSOM are obtained for $\alpha_0 = 0.4$ and $t = 29$ for ALL and AML data and hence, a dotted vertical line is drawn in the figure at $\alpha_0 = 0.4$. In general, the value of t is varied in steps of 10, but the GSOM

TABLE II

COMPARISON OF THE CLUSTERING RESULTS OF GSOM AND SOM. HERE, c = NUMBER OF CLUSTERS, t = NUMBER OF ITERATIONS, AND α_0 = LEARNING PARAMETER

Data set	Method	DB-index	Dunn-index	Fuzzy Rough Entropy(FRE)	DB-index	Dunn-index	Fuzzy Rough Entropy(FRE)	DB-index	Dunn-index	Fuzzy Rough Entropy(FRE)
		c = **2**			c = 4			c = 6		
ALL&	GSOM	2.2150	0.855	0.3687	2.3612	0.7631	0.4139	2.6819	0.5188	0.4325
		t = 29, α_0 = 0.4			t = 50, α_0 = 0.5			t = 35, α_0 = 0.89		
AML	SOM	2.3074	0.8438	0.3751	2.5626	0.5934	0.4434	2.7112	0.4170	0.4494
		t = 30, α_0 = 0.3			t = 150, α_0 = 0.62			t = 200, α_0 = 0.95		
Prostate	GSOM	1.5127	0.9533	0.2939	2.0575	0.5055	0.3024	2.9182	0.3615	0.2984
		t = 30, α_0 = 0.6			t = 30, α_0 = 0.6			t = 15, α_0 = 0.82		
cancer	SOM	1.6772	0.9190	0.3147	2.1854	0.3473	0.3245	2.9891	0.2814	0.3572
		t = 100, α_0 = 0.95			t = 100, α_0 = 0.95			t = 300, α_0 = 0.95		
Diabetic	GSOM	2.2733	0.6069	0.2105	2.7219	0.3642	0.2404	2.4453	0.2777	0.2514
		t = 31, α_0 = 0.16			t = 40, α_0 = 0.82			t = 5, α_0 = 0.22		
	SOM	2.8713	0.5700	0.2540	3.0705	0.3240	0.2552	2.5816	0.2568	0.2605
		t = 1000, α_0 = 0.855			t = 400, α_0 = 0.85			t = 410, α_0 = 0.95		
Upf1	GSOM	1.8599	1.0643	0.1796	2.1128	0.7722	0.2002	2.3603	0.6585	0.2223
		t = 98, α_0 = 0.0555			t = 10, α_0 = 0.25			t = 50, α_0 = 0.55		
null mutant	SOM	1.9169	1.0206	0.2287	2.3281	0.7364	0.2184	2.4263	0.6486	0.2410
		t = 1000, α_0 = 0.095						t = 225, α_0 = 0.75		
Cigarette	GSOM	2.0582	0.9305	0.2020	2.5789	0.5971	0.2111	2.6715	0.5383	0.2183
		t = 15, α_0 = 0.16			t = 12, α_0 = 0.65			t = 34, α_0 = 0.8		
smokers male	SOM	2.0994	0.8761	0.2065	2.5834	0.5945	0.2179	2.7950	0.5164	0.2360
		t = 400, α_0 = 0.32			t = 300, α_0 = 0.52			t = 200, α_0 = 0.82		
Cigarette	GSOM	1.9952	0.9754	0.2221	2.0851	0.6281	0.2318	2.1856	0.6091	0.2365
		t = 8, α_0 = 0.25			t = 30, α_0 = 0.56			t = 10, α_0 = 0.65		
smokers female	SOM	2.1093	0.9707	0.2266	2.3448	0.6099	0.2376	2.4226	0.5740	0.2428
		t = 400, α_0 = 0.82			t = 300, α_0 = 0.82			t = 200, α_0 = 92		
Resistance	GSOM	1.7375	0.8097	0.2226	2.5004	0.4412	0.2673	2.7663	0.1638	0.2871
		t = 100, α_0 = 0.96			t = 130, α_0 = 0.9			t = 100, α_0 = 0.95		
data	SOM	1.8318	0.7773	0.2470	2.5333	0.3373	0.2774	2.8453	0.1626	0.3103
		t = 1000, α_0 = 0.85			t = 1500, α_0 = 0.9			t = 1500, α_0 = 0.95		
Ethanol	GSOM	1.1232	1.7206	0.2386	1.4988	0.9397	0.3009	1.5334	0.5908	0.3177
		t = 25, α_0 = 0.255			t = 15, α_0 =0.95			t = 8, α_0 = 0.95		
data	SOM	1.7694	1.0198	0.4407	1.5022	0.7798	0.4415	1.6225	0.5686	0.4498
		t = 100, α_0 = 0.92			t = 100, α_0 =92			t = 100, α_0 =92		
		c = **3**			c = 6			c = 8		
Breast	GSOM	2.9806	0.5516	0.2145	3.6969	0.3869	0.2308	3.8946	0.3458	0.2445
		t = 10, α_0 = 0.15			t = 10, α_0 =0.45			t = 10, α_0 =0.55		
cancer	SOM	3.1952	0.4825	0.2231	3.9041	0.3463	0.2412	4.0754	0.2650	0.2480
		t = 200, α_0 = 0.009			t = 200, α_0 = 0.6			t = 300, α_0 = 0.8		
DLBCL-A	GSOM	2.4021	0.7945	0.1652	2.6794	0.6123	0.1657	2.7986	0.4885	0.1709
		t = 25, α_0 = 0.05			t = 10, α_0 = 0.05			t = 15, α_0 = 0.0645		
	SOM	3.6503	0.4452	0.2027	2.7767	0.5436	0.1763	3.0576	0.4787	0.2332
		t = 100, α_0 = 0.05			t = 100, α_0 = 0.04			t = 200, α_0 = 0.05		
		c = **4**			c = 6			c = 8		
Lung	GSOM	2.1395	0.6940	0.1441	3.5201	0.4160	0.1516	4.3477	0.3166	0.1522
		t = 8, α_0 = 0.05			t = 10, α_0 =0.2			t = 10, α_0 = 0.2		
cancer	SOM	2.2199	0.6829	0.1526	3.9848	0.2860	0.1773	4.4960	0.2726	0.2025
		t= 300, α_0 = 0.39			t= 300, α_0 = 0.9			t= 200, α_0 = 0.9		
	GSOM	2.1136	0.7531	0.3348	3.3671	0.4224	0.3468	3.5336	0.3524	0.3477
		t = 10, α_0 = 0.0686			t = 10, α_0 = 0.52			t = 10, α_0 = 0.52		
Multi-A	SOM	2.1823	0.7071	0.3351	3.1192	0.3951	0.3504	3.9269	0.3520	0.3566
		t = 200, α_0 = 0.15			t = 300, α_0 = 0.9			t = 200, α_0 =0.9		

is found to be obtained better results at $t = 29$ for this data. The figure also shows the selected numbers of iterations within parentheses, whereas the best results for SOM are attained at $\alpha_0 = 0.3$ for $t = 30$. The variation of Dunn-index with α_0 for various values of t is provided in the supplementary material online.

For choosing the number of clusters (c), we varied c from 2 to $\sqrt{s}$ in steps of 1, where s is the number of samples in the data. For each value of c, we repeated the aforementioned procedure for different values of α_0 and t, and the top three results (for three different c values), including the values of α_0 and t for all the data sets, in terms of DB-index, Dunn-index, and FRE, are shown in Table II. The best results of GSOM and SOM for ALL and AML data are achieved at $c = 2$. Moreover, the GSOM performs better than SOM in terms of DB-index, Dunn-index, and FRE. For the remaining data sets, the best c values are marked in bold. It can be concluded that the DB-index, Dunn-index, and FRE for all data sets conform c value equal to the number of clusters truly present in the data. Any value other than the true c value makes clustering indices change sharply.

2) Visualization of Output Clusters of GSOM and SOM: Fig. 3 shows the 2-D plot of all samples, normalized within 0 to 1, demonstrating the actual categories/groups of ALL and AML data. Figs. 4 and 5 show the corresponding output clusters ($c = 2$) of GSOM and SOM.

It is clear from Fig. 3 that the boundary region of cluster 1 overlaps with that of cluster 2. This information is reflected and preserved in Fig. 4 for GSOM, where the overlapping patterns are exactly clustered as in Fig. 3 and the winner neurons are represented as $\bullet$ and $*$. In Fig. 5, using SOM, four overlapping patterns are not exactly the same as in Fig. 3.

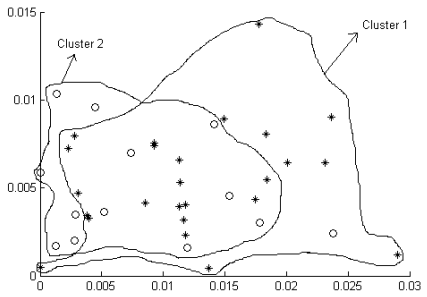

Fig. 3. 2-D plot of overlapping patterns of actual clusters for ALL and AML data.

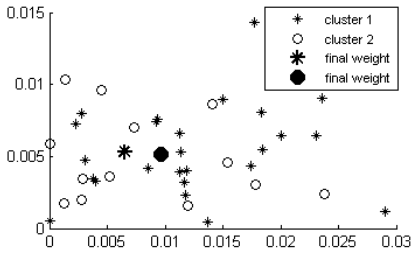

Fig. 4. 2-D plot of the output clusters of GSOM for ALL and AML data with no misclassified patterns.

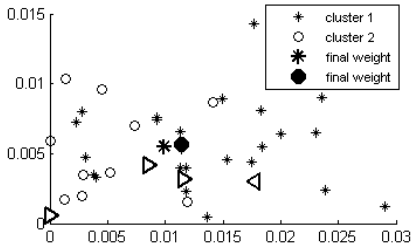

Fig. 5. 2-D plot of the output clusters of SOM for ALL and AML data where ▷ represents the patterns wrongly assigned to cluster 2 and ◁ represents the patterns wrongly assigned to cluster 1.

Here, three patterns actually belonging to cluster 1, indicated with ▷, are assigned to cluster 2 and 1 pattern actually belonging to cluster 2, represented with ◁, is assigned to cluster 1. Moreover, the distance between the winner neurons of GSOM is greater than that of SOM. As mentioned earlier, incorporation of fuzziness in the neighborhood function of GSOM handles the uncertainty arising from overlapping regions and incorporation of lower and upper approximation of rough set theory further helps in determining the exactness in class shapes; thereby making the GSOM superior to SOM.

3) Comparison of GSOM With Other Clustering Algorithms: The GSOM is compared with algorithms like RRFCM, RFPCM, RPCM, FCM, *c*-mediods, and AP method. The results are shown in Table III. The related parameters of these algorithms are shown in the last column of the table. Here for GSOM and SOM, the values of α_0 and t are provided, whereas the values of thresholds for RRFCM (δ and Tr), probabilistic constant and threshold for RFPCM (a and Tr), and threshold value for RPCM (Tr) are shown. The weighting exponents w and $\tilde{w}$, fuzzifiers $\acute{m}_1$ & $\acute{m}_2$, and possiblistic constant b for RRFCM, RFPCM, and RPCM are chosen to be 0.5 and 0.5, 2 and 2, 0.7, respectively, for all data sets as mentioned in the related articles [8], [9]. For RRFCM, RFPCM and RPCM, the possiblistic constant

TABLE III

COMPARISON OF CLUSTERING ALGORITHMS IN TERMS OF β-INDEX, DB-INDEX, DUNN-INDEX, AND FRE FOR MICROARRAY DATA FOR $c = 2$. PARAMETERS USED ARE MENTIONED IN THE SEVENTH COLUMN

Data	Method	β	DB	Dunn	FRE	Parameters
ALL & AML	GSOM	**1.175**	**2.215**	**0.855**	**0.368**	29(t), 0.4(α_0)
	SOM	1.173	2.307	0.845	0.3751	30(t), 0.3(α_0)
	RRFCM	1.174	2.216	0.850	0.373	0.04(δ), 0.5(Tr)
	RFPCM	1.173	2.307	0.844	0.3751	0.6(a), 0.5(Tr)
	RPCM	1.134	2.725	0.715	0.3801	0.25(Tr)
	FCM	1.134	2.731	0.661	0.3886	
	c-medo.	1.087	3.314	0.551	0.3933	
	AP met.	1.089	3.009	0.632	0.3888	
Ethanol data	GSOM	**1.574**	**1.123**	**1.721**	**0.2386**	11(t), 0.755(α_0)
	SOM	1.313	1.769	1.020	0.4407	100(t), 0.92(α_0)
	RRFCM	1.417	1.473	1.264	0.4314	0.01(δ), 0.05(Tr)
	RFPCM	1.224	1.473	0.854	0.4307	0.265(a), 0.9(Tr)
	RPCM	1.221	2.091	0.807	0.4311	0.455(Tr)
	FCM	1.134	2.527	0.789	0.4342	
	c-medo.	1.120	2.561	0.721	0.4498	
	AP met.	1.164	2.148	0.921	0.4377	
Upf1 null mutant	GSOM	**1.289**	**1.859**	**1.064**	**0.1796**	98(t), 0.055(α_0)
	SOM	1.271	1.917	1.021	0.2287	1000(t), 0.095(α_0)
	RRFCM	1.288	1.861	1.065	0.1801	0.51(δ), 0.589(Tr)
	RFPCM	1.275	1.882	1.045	0.2228	0.015(a), 0.65(Tr)
	RPCM	1.252	1.966	0.978	0.2302	0.75(Tr)
	FCM	1.204	2.117	0.882	0.2414	
	c-medo.	1.178	2.367	0.834	0.2466	
	AP met.	1.212	2.128	0.910	0.2333	
Diabetic	GSOM	**1.131**	**2.273**	**0.607**	**0.2105**	31(t), 0.16(α_0)
	SOM	1.123	2.871	0.570	0.2540	1000(t), 0.85(α_0)
	RRFCM	1.127	2.721	0.602	0.2180	0.051(δ), 0.04(Tr)
	RFPCM	1.079	3.535	0.449	0.2577	0.085(a), 0.9(Tr)
	RPCM	1.124	2.809	0.592	0.2446	0.75(Tr)
	FCM	1.127	2.721	0.602	0.2180	
	c-medo.	1.054	3.616	0.446	0.2645	
Prost -ate cancer	GSOM	**1.403**	**1.513**	**0.953**	**0.2939**	30(t), 0.6(α_0)
	SOM	1.355	1.677	0.919	0.3147	100(t), 0.95(α_0)
	RRFCM	1.377	1.584	0.948	0.2992	0.1(δ), 0.05(Tr)
	RFPCM	1.374	3.024	0.930	0.3014	0.09(a), 0.54(Tr)
	RPCM	1.162	3.679	0.737	0.3356	0.5(Tr)
	FCM	1.112	5.973	0.566	0.3725	
	c-medo.	1.115	5.612	0.589	0.3384	
Primary bovine mammary gland epith -elial cells	GSOM	**1.305**	**1.761**	**1.015**	**0.2197**	49(t), 0.3(α_0)
	SOM	1.234	2.061	0.864	0.239	100(t), 0.003(α_0)
	RRFCM	1.284	1.809	0.988	0.2236	0.05(δ), 0.5(Tr)
	RFPCM	1.292	1.801	0.988	0.2282	0.08(a), 0.5(Tr)
	RPCM	1.236	1.888	0.920	0.2387	0.05(Tr)
	FCM	1.217	2.099	0.836	0.2407	
	c-med.	1.229	2.089	0.853	0.2404	
	Ap met.	1.126	2.325	0.801	0.2954	
Carci -noma data	GSOM	**1.228**	**1.123**	**0.899**	**0.3755**	11(t), 0.5 (α_0)
	SOM	1.228	1.123	0.899	0.3755	100(t), 0.3(α_0)
	RRFCM	1.211	2.133	0.872	0.3857	0.01(δ), 0.002(Tr)
	RFPCM	1.162	2.471	0.715	0.3860	0.25(a), 0.25(Tr)
	RPCM	1.101	2.878	0.607	0.3996	0.01(Tr)
	c-medo.	1.163	2.267	0.769	0.4048	
	AP met.	1.128	2.787	0.698	0.4123	
Phago- cytosis	GSOM	**1.643**	**1.243**	**1.563**	**0.2986**	29(t), 0.3(α_0)
	SOM	1.627	1.254	1.559	0.3342	300(t), 0.5(α_0)
	RRFCM	1.512	1.254	1.411	0.3458	0.25(δ),0.5(Tr)
	RFPCM	1.495	1.358	1.366	0.3822	0.08(a), 0.5(Tr)
	RPCM	1.388	1.409	1.202	0.3939	0.5(Tr)
	FCM	1.263	1.844	1.076	0.4119	
	c-medo.	1.164	1.891	0.905	0.4135	
	AP met.	1.330	1.643	1.215	0.4012	
Pedia -tric cere bral palsy	GSOM	**1.413**	**1.556**	**1.169**	**0.2057**	29(t), 0.4(α_0)
	SOM	1.409	1.557	1.167	0.2267	500(t), 0.2(α_0)
	RRFCM	1.259	1.808	0.999	0.2446	0.05(δ), 0.05(Tr)
	RFPCM	1.142	1.921	0.897	0.2654	0.08(a), 0.01(Tr)
	RPCM	1.131	2.594	0.732	0.268	0.05(Tr)
	c-med.	1.077	3.243	0.534	0.2793	
	Ap met.	1.103	3.058	0.595	0.2787	
Pedia -tric devel -opment control	GSOM	**1.363**	**1.656**	**1.167**	**0.2025**	10(t), 0.2(α_0)
	SOM	1.362	1.662	1.128	0.2086	400(t), 0.5(α_0)
	RRFCM	1.302	1.782	1.016	0.2176	0.05(δ), 0.05(Tr)
	RFPCM	1.201	2.024	0.864	0.2311	0.08(a), 0.02(Tr)
	RPCM	1.088	3.296	0.606	0.2521	0.5(Tr)
	FCM	1.124	2.068	0.839	0.2333	
	c-medo.	1.182	1.968	0.958	0.2300	
	Ap met.	1.157	2.480	0.792	0.2457	

TABLE III

(*Continued*.) COMPARISON OF CLUSTERING ALGORITHMS IN TERMS OF β-INDEX, DB-INDEX, DUNN-INDEX, AND FRE FOR MICROARRAY DATA FOR $c = 2$. PARAMETERS USED ARE MENTIONED IN THE SEVENTH COLUMN

Continuation of TABLE III						
Data	Method	β index	DB index	Dunn index	FRE	Param -ters
Resist -ance data	GSOM	**1.300**	**1.738**	**0.810**	**0.2226**	30(t), 0.6(α_0)
	SOM	1.289	1.832	0.777	0.247	1000(t), 0.85(α_0)
	RRFCM	1.251	2.031	0.696	0.2532	0.05(δ), 0.89(Tr)
	RFPCM	1.123	2.128	0.663	0.254	0.01(a), 0.9(Tr)
	RPCM	1.185	2.141	0.634	0.2655	0.65(Tr)
	c-medo.	1.015	4.738	0.363	0.2800	
Pancr -eatic ductal adeno- carci- noma	GSOM	**1.266**	**1.910**	**1.046**	**0.1874**	10(t), 0.2(α_0)
	SOM	1.245	2.008	0.954	0.210	100(t), 0.5(α_0)
	RRFCM	1.252	1.988	0.993	0.2024	0.05(δ), 0.02(Tr)
	RFPCM	1.144	2.350	0.745	0.2182	0.08(a), 0.01(Tr)
	RPCM	1.183	2.255	0.798	0.215	0.01(Tr)
	FCM	1.108	2.497	0.758	0.2235	
	c-medo.	1.070	2.553	0.744	0.2268	
	Ap met.	1.192	2.759	0.219	0.2371	
Nasal lavage cells	GSOM	**1.302**	**1.784**	**0.932**	**0.0573**	10(t), 0.3(α_0)
	SOM	1.242	2.014	0.826	0.1372	200(t), 0.9(α_0)
	RRFCM	1.272	1.835	0.924	0.1298	0.49(δ), 0.05(Tr)
	RFPCM	1.151	2.343	0.676	0.1464	0.05(a), 0.01(Tr)
	RPCM	1.272	1.835	0.924	0.1298	0.5(Tr)
	c-medo.	1.100	2.463	0.633	0.1506	
J774.A1 macro -phage cells	GSOM	**1.535**	**1.273**	**1.128**	**0.2351**	10(t), 0.3(α_0)
	SOM	1.450	1.490	1.080	0.2592	400(t), 0.95(α_0)
	RRFCM	1.362	1.660	1.030	0.2672	0.5(δ), 0.5(Tr)
	RFPCM	1.328	1.393	0.992	0.2706	0.08(a), 0.01(Tr)
	RPCM	1.163	1.402	0.895	0.2762	0.05(Tr)
	FCM	1.161	2.441	0.747	0.2768	
	c-medo.	1.087	3.376	0.591	0.2857	
Cig. smokers female	GSOM	**1.245**	**1.995**	**0.975**	**0.2221**	8(t), 0.25(α_0)
	SOM	1.237	2.109	0.971	0.226	400(t), 0.82(α_0)
	RRFCM	1.240	2.040	0.952	0.223	0.01(δ), 0.5(Tr)
	RFPCM	1.226	2.095	0.931	0.2242	0.25(a), 0.65(Tr)
	RPCM	1.215	2.145	0.863	0.2255	0.054(Tr)
	FCM	1.197	2.162	0.798	0.2298	
	c-medo.	1.192	2.269	0.766	0.2259	
	AP met.	1.116	2.686	0.734	0.2308	
Cig. smokers male	GSOM	**1.235**	**2.058**	**0.931**	**0.2020**	15(t), 0.16(α_0)
	SOM	1.229	2.099	0.876	0.2065	400(t), 0.32(α_0)
	RRFCM	1.233	2.072	0.929	0.2045	0.03(δ), 0.05(Tr)
	RFPCM	1.229	2.089	0.877	0.2048	0.015(a), 0.65(Tr)
	RPCM	1.225	2.092	0.876	0.213	0.0545(Tr)
	FCM	1.230	2.084	0.881	0.2040	
	c-medo.	1.209	2.163	0.791	0.2193	
	AP met.	1.109	2.917	0.662	0.2283	
Unresec -table colorectal cancer	GSOM	**1.074**	**3.646**	**0.382**	**0.3598**	29(t), 0.4(α_0)
	SOM	1.062	3.953	0.363	0.385	500(t), 0.6(α_0)
	RRFCM	1.062	3.886	0.364	0.385	0.05(δ), 0.05(Tr)
	RFPCM	1.046	4.131	0.352	0.388	0.05(a), 0.01(Tr)
	RPCM	1.015	5.222	0.270	0.402	0.5(Tr)
Quadri -ceps muscles	GSOM	**1.245**	**2.005**	**0.889**	**0.1075**	30(t), 0.3(α_0)
	SOM	1.229	2.073	0.843	0.1237	600(t), 0.9(α_0)
	RRFCM	1.242	2.015	0.875	0.1133	0.89(δ), 0.5(Tr)
	RFPCM	1.165	2.165	0.784	0.1474	0.08(a), 0.01(Tr)
	RPCM	1.092	2.202	0.778	0.1434	0.05(Tr)
	FCM	1.003	3.063	0.648	0.1596	
	c-medo.	1.074	3.650	0.531	0.1719	
	AP met.	1.170	2.174	0.858	0.1172	
Patient derived color -ectal cancer explants	GSOM	**1.301**	**1.817**	**1.050**	**0.2193**	10(t), 0.1(α_0)
	SOM	1.298	1.823	1.042	0.2201	200(t), 0.3(α_0
	RRFCM	1.284	1.875	1.011	0.2267	0.9(δ), 0.9(Tr)
	RFPCM	1.280	1.890	0.957	0.238	0.05(a), 0.01(Tr)
	RPCM	1.280	1.890	0.957	0.2380	0.05(Tr)
	FCM	1.037	4.64	0.417	0.2452	
	c-medo.	1.056	3.556	0.553	0.2482	
	AP met.	1.146	2.593	0.719	0.2474	
Color -ectal cancer tumors	GSOM	**1.355**	**1.641**	**1.055**	**0.1653**	10(t), 0.2(α_0)
	SOM	1.338	1.715	1.044	0.166	400(t), 0.5(α_0)
	RRFCM	1.349	1.693	1.179	0.1713	0.05(δ), 0.05(Tr)
	RFPCM	1.335	1.663	1.034	0.1789	0.06(a), 0.01(Tr)
	RPCM	1.335	1.663	1.034	0.1789	0.5(Tr)
	FCM	1.137	1.877	0.971	0.2296	
	c-medo.	1.298	1.832	0.979	0.2054	
Atopic dermat -itis patie -nts	GSOM	**1.1604**	**2.484**	**0.750**	**0.1789**	30(t), 0.3(α_0)
	SOM	1.143	2.635	0.725	0.2086	400(t), 0.9(α_0)
	RRFCM	1.136	2.662	0.699	0.2382	0.5(δ), 0.5(Tr)
	RFPCM	1.104	2.550	0.701	0.2378	0.05(a), 0.01(Tr)
	RPCM	1.104	3.080	0.626	0.2396	0.05(Tr)
	FCM	1.088	3.379	0.560	0.2414	
	c-medo.	1.098	2.714	0.606	0.2542	
	AP met.	1.088	2.808	0.700	0.2550	

b is chosen greater than probabilistic constant a. The value of c for all the methods is shown within parenthesis in the first column of Table III. The maximum value of t for all algorithms, except GSOM, SOM, and AP method, is set equal to 200. For every data set, the best results are shown in bold font in the table.

TABLE IV

COMPARISON OF CLUSTERING ALGORITHMS IN TERMS OF β-INDEX, DB-INDEX, DUNN-INDEX, AND FRE FOR MICROARRAY DATA FOR $c = 3$. PARAMETERS USED ARE MENTIONED IN THE SEVENTH COLUMN

Data	Method	β	DB	Dunn	FRE	Parameters
DLBCLA	GSOM	**1.251**	**2.402**	**0.795**	**0.1652**	25(t), 0.05(α_0)
	SOM	1.184	3.650	0.445	0.2027	100(t), 0.05(α_0)
	RRFCM	1.232	2.454	0.723	0.1947	0.01(δ), 0.05(Tr)
	RFPCM	1.227	2.614	0.726	1.1990	0.09(a), 0.8(Tr)
	RPCM	1.215	2.672	0.655	0.2008	0.5(Tr)
	c-medo.	1.112	3.828	0.425	0.2055	
	AP met.	1.137	3.031	0.574	0.2046	
Breast cancer	GSOM	1.245	**2.981**	**0.552**	**0.2145**	10(t), 0.15(α_0)
	SOM	**1.266**	3.195	0.483	0.2231	200(t), 0.009(α_0)
	RRFCM	1.258	3.122	0.537	0.2152	0.01(δ), 0.05(Tr)
	RFPCM	1.245	3.130	0.529	0.2211	0.1(a), 0.89(Tr)
	RPCM	1.245	3.404	0.477	0.2247	0.8(Tr)
	FCM	1.245	3.406	0.453	0.2285	
	c-medo.	1.244	3.555	0.414	0.2307	
Perip -heral blood mononu -clear cells	GSOM	**1.368**	**2.085**	**0.902**	**0.1134**	12(t), 0.2(α_0)
	SOM	1.344	2.249	0.837	0.1227	100(t), 0.8(α_0)
	RRFCM	1.305	2.301	0.762	0.1324	0.49(δ), 0.05(Tr)
	RFPCM	1.333	2.120	0.883	0.117	(0.08a), 0.01(Tr)
	RPCM	1.226	2.549	0.744	0.1395	0.5(Tr)
	FCM	1.328	2.202	0.809	0.1196	
	c-medo.	1.262	2.504	0.752	0.1344	
	AP met.	1.144	3.221	0.572	0.1528	
RRPEB	GSOM	**1.467**	**1.863**	**0.222**	**0.097**	10(t), 0.2(α_0)
	SOM	1.320	2.288	0.259	0.1442	40(t), 0.955(α_0)
	RRFCM	1.325	2.081	0.300	0.1415	0.019(δ), 0.5(Tr)
	RFPCM	1.169	5.522	0.192	0.1676	0.1(a), 0.5(Tr)
	RPCM	1.173	2.306	0.196	0.1549	0.5(Tr)
Cigarette smoke of light flavor effect	GSOM	1.485	**2.067**	**0.858**	**0.0302**	18(t), 0.4(α_0)
	SOM	**1.579**	2.1969	0.796	0.0937	40, 0.955(α_0)
	RRFCM	1.436	2.079	0.851	0.0353	0.05(δ), 0.5(Tr)
	RFPCM	1.450	3.229	0.418	0.1037	0.08(a), 0.001(Tr)
	RPCM	1.426	2.088	0.783	0.0772	0.05(Tr)
	c-medo.	1.464	2.836	0.4822	0.1025	

Table III shows the results of twenty one microarray data for two output clusters. The results indicate that GSOM performs better in terms of β-index, DB-index, Dunn-index, and FRE for all the cases except for carcinoma where the results are same with that of SOM since for carcinoma data cluster boundaries are not much overlapped.

Table IV provides the results of five microarray data for three output clusters. The results reveal that the performance of GSOM is superior to all the methods for all the data sets in terms of DB-index, Dunn-index, and FRE, except for breast cancer using SOM and RRFCM and cigarette smoke of light flavor effect using SOM where β-index is lower.

The results in Table V for five data sets with four output clusters show that GSOM outperforms the remaining methods in terms of cluster evaluation measures for all the data except for Multi-A and peripheral white blood cells-A and B using SOM where β-index is lower. A possible explanation of this may be that β-index does not consider the difference between the means of different clusters, unlike other indices. In summary, out of 868 comparisons from Tables III–V, only in six cases the GSOM is inferior.

4) Comparison in Terms of Confusion Matrices: The output clusters of GSOM, SOM, RRFCM, RFPCM, RPCM, FCM, c-mediods, and AP method, arranged in the form of confusion matrix, are shown in Table VI for ALL and AML data, as examples. For GSOM, the samples belonging to output

TABLE V
COMPARISON OF CLUSTERING ALGORITHMS IN TERMS OF β-INDEX, DB-INDEX, DUNN-INDEX, AND FRE FOR MICROARRAY DATA FOR $c = 4$. PARAMETERS USED ARE MENTIONED IN THE SEVENTH COLUMN

Data	Method	β	DB	Dunn	FRE	Parameters
Lung cancer	GSOM	**1.627**	**2.139**	**0.694**	**0.1441**	8(t), 0.05(α_0)
	SOM	1.593	2.219	0.683	0.153	300(t), 0.39(α_0)
	RRFCM	1.571	2.321	0.540	0.1820	0.21(δ), 0.005(Tr)
	RFPCM	1.569	2.456	0.492	0.1881	0.3(a),0.8(Tr)
	RPCM	1.566	2.5267	0.438	0.2299	0.5(Tr)
	c-medo.	1.534	2.701	0.465	0.2634	
	AP met.	1.358	2.895	0.478	0.2639	
Multi-A	GSOM	1.499	**2.114**	**0.753**	**0.3348**	10(t), 0.0686(α_0)
	SOM	**1.500**	2.182	0.707	0.3351	200(t), 0.15(α_0)
	RRFCM	1.498	2.328	0.641	0.3372	0.01(δ), 0.005(Tr)
	RFPCM	1.367	2.764	0.645	0.3453	0.15(a),0.89(Tr)
	RPCM	1.266	2.992	0.576	0.3499	0.01(Tr)
	c-medo.	1.256	2.816	0.560	0.3680	
	AP met.	1.178	2.791	0.499	0.3809	
HepG2 liver cells	GSOM	**1.391**	**2.248**	**0.696**	**0.2288**	10(t), 0.051(α_0)
	SOM	1.364	2.410	0.651	0.2293	10(t), 0.051(α_0)
	RRFCM	1.372	2.323	0.631	0.2528	0.019(δ), 0.5(Tr)
	FRFCM	1.336	2.622	0.554	0.2575	0.06(a), 0.01(Tr)
	RPCM	1.347	2.575	0.563	0.2545	0.05(Tr)
	c-medo.	1.343	3.026	0.447	0.259	
Perip -heral white blood cells-A	GSOM	1.855	**1.794**	**0.191**	**0.3095**	10(t), 0.2(α_0)
	SOM	**1.858**	3.559	0.153	0.3137	50(t), 0.8(α_0)
	RRFCM	1.512	2.311	0.318	0.3181	0.5(δ), 0.05(Tr)
	RFPCM	1.225	1.874	0.173	0.3316	0.08(a), 0.01(Tr)
	RPCM	1.209	1.954	0.173	0.3358	0.5(Tr)
Perip -heral white blood cells-B	GSOM	1.972	**1.783**	**0.235**	**0.2641**	29(t), 0.4(α_0)
	SOM	**2.091**	2.224	0.184	0.2788	80(t), 0.9(α_0)
	RRFCM	1.361	1.801	0.222	0.2686	0.5(δ), 0.05(Tr)
	RFPCM	1.609	1.793	0.219	0.2708	0.08(a), 0.01(Tr)
	RPCM	1.286	2.926	0.141	0.2910	0.5(Tr)

TABLE VI
CONFUSION MATRICES FOR ALL AND AML DATA USING DIFFERENT CLUSTERING METHODS

a) GSOM

27	0
0	11

b) SOM

24	3
1	10

c) RRFCM

25	2
1	10

d) RFPCM

24	3
1	10

e) RPCM

22	5
1	10

f) FCM

19	8
2	9

g) c-mediods

16	11
2	9

h) AP method

25	2
1	10

clusters 1 and 2 are seen to be exactly equal to the actual samples, 27 and 11, respectively, in this data; thereby resulting in zero off diagonal elements. This is not true for other methods.

C. Computation Complexity

The computation complexity of GSOM involving normalization of data, initial connection weights, SOM, and fuzzy rough neighborhood function is discussed as follows.

1) Complexity for Normalizing Data: The cost of computation in finding global minimum and global maximum for data is O(sn), where s is the number of patterns and n is the number of features. Therefore, the asymptotic complexity for normalization of data (see Section III) is O(sn).

2) Complexity for Presenting a Pattern as an Input Vector to the SOM: It is O(n) as a pattern contains n number of features.

3) Complexity for Initializing Connection Weights of SOM: The cost in initializing $c \times n$ random numbers within 0–1 as initial connection weights of SOM is O(cn), where c is the number of output nodes.

4) Complexity of SOM: During training of SOM, for every pattern, the complexity in computing the distance from the input vector to weight vector for one output node [(14), Section III-A] is O(n). This is similar to that for computing the Euclidean distance. For c output nodes, this complexity is O(cn). The computational cost for finding the winning node (out of c nodes) in the output layer [(15), Section III-A] is O(c). The complexity for updating the $c \times n$ connection weights [see (18)] is O(cn). Thus, the complexity SOM (including input vector presentation and initialization) is O($ts(n+cn+cn+c+cn)$) where t is the number of iterations and s is the number of patterns. Therefore, the asymptotic complexity is O($tscn$). In our case, the asymptotic complexity is O(scn) as GSOM uses the conventional SOM for $t = 1$.

5) Complexity of Fuzzy Rough Neighborhood Function: The neighborhood function involves fuzzy decision classes, fuzzy reflexive relational matrix, lower and upper approximations, and boundary region. The complexities of these components are as follows.

1) The complexity of fuzzy decision classes depends on membership values of patterns. The membership values (3) of all the patterns to its own class are a function of weighted distance of the patterns to the mean of its class (2) and fuzzyfiers f_d and f_e. Let us consider that s_c indicates the number patterns in the cth class. Hence, the complexity in computing the membership values of s_c number of patterns is O($s_cn+s_cn+s_cn+s_c$), where the first term denotes the complexity of the weighted distance of s_c number of patterns (with n features) from the mean of the class, the second term indicates the complexity of the mean of the class, the third term represents the standard deviation of the class by using the mean in previous step, and the last term is the complexity of computing membership values of the patterns using weighted distance.
   The complexity in computing the membership value of all the patterns (s_c) in a class to the other classes is O($s_c(c-1)$) as the number of remaining classes is $c-1$ and the memberships for the patterns are directly assigned as 1 without any computation.
   Considering all the classes, the total complexity in computing fuzzy decision classes is the sum of the aforementioned complexities, which is O($c((s_cn+s_cn+s_cn+s_c)+(s_c(c-1)))$). Therefore, the asymptotic complexity is O($cs_c(n+c)$).
2) The fuzzy reflexive relational matrix for each feature involves the standard deviation of a class and one minimum operation and one maximum operation between a pair of patterns as shown in (1). The complexity of standard deviation of a class with s_c patterns is O(s_c). For c classes, this value is O(cs_c). The complexity for one minimum operation and one maximum operation between a pair of patterns is of constant order O(2). For all possible pairs of patterns, this value is O($2(s \times s)$), where s is the total number of patterns in all the classes. Hence, for each feature, the complexity of fuzzy reflexive relational matrix of size $s \times s$ is O($cs_c + 2s^2$) and the asymptotic complexity is O($cs_c + s^2$).

3) For each feature, the membership value of a pattern in a class belonging to the lower approximation is computed using (7), which involve the aforementioned fuzzy decision classes and fuzzy reflexive relational matrix. Moreover, (7) includes one minimum operation between a pair involving complexity O(2), one minimum operation among s_c elements (8) involving complexity O(s_c), $s - s_c$ maximum operations (9), each operation performing between two elements, involving complexity O($(s - s_c)2$), and one minimum operation among $s - s_c$ elements involving complexity O($s - s_c$) (also in (9)). Here, the complexity of fuzzy decision classes (O($cs_c(n + c)$)) involving (8) and (9) is obtained from precomputed results. Considering all the above factors, for each feature, the complexity of membership values of all the s patterns belonging to the lower approximations of all the classes is O($cs_c(n + c)$) + O($(cs_c + s^2) + s2 + ss_c + s(s - s_c)2 + s(s - s_c)$). Therefore, for all features (n) the complexity is O($cs_c(n+c)$) + O($n(cs_c + s^2) + s2 + ss_c + s(s - s_c)2 + s(s - s_c)$). The asymptotic complexity is O($cs_c(n + c) + ns^2$). The complexity for upper approximation (10) is similar to that of the lower approximation as the number of operations is equal. Hence, the overall asymptotic complexity remains the same with that mentioned before.
4) For every feature, the complexity in computing the average of lower membership values and the average of upper membership values for all the patterns (s_c) in a class is O($2s_c$). Considering all n features and all the c classes, the complexity is O($2cs_cn$).
5) For every feature, the complexity in finding the boundary region of an output node is O(1) (see Step 5 in Section III-A1). Hence, for all the features (n) and c output nodes (classes) the complexity is O(cn). This region is employed in computing the neighborhood function. Thus, the complexity in defining the neighborhood function using (20) is O(cn).

6) Complexity in Finding the Winner and Its Neighborhood Nodes for GSOM: The computational cost of finding winning neuron, for a pattern with n features, in the output layer of GSOM having c nodes is O($cn+c$). The complexity of finding neighborhood neurons of the winner is O($(c - 1)n + n + K$)), where $(c - 1)n$ and n denote the complexity for sum of the average memberships in the lower and boundary region of all n features for $(c - 1)$ output neurons and the winner, respectively, and K is a constant denoting one comparison. The asymptotic complexity is O(cn). For all s patterns, the value is O(scn).

7) Complexity of GSOM: This complexity, involving normalization of data, SOM for one iteration, all the components of fuzzy neighborhood function and finding the winner neuron, is O($sn + scn + t(((cs_c(n + c) + ns^2) + 2cs_cn + cn + cn) + scn$)), where t is the number of iterations. Therefore, the asymptotic complexity of GSOM is O($t((cs_c(n + c)) + ns^2 + scn)$).

8) Complexity of UFRFS: The complexity of UFRFS is based on the aforementioned complexities of normalization of data (O(sn)), membership values of s patterns belonging to the lower approximations of all the classes (O($cs_c(n + c) + ns^2$) which includes the complexities of fuzzy reflexive relations and fuzzy decision classes), and the complexity of the dependence values of n features using (13) (O(sn)). Therefore, the complexity of UFRFS is O($sn + (cs_c(n + c) + ns^2) + sn$). The asymptotic complexity of UFRFS is O($cs_c(n + c) + ns^2$).

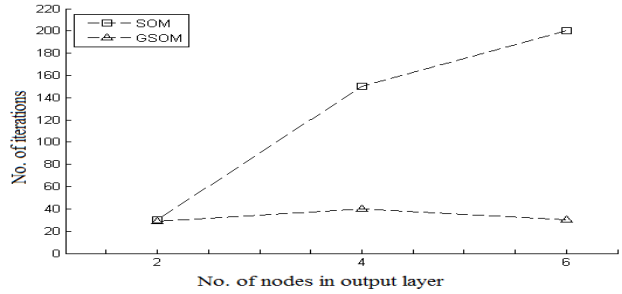

Fig. 6. Comparison of convergence speed between GSOM and SOM in terms of the number of iterations for different numbers of output nodes for ALL and AML data.

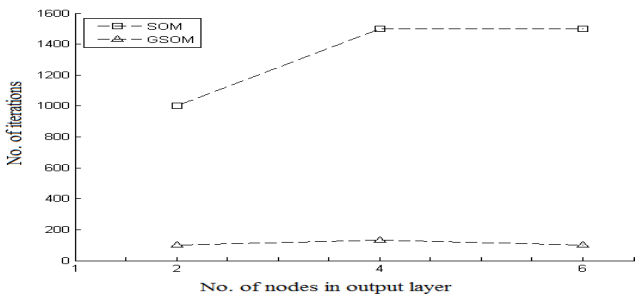

Fig. 7. Comparison of convergence speed between GSOM and SOM in terms of the number of iterations for different numbers of output nodes for resistance data.

D. Convergence Speed

As explained in Section V-B1, the GSOM is trained using different numbers of output nodes (c) for different numbers of iterations (t). For different values of c, the convergence speed of GSOM in terms of the number of iterations t is shown in Figs. 6 and 7 for ALL and AML data and resistance data, respectively. The value of t for which the GSOM attains the minimum quantization error and provides better results is chosen. It is evident from Fig. 6 for ALL and AML data that the plot of variation of the numbers of iterations for GSOM is seen much lower than that of SOM for $c = 4$ and 6, except $c = 2$ where it is equal to SOM. From Fig. 7, the convergence speed of GSOM (100, 130, and 100) for resistance data is found to be faster than SOM (1000, 1500, and 1500) for all values c (2, 4, and 6). Note that, for the same number of iterations, the SOM provides lower CPU times than GSOM as the GSOM has the components like SOM, fuzzy decision classes, fuzzy reflexive relation matrix, and fuzzy rough neighborhood function. The speed of GSOM and its quantization error for lung cancer data, as an example, is shown in Figs. 1 and 2, respectively, at http://avatharamg.webs.com/GSOM-UFRFS.pdf.

E. Results of Gene Selection

As mentioned in Section IV, the UFRFS assigns a dependence value to every gene in a data set. Since the gene with

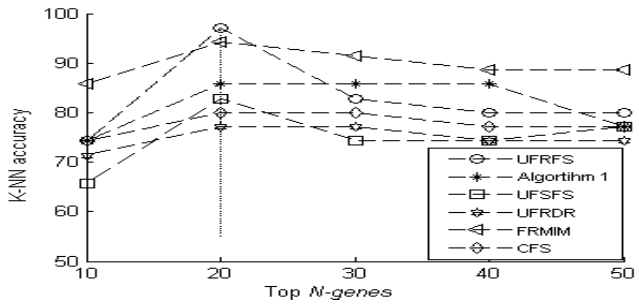

Fig. 8. Variation in classification accuracies obtained using K-NN for top-N numbers of genes selected by different algorithms for ALL and AML data.

a small dependence value is important for selection, they are arranged in increasing order, according to their dependence values. The top-N number of genes is then selected based on their classification capability of samples (patients), as measured by classifiers—FRGNN, FMLP, and K-NN.

1) Selection of Top Genes: For training the classifiers, we used fivefold cross validation, designed with stratified sampling. Training is performed on fourfolds of samples selected randomly. Testing is performed using one fold of data. This process is repeated five times and the average of the classification accuracies of five folds of test data is computed. The average of the classification accuracies, using FRGNN, FMLP, and K-NN, are calculated for top $N = 10$, 20, 30, 40, and 50 selected genes. The one with the highest accuracy value constitutes the resulting selected genes. As an example, using K-NN, for ALL and AML data, the variation in the average classification accuracies with the numbers of top genes is shown in Fig. 8 for all gene selection algorithms. It can be observed from the figure that the highest accuracy for UFRFS is observed at $N = 20$ (97.14%). The UFRFS has also achieved the best accuracies for top 20 genes using FMLP (100%) and FRGNN (100%). For the remaining data sets, the values of N thus obtained are shown in third column of Tables VII and VIII.

2) Comparison of UFRFS With Unsupervised Gene Selection Methods: Table VII also illustrates the results of UFRFS and the other unsupervised methods, Algorithm 1, UFSFS and UFRDR. The results in bold font in the table indicate that the UFRFS performs better than all the unsupervised methods for most of the data sets. Here, the performance of UFRFS is equal to that of Algortihm 1 only for the lung cancer and breast cancer data using FMLP and K-NN, and for ethanol data when FRGNN and FMLP are used as classifiers. However, based on the average results of all data sets, the UFRFS is seen to be superior to all the remaining algorithms. In other words, the results of unsupervised methods (Table VII) indicate that out of 128 comparisons UFRFS shows superior performance in 122 cases and equal performance in six cases. Note that, UFSFS and UFRDR cannot complete execution for data sets having large dimension, 54 675, (e.g., cigarette smokers male and female).

In terms of CPU time, Algorithm 1 is better than UFRFS, as expected, as it uses SOM instead of GSOM. Again, the UFRFS is superior to UFRDR and UFSFS for all data sets except for lung cancer data, where only UFSFS takes lower CPU time.

TABLE VII
COMPARISON OF UFRFS WITH UNSUPERVISED GENE SELECTION METHODS IN TERMS OF CLASSIFICATION ACCURACY (USING FRGNN, FMLP, AND K-NN) AND ENTROPY OF SELECTED GENES

Data Set	Method	No. of selected genes	FRGNN	FMLP	K-NN	Entropy	CPU time in Sec.
AML & ALL	UFRFS	20	**100.00**	**100.00**	**97.14**	**0.2177**	40.86
	Algorithm 1	20	97.14	94.28	85.71	0.2401	10.11
	UFRDR	20	94.29	91.43	77.14	0.2421	1368.60
	UFSFS	20	94.29	91.43	82.85	0.2407	95.45
DLBCL-A	UFRFS	50	**88.89**	**88.15**	**82.22**	**0.2430**	63.98
	Algorithm 1	50	86.67	85.93	80.00	0.2432	4.51
	UFRDR	50	84.44	80.00	79.26	0.2449	524.40
	UFSFS	50	88.15	87.85	80.00	0.2449	79.76
Lung cancer	UFRFS	10	**90.53**	**85.26**	**88.42**	**0.2246**	58.78
	Algorithm 1	10	88.95	85.26	88.42	0.2301	15.08
	UFRDR	10	86.84	83.68	86.32	0.2407	2353.80
	UFSFS	10	85.78	82.63	85.78	0.2407	44.11
Prostate cancer	UFRFS	10	**78.00**	**72.00**	**69.00**	0.2280	441.71
	Algorithm 1	10	63.00	62.00	60.00	0.2232	25.72
	UFRDR	10	63.00	61.00	59.00	**0.2179**	195480.00
	UFSFS	10	62.00	61.00	58.00	0.2279	9582.93
Upf1 null mutant	UFRFS	20	**94.38**	**91.88**	**91.25**	**0.2126**	868.78
	Algorithm 1	20	93.12	91.88	85.00	0.2188	60.82
	UFRDR	20	88.13	88.13	83.75	0.2404	8100.00
	UFSFS	20	88.13	85.62	83.75	0.2400	4607.21
Resistance data	UFRFS	20	**92.50**	**90.00**	**87.50**	**0.2004**	252.31
	Algorithm 1	20	90.00	87.50	87.50	0.2229	11.68
	UFRDR	20	67.50	67.50	62.50	0.2307	15000.00
	UFSFS	20	67.50	65.00	60.00	0.2307	12764.46
Multi-A	UFRFS	50	**92.63**	**92.63**	**90.57**	**0.2426**	139.07
	Algorithm 1	50	91.58	89.47	88.42	0.2431	26.68
	UFRDR	50	90.52	89.47	90.52	0.2437	27469.8
	UFSFS	50	87.37	86.31	83.13	0.2437	1095.21
Diabetic	UFRFS	20	**83.33**	**80.00**	**76.67**	**0.1824**	53.76
	Algorithm 1	20	83.33	80.00	73.33	0.1826	28.77
	UFRDR	20	76.67	73.33	70.00	0.1913	30600.00
	UFSFS	20	76.67	73.33	70.00	0.1913	12764.46
Breast cancer	UFRFS	10	**85.26**	**84.21**	**82.11**	**0.0281**	20.81
	Algorithm 1	10	84.21	84.21	82.11	0.0283	4.86
	UFRDR	10	82.11	82.11	81.05	0.0283	944.4
	UFSFS	10	83.57	83.57	81.05	0.0282	54.33
Ethanol data	UFRFS	50	**97.50**	**97.50**	**95.00**	**0.2335**	11.84
	Algorithm 1	50	97.50	97.50	65.00	0.2388	11.61
	UFRDR	50	82.50	75.00	72.50	0.2357	8028.00
	UFSFS	50	65.00	65.00	60.00	0.2391	15582.93
Cigarette smokers male	UFRFS	20	**87.33**	**80.56**	**80.56**	**0.2281**	123.47
	Algorithm 1	20	83.33	80.56	75.00	0.2285	90.23
Cigarette smokers female	UFRFS	20	**91.67**	**88.83**	**80.00**	**0.2384**	200.91
	Algorithm 1	20	86.67	83.33	73.33	0.2384	89.92
Average results of all data sets	UFRFS	-	**90.17**	**87.59**	**85.04**	**0.2066**	189.69
	Algorithm 1	-	87.13	85.16	78.65	0.2115	31.66
	UFRDR	-	81.60	79.17	76.21	0.2116	28986.9
	UFSFS	-	79.85	78.17	74.46	0.2127	5667.09

Furthermore, the UFRDR takes 54.30 h (with Intel Core i7 CPU 880 at 3.07 GHZ processor and 16 GB RAM) to complete the execution as compared with 0.1226 h (441.709 s) by UFRFS for prostate cancer data. This limitation of the former for high dimensional data corroborates the earlier finding [15].

3) Comparison of UFRFS With Supervised Gene Selection methods: Although UFRFS is an unsupervised method, we compared its performance with two supervised methods, namely, FRMIM [12] and CFS algorithm [33]

TABLE VIII
COMPARISON OF UFRFS WITH SUPERVISED FRMIM AND CFS IN TERMS OF CLASSIFICATION ACCURACY (USING FRGNN, FMLP, AND K-NN) AND ENTROPY OF SELECTED GENES

Data Set	Method	No. of selected genes (N)	FRGNN	FMLP	K-NN	Entropy	CPU time in Sec.
AML & ALL	UFRFS	20	**100.00**	**100.00**	**97.14**	**0.2177**	40.86
	FRMIM	20	100.00	100.00	94.28	0.2177	4.59
	CFS	20	80.00	74.29	80.00	0.2391	0.10
DLBCL-A	UFRFS	50	88.89	88.15	82.22	0.2430	63.98
	FRMIM	50	88.89	88.15	82.22	0.2432	0.45
	CFS	50	**94.07**	**93.33**	**83.70**	**0.2429**	0.39
Lung cancer	UFRFS	10	90.53	85.26	88.42	0.2246	58.78
	FRMIM	10	**95.26**	**95.26**	**93.16**	**0.2245**	0.09
	CFS	10	88.94	85.26	88.42	0.2345	0.11
Prostate cancer	UFRFS	10	78.00	75.00	70.00	**0.2280**	441.71
	FRMIM	10	**88.00**	**86.00**	**85.00**	0.2340	4.28
	CFS	10	71.00	66.00	68.00	0.2306	2543.40
Upf1 null mutant	UFRFS	20	94.38	91.88	91.25	0.2126	868.78
	FRMIM	20	**97.50**	**97.50**	**91.25**	**0.2100**	5.36
	CFS	20	97.50	97.50	92.50	0.2126	1.02
Resistance data	UFRFS	20	92.50	90.00	87.50	0.2004	252.31
	FRMIM	20	92.50	90.00	87.50	0.2004	9.22
	CFC	20	**92.50**	**90.00**	**90.00**	**0.2000**	0.58
Multi-A	UFRFS	50	92.63	92.63	90.57	0.2426	139.07
	FRMIM	50	**98.95**	**97.90**	**93.68**	**0.2420**	5.23
	CFS	50	98.95	97.90	91.58	0.2423	0.23
Diabetic	UFRFS	20	83.33	80.00	76.67	0.1824	53.76
	FRMIM	20	96.67	96.67	86.67	**0.1477**	9.29
	CFS	20	**96.67**	**96.67**	**90.00**	0.1976	0.32
Breast cancer	UFRFS	10	85.26	84.21	82.11	0.0281	20.81
	FRMIM	10	**92.63**	**89.47**	**87.37**	0.0284	0.45
	CFS	50	90.52	88.42	86.32	0.0280	0.10
Ethanol data	UFRFS	50	**97.50**	**97.50**	**95.00**	**0.2335**	11.84
	FRMIM	50	95.00	85.00	82.50	0.2348	51.11
Cigarette smokers male	UFRFS	20	87.33	80.56	80.56	0.2281	123.47
	FRMIM	20	**88.89**	**88.89**	**83.33**	**0.2279**	150.01
Cigarette smokers female	UFRFS	20	91.67	88.83	80.00	0.2384	200.91
	FRMIM	20	**93.33**	**93.33**	**85.00**	**0.2306**	155.09
Average results of all data sets	UFRFS	-	90.17	87.59	85.04	0.2066	189.69
	FRMIM	-	**93.97**	**92.35**	**87.66**	0.2034	32.93
	CFS	-	90.02	87.71	85.61	**0.2031**	282.92

in Table VIII. The comparison with FRMIM indicates that out of 48 cases, the performance of UFRFS is better in eight cases and equal in nine cases. On the other hand, out of 36 cases the UFRFS performs better than CFS in nine cases and equal in four cases. For each of the data sets, the best results are shown in bold font in the table. For carcinoma data, the results of UFRFS for top 70 selected genes are equal to those of the remaining supervised and unsupervised gene selection algorithms as the uncertainty is not much involved in this data. Note that, CFS cannot complete execution for data sets having dimension greater than or equal to 18 952 (e.g., cigarette smokers male and ethanol).

F. Statistical Analysis of Gene Selection Algorithms

For performing statistical analysis, first one has to decide the suitability between the two test procedures, parametric and nonparametric. In general, nonparametric tests are performed when the data sets do not satisfy the normality condition. The data are also not normally distributed if the data are an ordinal variable or a rank, or there are definite outliers or some data points are not measured appropriately due to clear limits of detection of the measuring instrument. One can perform statistical tests like Kolmogorov–Smirnov test, D'Agostino–Pearson test, and Shapiro–Wilk test to check if the data are normally distributed or not. Using these tests with significance level 0.01, we checked the normality of the data

TABLE IX
RESULTS FOR NORMALITY TESTS USING CLASSIFICATION ACCURACY OF DIFFERENT GENE SELECTION METHODS

	UFRFS	Algorithm 1	UFRDR	UFSFS	FRMIM	CFS
K.-Smirnov	0.00	0.00	0.00	0.00	0.00	0.00
D.-Pearson	0.00	0.00	0.02	0.00	0.00	0.00
S.-Wilk	0.00	0.00	0.00	0.00	0.00	0.00

using all the classification accuracies for a particular gene selection method. The level implies that the null hypothesis "the accuracy values for any particular gene selection method follow a normal distribution" should be rejected if the p-value is less than the significance level. In other words, a test at the 0.01 level of significance is such that the null hypothesis is falsely rejected only in 1% of the cases. The p-values corresponding to the statistics of these tests are shown in Table IX, which indicate that distribution is not normal for the data sets using any of the methods, the null hypothesis should be rejected and nonparametric test should be conducted for statistical evaluation of our results. In addition, if the data sample is small, failure to reject the null hypothesis by any statistical test, including the test of normality, does not necessarily guarantee that the null is actually true. A more valid reason to conduct nonparametric test on our results is that we want to perform comparison of more than two methods over multiple data sets [34]. In this regard, a nonparametric test involving Friedman's test is appropriate. Note that, comparison of methods performance, averaged across multiple domains or data sets, is usually not recommended using other statistical tests, since the evaluation measures in different domains should not be treated as commensurable. Furthermore, the statistical tests may be subject to threats of internal, external, construct, and statistical conclusion validity.

Regarding threats to internal validity, we want to mention that as the gene selection procedures are performed on the same data sets using the same computer, chances for change in the measuring instrument or condition (or other issues like aging, environmental change, change of samples, and interaction between samples) are negligible. Further, any gene selection method does not influence the results of another gene selection method as they are independent of each other. Therefore, there are no threats to internal validity for the statistical tests performed. The threats related to the external validity are minimized by performing fivefold cross validation, involving stratified random selection process, in the training and testing phase of each classifier in evaluation of the selected genes by each feature selection method. The accuracy values of all the test folds using the classifiers FRGNN, FMLP, and KNN are employed to perform the statistical tests. The statistical tests can also be subject to threats related to statistical conclusion validity. One of them is the low statistical power (when the size of the sample is too small and the value of p is low) which increases the likelihood of making a Type II error (fail to reject the null hypothesis when it is false). Here, this threat remains for normality test, which we can also ignore, as there are aforementioned other factors to perform nonparametric test. For Friedman's test the threats to low statistical power is less as there are 135 accuracy values (9 data set $\times$ 5 fold cross validation $\times$ 3 classifiers) for each

TABLE X
RESULTS OF FRIEDMAN'S TEST USING THE CLASSIFICATION ACCURACY VALUES OF EACH GENE SELECTION METHOD

	UFRFS	Algorithm 1	UFRDR	UFSFS	FRMIM	CFS
Mean of ranks	17.17	14.01	10.50	10.47	21.87	18.98
χ^2 test statistic	207.29					
p-value	$7.83331e^{-43}$					

TABLE XI
RESULTS OF BONFERRONI–DUNN *Post Hoc* TEST

Differences between the mean ranks	3.16	6.67	6.7	-4.7	-1.81
Adjusted p-value	0.002				
Critical Difference CD	0.8061				

gene selection method while comparing them. The significance level is chosen as 0.01 which provides stronger evidence than the widely used standard norm of 0.05. Some other threats related to the statistical conclusion validity are mostly related to gene expression data preparation, which are beyond the scope of this investigation. In the gene selection procedure, there may lie threats related to construct validity as the results are not only dependent on the gene selection methods but also on the performance (in terms of accuracy) of the used classifiers. However, the threats are minimized as the same classifiers are used for all the gene selection methods.

Now, we discuss the Friedman's test, which uses accuracy values (obtained using various classifiers), of different feature selection methods. The test is aimed at checking whether there is overall statistically significant difference among the mean of ranks of the related gene selection methods using two tailed test. The null hypothesis that "the mean values of ranks of UFRFS, Algorithm 1, UFRDR, UFSFS, FRMIM, and CFS have no difference" is considered. The Friedman's test is performed using six feature selection methods and so the degrees of freedom are 5. There are 135 accuracy values for each of the methods. The results of the Friedman's test in terms of the mean of ranks of UFRFS, Algorithm 1, UFRDR, UFSFS, FRMIM, and CFS are provided in Table X. The value of the test statistic (χ^2) and the corresponding p-value are obtained as 207.29 and $7.83331e^{-43}$, respectively, and are also shown in the table. From the results, we reject the null hypothesis with the significance level $7.83331e^{-43}$. This suggests that a *post hoc* test is needed to check whether there exists any statistically significant difference between the mean ranks of the proposed UFRFS and any other feature selection method. In this regard, the Bonferroni–Dunn *post hoc* test is conducted with the null hypothesis that "there is no difference between the mean ranks of two methods." The performance of the two methods will be significantly different if the difference between the mean ranks of the methods is greater than the critical difference (CD). The value of CD is found to be 0.8061, which is calculated using critical value 3.540 at significance level 0.002 (for six feature selection methods) and standard error 0.2277. The significance level 0.002 is obtained by the Bonferroni correction (0.01/5) of standard significance level 0.01 for five comparisons. The results of Bonferroni–Dunn *post hoc* test are shown in Table XI. The differences between the mean ranks for UFRFS and Algorithm 1, UFRFS and UFRDR, and UFRFS and UFSFS are found to be 3.16, 6.67, and 6.7, respectively, which are greater than the CD (0.8061). The results indicate that the null hypothesis should be rejected and the UFRFS performs significantly better than the unsupervised methods in Algorithm 1, UFRDR, and UFSFS. The differences between the mean ranks for UFRFS and FRMIM and UFRFS and CFS are -4.7 and -1.81, respectively, which are less than CD (0.8061). Hence, we fail to reject the null hypothesis at significance level 0.002. Note that, as FRMIM and SFS are supervised methods, it is expected that the genes selected by them should be more significant than the unsupervised UFRFS in terms of the accuracy values. The details of the statistical test procedures are available at http://avatharamg.webs.com/GSOM-UFRFS.pdf.

TABLE XII
BIOLOGICAL SIGNIFICANCE OF GENES, SELECTED BY THE UFRFS

Data set	GO biological category	Gene ontology term	p-value	Adjusted p-value
DLBCL	Biological process	translational elongation	3.767e-21	8.27e-18
	Molecular function	structural constituent of ribosome	7.014e-14	6.909e-11
	Cellular component	cytosolic ribosome	4.833e-19	1.121e-16
Breast cancer	Biological process	cellular homeostasis	9.058e-8	0.0001268
	Cellular component	extracellular space	3.032e-9	7.033e-7
Multi-A	Biological process	translational elongation	2.36e-19	5.181e-16
	Molecular function	structural constituent of ribosome	7.301e-14	7.191e-11
	Cellular component	cytosolic ribosome	1.796e-25	4.167e-23
Lung cancer	Biological process	protein-chromophore linkage	6.204e-8	0.0001362
	Molecular function	antigen binding	1.982e-9	0.0000019
	Cellular component	membrane fraction	0.00029	0.0391
Diabetic data	Biological process	transforming growth factor beta receptor signaling pathway	1.004e-8	2.2050e-005
	Molecular function	oxygen transporter activity	6.1590e-005	0.03033
	Cellular component	fibrillar collagen	5.2820e-005	0.01225
Prostate data	Biological process	smooth muscle contraction	2.4640e-06	0.00540
	Cellular component	contractile fiber	1.6890e-005	0.0039
Cigarette smokers female	Biological process	oxygen transport	4.333e-15	9.511e-12
	Molecular function	oxygen transporter	1.281e-15	1.262e-12
	Cellular component	cytosolic part	6.826e-9	1.5840e-006
Cigarette smokers male	Biological process	oxygen transport	4.34e-15	9.526e-12
	Molecular function	oxygen transporter activity	1.283e-15	1.264e-12
	Cellular component	cytosolic part intermediate filament	1.025e-10	2.377e-8
Resistance data	Cellular component	steroid dehydrogenase activity	5.026e-006	0.004951
Upf1 null mutant	Molecular function	alpha-1, 3-mannosyltransferase activity	1.2620e-004	0.04934

G. Biological Significance

The significance of biological categories like biological process and molecular function of genes selected by the UFRFS is determined using FatiGO genome database (http://fatigo.bioinfo.cnio.es) [35]. The gene ontology (GO) terms associated with the set of genes are evaluated using p-value and adjusted p-value (using Bonferroni correction) and the most significant subcategory under each main

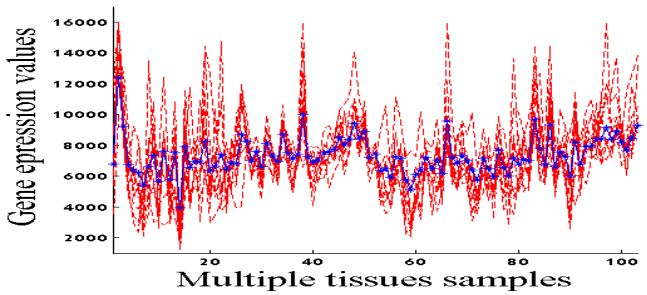

Fig. 9. Plot of the expression profiles for ten genes (RPS11, RPL19, RPS16, RPL23A, RPL18A, RPS27, RPS3A, RPL38, RPS15, and RPS19) belonging to GO biological process translational elongation for multi-A data with 103 samples. The average of Pearson's correlations for all possible gene pairs is 0.77. The average of expression profiles is shown in blue color.

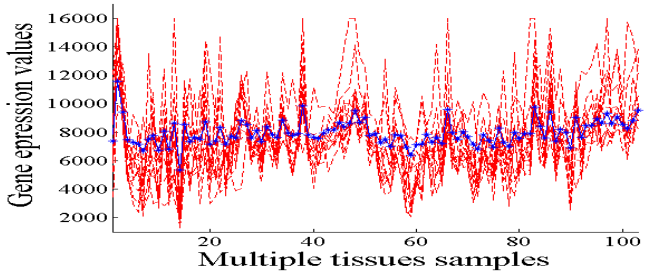

Fig. 10. Plot of the expression profiles for 11 genes (EIF2AK2, HSPB1, RPS16, RPL8, RPL23A, RPL18A, RPS27, RPS3A, RPL38, RPL10 and RPS18) belonging to GO biological process 'translation' for multi-A data with 103 samples. The average of Pearson's correlations for all possible gene pairs is 0.76. The average of expression profiles is shown in blue color.

category is provided in Table XII for UFRFS using all the data sets except ethanol, where no significant GO term is obtained for any of the methods. The other significant subcategories are provided in Table II of the supplementary file (http://avatharamg.webs.com/GSOM-UFRFS.pdf). For example, out of 50 selected genes in multi-A data, 46 genes belong to biological process and out of them ten genes belong to the most significant subcategory translational elongation which is shown in Table XII. Two other subcategories within biological process are translation (11 genes) and tissue development (nine genes) and shown in Table II of the supplementary material online.

In the biological process category, the characteristics and relationship of the genes (selected by the UFRFS) in terms of gene profiles are provided for multi-A data as an example. The profiles of ten genes (RPS11, RPL19, RPS16, RPL23A, RPL18A, RPS27, RPS3A, RPL38, RPS15, and RPS19) belonging to translational elongation and 11 genes (EIF2AK2, HSPB1, RPS16, RPL8, RPL23A, RPL18A, RPS27, RPS3A, RPL38, RPL10, and RPS18) belonging to translation for multi-A data are shown in Figs. 9 and 10, respectively. The figures also provide the average profiles of genes shown in blue line. It is evident from the figures that the profiles are very similar for most of the genes within a subcategory. For example, by considering ten genes corresponding to Fig. 9, the average of Pearson's correlations for all possible gene pairs (90 pairs excluding self similarity) is 0.77. Similarly, for 11 genes, corresponding to Fig. 10, the average of Pearson's

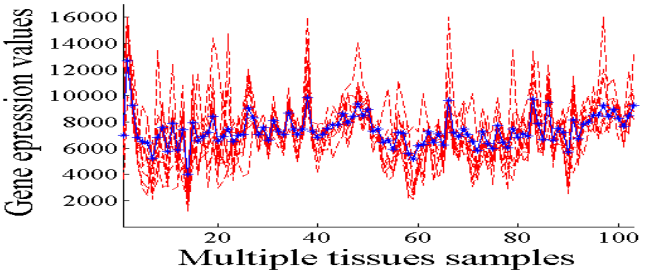

Fig. 11. Plot of the expression profiles for 9 genes (RPL8, RPL10, RPS16, RPL23A, RPL18A, RPL38, RPS15, RPS18 and RPS19) belonging to GO molecular function 'structural constituent of ribosome' for multi-A data with 103 samples. The average of Pearson's correlations for all possible gene pairs is 0.78. The average of expression profiles is shown in blue color.

correlations is 0.76. Similar type of correlation is also observed for other subcategories as in Fig. 11 as an example. Here, expression profiles of nine genes are presented and the average of Pearson's correlations for all possible gene pairs is 0.78. Moreover, all of the gene expressions profiles are seen to be correlated with the average of those profiles.

Further, from Figs. 9 and 10, it can be observed that there is also some correlation between the genes in two different subcategories (translational elongation and translation) within a main category (biological process) and this correlation is less than those of the genes within individual subcategory. This visual interpretation is also supported by the fact when we computed the average of Pearson's correlations as 0.71 (which is less than both 0.77 and 0.76) by considering all the genes in subcategories translational elongation and translation.

Even for two different categories there exist correlation in expression profiles for the genes selected within the same data. It is clear from Fig. 11, which shows expression profile of 9 genes for subcategory structural constituent of ribosome within main category molecular function, and Fig. 9 (described above) that there exists some correlation between the gene profiles for different subcategories using multi-A data. This fact is also established when we considered all the genes within subcategory structural constituent of ribosome and translational elongation together and found the average of Pearson's correlations for all possible gene pairs as 0.72.

In another example for prostate cancer data, out of ten selected genes three genes belong to category cellular component contractile fiber (shown in Table XII) and three genes belong to category cellular component actin cytoskeleton. The profiles of three genes belonging to contractile fiber are shown in Fig. 12 using 102 samples. From the figure, it can be observed that the profiles of the genes are similar (correlated). Here, the average of Pearson's correlations between three pairs of genes (excluding self similarity) is 0.88.

The functionally enriched biological categories should be related to the cancer type of the data set. In this regard, we conducted a literature study. For example, the breast cancer data shows functional enrichment in the categories biological process, cellular homeostasis and cellular component, extracellular space. These functional categories are related to breast cancer according to [36]. In a similar way, the relations

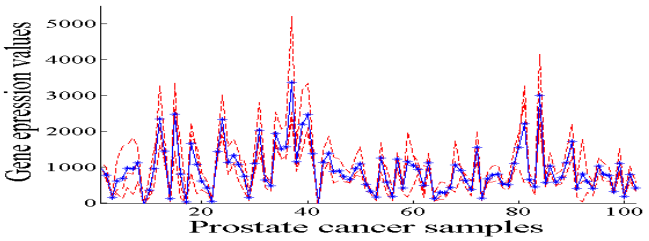

Fig. 12. Plot of the expression profiles for 3 genes (37407_s_at, 32755_at, and 767_at) belonging to GO cellular component contractile fiber for prostate cancer data with 102 samples. The average of Pearson's correlations for three pairs of genes is 0.88. The average of expression profiles is also shown.

TABLE XIII
BIOLOGICAL COMPARISON OF GENES SELECTED BY UFRFS, ALGORITHM 1, AND FRMIM FOR ALL AND AML DATA

Data set	Method	GO biological category	Gene ontology term	*p*-value	Adjusted *p*-value
ALL & AML	UFRFS	Biological process	oxygen transport	3.725e-10	8.176e-7
			gas transport	1.438e-9	1.579e-6
		Molecular function	oxygen transporter activity	1.23e-10	1.211e-7
			oxygen binding	2.711e-8	1.335e-5
		Cellular component	cytosolic part	3.165e-7	7.343e-5
			extracellular space	3.105ee-4	0.03602
	Algorithm 1	Biological process	oxygen transport	1.996e-7	3.862e-4
			gas transport	4.988e-7	3.862e-4
		Molecular function	oxygen transporter activity	9.666e-8	9.521e-5
			oxygen binding	3.961e-6	1.951e-3
		Cellular component	cytosolic part	1.107e-5	0.002567
	FRMIM	Biological process	glial cell differentiation	2.401e-5	0.04159
			gliogenesis	3.79e-5	0.04159

of functional categories with cancers for other data sets are mentioned in Table III of the supplementary file.

Table XIII compares the proposed UFRFS, Algorithm 1, and FRMIM using ALL and AML data. The genes selected by UFSFS, UFRDR, and CFS are not biologically significant for this data and hence not provided in the table. The p-values and adjusted p-values for GO terms using UFRFS are lower than those of Algorithm 1 and FRMIM for all the biological categories. No significant GO terms belonging to molecular function and cellular component are found using FRMIM.

VI. CONCLUSION

A new GSOM is developed by integrating fuzzy rough sets with SOM to capture the uncertainty and underlying clusters in the data. Fuzzy rough sets use fuzzy reflexive relation and fuzzy decision classes, which deal with the vagueness in the class information. The new neighborhood function in GSOM updates the connection weights associated with the output nodes. Based on the output clusters of GSOM, the dependence factor of each attribute (gene), with respect to all the clusters, is computed using fuzzy rough sets. Lower the dependence degree of a gene is, the higher its relevance is in diagnosing cancer. Using this criterion, genes are ranked for selection.

The superiority of GSOM, as compared with RRFCM, SOM, RFPCM, RPCM, FCM, c-medoids, and AP method is demonstrated in clustering microarray data in terms of β-index, DB-index, Dunn-index, and FRE. Intuitively, for handling data with overlapping patterns (like microarrays), any RFC method, like GSOM, will perform better than SOM, c-medoids, AP method, fuzzy clustering (e.g., fuzzy c-means), and rough clustering method (e.g., RPCM) as RFC has one/two more component/s. Furthermore, the conventional SOM has advantages over c-means when the shape of the real clusters in the data is noncircular. Hence, the performance of GSOM is expected to be better than the RPCM, FRPCM, and RRPCM which uses c-means instead of SOM. Moreover, the granulation structures in GSOM preserve the relative information among patterns using fuzzy similarity matrix and then transfer it to the neighborhood function through rough approximation operators, which is missing in other rough fuzzy clustering methods.

The genes selected by the UFRFS are more relevant than those of the other unsupervised methods, in terms of feature evaluation index and classification accuracy in most of the cases. The proposed UFRFS provides statistically more significant results than related unsupervised methods (Algorithm 1, UFRDR, and UFSFS) and have similar expression profiles. The genes selected by the UFRFS are more biologically meaningful for all data sets except ethanol data, where no significant biological terms are found.

ACKNOWLEDGMENT

Prof. S. K. Pal would like to thank the J. C. Bose Fellowship and the INAE Chair Professorship, Government of India.

REFERENCES

[1] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets Syst.*, vol. 90, no. 2, pp. 111–127, 1997.
[2] L. Polkowski and A. Skowron, "Towards adaptive calculus of granules," in *Proc. 7th IEEE Int. Conf. Fuzzy Syst.*, Anchorage, AK, USA, May 1998, pp. 111–116.
[3] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464–1480, Sep. 1990.
[4] J. P. Herbert and J. Yao, "A granular computing framework for self-organizing maps," *Neurocomputing*, vol. 72, nos. 13–15, pp. 2865–2872, 2009.
[5] S. K. Pal, B. Dasgupta, and P. Mitra, "Rough self organizing map," *Appl. Intell.*, vol. 21, no. 3, pp. 289–299, 2004.
[6] A. Ganivada, S. Dutta, and S. K. Pal, "Fuzzy rough granular neural networks, fuzzy granules, and classification," *Theoretical Comput. Sci.*, vol. 412, no. 42, pp. 5834–5853, 2011.
[7] A. Ganivada, S. S. Ray, and S. K. Pal, "Fuzzy rough granular self-organizing map and fuzzy rough entropy," *Theoretical Comput. Sci.*, vol. 466, pp. 37–63, Dec. 2012.
[8] P. Maji and S. K. Pal, "Rough set based generalized fuzzy C-means algorithm and quantitative indices," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 6, pp. 1529–1540, Dec. 2007.
[9] P. Maji and S. Paul, "Rough-fuzzy clustering for grouping functionally similar genes from microarray data," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 10, no. 2, pp. 286–299, Mar./Apr. 2013.
[10] S. S. Ray, S. Bandyopadhyay, and S. K. Pal, "Dynamic range-based distance measure for microarray expressions and a fast gene-ordering algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 3, pp. 742–749, Jun. 2007.
[11] J.-H. Chiang and S.-H. Ho, "A combination of rough-based feature selection and RBF neural network for classification using gene expression data," *IEEE Trans. Nanobiosci.*, vol. 7, no. 1, pp. 91–99, Mar. 2008.
[12] P. Maji and S. K. Pal, "Fuzzy–rough sets for information measures and selection of relevant genes from microarray data," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, pp. 741–752, Jun. 2010.
[13] S. K. Pal, R. K. De, and J. Basak, "Unsupervised feature evaluation: A neuro-fuzzy approach," *IEEE Trans. Neural Netw.*, vol. 11, no. 2, pp. 366–376, Mar. 2000.
[14] R. Jensen and Q. Shen, "New approaches to fuzzy-rough feature selection," *IEEE Trans. Fuzzy Syst.*, vol. 17, no. 4, pp. 824–838, Aug. 2009.

[15] N. M. Parthaláin and R. Jensen, "Unsupervised fuzzy-rough set-based dimensionality reduction," *Inf. Sci.*, vol. 229, pp. 106–121, Apr. 2013.
[16] P. Mitra, C. A. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 301–312, Mar. 2002.
[17] A. M. Radzikowska and E. E. Kerre, "A comparative study of fuzzy rough sets," *Fuzzy Sets Syst.*, vol. 126, no. 2, pp. 137–155, 2002.
[18] D. S. Yeung, D. Chen, E. C. C. Tsang, J. W. T. Lee, and W. Xizhao, "On the generalization of fuzzy rough sets," *IEEE Trans. Fuzzy Syst.*, vol. 13, no. 3, pp. 343–361, Jun. 2005.
[19] A. Ganivada, S. S. Ray, and S. K. Pal, "Fuzzy rough sets, and a granular neural network for unsupervised feature selection," *Neural Netw.*, vol. 48, pp. 91–108, Dec. 2013.
[20] S. K. Pal and S. Mitra, "Multilayer perceptron, fuzzy sets, and classification," *IEEE Trans. Neural Netw.*, vol. 3, no. 5, pp. 683–697, Sep. 1992.
[21] H. Yan, "Convergence condition and efficient implementation of the fuzzy curve-tracing (FCT) algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 210–221, Feb. 2004.
[22] T. R. Golub *et al.*, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, no. 5439, pp. 531–537, 1999.
[23] D. Singh *et al.*, "Gene expression correlates of clinical prostate cancer behavior," *Cancer Cell*, vol. 1, no. 2, pp. 203–209, 2002.
[24] S. Monti *et al.*, "Molecular profiling of diffuse large B-cell lymphoma identifies robust subtypes including one characterized by host inflammatory response," *Blood*, vol. 105, no. 5, pp. 1851–1861, 2005.
[25] Y. Hoshida, J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Subclass mapping: Identifying common subtypes in independent disease data sets," *PLoS One*, vol. 2, no. 11, pp. e1195-1–e1195-8, 2007.
[26] L. J. van't Veer *et al.*, "Gene expression profiling predicts clinical outcome of breast cancer," *Nature*, vol. 415, no. 6871, pp. 530–536, 2002.
[27] A. Bhattacharjee *et al.*, "Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses," *Proc. Nat. Acad. Sci. USA*, vol. 98, no. 24, pp. 13790–13795, 2001.
[28] U. Alon *et al.*, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proc. Nat. Acad. Sci. USA*, vol. 96, no. 12, pp. 6745–6750, 1999.
[29] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.
[30] S. K. Pal, A. Ghosh, and B. U. Shankar, "Segmentation of remotely sensed images with fuzzy thresholding, and quantitative evaluation," *Int. J. Remote Sens.*, vol. 21, no. 11, pp. 2269–2300, 2000.
[31] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979.
[32] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *J. Cybern.*, vol. 3, no. 3, pp. 32–57, 1973.
[33] M. A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning," in *Proc. 17th Int. Conf. Mach. Learn.*, San Francisco, CA, USA, Jun. 2000, pp. 359–366.
[34] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.
[35] F. Al-Shahrour, R. Díaz-Uriarte, and J. Dopazo, "FatiGO: A Web tool for finding significant associations of gene ontology terms with groups of genes," *Bioinformatics*, vol. 20, no. 4, pp. 578–580, 2004.
[36] C. Dabrosin, J. Chen, L. Wang, and L. U. Thompson, "Flaxseed inhibits metastasis and decreases extracellular vascular endothelial growth factor in human breast cancer xenografts," *Cancer Lett.*, vol. 185, no. 1, pp. 31–37, 2002.

Shubhra Sankar Ray received the M.Sc. degree in electronic science and the M.Tech. degree in radio physics and electronics from the University of Calcutta, Kolkata, India, in 2000 and 2002, respectively, and the Ph.D. (Eng.) degree from Jadavpur University, Kolkata, in 2008.

He was a Post-Doctoral Fellow with the Saha Institute of Nuclear Physics, Kolkata, from 2008 to 2009. His current research activities are in Bioinformatics, Neural Networks, Genetic Algorithms and Soft Computing. Three of his publications are listed as curated paper in Saccharomyces Genome Database, Stanford University, Stanford, CA, USA. His current research interests include bioinformatics, neural networks, genetic algorithms, and soft computing.

Dr. Ray was a recipient of the Microsoft Young Faculty Award in 2010.

Avatharam Ganivada received the B.Sc. (Hons.) degree in statistics and the M.Sc. degree in mathematics from Andhra University, Visakhapatnam, India, in 2003 and 2005, respectively, and the M.Tech. degree in computer science and technology from the University of Mysore, Mysore, India, in 2008.

He has been with the Center for Soft Computing Research, Indian Statistical Institute, Kolkata, India, since 2009, as a Visiting Scientist. He has submitted his Ph.D. thesis to the University of Calcutta, Kolkata. His current research interests include fuzzy rough sets, neural networks, pattern recognition, and bioinformatics.

Sankar K. Pal (M'81–SM'84–F'93) received the Ph.D. degrees from the University of Calcutta, Kolkata, India, and Imperial College, London, U.K.

He joined the Indian Statistical Institute, Kolkata, in 1975, as a CSIR Senior Research Fellow, where he became a Full Professor in 1987, a Distinguished Scientist in 1998, and the Director in 2005. He founded the Machine Intelligence Unit and the Center for Soft Computing Research with Institute in Calcutta, which is enjoying international recognition. He was with the University of California at Berkeley, Berkeley, CA, USA, the University of Maryland, College Park, MD, USA, NASA-JSC, Houston, TX, USA, and the U.S. Naval Research Laboratory, Washington, DC, USA. He has been an IEEE-CS Distinguished Visitor since 1987, and held several visiting positions in Italy, Poland, Hong Kong, and Australia. He is currently a J. C. Bose Fellow of the Government of India and the INAE Chair Professor. He has co-authored 17 books and over 400 research publications in the areas of pattern recognition, machine intelligence, image processing, data mining, Web intelligence, soft computing, bioinformatics, and cognitive machines.

Dr. Pal is a fellow of the Academy of Sciences for the Developing World, the International Association for the Psychology of Religion, the International Forestry Students' Association, and four national academies for science/engineering in India. He received several national and international awards, including the most coveted S.S. Bhatnagar Prize in India in 1990, the Padma Shri Award in 2013, and the Al-Khwarizmi International Award in 2000. He is/was on the editorial boards of 20 journals, including some IEEE TRANSACTIONS.