Rough Tolerance Growing Self-Organizing Map

(RT-GSOM)

Anima Pramanik, Sobhan Sarkar, J.Maiti, Pabitra Mitra

## Abstract

Rough tolerance growing self-organizing map (RT-GSOM) is an extension of growing self-organizing map (GSOM), which is a variation of popular self-organizing map (SOM). RT-GSOM is developed to address the issues of i) identifying the suitable shape and size of clusters in SOM and ii) fuzzy discretized data clustering problem in GSOM. Here, we integrate rough tolerance with GSOM to offer a fast and robust solution to the fuzzy discretization of feature space problem; thereby designing RT-GSOM. Two types of discernibility reducts are obtained using rough set theory (RST) and rough tolerance value (RTV). Discernibility reducts are used to extract domain knowledge in an unsupervised framework, are further used to determine the initial weights of the network, which are further refined using competitive learning. Superiority of the network in terms of quality of clusters, learning time and representation of data is demonstrated quantitatively through experiments over the conventional state-of-the-art algorithms.

## Keywords:

Self-organizing map, rough sets, data mining, growing self-organizing map, rough tolerance

# Introduction

A clustering is an unsupervised method to group the elements or objects based on some similarity (Vesanto & Alhoniemi, 2000). The aim of this approach is to find a structure in a collection of unlabeled data. While the data are unlabeled, the proper visualization and summaries become very hard to obtain. Under these circumstances, clustering is an effective approach that helps in better visualization and descriptions of data. There are several clustering algorithms available, for example, k-means, hierarchical, self-organizing map (SOM), etc., which are widely used in diverse application areas, such as pattern recognition, machine learning, image processing, statistics, and business, etc.

Of them, SOM is very popular because it can reduce an N dimensional feature representation to a 2D or 1D representation with prominent visualization properties. It creates a set of prototype vectors representing the data set and carries out a topology preserving projection of the prototypes from the high dimensional input space onto a low-dimensional grid. This ordered grid can be used as a convenient visualization surface for showing different features of the SOM (and thus of the data), for example, the cluster structure [3].

However, while using SOM, the size of the map and the number of nodes are to be predetermined. The need for predetermining the structure of the network results in a significant limitation on the final mapping. It is often known only at the completion of the simulation that a different sized network would have been more appropriate for the application. Therefore, simulations have to be run for several times on different sized networks to pick the optimum network [10]. A further limitation when using SOM for algorithm clustering occurs due to the structure of the sample sets is unknown. Therefore, it not only becomes difficult to predetermine the size of the network but it is also not possible to identify when the map is organized into a proper cluster structure. The solution to this problem is to determine the shape as well as the size of the network during the training of the network: the growing self-organizing map (GSOM) [11], an extension of the SOM. In addition, SOM suffers from the problem of slow convergence and local minima; Rough-SOM (RSOM) is developed by Pal et al. (2004), which offer a fast and robust solution to the initialization and local minima problem. The working principle of RSOM is based on rough set theory (RST), which becomes an important tool for handling inconsistency and uncertainty inherent to decision. In RSOM, discernibility reducts are obtained using RST, which are further used to extract domain knowledge. Whole procedure is done in an unsupervised way as RST uses only internal knowledge of input data and never depend on previous model assumptions. This network shows superiority over traditional SOM in terms of quality of clusters, learning and representation of data. However, there is a drawback of RST, i.e., the equivalence relation of RST is constructed based on equal relation of the attribute values, which results in reduction of information at boundary region, which is the most sensitive portion in decision space (Borderline SMOTE->). To solve the above-mentioned problems in the algorithms, i.e., SOM, GSOM, and RSOM, a new clustering algorithm, namely rough tolerance GSOM (RT-GSOM) is developed in this study. Here, rough tolerance set is extracted based on the predefined threshold value that is further incorporated with GSOM. By doing so, we solve the aforesaid inherent issues of SOM, RSOM, and GSOM algorithms.

The remaining paper contains as follows: Section 2 presents preliminaries of our work. Our proposed design and methodology are shown in Section 3. Section 4 shows the experimental result. Section 5 concludes the paper.

# Preliminaries

In this section, we briefly present some preliminaries of the methods, namely Rough Set Theory [citation], Tolerance Rough Set, and Growing Self-organizing Map (GSOM) [citation].

## 2.1 Rough Set theory (RST)

Rough set theory is introduced by Pawlak in 1982 to deal with the uncertainty in data. A universeis a pair, where represents the non-empty finite set called decision table or universe, and denotes a non-empty finite set of attributes. An attribute *a* can be regarded as a function from the domain *U* to some value set. With every subset of attributes, one can easily associate an equivalence relation on, expressed as:



and the equivalence class containing *x* in is given by 

If, the sets and are called the *B-Lower*and *B-Upper approximation*of *X* in *S*, where denotes the equivalence class of the object relative to.

# 2.2 Tolerance rough set (TRS)

Tolerance rough set [6] is an extension of rough set by replacing equivalence relation with similarity relation. Here, the target concept is as same as in the rough set.

Given a universe, where B is the similarity region defined on *U*, if and only if *B* satisfies the two conditions i.e. reflexivity and symmetry, given below:

1. Reflexivity, i.e. for each 
2. Symmetry, i.e. for each 

There are three types of similarity relations on *U*, which are defined in [6]. It can be expressed by Eq. (2)



In Eq. (2),and  is the variance of values in attribute *a*.

The similarity class of *x* is defined in the following way:



Where, is the similarity threshold and, .

Given an information table and a similarity threshold for arbitrary decision class, the tolerance lower approximation and upper approximation of with respect to B are defined by Equation. (4)-(5).





## 2.3Growing Self Organizing Map (GSOM)

The GSOM algorithm is developed to address some of the weaknesses faced by the traditional SOM algorithm, which include the problem of identifying the correct dimensions (height and width) of the rectangular map [7]. The GSOM algorithm typically consists of three phases: i) initialization phase, ii) growing phase, and iii) smoothing phase. The detailed explanation of each of the three phases is given as follows:

**2.3.1. Initialization phase**

The network is initialized with an input space of four nodes, as illustrated in Fig 1. Here, all the starting nodes act as the boundary nodes and are free to grow in its own direction (four directions are considered). Four nodes are initialized with random weight vectors. Since all the input attribute vectors are normalized in the range [0, 1], the initial weight vector  attributes can take any random values in this range closer to 0.5. Thus, the initial square shape lets the map grow in any of four directions solely relying on the input space.



Fig 1: Growing of boundary winning node in GSOM

**2.3.2. Growing phase**

After the initialization phase, this phase starts with the minimal number of nodes considered in the previous stage. Nodes are gradually spread depending on the distribution of the data. The detailed procedure of growing phase is described below [7].

1. First, a weight vector, closest to the input vector space, *x* is determined based on Euclidean distance. The winning vector () is shown in Eq. 6.



1. If is the boundary node, then total error is accumulated on this node, otherwise the errors are shared betweenand its neighbors, which is depicted in Eq. 7.



Where *E* is the accumulated error from each node, *t* is the input data, *D* is the dimension of the input vector space, and is the wining weight vector. The error value of will be accumulated continuously during the training of network until it exceeds the value of growth threshold  [7]. is calculated by Eq. (8).



Where is the spread factor between [0,1] to determine the level of spread required by the map. A highervalue gives a wider spread and more detailed clusters, whereas a lower gives more number of concrete clusters.

1. When an accumulated error value of the boundary winning node in the network exceeds the, a new node is generated as shown in Fig--. Here, each boundary node can generate new node in its neighboring directions.
2. The nodes other than the boundary winning nodes, the weight adaptation between winner and its neighbors are done in a similar manner as in SOM. The training of SOM is done by the following ways:
3. First, the topology of SOM network is defined. The weight vectors,  are initialized randomly.
4. Then, an input space *x* is provided to the network. Consequently, the winning output node *J* is determined, where the distance between *j* and input space is closest to *x*, i.e.,



1. Thereafter, the weight vectors are updated using the following equation.



Where is the neighborhood function defined as:



Whereis the monotonically decreasing learning rate, represents the position of the corresponding node, and  implies the monotonically decreasing kernel width function. Here if node c belongs to the neighborhood of the winning node J, otherwise.

1. Repeat steps (b)-(d) until no change of neuron is observed.
2. Repeat steps (i)-(v) until all the inputs are passed through the whole network and the node growth is reduced to a minimum level.

**2.3.3. Smoothing phase**

The growing phase stops when the generation of new nodes gets saturated. After the completion of growing phase, the weight adaptation is continued at a lower rate. The main purpose of this phase is to smooth out any existing quantization error in the nodes grown the latter stages of the growing phase [7]. Since the quantization error should not fluctuate too much, the starting learning rate in this phase is less than that of the growing phase. The smoothing phase is stopped when the error values in the map become substantially small.

# 3. Proposed Rough Tolerance GSOM: An extension of GSOM

Our proposed rough tolerance-based GSOM (RT-GSOM) consists of two phases: i) setting up a rough tolerance, and ii) hybridization of rough tolerance sets in GSOM. First, domain knowledge-based discernibility reducts are obtained from the input feature space in an unsupervised way. After getting the reducts, a tolerance value is introduced in extracted rough reducts to get rough tolerance set. Finally, set, namely ‘rough lower approximation’ is used as inputs to the GSOM. Detailed description of each phase is given below.

## 3.1 Rough Tolerance Set

Rough tolerance (RT) set consists of the *rough lower approximation*. Rough lower approximation set is obtained by the positive region of the equivalence relation. The details of extracting of RT sets are given below.

Given, an information table, *U* represents the universe, denotes the attributes, and implies the decision class in *U*. The steps of obtaining rough tolerance set are given below.

*Step 1: Rough approximation set extraction*

Second, rough approximation sets are obtained from *U* using equivalence relation, such that. The sets are expressed as from Eqs. (11-13).





Whereand  represent the rough lower approximation set and rough upper approximation set respectively.

|  |  |
| --- | --- |
| **Algorithm 1:** Rough approximation set | |
| **Inputs:** Information table,,, and, where N represents the number of instances in , denotes the number of attributes in , and indicates the number of decision classes in . | |
| **Outputs:** A rough approximation set, and. | |
| 1 | **for** i=1; i<k; i++ |
| 2 | , , and (is a decision attribute) |
| 3 | **end for** |
| 4 | **for** i=1; i<k; i++; |
| 5 | **for** j=1; j<k; j++; |
| 6 | **if**and |
| 7 | based on |
| 8 |  |
| 9 |  |
| 10 | **elif**and |
| 11 | Repeat Step 7 and 8 |
| 12 |  |
| 13 |  |
| 14 | **end if** |
| 15 | **end for** |
| 16 |  |
| 17 |  |
| 18 | **end for** |

**3.2 Incorporation of Rough Tolerance Set in GSOM**

In this section, the rough tolerance set is incorporated with the GSOM. Here, we have used the rough tolerance dependency and lower discernibility reducts to get the crude knowledge of the clusters information of the input pattern to be fed to The GSOM. Since, the crude knowledge of the lower approximation is encoded into the GSOM network; the learning time reduces greatly with improved performance.

The steps involved in the process, are stated below:

1. From the input feature space, a rough tolerance information table is extracted (Section 4.1).
2. The growing self-organizing map (GSOM) is initialized at the competitive layer of grid units, using (*C* is the number of decision classes and *n* is the number of informative instances under each decision class), which correspond to the attributes *A*. The structure of the competitive layer gradually grows with the.
3. The winning node  in the competitive layer is determined based on the closest distance (shown in Eq. 6) between each weight vector in the competitive layer and. Then, GSOM network gradually formed on the competitive layer, as discussed in Section 3.3.

Discernibility reduct and rough tolerance instances under a decision class are the most sensitive to its class compared to other decision classes. Therefore, the learning process in RT-GSOM becomes more efficient compared to state-of-the-art SOM and GSOM.

# 4. Result & Discussion

In order to verify the effectiveness of the proposed algorithm RT-GSOM, we experimentally compared RT-GSOM with three state-of-the-art approaches which are SOM [2], GSOM [3] and RSOM [4] on three aspects: Siloutee index, DB index and CPU time. The details are given below:

## 4.1 Experimental Setup

The code we have used in executed in Windows 10 with Python 3.6 (anaconda) using an Intel i3 processor clocked at 3.30 GHz and 8.00 GB of memory. The libraries used are numpy, pandas, scipy, math and random. Our experiment consists with three phases: i) rough lower approximation set generation, ii) rough tolerance set generation and iii) incorporate both the two sets with GSOM. There is no threshold used for generation of rough lower approximation set. We have used a tolerance score for rough tolerance generation, which is discussed in Section 3.1. Growth threshold is used in GSOM for growing of this network. The value of GT is depend on the spread factor, which we have considered as in this experiment

## 4.2 Dataset used

In this experiment, we use 15 different benchmark datasets collected from the University of California Irvine (UCI) machine learning repository, which is freely available in the website (<https://archive.ics.uci.edu/ml/datasets.html>). A brief summary of the 11 datasets with number of conditional attributes, number of instances, i.e., sample size, and number of decision classes is presented in Table 1. The datasets are of continuous in nature with the varying level of sample size, and number of decision classes, which provide an adequate platform for the evaluation of our proposed algorithm.

Table 1: A summary of the used datasets

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset name | # Instances | # Attributes | # Decision class |
| Balance Scale | 625 | 5 | 3 |
| Ballons | 76 | 6 | 2 |
| Breast cancer | 286 | 9 | 2 |
| Connec | 518 | 10 | 3 |
| Hay | 132 | 6 | 3 |
| Monk | 554 | 8 | 2 |
| Nursery | 585 | 10 | 3 |
| Thiroid | 334 | 9 | 3 |
| Tumar | 339 | 7 | 2 |
| Voting Records | 428 | 18 | 2 |
| Ecoli | 336 | 9 | 8 |
| Glass | 214 | 11 | 6 |
| Image | 210 | 21 | 7 |
| Iris | 150 | 6 | 3 |
| Seeds | 210 | 9 | 3 |

## 4.3 Performance metric

We have extracted informative input space from the original information table that is further fed to the GSOM network to get better clustering accuracy and computational complexity. The following performance metrics used in this study are listed below:

1. Quantization Error (QE)

The quantization error [4] measures how fast the weight vector of the winning node in the competitive layer are aligning themselves with the input space presented during training. *QE* is expressed as:



where, *N* is the number of instances in *U*, is the component in decision class, is the total number of patterns and is the winning node with respect to . Hence, higher the, more is the difference between winning node and input vector node in the competitive layer.

1. Silhouette Index (SI)

The silhouette index [8] is a measure of how similar an instance is to its own class compared to other classes. Is expressed as:



Whereis the silhouette index of cluster, expressed as: here,is the silhouette width of the vector in the cluster, expressed as: . is the average distance between vector in cluster with the other vectors in the same cluster, and is the minimum average distance between vector in the  cluster with all the vectors clustered in other than the cluster.

1. Davies-Bouldin (*DB*) Index

The *DB* index [8] is the measure of how good the clustering scheme is. DB index is expressed as:



where*K* is the total number of cluster in the input feature space, is the  cluster, is the intra-cluster distance in , expressed as: here, is the centroid diameter in the , expressed as: ,is the number of instances in , andis the inter-cluster distance, expressed as: , here, *d* is the centroid linkage inter-cluster distance i.e. Euclidean distance between and .

1. CPU Time

CPU time is the required time to execute the proposed algorithm. This is the measure of how fast our proposed algorithm compare to the state-of-the-art algorithms.

## 4.4 Overall performance

Overall performance of our proposed algorithm with respect to performance metric is evaluated here:

**4.4.1 Results of clustering**

The RT-GSOM is compared with three existing state-of-the-art clustering algorithm like SOM [2], GSOM [3] and RSOM [4].The performance of all the clustering algorithms is evaluated with three different cluster validation measures, measure [4], Silhouette index (*SI*) [8],and *DB* index [8]. For a cluster, the lower values of measure and *DB* index and higher value of Silhouette index signifies that the cluster is better [9]. These measures are considered in this study as these are widely studied and evaluated the compactness of output clusters in terms of low intra-cluster distance, low distance between winning node and input space and higher inter-cluster distances. Further are calculated based on Euclidean distance between instances of either inter-cluster or intra-cluster in the overall input feature space.

Before we explain the results of clustering, some key information about GSOM is mentioned here. We have started GSOM network topology with 4 nodes on the competitive layer. With the input feature instances GSOM gradually grown. The initial connection weights between nodes in the input feature space and the competitive layer are initialized by random number chosen within 0 to 0.5. The number of random samples is chosen for getting boundary set is 10. Tolerance value depends on the number of instances in *U* and the nearness relation between instances in *U*.

1. Selection of Parameters of RT-GSOM, GSOM, RSOM and SOM

Three parameters, initial learning rate, iteration *t*, and initial radius are used in RT-GSOM, GSOM, RSOM and SOM. [Citation] is chosen as the maximum number of neuron in either row or column in the competitive layer in RSOM and SOM. We set is set to 1 number of neuron in either row or column in the competitive layer for RT-GSOM and GSOM. For all RT-GSOM, GSOM, RSOM and SOM, the value of is varied between [0, 1] in steps of 0.05 for a particular number of iterations (*t*). Then, we repeat the same process for the iterations 50 to 200. Considering the initial number of clusters  and for RSOM and SOM, and  for RT-GSOM and GSOM. We proceed as follows for selecting  for different values of *t*. In this study, we got best value of and for RT-GSOM, whereas GSOM performs better at and, RSOM works better at  and and SOM gives good result at and . Therefore, further operations are done based on these parameters.

For choosing the number of best clusters (*c*), we varied *c* from 2 to [9] in steps of 1, where *N* is the total number of instances in *U*. for each value of c, we repeated the aforementioned procedure for different values of and t, and top four results (for four different c values), including the values of and *t* for all the datasets, in terms of Silhouette index, DB index and Quantization error, are shown in Table 3 - 5. RT-GSOM performs better compare to GSOM, RSOM and SOM in terms of performance metric. It can be concluded that Silhouette index, DB index and Quantization error all these data sets confirm *c* value equal to the number of clusters truly present in the data.

1. Visualization of output clusters for RT-GSOM, GSOM, RSOM and SOM

Fig 2shows the 2-D plot for all the samples normalized within [0, 1], demonstrate the actual classes in the dataset, namely ‘Ballons’. Fig 3a-3d show the corresponding output classes of RT-GSOM, GSOM, RSOM and SOM.



Fig 2: 2D plot of samples in Ballons

It is clear from Fig 2 that the boundary of the classes is overlap with each other. This overlapping information is used to extract rough tolerance set in RT-GSOM, which result is shown in Fig 3d, where exactly same clusters are obtained and the winner neurons for each class are represented as. Fig 3b and 3c, also show the actual cluster pattern for RSOM and GSOM. In Fig 3a, using SOM, overlapping patterns are not exactly same as in Fig 2. Moreover, the distance between the winner nodes in RT-GSOM is greater than GSOM, RSOM and SOM. As mentioned earlier, incorporation of the lower and upper approximation of rough set theory helps in determining the exact shape of the decision classes in the input feature space. Then, a tolerance value is applied on the resulted boundary region to extract actual boundary instances that is further used as an input to the RT-GSOM. Here, we consider tolerance value for choosing sensitive overlapping instances to handle the uncertainty arising in the overlapping region. In addition, GSOM grows in the competitive layer from 4 nodes, thereby reducing the competitive feature space. Aforementioned reasons make RT-GSOM superior to GSOM, RSOM and SOM.

1. Comparison of RT-GSOM with the state-of-the-art clustering algorithms for different dataset

The RT-GSOM is compared with the state-of-the-arts like GSOM, RSOM and SOM for 15 type of benchmark dataset that acquired from UCI Machine Learning Repository. The results are shown in Table 2-5. The related parameters of these algorithms are shown in the last column in table 3-5. Here, for GSOM, RSOM and SOM, the values of are provided, whereas the value of tolerance, are provided for RT-GSOM. For every dataset, best results are shown in bold for better understanding.

Table 2: Informative knowledge from raw data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset name | # Instances | # Lower Approximation | # Rough Tolerance | # Useful instances for RT-GSOM |
| Balance Scale | 625 | 225 | 47 | 272 |
| Ballons | 76 | 45 | 15 | 60 |
| Breast cancer | 286 | 80 | 69 | 149 |
| Connec | 518 | 112 | 249 | 361 |
| Hay | 132 | 40 | 63 | 103 |
| Monk | 554 | 50 | 160 | 210 |
| Nursery | 585 | 238 | 121 | 359 |
| Thiroid | 334 | 75 | 93 | 168 |
| Tumar | 339 | 41 | 98 | 139 |
| Voting Records | 428 | 107 | 111 | 218 |
| Ecoli | 336 | 336 | 0 | 336 |
| Glass | 214 | 214 | 0 | 214 |
| Image | 210 | 210 | 0 | 210 |
| Iris | 150 | 150 | 0 | 150 |
| Seeds | 210 | 210 | 0 | 210 |

Table 2 shows the result of rough tolerance boundary set. It is the knowledge reduction from raw data. This knowledge is further used for RT-GSOM. Whereas, lower approximation is used for RSOM and raw data is used for both SOM and GSOM.

Table 3: Comparison of clustering algorithms in terms of DB and SI index for c = 2, parameters used are mentioned in the table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data | Methods | DB Index | SI Index | Parameters |
| Ballons | SOM | 1.954027 | 0.192188 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |
| RSOM | 1.947744 | 0.191063 |
| GSOM | 0.931438 | 0.362952 |
| RT-GSOM | **0.765998** | **0.429071** |
| Breast cancer | SOM | 2.814005 | 0.107609 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |
| RSOM | 2.749144 | 0.111721 |
| GSOM | 0.545899 | 0.564122 |
| RT-GSOM | **0.175655** | **0.655997** |
| Monk | SOM | 2.72 | 0.045312 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |
| RSOM | 2.515 | 0.129059 |
| GSOM | 0.84 | 0.544446 |
| RT-GSOM | **0.592** | **0.701216** |
| Voting records | SOM | 4.367975 | 0.0487 |  |
| RSOM | 3.903219 | 0.056439 |
| GSOM | 0.677231 | 0.552567 |
| RT-GSOM | **0.610899** | **0.577027** |
| Tumar | SOM | 4.24413 | 0.051349 |  |
| RSOM | 0.719274 | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0.33171 |  |  |  |  | |
| GSOM | 0.940206 | 0.419779 |
| RT-GSOM | **0.344823** | **0.7665** |

Table 3 shows the result for 5 datasets having two decision classes. The results indicate that RT-GSOM is better than other state-of-the-art clustering approach in terms of DB-Index and Silhouette Index for all datasets. Table 4: Comparison of clustering algorithms in terms of DB and SI index for c = 3, parameters used are mentioned in the table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data | Methods | DB Index | SI Index | Parameters |
| Balance scale | SOM | 1.32173 | 0.01247 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  |  |  | |  |  |  |  | |
| RSOM | 1.12 | 0.242251 |
| GSOM | 0.913527 | 0.406363 |
| RT-GSOM | **0.774988** | **0.457052** |
| Connec | SOM | 2.537137 | 0.072379 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  |  |  | |  |  |  |  | |
| RSOM | 2.418137 | 0.09923 |
| GSOM | 0.830542 | 0.419319 |
| RT-GSOM | **0.561752** | **0.639903** |
| Hay | SOM | 1.866528 | 0.177012 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  |  |  | |  |  |  |  | |
| RSOM | 1.684054 | 0.158655 |
| GSOM | 0.826541 | 0.446068 |
| RT-GSOM | **0.576741** | **0.602581** |
| Nursery | SOM | 2.671912 | 0.00502 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  |  |  | |
| RSOM | 2.454891 | 0.098095 |
| GSOM | 0.766018 | 0.465389 |
| RT-GSOM | **0.405764** | **0.564505** |
| Thiroid | SOM | 2.499824 | 0.094982 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |
| RSOM | 2.494507 | 0.095026 |
| GSOM | 0.838463 | 0.507481 |
| RT-GSOM | **0.811663** | **0.521382** |
| Iris | SOM | 0.93021 | **0.872037** | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  | |
| RSOM | 0.93021 | **0.872037** |
| GSOM | **0.520395** | 0.532515 |
| RT-GSOM | **0.520395** | 0.532515 |
| Seeds | SOM | 2.468457 | 0.148255 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  | |
| RSOM | 2.468457 | 0.148255 |
| GSOM | **0.565209** | **0.534168** |
| RT-GSOM | **0.565209** | **0.534168** |

Table 4 shows the result for 7 datasets having three decision classes. The results reveal that RT-GSOM is superior to the state-of-the-art based on all the evaluation measures, except for iris and seeds. In iris, SOM and RSOM are superior to GSOM and RT-GSOM. In seeds, GSOM and RT-GSOM are superior to SOM and RSOM. RT-GSOM and RSOM both are working as same as with the GSOM and SOM clustering for two dataset namely, iris and seeds. The reason is that there is no overlapping boundary in indecision class in these two datasets. The same result also shown in Table 5.

Table 5: Comparison of clustering algorithms in terms of DB and SI index for c = 7 and 8, parameters used are mentioned in the table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data | Cluster number (c) | Methods | DB Index | SI Index | Parameters |
| Glass | 7 | SOM | 1.963992 | 0.094221 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  |  |  | |  |  |  |  | |
| RSOM | 1.963992 | 0.094221 |
| GSOM | **0.645986** | **0.709781** |
| RT-GSOM | **0.645986** | **0.709781** |
| Image | 7 | SOM | 3.443113 | 0.034346 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  |  |  | |  |  |  |  | |
| RSOM | 2.464308 | 0.055129 |
| GSOM | 0.705428 | 0.42188 |
| RT-GSOM | **0.626466** | **0.485367** |
| Ecoli | 8 | SOM | 1.488071 | 0.150952 | |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | |  |  |  |  | |  |  |  |  | |
| RSOM | 1.488071 | 0.150952 |
| GSOM | **0.862087** | **0.365143** |
| RT-GSOM | **0.862087** | **0.365143** |

The results in Table 5 for one data set with seven decision classes show that RT-GSOM outperforms the remaining state-of-the-art in terms of cluster evaluation measures. The datasets other than the dataset, namely Image, RT-GSOM acts as a conventional GSOM.

1. Comparison of RT-GSOM with other clustering algorithm in terms of number of clusters iterations

We compute the number of iterations at which error does not change much. The comparative result for only one dataset, namely Ballons is presented in Table 6.

Table 6: Comparison of clustering algorithms for dataset, namely ‘Ballons’ in terms of Quantization error

|  |  |  |  |
| --- | --- | --- | --- |
| **Dataset** | **Number of cluster** | **Method** | **Quantization error** |
| Ballons | 2 | SOM | 0.1 |
| RSOM | 0.07 |
| GSOM | 0.034 |
| RT-GSOM | 0.014 |

The following conclusion can be made from the obtained result, are given below:

* Better cluster quality

As seen from tables3-5 RT-GSOM has lower DB index, thus implying lower intra-cluster distance and higher inter-cluster distance in the clustered space compared to the GSOM, RSOM and SOM. Moreover, in table 6, RT-GSOM has lower than other state-of-the-art implies less difference between the reference vectors and the input vectors of the nodes in the competitive layer. RT-GSOM also has higher Silouhette index compared to other conventional state-of-the-art clustering algorithms, indicating more homogeneity within its clustered regions. Fig 3-4 show the comparison of clustering results of RT-GSOM, GSOM, RSOM and SOM for different clusters in terms of DB index and SI index. Both figure shows that RT-GSOM outperforms over the state-of-the-art GSOM, RSOM and SOM except for the four datasets. This is happening as there is no overlapping decision classes are occurred. Therefore RT-GSOM acts as a conventional GSOM.

Fig3: Comparison of clustering results of RT-GSOM, GSOM, RSOM and SOM in terms of DB index

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Fig. 4: Comparison of clustering results of RT-GSOM, GSOM, RSOM and SOM in terms of SI index

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* Quantization error minimization

The quantization error (qE) measures how fast the weight vectors of the winning nodes in the competitive layer are aligning themselves with the input vectors presented during training. The higher the quantization error (qE) more is the difference between the reference vectors and input vectors of the nodes in the competitive layer. Hence, quantization error (qE) should be less. It is observed that in case of RT-GSOM the quantization error (qE) is less compared to SOM, RSOM and GSOM .Fig 5 shows the comparison of quantization errors with number of clusters for all the used dataset for the proposed algorithm.

Fig 5: Comparison of clustering results of RT-GSOM, GSOM, RSOM and SOM in terms of Quantization error (qE).

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

From Fig. 5, it is clear that quantization error for all the datasets is very low for our proposed RT-GSOM.

* Compact representation of data

It is also seen from the experiment that in the case of RT-GSOM and RSOM fewer nodes in the competitive layer dominates other i.e. they win for most of the time for training data. Whereas, in conventional GSOM and SOM this number is higher. This is quantified by the frequency of winning of the top *k* nodes. It is observed that frequency of occurring of winning node is much higher in RT-GSOM and RSOM compared to the GSOM and SOM. In other words, RT-GSOM and RSOM achieve a more compact representation of the input feature.

**4.4.2. Computation complexity**

The computation complexity of RT-GSOM involving i) rough lower approximation, ii) sensitive rough set, iii) tolerance-based rough set and iv) GSOM network topology is as follows:

1. Complexity for rough lower approximation

The cost of computation in finding rough lower approximation is expressed as:



whereis the computation complexity to remove duplicate instances from the input feature space (*U*), here *N* is the total instances in *U*, is the frequency of occurrence of the instance in *U*. is the computation time to generate rough upper approximation set, here *P* is the total number of duplicate instances that are discarded from *U*.  is the computation time for rough lower approximation, here is the number of instances present in decision class i.e. and *K* is the number of decision class in *U*.

1. Complexity for sensitive rough set

The computation time for sensitive rough set is expressed as:



Whereare the total instances in the decision class in rough set; i.e. and are the total instances in decision class in borderline set: i.e. .

1. Complexity for tolerance-based rough set

The computation complexity for tolerance-based rough set is expressed as:



Whereis the time computation to obtain tolerance value and  is the total instances in the decision class in roughset.

Table 7 shows the comparison of CPU time for all the clustering algorithms. It is seen from this table GSOM is superior to others in terms of CPU time.

Table 7: Comparison of clustering algorithm with CPU time

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data | Methods | CPU Time in seconds | Dataset name | Method | CPU Time in seconds |
| Ballons | SOM | 10.95 | Balance scale | SOM | 2652.99 |
| RSOM | **1.19** | RSOM | 301.7 |
| GSOM | 1.59 | GSOM | 7.509 |
| RT-GSOM | 5.56 | RT-GSOM | 442.44 |
| Breast cancer | SOM | 641.48 | Connec | SOM | 2933.67 |
| RSOM | 69.32 | RSOM | 142.63 |
| GSOM | **6.36** | GSOM | 7.506 |
| RT-GSOM | 113.12 | RT-GSOM | 461.7 |
| Monk | SOM | 2116.2 | Hay | SOM |  |
| RSOM | 25.97 | RSOM |  |
| GSOM | 7.48 | GSOM |  |
| RT-GSOM | 567.2 | RT-GSOM | 17.73 |
| Voting records | SOM | 793.089 | Nursery | SOM | 3013.48 |
| RSOM | 15.32 | RSOM | 135.47 |
| GSOM | 13.58 | GSOM | 13.06 |
| RT-GSOM | 161.38 | RT-GSOM | 101.2 |
| Tumar | SOM | 1659.41 | Thiroid | SOM |  |
| RSOM | 112.25 | RSOM |  |
| GSOM | 8.89 | GSOM |  |
| RT-GSOM | 337.9 | RT-GSOM |  |
| Iris | SOM |  | Image | SOM |  |
| RSOM |  | RSOM |  |
| GSOM |  | GSOM |  |
| RT-GSOM |  | RT-GSOM |  |
| Seeds | SOM |  | Ecoli | SOM |  |
| RSOM |  | RSOM |  |
| GSOM |  | GSOM |  |
| RT-GSOM |  | RT-GSOM |  |

**4.4.3. Statistical analysis of the proposed algorithm compared to the state-of-the-art clustering approach**

In statistical analysis, two test procedures are performed i.e. parametric and non-parametric. In this study, we choose non-parametric procedure as the tested data sets do not satisfy the normality condition. Moreover, some instances in datasets are definite outliers or not measured appropriately due to uncertainty in datasets. Therefore, we perform statistical tests like Kolmogorov-Smirnov test, D’ Agostino-Pearon test and Shapiro-Wilk test to check if the data are normally distributed or not.

………………………………………………………………………………….

# 5. Conclusions

A new RT-GSOM is developed by integrating rough lower approximation set and tolerance-based rough set with GSOM to capture the uncertainty and underlying decision classes in the data. Rough lower approximation is done based on Pawlak’s Rough Set Theory, whereas tolerance-based rough set is obtained by using similarity relation within overlapping equivalence relation, which deal with the vagueness in the decision class information. The new RT in GSOM solves two problems i) uncertainty in data and ii) prior specification of an output space lattice to achieve an optimal preservation of knowledge-based neurons in the state-of-the-art clustering approaches.

The superiority of RT-GSOM as compared with GSOM, RSOM and SOM is demonstrated in twelve datasets in terms of Silhouette Index, DB index, Quantization error, and CPU time. Intuitively, for handling data with overlapping patterns, RT-GSOM performs better than state-of-the-art. GSOM and RSOM perform better than conventional SOM. Moreover, the RT in RT-GSOM preserve the relative information among structures using rough lower approximation, tolerance similarity and then transfer it to decision classes, which is missing in other RSOM and tolerance rough set. The instances selected in RT-GSOM are more relevant than those of the other unsupervised state-of-the-art in terms of feature evaluation index and clustering performance metric in most of the cases. The proposed RT-GSOM provides statistically more significant results than other state-of-the-art clustering approaches.

**Future Scope:**

The main weakness of GSOM is, here winning node can spread in only four directions in its neighborhood in competitive layer. Therefore, some information about closest node of the winning node can be neglected. This problem may be overcome by considering all eight neighborhood direction of any winning node to move. Thus all the information related closest node is preserved here.

………………………………………..

# References

* + - 1. Clustering of time series data—a survey by T. Warren Liao, 2005
      2. Clustering of the self-organizing map (2000)
      3. GROWING HIERARCHICAL SELF ORGANISINGMAP (GHSOM) TOOLBOX:VISUALISATIONS AND ENHANCEMENTS
      4. Rough Self Organizing Map
      5. I-NICE
      6. Tolerance rough fuzzy decision tree
      7. Integration of growing self-organizing map and continuous genetic algorithm for grading lithium-ion battery cells
      8. A comparison between the Silhouette index and the davies-bouldin index in the IDS clusters
      9. A granular self-organizing map for clustering and gene selection in microarray data
      10. Z. Pawlak, Rough sets. International Journal of Computer and Information Sciences, 11, 341-356, 1982.