

Concentration of Outside Elements in High Dimensional data using Provokable Topographic Mapping

S. Gayathri¹, Dr. M. Mary Metilda², Sanjai babu srinivasan³

¹ Research scholar, Bharathiar University, Coimbatore,

²Asst. Prof. Queen Marys College, Chennai,

³ Research scholar, Bharathiar University, Coimbatore,

ABSTRACT

Feature selection involves identifying a subset of the most useful features that produces compatible results as the original entire set of features. A feature selection algorithm may be evaluated from both the efficiency and effectiveness points of view. While the efficiency concerns the time required to find a subset of features, the effectiveness is related to the quality of the subset of features. The Provokable Topographic Mapping has proven to be one of the most powerful algorithm. The Provokable Topographic Mapping maps the high-dimensional input vectors onto a two-dimensional grid of prototype vectors and orders them. In this paper, we propose an algorithm, called Local Adaptive Receptive Field Dimension Selective Provokable Topographic Mapping of outsider in high dimensional data (LARFSPTMOH), to cluster the data points that are in a high-dimensional subspaces and also cluster the outliers that are available in the high dimensional data. The proposed mapping scheme enhances the system efficiency by providing better quality of clustering when compared to its conventional counterpart. Finally, we have implemented the capability of the proposed algorithm through experiments on unnatural data set as well as the natural data set.

1. INTRODUCTION

Cluster analysis seeks to discover groups, or clusters, of similar objects. The objects are usually represented as a vector of measurements, or a point in multidimensional space. The similarity between objects is often determined using distance measures over the various dimensions in the dataset [44; 45]. Technology advances have made data collection easier and faster, resulting in larger, more complex datasets with many objects and dimensions. As the datasets become larger and more varied, adaptations to existing algorithms are required to maintain cluster quality and speed. Traditional clustering algorithms consider all of the dimensions of an input dataset in an attempt to learn as much as possible about each object described. In high dimensional data, however, many of the dimensions are often irrelevant. These irrelevant dimensions can confuse clustering algorithms by hiding clusters in noisy data. In very high dimensions it is common for all of the objects in a dataset to be nearly equidistant from each other, completely masking the clusters. Feature selection methods have been employed somewhat successfully to improve cluster quality. These algorithms find a subset of dimensions on which to perform clustering by removing irrelevant and redundant dimensions. Unlike feature selection methods which examine the dataset as a whole, subspace clustering algorithms localize their search and are able to uncover clusters that exist in multiple, possibly overlapping subspaces.

Another reason that many clustering algorithms struggle with high dimensional data is the curse of dimensionality. As the number of dimensions in a dataset increases, distance measures become increasingly meaningless. Additional dimensions spread out the points until, in very high dimensions, they are almost equidistant from each other. Figure 1 illustrates how additional dimensions spread out the points in a sample dataset. The dataset consists of 20 points randomly placed between 0 and 2 in each of three dimensions. Figure 1(a) shows the data projected onto one axis. The points are close together with about half of them in a one unit sized bin. Figure 1(b) shows the same data stretched into the second dimension. By adding another dimension we spread the points out along another axis, pulling them further apart. Now only about a quarter of the points fall into a unit sized bin. In Figure 1(c) we add a third dimension which spreads the data further apart. A one unit sized bin now holds only about one eighth of the points. If we continue to add dimensions, the points will continue to

spread out until they are all almost equally far apart and distance is no longer very meaningful. The problem is exacerbated when objects are related in different ways in different subsets of dimensions. It is this type of relationship that subspace clustering algorithms seek to uncover. In order to find such clusters, the irrelevant features must be removed to allow the clustering algorithm to focus on only the relevant dimensions. Clusters found in lower dimensional space also tend to be more easily interpretable, allowing the user to better direct further study. Methods are needed that can uncover clusters formed in various subspaces of massive, high dimensional datasets and represent them in easily interpretable and meaningful ways [48; 66].

This scenario is becoming more common as we strive to examine data from various perspectives. One example of this occurs when clustering query results. A query for the term “Bush” could return documents on the president of the United States as well as information on landscaping. If the documents are clustered using the words as attributes, the two groups of documents would likely be related on different sets of attributes. Another example is found in bioinformatics with DNA microarray data. One population of cells in a microarray experiment may be similar to another because they both produce chlorophyll, and thus be clustered together based on the expression levels of a certain set of genes related to chlorophyll. However, another population might be similar because the cells are regulated by the circadian clock mechanisms of the organism. In this case, they would be clustered on a different set of genes. These two relationships represent clusters in two distinct subsets of genes. These datasets present new challenges and goals for unsupervised learning. Subspace clustering algorithms are one answer to those challenges. They excel in situations like those described above, where objects are related in multiple, different ways. Cluster analysis, at present, has been used to combine the related data. It means data is separated into groups of the same objects. It is done through the dataset, which includes data of high, medium or low dimension. Clustering is used in various fields viz., statistics, pattern recognition and machine learning.

The dataset is a collection of relative sets of information composed of individual elements. There are lots of packages for doing cluster in various dataset [6]. Various algorithms are implemented to provide efficient clustering. But clustering of high dimensional data is not possible in the traditional algorithm. High-Dimensional data are pervasive in many areas of machine learning, signal and image. To overcome this problem, a novel system is proposed to implement the concept to cluster the outsiders in the high dimensional data. This paper, a local adaptive receptive field dimensional selective Provokable Topographic Mapping of Outsider in High dimensional data (LARFDSPTMOH), aims to cluster the outside elements in the high dimensional data. The proposed algorithm deals in four phases viz., self -organizing map in high dimensional data, combination phase, clustering of high dimensional data and outsider in the high dimensions. Special requirements includes clustering techniques and efficient implementation of high dimensionality. An object naturally has many elements and the domain for every elements can be major.

It is not delightful to look for clusters in such a high dimensional space as the average density of points anywhere in the data space is likely to be quite low. Compounding this problem, many dimensions or combinations of dimensions can have noise or values that are uniformly distributed. Therefore, distance functions that utilize all the elements of the data may be inefficient. Many clusters may occur in various subspaces decomposed of different combinations of elements. Data mining applications naturally need cluster implementations that can be easily maintained by an end-user as insight and explanations are of complex importance. It aims specifically to have simple implementations because most visualization techniques do not work well in high dimensional spaces. The clustering methods should be high and scale with the number of dimensions and the size of process. It should be important to the order in which the data records are demonstrated. Current clustering techniques do not address all these points of clustering measures in the high dimensional data. This paper demonstrate the proposed methods that can be implemented along with the existing approaches.

1. Related Work

The existing Dimension Selective Self-Organizing Maps (DSSOM) with Time-Varying Structure for Subspace and Projected Clustering [2], only concentrates on the low dimensional data. It provides less efficiency and a low cluster quality. LARFDSPTMOM abbreviates Local Adaptive Receptive Field Dimension Selective Self-Organizing Map and it is a self-Organizing Map with a time-varying design depends on Dimension Selective elements. This algorithm provide phases to determine the existing work, such as the arrangement of LARFDSPTMOM, convergence of LARFDSPTMOM and in clusters of LARFDSPTMOM. Here, they did not concentrate on the high dimensional data as well as the outside elements in the high dimensional data. This paper mainly focus on clustering of outsiders in the high dimensional data using self-organizing map. In this proposed system, we have implemented a algorithm to Cluster the high dimensional data and outside elements with Self Organizing Mapping. The proposed clustering technique has been determined by various steps. It includes establishment of LARFDSPTMOH, conjunction in LARFDSPTMOH, clustering in LARFDSPTMOH and finally outsider in High Dimensional data. The proposed clustering scheme when tested, resulted in enhanced cluster quality, system efficiency and thereby enhanced effectiveness of the system.

2. FEATURE SUBSET SELECTION ALGORITHM

2.1 Framework and Definitions

Irrelevant features, along with redundant features, severely affect the accuracy of the learning machines [14]. Thus, feature subset selection should be able to identify and remove as much of the irrelevant and redundant information as possible. Moreover, “good feature subsets contain features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other.” [13] Keeping these in mind, we develop a novel algorithm which can efficiently and effectively deal with both irrelevant and redundant features, and obtain a good feature subset. We achieve this through a new feature selection framework (shown in Fig. 1) which composed of the two connected components of irrelevant feature removal and redundant feature elimination. The former obtains features relevant to the target concept by eliminating irrelevant ones, and the latter removes redundant features from relevant ones via choosing representatives from different feature clusters, and thus produces the final subset. The irrelevant feature removal is straightforward once the right relevance measure is defined or selected, while the redundant feature elimination is a bit of sophisticated. In our proposed FAST algorithm, it involves 1) the construction of the minimum spanning tree from a weighted complete graph; 2) the partitioning of the MST into a forest with each tree representing a cluster; and 3) the selection of representative features from the clusters. In order to more precisely introduce the algorithm, and because our proposed feature subset selection framework involves irrelevant feature removal and redundant feature elimination, we first present the traditional definitions of relevant and redundant features, then provide our definitions based on variable correlation as follows.

Subspace Clustering of High Dimensional data

Subspace clustering [5] [1] is the advance version of traditional clustering mechanism that attempts to find clusters in various subspaces. Frequently in high dimensional data, many elements are non relative, that makes the existing system to be more noisy. This subspace clustering provides two major concepts viz., top down approach and bottom up approach. These two approaches are mainly designed for subspace clustering of high dimensional data, which defines clustering of data from a few dozen to many thousands of data. Subspace clustering has been examined additionally since traditional clustering algorithms were often unsuccessful in finding proper clusters in high-dimensional data.

Local Adaptive Receptive Field Dimensional Selective Provokable Topographic Mapping of Outsiders In High Dimensional Data (LARFDSPTMOH)

LARFDSPTMOH stands for Local Adaptive Receptive Field Dimensional Selective Provokable Topographic Mapping of outsiders in high dimensional data. The existing schemes have implemented the time varying structure of LARFDSPTMOH which focused on to enhance the quality of clustering properties and computational cost [2]. But the proposed LARFDSPTMOH focuses on the outside elements in the high dimensional data in which clustering of the data happens from a few dozen to millions. Here it also concentrates on the outside elements when doing cluster. As mentioned earlier, the proposed LARFDSPTMOH mainly focus on the four processes, which provides the following:

- Establishment of LARFDSPTMOH
- Conjunction of LARFDSPTMOH
- Clustering of LARFDSPTMOH
- Outsiders in High dimensional data.

In Establishment of LARFDSPTMOH, the nodes are formed into a cluster from arbitrarily chosen input. The winner node is always active, and if it is below the threshold value, then the new node will be inserted. The nearest node is formed by similar types of subset. The establishment and competition can be repeated by a limited number of times, if the node does not win then it is removed. In conjunction phase, all the nodes are inserted and deleted when it needed. It is similar to that of the establishment phase. Here, in case there is no insertion, then the node will start decreasing. The clustering mechanism will start after finishing conjunction phase. In this clustering phase, it groups the outside elements in the high dimensional data. Finally, LARFDSPTMOH consists of outsider(Outside elements) phase.

Algorithm

Algorithm 1: Establishment in high dimensional data

Input: $a(r)$, $e(a, e(x))$, β, q , N majority $N(maj)$, Major Component maj comp, $m(p)$, t majority $t(maj)$
begin the map with one node with $C(j)$ initialized at the first input pattern, $R(j) \leftarrow 0$, $R(k) \leftarrow 1$ and $win(sn) \leftarrow 0$;
begin the variable n wins $\leftarrow 1$;

for $t \leftarrow 0$ to t (maj) do

provide a arbitrarily chosen input pattern X to the map;
 calculate the activation of all nodes (2);
 find the winner s with large activation (a_s) (equ1);
if $a(s) < a(r)$ and $N < N$ maj then
 create new node j and set: $w(j) \leftarrow X$, $R(j) \leftarrow 0$ and
 $\text{win } S(n) \leftarrow l(p) \times n \text{ win } (s)$;
 connect j to the other nodes as per equ [7]
else
 update the distance vector $s(s)$ of the winner node and its nearest (5);
 update the relevance vector $w(s)$ of the winner node and its neighbors (6);
 update the weight vector $C(s)$ of the winner and of its nearest (4);
 Set $\text{win } S(s) \leftarrow \text{win } S(s) + 1$;
If $n\text{win}(s) = \text{maj}(\text{comp})$ **then**
 delete nodes with $\text{win } s(n) < l(p) \times \text{maj}(\text{comp})$;
 update the connections of the remaining nodes as per (7)
 reset the number of wins of the remaining nodes;
 $\text{wins}(n) \leftarrow 0$;
 $n\text{win}(s) \leftarrow 0$;
end
 $n \text{ wins} \leftarrow n \text{ wins} + 1$;

end

proceed conjunction algorithm (alg 2)

Algorithm 2: Conjunction in high dimensional data

while true do

N majority N maj;

N maj $\leftarrow N$

delete nodes with $\text{win } s(n) < l(p) \times \text{maj} \text{ comp}$;

if ($N = N$ maj) then give back

Update connections of all nodes;

$\text{win } s(n) \leftarrow 0$;

for $t \leftarrow 0$ to t maj do

Provide a arbitrarily chosen input pattern X to the map;

Compute the activation of all nodes (2);

find the winners with the highest activation (1);

update the distance vector $s(s)$ of winner node and of its nearest (5);

update the relevance vectors $R(j)$ of the winner node and of its nearest (6);

update the weight vector $C(s)$ of the winner and of its nearest (4);

set $\text{win } s(n) \leftarrow \text{win } s(n) + 1$;

end

end

Algorithm 3: clustering with LARFDSPTM in High Dimensional data

Input: High Dimensional data HD (data)

Output: clustering of HD (data)

initiate

(1) for each input pattern y in the high dimensional dataset [alg4] do

(2) Provide y to the map in HD (data)

(3) Compute activation of all nodes in HD (data) [2]

(4) if $a(c) \geq a(d)$ then

(5) repeat

assign y to the cluster with winner node s ;

if estimate cluster then

break;

```

find the nearest node in the map with majority activation a(c) irrelative to the past winners
until |a(c) < a(d)
else
assign y to the outsider in HD (data) [Algorithm 4]
find the clustering of outsider in HD (data) [Algorithm 4]
end

```

end

Algorithm 4: outsider in High Dimensional data

Input: Number of nearness determined, High Dimensional data HD (data), P, M, number of outsiders, Major distance Major (dist), Minority min

Output: outsider is HD data

Assume: Coterminous (o, R, P) give back P elements in HD (data), Major (dist) (o, R) give back Major (dist) between o, R in HD (data)

initiate

```

(1) C(min) ← 0
(2) HD (data)(0) ← null
(3) For all segmentation s in M do
(4) For all elements e in s do
Nearest ← null
{Finding for nearest in HD (data)}
{high dimensional data}
for n=j , n greater than 0 then increment n do
end for
for all segmentation n in M, beginning by segmentation s do, v in t do, v ≠ 0 do
if |nearest (0) | >= P and Cp (min)
Majority (dist)( 0, nearest(0)) then
break
end if
end for
O= outsider in HD data
end for
end for

```

end

The above mentioned algorithms depict the methods of four phases of LARFDSPTMOH. In Establishment phase, nodes from the cluster are chosen randomly as input. The winner node is always an active mode. As mentioned earlier, if the winner is below the threshold value, then the node is removed. Then the nearest node formed by a similar type of data is chosen. Next in the Conjunction phase the same procedure is followed. In this phase also, if the node does not win, then it is removed. Also in the conjunction phase all the nodes are inserted and deleted when needed. In conjunction, if there are no insertion, then the nodes will start decreasing. Then in the clustering phase all the nodes are clustered in high dimensional data based on the distance factor. Finally, in Outsider phase, all the nodes are cluster as well as the outsider node.

Establishment of LARFDSPTMOH

In this establishment of LARFDSSOMOH, the nodes are from a clutch when the inputs are chosen as random manner. In this phase, the winner node is always in active mode. Here the new node will insert when the active node below the threshold range. The neighbor node is implemented by the same type of subset elements. The enactment and competition can be recurring by a limited amount of time duration. At the end the node will delete when it does not attain the winning target.

Conjunction of LARFDSPTMOH

In this phase, all nodes are inserted and removed depending on the requirement. Here, It is same as that of the establishment phase, which is used to form the clutch, defines the winner node, and implementation of a neighbor node.

The nodes are decreased when there is no insertion of elements. Finally the clustering or clutching mechanism will start after the conjunction phase.

Clustering of Outsider in High Dimensional

In this phase, combines each and every outsider element in the high dimensional data. Here the outsider refers to outside of elements in the high dimensional data or uncover elements in the high dimensional data. In this phase of clutching of data in high dimensional, starts clutch from a few dozen and moves to the millions of data. Aggregate of this phase, it cluster all elements of outsiders in high dimensional data.

3. MATHEMATICAL CALCULATION

In this insertion node, first the highest activation value can be calculated by,

$ac(D(w)(x, c(j)), W(j)) \rightarrow$ input pattern

$R(x) \rightarrow$ highest activation value

$$R(x) \rightarrow \arg \max_j [mn(D(w)(x(i), P(j)) W(j))] \rightarrow (1)$$

The Circular basis function can be determined by,

$$mn(D(w)(x, P(j)), w(j)) = 1 / (1 + M / (\|s\|^2 + \epsilon)) \rightarrow (2)$$

$R \rightarrow$ relative vector

$D(w)(x, P(j)) \rightarrow$ distance of weighted component

$\epsilon \rightarrow$ indicates less value

$$D(w)(x, n_j) = \sqrt{\sum_{i=1}^y w_{ji} (x(i) - n_{ji})^2} \rightarrow (3)$$

Then the activation threshold in LARFPTMHO

$a(t) \rightarrow$ threshold value

here, if the threshold value below then insert new node

To update winner and nearest node by using following,

$$n(j)(z+1) = n_j(z) + l(x - n_j(z)) \rightarrow (4)$$

$l = l(b)$

$l = l(x)$

Updating via the input moving average of the distance between input and current center vector

$n(j) = s(j)$

$$s(j)(z+1) = (1 - l(\beta)) s(j)(z) + l(\beta) (\|z - n(j)(z)\|) \rightarrow (5)$$

Relative vector calculated by an inverse logistic function in high dimensional data,

$$w(ji) = \begin{cases} 1 / (1 + \exp(A - s_{ji}/s(B - C))) & \rightarrow (6) \\ 1 & ; \text{otherwise} \end{cases}$$

here, $A = s(ji)$ mean

$B = s(ji)$ maj

$C = s(ji)$ min

A, B, C are maximum, minimum and mean component of distance vector.

To update a nearest node by the following,

$$i \text{ and } j \text{ nodes are in the high dimensional data} = \begin{cases} \text{connected if } \|a(i) - a(j)\| < n\sqrt{m} \\ \text{disconnected ; otherwise} \end{cases} \rightarrow (7)$$

$m \rightarrow$ input dimensions

To detect outlier data, by the following

$$OHD(\text{maj}) = (a(\text{max}) - a(\text{min})) / (s(n) - a(\text{min}) + n) \{ > ((\frac{2}{n}) + w(i)) \text{ majority}$$

$$\leq ((\frac{2}{n}) + w(i)) \} \rightarrow (8)$$

$$OHD(\text{min}) = a(\text{max}) - a(\text{min}) / a(\text{max}) * n - s(n) \{ \leq ((\frac{2}{n}) + w(i)) > ((\frac{2}{n}) + w(i)); \} \rightarrow (9)$$

From (8) and (9) equation shows the minimum and maximum outliers in the high dimensional data.

4. CASE STUDY OF PROPOSED ALGORITHM

To verify the efficiency of the proposed algorithm, an educational system model has been implemented. The information about students in an university are taken into account. The university data set are consider as a high dimensional data set, as illustrated in Figure 1. The data set consists of type of admission viz., regular, part time student, good character student, bad character, not interested candidate, money oriented problem, quit without any reason and academic percentage. This case study is implemented with the Provokable Topographic Mapping and to cluster the outsiders in the high dimensional data. From these parameters clustering of outside elements in high dimensional data can be achieved.

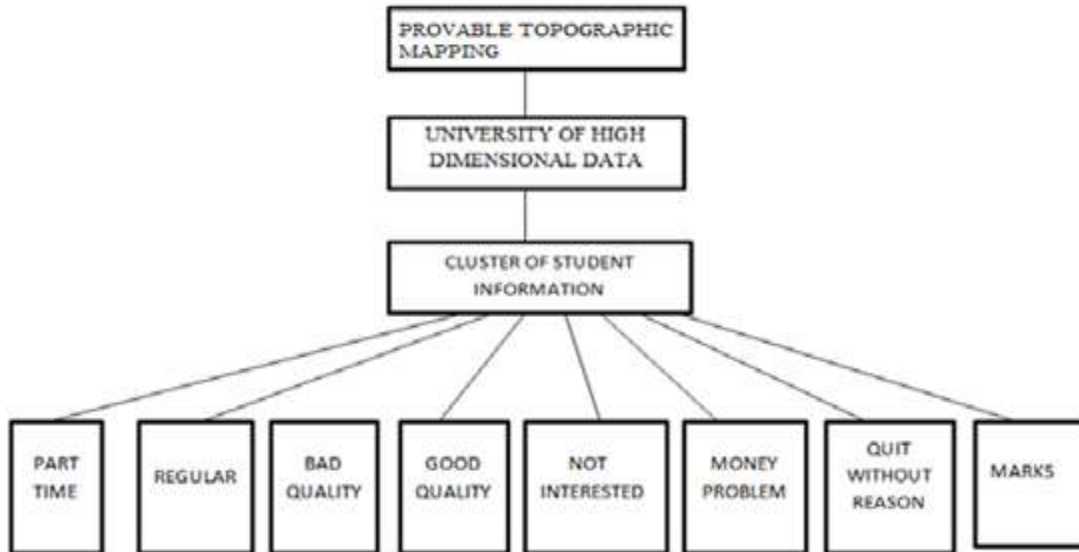
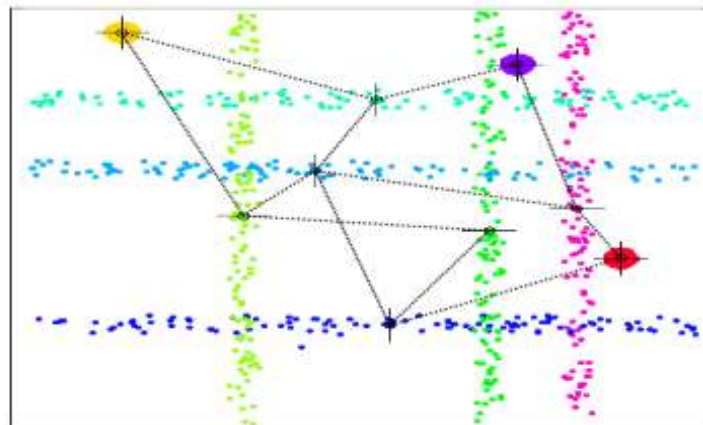
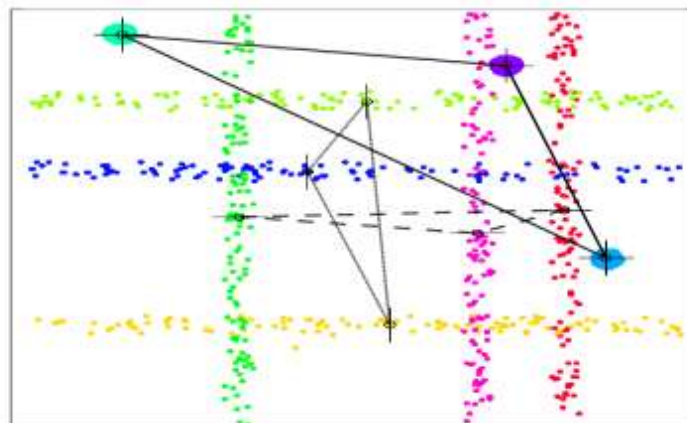


Figure 1. Case Study - Information about Students

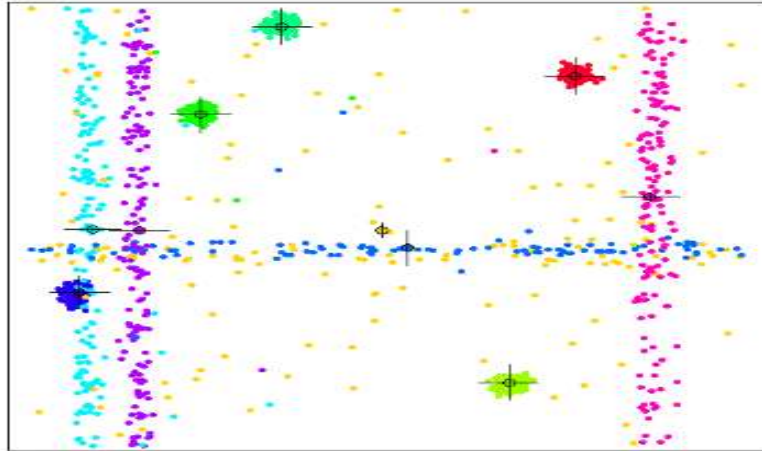
Performance Analysis



(a) Formation of First nodes



(b) Formation of relative and neighbourhood nodes



(c) Formation of long distance clustering

Figure 2 depicts the formation of nodes by DSSSOM method. Figure 2a, reveals the formation of first nodes and Figure 2b, illustrates the formation of relative and neighbor hood nodes. The formation of long distance node are shown in Figure 2c. Once the nodes are formed, then clustering the nodes are done based on the data set. Figure 3 depicts the method of clustering the inside elements.

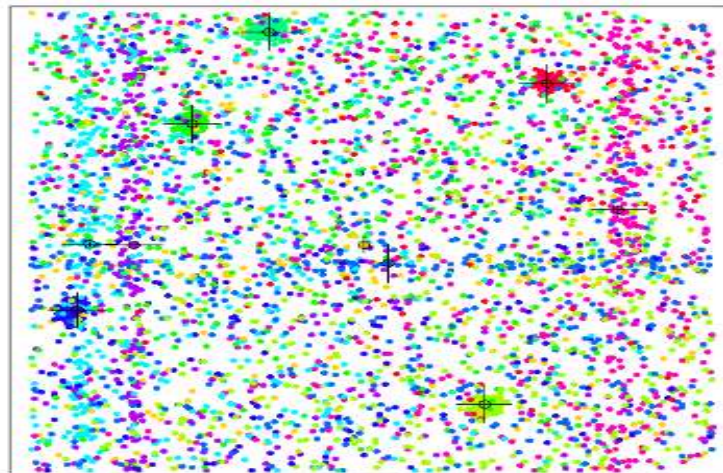
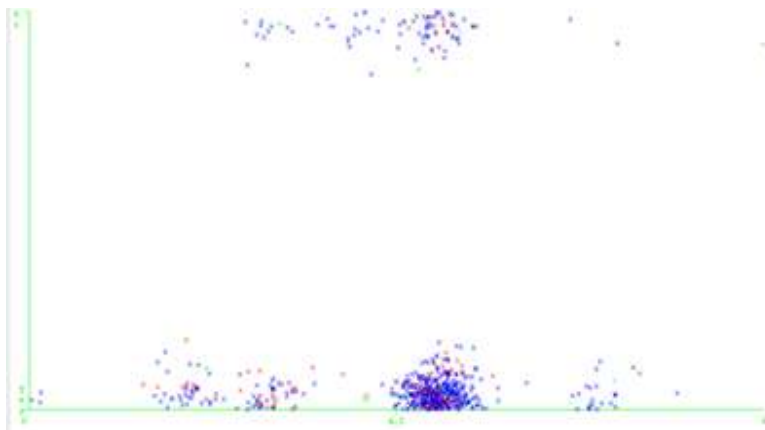
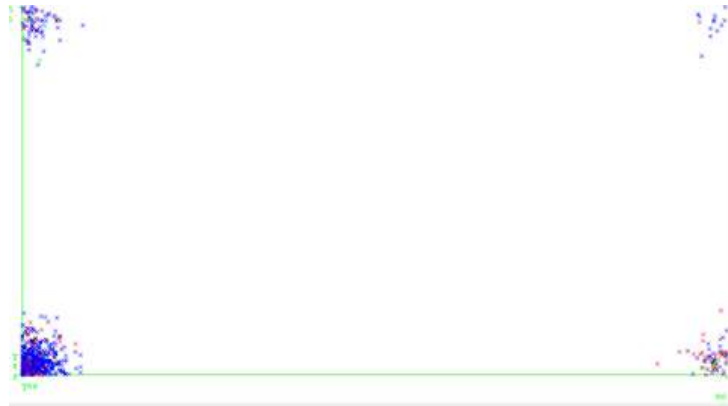


Figure 3. Clustering of inside elements



(a) Part time Vs Regular



(b) Money Problem Vs Student

Figure 4. High Dimensional Clustering

Figure 4 illustrates the analysis of information and how the students cluster are formed. The parameters taken up for study includes part time students Vs regular and the money problem Vs students. The nearest clustering is formed after forming the high dimensional data of the student as regular or part time. Once that cluster is made, then with the studnets, the money problem is formed as a nearest cluster. The implementation of these graphs infers how efficiently the clustering of nodes, clustering of relative and nearest nodes, projection of nodes and clustering of nodes with distance are done with the proposed algorithm.

Table. 1 Parameters of LARFDSSOM

| Number of Nodes | Minimum2 | Maximum 18 |
|------------------------|----------|------------|
| Leaming rate (A) | 0.001 | 0.1 |
| Leaming rate decay(B) | 0.0 | 2 |
| Neighborhood (C) | 0.001 | 0.99 |
| Neighborhood Decay (D) | 0.01 | 2 |
| Revalance rate (E) | 0.001 | 0.1 |
| Min Revalance (F) | 0 | 1 |
| New Winner (G) | 0 | 1 |
| Maximum Winner (H) | 1 | 3 |
| Revalance rate 1 (I) | 0.001 | 0.1 |
| Revalance rate 2(J) | 0.8 | 0.99 |
| Revalance rate 3(K) | 1.01 | 1.2 |

Table. 2 LARFDSPTMOH – Outsider in High Dimensional

| Number of Nodes | Minimum40 | Maximum 200 |
|-----------------------|-----------|-------------|
| Leaming rate (A) | 7.42 | 49.45 |
| Leaming rate decay(B) | 70.05 | 70.05 |
| Nearest (C) | 46.07 | 46.07 |
| Nearest Decay (D) | 70.05 | 70.05 |
| Revalance rate (E) | 46.7 | 46.7 |
| Min Revalance (F) | 9.47 | 9.47 |
| New Winner (G) | 9.62 | 9.62 |
| Maximum Winner (H) | 25.40 | 25.40 |
| Revalance rate 1 (I) | 7.42 | 49.45 |
| Revalance rate 2(J) | 10.75 | 50.07 |
| Revalance rate 3(K) | 20.46 | 55.47 |
| Outlier threshold (L) | 1.46 | 9.75 |

The tables 1 and 2 infer the parameters of the LARFDSSOM and LARFDSPTMH model. The parameters tabulated illustrate that when the number of nodes is maximum in both the LARDFSSPTM and LARFDSPTMH models, the learning rate increases with increase in the number of nodes. Similarly the relevance rate is also increased with increase in the number of nodes. These points infer that the efficiency of the proposed algorithm is much better compared to its conventional counterpart.

CONCLUSION

This paper made an attempt to research on Clustering of Outsiders in High Dimensional data with Self Organizing Mapping with focus on the Provokable Topographic Mapping for high dimensional data as well as the outsiders. The proposed LARFDSPTMOH model has been implemented in a case study. The novel algorithm has been created for clustering outside elements in the high dimensional data. The LARFDSPTMOH algorithm was used to cluster the high dimensional data with high efficiency than the existing methods. The analysis of the same proves that the proposed model provides enhanced effectiveness of clustering. In future we need to increase the algorithm performance by enhancing the mapping considerations.

REFERENCES

- [1]. Adler. A(2015), "Linear-Time Subspace Clustering via Bipartite Graph Modeling," PP: (99).
- [2]. Bassani. F(2014),"Dimension Selective Self-Organizing Maps with Time-Varying Structure for Subspace and Projected Clustering," 26: (3), 458-471.
- [3]. Coelho. M(2011), "Subspace clustering in information retrieval in urban scene databases," pp, 173-180.
- [4]. Gan. G(2006), "PARTCAT: A subspace clustering algorithm for high dimensional categorical data," pp, 4406-4412.
- [5]. Han Hu(2014), "Exploiting Unsupervised and Supervised Constraints for Subspace Clustering," 37: (8), 1542-1557.
- [6]. Mazel. J(2011), "SubSpace clustering, Inter-Clustering Results Association & anomaly correlation of unsupervised network anomaly detection," 1-8.
- [7]. Milano. M "Self Organizing Nets for optimization," 15 :(3), 2004, 758-756.
- [8]. Moise. G(2015), "Finding non-redundant, statistically significant regions in high dimensional data: A novel approach to project and subspace clustering," pp, 533-541.
- [9]. Vidal. R(2011), "Subspace clustering," pp: (28), 52-68.
- [10]. Zhenya Zhang(2008), "Clustering aggregation based on genetic algorithms for documents clustering," 3156-3161.
- [11]. J. Biesiada and W. Duch(2008), "Features Election for High-Dimensional data a Pearson Redundancy Based Filter," Advances in Soft Computing, vol. 45, pp. 242-249.
- [12]. R. Butterworth, G. Piatetsky-Shapiro, and D.A. Simovici(2005), "On Feature Selection through Clustering," Proc. IEEE Fifth Int'l Conf. Data Mining, pp. 581-584.
- [13]. C. Cardie(1993), "Using Decision Trees to Improve Case-Based Learning," Proc. 10th Int'l Conf. Machine Learning, pp. 25-32.
- [14]. P. Chanda, Y. Cho, A. Zhang, and M. Ramanathan(2009), "Mining of Attribute Interactions Using Information Theoretic Metrics," Proc. IEEE Int'l Conf. Data Mining Workshops, pp. 350-355.
- [15]. S. Chikhi and S. Benhammada(2009), "ReliefMSS: A Variation on a Feature Ranking Relief Algorithm," Int'l J. Business Intelligence and Data Mining, vol. 4, nos. 3/4, pp. 375-390.
- [16]. W. Cohen (1995), "Fast Effective Rule Induction," Proc. Machine Learning (ICML '95), pp. 115-123.