A Fuzzy Clustering Algorithm for High Dimensional Streaming Data

Diksha Upadhyay, Susheel Jain, Anurag Jain
Department of Computer Science, RITS, Bhopal, India
Email: {diksha.du31@gmail.com, jain_susheel65@yahoo.co.in, anurag.akjain@gmail.com}

Abstract

In this paper we propose a dimension reduced weighted fuzzy clustering algorithm (sWFCM-HD). The algorithm can be used for high dimensional datasets having streaming behavior. Such datasets can be found in the area of sensor networks, data originated from web click stream and data collected by internet traffic flow etc. These data's have two special properties which separate them from other datasets: a) They have streaming behavior and b) They have higher dimensions. Optimized fuzzy clustering algorithm has already been proposed for datasets having streaming behavior or higher dimensions. But as per our information, nobody has proposed any optimized fuzzy clustering algorithm for data sets having both the properties, i.e., data sets with higher dimension and also continuously arriving streaming behavior. Experimental analysis shows that our proposed algorithm (sWFCM-HD) improves performance in terms of memory consumption as well as execution time

Keywords-K-Means, Fuzzy C-Means, Weighted Fuzzy C-Means, Dimension Reduction, Clustering.

I. INTRODUCTION

In recent years there are various sources , for generating data streams of continuous behavior has Came in to existence , such as data from sensor networks, data generated by web click stream and data stream from internet traffic data transfer, now a days data stream become an important source of data. As a result, many researchers are giving importance on it. Finding efficient data stream mining algorithm has become an important research subject. Data stream [1] is potential infinite, with uncertain arriving speed and can be scanned one pass. The processing of data stream has to implement within a limited space (memory) and a strict time constraint. Due to this, an efficient data stream mining algorithms must satisfy a more strict demand.

The simple comparative analysis of various dimension reduction techniques and various clustering techniques (survey) has been provided in the [20]. Cluster analysis pays a very important role in data mining field. Clustering algorithm based on data stream model has gained an extensive research [1], [2], [3], [4], [5]. Fuzzy C means (FCM) and its improvements [6], [7] as important clustering methods have been abroad used in many aspects such as in the field of data mining, in pattern recognition, in the field of machine learning and so on. In [8] the author proposed a weighted fuzzy c-means (sWFCM) clustering algorithm for datasets having streaming behavior. The various effects and issues of high dimensionality property of data sets on clustering, in solving nearest neighbor problem and on indexing has been observed by various researchers in detail. Due to high dimensions the data becomes sparse; the Conventional (previous) indexing and algorithmic procedures fail from an efficiency and effectiveness perspective.

On high dimensional data it has been observed that, the various parameters such as proximity measures, distance calculation or finding nearest neighbor may not be that much effective and meaningful may not even be qualitatively meaningful. In the Recent research result shows the dimensionality property of data sets from the prospective of distance metrics which will be further used to find the similarity between data objects [9]. Further, high-dimensional data will create various challenging issues for various conventional clustering algorithms which require definite solutions.

In high dimensional data, traditional similarity measures as used in conventional clustering algorithms are usually not meaningful. Common approaches to handle high dimensional data are subspace clustering, projected clustering, pattern based clustering or correlation clustering [10]. Due to the presence of various irrelevant features or of Correlation among subsets of features will heavily impact the generation and visualization of clusters in the full-dimensional space. The major challenge the clustering will face is that the clusters will be formed as per the subspace of features from the total feature space but the feature subspace for various clusters may be different.

The K-Means is the famous clustering algorithm which is simple and widely applicable partitioned clustering technique. The space complexity of the K-Means algorithm is O ((n + k) d) and the time complexity is O (nKtd) where n is the number of data, K is the possible number of clusters, d is denoting the dimension of the data and t is the number of iterations. In [11] the authors proposed a technique to convert high dimensional data into two dimensional data and then simple K Means algorithm has been applied on the transformed dataset. The intention of this modified algorithm is to reduce the dimension of the data to increase the efficiency of the K-Means clustering algorithm.

In this paper we propose a dimension reduced weighted fuzzy c-means algorithm (sWFCM-HD). The algorithm will be applicable for those high dimensional data sets that have streaming behavior. An example of such data sets is live high-definition videos in internet. These datas have two special properties which separate them from other data sets: a) They have streaming behavior and b) They have higher dimensions. As we discussed above optimized K-means algorithm has already been proposed for data sets having streaming behavior as well as data sets having higher dimension. But as per our information, nobody has proposed any optimized K-means algorithm for data sets having both the properties, i.e., data sets with higher dimension and also continuously arriving streaming behavior. So, our work will be a combination of the work done in [11] and [8]. But to the best of our knowledge, all clustering algorithms for data stream commonly belong to hard cluster. Fuzzy clustering algorithm provided in the present is not used directly for data streams.

The rest of the paper is organized as per following. In the next section we discussed related research works. Section III will provide the background details required for this paper. We explained our proposed algorithm in section IV, and then after experimental comparisons and analysis are given in section V.And then after finally we conclude the paper in section VI

II. RELATED WORK

There are substantial amount of clustering data stream algorithms presented. In [2], the STREAM algorithm is proposed to cluster data streams. STREAM first determines the size of sample. If the condition arises where size of data chunk is larger than the sample size of data, then in such case a LOCALSEARCH procedure (algorithm) will be invoked for obtaining the clusters of the data chunks. And then after, the LOCALSEARCH procedure is applied on previous iterations generated all the cluster centers.

The k-means algorithm is extended and the VFKM algorithm is proposed in [3]. It is guaranteed that the Generated model produced will not differ significantly from the one that would be obtained with infinite data. A variant of the k-means algorithm, incremental k-means, is proposed to obtain high quality solutions. In [4] the authors have proposed a system (time series clustering technique) which will create the hierarchy of clusters on the incremental basis .The correlation between time series is used as similarity measure. Cluster decomposition or composition (aggregation) will be performed at each step.In [5], the CluStream algorithm is proposed to cluster evolving data streams. CluStreams idea is dividing the clustering method in the online component which will afterwards periodically stores complete summary measures (statistics) and an offline component which uses only this summary statistics. Pyramidal time frame parameters in collaboration with a micro-clustering approach is used to deal with the problems of generating efficient choice, providing storage, and use of the present statistical data for a continuous fast data stream.

For the purpose of clustering image data which is larger in size a method has been proposed based on sampling phenomena in [12], where the samples are chosen by the chisquare or hypothesis test as per divergence. In [13], speeding up is obtained by performing the random sampling of the data and then after clustering it. The centroids which will be obtained are then after used for initializing the entire data set. Two well known techniques for dimensional reduction techniques are feature selection and feature extraction; firstly before applying any data mining task, a practical approach to overcome the problems of high dimensional dataset where several features are correlated is to perform feature selection [9]. For feature selection there may be unsupervised (PCA [14], LLE [15], ISOMAP [16]) learning techniques which Will understand the low dimensional space that classify (represents) well the data without need to any specific task. Principal Component Analysis (PCA) can be used to map the original provided data sets in higher dimensions to a lower dimensional data space where the points may better cluster and the resulting clusters may be more meaningful. For nonlinear approaches, Sammons mapping, multidimensional scaling and LTSA [17] are available. Dimensionality reduction techniques which are supervised in nature Discriminative PLVM [18]) try to estimate a low dimensional representation which has sufficient information for predicting the task target values. The above provided these supervised techniques presumes that the latent space and/or the given data are being generated from some restricted distribution phenomena.

Various soft computing tools are also available for feature selection and feature extraction [19]. Next, the decision tree induction can be used for attribute subset selection, a decision tree is constructed from the whole data and the attributes that did not appear in tree are assumed to be less dominant. After analyzing the tree where the attributes do appear are to be selected as important attribute. Unfortunately, such dimensionality reduction techniques cannot be applied to clustering problems because such techniques are global since they generally compute only one composite subspace of the provided original data space in which the clustering can then be performed, considering complete set of points. In contrast, the problem of local feature relevance and local feature correlation classify that many subspaces are needed because each cluster may exist in a different subspace [10]. In [11] dimension reduction technique has been proposed Which will first convert the high dimensional data sets in to two dimensional data and then for increasing the clustering efficiency K-Means clustering algorithm have been applied on the resultant data (two dimensional data) .

The difference of the above proposals with our proposal is none of them tried to handle higher dimensional dataset with streaming behavior. Both high dimensional dataset and streaming datasets has been widely studied before but as per our knowledge no one has tried yet to propose any clustering technique dedicated for datasets having higher dimension as well as streaming behavior.

III. BACKGROUND

A. FCM algorithm

Consider a data set $X = \{x_1, x_2, x_3,,,,,,, x_n\}$, the FCM algorithm partitions X into c fuzzy clusters and find out each clusters center so that the cost function (objective function) of dissimilarity measure is minimization or below a certain threshold. FCM analyze membership value of each data in each cluster, it is presented as follows:

Objective function:

$$J_m(U,v) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m (d_{ik})^2 \qquad (1)$$

U and v can be calculated as:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{(m-1)}}} \qquad (2)$$

$$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m} \qquad (3)$$

Where u_{ik} is the membership value of the k^{th} data x_k in the i^{th} cluster. $d_{ik} = \| x_k - v_i \|$ is the Euclidean distance between data x_k and the cluster centroid v_i ,$1 \le i \le c$, $1 \le k \le n$, exponent $m > 1$.

The FCM algorithm determines the cluster centroid v_i and the membership matrix U through iterations using the following steps:

1. Initialize the membership matrix U , u_{ik} randomly comes from (0, 1) and satisfy:

   $$\sum_{i=1}^{c} (u_{ik}) = 1, 1 \le k \le n$$

2. Calculate c fuzzy clusters v_i i=1,……… c using Equation 3.
3. Compute the objective function according to Equation 1. Stop if objective function of dissimilarity measure is minimization or concentrate on a particular value or if its improvement results over previous iteration outcomes is below a certain threshold or iterations reach a certain tolerance value.
4. Compute a new U using Equation 2. Go to step 2.

As FCM is clustering on the total data set, and data stream may contain a very large data set, so let FCM deal with data stream directly may consume significant amounts of CPU time to converge, or result in an intolerable iteration quantity. Based on this situation, [8] proposed one alternative called weighted FCM algorithm (swFCM) for data stream as discussed in the next section.

B. Weighted FCM (swFCM)

First, divide data stream into chunks X_1, $X_{2,…………}$ X_s according to the reaching time of data, and the size of each chunk is determined by main memory of the processing system, let n_1, n_2,………n_s be the data numbers of chunks

X_1, X_2,………… X_s respectively. Due to its stream setting, a time weight w (t) is imposed on each data representing the datum influence extent on the clustering process, and

$$\int_{t_0}^{t_c} w\,(\,t\,)\,dt \;=\; 1$$

Where t_0 is the initial time of stream and t_c is the current time.

The main idea of sWFCM is renewing the weighted clustering centers by iterations till the cost function gets a satisfying result or the number of iteration is to a tolerance. Moreover, during the processing, we give the singleton a constant weight as 1. The procedure is presented as follow:

1) Import the chunk X_l ($1 \le l \le s$).

2) Update the weight of cluster centroids.

- If l= 1: Apply FCM to gain cluster centroids v_i i=1,…… c ,and compute:

$$w_i^{'} = \sum_{j=1}^{n_1} (u_{ij}) w_j \qquad 1 \le i \le c$$

Where $w_j = 1, \forall 1 \le j \le n_1$

- If ll> 1:

$$w_i^{'} = \sum_{j=1}^{n_l + c} (u_{ij}) w_j \qquad 1 \le i \le c$$

Where $w_j = 1, \forall c + 1 \le j \le n_l + c$

The centroid weight w_i then updates as $w_i = w_i^{'}$

3) Update cluster centroids:

$$v_i = \frac{\sum_{k=1}^{n_l + c} w_k (u_{ik})^m x_k}{\sum_{k=1}^{n_l + c} w_k (u_{ik})^m}$$

Where $x_k \in \{ v_i \mid 1 \le i \le c \} \,\mathrm{U}\, X_l$

4) Compute objective function:

$$J_m(U, v) = \sum_{k=1} c + n_l \sum_{i=1}^{c} w_k (u_{ik})^m (d_{ik})^2$$

Stop if objective function is minimization or concentrate on a certain value, or its improvement
Over previous results obtained from iterations is below a certain threshold, or iterations reach a
Certain tolerance value.

5) Compute a new U using Equation 2. Go to step 2.

6) If $l = s$ then stop, else go to step 1.

C. Converting high dimensional dataset into two dimensional data set

We used the technique proposed in [11] for reducing dimension of higher dimensional datasets. In this technique each high dimensional data in the dataset is converted to a two dimensional co-ordinate point. So the clustering algorithm can take the converted two dimensional dataset as input instead of higher dimensional dataset. The working of the dimension reduction technique [11] is explained below: Let O = $o_{i,} o_2, \ldots, o_n$ be a d-dimensional dataset. Now to convert each *d*-dimensional data o_i Є *O* two dimensional coordinate point (*Xi, Yi)* do the following:

Calculate X_i and Y_i as

$$X_i = \frac{x_{i0} + x_{i1} + ...x_{id-1}}{d}$$

And

$$Y_i = \frac{y_{i0} + y_{i1} + ...y_{id-1}}{d}$$

For each j^{th} dimensional value of i^{th} data in *O* (i.e., o_{ij}), we can get a co-ordinate point (x_{ij}, y_{ij}). Where $x_{ij} = r_{ij} \cos \theta_j$ and. $y_{ij} = r_{ij} \sin \theta_j$ r_{ij} means the value of o_{ij} (value in j^{th} dimension of i^{th} data). $\theta_j = \theta_{j-1} + 360 / d$, and $\theta_0 = 0^0$. In other words for each data o_i Є *O*, $1 \le i \le n$ there must be d numbers of coordinate points (x_{ij}, y_{ij}) ,$1 \le i \le n$, $1 \le j \le d$ and with help of these coordinate point (x_{ij}, y_{ij}) we can get the mean value (X_i, Y_i) . Plot all the n numbers of mean points on the two dimensional plane and then apply clustering algorithm on the plotted mean points.

IV. OUR PROPOSED TECHNIQUE (SWFCM-HD)

The disadvantages of using higher dimensional datasets in clustering algorithms are already explained in section I. A dimension reduction technique is proposed in [11] to overcome such difficulties. But if the dataset has streaming behavior then even after converting it into a lower dimensional dataset the problem still remains [8], [11]. In section I, we explained the disadvantages of applying FCM algorithm on a dataset having streaming behavior. We combine both dimension reduction and sWFCM technique together to propose a better fuzzy clustering algorithm for large size high dimensional stream datasets. We call our propose algorithm as sWFCM-HD as we used sWFCM and a dimension reduction technique for higher dimensional streaming datasets. Our algorithm is discussed as follows:

Algorithm: sWFCM-HD

Input: High dimensional (*d*-dimensional) large dataset *O* with having streaming behavior.

1) Convert the *d*-dimensional dataset *O* into two dimensional dataset *X* using the dimension reduction technique discussed in section III-C.

2) Apply sWFCM algorithm on the converted two dimensional dataset *X*. The sWFCM algorithm is discussed in section III-B.

Note that, since the dataset *O* has streaming behavior it is not possible to reduce the dimension of the entire dataset at a time. But that doesn't create any problem because sWFCM algorithm uses a chunk of data from dataset at a time. We can see from section III-B that before applying sWFCM, we need to divide the dataset into number of data chunks. Main reason for this is because in real scenario these data are streaming in nature and will not be loaded into main memory all together. Hence, the dimension reduction technique has been applied on chunk basis and not all together

V. EXPERIMENTAL ANALYSIS

We take higher dimensional dataset as input and converted them into two dimensional data set as discussed in section III-C. After reducing the dimension of the dataset we run SWFCM on it. Though sWFCM already exists we used it here for clustering higher dimensional data after reducing their dimension. Experiment shows that sWFCM performs better than FCM for higher dimensional dataset having streaming behavior. Our main intention here is to show that if we combine the techniques proposed in [11] and [8] together for a clustering algorithm then performance will get enhanced much in comparison to the performance of any individual one. Note that, our proposed algorithm (sWFCM-HD) is a combination of the techniques proposed in [11] and [8] (see section IV). We use FCM algorithm on the reduced (2D) dataset as baseline algorithm. For the experiments we use three higher dimensional large size dataset: KDDCUP 1999, Nursery and Letter recognition. All three datasets are available in http://archive.ics.uci.edu/ml/datasets.html. Since KDDCUP 1999 is a very large dataset we used the first 5000 data from it.

A. Cluster Validity

We adopt validity functions [8] to compare cluster efficiency. The validity functions are based on partition coefficient and partition entropy of *U*.

Partition coefficient for FCM

$$V_{pc}(U) = \frac{1}{n}\left(\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij}^{2}\right)$$

Partition coefficient for sWFCM

$$V_{pc}(U) = \frac{1}{n}\left(\sum_{j=1}^{n}\sum_{i=1}^{c} w_i u_{ij}^{2}\right)$$

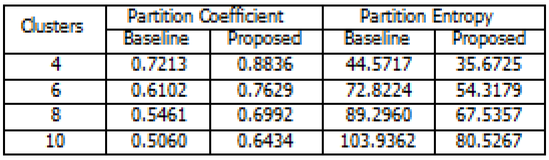
Partition entropy for FCM

$$V_{pe}(U) = -\frac{1}{n}\left(\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij} \log u_{ij}\right)$$

Partition entropy for sWFCM

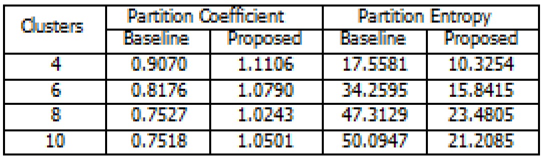
$$V_{pe}(U) = -\frac{1}{n}\left(\sum_{j=1}^{n}\sum_{i=1}^{c} w_i u_{ij} \log u_{ij}\right)$$

Where n is the total number of data in the dataset, w_i , u_{ij}, U are weight of centroids and membership matrix respectively (see section III for details.)



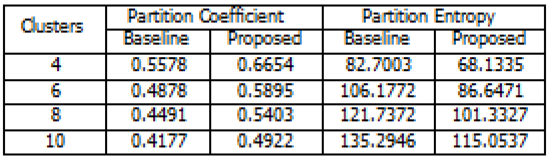
Clusters	Partition Coefficient		Partition Entropy	
	Baseline	Proposed	Baseline	Proposed
4	0.7213	0.8836	44.5717	35.6725
6	0.6102	0.7629	72.8224	54.3179
8	0.5461	0.6992	89.2960	67.5357
10	0.5060	0.6434	103.9362	80.5267

Table I
CLUSTER VALIDITY BASED ON NURSERY DATASET



Clusters	Partition Coefficient		Partition Entropy	
	Baseline	Proposed	Baseline	Proposed
4	0.9070	1.1106	17.5581	10.3254
6	0.8176	1.0790	34.2595	15.8415
8	0.7527	1.0243	47.3129	23.4805
10	0.7518	1.0501	50.0947	21.2085

Table II
CLUSTER VALIDITY BASED ON KDDCUP 1999 DATASET



Clusters	Partition Coefficient		Partition Entropy	
	Baseline	Proposed	Baseline	Proposed
4	0.5578	0.6654	82.7003	68.1335
6	0.4878	0.5895	106.1772	86.6471
8	0.4491	0.5403	121.7372	101.3327
10	0.4177	0.4922	135.2946	115.0537

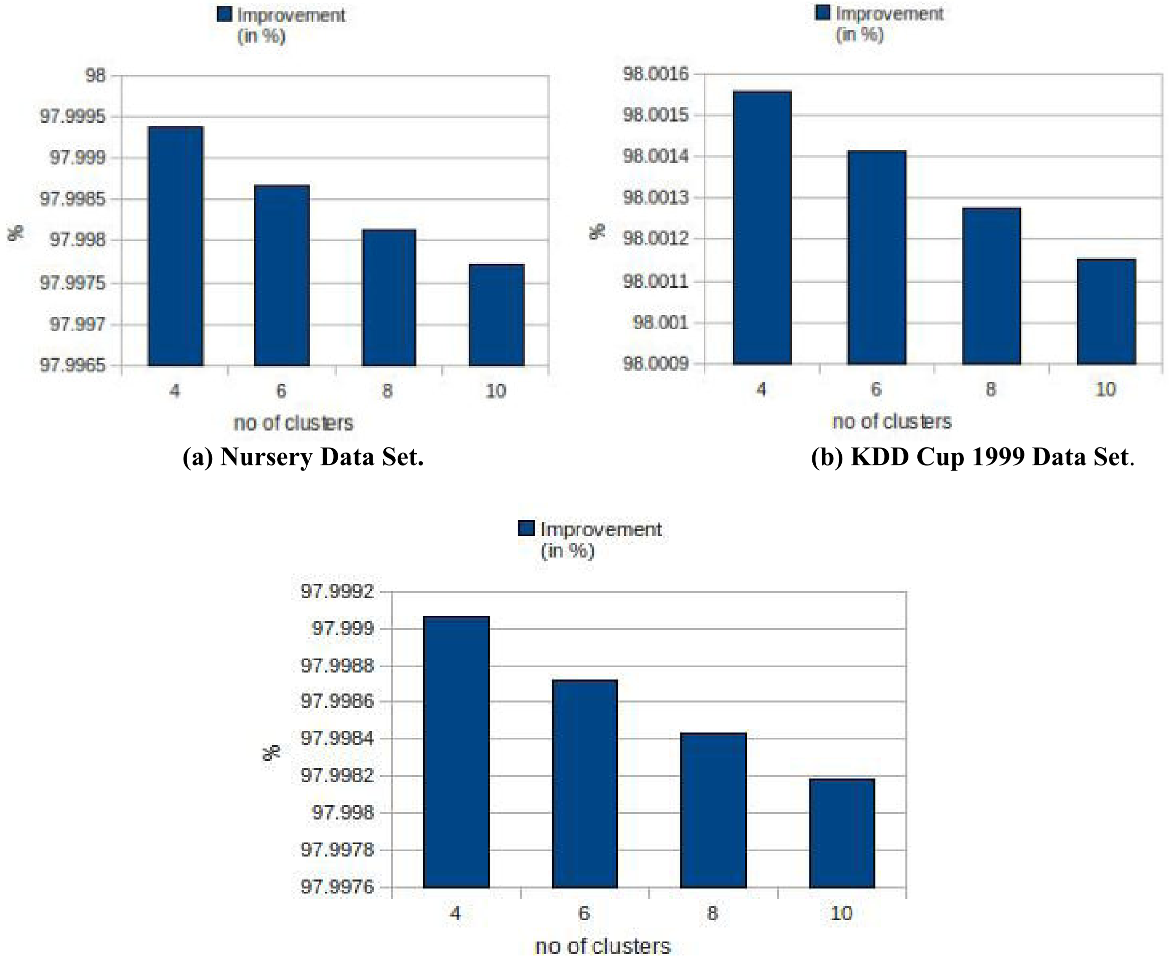
Table III
CLUSTER VALIDITY BASED ON LETTER RECOGNITION DATASET

Table I, II and III shows cluster validity in terms of partition coefficient and partition entropy for the three datasets: nursery, KDDCUP 1999 and letter recognition respectively.

B. Memory Used

Since sWFCM process data as number of chunks we calculated the memory consumption of each chunk separately and take the largest value as the final memory consumption for sWFCM-HD. Since the dataset is streaming in nature, it is not required for sWFCM-HD to access more than one chunk at a time. Figure 1 shows the percentage of improvement in terms of memory consumption by proposed (sWFCM-HD) as compared to the baseline algorithm. The improvement is more than 97% for all three datasets. Baseline

Algorithm (FCM) uses entire dataset at a time and hence it requires enough memory to hold the complete dataset. This is the reason why baseline requires much higher memory than our proposed algorithm.



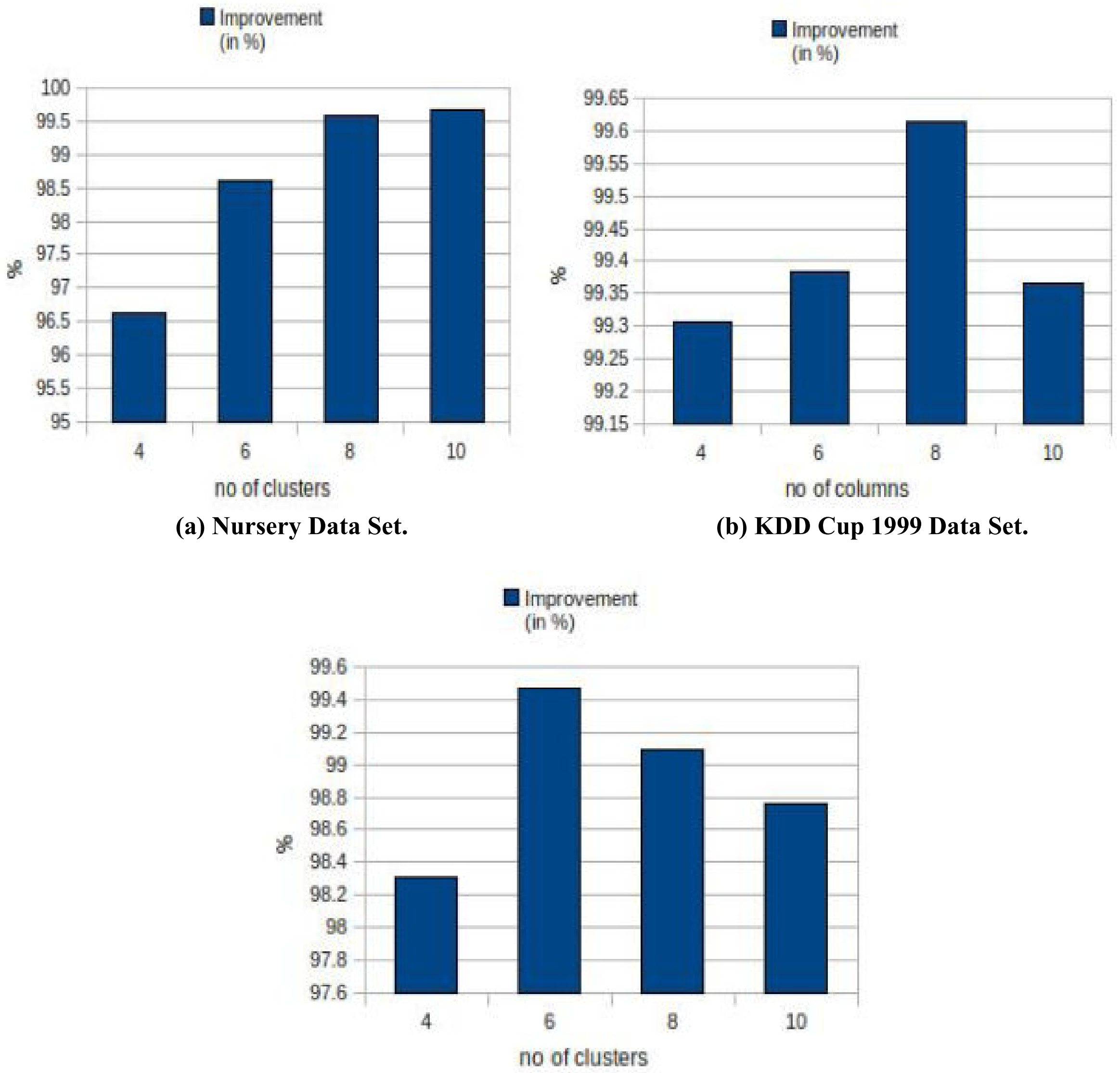
(a) Nursery Data Set. **(b) KDD Cup 1999 Data Set**.

(c) Letter Recognition Data Set.

Figure 1. Percentage improvement for memory consumption in proposed sWFCM-HD over baseline (FCM)**.**

C. Execution Time

Similar as memory consumption we also calculated execution time for each chunk separately and take the largest value as the final execution time for our proposed algorithm. Our main aim here is to calculate the execution time of algorithm and sWFCM-HD will only process one chunk at a time and there is no time bound as when the next chunk will arrive. Figure 2 shows the percentage of improvement in sWFCM-HD as compared to baseline in terms of execution time. The huge improvement shown is possible because we compare the execution time of baseline (which uses entire dataset at a time) with the largest execution time by a chunk in sWFCM-HD. The total execution time (adding the execution time of



(a) Nursery Data Set.

(b) KDD Cup 1999 Data Set.

(c) Letter Recognition Data Set

Figure 2. Percentage improvement for execution time in proposed sWFCM-HD over baseline (FCM).

VI. CONCLUSION

To mine the data from the data streams is very difficult because of the limited amount of memory availability and real time query response requirement. The major task to perform mining on any input data is through clustering. On the other hand, high-dimensional data poses different problem (challenges) for clustering algorithms that require specialized solutions. In high dimensional data, for clustering traditional similarity measures as which used in conventional clustering algorithms are usually not meaningful. In this paper we propose a dimension reduced weighted fuzzy clustering algorithm (sWFCM-HD). The algorithm can be used for high dimensional datasets having streaming behavior. Such as data from sensor networks, data generated by web click stream and data stream from internet traffic data transfer etc these data's have two special properties which separate them from other datasets: a) They have streaming behavior and b) They have higher dimensions. Optimized fuzzy clustering algorithm has already been proposed for datasets having streaming behavior or higher dimensions. But as per our information, nobody has proposed any optimized fuzzy clustering algorithm for data sets having both the properties, i.e., data sets with higher dimension and also continuously arriving streaming behavior. Experimental analysis shows that our proposed algorithm (sWFCM-HD) improves performance in terms of memory consumption as well as execution time.

REFERENCES

[1] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom, "Models and issues in data stream Systems," in Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, ser. PODS '02, 2002, pp. 1–16.

[2] L. O'Callaghan, N. Mishra, A. Meyerson, S. Guha, and R. Motwani, "Streaming-data algorithms for high-quality Clustering," in Data Engineering, 2002. Proceedings. 18thInternational Conference on, 2002, pp. 685–694.

[3] P. Domingos and G. Hulten, "A general method for scaling up machine learning algorithms and its Application to Clustering," in Proceedings of the Eighteenth International Conference on Machine Learning, ser. ICML '01, 2001, pp. 106–113.

[4] P. Rodrigues, J. Gama, and J. Pedroso, "Hierarchical clustering of time-series data streams," Knowledge and Data Engineering, IEEE Transactions on, vol. 20, no. 5, pp. 615– 627, 2008.

[5] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "A framework for clustering evolving data streams," in Proceedings of the 29th international conference on Very large data bases - Volume 29, ser. VLDB '03, 2003, pp. 81–92.

[6] S. Eschrich, J. Ke, L. Hall, and D. Goldgof, "Fast fuzzy clustering of infrared images," in IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th, vol. 2, 2001, pp. 1145–1150 vol.2.

[7] M. B. Al-Zoubi, A. Hudaib, and B. Al-Shboul, "A fast fuzzy clustering algorithm," in Proceedings of the 6thConference on 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases - Volume 6, ser. AIKED'07, 2007, pp. 28–32.

[8] R. Wan, X. Yan, and X. Su, "A weighted fuzzy clustering algorithm for data stream," in Proceedings Of the 2008 ISECS International Colloquium on Computing, Communication, Control, and Management - Volume 01, ser. CCCM '08, 2008,pp. 360–364.

[9] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional spaces," in Proceedings of the 8th International Conference on Database Theory, ser. ICDT '01, 2001, pp. 420– 434.

[10] H.-P. Kriegel, P. Kr¨oger, and A. Zimek, "Clustering high dimensional data: A survey on subspace Clustering, pattern based clustering, and correlation clustering," ACM Trans. Knowl. Discov. Data, vol. 3, no. 1, pp. 1:1–1:58, Mar. 2009.

[11] P. Bishnu and V. Bhattacherjee, "A dimension reduction technique for k-means clustering algorithm," in Recent Advances in Information Technology (RAIT), 2012 1st International Conference on, 2012, pp. 531–535.

[12] N. R. Pal and J. C. Bezdek, "Complexity reduction for "large image" processing," Trans. Sys. Man Cyber. Part B, vol. 32, no. 5, Oct. 2002.

[13] D. Altman, "Efficient fuzzy clustering of multi-spectral images," in Geoscience and Remote Sensing Symposium,1999. IGARSS '99 Proceedings. IEEE 1999 International, vol. 3, 1999, pp. 1594–1596 vol.3.

[14] I. Fodor. (2002) A Survey of Dimension Reduction Techniques. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.8.5098

[15] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," SCIENCE, vol.290, pp. 2323–2326, 2000.

[16] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," Science, vol. 290, no. 5500, pp. 2319–2323, 2000.

[17] L. Teng, H. Li, X. Fu, W. Chen, and I.-F. Shen, "Dimension reduction of microarray data based on Local tangent space alignment," in Proceedings of the Fourth IEEE International Conference on Cognitive Informatics, ser. ICCI '05, 2005, pp. 154–159.

[18] R. Urtasun and T. Darrell, "Discriminative gaussian process latent variable model for classification," In Proceedings of the 24th international conference on Machine learning, ser. ICML '07, 2007, pp. 927–934.

[19] L. Tan and Y. Zhang, "A comparative study of dimension reduction based on data distribution," in Intelligent Systems (GCIS), 2010 Second WRI Global Congress on, vol. 3, 2010, pp. 309–312.

[20] Diksha Upadhyay, Susheel Jain, Anurag Jain "Comparative Analysis of Various Data Stream Mining Procedures and Various Dimension Reduction Techniques" International journal of Advanced Research in computer science "Volume 4, No.8, May-June 2013.