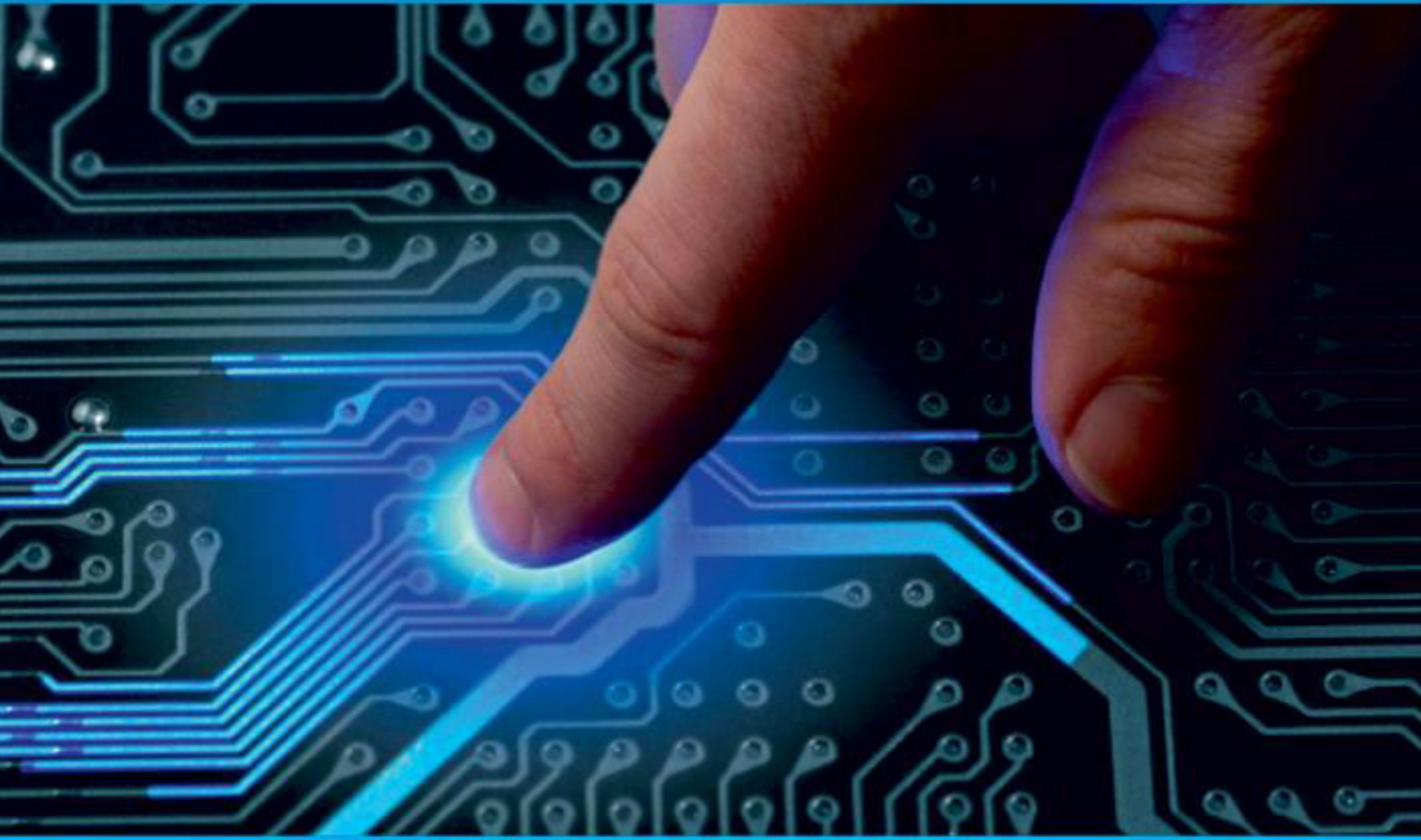
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 8, August 2022

ISSN
INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.165

Clustering Data Streams Based on Shared Density between Micro-Clusters

Rohini R.Panddilolu, Pottigar Vinayak V

M. E Student, Department of Computer Science and Engineering, N B Navale Sinhgad College of Engineering Kegaon, Solapur, Solapur University, India

Professor, Department of Computer Science and Engineering, N B Navale Sinhgad College of Engineering Kegaon, Solapur, Solapur University, India

ABSTRACT: Micro-clusters represent local density estimates by aggregating the information of many data points in a defined area. On demand, a (modified) conventional clustering algorithm is used in a second offline step to recluster the micro-clusters into larger final clusters. For reclustering, the centers of the micro-clusters are used as pseudo points with the density estimates used as their weights. However, information about density in the area between micro-clusters is not preserved in the online process and reclustering is based on possibly inaccurate assumptions about the distribution of data within and between micro-clusters (e.g., uniform or Gaussian). This paper describes DBSTREAM, the first micro-cluster-based online clustering component that explicitly captures the density between micro-clusters via a shared density graph. The density information in this graph is then exploited for reclustering based on actual density between adjacent micro-clusters. We discuss the space and time complexity of maintaining the shared density graph.

KEYWORDS: Data Streams, Density Based Clustering, Micro Cluster

I. INTRODUCTION

In recent years demands of data stream clustering increases rapidly .Data stream are observed in network monitoring, critical scientific application, weather monitoring and astronomical applications, elect ronic business, stock trading ,social networks ,sensor network etc. In these applications ,data stream arrives continuously and evolve significantly over time.

1.1 Background

There are many technologies available which facilitates us to record day to life transactions at rapid rate. Such process lead to large volume of continuous data. This data term as 'Data Stream'. Data streams are highly dynamic, massive and unbounded in nature. Due to these characteristics real-time data stream clustering is challenging problem. Data stream clustering puts additional constraints on clustering algorithms. Clustering in data stream environment needs some special requirements due to data stream's characteristics such as clustering in bounded memory and within limited processing time as well as with single pass over evolving data streams.

1.2 Motivation

Data stream clustering is generally divided in two phases online and offline. Online phase summarized data into many micro clusters and then in offline phase micro clusters are merged and form macro cluster. Reclustering is offline process hence its does not have limited time bound. In literature various data stream clustering methods are discussed like hierarchical and partitioning which are use to create spherical-shape clusters. Density based clus tering is one of the most important method to discover non-spherical shape and outliers. DENCLUE, DBSCAN, OPTICS, are density based clustering algorithm. These algorithm focuses on dense area of data points in data space and identify as cluster as they are separated by low density area .Another important method of clustering is grid based. Grid based clustering method has fast processing time and it is not depended on number of data points.

II. RELATED WORK

In this paper we address the issue of overwhelmingly large output size. We also specify a bound on the number of extra sets that are allowed to be covered. We examine different problem variants for which we demonstrate the hardness of the corresponding problems and we provide simple polynomial-time approximation algorithms. We give empirical evidence showing that the approximation methods work well in practice .

The algorithms for finding frequent patterns are complete: they find all patterns that occur sufficiently often. Completeness is a desirable property,of course. How-ever,in many cases the collection of frequent patterns is large,and obtaining a global understanding of which pat-terns are frequent and which are not is not easy. Even re-stricting the output to the border of the frequent item-set collection does not help much in alleviating the problem. The case when the input is the original database is per-haps the most interesting open algorithmic question. This case presents significant difficulties. First,computing the border in time polynomial to its size is a main open prob-lem. Furthermore,the size of the border can be exponential in the size of the database,and therefore one cannot afford looking at the whole search space—some kind of sampling method needs to be applied .

This goal,however,is different from our set-tingwhereweaskforthe k sets that best approximate the frequent item-set collection in the sense of set coverage. The work on frequent closed item sets attempts to compress the collection of frequent sets in a lossless manner,while for the condensed frequent item sets the idea is to be able to reduce the output size by allowing a small error on the support of the frequent item sets. The second set, Course,is from anonymized student/course registra-tion data in the Department of Computer Science at the University of Helsinki. Frequent course sets were obtained using a support threshold of 2.2%,yielding a collection of size 1637 .

Diabetes is part of the growing epidemic of non communicable diseases, with a high burden for the society on developing countries in future.For suppressing the development of diabetes mellitus and the onset of complications to manage their healthcare or personal data. We aim to apply association rule mining to electronic medical records to discover sets of risk factors. The four methods summaries the high risk of diabetes. Our extension to the bottom up summarization algorithm produced the most suitable summary .

Association rules are implications that associate a set of potentially interacting conditions (e.g. high BMI and the presence of hypertension diagnosis) with elevated risk. The use of association rules is particularly beneficial because in addition to quantifying the diabetes risk, they also readily provide the physician with a “justification”, namely the associated set of conditions. This set of conditions can be used to guide treatment towards a more personalized and targeted preventive care or diabetes management .

A clinical application of association rule mining to identify sets of co-morbid conditions that imply significantly increased risk of diabetes. Association rule mining on this extensive set of variables resulted in an exponentially large set of association rules. The main contribution is a comparative evaluation of these extended summarization techniques that provides guidance to practitioners in selecting an appropriate algorithm for a similar problem .

Association rule mining to identify sets of risk factors and the corresponding patient subpopulations who are at significantly increased risk of progressing to diabetes.An excessive number of association rules were discovered impeding the clinical interpretation of the results. For this method to be useful, the number of rules is used for clinical interpretation is make feasible .

Many of these rules are slight variants of each other leading to the obfuscation of the clinical patterns underlying the ruleset. One remedy to this problem, which constitutes the main focus of this work, is to summarize the ruleset into a smaller set that is easier to overview. We first review the existing rule set and database summarization methods, then propose a generic framework that these methods fit into and finally, we extend these methods so that they can take a continuous outcome variable (the martingale residual in our case) into account

The significance is usually defined by the context of applica-tions. Previous studies have been concentrating on how to compute top-ksignificant patterns or how to remove redundancy among patterns separately. There is limited work on finding those top-kpatterns which demonstrate high-significance and low-redundancy simultaneously. In this paper, we study the problem of extracting redundancy-aware top-kpatterns from a large collection of frequent patterns. We first examine the evaluation functions for measuring the combined significance of a pattern set and propose theMMS(Maximal Marginal Significance) as the problem formulation .

The second example isdocument theme extraction, where each document (or each sentence) is treated as a transaction. The goal is to extract the frequent patterns of term occurrence, calledthemes, buried in a large set of documents. Given a document set, the top- kfrequent patterns returned by a mining algorithm are not necessarily the bestkthemes one can find. Many frequent term sets could overlap significantly with each other. Such overlapping may render top-k important themes very redundant .

In this paper, we formulate the redundancy-aware top-kpattern extraction problem through a general rank-ing model which integrates two measures, significance and redundancy, into one objective function. We first examine the evaluation functions for measuring the com-bined significance of a pattern set and propose theMMS (Maximal Marginal Significance) as the problem formu-lation. TheMMSproblem is equivalent to searching a constrained rooted minimum spanning tree on the directed redundancy graph such that the overall weights on the root node and on the edges in the tree are maximized. The constraint specifies that the root must be the most significant pattern in the tree .

To extract redundancy-aware top-kpatterns, we ex-amined two problem formulations:MASandMMS. We studied a unified greedy approach to compare these two functions and show that MMSis a reasonable formulation to our problem. We further present an improved algorithm forMMSand show that the performance is bounded byO(logk). We present two case studies to examine the performance of our proposed approaches. BothMMSalgorithms are able to find high-significant and low-redundant top-kpatterns. Particularly, in block correlation experiments, we observe that our improved algorithm performs better. This study opens a new direction on finding both diverse and significant top-kanswers to querying, search- ing, and mining, which may lead to promising further studies .

In this paper, we study the problem of compressing frequent-pattern sets. Typically, frequent patterns can be clustered with a tight-ness measure±(called ±-cluster), and arepre-sentative patterncan be selected for each clus-ter. Unfortunately, finding a minimum set of representative patterns is NP-Hard. We develop two greedy methods,RPglobalandRPlo- cal. The former has the guaranteed compres-sion bound but higher computational com-plexity. The latter sacrifices the theoretical bounds but is far more efficient. Our per-formance study shows that the compression quality usingRPlocalis very close to RPglobal, and both can reduce the number of closed frequent patterns by almost two orders of magnitude. Furthermore, RPlocalmines even fasterthan FPClose, a very fast closed frequent-pattern mining method. We also show that RPglobaland RPlocal can be combined to- gether to balance the quality and efficiency .

There have been many scalable methods developed for frequent-pattern mining. However, the real bottleneck of the problem is not at the efficiency but at the usability. Typically, ifminsupis high, min-ing may generate only commonsense patterns, how-ever, with a lowminsup, it may generate an explosive number of results. This has severely restricted the us-age of frequent-pattern mining. To solve this problem, it is natural to explore how to “compress” the patterns, i.e., find a concise and succinct representation that describes the whole col-lection of patterns. Two major approaches have been developed in this direction: lossless compression and lossy approximation .

TheRP-globalmethod has theoretical bound, and works well on small collections of frequent patterns. TheRPlocalmethod is quite efficient, and preserves reasonable compression quality. We also discuss a combined ap-proach, RPcombine, to balance the quality and effi-ciency. TheRPlocal method can be used as sampling procedure as we did in RPcombine, since it is efficient and achieves con-siderable compression. Second, the compressed pat-tern sets generated by our method can be used for queries of finding approximate supports .

The objective of the clustering is to minimize the number of clusters (hence the number of repre-sentative patterns). Finally, we show the problem is equivalent to set-covering problem, and it is NP-hard w.r.t. the number of the frequent patterns to be com-pressed. We propose two greedy algorithms: the first one, RPglobal, has bounded compression quality but higher computational complexity; whereas the second one, RPlocal, sacrifices the theoretical bound but is far more efficient .

III. PROPOSED SYSTEM ARCHITECTURE

Reclustering represents the algorithm’s offline component which uses the data captured by the online component. For simplicity we discuss two-dimensional data first and later discuss implications for higher-dimensional data. For reclustering, we want to join MCs which are connected by areas of high density. This will allow us to form macro-clusters of arbitrary shape, similar to hierarchical clustering with single-linkage or DBSCAN's reachability, while avoiding joining MCs which are close to each other but are separated by an area of low density.

For two-dimensional data the intersection factor a has a theoretical maximum of 0.391 for an area of uniform density when the two MCs are optimally packed (the centers are exactly r apart). However, in dynamic clustering situations MCs may not be perfectly packed all the time and minor variations in the observed density in the data are expected. Therefore, a smaller value than the theoretically obtained maximum of 0.391 will be used in practice. It is important to notice that a threshold on a is a single decision criterion which combines the fact that two MCs are very close to each other and that the density between them is sufficiently high.

Two MCs have to be close together or the intersecting area and thus the expected weight in the intersection will be small and the density between the MCs has to be high relative to the density of the two MCs. This makes using the concept of a -connectedness very convenient.To remove noisy MCs from the final clustering, we have to detect these MCs. Noisy clusters are typically characterized as having low density represented by a small weight. Since the weight is related to the number of points covered by the MC, we use a user-set minimum weight threshold to identify noisy MCs. This is related to min Points in DBSCAN or Cm used by D-Stream

In dimensions higher than two the intersection area becomes an intersection volume. To obtain the upper limit for the intersection factor a we use a simulation to estimate the maximal fraction of the shared volume of MCs (hyper-spheres) that intersect in d ¼ 1; 2 ;; 10; 20 and 50-dimen-sional space. The results are shown in Table 1. With increasing dimensionality the volume of each hyper sphere increases much more than the volume of the intersection. This leads at higher dimensions to a situation where it becomes very unlikely that we observe many data points in the intersection. This is consistent with the problem known as the curse of dimensionality which effects distance-based clustering as well as Euclidean density estimation. This also effects other density based algorithms (e.g., D-Stream's attraction) in the same way. For high-dimensional data we plan to extend a subspace clustering approach like HPStream to maintain a shared density graph in lower-dimensional subspaces.

Advantages:

1. This improves performance and, in many cases, the saved memory more than offset the memory requirement for the shared density graph.
2. Shared-density reclustering already performs extremely well when the online data stream clustering component is set to produce a small number of large MCs.

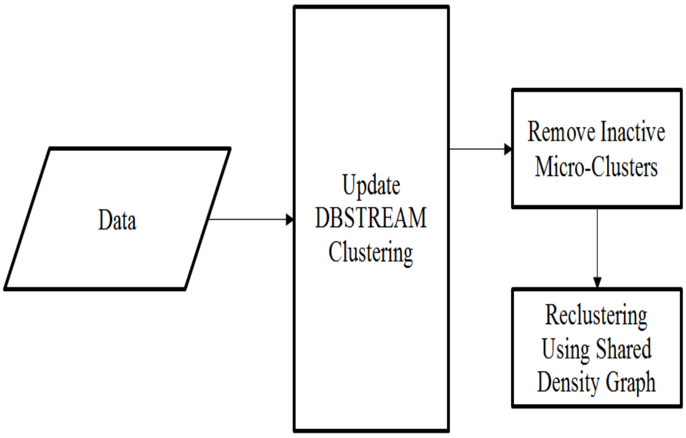
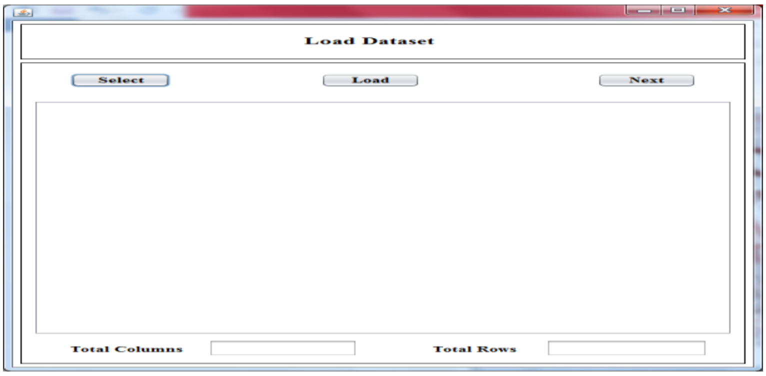

Data
Update DBSTREAM Clustering
Remove Inactive Micro-Clusters
Reclustering Using Shared Density Graph

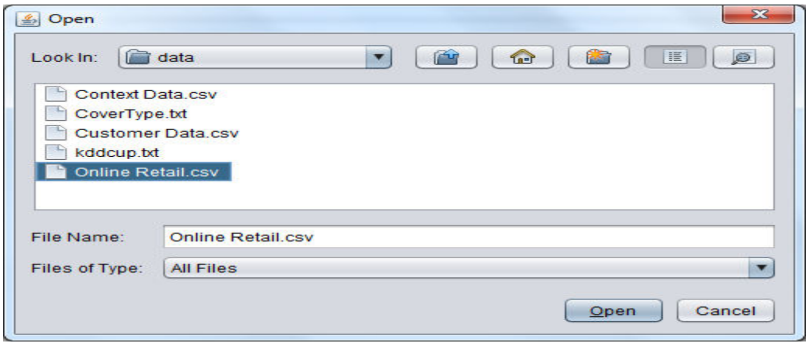
Figure 01. Proposed System Architecture

IV. RESULTS

As shown in screenshot A, three options here first click on select and select dataset then open.
Some datasets are shown in Screenshot B, we select one of them i.e.online retail.csv.



Screenshot A



Look In: data

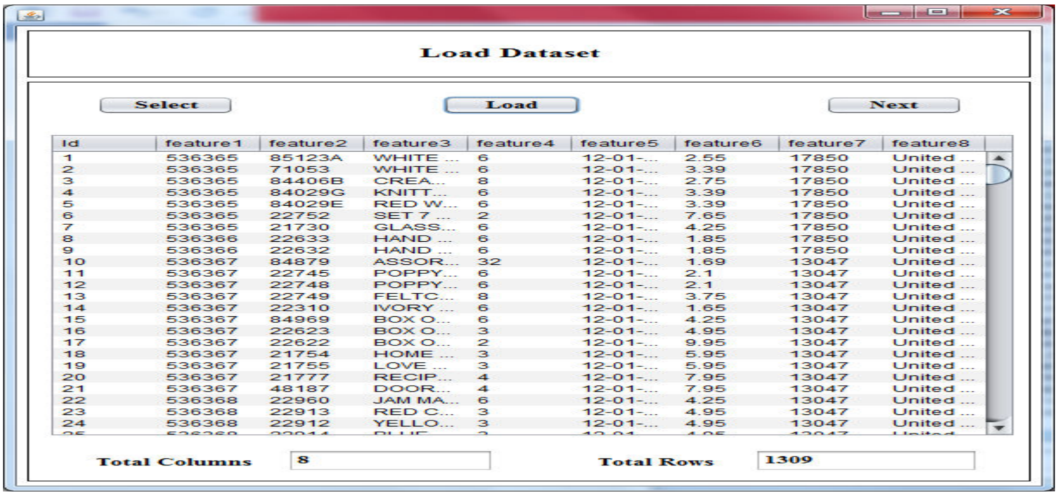
Context Data.csv
CoverType.txt
Customer Data.csv
kddcup.txt
Online Retail.csv

File Name: Online Retail.csv

Files of Type: All Files

Screenshot B

After this Load dataset. Screenshot C shows load dataset which related to our system. Load data set into data base and show the loaded database.The input of Big Data comes from social networks , Web servers, satellite imagery, sensory data, banking transactions, etc. Load remote sensing data into database. Pre-process data for remove irrelevant data. Preprocess done successfully.



Id	feature1	feature2	feature3	feature4	feature5	feature6	feature7	feature8
1	536365	85123A	WHITE ...	6	12-01-...	2.55	17850	United ...
2	536365	71053	WHITE ...	6	12-01-...	3.39	17850	United ...
3	536365	84406B	CREA...	8	12-01-...	2.75	17850	United ...
4	536365	84029G	KNITT...	6	12-01-...	3.39	17850	United ...
5	536365	84029E	RED W...	6	12-01-...	3.39	17850	United ...
6	536365	22752	SET 7 ...	2	12-01-...	7.65	17850	United ...
7	536365	21730	GLASS...	6	12-01-...	4.25	17850	United ...
8	536366	22633	HAND ...	6	12-01-...	1.85	17850	United ...
9	536366	22632	HAND ...	6	12-01-...	1.85	17850	United ...
10	536367	84879	ASSOR...	32	12-01-...	1.69	13047	United ...
11	536367	22745	POPPY...	6	12-01-...	2.1	13047	United ...
12	536367	22748	POPPY...	6	12-01-...	2.1	13047	United ...
13	536367	22749	FELTC...	8	12-01-...	3.75	13047	United ...
14	536367	22310	IVORY ...	6	12-01-...	1.65	13047	United ...
15	536367	84969	BOX O...	6	12-01-...	4.25	13047	United ...
16	536367	22623	BOX O...	3	12-01-...	4.95	13047	United ...
17	536367	22622	BOX O...	2	12-01-...	9.95	13047	United ...
18	536367	21754	HOME ...	3	12-01-...	5.95	13047	United ...
19	536367	21755	LOVE ...	3	12-01-...	5.95	13047	United ...
20	536367	21777	RECIP...	4	12-01-...	7.95	13047	United ...
21	536367	48187	DOOR...	4	12-01-...	7.95	13047	United ...
22	536368	22960	JAM MA...	6	12-01-...	4.25	13047	United ...
23	536368	22913	RED C...	3	12-01-...	4.95	13047	United ...
24	536368	22912	YELLO...	3	12-01-...	4.95	13047	United ...

Total Columns 8

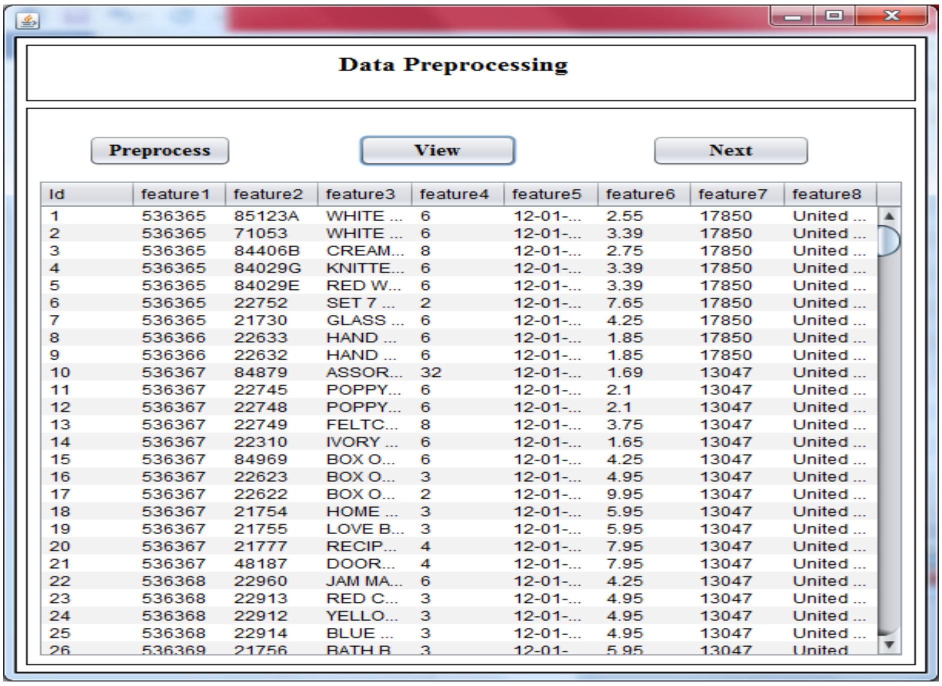
Total Rows 1309

Screenshot C

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165

|| Volume 10, Issue 8, August 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1008006 |

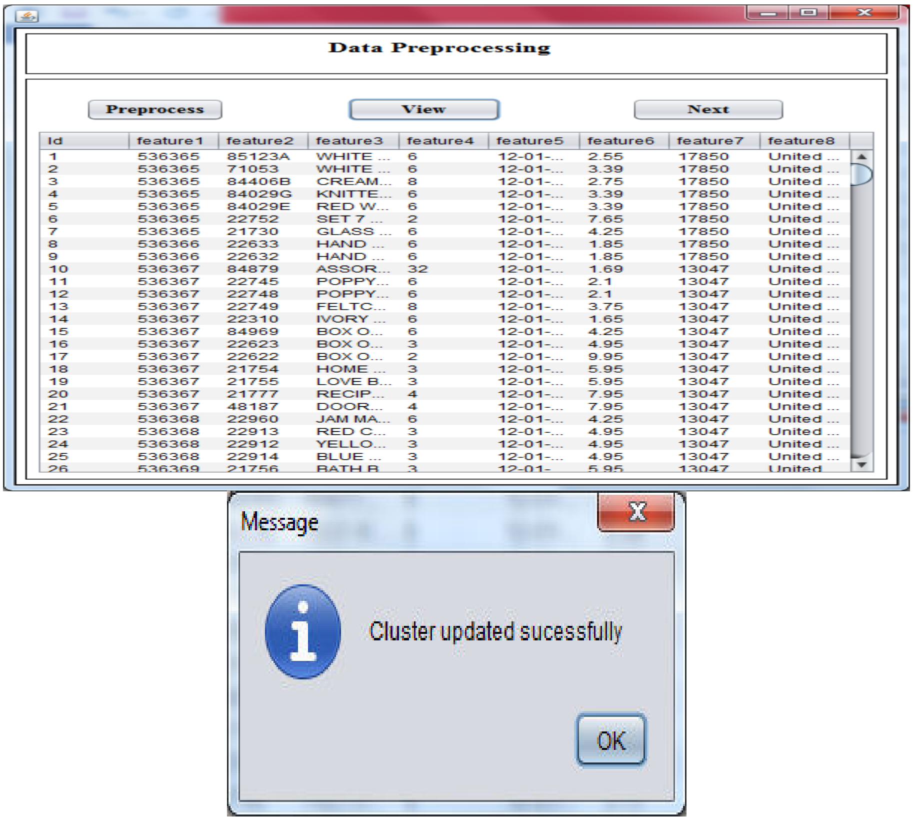


Data Preprocessing

Id	feature1	feature2	feature3	feature4	feature5	feature6	feature7	feature8
1	536365	85123A	WHITE ...	6	12-01-...	2.55	17850	United ...
2	536365	71053	WHITE ...	6	12-01-...	3.39	17850	United ...
3	536365	84406B	CREAM...	8	12-01-...	2.75	17850	United ...
4	536365	84029G	KNITTE...	6	12-01-...	3.39	17850	United ...
5	536365	84029E	RED W...	6	12-01-...	3.39	17850	United ...
6	536365	22752	SET 7 ...	2	12-01-...	7.65	17850	United ...
7	536365	21730	GLASS ...	6	12-01-...	4.25	17850	United ...
8	536366	22633	HAND ...	6	12-01-...	1.85	17850	United ...
9	536366	22632	HAND ...	6	12-01-...	1.85	17850	United ...
10	536367	84879	ASSOR...	32	12-01-...	1.69	13047	United ...
11	536367	22745	POPPY...	6	12-01-...	2.1	13047	United ...
12	536367	22748	POPPY...	6	12-01-...	2.1	13047	United ...
13	536367	22749	FELTC...	8	12-01-...	3.75	13047	United ...
14	536367	22310	IVORY ...	6	12-01-...	1.65	13047	United ...
15	536367	84969	BOX O...	6	12-01-...	4.25	13047	United ...
16	536367	22623	BOX O...	3	12-01-...	4.95	13047	United ...
17	536367	22622	BOX O...	2	12-01-...	9.95	13047	United ...
18	536367	21754	HOME ...	3	12-01-...	5.95	13047	United ...
19	536367	21755	LOVE B...	3	12-01-...	5.95	13047	United ...
20	536367	21777	RECIP...	4	12-01-...	7.95	13047	United ...
21	536367	48187	DOOR...	4	12-01-...	7.95	13047	United ...
22	536368	22960	JAM MA...	6	12-01-...	4.25	13047	United ...
23	536368	22913	RED C...	3	12-01-...	4.95	13047	United ...
24	536368	22912	YELLO...	3	12-01-...	4.95	13047	United ...
25	536368	22914	BLUE ...	3	12-01-...	4.95	13047	United ...
26	536369	21756	BATH B	3	12-01-	5.95	13047	United

Screenshot D

Cluster update output.



Data Preprocessing

Id	feature1	feature2	feature3	feature4	feature5	feature6	feature7	feature8
1	536365	85123A	WHITE ...	6	12-01-...	2.55	17850	United ...
2	536365	71053	WHITE ...	6	12-01-...	3.39	17850	United ...
3	536365	84406B	CREAM...	8	12-01-...	2.75	17850	United ...
4	536365	84029G	KNITTE...	6	12-01-...	3.39	17850	United ...
5	536365	84029E	RED W...	6	12-01-...	3.39	17850	United ...
6	536365	22752	SET 7 ...	2	12-01-...	7.65	17850	United ...
7	536365	21730	GLASS ...	6	12-01-...	4.25	17850	United ...
8	536366	22633	HAND ...	6	12-01-...	1.85	17850	United ...
9	536366	22632	HAND ...	6	12-01-...	1.85	17850	United ...
10	536367	84879	ASSOR...	32	12-01-...	1.69	13047	United ...
11	536367	22745	POPPY...	6	12-01-...	2.1	13047	United ...
12	536367	22748	POPPY...	6	12-01-...	2.1	13047	United ...
13	536367	22749	FELTC...	8	12-01-...	3.75	13047	United ...
14	536367	22310	IVORY ...	6	12-01-...	1.65	13047	United ...
15	536367	84969	BOX O...	6	12-01-...	4.25	13047	United ...
16	536367	22623	BOX O...	3	12-01-...	4.95	13047	United ...
17	536367	22622	BOX O...	2	12-01-...	9.95	13047	United ...
18	536367	21754	HOME ...	3	12-01-...	5.95	13047	United ...
19	536367	21755	LOVE B...	3	12-01-...	5.95	13047	United ...
20	536367	21777	RECIP...	4	12-01-...	7.95	13047	United ...
21	536367	48187	DOOR...	4	12-01-...	7.95	13047	United ...
22	536368	22960	JAM MA...	6	12-01-...	4.25	13047	United ...
23	536368	22913	RED C...	3	12-01-...	4.95	13047	United ...
24	536368	22912	YELLO...	3	12-01-...	4.95	13047	United ...
25	536368	22914	BLUE ...	3	12-01-...	4.95	13047	United ...
26	536369	21756	BATH B	3	12-01-	5.95	13047	United

Message

Cluster updated sucessfully

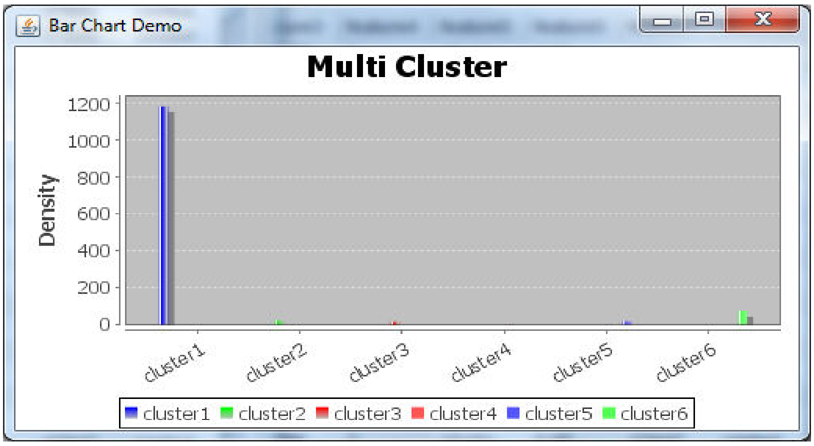
Screenshot E

Graph A shows the Multi cluster graph. In this graph horizontally see the naming cluster1,cluster2,cluster3,cluster4,cluster5,cluster6 and vertically see the density of cluster.

Here in graph showing the various clusters have different density. Cluster 1 having more density and other clusters have low density.

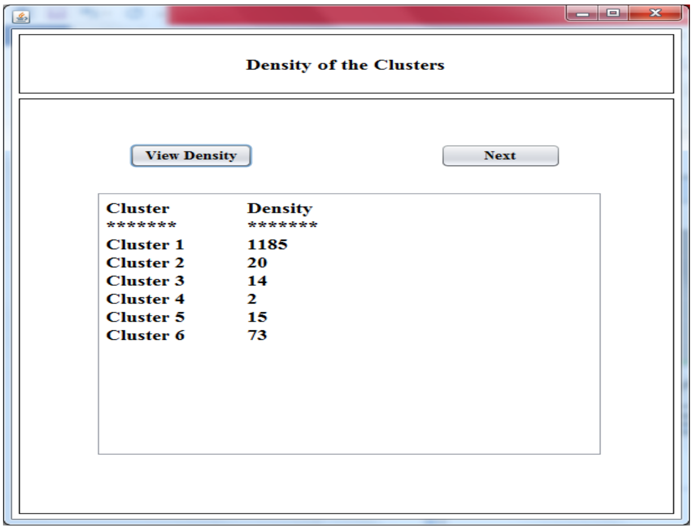
Now, screenshot F shows the density of each cluster. In that clusters find the active cluster and weak cluster.

Screenshot G shows the cluster 1 is active cluster which have density 1185 and other clusters are weak cluster which are cluster 2 having density 20, cluster 3 having density 14,cluster 4 having density 2,cluster 5 having density 15,cluster 6 having density73.



Graph A

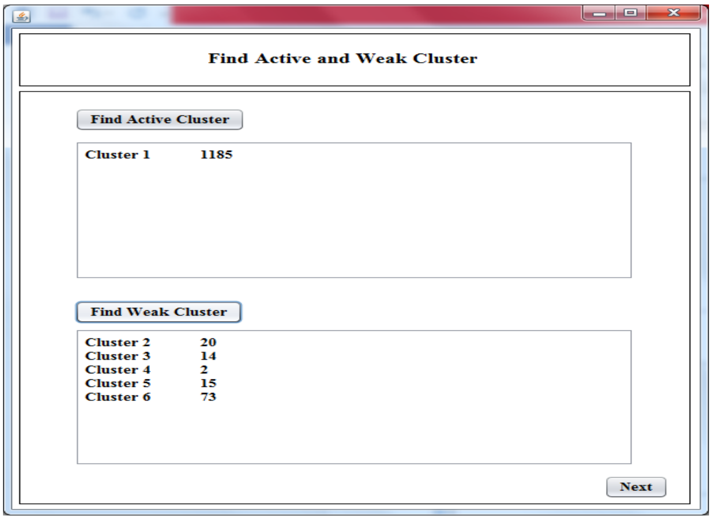
Output of Density of Cluster .



Density of the Clusters

```
Cluster        Density
*******        *******
Cluster 1      1185
Cluster 2      20
Cluster 3      14
Cluster 4      2
Cluster 5      15
Cluster 6      73
```

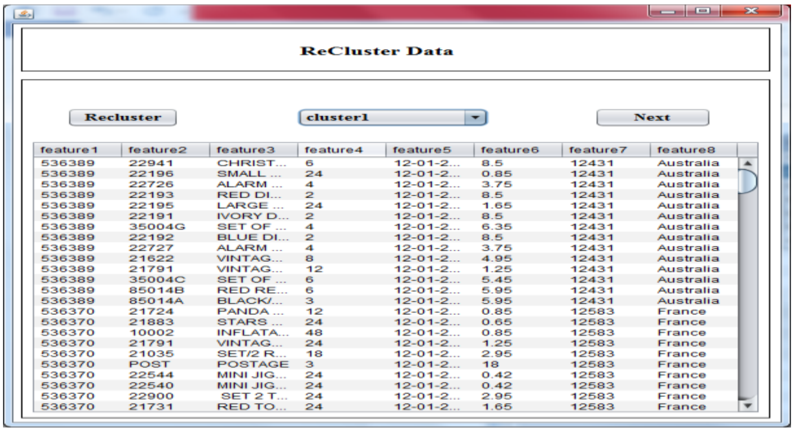
`` Screenshot F



Cluster 1	1185

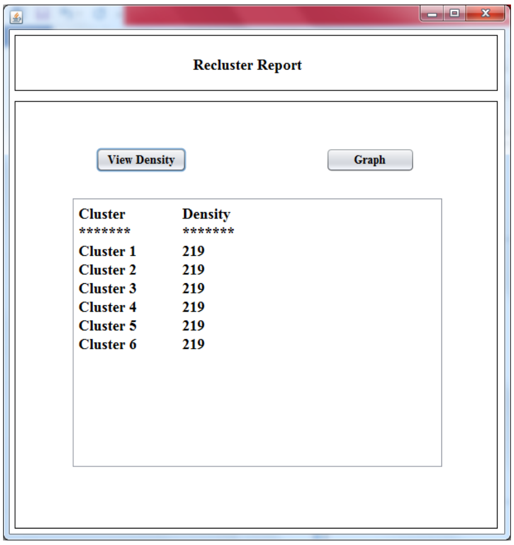
Cluster 2	20
Cluster 3	14
Cluster 4	2
Cluster 5	15
Cluster 6	73

Screenshot G



feature1	feature2	feature3	feature4	feature5	feature6	feature7	feature8
536389	22941	CHRIST...	6	12-01-2...	8.5	12431	Australia
536389	22196	SMALL ...	24	12-01-2...	0.85	12431	Australia
536389	22726	ALARM ...	4	12-01-2...	3.75	12431	Australia
536389	22193	RED DI...	2	12-01-2...	8.5	12431	Australia
536389	22195	LARGE ...	24	12-01-2...	1.65	12431	Australia
536389	22191	IVORY D...	2	12-01-2...	8.5	12431	Australia
536389	35004G	SET OF ...	4	12-01-2...	6.35	12431	Australia
536389	22192	BLUE DI...	2	12-01-2...	8.5	12431	Australia
536389	22727	ALARM ...	4	12-01-2...	3.75	12431	Australia
536389	21622	VINTAG...	8	12-01-2...	4.95	12431	Australia
536389	21791	VINTAG...	12	12-01-2...	1.25	12431	Australia
536389	35004C	SET OF ...	6	12-01-2...	5.45	12431	Australia
536389	85014B	RED RE...	6	12-01-2...	5.95	12431	Australia
536389	85014A	BLACK/...	3	12-01-2...	5.95	12431	Australia
536370	21724	PANDA ...	12	12-01-2...	0.85	12583	France
536370	21883	STARS ...	24	12-01-2...	0.65	12583	France
536370	10002	INFLATA...	48	12-01-2...	0.85	12583	France
536370	21791	VINTAG...	24	12-01-2...	1.25	12583	France
536370	21035	SET/2 R...	18	12-01-2...	2.95	12583	France
536370	POST	POSTAGE	3	12-01-2...	18	12583	France
536370	22544	MINI JIG...	24	12-01-2...	0.42	12583	France
536370	22540	MINI JIG...	24	12-01-2...	0.42	12583	France
536370	22900	SET 2 T...	24	12-01-2...	2.95	12583	France
536370	21731	RED TO...	24	12-01-2...	1.65	12583	France

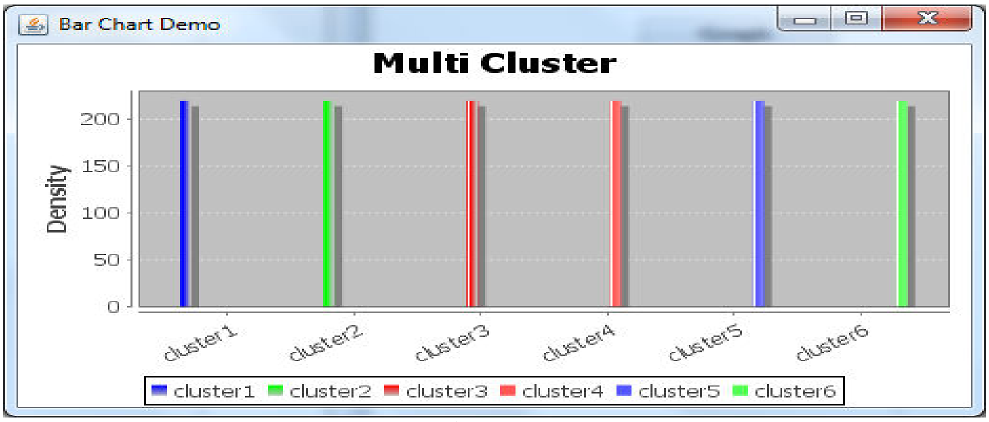
Screenshot H



Cluster	Density
*******	*******
Cluster 1	219
Cluster 2	219
Cluster 3	219
Cluster 4	219
Cluster 5	219
Cluster 6	219

Screenshot I

Recluster the data and recluster report. Various clusters have different density ,therefore recluster the data .
In Screenshot I shows the recluster report .In that all clusters have same density.
Graph B shows the graph of multi cluster having cluster1 to cluster 6 and density of each cluster.
Density of each cluster indicated by different colure lines.Graph shows density is shared in between clusters



Graph B

V. CONCLUSION

In this paper, we have developed the first data stream clustering algorithm which explicitly records the density in the area shared by micro-clusters and uses this information for reclustering. We have introduced the shared density graph together with the algorithms needed to maintain the graph in the online component of a data stream mining algorithm. Although, we showed that the worst-case memory requirements of the shared density graph grow extremely fast with data dimensionality, complexity analysis and experiments reveal that the procedure can be effectively applied to data sets of moderate dimensionality. Experiments also show that shared-density reclustering already performs extremely well when the online data stream clustering component is set to produce a small number of large
MCs. Other popular reclustering strategies can only slightly improve over the results of shared density reclustering and need significantly more MCs to achieve comparable results. This is an important advantage since it implies that we can tune the online component to produce less micro-clusters for shared-density reclustering. This improves performance and, in many cases, the saved memory more than offset the memory requirement for the shared density graph.

REFERENCES

[1] S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan, "Clusteringdata streams," in Proc. ACM Symp. Found. Comput. Sci., 12–14Nov. 2000, pp. 359–366.

[2] C. Aggarwal, Data Streams: Models and Algorithms, (series Advancesin Database Systems). New York, NY, USA: Springer-Verlag, 2007.

[3] J. Gama, Knowledge Discovery from Data Streams, 1st ed. London,U.K.: Chapman & Hall, 2010.

[4] J. A. Silva, E. R. Faria, R. C. Barros, E. R. Hruschka, A. C. P. L. F. d.Carvalho, and J. A. Gama, "Data stream clustering: A survey," ACM Comput. Surveys, vol. 46, no. 1, pp. 13:1–13:31, Jul. 2013.

[5] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "A framework forclustering evolving data streams," in Proc. Int. Conf. Very LargeData Bases, 2003, pp. 81–92.

[6] F. Cao, M. Ester, W. Qian, and A. Zhou, "Density-based clusteringover an evolving data stream with noise," in Proc. SIAM Int. Conf.Data Mining, 2006, pp. 328–339.

[7] Y. Chen and L. Tu, "Density-based clustering for real-time streamdata," in Proc. 13th ACM SIGKDD Int. Conf. Knowl. Discovery DataMining, 2007, pp. 133–142.

[8] L.Wan, W.K.Ng, X.H.Dang,P.S.Yu,andK. Zhang, "Density-based clustering of data streams at multiple reso-lutions," ACM Trans. Knowl. Discovery from Data,vol. 3,no. 3,pp. 1–28, 2009 .

[9] L. Tu and Y. Chen, "Stream data clustering based on grid densityand attraction," ACM Trans. Knowl. Discovery from Data, vol. 3,no. 3, pp. 1–27, 2009.

[10] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-basedalgorithm for discovering clusters in large spatial databases withnoise," in Proc. ACM SIGKDD Int. Conf. Knowl. Discovery DataMining, 1996, pp. 226–231.

[11] A. Hinneburg, E. Hinneburg, and D. A. Keim, "An efficient approach to clustering in large multimedia databases with noise," in Proc. 4th Int. Conf. Knowl. Discovery Data Mining, 1998, pp. 58–65.

[12] L. Ertoz, M. Steinbach, and V. Kumar, "A new shared nearestneighbor clustering algorithm and its applications," in Proc. Work-shop Clustering High Dimensional Data Appl. 2nd SIAM Int. Conf.Data Mining, 2002, pp. 105–115.

[13] G. Karypis, E.-H. S. Han, and V. Kumar, "Chameleon: Hierarchi-cal clustering using dynamic modeling," Computer, vol. 32, no. 8,pp. 68–75, Aug. 1999.

[14] S. Guha, A. Meyerson, N. Mishra, R. Motwani, andL. O'Callaghan, "Clustering data streams: Theory and practice,"IEEE Trans. Knowl. Data Eng. , vol. 15, no. 3, pp. 515–528, M ar. 2003.

[15] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "A framework forprojected clustering of high dimensional data streams," in Proc.Int. Conf. Very Large Data Bases, 2004, pp. 852–863.

[16] D. Tasoulis, N. Adams, and D. Hand, "Unsupervised clustering instreaming data," in Proc. 6th Int. Conf. Data Minning IEEE Int.Workshop Mining Evolving Streaming Data , Dec. 2006, pp. 638–642.

[17] D. K. Tasoulis, G. Ross, and N. M. Adams, "Visualising the clusterstructure of data streams," in Proc. 7th Int. Conf. Intell. Data Anal.VII, 2007, pp. 81–92.

[18] K. Udommanetanakit, T. Rakthanmanon, and K. Waiyamai, "E-stream: Evolution-based technique for stream clustering," in Proc.3rd Int. Conf. Adv. Data Mining Appl. , 2007, pp. 605–615.

[19] P. Kranen, I. Assent, C. Baldauf, and T. Seidl, "The clustree: Index-ing micro-clusters for anytime stream mining," Knowl. Inf. Syst.,vol. 29, no. 2, pp. 249–272, 2011.

[20] A. Amini and T. Y. Wah, "Leaden-stream: A leader density-basedclustering algorithm over evolving data stream," J. Comput. Com-mun., vol. 1, no. 5, pp. 26–31, 2013.

[21] J. A. Hartigan, Clustering Algorithms, 99th ed. New York, NY, USA:
Wiley, 1975.

[22] J. L. Bentley, "A survey of techniques for fixed radius near neigh-bor searching," Stanford Linear Accelerator Center, Menlo Park,CA, USA, Tech. Rep. CS-TR-75-513, 1975.

[23] J. L. Bentley, "Multidimensional binary search trees used for asso-ciative searching,"Commun. ACM, vol. 18, no. 9, pp. 509–517, Sep.1975.

[24] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A geo-metric framework for unsupervised anomaly detection: Detectingintrusions in unlabeled data," in Data Mining for Security Applica-tions . Norwell, MA, USA: Kluwer, 2002.

[25] M. Hahsler and M. H. Dunham, "Temporal structure learning forclustering massive data streams in real-time," in Proc. SIAM Conf.Data Mining, Apr. 2011, pp. 664–675.

[26] C. Isaksson, M. H. Dunham, and M. Hahsler, "Sostream: Self orga-nizing density-based clustering over data stream," in Proc. Mach.Learn. Data Mining Pattern Recog., 2012, vol. 7376, pp. 264–278.

[27] T. Kohonen, "The self-organizing map," Neurocomputing , vol. 21,pp. 1–6, 1998.

[28] T. Kohonen, "Self-organized formation of topologically correctfeature maps," in Neurocomputing: Foundations of Research ,J.A.Anderson and E. Rosenfeld, Eds. Cambridge, MA, USA: MITPress, 1988, pp. 509–521.

[29] J. H. Conway, N. J. A. Sloane, and E. Bannai, Sphere-Packings, Latti-ces, and Groups. New York, NY, USA: Springer-Verlag, 1987.

[30] M. Hahsler, M. Bolanos, and J. Forrest, Stream: Infrastructure forData Stream Mining , 2015, R package version 1.2-2, http://cran.r-project.org/package=stream

[31] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, "MOA: Massiveonline analysis," J. Mach. Learn. Res., vol. 99, pp. 1601–1604, Aug.2010.

[32] H. Kremer, P. Kranen, T. Jansen, T. Seidl, A. Bifet, G. Holmes, andB. Pfahringer, "An effective evaluation measure for clustering onevolving data streams," in Proc. 17th ACM SIGKDD Int. Conf.Knowl. Discovery Data Mining , 2011, pp. 868–876.

[33] A. K. Jain and R. C. Dubes, Algorithms for Clustering Data. UpperSaddle River, NJ, USA: Prentice-Hall, 1988.

[34] J. Gama, R. Sebasti ~ao, and P. P. Rodrigues, "On evaluating streamlearning algorithms," Mach. Learn., vol. 90, pp. 317–346, 2013.

[35] A. Bifet, G. de Francisci Morales, J. Read, G. Holmes, and B.Pfahringer, "Efficient online evaluation of big data stream classi-fiers," in Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery DataMining, 2015, pp. 59–68.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

📱 9940 572 462 © 6381 907 438 ✉ ijircce@gmail.com

Scan to save the contact details

www.ijircce.com