International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2020): 7.803

A Survey of Clustering Algorithms for Streaming

Denis Patrick Bell[1], Yang Chunting[2]

[1]School of Information and Electronic Engineering, Zhejiang University of Science and Technology,
No.318, Liu-he-Road, Hangzhou, Zhejiang Province, 310023, China

[2]School of Information and Electronic Engineering, Zhejiang University of Science and Technology,
No.318, Liu-he-Road, Hangzhou, Zhejiang Province, 310023, China

Abstract: ***Data analysis of real time data streams continues to attract increased attention because of its importance in decision making, which can directly affect real life activities. It is no secret that these real time data streams are generated from numerous software applications and hardware creating a sizeable volume of data that is continuously generated, with evolving features over time. Data evolution with time is referred to as concept drift. Analysis of such streams is quite problematic due its volume. Clustering is not just a method of analyzing data streams of such size but it is additionally less complicated compared to other forms analysis. This paper takes a survey of some important clustering algorithms applicable to the analysis of data streams.***

Keywords: Clustering Algorithms, Outliers, Data Streams, Unsupervised Learning, Concept Drift

1. Introduction

The concept of a data stream is more applicable than that of a static data set in several applications. Unlike static data sets, data streams involve high volume of data that are continuously generated which sometimes cannot be stored in memory as they can be quite large. In other cases where data is stored in memory, the size of the data can be very large that it is impossible to for analysts and researchers to get multiple passes over the data. With these existing issues, insights into such a pool of data can be quite restricted limiting the scope of analysis.

One method of addressing this problem is by grouping data into categories called clusters.Categorizing or classifying systems can be distinctly supervised or unsupervised, with respect to the assignment of new inputs to one of a finite number of discrete supervised classes or unsupervised categories, respectively [1], [2], [3].

Cluster analysis is an unsupervised learning method that deals with the data structure partition [4] so that elements of the same cluster exhibits matching properties while elements of a different cluster display distinguishable properties. By grouping identical clusters, cluster analysis enables the study of patterns and interrelationships.
Clustering data streams is not shielded from challenges. For clustering algorithm to effectively cluster data streams, it must address these issues such that (i) it should be able to scan a stream of data in one pass (ii) must be capable of managing evolving streams of data (iii) must be fast (iv) possess the ability of detecting and managing outliers (v) work online with bounded memory (vi) must be capable of maintaining a sketch of historical data.

Since the size of real time data streams are massive, the algorithms are only interested in a sketch of such streams that are important for learning. The significance of window models is brought up in this context as it indicates which part of the stream history is useful for learning. The common window models include, the sliding model, damped model, titled model and the landmark model.

For the purpose of standardization, Clustering algorithms are categorized into Density based algorithms, Grid based algorithms, Hierarchical algorithms, Model based algorithms and Partition based algorithms which forms the body of this survey.

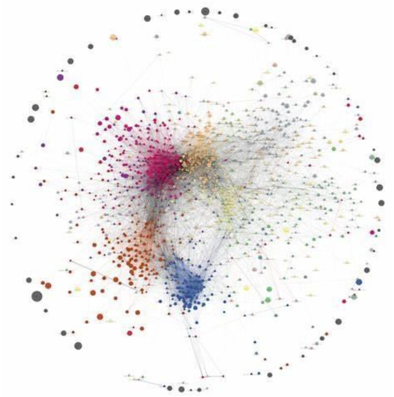

Figure 1: Showing Clustered data

2. Density based algorithms

These algorithms work on data clusters based on the concentration of data points in a region. The main concept of density-based clustering is that for each instance of a cluster the neighborhood of a givenradius (*Eps*) has to contain at least a minimum number of instances (*MinPts)* [5]. It is capable of detecting noisy data while clustering and as such does not require identification in advance. These algorithms only require a single scan of the input dataset and related details to be initialized. Density based clustering algorithms include D-Stream, DenStream, rDenStream, MR-Stream, DBSCAN, GDBSCAN, DENCLUE, DSCLUE, OPTICS and OPClueStream.

2.1 D- Stream

ExclusionLi et al. (2009) proposed the D-Stream for clustering data streams on the basis of density. This algorithm works by creating grids generated from input data online and an offline computation of grid density. The resulting clusters are generated on the basis of the density. D-Stream uses a density decaying factor is used to identify changes in stream of data. This algorithm can be used for real time data streams.

2.2. DenStream

The DenSTREAM algorithm is a two phased density-based algorithm that can work on data streams by clustering emerging data streams [7]. By extending the idea of micro clusters, the Den-Stream algorithm uses potential micro clusters and outliers to identify real data from outliers.

During the first phase the algorithm works by creating micro cluster of the data points. During the second phase, the micro cluster created by the algorithm in the first phase is utilized. TheDen-Stream is built on a fading window paradigm, which reduces the relevance of micro-clusters over time if no new data arrives.

2.3 rDenSTREAM

rDenStream is a three phased upgraded DenStreamalgorithm [8] that works by enhancing the probability of the data to form clusters. Been an upgraded DenStream, rDenStream shares two identical phases with DenStreambut in addition exhibits a third phase that increase accuracy by learning from clusters that are both unimportant and discarded. This discarded data get the chance to increase its weight and form cluster in the future [9] if the depth of the outlier micro cluster is more than the threshold, then they are discarded. Because of its additional work in enhancing the formation of clusters from discarded data, rDenStream requires additional memory and time.

2.4DBSCAN, DENCLUE, GDBSCAN and OPTICS

DBSCAN [10], GDBSCAN [11] DENCLUE [12] and OPTICS are all density-based clustering algorithms that can be effective in identifying random shapes of clusters. Since multiple scanning of data is required in data stream mining, these algorithms are not well suited for this purpose as they only require just one pass to scan the data.

Incremental-DBSCAN, an extension of DBSCAN is capable of mining data streams since it possess the ability to effectively update data in the data warehouse.

An upgrade of the DBSCAN algorithm LDBSCAN can recognize density- based noise and local distinctive characteristics. Unlike the Incremental-DBSCAN, the LDBSCAN cannot properly work on data streams.

OPTICS algorithm is the clustering algorithm of choice for static data sets that are reliant on parameters [13].

OPTICS-Stream [14] is a variant of OPTICS that applies the fading window model to create a sketch of data streams. OPTICS-Stream has both an online and an offline phase. The offline phase create clusters by careful interpretation on the reachability distance and core-distance in OPTICS [15].

2.5 MR-Stream

Formally introduced by Wan et al. (2009), MR-Stream is capable of clustering streams of data at several resolutions applying the fading window model.

MR-Stream work in a two-phase process which are the offline phase and the online phase. The online phase part of the process works by storing concise information regarding streams of data. This stored data is processed by the algorithm in the offline phase to create clusters.

The algorithm consists of a data structure that resembles a tree which maintains the space that is partitioned into cells by MR-Stream. The tree like data structure consists of nodes where each node is a cell at a certain height capable of storing a sketch of data regarding the parent node and the successor nodes. In order to reserve memory, it regularly trim grid cells. Employing the use of minimum distance, the algorithm is able to mergeclusters.

3. Grid Based Algorithm

This is one of the density base algorithm in which a grid like structure is used to locate the density of data points[9].Clustering on Grid-based paradigm divides the space into a finite number of cells, forming a grid like structure where all clustering operations are carried out[17]. Processing time is usually fast and independent of the number of data objects but not independent of the number of cells in every dimension of the quantized space. Grid-based clustering algorithms include CLIQUE, WaveCluster, STING, GCHDS and DGClust.

3.1 CLIQUE

This algorithm formally introduced by Agrawal et al. (1998) works by sampling the data sub-space through a grid and evaluates the density by computing the number of points in a grid cell. By implementing the concept of Apriori, CLIQUE can spontaneously determine subspaces of a high dimensional data space that enhance better clustering than original space.

3.2 DGClust

DGClust is a distributed grid based clustering algorithm proposed Gama et al.(2011)for the data generated by the sensor network. DGClust algorithms enables local sensor to store online sampling of the data that is created. These data are clustered at a fixed interval so that every new incoming data point triggers the creation of a grid [9].

3.3 GCHDS

GCHDS is a grid-based clustering algorithm that is employed to cluster streaming data of high dimensions [20] [21]. GCHDS can process data at a very high speed requiring a single scan of the data. Experimental results have revealed this algorithm possesses high clustering accuracy even better than the HPStream algorithm, which is another subspace algorithm for clustering high dimensional streams of data. A limitation of this algorithm is that it can only locate clusters belonging to the same subspace.

3.4 GSCDS

GSCDS [20] another grid-based algorithm came about due to the limitation of GCHDS been one-dimensional. GSCDS been a multi-dimensional algorithm allows it to identify clusters in various subspaces. This algorithm requires only a single pass through a stream of data to identify clusters in subspaces. GSCDS makes use of an evenly partitioned grid data structure to create a sketch of the streaming data. Using a combination of top-down and bottom-up grid-based approach the algorithm can both locate the subspaces that contain clusters and identify clusters in every single subspace.

3.5 STING

Statistical Information Grid Approach to Spatial Data Mining (STING) formally introduced by Wang et al. (1997) is multi-dimensional grid data structure where by top-down structure is achieved by is dividing data space into many rectangular cells. STING [23] can process at data at a very fast rate because it is independent of the number of data points.

3.6 WaveCluster

WaveCluster is a grid based algorithm that employs the use of a multi-resolution grid structure. The algorithm was proposed by Sheikholeslami et al. (2000) from the viewpoint of signal processing. The basis of WaveCluster is the transformation of wavelets, a signal processing procedure that decomposes a signal into different frequency sub-bands. This algorithm can identify clusters that do not have a specified shape while possessing the ability to detect these clusters at different scales as well as manage noise satisfactorily.

4. Hierarchical Based Clustering Algorithms

This method of clustering involves grouping data on the basis of a hierarchical structure. This technique allows investigation at every level of the hierarchy. With exception of the cluster at the top, every other cluster is branch of a parent cluster and they as well can become parent clusters. Hierarchical clustering approach is divided into agglomerative and divisive types. Hierarchical clustering algorithms include BIRCH [25], CURE [26], E-Stream [27], HUE-Stream [28], ODAC [29] and ROCK [30].

4.1 BIRCH

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies), proposed by Zhang et al. (1996) works by the concept of micro and macro clustering. It is an unsupervised data mining algorithm used to cluster large volume of data. As a result of its ability to cluster massive sets of data, it is ideally employed in data stream mining. It is a scalable, incremental learning based clustering technique based on the idea of a clustering feature CF. The structure of BIRCH includes a clustering feature tree which is a height balance tree that contains the clustering feature. The Clustering Feature is composed of the number of data points in the cluster 'N', the linear sum of the data points 'LS' and the squared sum of data points 'SS'. Using a two-step process, BIRCH scans database and creates the tree containing the Clustering Features, then remove outliers and create new clusters.

4.2 CURE

CURE proposed by Guha et al. (1998) is a hierarchical agglomerative clustering algorithm where every cluster is a representation of scattered points. To obtain the distance between two clusters, it computes the least distance between two scattered points. In order to attain scalability, CURE uses data sampling and data partitioning to achieve this result. This algorithm has the ability to detect clusters of varying shapes and sizes and is not susceptible to noise.

4.3E-Stream and HUE-Stream

This algorithm proposed by Udommanetanakit et al. (2007) is an evolution based hierarchical clustering technique that supports five types of evolution. They are appearance, disappearance, self-evolution, merge and split. The appearance of a new cluster depends on adequate density of data points in a particular area. Self-evolution involves clusters changing behavior due to data changes. Cluster pairs with similar features can merge. An active cluster can split into smaller clusters if it exhibits distinguishing characteristics. Active clusters with old data can disappear as the old data fades away. E-Stream has a polynomial run time with respect to the number of clusters.

HUE-Stream algorithm [28], is an extension of E-Stream and supports uncertainty in heterogeneous streaming of data. HUE-Steam is able to manage heterogeneous data because of the modification of the distance function.

4.4 ODAC

Online divisive agglomerative clustering is a time series data stream clustering technique. It is capable of managing concept drift using both agglomerative and divisive hierarchical methods [29]. It uses a top-down approach to form a tree-like hierarchy of clusters. The algorithm uses correlation-based dissimilarity measure and an agglomerative technique to detect changes in the concept been studied.

5. Model Based Algorithms

Model based clustering works on the hypothesis of aligning the most suitable model to a cluster. This clustering technique can be classed into statistical learning method and neural network learning method. Algorithms using this model include COWEB, SWEM and CluDistream.

5.1 COBWEB

This algorithm is an unsupervised learning algorithm that works by creating hierarchical clustering using a tree-like structure classification. COWEB is a conceptual clustering algorithm that creates a concept label of clusters. In the tree classification, the tree is formed by nodes where each node is a cluster with a probabilistic label of the attributes related to that class of cluster.

Using the agglomerative and divisive approaches, COBWEB is able toform a tree. Employing 'Category Utility' a heuristic measure, COWEB can successfully manage searches [31]. Since it is a model based clustering, each cluster is a model where COBWEB applies incremental learning to form clusters.

5.2 SWEM

SWEM is a model-based algorithm used in clustering streams of data in a time-based sliding window with the expectation maximization technique [32]. Using a two phase process, the algorithm is configured to manage the issue of memory restrictions and single scan processing of data streams. Micro components created by scanning the data in the first phase are used in the second phase to form global data clusters. This flexible technique was employed by Dang et. al (2009) to position micro components in order to solve evolving features of a data stream.

5.3 CluDistream

This algorithm (Zhou et al., 2007) is used to cluster distributed streams of data on the basis of Expectation Maximization (EM). CluDistream works by a remote site processing and a coordinator processing where it combines some Gaussian mixture distributions with a coordinator which combines these models directly from each site. CluDistream can cluster streams of data in the landmark window and the sliding window. The landmark widow is the right fit for this algorithm because there is only insertion compared to the sliding window which contains both insertion and deletion.

6. Partition Based Algorithms

Partitioning based clustering algorithms work as centroid or medoid algorithms. Centroid algorithms characterize any cluster by utilizing the center of gravity of the instances. Medoid algorithms characterize any cluster by means of the instances closest to the center of gravity. Several data stream clustering approaches work on the basis of partitioning. Algorithms in this group include K-means, K-medoids, K-median, CluStream, HPStream [34], SWClustering, Stream LSearch, CLARA [35], CLARANS [36] and PAM [37].

6.1 k-means, k-medoid, k-median

K-Means clustering algorithm is an unsupervised partition based learning technique that uses a repetitive cycle to partition a set of data into k number of clusters in such a way that the center of data points indicate the center of the cluster. This algorithm is simple, fast, adaptable, accurate and effective in processing big datasets. Besides data stream clustering, k-mean has many uses such as image compression and Vector quantization.

There are several variants of this algorithm which includes but not exclusive to Incremental K-means, Scalable K-means, Online K-means, STREAMKM++. Incremental k-means is a balance between On-line K-means and the Standard K-means [79].It can produce clusters from binary data streams. STREAMKM++[38] is used to cluster data streams from a Euclidean space.

K-medoid is the most appropriate data point within a cluster that represents it [39]. K-mediod is advantageous in that it is not as sensitive to outliers as k-means.

StreamLSearch one of the k-median-based clustering algorithms, can [40] cluster high quality data streams. Size matters if the algorithm is to cluster data streams. Therefore it uses the LSEARCH to break streams into smaller portions if the size of the streams are too large from which it can then create clusters.

STREAM is an extension of the k-medians algorithm. It applies the famous divide and conquer strategy to create clusters gradually [40]. Using this approach, STREAM is able to manage memory restrictions and single scan hindrance.

6.2 CLU Stream

Proposed by Aggarwal et al. (2003), CLUStream algorithm works in two phases; one phase of the process being online and the other being offline. The online phase is a warehouse for compressed data streams also known as microclusters since the summarized information from the streams are stored as microclusters. The offline phase of this algorithm works by clustering the summarized data stored in the online phase. It uses pyramidal time frame to cluster evolving data streams. CluStream has difficulty identifying outliers especially in the offline phase because of the presence of k-means algorithm in the offline phase creation of macroclusters. K-means act offline on the streams compressed online to form these macroclusters.

6.3 SW Clustering

SW Clustering [42] is used to cluster streams of data over sliding windows and is capable of analyzing the clusters and their evolution. SWClustering consists of two stages. In the first stage a data structure within the cluster called an Exponential Histogram of Cluster Feature (EHCF) proposed by the authors maintains the features of the cluster. In the second stage it calculates the result of the clustering based on the collection of EHCF. The algorithm promotes better quality of clustering by removing the impact of older cluster while integrating new clusters. CluStream has some similiarity with this algorithm but it uses more memory to store snap shots of clustering. In this regard the sliding window method is not suitable for Clustream.

Table 1: Comparison of Clustering Algorithms

Clustering Algorithm	*Merits*	*Limitations*
Density-Based	Ability to identify shapeless clusters. Ability to handle noise	High computational costs. Requires many predefined parameters.

Grid-Based	Fast processing time. Ability to manage noise. Can process large datasets.	Quality of cluster is reliant on cell size and density.
Hierarchical-Based	Simple and easy to implement. Informative structure.	Not suitable for large datasets. Sensitive to outliers.
Model-Based	Ability to manage noise. Uses probability-based approach to suggest the most suitable model.	Reliance on the model. Predetermined number of clusters. Low Scalability.
Partition-Based	Ability to identify spherical shaped clusters. Can manage noise.	Requires predetermined clusters. Not suitable forlarge datasets.

7. Conclusion

The influence of clustering algorithms on real time data streams has been and is still presently a focal point of many researches aimed at providing improved analysis of data stream.The survey covered five categories of commonly discussed algorithms for data streams. Even though there are more than thirty algorithms in this survey, it is by no means exhaustive since there are many other algorithms not included in this literature. The purpose of the survey is to focus on essential aspects of these algorithms with respect to streaming of data.In this regard an array of important clustering algorithms has been presented in the survey. The different categories of algorithms offer different advantages based on their characteristics. It is essential that these algorithms are used under the right conditions so that they can optimize streaming of real time data Applied use of clustering includes medical imaging, network analysis, traffic analysis, trending hashtags, business analysis and many more.

There are still formidable challenges that clustering of real time data streams must effectively address in order to provide perfect analysis. Since the algorithms cluster a sketch of data based on the important window, it can be rightly assumed that many more sensitive data are lost in the process which cannot be recovered.

References

[1] J.Bezdek and R.Hathaway,“Numerical convergence and interpretation of the fuzzy c-shells clustering algorithms,” IEEE Trans. Neural Netw., vol. 3, no. 5, pp. 787–793, Sep. 1992.

[2] J. Bezdek and N. Pal, “Some new indexes of cluster validity,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 28, no. 3, pp. 301–315, Jun. 1998.

[3] C. Bishop, Neural Networks for Pattern Recognition. New York: Oxford Univ. Press, 1995.

[4] Xu, D. and Tian, Y., 2015. A Comprehensive Survey of Clustering Algorithms. *Annals of Data Science*, 2(2), pp.165-193.

[5] KOTSIANTIS, S. and PINTELAS, P., n.d. Recent Advances in Clustering: A Brief Survey.

[6] Tu, Li, and Yixin Chen. "Stream data clustering based on grid density and attraction." ACM Transactions on Knowledge Discovery from Data (TKDD) 3.3 (2009): 12.

[7] F. Cao, M. Ester, W. Qian, and A. Zhou, "Density-based clustering over an evolving data stream with noise," in Proceedings of the 2006 SIAM International Conference on Data Mining, 2006, pp. 328-339.

[8] L. Li-xiong, K. Jing, G. Yun-fei, and H. Hai, "A three-step clustering algorithm over an evolving data stream," in Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on, 2009, pp. 160-164.

[9] Sharma, N., Masih, S. and Makhija, P., 2018. A Survey on Clustering Algorithms for Data Streams. International Journal of Computer Applications, 182(22), pp.18-24.

[10] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," 1996.

[11] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm gdbscan and its applications," Data Mining and Knowledge Discovery, vol. 2, pp. 169-194, 1998.

[12] A. Hinneburg and D. A. Keim, An efficient approach to clustering in large multimedia databases with noise: Bibliothek der Universität Konstanz, 1998.

[13] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: ordering points to identify the clustering structure," ACM SIGMOD Record, vol. 28, pp. 49-60, 1999.

[14] Tasoulis DK, Ross G, Adams NM (2007) Visualising the cluster structure of data streams. In: Proceedings of the 7th international conference on intelligent data analysis, pp 81–92, 1771633. Springer.

[15] Ankerst M, Breunig MM, Kriegel H-P, Sander J (1999) Optics: ordering points to identify the clustering structure. In: Proceedings of the 1999 ACM SIGMOD international conference on management of data, vol 28, pp 49–60, 304187. ACM.

[16] Wan, L., Ng, W. K., Dang, X. H., Yu, P. S., & Zhang, K. (2009). Density-based clustering of data streams at multiple resolutions. ACM Transactions on Knowledge discovery from Data (TKDD), 3(3), 14.

[17] Lovely Sharma P. and Ramya K.A., Review on density based clustering algorithms for very large datasets, International Journal of Emerging Technology and Advanced Engineering 3(12) (2013), 398–403.

[18] Agrawal R., Gehrke J., Gunopulos D. and Raghavan P. (1998). Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications. In Proc. of the 1998 ACM-SIGMOD Conf. On the Management of Data, 94-105.

[19] Gama, Joao, Pedro Pereira Rodrigues, and Luís Lopes. "Clustering distributed sensor data streams using local processing and reduced communication." Intelligent Data Analysis 15.1 (2011): 3-28.

[20] Sun, Yufen, and Yansheng Lu. "A grid-based subspace clustering algorithm for high-dimensional data streams." International Conference on Web

Information Systems Engineering. Springer, Berlin, Heidelberg, 2006.

[21] Y. Lu, Y. Sun, G. Xu, and G. Liu, "A grid-based clustering algorithm for high-dimensional data streams," in Advanced Data Mining and Applications, ed: Springer, 2005, pp. 824-831.

[22] Wang W., Yang J. and Muntz.R. (1997), STING: A Statistical Information Grid Approach to Spatial Data Mining, Proceedings of the 23rd VLDB Conference Athens, Greece, 1997.

[23] Y. Lu, Y. Sun, G. Xu, and G. Liu, "A grid-based clustering algorithm for high-dimensional data streams," in Advanced Data Mining and Applications, ed: Springer, 2005, pp. 824-831.

[24] Sheikholeslami, Gholamhosein, SurojitChatterjee, and Aidong Zhang. "WaveCluster: a wavelet-based clustering approach for spatial data in very large databases." The VLDB Journal—The International Journal on Very Large Data Bases 8.3-4 (2000): 289-304.

[25] Zhang T, Ramakrishnan R, Livny M (1996) BIRCH: an efficient data clustering method for very large databases. ACM SIGMOD Rec 25:103–104

[26] Guha S, Rastogi R, Shim K (1998) CURE: an efficient clustering algorithm for large databases. ACM SIGMOD Rec 27:73–84

[27] K. Udommanetanakit, T. Rakthanmanon, and K. Waiyamai, "E-stream: Evolution-based technique for stream clustering," in Advanced Data Mining and Applications, ed: Springer, 2007, pp. 605-615

[28] Meesuksabai, Wicha, ThanapatKangkachit, and KitsanaWaiyamai. "Hue-stream: Evolution-based clustering technique for heterogeneous data streams with uncertainty." International Conference on Advanced Data Mining and Applications. Springer, Berlin, Heidelberg, 2011.

[29] Rodrigues, Pedro Pereira, Joao Gama, and Joao Pedro Pedroso. "ODAC: Hierarchical clustering of time series data streams." Proceedings of the 2006 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2006.

[30] Guha S, Rastogi R, Shim K (1999) ROCK: a robust clustering algorithm for categorical attributes. In: Proceedings of the 15th international conference on data engineering, pp 512-521

[31] Fisher, Doug. "Iterative optimization and simplification of hierarchical clusterings." Journal of artificial intelligence research 4 (1996): 147-178.

[32] X. H. Dang, V. C. Lee, W. K. Ng, and K. L. Ong, "Incremental and adaptive clustering stream data over sliding window," in Database and Expert Systems Applications, 2009, pp. 660-674.

[33] Zhou, Aoying, et al. "Distributed data stream clustering: A fast EM-based approach." Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on. IEEE, 2007.

[34] Aggarwal, Charu C., et al. "A framework for projected clustering of high dimensional data streams." Proceedings of the Thirtieth international conference on Very large data bases-Volume 30. VLDB Endowment, 2004.

[35] Kaufman L, Rousseeuw P (2008) Finding groups in data: an introduction to cluster analysis, vol 344. Wiley, Hoboken. doi:10.1002/9780470316801

[36] Ng R, Han J (2002) Clarans: a method for clustering objects for spatial data mining. IEEE Trans Knowl Data Eng 14:1003–1016

[37] Kaufman L, Rousseeuw P (1990) Partitioning around medoids (program pam). Finding groups in data: an introduction to cluster analysis. Wiley, Hoboken.

[38] M. R. Ackermann, M. Märtens, C. Raupach, K. Swierkot, C. Lammersen, and C. Sohler, "StreamKM++: A clustering algorithm for data streams," Journal of Experimental Algorithmics (JEA), vol. 17, p. 2.4, 2012.

[39] Berkhin P. (2006) A Survey of Clustering Data Mining Techniques. In: Kogan J., Nicholas C., Teboulle M. (eds) Grouping Multidimensional Data. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-28349-8_2

[40] O'Callaghan L, Mishra N, Meyerson A, Guha S, Motwani R (2002) Streaming-data algorithms for high-quality clustering. In: Proceedings of the 18th international conference on data engineering, pp 685–694

[41] Aggarwal C, Han J, Wang J, Yu P (2003) A framework for clustering evolving data streams. In: VLDB, pp 81–92

[42] Zhou A, Cao F, Qian W, Jin C "Tracking clusters in evolving data streams over sliding windows." Knowledge and Information Systems 15.2 (2008): 181-214.