2581-4575

International Journal For Recent Developments in Science & Technology

Crossref

A Peer Reviewed Research Journal

ADAPTIVE DYNAMIC DATA STREAMS USING CLUSTERING ALGORITHM

[1]G.VENENDRA, [2]R.SATYA CHINMAYIE, [3]SK.SHAKEERA AHMADE, [4]MD.ABDUL AZEEZ

[1]ASSISTANT PROFESSOR, CSE DEPARTMENT, NRI INSTITUTE OF TECHNOLOGY, POTHAVARAPPADU.

[2,3,4]UG STUDENT, CSE DEPARTMENT, NRI INSTITUTE OF TECHNOLOGY, POTHAVARAPPADU.

ABSTRACT — In this paper, we present a novel dynamic clustering algorithm that efficiently deals with data streams and achieves several important properties which are not generally found together in the same algorithm. The dynamic clustering algorithm operates online in two different time-scale stages, a fast distance based stage that generates micro-clusters and a density-based stage that groups the micro-clusters according to their density and generates the final clusters. The algorithm achieves novelty detection and concept drift thanks to a forgetting function that allows micro-clusters and final clusters to appear, drift, merge, split or disappear. The outlier identification is made in a natural way using micro-clusters density. This algorithm has been designed to be able to detect complex patterns even in multi-density distributions and making no assumption of cluster convexity. The performance of the dynamic clustering algorithm is assessed theoretically through complexity analysis and empirically through a set of experiments that compare the algorithm with other algorithms of the literature on popular data sets that have interesting characteristics regarding the emergence, disappearance and movement of clusters as well as multi-density, non-convexity and noise.

KEYWORDS — Dynamic clustering, incremental learning, online learning, multi-density, non-convex data sets.

I. INTRODUCTION:

With computer storage capacities of today, massive historical data can be recorded and stored in all the socioeconomic domains: engineering, environment, finance, etc. The cloud also brings an enormously cost-effective way to increase storage. Hence, huge amounts of data, arising from various sources, are collected and they are available for further analysis. Data can indeed be a tremendous source of information and knowledge if it is mined in a smart way. Machine learning, an essential ingredient in the data mining field, currently face new challenges, in particular, building classification techniques able to handle complex data sets and scaling to big data. Classification algorithms typically include two phases: a training phase that makes use of a training data set to generate a set of classes considered as the model and a recognition phase that uses the model to assign the new objects to one of the classes. The training phase may be supervised if the training data set is labeled, i.e. expertise of the domain has allowed labeling each object of this set. In this case, the training phase aims at defining the shape of the classes corresponding to the different concepts existing in the labels and one refers N. Barbosa Roa and L. Trave-Massuy ´ es are with LAAS-CNRS, University ` of Toulouse, Toulouse, France. N. Barbosa Roa and V. Hugo Grisales are with Universidad Nacional de Colombia. Bogota, Colombia. ´ to supervised classification. However, data mining and knowledge discovery generally require unsupervised classification schemes as the concepts of the domain may not even be (all) known a priori. In this case, also known as clustering, the training phase works by grouping the objects according to some predefined criteria. Every cluster is then interpreted as a class for which a concept can be assigned afterwards by an expert.

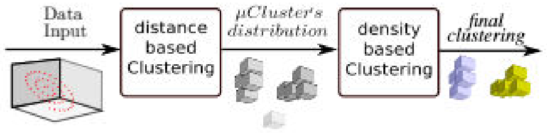

Fig. 1: Global description of the algorithm

II. TWO STAGES DYNAMIC CLUSTERING:

This paper uses the two stages distance- and density-based clustering approach proposed in [14] modified to be able to discover clusters of different densities. In our proposal both stages work on-line, but operate at different time scales. In addition, µ-clusters of similar densities can form clusters of any shape and any size. This multi-density feature allows the detection of novelty behavior in its early stages when only a few objects giving evidence of this evolution are present. The first stage operates at the rate of the data stream and creates µ-clusters putting together data samples that are close, in the sense of a given distance, to each other. µ-clusters are stored in the form of summarized representations including statistical and temporal information. The second stage takes place once each tslow seconds and analyses the distribution of µ-clusters. The density of a µcluster is considered as low, medium or high and is used to create the final clusters by a density based approach, i.e. dense µ-clusters that are close enough (connected) are said to belong to the same cluster. Similarly to [13], a cluster is defined as the group of connected µ-clusters where every inside µ-cluster presents high density and every outside µcluster exhibits either medium or low density. The above dense µ-cluster structure allows the algorithm to create clusters of non convex shapes even in high dimensional spaces and it has proved outliers rejection capabilities in evolving environments ([12], [14]).

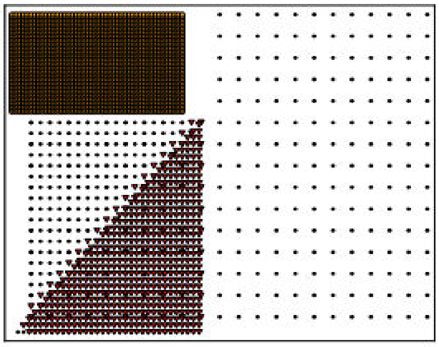

Fig. 2: DBSCAN fails to detect the four different clusters, instead it provides only the highest density clusters (Yellow and Red) and outliers

III. ASSESSING THE PROPERTIES AND PERFORMANCE OF THE ALGORITHM:

In this section the algorithm is analyzed, then tested and compared to other known algorithms to assess its performance. A. Complexity Analysis Since the algorithm has two stages that work independently; its complexity can be evaluated independently for the two stages. The complexity of the distance-based stage is O (dM) for each incoming d-dimensional object, where M is the number of μ-clusters. Since the algorithm is incremental, in general M << n, n being the number of elements to cluster, which leads us to a global complexity of O (n). It is worth noticing that in the exceptional case of small high dimensional datasets with sparse distributions, the inequalities M << n and d << n do not stand any more. In consequence, in these exceptional cases, the complexity becomes O

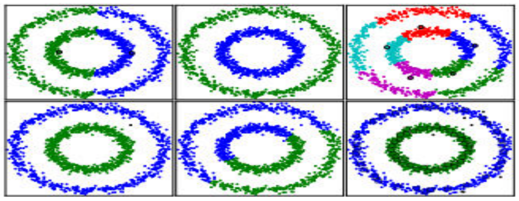

Fig. 3: Comparison of six algorithms for the concentric circles data set. Top left to right: MiniBatch KMeans, Agglomerative Clustering, Affinity Propagation. Bottom: DBSCAN, BIRCH, our algorithm

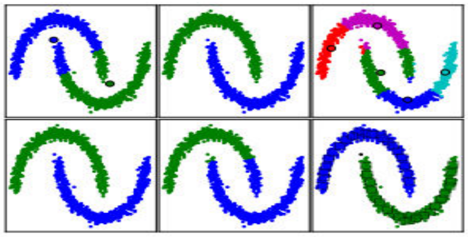

Fig. 4: Comparison of six algorithms for the moons data set. Top left to right: MiniBatch KMeans, Agglomerative Clustering, Affinity Propagation. Bottom: DBSCAN, BIRCH, our algorithm

Two noisy data sets were evaluated named concentric circles and moons. In each data set 1500 samples were considered. These data sets were generated using the scikit-learn.dataset module. The distributions are time invariant. In consequence, the forgetting process of our algorithm μ-clusters is disabled for this test. 1 The results of the tests for the concentric circles data set can be seen in Figure 3 and for the moons data set in Figure 4. In the figures the cluster centers found by MBK-m and AP are drawn as colored circles for illustrative purposes as well as our algorithm Dμ-clusters (colored squares). MBK-m, AC and BIRCH require the number of clusters as an initial parameter but even with this information, MBK-m and BIRCH are not able to cluster these non convex sets properly. AP does not perform well at all in these distributions. Moreover, it creates a high number of clusters. Our algorithm is able to detect non convex distributions as well as DBSCAN does, since they are both density based. Nevertheless, the test shows that our algorithm rejects more outliers than DBSCAN. 2) Robust path based clustering: Figure 5 shows the three spirals distribution used in [18]. In the centre of each spiral, samples are more abundant and then they become sparser as spirals grow out. This kind of distribution is path based and it is particularly difficult to handle for clustering algorithms only based on distance or only based on density. Since our algorithm uses both, distance and density it can overcome this kind of challenge. It indeed achieves results comparable to those of the original article presented by Chang and Yeung [18] obtained with their robust path-based spectral clustering method as can be seen in Figure 5

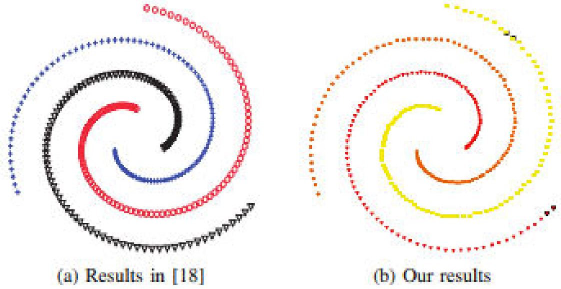
(a) Results in [18] (b) Our results

Fig. 5: Robust path-based spectral clustering and our algorithm tested against the three spirals distribution

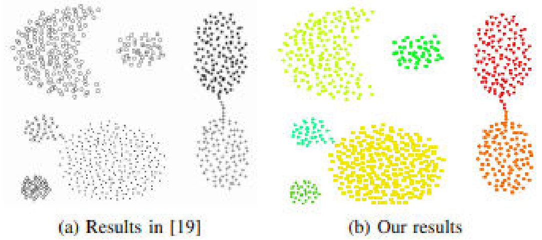
(a) Results in [19] (b) Our results

Fig. 6: Gionis *et al.* clustering aggregation results and our algorithm clustering results for the clustering aggregation problem

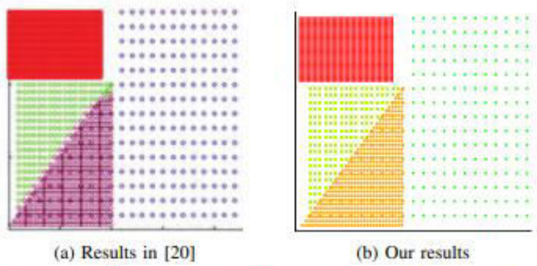
(a) Results in [20] (b) Our results

Fig. 7: Results on varied-density clusters with no separation among them.

The multi-density scheme that we propose helps to better shape clusters that share frontiers. Figure 7 shows that cluster borders are better shape in our results (right) that in the original results of [20] (left). To achieve this, our algorithm finds all possible clusters, and then it analyzes every μ-cluster in the border of the clusters. These μ-clusters are assigned to the connected cluster that has the most similar average density. In that way clusters frontiers can be precisely drawn which is key in highly overlapping distributions?

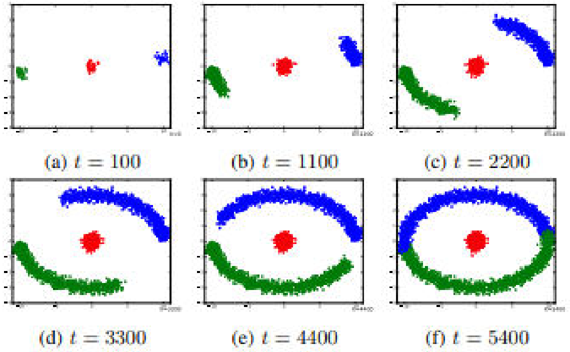
(a) $t = 100$ (b) $t = 1100$ (c) $t = 2200$
(d) $t = 3300$ (e) $t = 4400$ (f) $t = 5400$

Fig. 8: Concept drift toy example

Snapshots showing the distribution of μ-clusters (little boxes) and clusters (same color) found by our algorithm at several time instants are depicted in Figure 9. It can be seen how clusters evolution is tracked thanks to the drift of some of the existent μ-clusters and to the creation of new μ-clusters. Growth in the amount of clusters is particularly visible between snapshots one and two, and again between snapshots two and three. Oμ-clusters are represented as gray boxes.

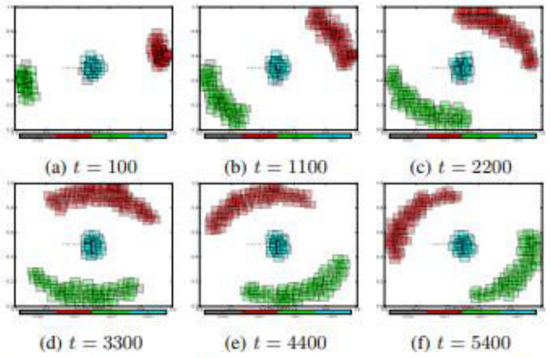
(a) $t = 100$ (b) $t = 1100$ (c) $t = 2200$
(d) $t = 3300$ (e) $t = 4400$ (f) $t = 5400$

Fig. 9: μ-clusters (little boxes) and clusters (same color) distribution obtained with our algorithm for the concept drift toy example

IV. DISCUSSION:

The toy data sets and the generated concept drift data set were selected upon their ability to show complex problems that may arise in real life. Path based clustering, for example, recreates the intuitive reasoning that two objects should be assigned to the same cluster if they can be connected by a mediating path of objects. However, in the best of our knowledge path based clustering is not achieved in dynamic clustering algorithms (stream based or not). The clustering aggregation problem and the multi-density problem test the ability of clustering overlapping distributions in which densities may vary, as is the case of industrial processes where samples of normal behavior will be more frequent that samples of faulty behavior. Quick detection of faulty behavior and drift of the normal state are also essential. To test concept drift, an artificial data set was created because, as stated in [21] and [22], an influential problem in most of the real-world data sets is that concept drift manifest over very long periods of time and hence results in huge data sets that are not freely available. For the test shown in the previous section, the forgetting processes were disabled in the static data sets, thus, only the concept drift experiment was subject to this process. We have shown that our algorithm can deal with these complex problems and in this section we also show that well known stream algorithms cannot handle them. We used the implementations of Clustream [10] and DenStream [12] available in the Massive online analysis open source framework [23].

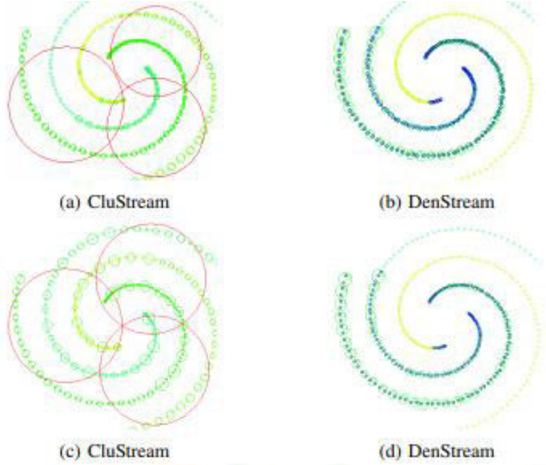
(a) CluStream (b) DenStream
(c) CluStream (d) DenStream

Fig. 10: Clustering results of streaming algorithms for the path-based test case. Screenshot from MOA software [23]. Neither of these algorithm achieve correct classification. Time horizon was set to 100 samples (a and b) and 300 samples (c and d)

TABLE I: Clustering evaluation of streaming methods over the path-based test case

Algorithm	Purity	Precision	Recall	NumClusters
Clustream	0.43	1.0	0.82	3
Denstream	1.0	1.0	0.6	32
Our proposal	1.0	1.0	1.0	3

DenStream, on the other hand, does not mix elements of several classes, but it creates 32 clusters putting apart samples of the same. The second scenario is the one giving the same importance to all the samples, this scenario was also the test scenario for our algorithm. Clustering results of CluStream and DenStream for this scenario are shown in the bottom of Figure 10. We can see that both algorithms fail again in finding the correct clustering. DenStream results do not change between both scenarios. Table I summarizes the clustering results for the tested algorithms. As second example, we analyze the streaming algorithm results in the case of multi-density distributions. CluStream and DenStream results for the test case introduced in [20] are shown in figure 11. Once again the tested algorithms fail to detect the data classes correctly. DenStream results are similar to those achieved by DBSCAN shown in Figure 2, that is, it can only detect the two denser classes. Even if in this example the target classes are convex sets, CluStream fails to correctly cluster them due mainly to the overlapping of the distributions. Table II summarize the clustering results.

V. CONCLUSIONS:

In this paper, a novel algorithm for dynamic clustering, its properties and performance have been explored. The performance of the algorithm is assessed theoretically through complexity analysis and empirically through a set of comparative experiments using popular data sets. The presented dynamic clustering method bases its structure on micro clusters allowing the handle of non-convex, multi-density clustering with outlier rejection even in highly overlapping situations. This approach to unsupervised learning implements a local-density analysis that allows detecting rare, infrequent behaviors improving novelty detection. This is possible since, with only a few objects, new classes can be characterized and recognized. The algorithm was compared with several different clustering algorithms, both static and dynamic, and it has proved to achieve similar or better results than several of them, hence pushing forward the state of the art. Even more, this proposal has proved to exceed the performance of well-known distance-based and density-based streaming algorithms. The dynamic clustering approach presented in this paper has shown excellent results in presence of concept drift. Future works will include dynamic clustering of dependent and auto correlated time series, investigating a proper representation of temporal information, and achieving dynamic clustering of multiple time-scale changing patterns.

VI. REFERENCES:

[1] A. Joentgen, L. Mikenina, R. Weber, and H.Zimmermann, “Dynamic fuzzy data analysis based on similarity between functions,” Fuzzy Sets and Systems, vol. 105, no. 1, pp. 81–90, 1999.

[2] M. Markou and S. Singh, “Novelty detection: a review–part 1:statistical approaches and part 2: neural network based approaches,” Signal processing, vol. 83, no. 12, pp. 2481–2497, 2003.

[3] J. Gama, I. Zliobait ˇ e, A. Bifet, M. Pechenizkiy, and A. Bouchachia, “A ˙ survey on concept drift adaptation,” ACM Computing Surveys (CSUR), vol. 46, no. 4, p. 44, 2014.

[4] L. Angstenberger, Dynamic fuzzy pattern recognition with applications to finance and engineering. Springer, 2001.

[5] P. Angelov and X. Zhou, “Evolving fuzzy-rule-based classifiers from data streams,” Fuzzy Systems, IEEE Transactions on, vol. 16, no. 6, pp.

[6] P. Angelov, “Fuzzily connected multimodel systems evolving autonomously from data streams,” Systems, Man, and Cybernetics, PartB: Cybernetics, IEEE Transactions on, vol. 41, no. 4, pp. 898–910,2011.

[7] K. Li, F. Yao, and R. Liu, “An online clustering algorithm,” in Fuzzy Systems and

Knowledge Discovery (FSKD), 2011 Eighth International Conference on, vol. 2. IEEE, 2011, pp. 1104–1108.

[8] T. Kempowsky, A. Subias, and J. Aguilar-Martin,
"Process situation assessment: From a fuzzy partition to a finite state machine," Engineering Applications of Artificial Intelligence, vol. 19, no. 5, pp. 461–477, 2006.

[9] A. Bouchachia and C. Vanaret, "Incremental learning based on growing gaussian mixture models," in Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on, vol. 2, IEEE. Elsevier, 2011, pp. 47–52.

[10] 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. VLDB Endowment, 2003, pp. 81–92.

[11] P. Kranen, I. Assent, C. Baldauf, and T. Seidl, "The ClusTree: indexing micro-clusters for any stream mining," Knoledge and information systems, vol. 29, no. 2, pp. 249–272, 2011.

[12] F. Cao, M. Ester, W. Qian, and A. Zhou, "Density-based clustering over an evolving data stream with noise," in SDM, 2006.

[13] Y. Chen and L. Tu, "Density-based clustering for real-time stream data,"
in Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and data mining, 2007,
pp. 133–142.

[14] N. Barbosa, L. T. Massuyes, and V. H. Grisales, "A data-based dynamic classification technique: A two-stage density approach," in SAFEPROCESS 2015, Proceedings of the 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes. IFAC, 2015,
pp. 1224–1231.

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

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion,O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss,
V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[17] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: A new data clustering
algorithm and its applications," Data Mining and Knowledge Discovery, vol. 1, no. 2, pp. 141–182, 1997.

[18] H. Chang and D.-Y. Yeung, "Robust path-based spectral clustering,"
Pattern Recognition, vol. 41, no. 1,
pp. 191–203, 2008.

[19] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering aggregation,"
ACM Trans. Knowl. Discov. Data,
vol. 1, no. 1, march 2007. [Online].
Available:
http://doi.acm.org/10.1145/1217299.1217303

[20] A. Fahim, A.-E. Salem, F. Torkey, M. Ramadan, G. Saake et al.,"Scalable varied

density clustering algorithm for large datasets," Journalof Software Engineering and Applications, vol. 3, no. 06, p. 593, 2010.

[21] R. Klinkenberg, "Learning drifting concepts: Example selection vs. example weighting," Intelligent Data Analysis, vol. 8, no. 3, pp. 281– 300, 2004.

[22] A. Tsymbal, M. Pechenizkiy, P. Cunningham, and S. Puuronen, "Dynamic integration of classifiers for handling concept drift," Information fusion, vol. 9, no. 1, pp. 56–68, 2008.

[23] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, "MOA: massive online analysis," Journal of Machine Learning Research, vol. 11, pp. 1601–1604, 2010. [Online]. Available:http://portal.acm.org/citation.cfm?id=185903