Research Article

PxEBCA: Proximity Expansion Based Clustering Algorithm

Bhumika S. Arora[1], Dr.Vijay Chavda[2], Dr Bhadresh R. Pandya[3]

[1]Research Scholar, KadiSarva Vishwavidyalaya, Gandhinagar, Gujarat, India
[2]Principal, N.P College of Computer Studies and Management, Kadi, Gujarat, India
[3]Head of Department, M.Sc.IT& B.Sc.CS, Gandhinagar, Gujarat, India
[1]bhumika.mca@gmail.com, [2]dr.vijaychavda@gmail.com, [3]prof_brpandya@yahoo.in

Article History: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

Abstract: Cluster analysis is one of the main techniques for analysing data. It is a technique for detecting groups of objects which are similar without specifying any criteria for the grouping. The matter of detecting clusters is challenging when the clusters are of varied size, density and shape. DBSCAN can find arbitrary shaped clusters along with outliers but it cannot handle different density. This paper presents a new method for detecting density based clusters which works on datasets having varied density. In this paper we propose PxEBCA that discovers clusters with arbitrary shape and also with varying density.Experimental evaluation of the effectiveness and efficiency of PEBCA was done using synthetic data. The results of experiments demonstrated that PxEBCA is significantly more effective in discovering clusters of arbitrary shapes with varying densities.

Keywords: Density based cluster, Proximity, Inter Cluster, Intra Cluster, Distance.

1. Introduction:

Now a Days, greater measures of data are gathered and put away in databases expanding the necessity for efficient and effective investigation techniques to utilize the data containing implicit patterns or groupings. Data analysis task of cluster analysis is expected to assist a user with understanding the natural grouping or design in a data set.

The goal of clustering is to identify group of objects which are more similar within the group than the objects of two different groups.

The Traditional Clustering algorithms are only suitable for small amount of datasets. As the volume of data is increasing regularly, so the clustering process becomes difficult and the result become unreliable. As a result, many clustering algorithms have trouble with increased large amount of data since it can reduce the accuracy rate and produce poor quality clusters.

There are interconnected reasons in determining the effectivity about clustering algorithms and relevant problems. First, practically all clustering algorithms require values as input parameters which are difficult to decide, particularly for real world datasets with high dimensionality. Second, the algorithms are absolutely sensible to these values ofinput parameters, often producing entirely exceptional partitioning data set even for somewhat unique boundary settings.Third, high-dimensional data sets frequently have skewed samples that can't uncovered by way of a clustering algorithm using just global parameter setting. Forth, since clustering algorithms include a number of parameters, regularly work in high dimensional spaces, and have to cope with noisy, inadequate and sample data, their overall performance can change considerably for various applications and types of data.

Density Based clustering techniqueDBSCAN discovers clusters of arbitrary shapes but when the dataset contains data with different densities it is not that efficient as its density based definition of core points cannot identify the core points of varying density clusters.

In this paper, we introduce a new algorithm based on Proximity Expansion. It is a bottom up approach. It discovers clusters with arbitrary shape and also with varying density. Algorithms logically divided into two parts. First part is forming Micro Clusters. Second part is Merging of Micro Clusters. Objects belonging to the micro clusters having less than 3 objects are considered as noise. In our experiments, we have used Euclidean distance, but the algorithm works on different distance methods like Manhattan, Correlation, Minkowski and Eisen.

The rest of the paper is arranged as follows. section 2 briefly discusseson clustering algorithm. In section 3, new algorithm is demonstrated. In section 4 discussion of performance Evaluation with different data sets. Section 5concludes the paper with a summary.

2. Clustering Algorithms:

Usually used strategies for Clustering are Hierarchical, Partitioning, Density Based, Grid based and Graph based. This section summarizes the numerous proposed versionsof DBSCAN including their research contributions and limitations.

The basic idea for the algorithm DBSCAN [1] works based on two parameters viz. 1) Eps: Specifies radius of neighborhood around data point p and 2) minPts: Specifies number of minimum data points in the neighborhood to identify it as a cluster. Using these two values, the algorithm discovers clusters by using concept of density reachability and connectivity. Since density reachability is non-linear, this algorithm can discover clusters with different shapes. All data points of the data set are categorized into i) Core points ii) Border points, and iii) Noise points.

DBSCAN algorithm has got a limitation that it is not able to find clusters with varying density.

In [2] OPTICS (Ordering Points to Identify the Clustering Structure) is an improved method upon DBSCAN, which uses random values for Eps which are used to identify clusters by generalizing technique of DBSCAN. This algorithm find clusters with varying density. It calculates outlier score for each point considering distance from its closest point.

Future research on this can be based on improving efficiency to support hyper sphere range queries for high-dimensional spaces having no index structures and also at discovering information of clusters in sparse datasets though it is good at finding them in dense areas.

In DENCLUE [3] works based on the Kernel Density Estimation technique which is aimed to find dense region ns. It was mainly developed to classify large multimedia databases which have high-dimensional data and contain large amount of noise.

In [4] proposed a new density-based clustering algorithm which enhances DBSCAN by first partitioning the dataset using CLARANS to reduce the search space to each segment instead of scanning the entire dataset, which improves the accelerate factor over the first DBSCAN algorithm.

VDBSCAN [5] arrives at the value of parameter Eps and MinPts by finding distance of k^{th} nearest neighbor of the point, after which it finds sharp change in the distance, which is called k-dist. This allows it to come out with different partitions of thegiven data sets and with multiple Eps values. Using which it generates multiple clusters with different densities.

The challenge here is that the magnitude of impact for finding k-dist depends fully on the characteristic of the data set.

ST-DBSCAN [6] is an improved version of density-based clustering algorithm, which has the capacity of finding groups as indicated by non-spatial, spatial, and temporal values of the objects. It is intended to run the algorithm in parallel in order to improve the performance. In addition, more useful heuristics may be found to decide the input parameters Eps and MinPts.

The BRIDGE algorithm [7] consolidates proceduresof K-means and DBSCAN algorithms to overcome limitations of each other. It empowers DBSCAN to deal with extremelylarge data whereas it also removes noisy points by improving proceduresof K-means. Itperforms K-means first and then density based clustering. It helps setting density threshold parameter properly. This approach makes it faster and computationally cost effective.

In Density based clustering methods allow the identification of arbitrary, not necessarily convex regions of data points that are densely populated. The number of clusters does not need to be specified beforehand; a cluster is defined to be a connected region that exceeds a given density threshold. The LSDBC algorithm [8] works on the technique of local scaling, has got two input parameters, 1) k – used to order points according to their distance to their kth neighbor and 2) α – used to determine the boundary of the current cluster expansion based on its density. The local scaling technique separate clusters using local statistics of the points. These parameters help to know how dense the region is around each point. Beginning with higher density regions, it connects points of dense regions until the density fall below the threshold.

The UDBSCAN algorithm [9] is designed to work on uncertain objects. Uncertain objects are the one which have certain attributes whose precise value can't be defined. This may be due to various reasons including data acquisition or property of the object itself. The U-DBSCAN algorithm uses a deviation function that approximates value of such attribute and creates clusters using an associated probability density function.

3. PxEBCA: Proximity Expansion Based Clustering Algorithm

The key concept of the [PxEBCA] is the expansion of neighbourhood by applying (based on the) proximity expansion parameter to the distance in reference. For example objects A1 and A2 are two closest objects in the given space, an object P is in its proximity if either dist(A1,P) <= dist(A1, A2) * PEP or dist(A2,P) <= dist(A1,A2) * PEP, where proximity expansion parameter PEP will have a value greater than 1.

The algorithm works in two steps, viz. 1) formation of micro clusters and 2) merging of micro clusters.

Formation of micro cluster starts by considering two closest objects of the dataset as micro cluster and adding an object to the micro cluster if it is in its proximity. In an iterative process an object P is added to the micro cluster C = {A1,A2,A3,…..An} if min(dist(P,Ai)) <= max(dist(Aj,Ak)) * PEP [i= 1…n, j,k = 1..n, j != k]. This process forms micro clusters having at least two or more objects.

Objects belonging to the micro clusters having less than 3 objects are considered noise. These are objects merged to a cluster if found in the proximity of any cluster during merging process. The objects which do not belong to any cluster at the end of the process are considered outliers.

In the step-2, merging of clusters is done based on intra-cluster distance and inter cluster distance. In the merging process, starting with two closest clusters, they are merged if they are in the proximity. In this process, objects identified as noise are also added to cluster if they are within the proximity. Iterative process of merging stops when no merging of clusters happen in an iteration.

Object in proximity: An object P is in proximity of a cluster C = {A1,A2,A3,…..An} if min(dist(P,Ai)) <= max(dist(Aj,Ak)) * PEP [i= 1…n, j,k = 1..n, j != k].

Intra cluster distance: (Average distance within the cluster)

Intra cluster distance of the cluster C is average distance between all the objects of a cluster under consideration.

IntraClustDist(C) = ∑ (dist(Ai,Aj)) / ∑ k [i, j: 1..n, i< j ; k = 1..n-1]

Inter cluster distance:

Inter cluster distance is the distance between two nearest objects of the two clusters under consideration.

InterClustDist(C1,C2) = min(dist(A1i,A2j)) [i = 1..n, j = 1..m],

Clusters in proximity: Two clusters C1 and C2 are in proximity if InterClustDist(C1,C2) <= IntraClustDist(C1) * PEP or InterClustDist(C1,C2) <= IntraClustDist(C2) * PEP.

Algorithm Steps:

Algorithm logically divided in two parts,

1. Forming Micro Clusters
2. Merging of Micro Clusters

Part – 1 Forming Micro Clusters:

Step-1: Prepare Distance Matrix which calculates distance between each object from all other objects.

Step-2: Read (next) two closest objects from the distance matrix.

Step-3: (i) If any of the two objects belong to a micro cluster, other object is added to it if it is in proximity of the micro cluster else it considered as noise.

(ii) If none of the two objects belong to any micro cluster, these two objects are considered new micro cluster.

(iii) If both the objects belong to micro clusters, the pair is skipped.

The iterative process executes step-2 and step-3 till all the objects are read.

Part- 2 Merging of Clusters:

Step-1: Two closest clusters are read

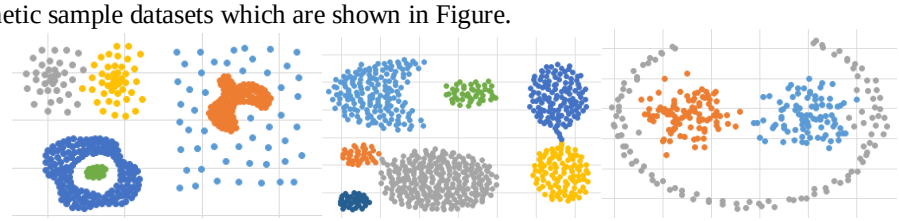
Step-2: These two clusters are merged if they are in the proximity.

Step-3: The iterative process executes step-1 and step-2 till no merging of clusters happen in an iteration.

Step-4: Objects considered as noise are merged to the nearest cluster if it is in its proximity.

4. Performance Evaluation:

In this section, the performances of algorithm are evaluated by using the 2-Dimensional synthetic dataset. We use four synthetic sample datasets which are shown in Figure.



Dataset1 Dataset2 Dataset3

Figure1: Sample Dataset

Sample database 1 has six different shaped clusters with significantly differing sizes. Sample database 2 has seven clusters of ball shape. Sample database 3 has three clusters of different shapes. Sample database 4 hasdata with four dimensions.

Table 1 shows Performance Comparison of different algorithms in terms of Rand Index.

Rand Index for Datasets				
Dataset	**DBSCAN**	**PxE2.0**	**PxE1.5**	**PxE2.5**
Dataset1	0.927	0.927	0.98	-
Dataset2	0.941	0.946	0.955	0.929
Dataset3	0.714	0.723	0.656	-
Dataset4	0.772	0.756	0.705	0.79

Table 2 shows Performance Comparison of different algorithms in terms of Dunn Index.

Dunn Index for Datasets				
Dataset	**DBSCAN**	**PxE2.0**	**PxE1.5**	**PxE2.5**
Dataset1	0.107	0.107	0.024	-
Dataset2	0.125	0.102	0.1	0.062
Dataset3	0.022	0.036	-	0.033
Dataset4	0.042	0.052	0.059	0.045

Table 3 shows Performance Comparison of different algorithms in terms of Error Rate.

Error Rate for Datasets				
Dataset	**DBSCAN**	**PxE2.0**	**PxE1.5**	**PxE2.5**
Dataset1	0.073	0.073	0.02	-
Dataset2	0.059	0.054	0.045	0.071
Dataset3	0.186	0.277	0.344	-
Dataset4	0.23	0.24	0.29	0.18

Table 4 shows Performance Comparison of different algorithms in terms of F-Measure (P=precision, R= recall, F=F-measure)

F-measure for Datasets												
Dataset	**DBSCAN**			**PxE2.0**			**PxE1.5**			**PxE2.5**		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Dataset1	0.748	0.997	0.855	0.749	1	0.856	0.958	0.956	0.957	-	-	-
Dataset2	0.833	0.954	0.889	0.86	0.933	0.895	0.89	0.935	0.912	0.807	0.94	0.868
Dataset3	0.704	0.761	0.731	0.829	0.21	0.335	0.48	0.41	0.442	-	-	-
Dataset4	0.597	0.948	0.733	0.588	0.861	0.699	0.54	0.693	0.607	0.582	0.949	0.722

For comparingperformance of PxEBCA with DBSCAN four sample databases were used which are shown in figure 1. We have tested proposed algorithm with different Proximity Expansion Parameter for these databases. The Proposed algorithm works on both sparse and dense data. It is capable to handle the density variations that exist within the dataset. The clusters detected by the proposed algorithm are having considerable density

variation within clusters. From the content of the above mentioned tables, it has been observed that the Rand Index, Dunn Index, Error rate and F-Measure calculated gives most promising result.

5. Conclusion:

In this paper we presented a new clustering algorithm which overcomes challenge of density based clustering algorithms. In addition, our clustering approach works well for datasets with varying densities.This is achieved by using expansion of neighbourhood by applying proximity expansion parameter to the distance in reference.We did performance evaluation on synthetic data. Results of these experiments demonstrate that PxEBCA is significantly more effective in discovering clusters of arbitrary shape than the well-known algorithm DBSCAN.

References:

1. Martin Ester, Hans-Peter Kriegel, Jiirg Sander, Xiaowei Xu (1996), A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, Publication: KDD'96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Pages 226–231.
2. MihaelAnkerst, Markus M. Breunig, Hans-Peter Kriegel, JörgSander(1999), OPTICS: Ordering Points To Identify the Clustering Structure, Publication: ACM SIGMOD Record, 28(2), Pages 49–60.
3. Alexander Hinneburg1 and Hans-Henning Gabriel2(2007), DENCLUE 2.0: Fast Clustering Based on Kernel Density Estimation,Publication: International Symposium on Intelligent Data Analysis Advances in Intelligent Data Analysis VII, Pages 70-80.
4. El-Sonbaty, Y., Ismail, M. A., & Farouk, M. (n.d.) (2004), An efficient density based clustering algorithm for large databases. 16th IEEE International Conference on Tools with Artificial Intelligence.
5. Liu, P., Zhou, D., & Wu, N. (2007). VDBSCAN: Varied Density Based Spatial Clustering of Applications with Noise. International Conference on Service Systems and Service Management.
6. DeryaBirant *, Alp Kut (2007), ST-DBSCAN: An algorithm for clustering spatial–temporal data, Data & Knowledge Engineering 60 Pages 208–221.
7. Dash, M., Liu, H., &Xiaowei Xu. (2001). "1+1>2": merging distance and density based clustering. Proceedings Seventh International Conference on Database Systems for Advanced Applications, Pages 32-39.
8. Ergun Bic¸ici and Deniz Yuret (2007), Locally Scaled Density Based Clustering, ICANNGA, Part I, LNCS 4431, Pages 739–748.
9. Tepwankul, A., &Maneewongwattana, S. (2010). U-DBSCAN : A density-based clustering algorithm for uncertain objects. IEEE 26th International Conference on Data Engineering Workshops. Pages 136-143.
10. Yin, J., Fan, X., Chen, Y., & Ren, J. (2005). High-Dimensional Shared Nearest Neighbor Clustering Algorithm. Lecture Notes in Computer Science, Pages 494–502.
11. LeventErtoz, M. Steinbach, V. Kumar (2002), A New Shared Nearest Neighbor Clustering Algorithm and its Applications, Computer Science.
12. Ertöz, L., Steinbach, M., & Kumar, V. (2003), Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data. Proceedings of the 2003 SIAM International Conference on Data Mining, Pages 47–58.
13. Borah, B., & Bhattacharyya, D. K. (n.d.)(2004), An improved sampling-based DBSCAN for large spatial databases. International Conference on Intelligent Sensing and Information Processing. Pages 92-96.
14. Smiti, A., &Eloudi, Z. (2013). Soft DBSCAN: Improving DBSCAN clustering method using fuzzy set theory. 6th International Conference on Human System Interactions,IEEE. Pages 380-385.
15. Santhisree, K., &Damodaram, A. (2011). CLIQUE: Clustering based on density on web usage data: Experiments and test results.3rd International Conference on Electronics Computer Technology IEEE. Pages 233-236.
16. Rehioui, H., Idrissi, A., Abourezq, M., &Zegrari, F. (2016). DENCLUE-IM: A New Approach for Big Data Clustering. Procedia Computer Science, Elsevier, Pages 560–567.
17. Chen, H. (2013). Density-accumulated arbitrary shaped clustering for large data sets. 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation, IEEE, Pages 1088-1092.
18. Shah, G. H. (2012). An improved DBSCAN, a density based clustering algorithm with parameter selection for high dimensional data sets. Nirma University International Conference on Engineering, IEEE.