Fast Density Based Clustering Algorithm

Priyanka Trikha and Singh Vijendra

***Abstract*—Clustering problem is an unsupervised learning problem. It is a procedure that partition data objects into matching clusters. The data objects in the same cluster are quite similar to each other and dissimilar in the other clusters. The traditional algorithms do not meet the latest multiple requirements simultaneously for objects. Density-based clustering algorithms find clusters based on density of data points in a region. DBSCAN algorithm is one of the density-based clustering algorithms. It can discover clusters with arbitrary shapes and only requires two input parameters.In this paper, we propose a new algorithm based on DBSCAN. We design a new method for automatic parameters generation that create clusters with different densities and generates arbitrary shaped clusters. The kd-tree is used for increasing the memory efficiency. The performance of proposed algorithm is compared with DBSCAN. Experimental results indicate the superiority of proposed algorithm.**

***Index Terms*—Clustering algorithm, Kd-tree, density based algorithm.**

I. Introduction

Clustering is one of the major data mining tasks and aims at grouping the data objects into meaningful classes (clusters) such that the similarity of objects within clusters is maximized, and the similarity of objects from different clusters is minimized [1]. Cluster analysis is one of the main tools for exploring the underlying structure of a data set.Several application domains such as molecular biology, medical applications like breast cancer and geography produce a tremendous amount of data which can no longer be managed without the help of efficient and effective data mining methods. There is an ever increasing need for efficient and effective data mining methods to make use of the information contained implicitly in that data. One of the primary data mining tasks is clustering, which is intended to help a user discovering and understanding the natural structure or grouping in a data set. In particular, clustering is the task of partitioning objects of a data set into distinct groups (clusters) such that two objects from one cluster are similar to each other, whereas two objects from distinct clusters are not. However, traditional clustering algorithms often fail to meet the latest multiple requirements simultaneously. The k-means [1] method is the standard clustering algorithm, enjoys widespread use. The method partitions the data into k clusters, where the k is supplied by the user. DBSCAN (Density Based Spatial Clustering of Applications with Noise) relying on a density-based notion of clusters which is designed to discover clusters of arbitrary shape. DBSCAN [2] requires only one input parameter and supports the user in determining an appropriate value for it. DBSCAN is significantly more effective in discovering clusters of arbitrary shape. DBCLASD [3] is another density-based clustering algorithm, but unlike DBSCAN, the algorithm assumes that the points inside each cluster are uniformly distributed. CLIQUE [4] has been designed to find clusters embedded in subspaces of high dimensional data without requiring the user to guess subspaces that might have interesting clusters. CLIQUE generates cluster descriptions in the form of DNF expressions that are minimized for ease of comprehension. OPTICS is an algorithm for finding density-based clusters in spatial data [5].Its basic idea is similar to DBSCAN but it addresses one of DBSCAN major weaknesses: the problem of detecting meaningful clusters in data of varying density. In order to do so, the points of the database are (linearly) ordered such that points which are spatially closest become neighbors in the ordering. Additionally, a special distance is stored for each point that represents the density that needs to be accepted for a cluster in order to have both points belong to the same cluster. The DENCLUE algorithm employs a cluster model based on kernel density estimation [6]. A cluster is defined by a local maximum of the estimated density function. Data points are assigned to clusters by hill climbing, i.e. points going to the same local maximum are put into the same cluster. Rough-DBSCAN [8] is a Density based clustering techniques like DBSCAN are attractive because it can find arbitrary shaped clusters along with noisy outliers. MITOSIS [10] finds arbitrary shapes of arbitrary densities in high dimensional data. Unlike previous algorithms, this algorithm uses a dynamic model that combines both local and global distance measures. The algorithm's ability to distinguish arbitrary densities in a complexity of order $O(D_n \log_2(n))$ renders it attractive to use. Validity indexes indicate that Mitosis out performs related algorithms as DBSCAN [2], which finds clusters of arbitrary shapes. In DENCOS [9] different density thresholds is utilized to discover the clusters in different subspace cardinalities to cop up with density divergence problem. Here the dense unit discovery is performed by utilizing a novel data structure DFP-tree (Density FP-tree), which is constructed on the data set to store the complete information of the dense units. As validated by our extensive experiments on various data sets, DENCOS can discover the clusters in all subspaces with high quality, and the efficiency of DENCOS outperforms previous works. PACA- DBSCAN [11] is based on partitioning-based DBSCAN and modified ant clustering algorithms. It can partition database into N partitions according to the density

Manuscript received September 10, 2012; revised November 30, 2012.

Priyanka Trikha is with the Department of Computer Science and Engineering, SBCET, Jaipur, Rajasthan, India (e-mail: trikha_priyanka@yahoo.com).

Vijendra Singh is with the Department of Computer Science and Engineering,Faculty of Engineering and Technology, Mody Institute of Technology and Science ,Lakshmangarh, Sikar, Rajasthan, India.

of data, then cluster each partition with DBSCAN [2]. Superior to DBSCAN, The new hybrid algorithm reduces the sensitivity to the initial parameters, and can deal with data of uneven density very well. APSCAN [12] is a parameter free clustering algorithm. The APSCAN has two advantages: first it does not need to predefine the two parameters as required in DBSCAN [2] and second, it not only can cluster datasets with varying densities but also preserve the nonlinear data structure for such datasets.

This paper is organized as follows. Density based definitions are given in Section 2. In Section 3 Proposed Fast Density Based Algorithm, is explained. Section 4 contains data description and result analysis. Finally, we conclude in Section 5.

II. Density Based Definitions for Clusters

Definition 1: (directly density-reachable) *A* point *p* is *directly density-reachable* from a point q wrt. ε, MinPts if

1) $p \in \varepsilon(q)$ and

2) $|N_{\varepsilon}(q)| \geq$ MinPts (core point condition).

Definition 2: (density-reachable) A point p is *density-reachable* from a point q wrt. ε and MinPts if there is a chain of points $p_1, \ldots, p_n, p_1 = q, p_n = p$ such that p_{i+1} is directly density-reachable from p_i.

Definition 3: (density-connected) *A* point *p* is *density-connected* to a point q wrt. ε and MinPts ,if there is a point o such that both, p and q are density-reachable from o wrt. ε and MinPts.

Definition 4: (cluster) Let *D* be a database of points. A *cluster C* wrt. ε and MinPts is a non-empty subset of *D* satisfying the following conditions:

1) If $p \in C$ and *q* is density-reachable from p wrt. ε and MinPts, then $q \in C$. (Maximality).

2) If $p, q \in C$: *p* is density-connected to q wrt. ε and MinPts. (Connectivity)

Definition 5: (noise) Let $C_1, \ldots, C_k$ be the clusters of the database *D* wrt. Parameters ε and MinPts_i, $i = 1, \ldots k$. Then we define the *noise* as the set of points in the database *D* not belonging to any cluster C_i, i.e. noise = $\{p \in D \mid p \notin C_i\}$.

III. Proposed Density Based Clustering Algorithm

A. Clustering Problem

Clustering is a formal study of algorithms and methods for classifying objects without category labels. A cluster is a set of objects that are similar to each-other, and objects from different clusters are not similar. The set of n objects $X=\{X_1, X_2 \ldots, X_n\}$ is to be clustered. Each $X \in R^p$ is an attribute vector consisting of p real measurements describing the object. The objects are to be clustered into non overlapping groups $C = \{C_1, C_2 \ldots C_k\}$ (C is known as a clustering), where k is the number of clusters, $C_1 \cup C_2 \cup \ldots \cup C_k = X$, $C_i \neq \phi$ and $C_i \cap C_j = \phi$ for $i \neq j$.

B. Kd-Tree

Kd-tree is a space-partitioning data structure for organizing points in a *K*-dimensional space [7]. A Kd-tree uses only those splitting planes that are perpendicular to one of coordinate axes. In the nearest neighbor problem a set of data points in *d*-dimensional space is given. These points are preprocessed into a data structure, so that given any query point *q*; the nearest or generally *k* nearest points of *p* to *q* can be reported efficiently.

C. Determining the Parameters ε and Min Pts

The dynamic method enables two input modes; the automatically generated parameters vary according to different inputs.

1) The distance between two objects

The distance function *d (a, b)* measures the dissimilarity of two objects

$$d(a,b) = q\sqrt{\left(\mu_1 |X_{a1} - X_{b1}|^q + \mu_2 |X_{a2} - X_{b2}|^q + \ldots\ldots \pm \mu_p |X_{ap} - X_{bp}|^q\right)} \quad (1)$$

where $a = (x_{a1}, xa2, \ldots, x_{ap})$ and $b = (x_{b1}, x_{b2}, \ldots, x_{bp})$ are two p-dimensional objects, and q is a positive integer.

2) Automatic parameters generation dynamic method

The proposed algorithm discovers clusters with different densities, by generating multiple pairs of ε and *MinPts* automatically.

1. Create a kd-tree for the given data x_i, $i = 1, \ldots n$.
2. For i=1 to total no. of cells
   Calculate distance between each pair of objects in C_i
   If cell does not contain any input objects, then calculate ε_i and *Minpts*
   Else
   If user submit *n* objects as input then
   Calculate ε_i and $Minpts_i$
   End if
   End for
3. Update ε_i and $Minpts_i$
4. For i=1 to NP
   Generate clusters and calculate priority of Each pair
   End for
5. For i=1 to *NP*
   Call *DBSCAN* (ε_i, $Minpts_i$,set of objects)
   End for

Fig. 1. (a) FDBA algorithm.

3) FDBA algorithm

IV. Experimental Results

We tested proposed Fast Density Based Algorithm (FDBA) using several synthetic data sets and real data sets. All experiments were run on a PC with a 2.0GHz processor and 1GB RAM. The synthetic data sets are generated by using the data generation method. The synthetic data sets dimensionalities increased from 2 to 20 and size increased from 250 to 5000. Five real data sets from UCI machine learning repository are also utilized to evaluate the clustering

result of FDBA. These real data sets are the adult data, Iris data, breast cancer, Pima data and the thyroid disease data.

The clustering results of DBSCAN and FDBA algorithm is shown in Fig. 2 and Fig. 3 for synthetic data set. This is a 5-dimensional data set consists of three clusters of 300 data points.

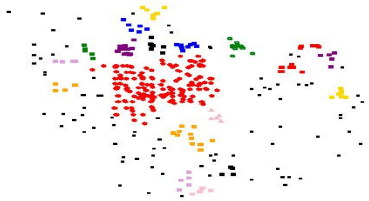

Fig. 2. DBSCAN.

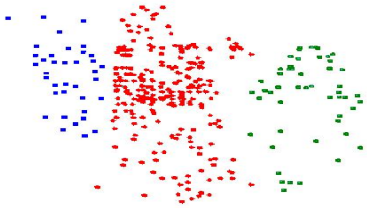

Fig. 3. Proposed algorithm (FDBA).

V. Conclusion

In this paper we present a new algorithm FDBA based on DBSCAN with a new method for automatic parameters generation. We employ artificial data sets and real data sets to prove the performance of our new proposed algorithm. The results of proposed FDBA are evaluated and compared. The experiment has proved that the performance of proposed algorithm is better than DBSCAN.

References

[1] L. Kaufman and P. J. Rousseeuw, *Finding groups in data: An introduction to cluster analysis*, John Wiley and Sons, Inc., New York, 1990.

[2] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Second Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD)*, 1996.

[3] X. W. Xu, M. Ester; H.-P. Kriegel, and J. Sander, "A distribution-based clustering algorithm for mining in large spatial databases," in *Proc. 14th Internat. Conf. on Data Eng. (ICDE98)*.Orlando, 1998, pp. 324–331.

[4] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. "Automatic subspace clustering of high dimensional data for data mining applications," in *Proc. ACM SIGMOD Int'l Conf,* 1998, pp. 51-70.

[5] M. Ankerst, M. M. Breunig, H. P. Kriegel, J. Sander, "OPTICS: ordering points to identify the clustering structure," in *Proc. ACM SIGMOD international conference on Management of data*, 1999. pp. 49–60.

[6] K. Kailing, H. P. Kriegel, and P. Kroger, "Density-Connected Subspace Clustering for High-Dimensional Data," in *Proc. Fourth SIAM Int'l Conf. Data Mining (SDM),* 2004.

[7] D. M. Mount and S. Arya, "ANN: A library for approximate nearest neighbor searching," 2005.

[8] P. Viswanath and V. S. Babu, "Rough -DBSCAN: A fast hybrid density based clustering method for large data sets," *Pattern Recognition Letters*, vol. 30, pp. 1477-1488, 2009.

[9] Y. H. Chu, J.W. Huang, K.T. Chuang, D.N. Yang, and M.S. Chen, "Density conscious subspace clustering for high-dimensional data," *IEEE Transactions on knowledge and data engineering*, vol. 22, pp. 16-30, 2010.

[10] N. A. Yousria, M.S. Kamel, and M.A. Ismail, "A distance-relatedness dynamic model for clustering high dimensional data of arbitrary shapes and densities," *Pattern Recognition*, pp.1193-1209, 2009.

[11] H. Jiang, J. Li, S. Yi, X. Wang, and X. Hu, "A new hybrid method based on partitioning-based DBSCAN and ant clustering," *Expert Systems with Applications*, vol. 38, no. 8, pp. 9373-9381, 2011.

[12] X. Chen, W. Liu, H. Qiu, and J. Lai, "APSCAN: A parameter free algorithm for clustering," *Pattern Recognition Letters,* vol. 32, Issue 7, pp. 973-986, 2011.

[13] S. Vijendra, **"**Efficient clustering for high dimensional data: subspace based clustering and density based clustering," *Information Technology Journal*, vol. 10, no. 6, pp. 1092-1105, 2011.

Priyanka Trikha has completed her B.Tech and M.Tech degree in Computer Science and Engineering from Faculty of Engineering and Technology, MITS, Lakshmangarh, Sikar, Rajasthan. Now She is assistant professor in SBCET, Jaipur, Inida.

Vijendra Singh received the M.Tech degree in Computer Science and Engineering from Birla Institute of Technology, Mesra (Ranchi), India. His research interests include Data Mining, Pattern Recognition, Evolutionary and Soft Computation and Bioinformatics. He is selected for Who's Who in Science and Engineering, (2011-2012). He is member of ISTE India, IEEE, IAENG Hong Kong, IACSIT Singapore, and AICIT Korea. He has programme committee member of IEEE conferences and word reputed international conferences. He authored more than 20 scientific papers in Data Mining, Pattern Recognition, Evolutionary and Soft Computation.