A Density based Algorithm for Discovering Density Varied Clusters in Large Spatial Databases

Anant Ram
G.L.A Institute of Technology & Management, Mathura, India

Sunita Jalal
G.B. Pant University of Agriculture & Technology, Pantnagar, India

Anand S. Jalal
G.L.A Institute of Technology & Management, Mathura, India

Manoj Kumar
G.L.A Institute of Technology and Management, Mathura, India

ABSTRACT
DBSCAN is a base algorithm for density based clustering. It can detect the clusters of different shapes and sizes from the large amount of data which contains noise and outliers. However, it is fail to handle the local density variation that exists within the cluster. In this paper, we propose a density varied DBSCAN algorithm which is capable to handle local density variation within the cluster. It calculates the growing cluster density mean and then the cluster density variance for any core object, which is supposed to be expended further, by considering density of its ε-neighborhood with respect to cluster density mean. If cluster density variance for a core object is less than or equal to a threshold value and also satisfying the cluster similarity index, then it will allow the core object for expansion. The experimental results show that the proposed clustering algorithm gives optimized results.

General Terms
Density Based Clustering

Keywords
Core object, Cluster density mean, Cluster density variance, Cluster Similarity Index, Density differs

1. INTRODUCTION
Clustering is an important data analysis task that tries to separate a collection of objects into reasonable homogeneous groups called cluster [1]. Clustering in data mining is a discovery progression that groups a set of data in such a way that the inter-cluster similarity is minimized and intra-cluster similarity is maximized .There are five different techniques for clustering i.e. Partitioning, Hierarchical, Density based, Grid based and Model based. In addition to above five one more clustering technique, known as subspace clustering technique, which tries to remove the irrelevant dimensions in high dimensional database to generate the good quality cluster, because the irrelevant dimensions hide the real facts in the high dimensional database [2]. Density based algorithms are more important to find out the clusters of different shapes and sizes. However, most of the density based clustering algorithms, are not able to handle the local density variation exist within the cluster in high dimensional database. DBSCAN [3], a pioneer density based clustering algorithms, also fail to detect the density varied clusters due to the global parameter 'minimum points'. There are a number of clustering algorithms exist as an improvement of DBSCAN for handling the density variation within the cluster.

In the proposed work, we present a new density varied algorithm which is successful to handle local density variation within the cluster. It calculates the growing cluster density mean and then the cluster density variance for any core object, which is supposed to be expended further, by considering density of its ε-neighborhood with respect to cluster density mean. If cluster density variance for a core object is less than or equal to a threshold value and also satisfying the cluster similarity index, then it will allow the core object for expansion.

Rest of the paper is organized as follows. Section 2 presents related work on density based clustering technique for high dimensional database. Section 3 discuses the existing DBSCAN clustering algorithm and required modification to get better clustering results. The proposed modification along with algorithm is discussed in section 4. Experimental results are presented in section 5. Finally, Section 6 presents conclusion and future work.

2. RELATED WORK
The DBSCAN (Density Based Spatial Clustering of Applications with Noise) [3] is a base algorithm of density based clustering. It requires user specified two global input parameters i.e. minimum objects (μ) and radius (ε).The density of an object is the number of objects in its ε-neighborhood of that object. DBSCAN does not specify upper limit of a core object i.e. how much objects may present in its ε-neighborhood. So due to this, the clusters detected by it, are having wide variation in local density. Such clusters may be represented by several smaller clusters so that each cluster may have reasonably uniform density. OPTICS [4] algorithm is an enhancement of DBSCAN to achieve this goal. In contrast to DBSCAN, OPTICS does not assign cluster memberships but computes an ordering in which the objects are processed and additionally generates the information, which would be used by an extended DBSCAN algorithm to assign cluster memberships. This information consists of two values for each object, the core-distance and the reachability-distance. The Valleys in reachability plot indicates clusters. The parameter ξ is crucial for identifying the valleys as ξ-clusters.

Another enhancement of the DBSCAN algorithm is DENCLUE [5], which is based on an influence function that describes the impact of an object upon its neighborhood. The algorithm allows a compact mathematical description of arbitrarily shaped clusters in high-dimensional data sets and is significantly faster as compared to the other density based clustering algorithms. It produces good clustering results even when a large amount of noise is present. EDBSCAN [6] algorithm is another improvement of DBSCAN; it keeps tracks of density variation which exists within the cluster. It calculates the density variance of a core object with respect to its ε-neighborhood. If density variance of a core object is less than or equal to a threshold value and also satisfying the homogeneity index with respect to

its ε-neighborhood then it will allow the core object for expansion. But it calculates the density variance and homogeneity index locally in the ε-neighborhood of a core object.

In [7] the author proposed a method known as DD_DBSCAN, which finds the clusters of different shapes, sizes which differ in local density. However, the method is unable to handle the density variation within the cluster, i.e. same cluster may have wide density variation from one end to another end of the cluster. In [8], a method DDSC(A Density Differentiated Spatial Clustering Technique) is proposed, which is again an extension of the DBSCAN algorithm. It detects clusters, which are having non-overlapped spatial regions with reasonable homogeneous density variations within them. If there is significant change in densities of adjacent regions then all are separated into different clusters. An added advantage is that the sensitivity of the input parameter ε, which is an important disadvantage of DBSCAN, is reduced significantly. In VDBSCAN [9] the author has also tried to improve the result using DBSCAN algorithm. The method computes k-distance for each object and sort them in ascending order, then plotted using the sorted values. The sharp change at the value of k-distance corresponds to a suitable value of ε. It divides the k-distance plot to identify the different values of ε to detect the different density varied clusters.

Thus, a good clustering method should allow a significant density variation within the cluster because, if we go for homogeneous clustering, a large number of smaller unimportant clusters may be generated. In this paper an enhancement of DBSCAN algorithm is proposed, which detects the clusters of different shapes, sizes which differ in local density in the presence of noise and outlier in data base.

3. DBSCAN ALGORITHM

DBSCAN (Density-Based Spatial Clustering of Application with noise) [3] is density based cluster formation algorithm for spatial and non spatial high dimensional data base in the presence of noise and outlier. The working is based on the following definitions, for more detail refer DBSCAN [3]:

Def.1: The ε-neighborhood of an object p, denoted by Nε (p), is defined as total number of objects lying in the radius ε, i.e. $N\varepsilon(p) = \{q \in D \mid dist(p, q) \leq \varepsilon\}$.

Def.2: An object p is said to be Core object if $|N\varepsilon(p)| \geq \mu$ (minimum objects).

Def.3: An object p is said to be directly density reachable from an object q with respect to ε and μ if $p \in N\varepsilon(q)$ and q is a Core object.

Def.4: An object p is said to density-reachable from an object q if there is a chain of objects $p_1 \ldots p_n$, $p_1 = q$, $p_n = p$ such that p_{i+1} is direct density-reachable from p_i with respect to ε and μ.

Def.5: An object p is said to density-connected to an object q with respect to ε and μ if there is an object o such that both, p and q are density reachable from o with respect to ε and μ.

Def.6: An object which is lying at the border is not a Core object, but it will be a part of cluster. An object which is not lying in any of the cluster is treated as a noise object.

Def.7: A cluster X is non empty subset of database with respect to ε and μ, for every p, q: if $p \in X$, q is density reachable from p then $q \in X$ and p is density-connected to q.

DBSCAN [3] detects density connected clusters by discovering one of its core object's p and computing all objects which are density-reachable from p. The collection of density-reachable objects is performed by iteratively computing directly density-reachable objects. DBSCAN checks the ε-neighborhood of each object p in the database. If Nε (p) of an object p consists of at least μ objects, i.e., if p is a core object, a new cluster X containing all objects of Nε (p) is created. Then, the ε-neighborhood of all objects $q \in X$, which have not yet been processed, is checked. If object q is also a core object, the neighbors of q, which are not already assigned to cluster X, are added to X and their ε-neighborhood is checked in the next step. This procedure is repeated until no new object can be added to the current cluster X.

4. THE PROPOSED ALGORITHM (DVBSCAN)

DBSCAN [3] is a pioneer density based clustering algorithms. However, it suffers from wide density variation within the clusters. To overcome this problem, a DVBSCAN (Density Variation Based Spatial Clustering of Applications with Noise) algorithm is proposed in this section. It is based on the concept that it starts the formation of the cluster by selecting core object. Then it computes the Cluster Density Mean(CDM) of the growing cluster before allowing the expansion of an unprocessed core object. After that it computes the Cluster Density Variance (CDV) including the ε-neighborhood of the unprocessed core object with respect to Cluster Density Mean (CDM). If the Cluster Density Variance (CDV) of the growing cluster with respect to CDM is less than a specified threshold value α and the difference between the minimum and maximum objects lying in the ε-neighborhood of the objects, which are the objects of the growing cluster, including the ε-neighborhood objects of the unprocessed core object, is less than a specified threshold value λ then only an unprocessed core object is allowed for expansion otherwise the object is simply added into the cluster.

In addition to DBSCAN [3] the following definitions are required in DVBSCAN (Density Variation Based Spatial Clustering of Applications with Noise) to allow the considerable density variation within the same cluster and wide density variation with other clusters.

Defnition1: (Cluster Density Mean): It is denoted by CDM(C).The Cluster Density Mean (CDM) of a growing cluster is defined as follows:

$$CDM(C) = \frac{\sum_{o \in C} |N_{\varepsilon}(o)|}{|C|} \quad (1)$$

Where the Nε (o) is the density of the object o around in the ε-neighborhood.

Defnition2: (Cluster Density Variance): Cluster Density Variance is denoted by CDV(C).The CDV of any growing cluster is calculated by including the ε-neighborhood of unprocessed core object x, before allowing the expansion of the unprocessed core object.

At any instant Let the total number of objects in the cluster C are K i.e. |C|= K and out of these objects if it select one of the unprocessed core object for expansion, say x then it calculate the ε-neighborhood of object x, and include these objects in the growing cluster i.e. P = C ∪ Nε (x).Then the Cluster density variance of the growing cluster is calculated as follows:

$$CDV(C) = \frac{\sum_{o \in P} \{|N_{\varepsilon}(o)| - CDM(C)\}^2}{|P|} \quad (2)$$

If the CDV(C) is less than specified threshold value, i.e. CDV(C) ≤ α, the unprocessed core object (x) may allowed for expansion in the cluster, if also satisfying the Cluster Similarity Index (CSI). If unprocessed core object (x) is allowed for the expansion than the growing cluster objects:

C= C ∪ Nε (x) =P.

The user-specified parameter i.e. {CDV(C)} α is used to detect the near about to homogeneous density clusters. But selection of parameter α is very crucial. If the value of α is small, then it generates a large number of small unimportant clusters, on the other hand if we have large value of α then it merges a number of good quality clusters into single cluster. So if we select suitable value of α then it not only separates the sparse region but also separate the region which does not have the significant density variation.

If formation of a cluster enters toward denser region from the dense region then the expansion of the cluster is stopped in the direction of its denser neighborhood. But when formation of a cluster enters toward dense region from the denser region, then the expansion of the cluster will not stop in the direction of its less dense neighborhood, due to above equation (2), because the cluster density variance of a growing cluster with respect to its ε-neighborhood of a dense core object will be less than specified threshold value α.So it will merge two or more regions into single, without having the considerable density variation. This problem can be removed by cluster similarity index, which is again a user specified parameter, used to maintain the reasonable similarity of density within the cluster.

Defnition3: (Cluster Similarity Index): The Cluster Similarity Index of a Cluster is denoted as CSI(C). The CSI of any growing cluster is calculated by including the ε-neighborhood of unprocessed core object x, before allowing the expansion of the unprocessed core object. It is defined as follows:

$$CSI(C) = Max_{O \in P}\{|N_{\varepsilon}(o)|\} - Min_{O \in P}\{|N_{\varepsilon}(o)|\} \quad (3)$$

Where P = C ∪ Nε (x), Nε (x) include x also. So the unprocessed core object x which is the object of cluster C will be allowed for the expansion if the CDV(C) ≤ α and to maintain the reasonable homogeneity of density variation within the cluster C, the Cluster Similarity Index CSI(C) ≤ λ .

Defnition4: (Density Connected Core Object): An object x is said to be Density Connected Core Object with respect to its ε-neighborhood along with the parameters μ, λ, K ∈N and ε , α ∈ R. If it is satisfying the following conditions:

(a) $|N_{\varepsilon}(x)| \geq \mu$ i.e. x must be a core object.

(b) CDV(C) ≤α.

(c) CSI(C) ≤ λ.

It is represented by DCCO(x).

Algorithm DVBSCAN (D, ε , α, μ, λ)

1. Initially all objects are unclassified.
2. For each unclassified object x ∈ D.
3. If Core(x) then
4. Generate new Cluster ID & Assign the clusterID to x.
5. Insert x into the Queue.
6. While Queue ≠ Empty.
7. Extract front object y from the Queue.
8. Calculate S={o ∈ D | dist (y, o) ≤ ε}.
9. For each object o ∈ S .
10. If o is unclassified and *DCCO(o)* .
11. Then insert o into Queue.
12. If o is unclassified or noise.
13. Then assign the clusterID to o.
14. End For.
15. End while.
16. Else x is noise
17. End for

4.1 Cluster Formation Procedure

The proposed algorithm starts to form a cluster by selecting the Core object; it inserts the selected Core object into the Queue. It pops out the front object from the seed list i.e. Queue. It calculates all the ε-neighborhood of that Core object and finds out all the Density Connected Core Objects. It inserts all Density Connected Core Objects for further expansion in the Queue, if still unclassified. The rest of the objects which are not Density Connected Core Objects with their surrounding, are simply added into the cluster, still unclassified. It expands all the Density Connected Core Objects one by one, popping out from the Queue, by following the above described procedure, which contributes more Density Connected Core Objects for further expansion. This is repeated until the queue is empty and the entire cluster is computed. All objects are either assigned a certain clusterID or marked as noise.

5. EXPERIMENTAL EVALUATION

To compare the performance of the proposed algorithm, we have also implemented the well known DBSCAN algorithm. JAVA is used as a language to implement the algorithms. The performances of above two algorithms are evaluated by using the 2-Dimensional synthetic dataset. The 2-Dimensional synthetic dataset is containing 4000 objects in 2-Dimensional plane. We performed the experiments by using the different values of parameters.

The figure 1-2, shows the clusters detected by the DBSCAN and DVBSCAN for the mentioned parameters values. The common parameters, i.e. μ and ε are having same values for both the algorithms. In figure 1, due to global input parameters ε and μ, DBSCAN algorithm detects only three clusters, because it cannot handle the density variations that exist within the cluster. The clusters out of three, two are having wide density variation within them, so such clusters can be broken down into a number of clusters that will have considerable density variation within them.

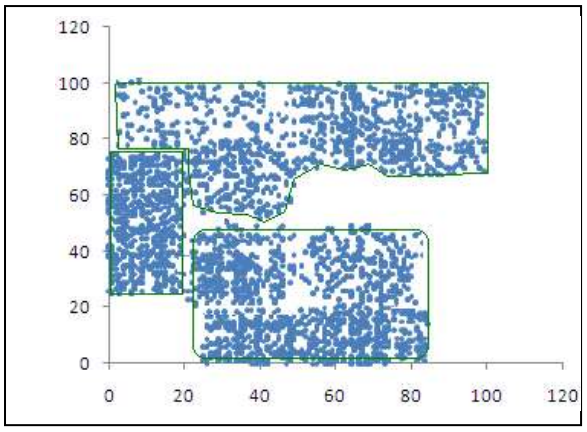

Figure 1. clusters generated by DBSCAN algorithm for the values, μ=20, ε=0.5,

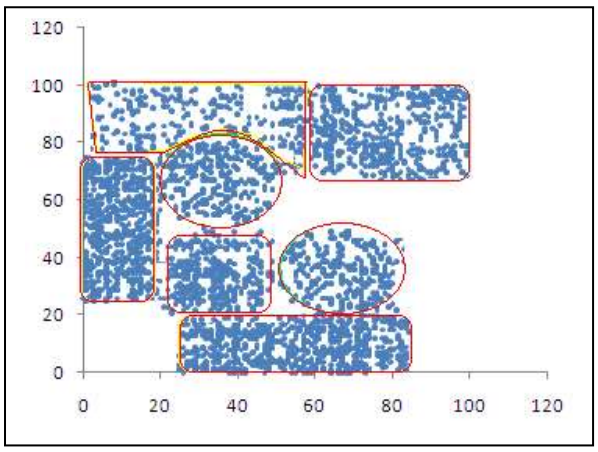

Figure 2**.** clusters generated by DVBSCAN algorithm for the values, λ=50, α=100, μ=20, ε=0.5

In addition to two global input parameters used by DBSCAN algorithm, DVBSCAN algorithm which uses two more parameters i.e. α and λ, detects seven clusters (as shown in figure 2). This shows that DVBSCAN algorithm is able to handle the density variations that exist within the cluster. The clusters detected by DVBSCAN algorithm are having considerable density variation within clusters. The detected clusters are not only separated by the sparse region but also separated by the regions, having the density variations. This illustrates that DVBSCAN outperform the DBSCAN, especially in the case of density variation within the clusters.

6. CONCLUSION

In this paper we proposed DVBSCAN, an enhancement of DBSCAN algorithm. The proposed clustering algorithm can find clusters that represent relatively uniform regions without being separated by sparse regions. A parameters α and λ are used to limit the amount of allowed local density variations within the cluster. The future work can be focused on to reduce the time complexity of algorithm and to determine the value parameters α and λ automatically for better clustering for any given data set.

7. REFERENCES

[1] Jain, A.K., Dubes, R.C. 1998. Algorithm for Clustering Data. Printice Hall Englewood cliffs NJ

[2] Han, J., Kamber, M. 2001. Data Mining: Concepts and Techniques. Morgan Kaufman

[3] Ester, M., Kriegel, H.P., Sander, J., Xu, X. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proc. KDD,

[4] Ankerst, M., Breunig, M., Kriegel, H.P., Sander, J. 1999. OPTICS: Ordering Objects to Identify the Clustering Structure. In proceedings of International Conference on Management of Data ACM SIGMOD. pp. 49–60.

[5] Hinneburg, A., Keim, D. 1998. DENCLUE: An efficient approach to clustering in large multimedia data sets with noise. In proceedings of 4th International Conference on Knowledge Discovery and Data Mining. pp. 58–65.

[6] Ram, A., Sharma, A., Jalal, A.S., Singh, R., agrawal, A. 2009. An Enhanced Density Based Spatial Clustering of Application with Noise. In proceedings of IEEE International Advance Computing Conference. pp.1475-1478

[7] Borach, B., Bhattacharya, D.K. 2007. A Clustering Technique using Density Difference. In proceedings of International Conference on Signal Processing, Communications and Networking. pp. 585–588.

[8] Borah, B., Bhattacharyya, D.K. 2008. DDSC:A Density Differentiated Spatial Clustering Technique, Journal of Computers. Vol. 3, No. 2.

[9] Peng Liu, Dong Zhou, Naijun Wu. 2007. VDBSCAN: Varied Density Based Spatial Clustering of Applications with Noise, In proceedings of IEEE Conference ICSSSM2007, pp.528-531.