Received: November 20, 2017

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Novel Density Based Clustering Algorithm by Incorporating Mahalanobis Distance

Margaret Sangeetha[1]* **Velumani Padikkaramu[2]** **Rajakumar Thankappan Chellan[3]**

[1]*Department of Computer Science, Manonmaniam Sundaranar University, Tirunelveli, India*
[2]*Department of Computer Science, The M.D.T Hindu College, Tirunelveli, India*
[3]*Department of Computer Science, St. Xavier's College, Tirunelveli, India*
* Corresponding author's Email: margaret.msu@gmail.com

Abstract: Data clustering is one of the active research areas, which aims to group related data together. The process of data clustering improves the data organization and enhances the user experience as well. For this sake, several clustering algorithms are proposed in the literature. However, a constant demand for a better clustering algorithm is still a basic requirement. Understanding the necessity, this paper proposes a density based clustering algorithm which is based on Density Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm. The main drawback of DBSCAN algorithm is it requires two important parameters as initial input. It is really difficult to fix the values for these parameters, as it requires some prior knowledge about the dataset. This requirement is overthrown by the proposed clustering algorithm by selecting the parameters automatically. The automated selection of parameters is achieved by analysing the dataset and it varies from dataset to dataset. This way of parameter selection improves the quality of service and produce effective clusters. The experimental results show that the proposed approach outperforms the DBSCAN algorithm in terms of purity, F-measure and entropy.

Keywords: Density based clustering, Data clustering, Clustering algorithm.

1. Introduction

Data is the lifeblood of today's world and the collected data are stored in voluminous databases. The data must be stored in an organized fashion, such that the required data can easily be located. Data analysis is one of the most essential necessities in all domains, such that the worth of the applications can be enhanced. Data analysis can be performed better, when the related data are stored together. The concept of data clustering hits the scene at this juncture. The major goal of data clustering is to group similar data together. The term data can be audio, video, text, numeric and so on.

The related data are grouped together, so as to form different clusters. This makes sense that entities within the cluster show maximum degree of similarity and the entities of different clusters show minimal degree of similarity. This makes the data processing easier and helps to enhance the performance of the application. Owing to its advantages, data clustering is utilized in almost all domains such as healthcare, finance, business oriented, data retrieval, image processing applications and so on. For instance, healthcare applications utilize clustering to group patients with similar symptoms or degree of severity [1]. The business oriented applications cluster the customers, who share the same buying habits [2].

Though the concept of clustering brings in numerous merits to an application, it is extremely difficult to achieve better clusters. A clustering algorithm has to handle several tough challenges such as the selection of better features, distance measures [3] and dealing with noise [4]. Apart from this, a good clustering algorithm must be scalable, capable of handling noise and to find clusters without considering the shape [5]. The clustering algorithms can be broadly divided into partitional, hierarchical, density and grid based clustering [6].

Each and every kind of clustering approach has its own merits and demerits.

This work focuses on density based clustering, which clusters the data based on the density. In this kind of clustering, the size of the cluster improves till the count of neighbouring points is greater or equal to the threshold. The threshold is chosen by the user and the cluster does not have any shape constraints. This feature makes the density based clustering popular. Taking all these points into account, this paper intends to present a clustering algorithm that is based on Density Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm. The DBSCAN algorithm is introduced by Martin Ester et.al. in the late 90's [7] and is claimed as the best density based clustering algorithms.

The main objective of this article is to present a clustering algorithm, which is an enhancement of DBSCAN algorithm. There are two important parameters associated with DBSCAN algorithm, which are epsilon (ε) and minimum number of points (min_pts). The ε value denotes the maximum distance between two data points and the min_pts denote the minimal number of points for building a cluster. The traditional DBSCAN algorithm employs static value for ε and utilizes Euclidean distance as the distance measure. The proposed algorithm enhances the traditional DBSCAN algorithm and the contributions of this work are listed below.

- The proposed clustering algorithm fixes the ε value by itself. The value of ε is chosen by calculating the distance between the points and the count of points in a particular radius.
- This way of automatic selection of ε simplifies the entire clustering process and improves the performance.
- Manual choice of ε is tiresome and may fail, in certain cases. The proposed work analyses the data distribution and fixes the value of ε, which makes the clustering process effective.
- The traditional DBSCAN algorithm utilizes Euclidean distance as the similarity measure. However, the major drawback of Euclidean distance is its sensitiveness to the geometrical shape of clusters. The proposed algorithm employs Mahalanobis distance, because of its insensitivity to cluster shape
- The performance of the proposed approach is analysed in terms of F-measure, entropy and purity. Additionally, the proposed approach studies the performances of Manhattan and Minkowski similarity measures.

The remainder of this paper is organized as follows. Section 2 reviews the related literature with respect to density based clustering algorithm. The proposed approach is elaborated in section 3 along with the overview of the work. Section 4 analyses the performance of the proposed approach and discusses the attained results. At last, section 5 presents the conclusive points of the proposed approach.

2. Review of literature

This section presents the state-of-the-art related literature with respect to density based clustering algorithm.

In [8], an effective density based clustering framework is proposed. This work separates the core and the non-core points by means of a neighbourhood density estimation model. Initially, the core points are treated by the clustering algorithm, followed by which the non-core points are treated. However, this work chooses the clustering parameters manually and consumes more time. The work proposed in [9] introduces a clustering algorithm, which can treat the data points and outliers separately. Initially, a density based clustering algorithm is employed to distinguish between the core points and the outliers in all the clusters. However, the main goal of this article is to detect outliers.

A density based clustering algorithm for location based services is proposed in [10]. This approach clusters the nearby locations with respect to a query location and returns the user with a set nearby points. A clustering algorithm based on dominant set is presented in [11]. This work produces the initial set of clusters by combining the dominant set algorithm and the histogram equalization transformation. The so produced clusters are then refined with the density information of the data points. This work involves computational and space complexity.

A density peak based semi-supervised clustering algorithm is proposed in [12], which exploits the label information. Initially, a density based clustering algorithm is employed to detect the density peaks. This is followed by the introduction of a graph based algorithm to assign the class label, by utilizing the seed information. The label information of the seed points is again utilized to form clusters, which increases the time consumption further. In [13], a density based clustering approach is proposed to diagnose neuromuscular disorders. This work proposes a clustering algorithm namely Neighbourhood Distance Entropy Consistency (NDEC) to construct arbitrary shaped clusters and

these clusters are passed to Support Vector Machine (SVM) and nearest neighbour classifiers. This work is reliable, however so many internal computations are carried out to achieve better clusters.

In [14], an improved version of DBSCAN algorithm, which is named as Different Densities-Based Spatial Clustering of Applications (DDBSCAN) is presented. The DDBSCAN algorithm calculates the cluster density with respect to epsilon and min_points. This is followed by the introduction of the density threshold, through which the data points are 'included to' or 'excluded from' a cluster. Hence, the efficiency of the work depends on the effective choice of the threshold. A Simplified Fingerprint Density-based Clustering Algorithm (SFDCA) is proposed in [15] for clustering wi-fi fingerprints. This work presents a case study by collecting wi-fi fingerprints from smartphones and the fingerprints are clustered.

In [16], a density based clustering algorithm based on density threshold is proposed. Initially, this work fixes a radius threshold and is analysed. The dense clusters are formed by merging several partial clusters. The major drawback of this work is fixing the threshold by manual analysis. An extension of DBSCAN algorithm, which is named as Spatio-Temporal DBSCAN (STDBSCAN) is proposed in [17]. The ST-DBSCAN algorithm clusters the data by taking the spatial and temporal information into account. This work utilizes rough set to cluster data and provides lower and upper approximation of the data. The lower approximation denotes the data points that must be a part of the cluster and the upper approximation indicates the cluster boundary that contains several data points, which might fall into the cluster. This work goes through several conflicts while clustering the data points.

In [18], an unsupervised learning algorithm namely Density Based Self Organizing Incremental Neural Network (DenSOINN) is presented to cluster data streams. This work is explained as a self organizing network, which expands incrementally by placing suitable nodes in a cluster and is achieved by Hebbian learning rule. By this way, DenSOINN constructs arbitrary shaped clusters. Though the performance of this work is better, the computational complexity of this work is high, as so many complex algorithms are involved in the clustering process. In [19], a transfer learning algorithm that relies on fuzzy neighbourhood density based clustering and resampling technique is proposed. This algorithm clusters the dataset in various shapes. The drawback of this work is that the clustering results are not convincing. A feature selection based DBSCAN algorithm namely FS-DBSCAN is proposed in [20]. The purpose of the algorithm is to handle high dimensional data and the performance of this work is better.

A real-time web based clustering application is proposed in [21], which is meant for clustering hotspot data being present in the peatlands by employing DBSCAN algorithm. This application clusters the hotspot data and showcases the clustering outcomes with respect to hotspots, type of peat, depth of land and so on. This work proposes a real-time clustering application and the clustering results are better. In [22], a novel density based clustering algorithm namely Probabilistic DBSCAN (PDBSCAN) is proposed for uncertain data. The PDBSCAN calculates the probability of the distance between two different objects instead of the sampling process followed by the existing DBSCAN algorithm. Besides this, the probability of the core object and support degree are utilized to compute the threshold. The method of threshold computation is complex, but the performance of this work is good in terms of clustering.

Motivated by the above works, this paper intends to propose a density based clustering algorithm which can choose the value of epsilon without human intervention. Besides this, as far as the similarity measure is concerned, DBSCAN's Euclidean distance is replaced by Mahalanobis distance, owing to its insensitiveness to the shape of the cluster. The proposed approach overthrows the head ache of choosing the value for epsilon and improves the quality of service. The following section elaborates the proposed approach.

3. Preliminaries

This section gives the basic idea of the DBSCAN clustering algorithm and the important terminologies associated with it.

3.1 Terminologies

The essences of DBSCAN algorithm are epsilon (ε) and minimum number of points (min_pts). The ε value denotes the maximum distance between two data points and the min_pts denote the minimal number of points for building a cluster. Let A be a data point and the purpose of ε is explained as follows.

- Epsilon (ε) : The ε value impacts over the data point A by forming a circle around the point A, with ε as the radius. Here, A is considered as the centroid of the circle.
- Epsilon neighbours (ε_N) : ε_N denotes the data points, which are enclosed by the so

formed circle with respect to data point A. The data points enclosed in the circle with respect to A are called as the epsilon neighbours and is denoted as $\varepsilon_N(A)$.

- Kinds of points : The constituent data points can be differentiated into three kinds, which are core point, border point and outlier point. A data point A is classified as core point, when the point A has many neighbouring points which are greater than the count of min_pts. A point A is stated as border point, when A has minimal neighbouring points. Finally, the outlier points do not come under core or border point. These points are usually considered as noise. The core and border points are denoted as follows.

$$core\ point: |\varepsilon_N(A)| \geq min\_pts \quad (1)$$

$$border\ point: |\varepsilon_N(A)| \leq min\_pts \quad (2)$$

where $|\varepsilon_N(A)|$ is the cardinality of $\varepsilon_N(A)$.

- Directly density reachability : A data point A is considered to be directly density reachable to data point B, when B is one of the points in $\varepsilon_N(A)$ and A is the core point. The $\varepsilon_N(B)$ are directly density reachable from B and the border points are directly density reachable from its own epsilon neighbours that are core points.
- Density reachability : A data point A is claimed to be density reachable from the data point B . Consider a set of interconnected points $A_1, A_2, A_3, \ldots, A_n$, such that $A_1 \leftarrow A$ and $A_n \leftarrow B$ and A_{i-1} is directly density reachable from p_i.
- Density connected : A data point A is said to be density connected to a point B, if a point P is present and the points A, B are density reachable from P.
- Density based cluster : Consider a set of points, which is mentioned as X. A density based cluster X is formed with atleast a core point and all other data points are density reachable from the core point.

Thus, the basic terminologies associated with DBSCAN algorithm are presented above and the traditional DBSCAN algorithm is presented below.

DBSCAN Algorithm

Input : $\varepsilon, min_{pts}, dataset;$
Output: Data clusters
Begin
{
$Cluster = 0;$
for each point x
{
If x is checked
Check the next point;
Compute neighbour points of x by passing region_query (x, ε);
If $\varepsilon_N(x) < min_{pts}$
Set x as outlier;
Else
{
Cluster=next cluster;
Growcluster(x, $\varepsilon_N(x)$, cluster, ε, min_{pts})
}
}
Growcluster(x, $\varepsilon_N(x)$, cluster, ε, min_{pts})
{
Include x in cluster;
For each point x' in $\varepsilon_N(x)$
{
If p' is not checked
{
Set p' as checked;
Compute neighbour points of x' by passing region_query (x', ε);
If $\varepsilon_N(x) \geq min_{pts}$;
Add neighbour points of x and x';
}
If x' is not a member of any cluster
Include x' in cluster;
}
}
region_query (x, ε)
return all the data points that are inside the neighbourhood of x;
}

This original DBSCAN algorithm avoids the need of pre-determining the count of clusters. DBSCAN can deal with noisy data effectively and can find clusters of irregular shape. However, this work cites two major drawbacks, which are as follows. Initially, it is quite hard to set the initial parameter epsilon (ε). Taking this issue into account, the proposed density based clustering algorithm intends to automate the choice of ε. Though the DBSCAN algorithm is claimed to produce arbitrary shaped clusters, the employed similarity measure 'Euclidean distance' is indeed sensitive to the shape of the cluster. This issue is resolved by the proposed approach by incorporating 'Mahalanobis distance' in the place of Euclidean distance, as mahalanobis distance is insensitive to the shape of the cluster.

The following section presents the proposed clustering algorithm.

4. Proposed density based clustering algorithm

The main goal of this algorithm is twofold. One is to automate the choice of ε and the second one is to study the performances of different similarity measures such as mahalanobis, manhattan, minkowski, which are compared with the Euclidean distance. Initially, this section presents the details about the automated choice of ε. The traditional DBSCAN algorithm prompts the user to provide the value for ε and min_pts. The efficiency of the DBSCAN algorithm strongly relies on the choice of ε. The feasible value of ε produces better clusters. Thus, preliminary knowledge about the dataset is necessary, such that the value of ε can be fixed. Yet, a novice user may not be able to select an optimal value of ε, which seriously impacts over the formation of clusters. Hence, the advice of a technical expert becomes necessary for the parameter fixation. However, it is not always possible to look for a technical expert.

The second issue is the demerits associated with Euclidean distance, which is the standard similarity measure of DBSCAN algorithm. Though the computation of Euclidean distance is simple, it has certain drawbacks to be addressed. Euclidean distance is sensitive to the shape of the cluster and it could not handle the correlated data items. All these issues are overthrown by mahalanobis distance, which is insensitive to the shape of the cluster and the correlated data items are processed effectively. Besides this, the mahalanobis distance can find the outliers effectively. The proposed work proves its superiority by including the automated choice of ε and mahalanobis distance as the similarity measure. The proposed clustering algorithm is as follows.

Proposed Algorithm for ε and min_pts computation

Input : Dataset DS
Output : data clusters
Begin
For each data point $dp_i \in DS$ do
Obtain the coordinates of dp_i
Compute mahalanobis distance;
Sort the distance outcome ($dist$) in ascending order;
Find the nearest neighbours nn of dp_i;
Count the nn for the top ranking $dist$;
For each available $dist_n(dp_i)$ do
Count $nn(dp_i)$ as no;
Store $no(nn(dist_j))$ and $(dist_j)$;
Repeat the process for all $dist_n$;
For all $dist$
Compute $avg(no(dist_i))$;
Assign $avg(no(dist_i))$ as min_pts;
List $dist_n \geq min\_pts$;
Assign $max(dist_n)$ as ε;
End for
End for
End for
End

The above presented algorithm describes the way to find the values for ε and min_pts. The so found values are passed as input to form the clusters. The clusters are formed with the computed ε and min_pts, which brings in simplicity and efficiency. Additionally, the overhead associated with the choice of ε and min_pts are eliminated. As the choice of these parameters decide the quality of clusters, it is better to choose optimal values for the parameters. Manual choice of ε and min_pts can be achieved by trial and error method, which consumes more time and involves computational overhead. All these overheads are overcome by the proposed approach, which fixes an optimal value for ε and min_pts, which is dependent on the nature of dataset. The proposed algorithm can work for any kind of dataset, which widens the applicability of the algorithm.

There is no need for providing the values of ε and min_pts initially, as in traditional DBSCAN algorithm. Additionally, the need for passing the count of clusters as that of k-means algorithm is also eliminated. All these factors together make it simple to deal with cluster formation. This algorithm requires no prior knowledge with respect to clustering or its associated parameters, hence it is suitable for novice users and supports experts as well.

As soon as the dataset is passed, the coordinates of all the points are obtained and the mahalanobis distance is computed for all the data points. By this way, the k-nearest neighbours of any particular data point is obtained. This is followed by sorting the computed distances in sorted order (ascending). This way of distance sorting, helps in finding the least possible distance between the processed data point and its neighbourhood points. The next step is to count the number of nearest neighbours of a specific point with respect to all the computed distances.

This is followed by computing the average of the count of neighbouring points of all distances being observed. This average value is set as the minimum

points. Now, the distance on which the neighbourhood points equals or greater than the minimum points are listed. The maximum distance which encloses more number of points is chosen as the ε value. This process continues till all the data points are included in a cluster. In case, if a point cannot come under any cluster then those points are considered as noise. The following section analyses the performance of the proposed approach.

5. Results and discussion

The performance of the proposed approach is tested with two different datasets namely 'online retail' and 'wholesale customer' datasets, which are downloaded from [23, 24] respectively. The online retail dataset contains the transactional details of a UK based online store. This dataset comprises eight different attributes such as invoice number, stock code, description, quantity, invoice date, unit price, customer ID and country. The wholesale customer dataset contains the annual expenditure details of eight different attributes such as fresh, milk, grocery, frozen, detergents&paper, delicatessen products, channel and region. Both these datasets contain about five hundred records each. The experimental analysis is carried out in a stand alone system with 4 GB RAM by utilizing MATLAB version 8.2.

The performance of the proposed approach is studied in two ways. Initially, the proposed approach is analysed by varying different similarity measures such as Euclidean, Mahalanobis, Manhattan and Minkowski. Out of all these performance measures, mahalanobis distance performs better for the utilized datasets. Secondly, the performance of traditional DBSCAN algorithm is compared with the proposed approach. The performance of the proposed approach is analysed in terms of standard performance metrics such as entropy, f-measure and purity. The definitions of these performance metrics are provided below.

- F-measure : The greater the F-measure, the better is the clustering results. The maximal F-measure results in the correct mapping of data points to the clusters. The F-measure of a particular cluster (cl) is computed by

$$F(cl) = \frac{2PrRc}{Pr+Rc} \quad (3)$$

$$\Pr(x,y) = \frac{C_{xy}}{C_y} \quad (4)$$

$$Rc(x,y) = \frac{C_{xy}}{C_x} \quad (5)$$

  where Pr and Rc are the precision and recall rates respectively. C_{xy} is the count of the entities of a particular category x in the cluster y. C_x and C_y are the total count of entities or points in class x and y respectively.

- Entropy : The entropy value determines the homogeneity of the cluster. The homogeneity of the cluster is inversely proportional to the entropy value. The entropy value of a cluster is calculated by

$$Ent(cl) = \sum_{y=1}^{C_{cl}} \frac{C_y}{C} \times Ent_y \quad (6)$$

  where C_y is the count of data points in cluster y, C is the total count of data points. Ent_y is computed by the following

$$Ent_y = -\sum_x prb_{xy} \log(prb_{xy}) \quad (7)$$

  In the above equation, prb_{xy} is the probability of data point in cluster y to exist in category x. Hence, a better clustering algorithm should prove maximum F-measure and minimal entropy value.

- Purity : Purity of the cluster denotes the wholeness of a cluster. The purity of a cluster cl_x whose size is sz_x is measured by

$$p(cl_x) = \frac{1}{sz_x} \max_d cl_x^d \quad (8)$$

  In Eq. (8), $\max_d sz_x^d$ is the count of data points, which are the parts of a particular category in cl_x and cl_x^d are the total data points in cluster that are allotted to the class d. Suppose, if the purity of a cluster is 1, then all the data points of the cluster belong to a single category. The greater the purity value, the better is the quality of the clusters.

All these performance metrics are taken into account to assess the quality of the proposed clustering algorithm. The performance of the proposed approach is proven by the results.

In Figs.1 and 2, the performance of the proposed approach is tested by varying the similarity measure and the purity, F-measure and entropy are computed. On analysis, it is found that Euclidean distance is the poor performer of all the similarity measures with the least F-measure, purity and the greatest entropy value. For the wholesale dataset, the entropy value being shown by Euclidean distance is 0.68. The purity and F-measure of the Euclidean distance is 0.69 and 0.64 respectively. The purity, F-measure and entropy values shown by Euclidean distance for

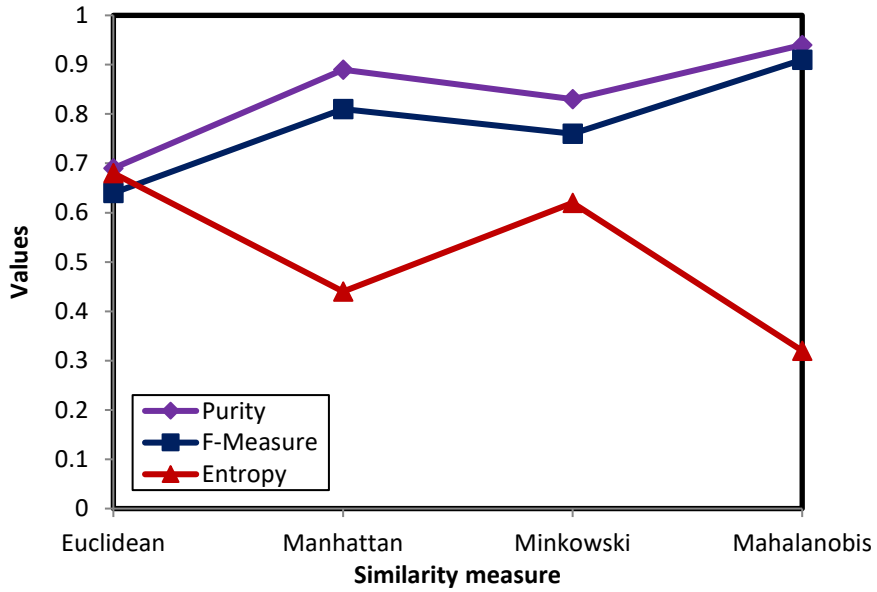

Figure. 1 Performance analysis on wholesale dataset

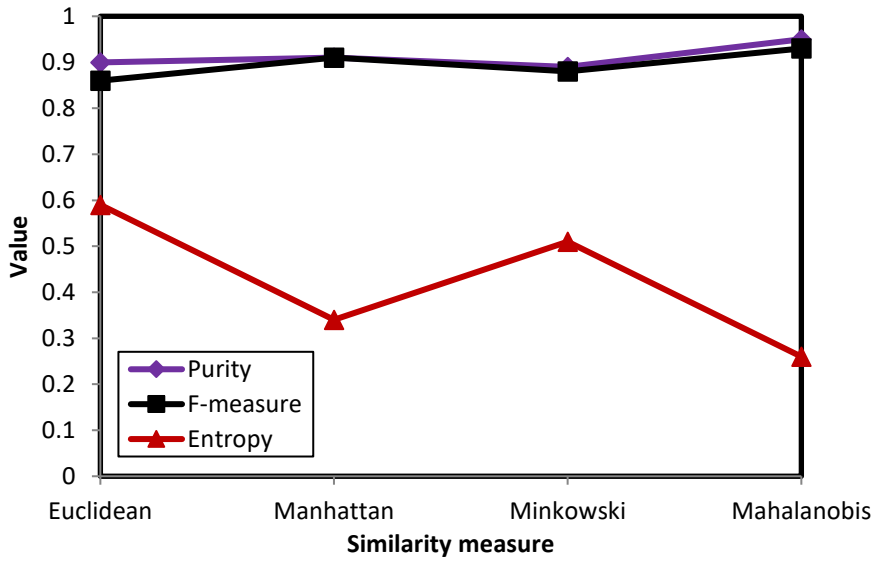

Figure. 2 Performance analysis on online retail dataset

the online retail dataset are 0.9, 0.86 and 0.59 respectively. Minkowski is the second poor performer that shows 0.83 and 0.76 as purity and F-measure respectively. The entropy value shown by minkowski distance is 0.62. In case of online retail dataset, the minkowski distance proves 0.89, 0.88 and 0.51 as purity, F-measure and entropy respectively. The performance of minkowski and mahanttan distances is more or less the same. This is because, the minkowski distance is the generalization of Euclidean and manhattan distances.

For the wholesale dataset, the manhattan distance shows 0.89, 0.81 and 0.44 as the purity, F-measure and entropy respectively. As far as the online retail dataset is considered, the manhattan distance shows 0.91, 0.90 and 0.34 for purity, F-measure and entropy. The major drawback of manhattan distance is it considers the mutual correlation of the data points alone and does not make decision out of the dominance. However, manhattan distance is insensitive to noise and can handle correlations between the data points. Finally, mahalanobis distance shows the greatest F-measure and purity value and the least entropy value. The purity and the F-measure of the formed clusters for the wholesale dataset are 0.94 and 0.91 respectively. The entropy value being shown by mahalanobis distance is the 0.32. The mahalanobis distance shows the best results even for the online retail

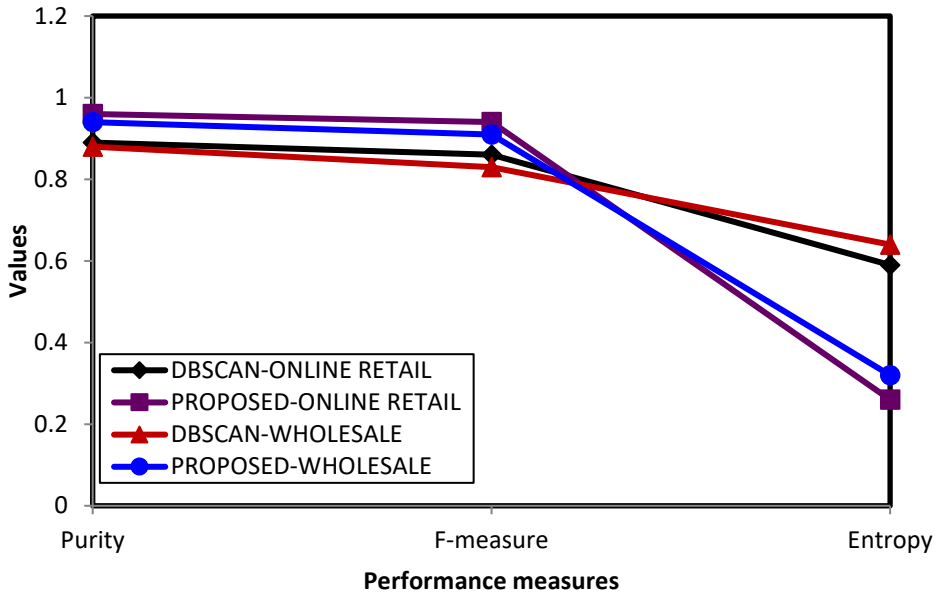

Figure. 3 Comparative analysis between DBSCAN and the proposed approach

dataset too. The purity and F-measure of the clusters are 0.95 and 0.93 respectively. The entropy value of the formed clusters is 0.26, which is the least. The reason for the better performance of mahalanobis distance is its ability to handle scale, correlation issues and outliers.

In Fig. 3, the performance of the DBSCAN algorithm proposed in [8] is compared with the proposed clustering algorithm. The performance difference between the DBSCAN and proposed approaches is obvious. The main reason for the poor performance of the DBSCAN algorithm is the incorporation of Euclidean distance, which does not take the data point correlation into account. Besides this, the effectiveness of the DBSCAN algorithm in [8] depends on the significant parameters such as ε and min_pts. These issues are addressed by the proposed approach by incorporating mahalanobis distance, which can deal with scale and correlation issues. Additionally, the outliers can be detected easily. Apart from this, the proposed approach eliminates the requirement of passing values for ε and min_pts manually. Instead, the optimal values for ε and min_pts are chosen by the algorithm itself. Thus, the proposed approach is efficient and improves the quality of service as well.

6. Conclusion

This paper introduces a density based clustering algorithm, which is based on traditional DBSCAN algorithm. The proposed approach is observed to be superior to the traditional DBSCAN algorithm, owing to two solid reasons. Initially, the traditional DBSCAN algorithm requires the values for ε and min_pts as input. As the efficiency of the clustering algorithm depends on the ε and min_pts values, it is necessary to choose the optimal values. However, this requires some prior knowledge about the dataset. This requirement is completely uprooted by the proposed approach, in which the values of ε and min_pts are chosen automatically, by analysing the dataset. However, the values of ε and min_pts varies with respect to the dataset. Secondly, mahalanobis distance is utilized as the distance measure in the place of Euclidean distance. This is because mahalanobis distance can deal with scale and correlation issues very well, which cannot be achieved by Euclidean distance. The performance of the proposed approach is satisfactory in terms of purity, F-measure and entropy. However, the performance of the proposed work is tested over static dataset. In future, this work is planned to be enhanced by introducing a dynamic dataset.

References

[1] Z. Nafar and A. Golshani, "Data Mining Methods for Protein–Protein Interactions", In: *Proc. of Canadian Conf. on Electrical and Computer Engineering*, pp. 991–994, 2006.

[2] W. Yu, G. Qiang, and L. X. Li, "A kernel aggregate clustering approach for mixed data set and its application in customer segmentation", In: *Proc. of International Conference on Management Science and Engineering*, pp. 121–124, 2006.

[3] S. H. Cha, "Comprehensive survey on distance/similarity measures between probability density functions", *International*

Journal of Mathematical Models and Methods in Applied Sciences, Vol.1, No.4, pp.300–307, 2007.

[4] C. Bahm, K. Haegler, N.S Maller, and C. Plant, "CoCo: coding cost for parameter-free outlier detection", In: *Proc. of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 149–158, 2009.

[5] J. Han, J. Pei J, and M. Kamber, "Data mining: concepts and techniques", Third Edition, Elsevier, 2011.

[6] E. Paquet, "Exploring anthropometric data through cluster analysis", *Digital Human Modelling for Design and Engineering,* Oakland University, Michigan, 2004.

[7] 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. of KDD,* Vol. 96, No. 34, pp. 226-231, 1996.

[8] J. Lu and Q. Zhu, "An Effective Algorithm Based on Density Clustering Framework", *IEEE Access*, Vol.5, pp.4991-5000, 2017.

[9] K. Maheshwari and M. Singh, "Outlier detection using divide-and-conquer strategy in density based clustering", In: *Proc. of International Conference on Recent Advances and Innovations in Engineering*, 2016.

[10] M.F. Rahman, W. Liu, S.B. Suhaim, S. Thirumuruganathan, N. Zhang, and G. Das, "Density Based Clustering over Location Based Services", In: *Proc. of International Conference on Data Engineering*, 2016.

[11] J. Hou, E. Xu, and H. Cui, "Density Based Dominant Sets Growing and Clustering", In: *Proc. of European Modelling Symposium*, 2016.

[12] W. Li, X. Li, Y. Ye, Y. Li, and E.K. Wang, "A novel density peak based semi-supervised clustering algorithm", In: *Proc. of International Conference on Machine Learning and Cybernetics*, 2016.

[13] T. Kamali and D. Stashuk, "A Density-Based Clustering Approach to Motor Unit Potential Characterizations to Support Diagnosis of Neuromuscular Disorders", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol.25, No.7, pp.956-966, 2017.

[14] M. F. Hassanin, M. Hassan, and A. Shoeb, "DDBSCAN: Different Densities-Based Spatial Clustering of Applications with Noise", In: *Proc. of International Conference on Control, Instrumentation, Communication and Computational Technologies*, 2015.

[15] S. Lau, C. Toh and Y. Saleem, "Wi-Fi Fingerprint localisation using Density-based Clustering for public spaces: A case study in a shopping mall", In: *Proc. of International Conference on Cloud System and Big Data Engineering*, 2016.

[16] K. Zhang, L. Huang, and Y. Chai, "An algorithm to adaptive determination of density threshold for density-based clustering", *In: Proc. of Chinese Control Conference*, 2016.

[17] B. Chakraborty, K. Chakma, and A. Mukherjee, "A density-based clustering algorithm and experiments on student dataset with noises using Rough set theory", In: *Proc. of IEEE International Conference on Engineering and Technology*, Coimbatore, 2016.

[18] B. Xu, F. Shen, and J. Zhao, "Density Based Self Organizing Incremental Neural Network for data stream clustering", In: *Proc. of International Joint Conference on Neural Networks*, 2016.

[19] Z. Liu, J. Yang, H. Liu, and W. Wang, "Transfer Learning by Fuzzy Neighbourhood Density Based Clustering and Re-sampling", In: *Proc. of International Conference on Computer Science and Applications*, 2015.

[20] A. Smiti and Z. Elouedi, "Fuzzy density based clustering method: Soft DBSCAN-GM", In: *Proc. of International Conference on Intelligent Systems*, 2016.

[21] R. Hermawati and I. S. Sitanggang, "Web-Based Clustering Application Using Shiny Framework and DBSCAN Algorithm for Hotspots Data in Peatland in Sumatra", *Procedia Environmental Sciences*, Vol.33, pp.317-323, 2016.

[22] M.U. Yaseen, A. Anjum, O. Rana, and R. Hill, "Cloud-based scalable object detection and classification in video streams", *Future Generation Computer Systems*, Vol.80, pp.286-298, 2018.

[23] http://archive.ics.uci.edu/ml/datasets/Online+Retail

[24] https://archive.ics.uci.edu/ml/datasets/Wholesale+customers