A Comparison Between Several Density-Based Anomaly Detection Approaches

Dr. Rajeev Tripathi[1], Mr. Alok Singh[2]

[1,2]Dept. of Computer Sciences, School of College Management Sciences, Lucknow, (Aff1||India)

ABSTRACT

The current world of internet, mobile devices, businesses, social media platforms, healthcare systems, and the Internet of Things all have a lot of data available online. The enormous volume of data, dimensionality, and dataset changes throughout time are these problems. Clustering algorithms are a useful tool for solving this type of problem. Consequently, the first step in resolving these issues is the application of clustering algorithms, which are necessary for data mining procedures to reveal the structure and hidden patterns in given datasets. Four clustering algorithms OPTICS (Ordering Points To Identify Clustering Structure), Hierarchical Clustering, DBSCAN (Density-Based Spatial Clustering of Applications with Noise), AGGLOMERATIVE

and HDBSCAN. The efficiency of each clustering approach is assessed using a range of external and internal parametric clustering assessment metrics.

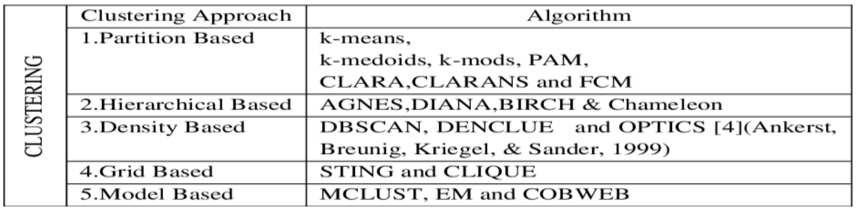
Keywords: Deep learning, Clustering Algorithms, Density based Algorithms, Clustering in Presence of Noise.

1. INTRODCUTION

There is a wealth of data available from several sources in the digital age. In order to properly evaluate these data and create the associated automated and intelligent applications, one has to be well-versed in machine learning (ML) and artificial intelligence (AI). This field encompasses supervised, unsupervised, semi-supervised, and reinforcement learning machine learning techniques. Additionally, deep learning, which is a subset of machine learning techniques, has the capacity to analyse massive amounts of data in an intelligent manner [1]. Based on extracted properties, time series classification creates classes from the temporal data for identification and recognition. [2].The feature extraction procedure would be far more reliable if preprocessing and feature extraction were done using signal processing techniques.

The process of sorting significant groupings of items with common characteristics into smaller groups referred to as clusters is called clustering. While there exists several different types of clustering [3], we will focus on the two that are relevant to our problem at hand partitioned and hierarchical clustering. In hierarchical clustering, nested clusters are formed like a tree, with the root cluster being the set of all the items and internal nodes, or clusters, being the union of its sub clusters. To uncover interesting patterns or trends in data, including customer groups based on behavior, it is widely used as a data analysis technique [4]. Numerous application domains, such as user modeling, health analytics, e-commerce, mobile data processing, cyber security, and behavioral analytics, can benefit from the usage of clustering.

Data is divided into subsets or clusters as part of the partitioned clustering process [Table 1]. Every partition's subsets must not overlap and share a single, comparable data item with no more than two subgroups. The problem statement states that n patterns need to be divided into k clusters in a d-dimensional space [5]. K might be mentioned or it could not. According to the partitioned clustering solution, which states that a criterion must be chosen and evaluated for each partition, the partition that best satisfies the criteria should be chosen.



CLUSTERING	Clustering Approach	Algorithm
	1.Partition Based	k-means, k-medoids, k-mods, PAM, CLARA,CLARANS and FCM
	2.Hierarchical Based	AGNES,DIANA,BIRCH & Chameleon
	3.Density Based	DBSCAN, DENCLUE and OPTICS [4](Ankerst, Breunig, Kriegel, & Sander, 1999)
	4.Grid Based	STING and CLIQUE
	5.Model Based	MCLUST, EM and COBWEB

Table1. Categorization of Clustering Algorithm.[12]

Partitioning methods: This clustering method divides the data into several groups or clusters according to the characteristics and commonalities in the data. Depending on the nature of the target applications, data scientists or analysts usually choose the number of clusters either statically or dynamically to construct for the clustering algorithms. K-means [6], K-Mediods [7], CLARA [8], and other clustering algorithms based on partitioning techniques are the most often used ones.

Our study primarily focuses on density-based partitional clustering approaches. Density based clusters are defined as clusters with varying densities that set them apart from other clusters. This suggests that a group of objects with a high density may be surrounded by low density zones.

Density-based connectivity and density-based functions are the two types of density-based techniques.

Density-based methods: The premise that a cluster in the data space is a continuous region of high point density separated from other comparable clusters by contiguous regions of low point density is used to identify distinct groupings or clusters. Points that are not part of a cluster are known as noisy points. The density-based techniques typically fail when dealing with big multidimensional data and clusters of equal density.

2. BACKGROUND AND RELATED WORK

This section will lay out the foundational ideas and background information required for the conversations to follow. This section outlines the key methods and the validation strategy associated with the gene clustering.

2.1 DBSCAN

DBSCAN stands for Density Based Spatial Clustering of Applications with Noise. Higher density regions are referred regarded as clusters and lower density parts as noise in partitional type clustering. Clusters are seen by the DBSCAN algorithm as high-density regions divided by low-density regions [9]. In contrast to k-means, which presumes that clusters are convex formed; DBSCAN's more flexible perspective allows clusters to be of any shape. The idea of core samples that is, samples found in densely populated areas is the foundation of the DBSCAN. Certain traits are used to define clusters, and these include:

- **Core:** Within density-based clusters, core points are within the user-specified Eps (radius or threshold value) and MinPts (minimum number of points) parameters.

- **Border:** A border point can be found close to a core point, and a single border point can be shared by many core points.
- **Noise:** The location that isn't a core or a border.
- **Directly Density Reachable:** When a belongs to NEps(s) and |NEps (s)| >= MinPts, then a point r is directly density accessible from s with regard to Eps and MinPts.
- **Density Reachable:**Density that may be accessed from a point r that is a point s.Eps and MinPts are identified if a sequence of sites, r1...rn, where r1 = s and rn = s, can be accessed directly from ri.

Algorithm

The DBSCAN algorithm's steps are as follows:

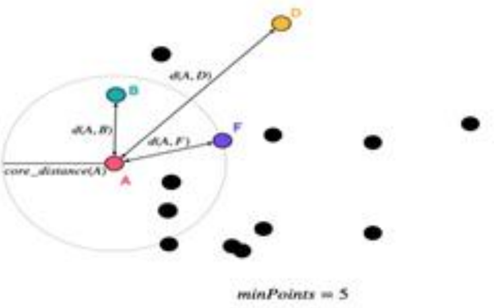
- Arbitrary choose a point r.
- Retrieve all locations density-reachable from r w.r.t Eps and MinPts.
- A cluster is created if r is a central point.
- If r is a boundary point, DBSCAN scans the next point in the database since no points can be density-reachable from r.
- Keep going until all of the points have been handled.

2.2 OPTICS

As an improved ordering method, OPTICS (ordered points to identify clustering structure) saves the order in which the points are handled rather than allocating cluster memberships. With one significant exception—the ε parameter is potentially superfluous—OPTICs uses the same minPoints and ε hyper parameters as DBSCAN. The sole practical reason to explicitly adjust this parameter's value is to lower the algorithm's runtime complexity [10]. Since many OPTICS algorithm implementations have the epsilon parameter set to a relatively large number, we will assume this for the following instances.

In addition to the concepts mentioned above of the DBSCAN algorithm, OPTICS introduces two more terms, namely:

- **Core distance:** the minimal separation needed to classify a data point p as a core point. The p's core distance is unknown if it is not a core point.
- **Reachability distance:** If p is a core point, the reachability distance between point q and point p is the minimum distance that allows q to be directly reached from p. Since there are no sites that are directly accessible from point p, at lesser distances, this value cannot be less than the core distance of point p [Figure 1]. The reachability distance between point q and point p is arbitrary if p is not a core point.



$$\left.\begin{array}{l} d(A,F) = 0.4489609 = core\_distance(A) \\ d(A,B) = 0.3452897 < core\_distance(A) \\ d(A,D) = 0.9740773 > core\_distance(A) \end{array}\right\} \Longrightarrow \begin{array}{l} reachability\_distance(B,A) = core\_distance(A) = 0.4489609 \\ reachability\_distance(F,A) = core\_distance(A) = 0.4489609 \\ reachability\_distance(D,A) = d(A,D) = 0.9740773 \end{array}$$

Figure 1.The concept of reachability distance between point q and point p is arbitrary [13].

2.3 HDBSCAN

One may think of the HDBSCAN algorithm as an expansion of OPTICS and DBSCAN. DBSCAN specifically presupposes that the density requirement, or clustering criteria, is globally homogenous. To put it another way, DBSCAN could have trouble capturing clusters with varying densities [11]. HDBSCAN creates a different representation of the clustering issue in order to mitigate this assumption and investigate all potential density scales. The following is a summary of the HDBSCAN algorithm:

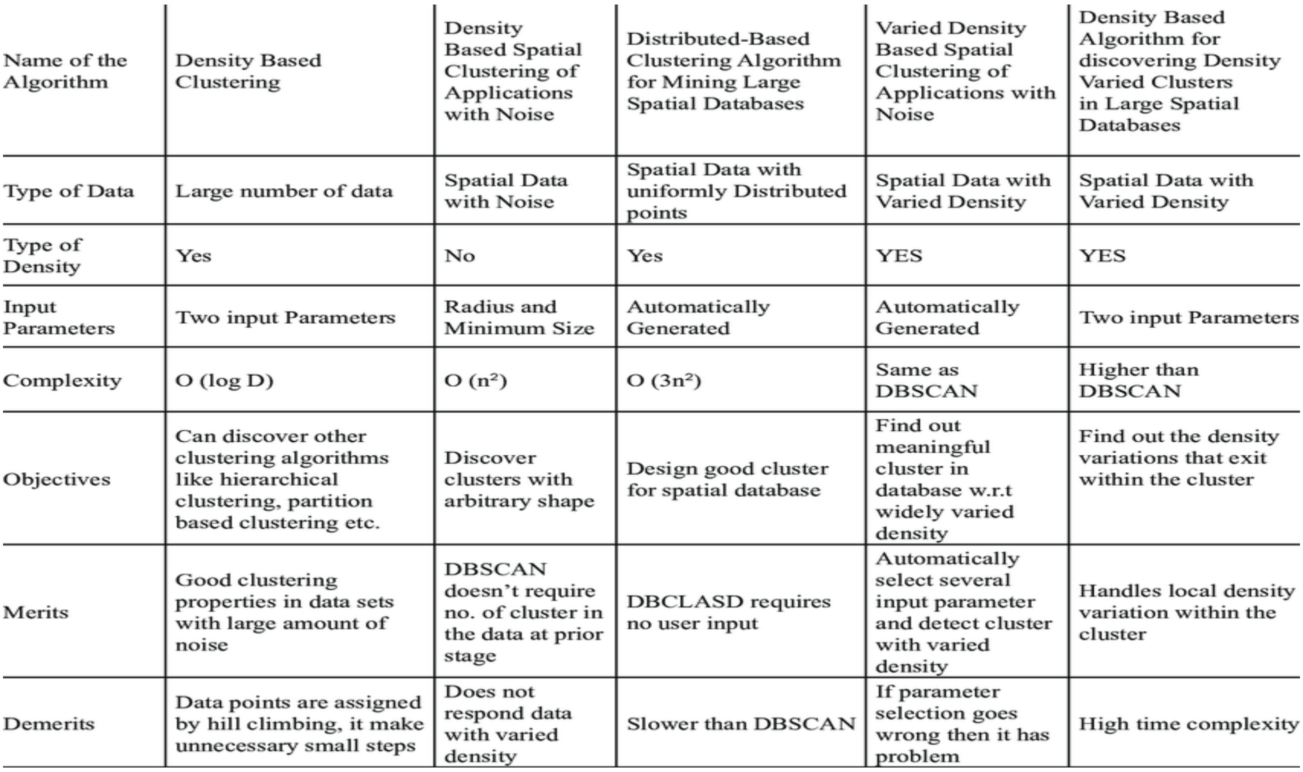
1. Take the MST out of
2. To extend the MST, add a "self edge" to each vertex, with a weight determined by the underlying sample's core distance.
3. Set up the MST with a single cluster and label.
4. Cut the MST edge that weighs the most (ties are cut at the same time).
5. Give the linked components that hold the ends of the now-removed edge cluster names. If the component does not have at least one edge it is instead allocated a “null” label designating it as noise.
6. Continue steps 4-5 until all related components are gone.

2.4 AGGLOMERATIVE

The agglomerative clustering is the most frequent kind of hierarchical clustering used to put things in clusters based on their similarity. Another name for it is Agglomerative Nesting, or AGNES. Every item is first seen by the algorithm as a singleton cluster. Subsequently, clusters are combined in pairs until all clusters are combined into a single large cluster that contains all of the items. The ultimate product is a dendrogram, which is an object representation based on a tree. Using a bottom-up methodology, the Agglomerative Clustering object carries out hierarchical clustering: each observation begins in its own cluster, and clusters are gradually combined. The measure for the merging technique is determined by the linkage criteria:

- Ward minimises the sum of squared differences within all clusters. This method, which uses an agglomerative hierarchical technique to address variance, is comparable to the k-means objective function in this regard.
- The greatest distance between observations of pairs of clusters is minimised when there is maximal or full connection.
- The average of the distances between every observation of a pair of clusters is minimised using average linkage.
- The distance between the closest observations of pairs of clusters is minimised via single connection.

A sort descriptive comparative study of other density based clustering algorithms for data mining are given bellow [Table 2]:



Name of the Algorithm	Density Based Clustering	Density Based Spatial Clustering of Applications with Noise	Distributed-Based Clustering Algorithm for Mining Large Spatial Databases	Varied Density Based Spatial Clustering of Applications with Noise	Density Based Algorithm for discovering Density Varied Clusters in Large Spatial Databases
Type of Data	Large number of data	Spatial Data with Noise	Spatial Data with uniformly Distributed points	Spatial Data with Varied Density	Spatial Data with Varied Density
Type of Density	Yes	No	Yes	YES	YES
Input Parameters	Two input Parameters	Radius and Minimum Size	Automatically Generated	Automatically Generated	Two input Parameters
Complexity	O (log D)	$O (n^2)$	$O (3n^2)$	Same as DBSCAN	Higher than DBSCAN
Objectives	Can discover other clustering algorithms like hierarchical clustering, partition based clustering etc.	Discover clusters with arbitrary shape	Design good cluster for spatial database	Find out meaningful cluster in database w.r.t widely varied density	Find out the density variations that exit within the cluster
Merits	Good clustering properties in data sets with large amount of noise	DBSCAN doesn't require no. of cluster in the data at prior stage	DBCLASD requires no user input	Automatically select several input parameter and detect cluster with varied density	Handles local density variation within the cluster
Demerits	Data points are assigned by hill climbing, it make unnecessary small steps	Does not respond data with varied density	Slower than DBSCAN	If parameter selection goes wrong then it has problem	High time complexity

Table2.Comparative Study of Density Based Clustering Algorithms for Data Mining [14]

Clustering Performance Evaluation

Analysing a clustering algorithm's performance is more complicated than calculating a supervised classification algorithm's accuracy and recall or the amount of mistakes it makes. Specifically, no evaluation metric should consider the absolute values of the cluster labels; instead, it should consider whether the clustering defines data separations that are comparable to a ground truth set of classes or that meet an assumption that, in terms of some similarity metric, members of the same class are more similar than members of different classes..

Rand index: This function, which disregards permutations, calculates how similar the two assignments are to one other. Lower values suggest dissimilar labelings, related clusterings have a high (adjusted or uncorrected) Rand index, 1.0 is the ideal match score. For the unadjusted Rand index, the score range is [0, 1], while for the adjusted Rand index, it is [-1, 1]. Even when there is a large difference in the clusterings themselves, the unadjusted Rand index is frequently near 1.0.

If C is a ground truth class assignment and K the clustering, let us define a and b as:

- **a**, the number of pairs of elements that are in the same set in C and in the same set in K
- **b**, the number of pairs of elements that are in different sets in C and in different sets in K

$$\text{RI} = \frac{a+b}{C_2^{n_{samples}}}$$

Where $C_2^{n_{samples}}$ is the total number of possible pairs in the dataset. It does not matter if the calculation is performed on ordered pairs or unordered pairs as long as the calculation is performed consistently.

Fowlkes-Mallows scores

The Fowlkes-Mallows index can be used when the ground truth class assignments of the samples is known. The Fowlkes-Mallows score FMI is defined as the geometric mean of the pairwise precision and recall:

$$\mathrm{FMI} = \frac{\mathrm{TP}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})}}$$

The variables TP, FP, and FN represent the number of True Positive, False Positive, and False Negative, respectively. TP is the number of pairs of points that belong to the same clusters in both the true labels and the predicted labels, while FP is the number of pairs of points that belong to the same clusters in the true labels and not in the predicted labels.

Significant agreement is shown by values near to one, whereas values close to zero show two label assignments that are essentially independent. Additionally, labels assigned with an FMI of exactly 1 imply equality between the two labels (with or without permutation), whereas values of exactly 0 indicate wholly independent label assignments.

Silhouette Coefficient

In the event that the ground truth labels are unknown, the model itself must be used for assessment. One such assessment is the Silhouette Coefficient, where a model with better-defined clusters is associated with a higher score. Each sample has a set silhouette coefficient, which consists of two scores:

- **a**: The mean distance between a sample and all other points in the same class.
- **b**: The mean distance between a sample and all other points in the *next nearest cluster*.

The Silhouette Coefficient *s* for a single sample is then given as:

$$s = \frac{b-a}{max(a,b)}$$

- The range of possible scores is -1 for improper clustering and +1 for extremely dense clustering. Overlapping clusters are indicated by scores around 0.
- According to the conventional definition of a cluster, a cluster has a better score when it is dense and well-separated.
- Compared to alternative cluster ideas, such density-based clusters like those produced by DBSCAN, the silhouette coefficient is often larger for convex clusters.

Davies-Bouldin Index

The Davies-Bouldin index can be used to assess the model in the event that the ground truth labels are unknown; a lower index indicates a higher degree of separation between the clusters in the model. The average "similarity" of clusters is represented by this index, which computes the similarity between clusters by comparing their respective sizes and distances from one another. The lowest score attainable is zero. A better split is shown by values that are closer to zero.

CONCLUSION

To extract valuable information regarding the anomaly identification about different systemic situations, a number of clustering techniques have been created. In order to identify and study a wide range of illnesses, including cancer, malaria, asthma, and TB, clustering has been used extensively in the medical industry. Typically, the conventional approach to data analysis is unable to uncover the datasets underlying patterns. Data mining is a valuable technique, then. In comparison of several approaches, the data under consideration yields the best results for AGGLOMERATIVE. In this article, we compared diverse techniques to analyze the dataset whether it contains some type of anomaly.

REFFERENCE

1. A. Hinneburg and D. Keim, “An efficient approach to clustering Large multimedia databases with noise,” in Proc4th Int. Conf. Knowledge Discovery and Data Mining (KDD‟98), 1998, pp. 58–65.
2. Tripathi, Rajeev (2023). Inventiveness of Text Extraction with Inspiration of Cloud Computing and ML using Python Logic, 22nd Intelligent Systems Design and Applications (ISDA’22), 2022/12
3. McCallum, A, K. Nigam, and L.H. Ungar. Efficient Clustering of High-dimensional Data Sets with Application to Reference Matching. in Knowledge Discovery and Data Mining. 2000.
4. Tripathi R (2021). Interpretive Psychotherapy of Text mining Approaches, Springer Lecture Notes in Networks and System.
5. Zhao, Y. and G. Karypis. Evaluation of Hierarchical Clustering Algorithms for Document Datasets. in CIKM. 2002. McLean, Viginia.
6. MacQueen J, et al. Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, 1967;volume 1, pages 281–297. Oakland, CA, USA.
7. Kaufman L, Rousseeuw PJ. Finding groups in data: an introduction to cluster analysis, vol. 344. John Wiley & Sons; 2009.
8. Park H-S, Jun C-H. A simple and fast algorithm for k-medoids clustering. Expert Syst Appl. 2009;36(2):3336–41.
9. “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise” Ester, M., H. P. Kriegel, J. Sander, and X. Xu, In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, OR, AAAI Press, pp. 226–231. 1996
10. R. Tripathi, V. K. Mishra, V. Kumar, A. S. Sengar, N. K. Pandey and A. K. Mishra, "Influence of Deepfake Terminology on Content-Emerging Threat Reduction," 2023 3rd International Conference on Innovative Sustainable Computational Technologies (CISCT), Dehradun, India, 2023, pp. 1-5, doi: 10.1109/CISCT57197.2023.10351448.
11. Stouky, A., Jaoujane, B., Daoudi, R., &Chaoui, H. (2017, November). Improving software automation testing using jenkins, and machine learning under big data. In International Conference on Big Data Technologies and Applications (pp. 87-96). Springer, Cham
12. Yanfang Ye, Tao Li, Donald Adjeroh, and S Sitharama Iyengar. A survey on malware detection using data mining techniques. ACM Computing Surveys (CSUR), 50(3):41, 2017.
13. M. Ester, H. P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise”, KDD, vol. 96, no. 34, pp. 226-231, 1996.
14. Preeti Baser and Dr. Jatinderkumar R. Saini, A Comparative Analysis of Various Clustering Techniques used for Very Large Datasets, International Journal of Computer Science & Communication Networks,Vol 3(4),271-275.