*Received:* 11 February 2021 | *Revised:* 30 March 2021 | *Accepted:* 31 March 2021
*DOI: 10.37917/ijeee.17.1.9* *Vol. 17 | Issue 1 | June 2021*

Open Access

*Iraqi Journal for Electrical and Electronic Engineering*
*Review Article*

# Local and Global Outlier Detection Algorithms in Unsupervised Approach: A Review

**Ayad Mohammed Jabbar**
Computer Science Department, Shatt Al-Arab University College, Basra, Iraq

**Correspondence**
*Ayad Mohammed Jabbar
Computer Science Department,
Shatt Al-Arab University College, Basra, Iraq
Email: ayadmohammed@sa-uc.edu.iq

**Abstract**
***The problem of outlier detection is one of the most important issues in the field of analysis due to its applicability in several famous problem domains, including intrusion detection, security, banks, fraud detection, and discovery of criminal activities in electronic commerce. Anomaly detection comprises two main approaches: supervised and unsupervised approach. The supervised approach requires pre-defined information, which is defined as the type of outliers, and is difficult to be defined in some applications. Meanwhile, the second approach determines the outliers without human interaction. A review of the unsupervised approach, which shows the main advantages and the limitations considering the studies performed in the supervised approach, is introduced in this paper. This study indicated that the unsupervised approach suffers from determining local and global outlier objects simultaneously as the main problem related to algorithm parameterization. Moreover, most algorithms do not rank or identify the degree of being an outlier or normal objects and required different parameter settings by the research. Examples of such parameters are the radius of neighborhood, number of neighbors within the radius, and number of clusters. A comprehensive and structured overview of a large set of interesting outlier algorithms, which emphasized the outlier detection limitation in the unsupervised approach, can be used as a guideline for researchers who are interested in this field.***

**KEYWORDS: Anomaly detection, Clustering, Classification, Outlier algorithms.**

## I. INTRODUCTION

The performance of any algorithm depends on important factors, such as the parameters, number of iterations, and outliers. The outlier directly affects the accuracy of any algorithm if not identified and removed. It used by different application includes classification, feature selection and clustering [1]–[6]. The outlier is a data item that does not fit into any group in the dataset. Outliers are data objects with low connectivity to their neighbors. Each data object is assigned an outlier degree, which is called the local outlier factor (LOF). The LOF is density measurement factor compares the density of each object in the dataset. Outliers are also data objects with lower density than their neighbors. Outlier detection is a well-studied problem in analysis in supervised and unsupervised approaches, which inevitably has drastic effects on data analysis [7]. Outlier detection is a critical problem in both approaches. For example, classification and clustering are sensitive to the existence of outliers in a given dataset with low accuracy results [8], [9]. Thus, the prediction model does not represent the actual classes because outlier objects may occupy different classes.

The outlier is also a critical issue in data analysis to achieve accurate results. Ignoring contaminated outliers leads to inaccurate estimation and produces weak results. Incorrectly distributed outliers are assumed to be attributed to several reasons, including human error and unusual behavior. From a knowledge discovery viewpoint, outliers are often more important than normal data objects [10]. The benefits of determining outliers can help improve the data quality and produce effective decision systems. For instance, abnormal behavior in records can lead to the identification of suspicious activities, including impersonation, credit card transactions, telecommunication fraud, and unusual behavior in military surveillance. From a medical perspective, outliers can provide information on patients who exhibit abnormal symptoms due to their specific disease or ailment. The outlier problem has been investigated and discussed in different areas, which have generated a set of approaches and methodologies classified on the basis of different criteria. However, no outlier detection approach is suitable for a multi-kind of datasets [11]. The variety of exact outlier

detection methods is significantly different from that of others in the method of dealing with outlier data. These methods depend on the characteristics of the dataset, which is a critical issue if the dataset comprises different levels of density [12].

The analysis of outliers in a static dataset containing a small number of instances is relatively easier compared with that of the dynamic dataset. Outlier detection approaches can be classified into three learning categories considering pre-defined labels: supervised, unsupervised, and semi-supervised learning approaches [13].

Outliers are identified in supervised learning approaches by learning the model based on given pre-labeled data [14]. The approach still suffers drawbacks despite its application in a variety of applications, such as fraud detection and intrusion detection, because real-life applications require pre-labeled data, which are difficult to obtain and lack the inclusion of new types of rare events [11].

In semi-supervised learning approaches, outliers are identified based on learning a single model such as pre-labeled normal data [15]. Although these approaches utilize a single model to determine others, they contain the same drawbacks as supervised learning approaches.

Outliers are determined in unsupervised learning approaches without pre-labeled data [16]–[18]. Outliers can be identified using the standard statistical distribution model or the nearest neighbor model, which relies on the similarity between points. These approaches are effective and suitable because of their capability to identify outliers based on the characteristics of neighborhoods. This methodology does not require pre-labeled data, which, in most cases, is difficult to find in practical applications. The mentioned learning approaches are broadly classified into the following four major categories as shown in Fig.1: statistical-based outlier detection approach, classification-based outlier detection approach, proximately-based outlier detection approach, and clustering-based outlier detection approache. The taxonomy represents a hierarchical tree includes supervised approach and unsupervised approach. The supervised approach is famous approach determines the outlier according to predefined class such in classification-based outlier detection approach, or use some measurements to draw the deviation of each objects according to all objects in the dataset. The unsupervised approach contains algorithms identifies the outlier objects without requires predefined class. It uses distance measurements between the objects as indicator of outlier or not. In this approach there are two important kind of outlier required to be identified includes the local and global outlier. The big difference between both that the global has high deviation and easy to know because does not sharing enough information with the rest of objects, while the local outlier hard to be identified where its information is similar to normal objects. This study focused in both types showing the most important researches in the literature. The contribution of this study shows how the setting of unsupervised approach parameters effects of the final result produced by this approach.

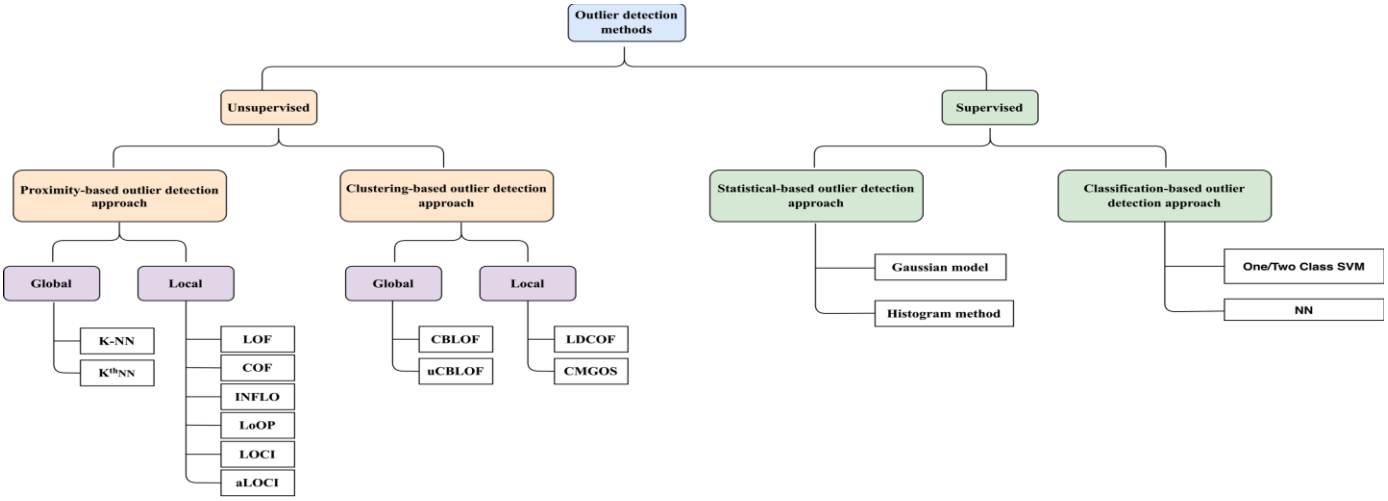


Fig. 1: Taxonomy of the anomaly detection algorithm.

## II. Statistical-Based Outlier Detection Approach

This approach is the earliest outlier detection method used in outlier analysis. The entire data are modeled as a statistical distribution model [19]. The process of identifying outliers can be performed by determining the degree of deviation [15]. The statistical approach operates in two phases: training and testing phases. The training phase involves the construction of a statistical model, while the testing phase evaluates the instances considering the model generated in the training phase. The training phase is regarded as a classification phase due to the different ways of comprising the statistical model estimation. The technique can be considered a semi-supervised technique because it uses the statistical model. Furthermore, this technique is an unsupervised approach because the observations are suitable in fitting the presented statistical model. The importance of the training phase is important and requires more attention because it considered to be the core of this approach because

it constructs the statistical model, which is sufficient to capture the data distribution. The necessary model fitting techniques in the training phase can be classified into parametric and non-parametric methods [20].

The parametric methods estimate the distribution parameters from the given sample based on one statistical distribution for a given dataset. Parameters calculate the means and covariance of the original data [19]. These methods identify outliers based on the deviation of objects in the data model. The advantage of these methods lies in the suitability to use real-life applications if the settings of the previously determined parameters and the data distribution model are known a priori. The most well-known parametric methods include the Gaussian model-based and regression model [21]. The Gaussian model determines outlier data when data originate from known distributions. The performance of the training phase requires mean and covariance estimation using maximum likelihood estimates. Several statistical discordancy tests are developed to evaluate the distribution assumed by the analyst. These statistical discordancy tests are conducted to discover whether the distribution is optimal or near-to-optimal [22]. The most commonly known outlier tests for Gaussian distributions are the mean-variance and box-plot tests [23], [24]. In the mean-variance test for a Gaussian model, the points with standard deviations more than or equal to three are considered to be outlier data. This constant value is regarded as a threshold and provides indicators to show the significant deviation of a point away from the data model. This test is generic and can thus be applied to Student-t and Poisson distributions, which respectively have fatter and longer right tails than a normal distribution.

Another statistical discordancy test is called Grubb's test [25], which assumes normal distribution to detect outliers in a univariate dataset. This process aims to identify one single point as an outlier based on mean and standard deviation. The threshold is determined on the basis of the upper critical value of the t-distribution test. Evaluation is performed on the basis of the $G$ value, which is calculated using Grubbs test statistic; if $G$ is larger than the threshold, then the point can be determined as an outlier, and an elimination process is performed to remove this outlier from the dataset. This procedure is performed iteratively until no further outliers are detected. The box-plot test includes five attributes to depict the distribution: the smallest, median, and largest values of the observation as well as the lower median quartile (Q1) and median upper quartile (Q3). The test values (Q3-Q1) are used to produce an interquartile range (IQR). IQR indicates the range of the lower and upper boundaries, which are used to evaluate the observation based on the obtained boundary. Hence, the point does not belong to the boundary determined as an outlier [23], [26].

Non-parametric methods identify outliers without making any assumptions regarding the statistical properties of the data. Outliers are deducted on the basis of the distance between observations in the full-dimensional space. The points that are distant from their neighbors in the dataset are considered to be outliers. The histogram method is regarded as a non-parametric approach to building a profile fitting the original data [27]. Techniques used to construct histograms based on the data frequency are available. Outliers are determined on the basis of the difference between new tested instances and the histograms. The difference is defined on the basis of how the histograms are built in the training phase. The difference can be identified using three possible ways: histograms constructed on the basis of normal data only, histograms constructed on the basis of outlier data only, and histograms constructed on the basis of the majority of normal data. In the first method, histograms maintain only labeled data. The test phase evaluates test instances considering the histograms to identify whether these instances fall into one numerous normal bins; otherwise, the object is considered to be an outlier [28], [29]. In the second method, data representation in the histogram maintains only outlier data. A test instance that does not fall into any one of the populated bins is labeled as normal data; otherwise, this instance is considered an outlier [30]. In the third method, histograms are constructed on the basis of a mixture of normal and abnormal data. This method is regarded as a typical case due to the hegemony of normal data relative to the abnormal dataset. Thus, the histogram constructs are based on the approximated majority of normal data. The testing phase calculates the ratio of frequency bin sparsity in the histogram against the average frequency of all the bins in the histogram. Points considered to be outliers are dependent upon the position when falling into sparse bins.

## III. Classification-based Outlier Detection Approach

The classification-based outlier approach is a machine learning method used to learn (i.e., training) labeled data instances and then classify them into a new test instance (i.e., testing) into one of the learned classes. This approach can be classified into supervised and semi-supervised techniques [12]. The training phase in the supervised technique constructs a classification model based on all available labeled training data. Thus, the classification model contains normal and abnormal labeled training data called the two-classifier class. The testing phase classifies a new test instance according to the learned model, which contains both labeled training data. In the semi-supervised technique, the training phase trains one of the classes, whether normal or abnormal, to identify the boundary around the defined class called the one-classifier class. The testing phase classifies a new test instance to learn the class, which is usually the normal class. Classification-based supervised techniques categorize a new test instance to one of the training classes as shown in Fig.2 (A). Meanwhile, a new test instance is considered an outlier in the semi-supervised technique if this instance does not belong to a training class as shown in Fig.2 (B). However, both techniques are useful for finding labels in real-life applications. Cases where only outlier class labels are available exist. Fig.2 shows the use of both techniques in identifying outliers. However, there is an extends approach knows as multi class classifier is used in recent years. This approach is similar to one class classifier (see Fig.2 B) but with multi classes distributed on the search space. Each class has own characteristic where high compactness between the

members of each class while outliers are distribute away [31].

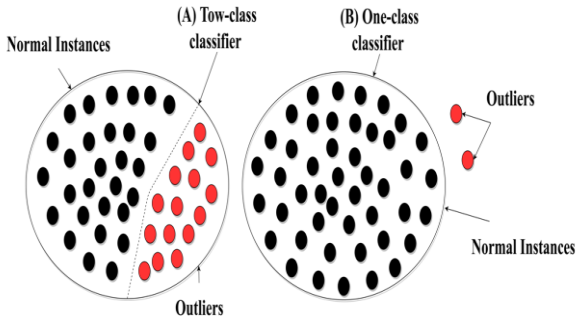


Fig. 2: Classification-based outlier detection approach [12].

Both techniques paved the way for the later initiation of subsequent techniques. The classification-based outlier detection approach is classified into several subcategories, such as SVM, LDA, KNN and NN.

One of the famous classifiers takes as example in this article is SVM. The SVM is a machine learning technique that classifies data into different classes by mapping data into the feature space. SVM is used in [32] as a supervised technique for the detection, and the capability of the unsupervised learning-based technique is examined to learn the normal and abnormal data based on the density regions of data [32]. The authors of this work considered the normal data present in regions with high density, while the low density was regarded as outlier data. The testing phase classifies a new test instance according to the region and declares the instance as normal or outlying accordingly.

NN is one of the most widely used classifiers [33]. This network can classify data using the data distribution model autonomously. Any instance rejected by the network is considered in this approach to be an outlier point. The network learns the weights of normal training data and then feeds the new test instance. Thus, the test instance can be considered an outlier. NN is also widely applied in several domains, such as intrusion, credit card fraud, and image sequence data [34]. In intrusion, a backpropagation is learned on normal data, which are a set of known commands used to identify who executes the commands. The testing phase identifies the class to which the instance belongs. If the test instance does not match any class of network, then this instance signifies an intrusion. However, sufficient suffering is observed in training NNs when dealing with high-dimensional datasets; thus, extra training time is almost always needed.

## IV. Proximity-based Outlier Detection Approach

In this approach, an object considers as outlier when deviates from the proximity of the object within the dataset. This approach can be classified as distance- and density-based according to its model. The former method defines the object that does not contain sufficient objects in its neighborhood as outliers, while the latter defines the density-based outlier object that contains lower density than its neighbors.

Distance-based methods define distance metrics according to the concepts of local neighborhood methods and $k$-nearest neighbors ($k$-NNs). The distance can be computed as a single one between neighbors denoted as $k^{th}$-NN or computed as average distances between neighbors denoted as $k$-NN. This detection method also identifies any point as an outlier relative to the distance with the neighborhood (see Fig.3). Any target object can be identified as an outlier if it contains a large fraction of objects located away from its radius, as presented in Knox and Ng (1998) [35]. They calculated the number of neighbors denoted as $k$ to identify the object as a normal object or belongs to outlier. Distance-based methods rely on the two pre-defined parameters of radius $R$ and the number of neighbors knows as $P$. Determining the former parameter is difficult for the user. Giving too small a value leads to all objects being treated as outliers. On the other hand, giving too large a value leads to all objects being detected as normal objects. Parameter $P$ determines the number of neighbors that should be located in a specific radius. However, finding suitable settings for these parameters can involve numerous trials and errors. Moreover, the algorithm does not provide a ranking to outlier objects but simply labels them as normal or abnormal objects.

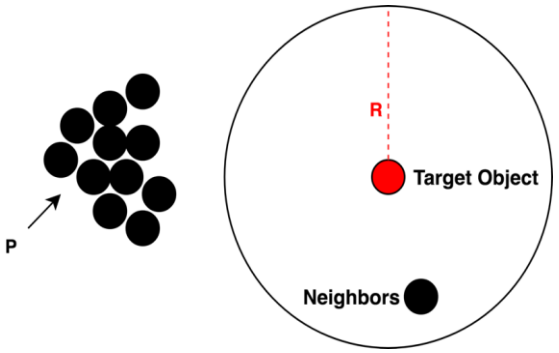


Fig. 3: Example of the distance outlier detection process.

In distance-based methods, the number of objects is calculated through the following three strategies: nested-loop, index-based, and cell-based strategies. Least neighborhood connectivity, which is denoted by $DB(P, R) - Outlier$, is used to idenify the objects. First, the index-based method calculates the number of objects within radius $R$ relative to the fraction $P$ in Fig 3 which set by the user. Let $N$ be the number of objects in dataset $T$, and $F$ is the function present in the distance between the pair objects in $T$. For an object $O$ in the $D$-neighborhood of $Q$, $O \in T$ is located within distance $D$ of object $O$ (i.e., $\{Q \in T | F(O, Q) \leq D\}$). Such as example in Fig 3 an object in red color denoted as target object required to identified as outlier or not. Firstly, the algorithm should calculate the minimum number of neighbors denoted as $M$ are located in target object (within R) as shown in Fig.3, whereas $M$ is calculated as $M = N(1 - P)$. The target object is an outlier if it contains less

than $M$ value in its local neighborhood. The index-based method also calculates a range of searches with radius $R$ for each target object if the value of $(M = M + 1)$ neighbors are available in the $D$-neighborhood of the object. Otherwise, the object is considered an outlier. The search range quickly reduces to $O(N)$ as the number of dimensions or attributes $K$ increases, giving the best constant time improvement.

Second, the nested-loop methods avoid the cost required in the index method to find outliers. The method builds a buffer of part in percentage of the dataset size denoted as $B$%, which is divided into the following two parts: first and second arrays. The method then breaks the dataset into blocks and moves two blocks progressively to the arrays. Moreover, the method calculates the distance between each pair of objects inside the single array and between the arrays. The algorithm then counts the distance of object $O$ relative to $D$-neighborhood and stops the counting whenever the number of $D$-neighborhood exceeds $M$. The complexity of the algorithm will be $O(KN^2)$ despite the reduction in time spent in the comparative process. Third, the cell-based algorithm performs a partitioning to convert the dataset into a set of cells. Subsequently, the algorithm performs a pruning to objects not belonging to outlier objects before finding outliers. The pruning mechanism aims to reduce the time of the search process to find outlier objects in the given dataset. The time complexity of this cell-based algorithm is $O(C^d + N)$, where $C$ is a number that is inversely proportional to $D$. Thus, this algorithm is more linear compared with the two previous algorithms.

An outlier definition based on $k$-NNs to measure the distance between objects is presented to avoid the time complexity and the lack of ranking objects [36]. The definition efficiently ranks each object based on distance without requiring a distancing parameter $D$ as a pre-defined parameter. This definition requires users to provide the total number of outliers $n$ that they are interested to discover. These outliers will be stored in accordance with the distance from one object to another. High priority is given to long distance.

Three algorithms are proposed as extensions of the work proposed by Knox and Ng (1998) [35] to compute $D_n^k$ outliers includes nested-loop, index-based, and partition-based algorithms. The nested-loop algorithm computes the distance between object $p$ and its $k$-NN, iteratively and synchronously updating its nearest neighbor $D^K(P)$. The algorithm updates the current value of $D^K(P)$ when the new distance value is less than the currently recorded one. Morever, the algorithm produces $n$ points containing the largest $D^K(P)$ values, which are considered to be the outliers. One of the disadvantages of the algorithm is its high computational complexity, thus requiring $O(KN^2)$ distance computations. Thus, this algorithm is expensive in the high-dimension space. The index-based algorithm reduces the complexity and spatial index structure; for example, R* tree is obtaied by calculating the distance between object $p$ and its $k$-NN. The algorithm stores objects based on the distance in similar subtrees. The basic idea of clustering two groups of objects is used in the partition-based algorithm. Thus, objects that have long distances will be further away from homogeneous groups. However, the three algorithms require a pruning strategy to work effectively. Pruning partitions strategies are applied to decrease the computational complexity during the search for neighbors and outliers [37].

In the Pruning Partitions During Search for Neighbors (PPSN) strategy, partitions are pruned on the basis of the distance between object $p$ and its $k$-NN during searching for $D^K(P)$. This strategy employs the minimum bounding rectangle (MBR) to embed all objects that belong to a given node. The MBR is a spatial structure that represents the smallest hyper-rectangle. During the search for $D^K(P)$ of object $p$, if the distance between $p$ and objects belonging to the rectangle is longer than the current $D^K(P)$, then none of the objects in the partition belong to the $k$-NN of $p$. Approximate nearest-neighbor search prunes objects containing short distances relative to list candidate $n$ [36]. During the search of $D^K(P)$ for object $p$, if the distance between $p$ and neighbor $q$ is smaller than the shortest distance recurring in the list candidate $n$ (denoted by $D_{min}^k$), then point $p$ will be pruned as an outlier.

By contrast, PPSN is proposed as a preprocessing step. This strategy prunes the partitions without outliers. This step can be conducted using data space partition through a clustering algorithm. Partitions that have remarkably few points will remain using MBR. Lower and upper statistical values are used in each partition to identify partitions that can be pruned on the basis of the $D_{min}^k$ value. These strategies are employed in myriad data processing and analytic workloads to accelerate the performance of the algorithm. An example from the user can be used in discovering the hidden user view of outliers [38]. This strategy uses characteristics inherited from distance-based and example-based methods for identifying outliers. The algorithm selects the most suitable subspace-based genetic algorithm (GA) to isolate user examples more significantly than any other subspaces. The examples isolate more than outliers. Thus, objects related in characteristics are more similar to the examples and are recognized as outliers. Each solution is evaluated using GA to find the best. However, the drawback of this work lies in the random selection of parameters $D$ and $P$.

A research in 2015 [39] shows that outlier objects have low density relative to normal objects containing high density in a dataset because they are substantially close to each other. Thus, the majority of data are considered normal objects. Selected random sampling from the dataset has higher proportionality than outlier objects. An observability factor (OF) strategy is proposed to measure the proportion for each object. The strategy selects random samples $m$ from the dataset. The random samples are then used iteratively to examine a part of the data space. In each iteration, $D^K(P)$ each random sample is determined on the basis of identical radius. Objects not belonging to the examined space will be considered outliers in the corresponding iteration. Evaluation showed performance improvement but lacked identification of the best $K$.

Distance-based methods failed in detecting local outliers. The problem arises in the presence of data containing different clusters of densities [40]. The authors successfully employed different classes of algorithms that focused on

density, which is denoted as a density-based method, to avoid this problem. One of the frequent density-based methods proposed in the literature is called the LOF method [41]. The current study is the first to introduce the idea of local anomalies. An outlier is measured in this method based on a new concept denoted as LOF. This factor measures the degree of density between an object and its neighborhood objects to determine outlier objects. This work has been extended in [42], which introduced connectivity-based outlier factor (COF). The new factor can avoid the limitation of LOF in the linear correlation. Moreover, COF is effective in detecting the nonlinear anomaly. Fig.4 shows the comparisons between LOF and COF in dataset has two attributes of a linear dependency. The results reveal that COF is stable in identifying anomalies. COF is better than LOF in identifying outlier objects with dataset has low density, this is because COF use the distance between the point to the points in its neighborhood [43]. The COF determines how the object in the search space is isolated to other objects in the dataset and how is it far to be considered an aberration. Other differences between both algorithms that LOF use Euclidean distance to calculates the nearest neighbors, while COF use the short path method called chain distance to calculates the nearest neighbors.

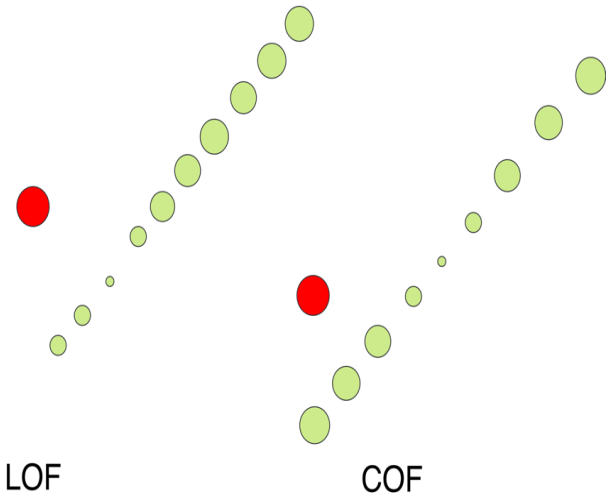


Fig. 4: Comparison of COF with LOF using linear correlation.

Breunig et al. improved the LOF by defining the influenced outlier-ness (INFLO) factor, which can be computed on the basis of neighbors and reverse neighbors [44]. The new factor is more effective when a dataset contains different levels of densities and demonstrates closeness to each other. The new factor is proposed due to the failure of LOF in scoring the objects located in the cluster border. The algorithm is realized on k-NNs and the reverse nearest-neighborhood set as shown in Fig.5. This indicates that LOF can identify red objects within six neighbors residing in the same radius only, while INFLO considers the blue objects as neighbors for red objects. These concepts reveal that the red objects have a low probability to be considered as an anomaly by INFLO. However, LOF and INFLO methods are substantially sensitive to parameter setting, which is difficult to initialize with an appropriate parameter $k$ [45].

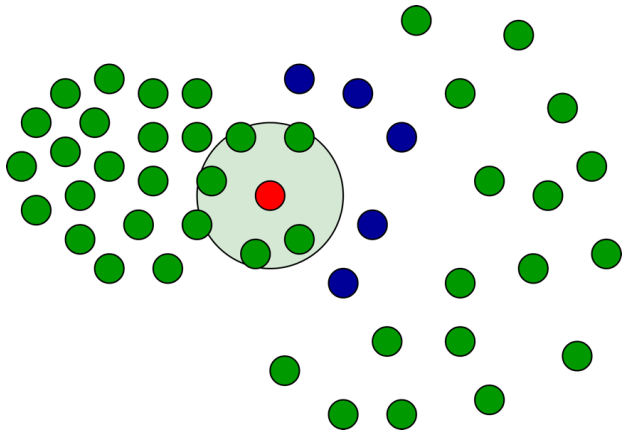

Fig. 5: Comparison of INFLO with LOF considering objects located in the clustering border.

Other research introduced a Local Outlier Probabilities (LoOP), which is a new factor based on probability score. LoOP proposed to solve the problem of the LOF factor, which assigns a similar score to different objects located in various locations [46]. The LoOP follows the previous method in its calculation when the neighborhood density used as the main factor in the outlier estimation. However, the density is computed differently, and the distances to the nearest neighbors follow a Gaussian distribution. The LoOP results are more accurate than those of the LOF method. LoOP is also similar to the previous local algorithms because it is insensitive to parameter $k$. Comparison between LoOP and the COF showed that the former produced better results because LoOP outputting an anomaly probability instead of a score, which might also result in better comparison of anomalous records between different datasets [47].

The authors in [13] introduced an instability factor (INS) based on the concept of the gravity center. The INS aims to solve the problem of detecting local outliers and low-density patterns in distance-based and density-based methods, respectively. This strategy ranks objects as normal or abnormal based on the center of gravity movement by examining such movement via increasing the number of objects repetitively. If the variation in the location is considerably small, then the objects will be considered normal data. Otherwise, these objects will be regarded as abnormal data. This concept is employed in the detection of local outliers and global outlier objects. This method produced robust results and performance insensitive to parameter $k$, which outperformed the existing approaches, including $k$-NN and LOF. The algorithm still failed to determine local outliers despite its insensitivity to parameter $k$.

Outlier objects are located in regions relatively far from objects of high density [48]. Local minima density outlier factor (LMDOF) is proposed to solve the problem of parameter $k$, which influences the performance of the distance- and density-based approaches. The LMDOF method measures the degree of an outlier object based on its

local density, region, and distance to other regions containing high density. Compared with LOF and OF, the LMDOF method expresses good stability in different parameters. However, the range of parameter $k$ is insufficient to guarantee stability.

The problem of the LOF algorithm in dealing with different density regions is addressed by the proposed rank-based detection algorithm [49]. The ranking of the distances between objects is based on the distance of object $O$ in $k$-neighborhood (in its neighborhood). This algorithm ranks each object by classifying objects into important and not important based on closeness to the $k$-neighborhood. If object $p$ belongs to dataset $D$ and $q$ is one of the neighbors of $p$, then $q$ and $p$ will be considered close. Thus, the distance $d\ (p, q)$ relative to $d\ (q, O)$ is ranked for all $o \in D$.

## V. Clustering-based Outlier Detection Approach

The clustering-based approach to detection, which is often categorized as the classification problem, mainly aims to find clusters and outliers [50]. The detected outliers can be removed to produce reliable clustering [51]. The clustering-based approach implicitly identifies outliers as objects that are located far from other clusters. This approach does not explicitly rank objects as outliers. Example algorithms of this approach include density-based spatial clustering of applications with noise (DBSCAN) [52], clustering large applications based upon randomized search [53], CHAMELEON [54], BIRCH [55], and clustering using representatives [56]. Clustering normally produces clusters with no attention to the outlier detection. The process of detecting outlier objects is inefficient [45]. Therefore, this process uses axillary algorithms, such as K-means, to perform such processes. For example, research in 2011 applied the K-means algorithm to deal with outlier objects [57]. This algorithm divides the dataset into clusters. Objects that are close to each other in obtained clusters are pruned to accelerate the algorithm, and the remaining objects are then calculated on the basis of an outlier score. The outlier score declares the top $n$ outlier list according to the degree of the score. However, the result is weak due to the difficulty in initializing the number of clusters $k$ by the user. Thus, the final results of the algorithm are based on the appropriated $k$.

Cluster-based local outlier factor (CBLOF) is one of the commonly used anomaly detection algorithms for the clustering approach in identifying outlier objects [58]. The outlier factor determines dense areas based on clustering. Any algorithm uses the clustering concepts can generally be used to cluster the data into different groups as a primary step. However, the most used algorithm in the literature is K-means because it maximizes the low computational complexity. The next step employs CBLOF to classify the clustering results produced by the clustering algorithm into two groups either large or small clusters. The CBLOF anomaly score computes the distance between every object, and its cluster center is multiplied by the instances belonging to its cluster. Meanwhile, the distance in small clusters is computed between every object, and the closest large cluster is used. CBLOF is then later extended as a new method called unweighted CBLOF (uCBLOF). The uCBLOF is effective in estimating the local density of the clusters. Another extinction called local density cluster-based outlier factor (LDCOF) is proposed in 2012 to address the shortcoming of uCBLOF by estimating the densities of clusters assuming a spherical distribution of the cluster members [59]. K-means is employed to cluster the data into different small and large clusters. The average distance between all objects of a cluster and its centroid is computed to score the objects by dividing the distance of an object to its cluster center by the average distance. However, similar to previous density-based outlier detection, CBLOF, LDCOF, and uCBLOF are sensitive to the number of initial clusters $k$, which is also a critical parameter directly affecting the results.

The clustering-based algorithm handles outliers based on the unsupervised extreme learning machine (UELM) clustering algorithm which is a classification algorithm employed extreme learning machine [60]. The algorithm produced better accuracy than SVM in classification. This algorithm divides datasets into $k$ clusters, and each $k$ cluster contains numerous objects that are close to each other and their centroids. The ranking of objects as outliers involves the computation of each object according to its cluster because the clusters have different densities. A pruning strategy should be used to improve the searching speed of *KNNs*. The results of the proposed method are compared with those of naive methods considering runtime rather than quality of results. UELM determines the final results based on the number of clusters. Liangi (2010) applies an agglomerative clustering algorithm to construct a hierarchical tree that shows global outliers at the top of the tree [61]. However, this algorithm does not show a justification threshold, which can determine the top tree outliers. Research in 2017 used K-means to remove outlier objects from the dataset [62]. K-means calculates a threshold and identifies the small generated clusters as outlier objects based on the said threshold. This approach is ineffective because the number of clusters is pre-defined values. DBSCAN can implicitly identify outliers during its run. This algorithm is time consuming and sensitive to the parameters, which determine the final clustering result and outlier objects [63]. DBSCAN also constructs hierarchy trees based on the high dependency of the clustering results, which may worsen due to poor parameter setting. Other similar studies that use the divisive hierarchical clustering algorithm have been employed to divide the dataset into K-partition according to a specific number of clusters initialized by the researcher [64]. However, the accuracy of detecting the right outlier objects is based on the number of clusters produced. Thus, this study required other preprocessing steps to identify the right number of clusters. Other related studies in 2017 proposed the use of outlier detection using the K-means algorithm and fuzzy modeling. Both algorithms reveal the use of LOF to score the degree of the local outlier based on the degree of becomes as membership. However, the number of outliers should be initialized by the research as predefined values, which become difficult when the datasets have overlapping classes [65]. Gan and Ng (2017) also proposed to use the K-means algorithm by dividing the dataset into

different numbers of clusters and identifying the outlier objects by introducing an additional cluster containing only the outlier objects simultaneously [66]. Zhao et al. (2018) represents an adaptive algorithm consist of three algorithms include K-means, LOF and Gaussian distribution [67]. However, different numbers of clusters can provide different numbers of outliers in both algorithms, and the algorithm fails to find the local outliers. In 2018, the authors proposed an unsupervised approach for outlier detection in a sequence dataset [68]. The proposed approach combines sequential pattern mining, cluster analysis, and a minimum spanning tree algorithm to identify clusters of outliers. Initially, sequential pattern mining is used to extract frequent sequential patterns. Next, the extracted patterns are clustered into groups of similar patterns. Finally, the minimum spanning tree algorithm is used to find groups of outliers.

Other studies proposed an algorithm employs the hierarchical clustering to identify outliers in circular regression models by using the single-linkage method as a similarity method [69]. The algorithm merges data with the shortest distance until one single cluster is established in a tree of sub-clusters. This strategy cuts the tree at a certain point because the last level of the tree is considered to be an outlier. However, the last level is not always an outlier because the dataset comprises different levels of densities. Thus, this strategy fails in determining outlier objects. The objects in the top level of the tree will be considered global outlier objects. Therefore, detecting local outlier objects located in the low levels of the tree is unsuitable. The MST-based clustering algorithm is proposed by John Peter to identify the outliers using the clustering principle. This principle considers the small clusters as outliers, while the rest of the objects in the remaining clusters can be detected accruing to the distance between each cluster and the centroid. The number of clusters is detected in accordance with the new validation criterion based on geometric property. However, the geometric property is substantially sensitive to its parameters, which need additional experiments to identify the right number of clusters. However, the identification of the wrong number of clusters causes inaccurate outlier detection [70]. Table I summarizes the existing work in proximity- and the outlier-based clustering approaches.

TABLE I
SUMMARY OF EXISTING RESEARCH IN PROXIMITY-BASED OUTLIER APPROACHES /CLUSTERING-BASED OUTLIER APPROACHES

| Reference | Problems | Objective | Weakness |
|---|---|---|---|
| [41] | Identify local density outlier objects | Outlier measurement based on a new concept denoted as LOF | Sensitive to parameter $K$ |
| [42] | Limitation of LOF in the linear correlation | Use of chaining distance | Sensitive to parameter $K$ |
| [44] | Limitation of LOF in dealing datasets containing different levels of densities | Outlier object computation based on neighbors and reverse neighbors | Sensitive to parameter $K$ |
| [38] | Discovery of the hidden user view of outliers | Identification of outlier based on user examples | Randomly selected parameters $D$ and $P$ |
| [46] | Solution of the LOF factor, in which a similar score is assigned to different objects located in various locations | Proposal of a new factor based on probability score | Sensitive to parameter $K$ |
| [39] | Increase identification of outlier objects | Selection of random sampling | Sensitivity of parameter $K$ to user input |
| [40] | Local outliers | LOF measurement of the degree of density between an object and its neighborhood objects | Sensitivity of parameter $K$ to user input |
| [58] | Discovery of outlier objects based on clustering algorithm | Employment of the K-means algorithm to cluster the data into long and small clusters to identify density cluster | Sensitivity of parameter $K$ to user input |

| | | | |
|---|---|---|---|
| [59] | Solution to the shortcoming of uCBLOF | Estimation of the cluster densities assuming a spherical distribution of the cluster members | Sensitivity of parameter $K$ to user input |
| [13] | Local outliers | Proposal of factor (INS) based on the center of gravity concept | Algorithm determines global outliers, fails to determine local outliers |
| [48] | Local outliers | Proposal of LMDOF to handle the parameter $k$ problem | Parameter $K$ is insufficient to guarantee the performance of LMDOF |
| [49] | Outliers determined in different density regions | Proposal of rank-based detection algorithm to solve the LOF algorithm problem in dealing with different density regions | Sensitivity of parameter $K$ to user input |
| [60] | Ranking objects as outliers | Handling outliers based on the UELM clustering algorithm | Results are sensitive based on the number of predefined clusters |
| [61] | Detecting global outliers | Application of the agglomerative clustering algorithm to construct a hierarchical tree, which shows global outliers in the top | Predefined threshold determines the top of the tree as outliers |
| [62] | Identifying outlier objects in datasets | Removal of outlier objects from the dataset based on the K-means algorithm | Results are sensitive based on user number of clusters as input |
| [63] | Identifying outlier objects in datasets | Proposal of hierarchical-based DBSCAN algorithm (HDBSCAN) | Parameter settings are sensitive |
| [64] | Identifying outlier objects in datasets | Proposal of divisive hierarchical clustering for determining outliers | Predefined threshold determines the top of the tree as outliers |
| [65] | Identifying local outlier objects in datasets | Use of K-means and fuzzy modeling | Sensitivity of parameter $K$ |
| [66] | Detecting global outliers | K-means identifies outliers as separated single clusters | Sensitivity of parameter $K$ |
| [67] | Local outliers | K-means clustering and multivariate Gaussian distribution | Sensitivity of parameter $K$ |
| [68] | Detecting global outliers | Local outliers using minimum spanning tree algorithm | Parameter settings are sensitive |
| [69] | Identifying outliers in circular regression models | Use of single-linkage methods in determining outliers | Failed to determine local outlier |
| [70] | Detecting global outliers | Local outliers using minimum spanning tree algorithm | Parameter settings are sensitive |

## VI. Conclusions

The main goal of outlier detection is to construct classifiers in supervised and unsupervised approaches. Novelty detection is an important learning paradigm and has drawn significant attention within the research community, as shown by the increasing number of publications in this field. A review of the current state-of-the-art in outlier detection in the unsupervised approach has been presented in the current study, and the supervised approach has demonstrated the main algorithms and characteristics. The unsupervised approach shows that the proximity- and clustering-based outlier approaches are sensitive to parameters. The algorithm in the proximity-based outlier approach explicitly identifies outlier objects, while the clustering-based outlier approach implicitly identifies outliers as objects located far from other clusters during the clustering process. Both approaches suffer in determining local and global outlier objects simultaneously. The clustering approach suffers in identifying local outlier. Different algorithms are used for clustering outliers, where clusters and outliers are identified in the final results. However, the algorithms perform clustering as the main task, in which identifying outlier objects is not the main concern. Moreover, these algorithms do not rank or identify the degree of an outlier or normal objects and require different parameter settings during the test. Examples of such parameters in the proximity-based outlier approach are the radius of the neighborhood and the number of neighbors within the radius. The former parameter determines the range of the neighborhood target, whereas the latter identifies neighborhood density. Meanwhile, the number of clusters is the main problem in the clustering-based outlier approach. Thus, poor parameter setting leads to weak outlier detection. An effective algorithm determines both kinds of outliers, which is the dilemma of outlier detection algorithms. Therefore, if the algorithm is effective in finding global outliers, then it fails to determine local outliers and vice versa. The unsupervised generally requires additional attention and research for its parameters. Thus, parameter tuning is required as an optimization problem to identify local and global outlier objects.

## Conflict of Interest

The authors have no conflict of relevant interest to this article.

## References


[1] A. M. Jabbar, K. R. Ku-Mahamud, and R. Sagban, "An improved ACS algorithm for data clustering," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 17, no. 3, pp. 1506–1515, 2019, doi: 10.11591/ijeecs.v17.i3.pp1506-1515.

[2] A. M. Jabbar, K. R. Ku-Mahamud, and R. Sagban, "Balancing Exploration And Exploitation In Acs Algorithms For Data Clustering," vol. 97, no. 16, pp. 4320–4333, 2019.

[3] H. N. K. AL-Behadili, K. R. Ku-Mahamud, and R. Sagban, "Hybrid ant colony optimization and genetic algorithm for rule induction," *J. Comput. Sci.*, vol. 16, no. 7, pp. 1019–1028, 2020, doi: 10.3844/JCSSP.2020.1019.1028.

[4] H. N. K. Al-Behadili, R. Sagban, and K. R. Ku-Mahamud, "Adaptive parameter control strategy for ant-miner classification algorithm," *Indones. J. Electr. Eng. Informatics*, vol. 8, no. 1, pp. 149–162, 2020, doi: 10.11591/ijeei.v8i1.1423.

[5] H. Almazini and K. R. Ku-Mahamud, "Grey Wolf Optimization Parameter Control for Feature Selection in Anomaly Detection," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 2, pp. 474–483, 2021, doi: 10.22266/ijies2021.0430.43.

[6] H. Al-Behadili, "Stochastic Local Search Algorithms for Feature Selection: A Review," *Iraqi J. Electr. Electron. Eng.*, vol. 17, no. 1, pp. 1–10, 2021, doi: 10.37917/ijeee.17.1.1.

[7] F. W. Young, P. M. Valero-Mora, and M. Friendly, *Visual Statistics: Seeing Data with Dynamic Interactive Graphics*. 2011.

[8] J. A. S. Almeida, L. M. S. Barbosa, A. A. C. C. Pais, and S. J. Formosinho, "Improving hierarchical cluster analysis: A new method with outlier detection and automatic clustering," *Chemom. Intell. Lab. Syst.*, vol. 87, no. 2, pp. 208–217, 2007.

[9] Mansoori and Eghbal, "GACH: A grid-based algorithm for hierarchical clustering of high-dimensional data," *Soft Comput.*, vol. 18, no. 5, 2014.

[10] K. Singh and S. Upadhyaya, "Outlier Detection: Applications And Techniques.," *Int. J. Comput. …*, vol. 9, no. 1, pp. 307–323, 2012.

[11] Y. Zhang, N. Meratnia, and P. Havinga, "A taxonomy framework for unsupervised outlier detection techniques for multi-type data sets," *Computer (Long. Beach. Calif).*, vol. 49, no. 3, pp. 355–363, 2007.

[12] Zhang, "Advancements of Outlier Detection : A Survey," *ICST Trans. Scalable Inf. Syst.*, vol. 13, no. 01, pp. 1–26, 2013.

[13] J. Ha, S. Seok, and J. S. Lee, "Robust outlier detection using the instability factor," *Knowledge-Based Syst.*, pp. 15–23, 2014.

[14] H. N. K. Al-Behadili, K. R. Ku-Mahamud, and R. Sagban, "Ant colony optimization algorithm for rule-based classification: Issues and potential solutions," *J. Theor. Appl. Inf. Technol.*, vol. 96, no. 21, pp. 7139–7150, 2018.

[15] S. S. Rakhe and A. S. Vaidya, "A Survey on Different Unsupervised Techniques to Detect Outliers," *nternational Res. J. Eng. Technol.*, pp. 514–519, 2015.

[16] M. Gupta, J. Gao, and C. C. Aggarwal, "Outlier Detection for Temporal Data : A Survey," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 1, pp. 1–20, 2013.

[17] V. J. Hodge and J. I. M. Austin, "A Survey of Outlier Detection Methodologies," no. 1969, pp. 85–126, 2004.

[18] A. M. Jabbar, K. R. Ku-Mahamud, and R. Sagban, "Ant-based sorting and ACO-based clustering approaches: A review," in *IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, Apr. 2018, pp. 217–223.

[19] N. Shahid, I. H. Naqvi, and S. Bin Qaisar, "Characteristics and classification of outlier detection

techniques for wireless sensor networks in harsh environments: a survey," *Artif. Intell. Rev.*, vol. 43, no. 2, pp. 193–228, 2012.
[20] M. Nayak and P. Dash, "Distance-based and Density-based Algorithm for Outlier Detection on Time Series Data," *Appl. Sci. Adv. Mater. Int.*, pp. 139 – 146, 2016.
[21] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection : A Survey," *ACM Comput.*, vol. 41, no. 3, pp. 1–58, 2009.
[22] V. Barnett, "The Ordering of Multivariate Data," *J. R. Stat. Soc.*, vol. 139, no. 3, pp. 318–355, 1976.
[23] J. Laurikkala, M. Juhola, and E. Kentala, "Informal identification of outliers in medical data," *Fifth Int. Work. Intell. Data Anal. Med. Pharmacol.*, pp. 20–24, 2000.
[24] H. E. Solberg and A. Lahti, "Detection of Outliers in Reference Distributions : Performance of Horn ' s Algorithm," *Gen. Clin. Chem.*, pp. 1–7, 2005.
[25] F. E. Grubbs, "Procedures for Detecting Outlying Observations in Samples," *Technometrics*, vol. 11, no. 1, pp. 1–21, 1969.
[26] P. S. Horn, L. Feng, Y. Li, and A. J. Pesce, "Effect of Outliers and Nonhealthy Individuals on Reference Interval Estimation," *Dep. Math. Sci. Univ. Cincinnati, Cincinnatinati*, vol. 2145, pp. 2137–2145, 2001.
[27] G. Tang, "New methods in outlier detection," Simon fraser university, 2015.
[28] D. Anderson, T. Frivold, A. Tamaru, and A. Valdes, "Next-generation intrusion detection expert system (nides), software users manual, beta-update release," *Tech. Rep.*, 1994.
[29] H. S. Javitz and A. Valdes, "The SRI IDES statistical anomaly detector," *Proceedings. 1991 IEEE Comput. Soc. Symp. Res. Secur. Priv.*, pp. 316–326, 1991.
[30] E. Eskin, "Anomaly detection over noisy data using learned probability distributions," *Seventeenth Int. Conf. Mach. Learn. Proc.*, pp. 255–262, 2000.
[31] S. Upadhyay and K. Singh, "Classification Based Outlier Detection Techniques," *Int. J. Comput. Trends Technol.*, vol. 3, no. 2, pp. 294–298, 2012.
[32] I. Steinwart, D. Gov, C. Scovel, and J. Gov, "A Classification Framework for Anomaly Detection Don Hush," *J. Mach. Learn. Res.*, vol. 6, pp. 211–232, 2005.
[33] P. Sykacek, "Equivalent Error Bars For Neural Network Classifiers Trained By Bayesian Inference," *... Eur. Symp. Artif. Neural ...*, pp. 1–7, 1997.
[34] S. Agrawal and J. Agrawal, "Survey on Anomaly Detection using Data Mining Techniques," *Procedia - Procedia Comput. Sci.*, vol. 60, pp. 708–713, 2015.
[35] E. M. Knox and R. T. Ng, "Algorithms for Mining Datasets Outliers in Large," *Proc. 24th Int. Conf. Very Large Data Bases*, pp. 392–403, 1998.
[36] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 427–438.
[37] G. H. Orair, C. H. C. Teixeira, W. M. Jr, B. Horizonte, and Y. Wang, "Distance-Based Outlier Detection : Consolidation and Renewed Bearing," *Proc. VLDB Endow.*, vol. 3, no. 2, 2010.
[38] Y. Li and H. Kitagawa, "DB-Outlier Detection by Example in High Dimensional Datasets," *IEEE Commun.*, pp. 73–78, 2007.
[39] J. Ha, S. Seok, and J. Lee, "A precise ranking method for outlier detection," *Inf. Sci. (Ny).*, vol. 324, pp. 88–107, 2015.
[40] M. A. F. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Processing*, pp. 215–249, 2014.
[41] M. M. Breunig, H. Kriegel, R. T. Ng, and J. Sander, "LOF : Identifying Density-Based Local Outliers," *ACM SIGMOD Int. Conf. Manag. Data*, pp. 93–104, 2000.
[42] J. Tang, Z. Chen, A. W. Fu, and D. W. Cheung, "Enhancing effectiveness of outlier detections for low density patterns," *Adv. Knowl. Discov. Data Min.*, pp. 535–548, 2002.
[43] A. Nowak-Brzezinska and C. Horyn, "Outliers in rules - The comparision of LOF, COF and KMEANS algorithms.," *Procedia Comput. Sci.*, vol. 176, pp. 1420–1429, 2020, doi: 10.1016/j.procs.2020.09.152.
[44] W. Jin, A. K. H. Tung, J. Han, and W. Wang, "Ranking Outliers Using Symmetric Neighborhood Relationship," *Springer-Verlag Berlin Heidelb.*, pp. 577–593, 2006.
[45] J. Huang, Q. Zhu, L. Yang, and J. Feng, "A non-parameter outlier detection algorithm based on Natural Neighbor," *Knowledge-Based Syst.*, pp. 71–77, 2016.
[46] H. Kriegel, E. Schubert, and A. Zimek, "LoOP: Local Outlier Probabilities," *IEEE Commun.*, pp. 1649–1652, 2009.
[47] M. Goldstein and S. Uchida, "A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data," *PLoS One*, no. April, pp. 1–31, 2016.
[48] J. Liu and G. Wang, "Outlier detection based on local minima density," *IEEE Commun.*, 2016.
[49] H. Huang, K. Mehrotra, and C. K. Mohan, "Rank-Based Outlier Detection SYR-EECS-2011-07," *Electr. Eng. Comput. Sci. Tech. Reports*, pp. 1–22, 2011.
[50] M. H. Marghny and A. I. Taloba, "Outlier Detection using Improved Genetic K-means," vol. 28, no. 11, pp. 33–36, 2011.
[51] N. Faraidah, M. Di, and S. Z. Satari, "algorithm for circular regression model The Effect of Different Distance Measures in Detecting Outliers using Clustering-based Algorithm for Circular Regression Model," *3rd ISM Int. Stat. Conf.*, pp. 1–13, 2017.
[52] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," *Proc. 2nd Int. Conf. Knowl. Discov. Data Min.*, pp. 226–231, 1996.
[53] R. T. Ng and J. Han, "Efficient and Effective Clustering Methods for Spatial Data Mining," *Proc. 20th Int. Conf. Very Large Data Bases*, pp. 144–155, 1994.
[54] G. Karypis, E.-H. Han, and V. Kumar, "Chameleon: hierarchical clustering using dynamic modeling," *Computer (Long. Beach. Calif).*, vol. 32, no. 8, pp. 68–75, 1999.
[55] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An Efficient Data Clustering Databases Method for Very Large," *ACM SIGMOD Int. Conf. Manag. Data*, vol. 1, pp. 103–114, 1996.

[56] S. Guha, R. Rastogi, and K. Shim, “CURE: an efficient clustering algorithm for large databases,” in *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, 1998, pp. 73–84.
[57] R. Pamula, J. K. Deka, and S. Nandi, “An Outlier Detection Method Based on Clustering An Outlier Detection Method based on Clustering,” *Second Int. Conf. Emerg. Appl. Inf. Technol.*, pp. 253–256, 2011.
[58] Z. He, X. Xu, and S. Deng, “Discovering cluster-based local outliers,” *Pattern Recognit. Lett.*, pp. 1641–1650, 2003.
[59] M. Amer and M. Goldstein, “Nearest-Neighbor and Clustering based Anomaly Detection Algorithms for RapidMiner,” *Proc. 3rd RapidMiner Community Meet. Conferernce (RCOMM 2012)*, pp. 1–12, 2012.
[60] X. Wang, M. Bai, D. Shen, T. Nie, Y. Kou, and G. Yu, “A Distributed Algorithm for the Cluster-Based Outlier Detection Using Unsupervised Extreme Learning Machines,” *Hindawi Math. Probl. Eng.*, pp. 1–12, 2017.
[61] B. Liangi, “A Hierarchical Clustering Based Global Outlier Detection Method,” *IEEE*, pp. 1213–1215, 2010.
[62] B. Anwesha and L. Dey, “Outlier Detection and Removal Algorithm in K-Means and Hierarchical Clustering,” *World J. Comput. Appl. Technol. 5(2)*, vol. 5, no. 2, pp. 24–29, 2017.
[63] R. J. G. B. Campello and C. Sciences, “Hierarchical Density Estimates for Data Clustering , Visualization , and Outlier Detection,” *ACM Trans. Knowl. Discov*, vol. 10, no. 1, pp. 1–51, 2015.
[64] X. Wang, X. Wang, and M. Wilkes, *New Developments in Unsupervised Outlier Detection*. 2021.
[65] A. Diez-Olivan, J. A. Pagan, R. Sanz, and B. Sierra, “Data-driven prognostics using a combination of constrained K-means clustering, fuzzy modeling and LOF-based score,” *Neurocomputing*, vol. 241, pp. 97–107, 2017, doi: 10.1016/j.neucom.2017.02.024.
[66] G. Gan and M. K. P. Ng, “K-Means Clustering With Outlier Removal,” *Pattern Recognit. Lett.*, vol. 90, pp. 8–14, 2017, doi: 10.1016/j.patrec.2017.03.008.
[67] S. Zhao, W. Li, and J. Cao, “A user-adaptive algorithm for activity recognition based on K-means clustering, local outlier factor, and multivariate gaussian distribution,” *Sensors (Switzerland)*, vol. 18, no. 6, 2018, doi: 10.3390/s18061850.
[68] S. Abghari, V. Boeva, N. Lavesson, H. Grahn, S. Ickin, and J. Gustafsson, “A Minimum Spanning Tree Clustering Approach for Outlier Detection in Event Sequences,” *Proc. - 17th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2018*, pp. 1123–1130, 2019, doi: 10.1109/ICMLA.2018.00182.
[69] S. Z. Satari, N. Faraidah, M. Di, and R. Zakaria, “The multiple outliers detection using agglomerative hierarchical methods in circular regression model The multiple outliers detection using agglomerative hierarchical methods in circular regression model,” *J. Phys. Conf. Ser.*, pp. 1–5, 2017.
[70] S. J. Peter, “Minimum spanning tree based clustering for outlier detection,” *J. Discret. Math. Sci. Cryptogr.*, vol. 14, no. 2, pp. 149–166, 2011, doi: 10.1080/09720529.2011.10698329.