International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Modified Fuzzy C-Means Clustering for Anomaly Detection in Bio-medical Data.

Srikanta Kumar Sahoo[1], Priyabrata Pattanaik[2], Mihir Narayan Mohanty[3]

Submitted: 24/12/2023 **Revised**: 30/01/2024 **Accepted**: 10/02/2024

Abstract*:* Bio-medical data for different diseases are always with noise and outliers. As the source and medium differs from place to place and time to time it happens to be noisy. In this work, the authors have tried to analyse biomedical data statistically using principal component analysis. Here, the fuzzy centroid is modified with opposition learning based algorithm. Due to optimal algorithms, the modified fuzzy c-means utilized for clustering that performs excellent in terms of outlier detection. The data taken from UCI machine learning repository are of classification type. It is shown in the result section that the outliers have been detected successfully.

Keywords: *Anomaly detection, Fuzzy-c-means, Opposition based learning, Outlier detection, Principal component analysis*

1. Introduction

A Finding patterns in records that do not follow expected behavior is known as anomaly detection [1]. These records with unusual behavior are referred to as anomalies, outliers, exceptions, or noise. Anomaly detection in a data set is crucial because they can have a detrimental impact on the outcomes of data analysis and, consequently, judgments based on that analysis. As a result, it is crucial to reduce anomalies prior to data processing. But not all anomalies are attacks/harmful. In many cases, detection of anomalies is also helpful in getting important crucial information and can be used to improve the applications. For example, in order to increase network security, anomaly detection can be used to spot unusual traffic in communication networks. Another example is the credit card business, which can be used in finding unauthorized uses or fraudulent transactions [2]. Thus, detecting anomalies is a significant issue and has been studied in various fields and application domains. Intrusion detection, fraud detection, medical anomaly detection, industrial damage detection, abnormality detection in sensor networks, and image processing are a few significant uses of anomaly detection.

The outlier detection techniques are broadly based on statistics, nearest neighbors, and clustering [3]. The assumption underlying statistical distribution-based approaches is that the data items under study adhere to a particular distribution [4]. The nearest neighbors-based approaches find outliers by calculating the number of neighbors each data object has [5]. Clustering-based anomaly detection is an unsupervised technique. In methods for outlier detection that rely on clustering, anomalies are grouped into small, sparse clusters, whereas regular data members are grouped into big, dense clusters [6]. The ability of the clustering algorithm to group the dataset into several clusters has a significant impact on the efficacy of clustering-based strategies.

In this work, authors concentrate on a clustering-based approach. Several approaches have been used to identify the anomalies with the cluster-based method. The first approach is to split the data into two clusters, with the anomalies belonging to one cluster and the normal data items to another. The Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [7] is an anomaly detection approach following this idea. The second approach is to break the data up into various clusters. Anomalies are the data objects that are located far away from their nearest center. This method is used in the anomaly identification technique proposed in [8] for intrusion detection. The third approach is to divide the data into different clusters, with the abnormal data items belonging to smaller clusters and normal data items to large clusters. The anomaly score called Cluster-Based Local Outlier Factor (CBLOF) is used by the algorithm findCBLOF in [9] to find the anomalies. Further, various application domains clustering-based anomaly detection techniques have been proposed recently in [10, 11, 12]. The majority of recent anomaly identification techniques have substantial computational costs.

The issue in the first approach discussed above is that one out of the two clusters specifies anomalies; it is not practically possible for large domains, so do not always find an optimum number of anomalies. The problem with the second approach is that detecting anomalies within a cluster will not yield the best results if the anomalies themselves create clusters. The question in the third case is on what criteria the cluster will be considered as a small cluster. Moreover, the capability of the clustering algorithm to classify data into normal classes and rare classes (clusters having anomalies) is an important factor. To address the above issues in this research, an anomaly detection technique using opposition learning and fuzzy-c-means clustering is

1 ITER, SOA deemed to be university, Bhubaneswar-751030, India
2 ITER, SOA deemed to be university, Bhubaneswar-751030, India
3 ITER, SOA deemed to be university, Bhubaneswar-751030, India
** Corresponding Author Email: srikantasahoo@soa.ac.in*

proposed. It combines the second and third approaches. First, the opposition learning-based fuzzy-c-means clustering is used to divide the dataset into a set of clusters. Then, it finds the rare classes using the *outlier factor* as a measure. Finally, finds *top-k* outliers across the clusters and computes the *outlier coverage ratio* of the *top-k* outliers to check the efficiency of the algorithm.

The rest part of the paper is organized as follows. Section 2 depicts the literature review. Section 3 presents different methodologies required for the work. Section 4 describes the modified fuzzy-c-means algorithm followed by the proposed anomaly detection method in Section 5. Section 6 shows the experimental results, and finally, Section 7 concludes the paper.

2. Literature Review

Clustering is typically an unsupervised machine learning method that is used to detect anomalies in literature. Despite the fact that clustering and anomaly detection seem to be primarily different from one another, a number of clustering-based anomaly detection algorithms have been developed. The fundamental tenet of these strategies is that anomalous data do not belong to a cluster or remote from the cluster centroid or part of a small, sparse cluster. The clustering approach used for anomaly detection proposed in [13], used a one-pass clustering algorithm to generate the clusters and finds the anomalies using the outlier factor. The authors claim that the approach can be successfully applied to large applications. The local search heuristics algorithm (LSA) is the basis of the outlier detection technique described in [14]. It is an optimization problem for categorical data and has higher computational complexity. In order to address the shortcomings of LSA outlier detection, a quick greedy approach is proposed in [15], which is based on the greedy search technique. The authors in [16] have focused on cluster-based anomaly detection to identify the anomalies in sensor data by grouping them into clusters. The anomalies are not a part of any of the recognized clusters. To detect anomalies in multivariate time series data, the authors in [17] proposed a technique based on fuzzy clustering and particle swarm optimization. The method for outlier detection suggested in [18] can be used with mixed data. The approach focused on the relationships between attribute values. It initially chose the sets of attributes that may be related and calculate the degree of abnormality, and used it to identify a group of outliers. A technique for effectively detecting human trajectory irregularity inside an indoor environment is proposed in [19]. It used the longest common sub-sequence and density-based DBSCAN techniques. In [20], a machine-learning strategy for anomaly detection is described using K-means clustering and sequential minimum optimization (SMO). It was found that the detection rate is increased, and the number of false-positive alarms is decreased when machine learning techniques are used. For timely and precise awareness of the node running state in publish/subscribe distributed systems, the fuzzy c means clustering-based anomaly node detection technique is proposed in [21], which effectively manages the spread of mistakes. The authors in [22] suggested two algorithms for categorical data, ODT and FastODT, to solve problems like poor detection rate and high computational complexity. The first one performed well for small datasets, whereas the second performed well for both large and small datasets with a low level of computational complexity. The proposed anomaly detection approach in [23] is intended for set-valued data instead of only categorical or numerical data. Here, the outlier factor for a set-valued information system was established based on granular computing (GrC) and rough set theory (RST).

3. Methodologies

3.1. Principal Component Analysis (PCA)

The Principal Component Analysis was first proposed in [24]. It is a particular type of feature extraction approach called "dimensionality reduction," which tries to reduce the number of input characteristics while preserving the majority of the initial information [25]. PCA is a common method for analyzing huge datasets with multiple features using dimensionality reduction. This is done by linearly translating the data into a fresh set of coordinates, where the variance in the data can potentially be expressed with a smaller number of dimensions than the original data. Five steps can be used to decompose principal component analysis [25].

Step 1: Standardize the continuous initial variable ranges to ensure that each of them participates equally in the analysis. To achieve this mathematically, subtract mean of the attribute from the value (X) of every variable and divide by the standard deviation (σ) as shown in (1). The outcome of this step is that all variables are converted to the same scale.

$$S = \frac{X - X_{mean}}{\sigma}, \tag{1}$$

Step 2: This step finds the covariance matrix. The purpose of this stage is to figure out how each variable's deviation from the mean relates to the other variables. Variables can occasionally be so closely connected that they include redundant data. Therefore, the covariance matrix is computed in order to find these relationships. The covariance matrix of 2-D data is of the form as shown in (2), and the covariance is computed using (3). Covariance value can be positive (as x increases y also increases), negative (as x increases y decreases) or zero.

$$\begin{bmatrix} covariance(x,x) & \cdots & covariance(x,y) \\ \vdots & \ddots & \vdots \\ covariance(y,x) & \cdots & covariance(y,y) \end{bmatrix}, \tag{2}$$

$$covariance(x,y) = \frac{\sum(x - x_{mean})(y - y_{mean})}{Number\ of\ data\ objects}, \tag{3}$$

Step 3: This step finds the principal components of the data given. It is done by finding the Eigenvector and Eigenvalue of the covariance matrix found in step 2 and sorting them according to the Eigenvalues in descending order. The idea here is to represent the maximum possible attributes of every data vector in the first principal component, the maximum remaining attributes in the second principal component, and so on. Thus, some principal components initially represent the maximum possible attributes without losing much information. For any square matrix X and a vector v (non-zero), if the following equation (4) holds, then $\propto$ is the Eigenvalue, and v is the Eigenvector of X.

$$Xv = \propto v, \tag{4}$$

This can also be written as:

$$\begin{cases} Xv - \propto v = 0 \\ \quad or \\ (X - \propto I)v = 0 \end{cases}, \tag{5}$$

Where, I is the identity matrix. The above equation holds true if $(X - \propto I)$ is non-invertible. That means,

$$|X - \propto I| = 0\,, \tag{6}$$

Step 4: This step chooses the number of principal components to be considered and selects these components as feature vectors.
Step 5: This is the final step that reorients the original dataset from the original axes to the new axes, which is done by multiplying the transposes of the original dataset and the feature vector as shown in (7).

$$Dataset_{Final} = FeatureVector^T \times Dataset^T_{Standardized}\,, \tag{7}$$

3.2. Fuzzy C-Mean (FCM) Clustering Algorithm

The fuzzy c-means clustering (FCM) technique [26] is one of the most popular soft clustering algorithms. Multidimensional data can be clustered using FCM clustering principle, which rate the point's membership in each cluster from 0 to 100 percent. When compared to conventional hard clustering, this can be extremely powerful.

Let's consider a dataset $D = \{v_1, v_2, \ldots\ldots, v_n\}$ with n data vectors. The dataset D is classified into c number of clusters with centers $C = \{C_1, C_2, \ldots\ldots, C_c\}$. Based on the Euclidean distance between the data point and cluster center, the FCM clustering algorithm determines each data point's membership score $s_{ij} \in [0,1]$. The closer data points have better membership scores. The partition matrix $M = [s_{ij}]_{nxc}$ represents the membership score of the data vector v_i with center C_j. The objective function to be minimized by FCM clustering [26] is as follows.

$$J_{FCM}(D, C, M) = \sum_{j=1}^{c} \sum_{i=1}^{n} s_{ij}^m d_{ij}^2(v_i, C_j), \tag{8}$$

where, m is the fuzziness parameter in [1 to ∞] and d_{ij} is the Euclidean distance between the i^{th} data point and the j^{th} cluster center. Here, the following conditions hold true always.

$$\begin{cases} \sum_{i=1}^{c} s_{ij} = 1\,, j = 1,2, \ldots . n \\ 0 < \sum_{j=1}^{n} s_{ij} < n, i = 1,2, \ldots . c \\ 0 \le s_{ij} \le 1, i = 1,2, \ldots . c \; and \; j = 1,2, \ldots n \end{cases}, \tag{9}$$

ALGORITHM 1: Fuzzy c-means clustering

Input: The dataset to be clustered *D* and number of clusters *n*.
Output: The clustering result is *n* number of clusters.

1. Initialize ***n*** number of clusters with ***n*** random centroids.
2. Find the membership score for each data element using (10).
3. Repeat steps 4 to 6 until membership values are greater than a threshold value.
4. Compute the centroids using (11).
5. Compute the Euclidean distance of each data element with the centroid.
6. Update the membership score for each data element using (10).
7. Print the clusters and their centroids.

The membership score is calculated using (10). The cluster centers are updated using (11) and the partition matrix is re-computed in every iteration until membership score is above a threshold value.

$$s_{ij} = \left[\sum_{k=1}^{c} \left(\frac{d(v_i, C_j)}{d(v_i, C_k)}\right)^{2/(m-1)}\right]^{-1}, \tag{10}$$

$$C_j = \frac{\sum_{i=1}^{n} s_{ij}{}^m v_i}{\sum_{i=1}^{n} s_{ij}{}^m}\,, \tag{11}$$

The FCM clustering algorithm performs task as per the steps in ALGORITHM 1.

4. Modified Opposition Learning based Fuzzy C-Means clustering (OFCM) algorithm

A modified version of Fuzzy c-means clustering algorithm is used for anomaly detection with an Opposition-based learning technique. As discussed in [27, 28], the opposite of a number β in one-dimensional space that falls in range $[p, q]$ can be computed using (12).

$$\beta' = p + q - \beta \quad , \tag{12}$$

Accordingly, in the n-dimensional plane, the opposite of an element β $(\beta_1, \beta_2, \ldots \beta_n)$ is another data element β' computed using (13). Here, β_i is a real number falling in the range $[p_i, q_i]$.

$$\beta_i' = p_i + q_i - \beta_i \;\; , \quad (i \ge 1 \; and \; i \le n) \tag{13}$$

The algorithm works in two phases: the initialization phase and the fuzzy c-mean clustering phase. In the initialization phase, the proposed algorithm starts with a set of ***n*** random centroids. It creates another set of opposite centroids using (13). Next to it all the data elements are allocated to both sets separately to create clusters. In the next step, the 'Sum of Intra Cluster Distances' (SICD) values are computed. The set of centroids with better SICD is considered for the next step and ignores the other set. Further, the opposite centroids of the selected centroids are created, and repeated the process for a predefined number of times. This predefined number is considered as the number of clusters ***n***. The initial centroids generated using Opposition Learning play an important role in faster convergence. In the next phase, the Fuzzy c-means clustering approach is adopted to find the final clusters with their centroids. The steps are presented in ALGORITHM 2.

ALGORITHM 2: Optimized Fuzzy C-Means (OFCM) clustering using Opposition Learning

Input: The dataset to be clustered *D* and number of clusters *n*.
Output: The clustering result is *n* number of clusters.

1. Initialize ***n*** random centroids and Find their opposite centroids using (13).
2. Allocate data elements to both sets of centroids separately.
3. Use Opposition-based Learning in the repeat to find the

required initial clusters and centroids.

4. Find the membership score for each data element using (10).
5. Repeat steps 6 to 8 until membership values are greater than a threshold value.
6. Compute the centroids using (11).
7. Compute the Euclidean distance of each data element with the centroid.
8. Update the membership score for each data element using (10).
9. Print the clusters and their centroids.

5. Anomaly Detection Method using OFCM

For simplification of the further analysis of the anomalies, the dataset's dimensions are reduced to two principal components (PC) using the principal component analysis (PCA) technique, as discussed in section 3.3. The two PCs for each data vector interpret the maximum possible attributes that define a data vector in the original dataset. The first step of the proposed anomaly detection method applies the PCA to get the PCs of each data vector. The rest part of the method works on these PCs. The benefits of this dimensionality reduction are interpretation and visualization of data become easier, and a smaller number of attributes simplifies the vector operations, vector matching, and other data analysis.

The proposed method for anomaly detection has three phases. In the first phase, the OFCM algorithm is used to get the required number of clusters. In the second phase, the rare classes are identified using the *outlier factor*. According to its definition in [6], the *outlier factor* of a cluster C_i is the weighted sum of its distances to other clusters C_j and is computed as follows:

$$outlier\_factor(C_i) = \sum_{i \neq j} size(C_j) \times d(C_i, C_j) \ , \qquad (14)$$

where, $size(C_j)$ is the number of data objects in the cluster C_j and d is the Euclidean distance. The *outlier factors* are sorted, and y clusters with larger *outlier factor*s are selected based on a predefined threshold value (t) as rare classes so that the following condition is satisfied.

$$\frac{\sum_{i=1}^{y} size(C_i)}{size(D)} < threshold\ (t) \ , \qquad (15)$$

The value of *the outlier factor* is large, the chance of being considered as a rare class increases. Thus, at the end of the second phase, we have a set of normal classes/clusters and another set of rare classes/clusters. However, at this point it is too early to declare all the data points belonging to these rare classes as anomalies. As a result it will lead the technique to be completely dependent on the accuracy of the clustering algorithm. Also, there is a chance that the real outliers/anomalies are present as part of normal classes. To overcome this issue in the third phase, the *top-k* outliers are selected across the clusters. This selection is made on the basis of the *outlier factor* of each data point. The *outlier factor* formula is modified for the data objects and is as follows:

$$outlier\_factor(O_m^i) = \sum_m \sum_{i \neq j} size(C_j) \times d(O_m^i, C_j) \ , \qquad (16)$$

where, $O_m{}^i$ is the m^{th} data object, and i represents its cluster association (that is, it belongs to the cluster C_i). Other terms are similar to [5]. The *outlier factor* for each data object is computed using (16), and the topmost k (a predefined constant) data objects are declared as anomalies. The proposed technique for anomaly detection is shown in ALGORITHM 3, and Fig.1 shows the high-level flow diagram of the procedure along with the analysis of detection accuracy.

It is considered that the *top ratio* and *outlier coverage ratio* are significant to analyze the anomaly detection accuracy. The *top ratio* is the proportion of data instances designated as anomalies (value of k in *top k* outliers) to the overall instances in the dataset. The ratio of the number of anomalies detected (or k in *top k* outliers) to the number of data instances present in the overall rare classes is known as the *outlier coverage ratio*. In other words, *the outlier coverage ratio* is the percentage of the true outlier objects among the *top k* outliers found.

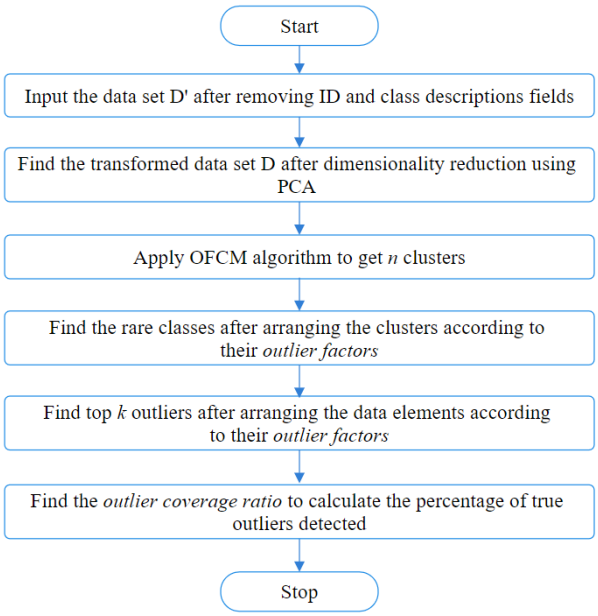

Fig.1. High-level flow diagram of OFCM-based anomaly detection

ALGORITHM 3. OFCM-based anomaly detection

Input: The predefined constant k, dataset D after dimensionality reduction using PCA and number of clusters n.

Output: Top k outliers.

Phase 1: Apply the proposed OFCM clustering algorithm to get the dataset clustered into n clusters (C_1, C_2, C_3, ……, C_n). Here, C_i represents the cluster center of the i^{th} cluster.

Phase 2: Finds the rare classes (clusters) that contain anomalies. Here, the following steps are executed.

- Step 1 Compute the outlier factors of all the clusters separately using (14).
- Step 2 Sort the clusters according to the computed outlier factors in decreasing order.
- Step 3 Find y clusters with larger *outlier factors* and label them as rare classes.

Phase 3: Find *top-k* outliers across the clusters. The following steps are executed here.

- Step 1 The *outlier factors* of all the data objects are

computed separately using (16).

Step 2 Sort the data objects according to the computed *outlier factors* in decreasing order.

Step 3 Find *top k* data objects that have larger *outlier factor*s and declare them as anomalies.

6. Simulation Results

The proposed algorithm is evaluated using a thorough performance analysis. The results of the experiments are described in this section. In order to assess the correctness of the technique, the algorithm is used to process the data set and find the outliers with the help of the built-in labels. In this study, two widely used standard data sets for anomaly detection tasks from the UCI machine learning repository are considered. These are the Lymphography dataset and the Wisconsin Breast Cancer dataset. The proposed method is verified and compared with earlier works.

Both datasets are labeled datasets with class descriptions. For experimentation, the class description and sample ID information fields are removed to make it unlabeled. PCA is applied to get the principal component vectors as the required dataset. The clustering approach is applied further to divide the data samples into *n* random clusters as part of the first phase of anomaly detection. Considering the threshold *t* in (14), the rare classes are identified in phase 2, and the top *k* outliers are identified using (16) in phase 3. The *top k* outliers are then analyzed to find how many and what percentage of the actual anomalies (referred to as anomaly detection accuracy) are detected by the method. For this task, the vector correlation is used. All the *top k* outliers are individually correlated with the actual outliers from the original dataset (with class description) to find a match in the transformed dataset *D* that is found after PCA is applied. If a match is found, the count is incremented to indicate the number of anomalies detected.

The anomaly detection accuracy is evaluated in terms of *the outlier coverage ratio*. The *outlier coverage ratio* is found for different *top ratio* values of specific datasets. The simulation results of both lymphography and breast cancer datasets are discussed below. Fig. 2(a) and Fig. 2(b) depict the data distributions of both datasets before clustering.

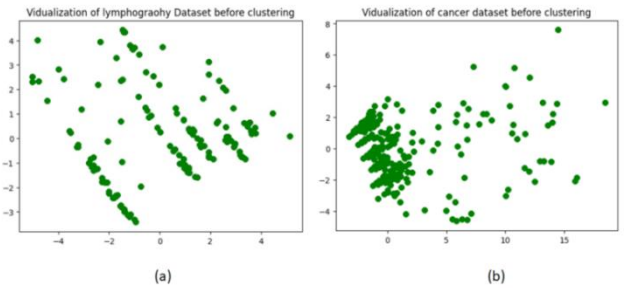

Fig. 2. Data distributions of (a) lymphography dataset and (b) breast cancer dataset

Lymphography Dataset: There are four classifications in the lymphography dataset, with 148 instances and 18 attributes. Two of them have a very small number of records (2 and 4, respectively). In light of the other two large classes, these two minor classes are combined and regarded as anomalies. The details of the dataset are shown in Table 1. First of all, the 'Class' attribute removed to make the dataset unlabeled, and the applied PCA to reduce the dimension to two dimensions, PC1 and PC2. FCM and OFCM algorithms are used for clustering separately to the reduced dataset, and the results are presented in the scatter plot in Fig. 3(a) and 3(b), respectively. Once the rare classes found, the top *k* data elements having unusual behaviors (top *k* outliers) are found. The top *k* outliers when the top ratio is 0.05 and 0.10 are shown in Fig. 4(a) and 4(b). The experimental result in terms of *outlier coverage ratio* is shown in Table 2 for various existing algorithms with varying top ratios. It is observed that most of the anomalies are included in the designated rare classes even when *the top ratio* is set to smaller values. In this experiment, the value of *k* is chosen as 7, 15, 16, 22, and 30. Out of the *top k* outliers when *k=7*, five outliers (83.33%) are true outliers. In the rest of the cases, all six true outliers (100%) are included in the top *k* outliers list.

Table 1. The lymphography dataset (148 instances)

Condition	Class Name/Number	Number of Instances	Percentage of Instances
Non-Rare Classes	2, 3	142	95.94
Rare Classes	1, 4	6	4.06

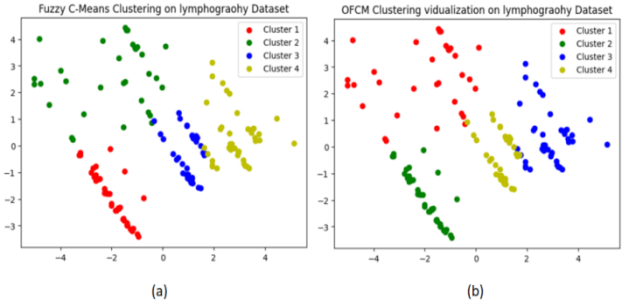

Fig. 3. Clustering result of lymphography dataset

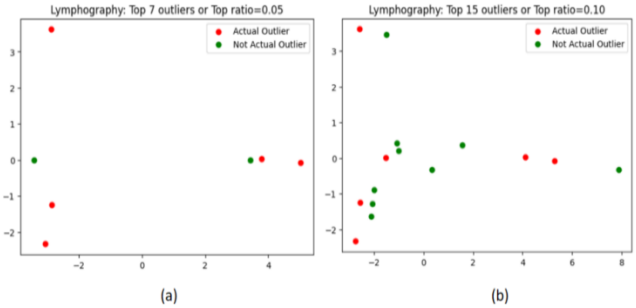

Fig. 4. Outliers of lymphography dataset (a) top ratio=0.05 (b) top ratio=0.10

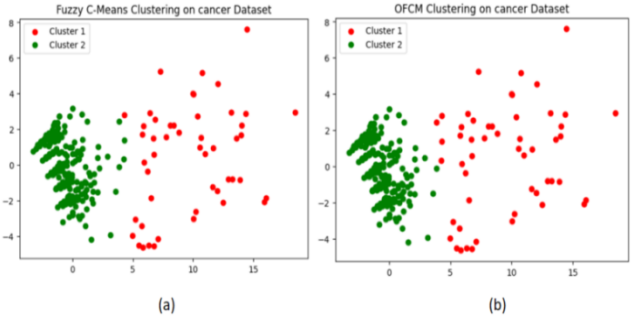

Fig. 5. Clustering result of breast cancer dataset

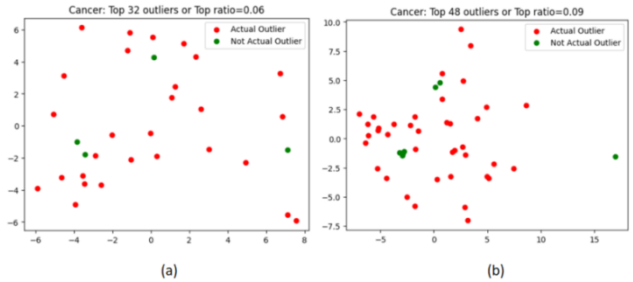

Fig. 6. Outliers visualization of breast cancer dataset (a) top ratio=0.06 (b) top ratio=0.10

Wisconsin Breast Cancer Dataset: There are two classifications (benign and malignant) in the Wisconsin breast cancer dataset with 699 instances and nine attributes. Data points in the dataset's malignant class are regarded as outliers or anomalies, whereas those in the benign class are regarded as inliers. The details of the dataset are shown in Table 3. The majority of the objects from the malignant class have been deleted in order to keep anomaly occurrences rare and fit for the detection method. For deletion the random records from both of the classes are chosen. In the experiment, 444 benign and 39 malignant objects are utilized, making the dataset contain 483 data objects in total. First of all, the 'Sample code number' and 'Class' attributes are removed to make the dataset unlabeled, and the applied PCA to reduce the dimension to two dimensions, PC1 and PC2. FCM and OFCM algorithms are applied for clustering separately, and the result is presented in the scatter plots in Fig. 5(a) and 5(b). Once the rare classes are obtained, the top *k* data elements having unusual behaviours (top *k* outliers) are found. The top *k* outliers when the top ratio is 0.06 and 0.10 are shown in Fig. 6(a) and 6(b). For several existing algorithms with variable *top ratios*, the experimental outcome in terms of *outlier coverage ratio* is provided in Table 4. It is observed that most of the anomalies are included in the top *k* outliers list even when *the top ratio* is is set to smaller values. The value of *k* is considered to 10, 15, 20, 30, 40, 45, 55, 65, and 70. When the the top ratio is 0.10 (*k*=45) or more, 100 % of the outliers are covered in the *top k* outliers. Various algorithms use different numbers of benign and malignant objects, so the number of outliers or *k* in top *k* outliers may differ. To maintain uniformity, *the top ratio* is considered instead of the number of outliers.

The above simulation results justify that the proposed technique has an edge over some existing anomaly detection techniques such as KNN, CBLOF, GA, LSA, and Fast-ODT. Moreover, the proposed technique does not heavily rely on clustering accuracy, which is a common issue in clustering-based approaches (the capability of the clustering algorithm to separate anomalous and normal data objects into separate clusters) and the *outlier factor* of each individual data object considered instead of the whole cluster which strengthens the anomaly detection approach.

Finally, a comparison is made of the anomaly detection techniques when the fuzzy-c-means (FCM) clustering was used in the first phase with the technique when opposition learning and fuzzy-c-means (OFCM) clustering were used in the first phase. It is found that for the higher value of *k*, both methods perform similarly for both datasets. However, for lower values of *k* the OFCM based technique has better accuracy. Table 5 shows this comparison.

Table 2. Outlier coverage ratio for lymphography dataset [9, 14, 15, 24, 25]

Top Ratio	Number of anomalies included (Outlier coverage ratio in %)					
	Proposed	KNN	CBLOF	GA	LSA	Fast-ODT
0.05	5 (83.33)	4 (66.66)	4 (66.66)	-	**6 (100.00)**	-
0.10	**6 (100.00)**	**6 (100.00)**	4 (66.66)	5 (83.33)	**6 (100.00)**	5 (83.33)
0.11	**6 (100.00)**	**6 (100.00)**	4 (66.66)	**6 (100.00)**	**6 (100.00)**	**6 (100.00)**
0.15	**6 (100.00)**	**6 (100.00)**	4 (66.66)	**6 (100.00)**	**6 (100.00)**	**6 (100.00)**
0.20	**6 (100.00)**	**6 (100.00)**	**6 (100.00)**	**6 (100.00)**	**6 (100.00)**	-

Table 3. The Wisconsin breast cancer dataset (699 instances)

Condition	Class Name/Number	Number of Instances (After Removal)	Percentage of Instances (After Removal)
Non-Rare Classes	benign	458 (444)	65.50 (91.92)
Rare Classes	malignant	241 (39)	34.50 (8.08)

Table 4. Outlier coverage ratio for Wisconsin breast cancer dataset [9, 14, 15, 24, 25]

Top Ratio	Number of anomalies included (Outlier coverage ratio in %)					
	Proposed	KNN	CBLOF	GA	LSA	Fast-ODT
0.02	7 (17.95)	8 (20.52)	7 (17.95)	7 (17.95)	8 (20.52)	-
0.03	15 (38.46)	16 (41.00)	14 (35.90)	15 (38.46)	15 (38.46)	-
0.05	22 (56.41)	20 (51.28)	21 (53.85)	22 (56.41)	22 (56.41)	14 (58.33)
0.06	28 (71.79)	27 (69.23)	27 (69.23)	27 (69.23)	29 (74.36)	21 (87.50)
0.08	34 (87.17)	32 (82.05)	32 (82.05)	33 (84.62)	33 (84.62)	**24 (100.00)**
0.10	**39 (100.00)**	37 (94.87)	35 (89.74)	36 (92.31)	38 (97.44)	**24 (100.00)**

0.11	**39 (100.00)**	**39 (100.00)**	38 (97.44)	**39 (100.00)**	**39 (100.00)**	-
0.13	**39 (100.00)**	**39 (100.00)**	**39 (100.00)**	**39 (100.00)**	**39 (100.00)**	-
0.14	**39 (100.00)**	**39 (100.00)**	**39 (100.00)**	**39 (100.00)**	**39 (100.00)**	-

Table 5. Outlier coverage ratio with FCM and OFCM for both the datasets

Dataset	Number of objects *k* (Top Ratio)	Number of anomalies included (Outlier coverage ratio in %)	
		FCM based method	OFCM based method
Lymphography	4 (0.02)	2 (33.33)	2 (33.33)
	6 (0.04)	3 (50.00)	4 (66.66)
	8 (0.05)	4 (66.66)	5 (83.33)
	12 (0.80)	5 (83.33)	5 (83.33)
	15 (0.10)	**6 (100.00)**	**6 (100.00)**
Breast cancer	18 (0.03)	12 (30.76)	15 (38.46)
	25 (0.05)	20 (51.28)	22 (56.41)
	32 (0.06)	27 (60.23)	28 (71.79)
	40 (0.08)	34 (87.17)	34 (87.17)
	48 (0.10)	**39 (100.00)**	**39 (100.00)**

7. Conclusions

In this study, a method for clustering-based anomaly detection that relies on fuzzy-c-means clustering and Opposition-based Learning is proposed. The proposed method divides the dataset into separate classes and looks for rare classes. The *top k* data objects are found based on outlier factors and are regarded as anomalies. The *top ratio* and *outlier coverage ratio* are taken into account to analyze how well the strategy performs. The experimental results show that for the specific applications, the technique is efficient in terms of *outlier coverage ratio* compared to other techniques that are considered in this work. Also, the performance of the technique is compared by using FCM and OFCM in the first phase. The results show that for lower values of *k* OFCM based technique performs better, whereas for higher values of *k* both perform similarly. Moreover, the efficacy of the technique needs to be tested for real life applications that are larger in size.

References

[1] Aggarwal CC, Aggarwal CC. An introduction to outlier analysis. Springer International Publishing; 2017.

[2] Ketepalli G, Tata S, Vaheed S, Srikanth YM. Anomaly Detection in Credit Card Transaction using Deep Learning Techniques. In2022 7th International Conference on Communication and Electronics Systems (ICCES) 2022 Jun 22 (pp. 1207-1214). IEEE.

[3] Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. ACM computing surveys (CSUR). 2009 Jul 30;41(3):1-58.

[4] Barnett V, Lewis T. Outliers in statistical data. New York: Wiley; 1994 Apr.

[5] Yang P, Huang B. KNN based outlier detection algorithm in large dataset. In 2008 International Workshop on Education Technology and Training & 2008 International Workshop on Geoscience and Remote Sensing 2008 Dec 21 (Vol. 1, pp. 611-613). IEEE.

[6] Jiang SY, An QB. Clustering-based outlier detection method. In 2008 Fifth international conference on fuzzy systems and knowledge discovery 2008 Oct 18 (Vol. 2, pp. 429-433). IEEE.

[7] Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. Inkdd 1996 Aug 2 (Vol. 96, No. 34, pp. 226-231).

[8] Smith R, Bivens A, Embrechts M, Palagiri C, Szymanski B. Clustering approaches for anomaly-based intrusion detection. Proceedings of intelligent engineering systems through artificial neural networks. 2002 Oct;9.

[9] He Z, Xu X, Deng S. Discovering cluster-based local outliers. Pattern recognition letters. 2003 Jun 1;24(9-10):1641-50.

[10] Elmogy A, Rizk H, Sarhan AM. Ofcod: On-the-fly clustering-based outlier detection framework. Data. 2020 Dec 30;6(1):1.

[11] Degirmenci A, Karal O. Efficient density and cluster-based incremental outlier detection in data streams. Information Sciences. 2022 Aug 1;607:901-20.

[12] Mazarbhuiya FA, Shenify M. A Mixed Clustering Approach for Real-Time Anomaly Detection. Applied Sciences. 2023 Mar 24;13(7):4151.

[13] Jiang SY, An QB. Clustering-based outlier detection method. In 2008 Fifth international conference on fuzzy systems and knowledge discovery 2008 Oct 18 (Vol. 2, pp. 429-433). IEEE.

[14] He Z, Deng S, Xu X. An optimization model for outlier detection in categorical data. In International conference on intelligent computing 2005 Aug 23 (pp. 400-409). Berlin, Heidelberg: Springer Berlin Heidelberg.

[15] He Z, Deng S, Xu X, Huang JZ. A fast greedy algorithm for outlier mining. In Advances in Knowledge Discovery and Data Mining: 10th Pacific-Asia Conference, PAKDD 2006, Singapore, April 9-12, 2006. Proceedings 10 2006 (pp. 567-576). Springer Berlin Heidelberg.

[16] Vanem E, Brandsæter A. Cluster-based anomaly detection in condition monitoring of a marine engine system. In2018

Prognostics and System Health Management Conference (PHM-Chongqing) 2018 Oct 26 (pp. 20-31). IEEE.
[17] Li J, Izakian H, Pedrycz W, Jamal I. Clustering-based anomaly detection in multivariate time series data. Applied Soft Computing. 2021 Mar 1;100:106919.
[18] Kim YG, Lee KM. Association-based outlier detection for mixed data. Indian Journal of Science and Technology. 2015 Oct;8(25):1-6.
[19] Lan DT, Yoon S. Trajectory Clustering-Based Anomaly Detection in Indoor Human Movement. Sensors. 2023 Mar 21;23(6):3318.
[20] Gadal S, Mokhtar R, Abdelhaq M, Alsaqour R, Ali ES, Saeed R. Machine Learning-Based Anomaly Detection Using K-Mean Array and Sequential Minimal Optimization. Electronics. 2022 Jul 10;11(14):2158.
[21] Wang D, Shen Z, Wu W. A fuzzy clustering based anomaly node detection method for publish/subscribe distributed systems. InJournal of Physics: Conference Series 2021 Feb 1 (Vol. 1813, No. 1, p. 012046). IOP Publishing.
[22] Du H, Ye Q, Sun Z, Liu C, Xu W. FAST-ODT: A lightweight outlier detection scheme for categorical data sets. IEEE Transactions on Network Science and Engineering. 2020 Sep 9;8(1):13-24.
[23] Lin H, Li Z. Outlier detection for set-valued data based on rough set theory and granular computing. International Journal of General Systems. 2023 May 19;52(4):385-413.
[24] Pearson K. LIII. On lines and planes of closest fit to systems of points in space. The London, Edinburgh, and Dublin philosophical magazine and journal of science. 1901 Nov 1;2(11):559-72.
[25] Smith LI. A tutorial on principal components analysis. 2002.
[26] Bezdek JC, Ehrlich R, Full W. FCM: The fuzzy c-means clustering algorithm. Computers & geosciences. 1984 Jan 1;10(2-3):191-203.
[27] Sahoo SK, Pattanaik P, Mohanty MN, Mishra DK. Opposition Learning Based Improved Bee Colony Optimization (OLIBCO) Algorithm for Data Clustering. International Journal of Advanced Computer Science and Applications. 2023;14(4).
[28] Sahoo SK, Pattanaik P, Mohanty MN. Modified bee colony optimization with opposition learning algorithm on use of medical data clustering. Intelligent Decision Technologies.(Preprint):1-6.