International Journal of Engineering

Journal Homepage: www.ije.ir

Outlier Detection in Test Samples using Standard Deviation and Unsupervised Training Set Selection

N. Mohseni[a], H. Nematzadeh[*b], E. Akbarib[b], H. Motameni[b]

[a] *Department of Computer Engineering, Babol Branch, Islamic Azad University, Babol, Iran*
[b] *Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran*

PAPER INFO

Paper history:
Received 09 June 2022
Received in revised form 27 September 2022
Accepted 28 September 2022

Keywords:
Outlier Detection
Training Set Selection
K-means
K-Nearest Neighbor
Standard Deviation

ABSTRACT

Outlier detection is a technique to identify and remove significantly different data from the more correct and consistent data in a data set. Outlier data can have negative impact on classification and clustering performance; that should be identified and removed to improve the classification efficiency. Regardless of whether a classifying technique classifies an outlier correctly, the very notion of identifying a data as outlier is of great significance. In this paper, a new approach is proposed for outlier data detection within a test data set along with unsupervised training set selection. The selected training set is used for two-step classification. After unsupervised clustering the training set, the closest cluster to a test sample is selected using the Euclidean distance measure. Then, the outlier in the test sample is identified with the concepts of standard deviation and mean value. The results showed by evaluating the distance of each sample of the test set with the new selected data set. the accuracy of the classifiers is enhanced after detection and elimination of outlier data.

***doi**: 10.5829/ije.2023.36.01a.14*

1. INTRODUCTION

The availability of large training data sets makes training set selection a significant task [1]. Extraction of a reduced and most pertinent training set is a data mining challenge for researchers. After training data selection, a learning session is performed by classifiers. Generally, classification should lead to a more accurate prediction and occasionally a reduced execution time [2-5]. TSS is also a common tool in image processing and image recognition where large data sets may be available for classification [6, 7]. Outlier data with unacceptable distance with other consistently dispersed data may at times exist. The efficient identification and eliminations of these data types will enhance the classification outcome [8]. Statistical methods such as linear regression model [9] and Principle Component Analysis (PCA) [10] have traditionally been used for outlier detection. As no single outlier detection approach can be regarded as the most efficient, several other methods have been proposed in the literature [5, 11-18]. In this paper, it is attempted to detect outlier data within the test data set together with training data set selection task. The common trend in the literature is to detect outliers from within training data sets. However, outlier data may also exist among test data. Therefore, this research is dedicated to outlier detection in test data sets. As reviewed, no previous research has focused on outlier detection in the test data while embarking on training set selection. Contribution of this paper are as follows:

- Clustering the training set with the unsupervised method.
- Dynamic training set selection using KNN for each test sample.
- Outlier detection using standard deviation and mean average.

The related works with more details on training set selection and outlier detection techniques are reviewed next.

2. RELATED WORK

Regarding training set selection, a pareto-based method is proposed by Acampora et al. [19] where a multi-

*Corresponding Author Institutional Email: hossein.nematzadeh@iau.ac.ir (H. Nematzadeh)

Please cite this article as: N. Mohseni, H. Nematzadeh, E. Akbarib, H. Motameni, Outlier Detection in Test Samples using Standard Deviation and Unsupervised Training Set Selection, ***International Journal of Engineering, Transactions A: Basics***, Vol. 36, No. 01, (2023), 119-129

objective function is optimized which aims to maximize accuracy and reduction rate. Due to scalability problems associated with support vector machine (SVM), this data mining technique becomes extremely complex with high execution time and memory usage. The mentioned approach reduces the training set in the preprocessing step through the selection of the most pertinent samples. Thus, regardless of the sample size the execution time and storage requirement remain the same. This method was demonstrated to enhance accuracy and reduction rate as applied to Shell Extraction algorithm [2].

Verbiest et al. [1] investigated through wrapper techniques which was applied to TSS, further improvements were made to SVM accuracy. The training set was converted into subsets which were evaluated by the mentioned SVM-based wrapper algorithms. According to the literature, five distinct TSS methods have been presented. Among these methods, the generational genetic algorithm (GGA) has delivered the best performance. As the paper concluded, using evolutionary techniques positively affects SVM accuracy though at the cost of longer execution times resulted from frequent iterations. Also, it is demonstrated that wrapper TSS exhibits better performance on experimental data sets as compared to filter TSS.

Mohammed et al. [4] have applied swarm intelligence techniques to carry out TSS. The procedure includes three steps where first, data sampling is performed to reduce input data scale which is then used to train classifiers with bagging and distance-based feature sampling algorithms. Finally, through a diverse set of meta-heuristic methods, such as grey-wolf optimization (GWO), Whale optimization algorithm (WOA) and moth-flame optimization (MFO), effective weight assignments are made to the classifiers. This method, which employs Matthews Correlatoin Coefficicent (MCC), perfectly suits binary as well as multi-class data sets. Also, WOA delivered the poorest performance in prediction accuracy for large data sets. Despite relatively short weight assignment time through parameter modifications, the method suffers from relatively long overall execution time. Thus, the method's application is limited to offline training tasks. Hence, it is concluded that meta-heuristic methods will lead to time-consuming solutions. In the following, a review of outlier detection techniques proposed so far is presented.

Mapping functions of differential geometry were used by Lejeune et al. [13] in a shape-based detection of outliers within multivariate functional data. This method demonstrates acceptable efficiency as compared to functional-depth-based methods as outliers can be difficult to detect from multivariate functional data due to complicated relations between parameters. The mentioned research, however, performs successfully through extraction of synthetic data sets using interpretable functional curve shape characteristics.

An ensemble outlier detection method was proposed by Wang and Mao [15]; they applied to an adaptive K-Nearest Neighbor. Having higher capabilities with respect to the traditional K-Nearest Neighbor, the adaptive method uses support vector data and explores those areas with constant class probabilities with respect to the test pattern. Through the use of posterior probabilities of classifiers, this method performs in a more efficient way than the traditional methods like majority voting and selects a more suitable data set for test patterns.

Another outlier detection approach was introduced by Tang and He [14] based on relative density-based scores as an index for outlying degree. In order to estimate distribution density at the proximity of an object, the Kernel Density Estimation (KDE) was employed. Also, in addition to the conventional K-Nearest Neighbor, reverse and shared nearest neighbors were examined for efficient detection. The proposed methods were tested on diverse large and small real-life data sets including medical data. The efficiency of these methods was verified by techniques such as Receiver Operator Characteristics (ROC) and Area Under Curve (AUC).

Outlier detection through clustering approach was proposed by Christy ey al. [12] with two different algorithms, namely distance-based and cluster-based, which utilized outlier scoring to detect and remove outliers of a healthcare data set. Clustering was based on similarity in only the critical properties of the data. As the results of implementation on three data sets from R package (Esoph, Diabetes, and KosteckiDillon) revealed, the cluster-based approach [20] is more competent than the other detection method. Furthermore, the results indicate that the F-score and likelihood values do not change with random data object removals.

A new method called Information Entropy-Pruning Multi-dimensional Outlier Detection (IPMOD) was introduced by Yang et al. [11] in which a combination of information entropy and a new index weight measurement were applied to multi-dimensional data helping to specify the effect of various attributes on data prediction. Consequently, a sliding window approach and subsequent distance measurements were used to determine if the data was outlier. They also lowered the proposed method complexity through pruning techniques. The provided results were indicative of the efficiency of the algorithm both in terms of accuracy and fastness of the outlier detection method as implemented on different real-life data sets such as medical databases.

A an unsupervised ensemble approach, namely Boosting based autoencoder ensemble (BAE) was proposed by Hamed et al. [16] which, through creating a chain of autoencoders, guarantees robust and enhanced results. Autoencoder elements are constantly trained through weighted data sampling which also attempts to elimintae outlier data while training or diversifying the

ensemble. The mentioned work includes several tests that unanimously demonstrate superior performance of the proposed approach against conventional algorithms.

Further, a mean-shift outlier detection scheme was introduced by Yang et al. [17] which, through data modification, removes possible bias caused by the existence of outlier data. Every object in this method is substituted by its k-nearest neighbors. This guarantees the removal of outlier effect and the clustering is performed in a more efficient way. The outliers can be detected through measuring the shifted distance. This approach is demonstrated to work well for any given number of outlier data within real-life or synthetic data sets. Furthermore, Beulah and Vamsi Krishna [21] presented a density-based outlier detection for unsupervised training. In this approach, adaptive Natural Value (NV) determination is achieved through Natural Neighbor (NaN) concept while location density for an object is predicted through weighted kernel density estimation (WKDE) technique. Also, k nearest neighbors (KNN) and reverse nearest neighbors (RNN) are utilized which contribute to the modeling flexibility for different data patterns. For the sake of smoothness, Gaussian kernel function is employed and the notion of adaptive kernel width is applied to make clear distinction between correct and outlier data. Through comprehensive test cases with both real and synthetic data, the proficiency of the method in outlier detection is demonstrated.

Biglari et al. [22] followed a semi-novel approach using " Ensemble of Unsupervised Incremental Learning" method for building models that by analyzing data, can find the behavior or state change in equipment or machinery over time. This paper focuses on the behavior of data mining of machines in process/manufacturing industries. Generally, such data are continuous numerical. In addition, time series data are captured by various industrial sensors.The proposed algorithm must run in two phases , offline and online with hierarchical clustering.

A novel hybrid feature selection technique was proposed by Fränti and Sieranoja [23], which can reduce the number of features drastically with an acceptable loss of prediction accuracy. The proposed approach operates in multiple stages, starting by removing irrelevant features with a low discrimination power, and then eliminating the ones with low variation range. Afterward, among each set of features with high cross-correlation, a single feature that is strongly correlated with the output is kept. Finally, a Genetic Algorithm with a customized cost function is provided to select a small subset of the remainder of features.

3. BACKGROUND

Most of the reviewed research works have focused on K-means clustering and K-Nearest Neighbors (KNN) algorithms and the results have been analyzed through SVM and Random Forest methods. In this section, the main pros and cons of K-means and KNN algorithms are studied.

3. 1. K-means Clustering Algorithm The main variations of clustering technique are the partitional and hierarchical clustering algorithms. Meanwhile, each of these methods can be divided into distance-based and density-based categories. Distance-based methods are usually based on Euclidean or city block distance analyses. A data space is considered dense if it forms a dense data region. The partitional clustering is first initiated by a human as opposed to the hierarchical clustering technique [24, 25]. K-means clustering is categorized as a distance-based technique. Through this approach, a set of X samples Equation (1) is classified into k groups with $k \leq n$ as determined by the user [26, 27].

Then the algorithm selects k samples from the set X in a random fashion and puts them as the centers of a vector M^0, as modeled by Equation (2). If a sample x_l has the minimum distance from m_i^r as in Equation (3), it is said to belong to cluster C_i^r. The use of a distance measure $(dist)$ is arbitrary in K-Means approach. However, in Equation (3) the Euclidean distance is employed as it has performed well in the previous works.

$$X = (x_1 . x_2 . \dots . x_n) \tag{1}$$

$$M^r = m_1^r, m_2^r, \dots, m_k^r \tag{2}$$

$$x_l \in \ C_i^r : dist(x_l, m_i^r) \leq \left\{dist\left(x_l, m_j^r\right)\right\}_{j=1, j\neq i}^{k} \quad 1 \leq i \leq \ k, \quad x_l \in X \tag{3}$$

The following relation is also used to calculate Euclidean distance between two d-dimensional samples of $x_i = (f_1^i, f_2^i, \dots, f_d^i)$ and $x_j = (f_1^j, f_2^j, \dots, f_d^j)$ as [19]:

$$ED = \left(\sum_{l=1}^{d} (f_l^i - f_l^j)^2\right)^{1/2} \tag{4}$$

Also, new centroids are determined through updating M^r at each iteration as in Equation (5). If the members of C_i^r and C_i^{r+1} stay unchanged in successive iterations, then it can be said that K-means procedure can stop.

$$m_i^{r+1} = \frac{1}{|C_i^r|} \sum_{x_j \in C_i^r} x_j \quad 1 \leq i \leq \ k \tag{5}$$

3. 2. K-Nearest Neighbors Regression The K-nearest neighbor (KNN) as a regression classification technique is used to specify nearest neighbor for a given sample based on some of the most common sample measurements. For a data set D with k clusters and similar member labels (p), KNN can be obtained for test sample TS_h within a test set (TS). For this purpose, the distance of the sample test TS_h, denoted as d_{ji} between

cluster members x_{ji} is calculated using Equation (6). Then, d_{ji} is sorted in ascending order to form ds_{ji} as Equation (7) and the average of K members from ds_{ji} is calculated as ave_i to yield the best K as modeled in Equation (8). The outlier detection within TS_h is based on the cluster with the least ave value, as determined by Equation (9).

$$d_{ji} = dist(TS_h, x_{ji}) \quad h = 1,2,\dots,|TS| \quad j = 1,2,\dots,|C_i| \quad i = 1.2.\dots.p \tag{6}$$

$$ds_{ji} = sort\,(d_{ji}) \tag{7}$$

$$ave_i = \frac{1}{K}\sum_{j=1}^{K} ds_{ji} \tag{8}$$

$$p = \min(ave) \tag{9}$$

Despite being regarded as a classification technique, KNN was used here as a means for distance determination. The efficiency of the proposed methods can be assessed through SVM [1-3, 28-30] and Random Forest (RF) [31] as applied to two-labeled and multi-labeled data sets, respectively. In this paper, SVM is based on linear kernel function considering box constraint of 2. Also, RF technique comprises several single decision trees bringing more accurate results compared to a single decision tree.

4. PROPOSED METHOD

In this section the Euclidean distance criterion applied in our proposed method is described.
for two vectors A and B, the distance measure is as Equation (10):

$$\boldsymbol{dist}\,(\boldsymbol{A}.\boldsymbol{B}) = \sqrt{\sum_{i=1}^{n}|\boldsymbol{A_i} - \boldsymbol{B_i}|^2}\ .\boldsymbol{n} = \boldsymbol{vector\ size} \tag{10}$$

The overall scheme of the proposed method is given below. As shown, the whole data set is divided into two groups of training and test sets. First, using k-means method, the training data are classified into k clusters where k is the square of the training set length. Then, for each sample within test set $\boldsymbol{ts_i}$, the closest cluster is determined using Euclidean distance and KNN regression algorithm. The data of the selected cluster are then selected and stored as the new sets of training set selection. Definitely, the more similar the determined cluster to the selected test sample $\boldsymbol{ts_i}$, the more similar the selected test sample is to the training set samples of TSS. If the determined cluster only contains one sample, which is a rare case that may happen for synthetic data, the next most similar cluster is merged with the initial one and a new training set is developed for a more accurate and more rational calculation. Then, the standard deviation $\boldsymbol{\sigma}$ and mean value $\boldsymbol{\mu}$ of the selected training set is calculated. To identify any outlier data, the standard deviation and mean value of the new training set is calculated. For a mean value of the new training set $\boldsymbol{\mu}$ with standard deviation $\boldsymbol{\sigma}$, a distance larger than $\boldsymbol{\mu + 2\,\sigma}$ or smaller than $\boldsymbol{\mu - 2\,\sigma}$ between the new training set and the test sample will indicate that the test sample cab be an outlier data, as shown in Figure 1 [32]. In this case, test sample outlier is detected and the classification operation is not performed. Otherwise, the test sample is regular and the finalized label $\mathbf{L_i}$ is identified using classifiers leading to improved classification outcome.

The overall scheme of Figure 2 can be formulated as follows:

Given a data set D as Table 1, each sample can be denoted by Equation (11). Each characteristic vector $\mathbf{F_j}$ is represented by Equation (12).

$$\bar{\boldsymbol{x}}_{\boldsymbol{i}} = (\boldsymbol{f_{i1}}.\boldsymbol{f_{i2}}.\dots.\boldsymbol{f_{in}}\ )\ \ \boldsymbol{i = 1.2.\dots.n} \tag{11}$$

$$\bar{\boldsymbol{F}}_{\boldsymbol{j}} = \left(\boldsymbol{f_{1j}}.\boldsymbol{f_{2j}}.\dots.\ \boldsymbol{f_{nj}}\right)^{\boldsymbol{t}} \tag{12}$$

if $\mathbf{L}=\{\boldsymbol{L_1}, \boldsymbol{L_2},\dots, \boldsymbol{L_k}\}$ is a set of labels, the samples belonging to each label are represented as Equation (13).

$$y^{lp} = \left\{x_1^{lp}.x_2^{lp}.\dots.x_n^{lp}\right\} = \left\{x_j^{lp}\right\}_{j=1}^{n} \tag{13}$$

$$\boldsymbol{where}\ \cup_{\boldsymbol{p=1}}^{\boldsymbol{k}}\ \ \boldsymbol{y_0^{lp}} = \sum_{\boldsymbol{p=1}}^{\boldsymbol{k}} \boldsymbol{n_p} = \boldsymbol{n}\ .\boldsymbol{p} = \boldsymbol{1.2.\dots.k}$$

For a total sample number of X within data set D, it is clustered into k clusters regardless of its labels using k-means classifier where k is the square of the total number of samples n. Each cluster is titled $\boldsymbol{C_r}$:

$$C_r = Kmeans\,(X.k) \qquad r = 1.2.\dots.k \tag{14}$$

where $\boldsymbol{C} = \{\boldsymbol{c_1}, \boldsymbol{c_2}, \dots, \boldsymbol{c_k}\}$

Then, the cluster $\boldsymbol{C_r}$ with the least Euclidean distance from $\boldsymbol{TS_i}$ is slected as the new training set as Equation (15):

$$TSS = min\{C_r\}_{r=1}^{k} \tag{15}$$

In the proposed unsupervised algorithm, the samples are clustered irrespective of their labels. Then, the mean distance of the cluster members $\boldsymbol{C_r}$ from a test sample in $\boldsymbol{TS_i}$ are calculated using algorithm 1.

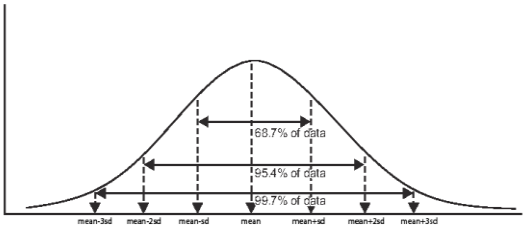

Figure 1. Normal distribution with mean and standard deviation

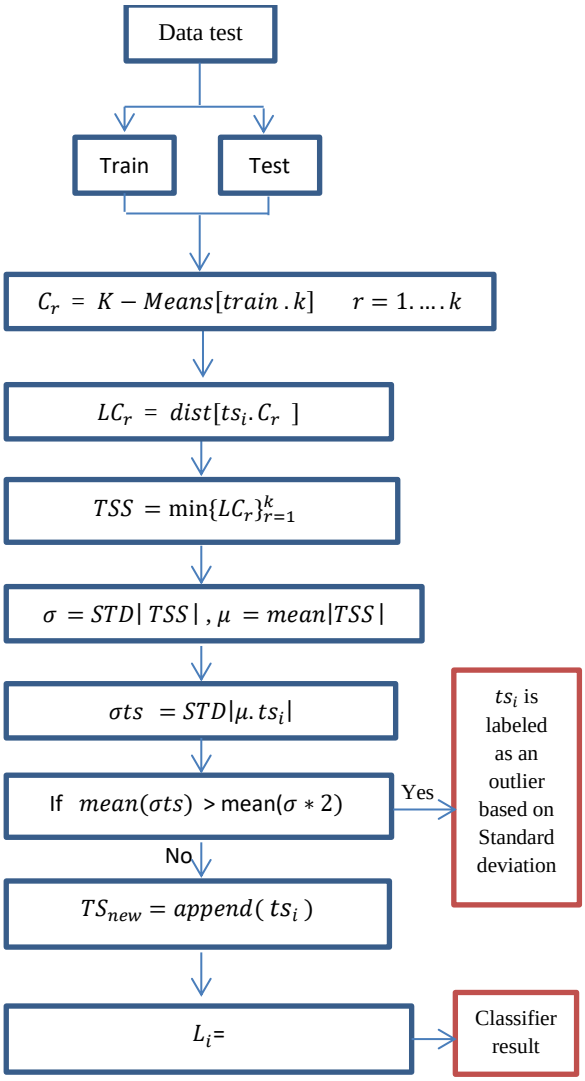

Figure 2. Overall procedure for the proposed outlier data detection method

TABLE 1. A sample data set

	F_1	F_2	F_3	⋯	F_m	LABEL
x_1	f_1^1	f_1^2	f_1^3	⋯	f_1^m	1
x_2	f_2^1	f_2^2	f_2^3	⋯	f_2^m	2
x_3	f_3^1	f_3^2	f_3^3	⋯	f_3^m	3
⋮	⋮	⋮	⋮	. . .	⋮	⋮
x_n	f_n^1	f_n^2	f_n^3	⋯	f_n^m	l

Finally, the new training set TSS with the least mean distance is dynamically determined for a test sample $\boldsymbol{TS_i}$. In the next step, it should be determined if the test sample $\boldsymbol{TS_i}$ is an outlier or not. For this purpose, first the standard deviation $\boldsymbol{\sigma}$ and then the mean value of the test set TSS are calculated. Afterwards, the distance between the test sample and the mean value of the new training set TSS is obtained. If the distance exceeds $\boldsymbol{\mu + 2\sigma}$ or is below $\boldsymbol{\mu - 2\sigma}$, the test sample is identified as an outlier data. This is illustrated in algorithm 2.

It is expected that through implementing Algorithm 4-2, outlier data will be removed and the classification accuracy will improve along with other measurement indices.

5. EXPERIMENTAL RESULT

We begin this section by introducing two- and multi-label data sets for the qualitative evaluation of the proposed method. Then the 5-fold validation function and the

Algorithm 1. finding $min\{C_r\}_{r=1}^{k}$

Input: $C_r = \{x_1.x_2. \ldots.x_k\}, ts_h$
Output: TSS

FOR $i = 1$ **to** k
  d[i] ← *Ave* dist$[ts_h.x_i]$
END
Min $= \infty$
FOR i=1 **to** k
  IF d[i] < Min **THEN**
    index = i
    Min = d[index]
  END IF
END FOR
$mc \leftarrow$ d$[index]$
$TSS = \boldsymbol{C_{mc}}$

Algorithm 2. Outlier data detection

Input:
1. $Train = (x_1.x_2.\ldots.x_n)$
2. $Ts = (t_1.t_2.\ldots.t_m)$
3. $k = \sqrt{n}$ $.k$ *is the number of cluster*

Output:
label of test Ts as an outlier and classsification

FOR $i \leftarrow 1$ **to** m
  $TSS = \min\{C_r\}_{r=1}^{k}.Ts$
  σ = *Standard Deviation*(TSS)
  μ = *mean*(TSS)
  $\sigma Ts = Standard\ Deviation(\mu.Ts)$
  IF mean(σTs) > $mean(2 * \sigma)$ ***THEN***
    Ts is labeled as an outlier
  ELSE
    $NewTs \leftarrow Ts$
  END *IF*
END *FOR*
$L_{pnew} \leftarrow$ Classifier $(Train.NewTs)$
$L_{pold} \leftarrow$ Classifier $(Train.Ts)$
$Compare(L_{pnew}.L_{pold})$

required hardware and software are given in section 5.2. The measurement criteria are presented in section 5.3. Finally, the results and discussions along with a comparison between the proposed method and other advanced algorithms are presented in section 5.4.

5. 1. Data Sets In this research, 12 data sets are used to evaluate the proposed method. Table 2 stated the number of samples, dimension (or characteristic) size and the class (or label) size of each data set. For evaluation of the proposed method, as described in section 4, eight credible data sets of Table 5 (rows 1 to 9) adopted from UCI reference are used. Also, three artificial data sets corresponding to rows 10 to 12 of Table 2 are used.

5. 2. Experimental Setup and Measurement Criteria In order to assess the efficiency of the proposed method, support vector machine (SVM) clustering is used for double-class data and Random Forest (RF) clustering is applied for multi-class data clustering. In this research, SVM is based on a linear core function with box limit 2. RF consists of several decision trees which usually performs better than a group of individual trees.

As mentioned, a 5-fold approach is adopted to evaluate the proposed technique such that the data set is divided into test and training sets. 20 percent of the data volume is randomly selected as the test data while the remaining 80 percent is used as the training set. Also, the final results are based on the means values obtained through several iterations.

The algorithm is run in MATLAB 2018 on a personal computer with Intel Core i5, 3.2 Hz along with 6GB RAM and hard drive capacity of 240GB SSD on 64bit Windows 10 operating system.

A major criterion in the algorithm evaluation is the classification accuracy. As mentioned, a certain classifier may not have the same accuracy for all classification tasks. Therefore, different methods are used for increasing the classifier accuracy. In this research, we use the training set selection (TSS) method.

TABLE 2. Data Sets

No	Data set	Sample size	Dimension size	Class size
1	Bcwisconsin	569	30	2
2	wisconsin	312	3	2
3	Australian	690	15	2
4	german	1000	25	2
5	heart	270	14	2
6	Ecoli	336	8	8
7	PV	210	19	7
8	Yeast	1484	9	10
9	CTG	2126	22	10
10	R15	600	3	15
11	Spr	312	3	3
12	pathbas	300	3	3

In order to determine the accuracy of the algorithms for double-class data, the following parameters should be determined:

True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN).

Given the above parameters, some indices like precision, recall and F-Measure can be calculated.

Accuracy: based on the above parameters, the classifier's accuracy for two-class data is determined by:

$$\mathbf{Accuracy_{two-labeled}} = \frac{\mathbf{TP+TN}}{\mathbf{TP+TN+FP+FN}} \quad (16)$$

In multi-class or multi-label data, the accuracy is the ratio of correctly-labeled data p_i to the total number of initially labeled data I_i as formulated in Equation (17).

$$\boldsymbol{Accuracy_{multi-labeled}} = \frac{1}{|TS|}\sum_{i=1}^{|TS|}\boldsymbol{Match}(\boldsymbol{p_i}.\boldsymbol{l_i}) \quad (17)$$

Precision: precision is defined as the ratio of data correctly classified into a certain class to the whole number of data put correctly or incorrectly into the same class, as calculated below:

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \quad (18)$$

Recall: is the ratio of data correctly classified into a certain class to the number of data put inside the same class, as calculated by:

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \quad (19)$$

F-measure: using the obtained Recall and Precision factors, the weighted index F-measure can be calculated. F-measure is a suitable parameter to evaluate the classification method and is also a weighted average of the precision and recall quantities. Ideally, F-measure is 1 for a well-performed classification algorithm while it becomes zero for the worst classification. This is calculated as:

$$\boldsymbol{F-measure} = \mathbf{2}\times\frac{\boldsymbol{Precision\times Recall}}{\boldsymbol{Precision+Recall}} \quad (20)$$

Normalized Mutual Information (NMI): This parameter gives the difference between the predicted labels and the true test data labels. In fact, this is a measure of the conformance of the labels between the two partitions.

For multi-class data sets having more than two labels, NMI is obtained as in Equation (21) where $\boldsymbol{h_a} = \{\boldsymbol{c_1^a}.\boldsymbol{c_2^a}.\ldots.\boldsymbol{c_{k_a}^a}\}$ and $\boldsymbol{h_b} = \{\boldsymbol{c_1^b}.\boldsymbol{c_2^b}.\ldots.\boldsymbol{c_{k_b}^b}\}$ are two clusters from data set D with n samples and ka and k_b clusters. n_{ij} is the cross-section of cluster c_i in the cluster set h_a with cluster c_j in the cluster set h_b. Also, n_{ia} is the

number of objects in cluster c_i of the cluster set h_a while n_{bj} is the number of objects in cluster c_j of the cluster set h_b.

$$\mathbf{NMI}(\mathbf{h_a}.\mathbf{h_b}) = \frac{-2\sum_{i=1}^{k_a}\sum_{j=1}^{k_b} n_{ij}\log\left(\frac{n\times n_{ij}}{n_{ia}\times n_{bj}}\right)}{\sum_{i=1}^{k_a} n_{ia}\log\left(\frac{n_{ia}}{n}\right)+\sum_{j=1}^{k_b} n_{bj}\log\left(\frac{n_{bj}}{n}\right)} \quad (21)$$

5. 3. Results of the Proposed Method The proposed method is applied on 12 sets of two-class and multi-class data as given in Table.2 Some of the data sets such as spr, r15 and pathbas are synthetic data sets. The data distribution within these sets are shown in Figure 3 where the data of the same labels are color-separated from others. K-fold averaging with k=5 is used to obtain the results. Also, the SVM classifier with linear core function and the RF classifier are used for two-class data and multi-class data, respectively. For two-class data sets, the number of outliers as well as the accuracy, precision, recall and f-measure parameters are obtained while for multi-class data, the accuracy and NMI indices are calculated to evaluate the proposed method.

Table 3 shows the accuracy, precision, recall and F-score parameters for two-class data sets while Table 4 gives the accuracy and NMI values for multi-class data sets prior to using the proposed approach.

Tables 5 and 6 show the same parameters after the application of the proposed method. The final rows give the number of detected outlier data. The accuracy does not necessarily increase with the increased outliers detected. This is because the classifiers may correctly determine the label of data whereas the data may later be identified as outlier. It is important to note that the

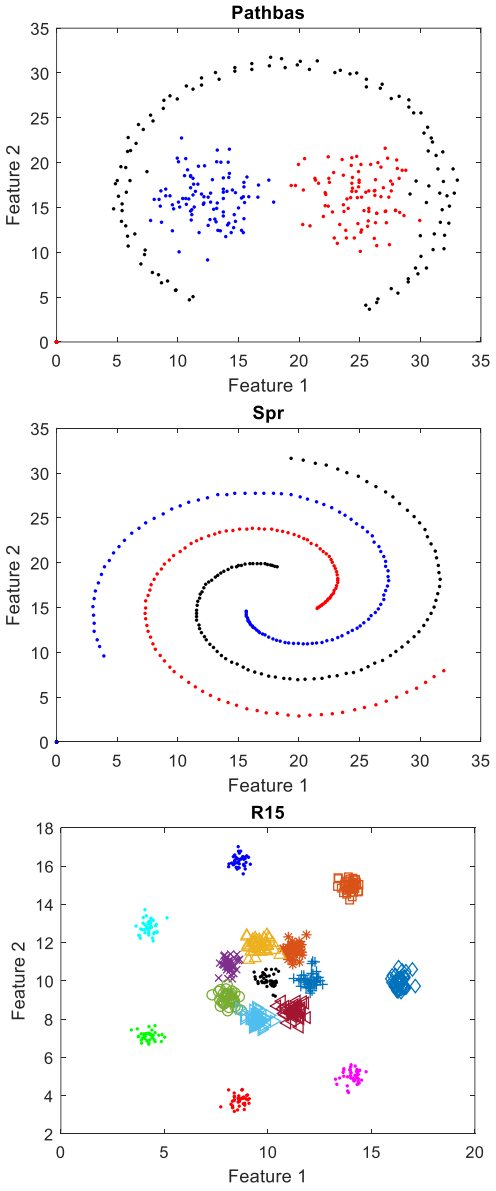

Figure 3. Artificial two-class data sets

TABLE 3. Evaluation parameters before application of the proposed method on the two-class data sets

Data sets	Accuracy	Precision	Recall	F-score
Bcwisconsin	0.95	0.97	0.95	0.96
Wisconsin	0.69	0.83	0.79	0.79
Australian	0.79	0.80	0.75	0.77
German	0.76	0.90	0.81	0.84
Heart	0.83	0.88	0.83	0.85

TABLE 4. Evaluation parameters before application of the proposed method on the multi-class data sets

Data sets	Accuracy	NMI
Yeast	0.61	0.40
CTG	0.91	0.82
Ecoli	0.84	0.74
PV	0.91	0.89
pathbas	0.99	0.95
R15	0.99	0.99
Spr	0.99	0.96

TABLE 5. Evaluation parameters after application of the proposed method on the two-class data sets

Data sets	MV Acc	MV Precision	MV Recall	MV Fscore	Outliers detected
Bcwisconsin	**0.96**	**0.98**	**0.96**	**0.97**	**7**
Wisconsin	0.69	0.83	0.79	0.79	0
Australian	0.79	**0.81**	0.75	**0.78**	**8**
German	0.76	0.90	0.81	0.84	**1**
Heart	**0.84**	**0.89**	**0.84**	**0.86**	**2**

TABLE 6. Evaluation parameters after application of the proposed method on the multi-class data sets

Data sets	MV Acc	MV NMI	Outliers detected
Yeast	**0.62**	0.40	**2**
CTG	0.91	0.82	**2**
Ecoli	**0.87**	**0.77**	**2**
Pv	**0.92**	**0.90**	**3**
Pathbas	0.99	0.95	0
R15	0.99	0.99	0
Spr	0.99	0.96	0

number of the identified outliers in tables 5 and 6 is the sum of the outliers in 5-fold execution. The bold parameters indicate an improvement compared to before application of the proposed method.

As indicated by the results of Tables 5 and 6, removing the outliers from the test samples and classification using the new test samples improves the final accuracy. Therefore, it can be said that outlier detection and removal has been effective on the results of the proposed method leading to improved measurement indices. The last column in Table 5 gives the number of detected outlier data. As seen, no outliers were detected using the proposed approach for Wisonsine data set while in BCwisconsine and Heart data sets, the accuracy has been improved through outlier detection and elimination.

Similarly, Table 6 gives the accuracy and NMI measurement for multi-class data sets after the application of the proposed method. Again, the last column is the number of outlier data in 5-fold execution. The bold values indicate improvement with respect to before implementing the proposed method.

Again, the results indicate that outlier detection and removal positively affects the outcome of the proposed approach leading to improved measurement indices. As shown in Table.6 , no outlier was detected in data sets spr, r15 and pathbas which can be justified by the kin of distribution shown in Figure 3 . As seen, the accuracy has been improved after outlier detection and removal from data sets *E. coli*, yeast and pv while the accuracy for CTG data set has remained the same despite the detection of two outlier data.

Furthermore, the experiment results and outlier data in Table 2 indicate that three data sets, namely, spr, r15 and pathbas do not contain any outliers. Nonetheless, to make sure of the accuracy of the proposed method, some outlier data were added to these data sets to observe if the proposed technique is able to detect these outliers. As the results indicate, the proposed method was able to detect all outliers from the three data sets. These outlier data are shown with mark x and in red color in Figure 4. The training sets are colored green and the regular test data are shown as blue dots.

The execution time given in Table 7 is the time-summation of k-means classification, determining the closest cluster, classification model and the classification of the test samples. The results of Table 7 indicate that even by including the k-means clustering time, the average execution time of the proposed approach, for all databases except german and Austration, is considerably better than ODT-SUTSS[1] that of the method proposed by Mohseni et al. [5].

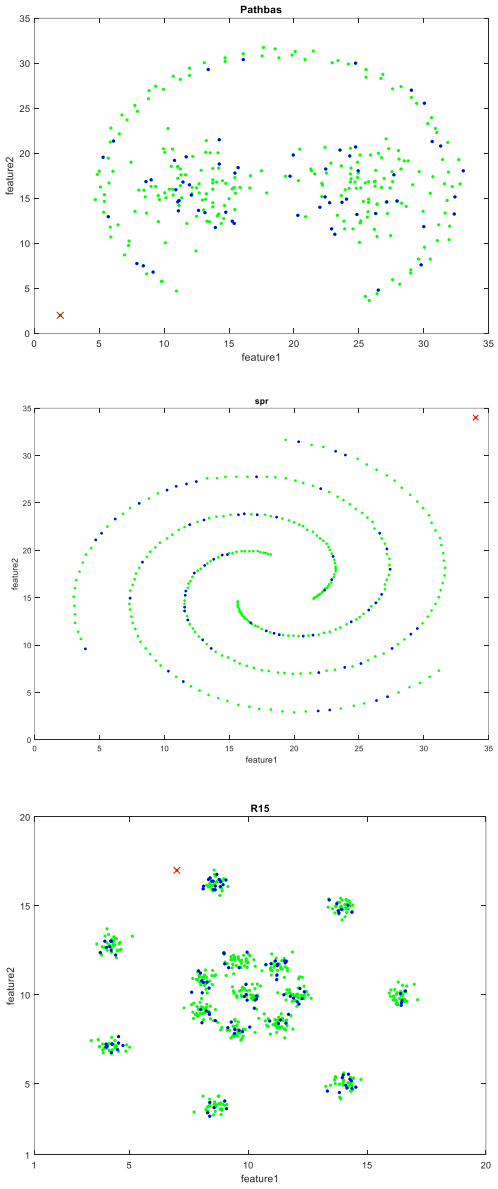

Figure 4. Training, test and outlier data distribution

[1] Outlier Detection Test - SUpervised Training Set Selection

TABLE 7. Comparison of the execution times

Methods / Data sets	Propose Method	Jacard Measure	Cosine Measure	Dice Measure	ODT-SUTSS method
Bcwisconsin	**2973**	608	2969	1024	4601
Wisconsin	**563**	207	480	287	974
Australian	5183	98	1671	385	**2154**
German	8903	67	2586	397	**3049**
Heart	**629**	410	629	539	1577
Yeast	**1208**	909	1163	1120	3192
CTG	**1903**	1625	1891	1874	5390
Ecoli	**195**	107	123	117	347
PV	**75**	41	65	54	160
Pathbas	**99**	88	97	94	278
R15	**240**	178	230	213	621
Spr	**108**	95	103	99	297

For a more accurate comparison, the execution time of the proposed method is calculated for each of jacard, cosine and dice criteria introduced in ODT-SUTSS method and is then compared against the total time and majority voting approach presented by Mohseni et al. [5].

For the ODT-SUTSS method proposed, three different criteria are used to select new training set for each sample of the test set. Then, using the majority voting technique, the outlier detection and predictions are performed. Therefore, the overall execution time given in the last column of Table 7 is higher. Given the results of execution times, the proposed method is faster than ODT-SUTSS method by Mohseni et al. [5] for a majority of cases.

In the following, the accuracy values in the proposed method is compared with ODT-SUTSS method, which is shown in Table 8. As indicated by the results of Table 8, accuracy values in the proposed method for most datasets is equal to ODT-SUTSS and for Yeast dataset the value has increased significantly.

Additionally, the accuracies of German and PV datasets are very close to ODT-SUTSS and are less than other datasets. *E coli* had the lowest accuracy . It should be noted that the accuracy values presented in Table 8 for ODT-SUTSS are the best values obtained for different alphas.

All in all, the interpretations in Table 8 confirms the proposed method achieved an acceptable degree of accuracy.

TABLE 8. Comparison of the accuracy

Methods / Data set	Propose method	OTD-SUTSS
Bcwisconsin	**0.96**	0.96
wisconsin	0.69	0.79
Australian	0.79	0.84
german	0.76	0.78
heart	**0.84**	0.84
Ecoli	0.62	0.86
PV	0.91	0.93
Yeast	**0.87**	0.63
CTG	**0.92**	0.92
R15	**0.99**	0.99
Spr	**0.99**	0.99
pathbas	**0.99**	0.99

6. CONCLUSIONS

In this paper, a new solution for outlier data detection in test samples and training data selection is proposed. As proposed, the outlier data can be detected and removed through the concepts of standard deviation prior to the classification. 12 different two-class and multi-class data sets were used to evaluate the efficiency of the proposed technique. As proposed, the data within training set are initially divided into several clusters and then, using distance criterion, the closest cluster to the test sample is identified. The data within the selected cluster are then used as the new training set. The standard deviation of the test sample with respect to the new training set is obtained and in case of falling outside the set limits, it is identified as an outlier data. As indicated by the results, the accuracy of classifiers is increased through efficient outlier detection while the execution time is reduced in majority of cases. The proposed method follows an unsupervised training procedure.

7. REFERENCES

1. Verbiest, N., Derrac, J., Cornelis, C., García, S. and Herrera, F., "Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis", ***Applied Soft Computing***, Vol. 38, (2016), 10-22. https://doi.org/10.1016/j.asoc.2015.09.006
2. Liu, C., Wang, W., Wang, M., Lv, F. and Konan, M., "An efficient instance selection algorithm to reconstruct training set for support vector machine", ***Knowledge-Based Systems***, Vol. 116, (2017), 58-73. https://doi.org/10.1016/j.knosys.2016.10.031
3. Zechner, M. and Granitzer, M., "A competitive learning approach to instance selection for support vector machines", in International Conference on Knowledge Science, Engineering and Management, Springer., (2009), 146-157.
4. Mohammed, A.M., Onieva, E. and Woźniak, M., "Training set selection and swarm intelligence for enhanced integration in multiple classifier systems", ***Applied Soft Computing***, Vol. 95, (2020), 106568. https://doi.org/10.1016/j.asoc.2020.106568
5. Mohseni, N., Nematzadeh, H. and Akbari, E., "Outlier detection in test samples and supervised training set selection", ***International Journal of Nonlinear Analysis and Applications***, Vol. 12, No. 1, (2021), 701-712. https://dx.doi.org/10.22075/ijnaa.2021.4878
6. Ren, Z., Wu, B., Zhang, X. and Sun, Q., "Image set classification using candidate sets selection and improved reverse training", ***Neurocomputing***, Vol. 341, (2019), 60-69. https://doi.org/10.1016/j.neucom.2019.03.010
7. Santiago-Ramirez, E., Gonzalez-Fraga, J.A., Gutierrez, E. and Alvarez-Xochihua, O., "Optimization-based methodology for training set selection to synthesize composite correlation filters for face recognition", ***Signal Processing: Image Communication***, Vol. 43, (2016), 54-67. https://doi.org/10.1016/j.image.2016.02.002
8. Smiti, A., "A critical overview of outlier detection methods", ***Computer Science Review***, Vol. 38, (2020), 100306. https://doi.org/10.1016/j.cosrev.2020.100306
9. Rath, S., Tripathy, A. and Tripathy, A.R., "Prediction of new active cases of coronavirus disease (covid-19) pandemic using multiple linear regression model", ***Diabetes & Metabolic Syndrome: Clinical Research & Reviews***, Vol. 14, No. 5, (2020), 1467-1474. https://doi.org/10.1016/j.dsx.2020.07.045
10. Chen, T., Martin, E. and Montague, G., "Robust probabilistic pca with missing data and contribution analysis for outlier detection", ***Computational Statistics & Data Analysis***, Vol. 53, No. 10, (2009), 3706-3716. https://doi.org/10.1016/j.csda.2009.03.014
11. Yang, Y., Fan, C., Chen, L. and Xiong, H., "Ipmod: An efficient outlier detection model for high-dimensional medical data streams", ***Expert Systems with Applications***, Vol. 191, (2022), 116212. https://doi.org/10.1016/j.eswa.2021.116212
12. Christy, A., Gandhi, G.M. and Vaithyasubramanian, S., "Cluster based outlier detection algorithm for healthcare data", ***Procedia Computer Science***, Vol. 50, (2015), 209-215. https://doi.org/10.1016/j.procs.2015.04.058
13. Lejeune, C., Mothe, J., Soubki, A. and Teste, O., "Shape-based outlier detection in multivariate functional data", ***Knowledge-Based Systems***, Vol. 198, (2020), 105960. https://doi.org/10.1016/j.knosys.2020.105960
14. Tang, B. and He, H., "A local density-based approach for outlier detection", ***Neurocomputing***, Vol. 241, (2017), 171-180. https://doi.org/10.1016/j.neucom.2017.02.039
15. Wang, B. and Mao, Z., "A dynamic ensemble outlier detection model based on an adaptive k-nearest neighbor rule", ***Information Fusion***, Vol. 63, (2020), 30-40. https://doi.org/10.1016/j.inffus.2020.05.001
16. Karlapalem, K., Cheng, H., Ramakrishnan, N., Agrawal, R., Reddy, P.K., Srivastava, J. and Chakraborty, T., "Advances in knowledge discovery and data mining: 25th pacific-asia conference, pakdd 2021, virtual event, may 11–14, 2021, proceedings, part i, Springer Nature, Vol. 12712, (2021).
17. Yang, J., Rahardja, S. and Fränti, P., "Mean-shift outlier detection and filtering", ***Pattern Recognition***, Vol. 115, (2021), 107874. https://doi.org/10.1016/j.patcog.2021.107874
18. Wahid, A. and Annavarapu, C.S.R., "Nanod: A natural neighbour-based outlier detection algorithm", ***Neural Computing and Applications***, Vol. 33, No. 6, (2021), 2107-2123. https://doi.org/10.1007/s00521-020-05068-2
19. Acampora, G., Herrera, F., Tortora, G. and Vitiello, A., "A multi-objective evolutionary approach to training set selection for support vector machine", ***Knowledge-Based Systems***, Vol. 147, (2018), 94-108. https://doi.org/10.1016/j.knosys.2018.02.022
20. Esfandian, N. and Hosseinpour, K., "A clustering-based approach for features extraction in spectro-temporal domain using artificial neural network", ***International Journal of Engineering***, ***Transactons B: Applications*** , Vol. 34, No. 2, (2021), 452-457. doi: 10.5829/ije.2021.34.02b.17.
21. Beulah, D. and Vamsi Krishna Raj, P., "The ensemble of unsupervised incremental learning algorithm for time series data", ***International Journal of Engineering***, ***Transactons B: Applications,*** Vol. 35, No. 2, (2022), 319-326. doi: 10.5829/ije.2022.35.02b.07.
22. Biglari, M., Mirzaei, F. and Hassanpour, H., "Feature selection for small sample sets with high dimensional data using heuristic hybrid approach", ***International Journal of Engineering***, ***Transactons B: Applications,*** Vol. 33, No. 2, (2020), 213-220. doi: 10.5829/ije.2020.33.02b.05.
23. Fränti, P. and Sieranoja, S., "How much can k-means be improved by using better initialization and repeats?", ***Pattern Recognition***, Vol. 93, (2019), 95-112. https://doi.org/10.1016/j.patcog.2019.04.014
24. Luchi, D., Rodrigues, A.L. and Varejão, F.M., "Sampling approaches for applying dbscan to large datasets", ***Pattern Recognition Letters***, Vol. 117, (2019), 90-96. https://doi.org/10.1016/j.patrec.2018.12.010
25. Akbari, E., Dahlan, H.M., Ibrahim, R. and Alizadeh, H., "Hierarchical cluster ensemble selection", ***Engineering Applications of Artificial Intelligence***, Vol. 39, (2015), 146-156. https://doi.org/10.1016/j.engappai.2014.12.005
26. Singh, D., Gosain, A. and Saha, A., "Weighted k-nearest neighbor based data complexity metrics for imbalanced datasets", ***Statistical Analysis and Data Mining: The ASA Data Science Journal***, Vol. 13, No. 4, (2020), 394-404. https://doi.org/10.1002/sam.11463
27. Chen, J., Zhang, C., Xue, X. and Liu, C.-L., "Fast instance selection for speeding up support vector machines", ***Knowledge-Based Systems***, Vol. 45, (2013), 1-7. https://doi.org/10.1016/j.knosys.2013.01.031
28. Nematzadeh, Z., Ibrahim, R. and Selamat, A., "Improving class noise detection and classification performance: A new two-filter cndc model", ***Applied Soft Computing***, Vol. 94, (2020), 106428. https://doi.org/10.1016/j.asoc.2020.106428
29. Speiser, J.L., Miller, M.E., Tooze, J. and Ip, E., "A comparison of random forest variable selection methods for classification prediction modeling", ***Expert Systems with Applications***, Vol. 134, (2019), 93-101. https://doi.org/10.1016/j.eswa.2019.05.028
30. Zhou, Q., Zhou, H. and Li, T., "Cost-sensitive feature selection using random forest: Selecting low-cost subsets of informative

features", ***Knowledge-Based Systems***, Vol. 95, (2016), 1-11. https://doi.org/10.1016/j.knosys.2015.11.010

31. Nematzadeh, Z., Ibrahim, R., Selamat, A. and Nazerian, V., "The synergistic combination of fuzzy c-means and ensemble filtering for class noise detection", ***Engineering Computations***, (2020). https://doi.org/10.1108/EC-05-2019-0242

32. Lee, D.K., In, J. and Lee, S., "Standard deviation and standard error of the mean", ***Korean Journal of Anesthesiology***, Vol. 68, No. 3, (2015), 220-223. https://doi.org/10.4097%2Fkjae.2015.68.3.220

Persian Abstract

چکیده

تشخیص دورافتاده تکنیکی برای شناسایی و حذف داده های بسیار متفاوت از داده های صحیح تر و سازگارتر در یک مجموعه داده است. داده های پرت می تواند تأثیر منفی بر عملکرد طبقه بندی و خوشه بندی داشته باشد. که باید شناسایی و حذف شوند تا کارایی طبقه بندی بهبود یابد. صرف نظر از اینکه یک تکنیک طبقه‌بندی، یک داده پرت را به درستی طبقه‌بندی می‌کند یا خیر، خود مفهوم شناسایی داده‌ها به‌عنوان پرت از اهمیت زیادی برخوردار است. در این مقاله، یک رویکرد جدید برای تشخیص داده‌های پرت در مجموعه داده‌های آزمایشی همراه با انتخاب مجموعه آموزشی بدون نظارت پیشنهاد شده‌است. مجموعه آموزشی انتخاب شده برای طبقه بندی دو مرحله ای استفاده می شود. پس از خوشه‌بندی بدون نظارت مجموعه آموزشی، نزدیک‌ترین خوشه به نمونه آزمایشی با استفاده از اندازه‌گیری فاصله اقلیدسی انتخاب می‌شود. سپس، نقطه پرت در نمونه آزمایشی با مفاهیم انحراف معیار و مقدار میانگین شناسایی می شود. نتایج با ارزیابی فاصله هر نمونه از مجموعه آزمون با مجموعه داده انتخابی جدید نشان داده شد. دقت طبقه‌بندی‌کننده‌ها پس از شناسایی و حذف داده‌های پرت افزایش می‌یابد.