ISSN:0975-9646

Supervised Distance Based Detection of Outliers by Reverse Nearest Neighbors Method

Trupti Rinayat[1] , Prof. Prashant Borkar[2]

Department of Computer Science and Engineering,

G.H.R.C.E. Nagpur, India

***Abstract*— Abstract - In data stream analysis, outlier detection has many applications as a branch of data mining and gaining more attention. . Data that is different or inconsistent from normal data set are identified by outlier detection. Unusual data records because of some data errors can be treated as outliers typically detecting outliers and analyzing large data sets identifies the problem such as medical problems, a structural defect, and experimental errors. This paper focuses the different methods for detection of anomalies. In order to handle the problems related to outlier detection because of uncertain data, outlier detection technique based on the AntiHub term is used.**

***Keywords*— Data stream, Data mining, outlier detection.**

I. Introduction

Data often arrives in stream and evolving over time. Rapid flow of data is termed as data streams, which are continuous in nature. As data is changing continuously, the chances of inaccurate predictions are high because of continuous change in data. Also opportunities to improve the accuracy might be missed. Most of the data systems assume that the training data are perfectly labeled for detecting the outliers. But due to uncertain data there may be data with imprecise labels. Addition of noise may also cause data uncertainty. When the new class are invoking in data stream, unseen classes evolve in data set that are outliers. Deceptiveness in performance of system, human or instrumental error, and abnormal deviations from populations induces outliers [1]. Anomalies are also referred to as outlier, novelties, noise, deviations and exceptions. Outlier detection is the task related to finding unexpected pattern which does not comply with rules or structure. Outliers in data are data objects which are unidentified values or inconsistent structures from normal pattern [2].

Detection of outliers from the dataset is concluded by characterizing the abnormal examples from the typical examples. The distance between the separating planes identifies how the examples or data is related to the trained data. Ranking done with the help of probability can also classify the outliers. Identification of the anomalous items leads to some kind of problems such as bank fraud or fraud detection. Like structural defect, medical problems or detecting errors in text.

Data occurrences with data labels determines that occurrence of data is or anomalous or normal. By identifying labels available for the data, the detection process for finding out the outliers is classified as three methods. First technique or first method is supervised detection, it works on trained database. Second is Semi-supervised detection, where no need of labels classes related to anomalous dataset and for normal labels is available. Third is unsupervised detection, does not need data to be trained.

II. Study Of Related Methods

For analysing the term cluster and the term anomalies discussion is there using the term cluster [3], [4]. In the intrusion domain for finding out deviations clustering method is used.

Inliers outlier detection which means inliers contains by training dataset are determined with the help of statistical method [5]. As the techniques for the density ratio calculation without analysis of estimated density in high dimensional are exists, which yields better results.

The angle-based outlier detection [6] method determines anomalies by analysing different angles for different vectors of objects for data which is high dimensional in nature. Difficulties occurring while detection are discussed in [7], [8]. An online anomaly detection method is proposed based on oversample PCA to identify the presence of rare but abnormal data. Anomaly detection problem formulations are:

- Recognizing abnormal pattern compared with normal succession;
- Determining different behavior for larger succession;
- Presence of frequency is abnormal is determined for succession

B. Liu et al. [9] used an SVDD-based approach for outlier detection on uncertain data by putting a kernel-based centre class method to generate a condensed score to each input sample. This indicates the likelihood of an example tending toward to normal class the information is used to refine the decision boundary of the distinctive classifier.

Better method for space containing slack is presented outside the decision boundary of each classification model is proposed [10], [12]. The new method is enhanced by making it more adaptive to the evolving stream. Outlier detection methods [11] on uncertain objects states that are originated to form a group are abnormal.

III. System Module

An outlier detector identifies data items that do not confirm an expected pattern or other data items in data stream. The detection of outlier helps in discovery of unexpected knowledge in data stream. System model for outlier detection process is shown in fig.1. It shows main phases for the process of identification of outliers and its components. These are explained as follows:

A. *Database collection and pre-processing*

For the input to the system, datasets are collected from the UCI depository sets. In the system module standard databases are used and the databases are supervised databases. Supervised dataset contains class labels according to the data type. Class labels are assigned on the basis of different attributes of the dataset. That means datasets already characterized into different classes based on the dataype. As per the type of dataset class labels according to class type are present. As a part of pre-processing, the missing values in the databases are filled with the value zero or as null.

The first pre-processing technique used is data cleaning. Data cleaning is carried out to find out the missing values in the data record. Also data cleaning is carried out to find out inconsistent data. The dataset used for outlier detection contained missing values and inconsistent data, for these data cleaning is carried out. As some attribute does not contain any value, for them as a part of data cleaning, values are inserted like zero and null.

The another part of pre-processing is data transformation. It includes conversion of data values present in the form of dataset into the data format for destination system requiring data from source system consisting data. Data transformation contains data mapping, which maps data from source to destination system. As the data is available on UCI depository, the data set is mapped into the destination system using data transformation method. Hence, the datasets are ready for the next step.

B. *AntiHub1 Method*

The method AntiHub1 is based on the ODIN method. ODIN method uses normal scores of outliers by analyzing the nearest neighbor count for the particular point. The ordered data set D is given as an input. Number of neighbors and distance from particular point is provided as input. Temporary variable AntiHub scores are used for comparing current discrimination score and raw outlier score. For each input 'Sn' is computed w.r.t. dist and data set (d/n).

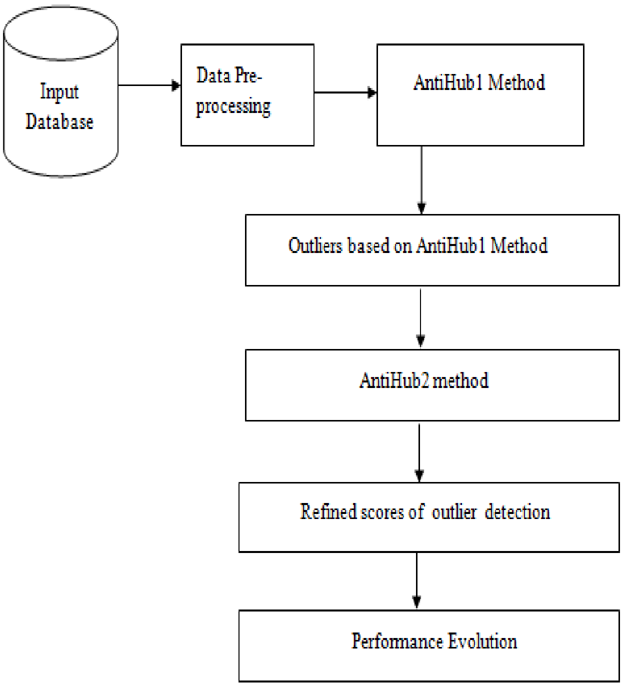

Fig. 1 Flowchart for outlier detection

The steps for AntiHub1 algorithm [13] are as follows:

1. For each point 'i' belongs to 1 to n,
2. $N_x(x) = D\backslash x_i$
3. s = f(t)

The steps for algorithm are described:

- Distance measured in the form of variable dist and ordered dataset as D is given as input for each point.
- For each point, distance from nearest neighbour is computed as $N_x(x)$ with respect to data set and dist variable.
- The score of outlier for point 'x' is computed using monotone function and stored in the vector's' as an output.

The Algorithm 1 AntiHub produces output in the form of vector as the outlier score of point x from data set D. Outliers are collected as output in the form of percentage outlier values.

C. *AntiHub2 Method*

As normalization is not applied on the outlier score, the outliers collected are not refined. To tackle weaknesses of AntiHub1, a simple heuristic method AntiHub2 is applied to the outlier score produced by the AntiHub1 method. AntiHub2 refines outlier scores produced by the AntiHub1 method by considering the Nk scores of the neighbors of x. For improvement in discrimination of scores that AntiHub2 introduces compared to AntiHub, for each point x, AntiHub2 proportionally adds the sum of Nk scores of the k nearest neighbors of x.

Input for AntiHub2 method contains measured distance for each point, ordered data set, number of neighbours and ratio of outliers for maximizing discrimination. Temporary variables are used for getting outlier score which are current discrimination score, current raw outlier scores, antihub score and sums of nearest neighbours' scores. AntiHub2 method is implemented for the refinement of the scores of outliers produced by AntiHub1 method.

The steps for AntiHub2 algorithm [13] are as follows:

```
1.  For each point 'i' belongs to 1 to n,
2.  AntiHub score a = AntiHub(D,k)
3.  $ann_i$ = summation of indices of k nearest neighbour
4.  disc = 0
5.  For each 'i' from 1 to n
6.  ct = $a_{i\,+}$ ann
7.  cdisc = discscore(ct,p)
8.  If cdisc > disc
9.  t=ct
10. s = f (t)
```

The steps for algorithm are described:

- Distance measured in the form of variable dist, ordered dataset as D and number of neighbours is given as input for each point.
- Ratio of outliers to maximize discrimination and search parameter for each step is initialised applicable to every point.
- Array for ordered dataset and number of neighbours is initialised as AntiHub(Dike).
- For each point sum of nearest neighbours' AntiHub scores as temporary variable 'ann' is calculated and stored. For each step from 0 to 1, loop is carried out, for each point.
- Raw outlier score 'ct' is calculated using proportion and point sum of nearest neighbours' AntiHub scores.
- Then the value raw outlier score 'ct' and ratio of outliers to maximize discrimination i.e. 'p' is transferred to temporary variable 'disc'.
- Comparison for temporary variables 'cdisc' and 'disc' is carried out, and if cdisc>disc then current raw outlier score't' is set as raw outlier score 'ct'.
- The score of outlier for point 'x' is computed using monotone function and stored in the vector's' as an output.

The second method considers the scores of neighbours for point x. And then adds sum of scores of nearest neighbour. To find aggregate of neighbours' scores summation is calculated. The discrimination scores compared using discScore parameter is provided to output vector as an outlier score.

IV. DATASET DESCRIPTION

For anomaly detection 4 real life datasets are used by other researchers earlier. The datasets are thyroid, Spam base, Diabetes, Abalone are available from UCI datasets [14]. The datasets are described in the table.

Thyroid dataset contain 29 attributes mostly the same set over all the databases. Mostly attributes are numeric or Boolean valued. Dataset contains 3 classes, 3772 training instances, 3428 testing instances. Outliers are calculated on the basis of weather patient is suffering from hypothyroid or not.

Table 1. Dataset description for datasets used in outlier detection

Dataset	Description	# of dataset	features
Thyroid	class 2 vs. rest 3	3772	21
Spam base	Others vs. spam	4601	57
Diabetes	Present vs. rest	768	8
Abalone	classes 1-8 vs. rest	4177	10

The SPAM E-mail Database includes 4601 instances, 58 attributes (57 continuous, 1 nominal class label). Diabetes dataset contains 768 instances. 8 plus classes are there. Abalone dataset is also taken which contains 4177 instances, 8 attributes. The value is continuous or abnormal is determined by the number of rings.

V. RESULTS

AntiHub1 method produces output in the form of vector as the outlier score of point x from data set D. Outlier percentage according to the dataset is enlisted in the table. The parameter 'n' represents the number of instances.

Table 2. Outlier detection percentage by AntiHub1 method

Dataset name	n	Outlier Values	Outlier%
Thyroid	3772	1704	45%
Spam base	4601	1813	39%
Diabetes	768	268	34%
Abalone	4177	2770	66%

Dimensionality is calculated by considering number of rows and number of columns. Skewness of distribution of N is represented in the form of Sn and calculated dividing'd' by 'n'. The parameters are discussed below:

- The parameter 'n' represents the number of instances.
- For Spam base dataset value of 'n' is 4601 as it contains 4601 number of instances.
- Dimensionality is calculated by considering number of rows and number of columns, which is represented as 'd'. Here value of'd' is 1303414.
- Skewness of distribution of N is represented in the form of'SN10 and calculated dividing'd' by 'n'.
- d/n = 1303414/4601
- Here, the value of SN10 obtained is 283.28928.
- Outlier values are 1813.
- The percentage of outlier is calculated as follows (Outlier value/n)*100 = (1813/4601)*100
- The percentage of outlier is calculated by dividing number of spams to number of instances which is 39%.

Outlier values are represented in the table. The Outlier scores are presented in the form of outlier percentage.

AntHub2 method

The method 2 takes the outlier values generated by the method1 as an input for refinement of outlier scores. The outlier percentage contains true class label and negative class label values. To refine the scores, more number of attributes is taken into consideration. So that the deviation from that point is exactly categorized into true or false class labels.

Table 3. Outlier detection percentage by AntiHub2 method

Dataset name	N	Outlier Values	Outlier%
Thyroid	3772	272	7.2%
Spam base	4601	1813	39.4%
Diabetes	768	268	34.9%
Abalone	4177	1044	24.99%

- The parameter ‘n’ represents the number of instances for the dataset a hypothyroid.
- Dimensionality is represented as ‘d’ and obtained by adding the values of attributes taken in consideration for each single row and calculated as 0.514+15+0.455 = 15.609.
- The dimension of single row is represented in the form of ‘D’, calculated as $\sum d = D[i]$.
- Nk represents the nearest neighbour deviation from the instances.
- Skewness of distribution of N is represented in the form of SN10.
- The time taken for computation is also shown in the form of star time and end time.
- The time taken for calculating the parameters shows the total time for computation.
- The percentage of outlier is calculated as follows
  (Outlier value/n)*100 = (3772/272)*100
- The percentage of outlier is calculated by dividing number of spams to number of instances which is 7.2%.

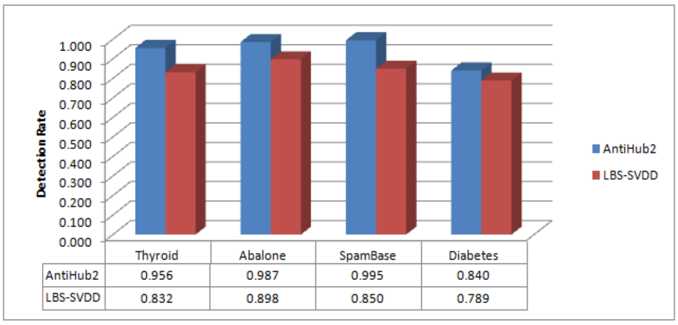

Fig. 2. Graphical representation of performance of outlier detection approaches on data sets

False alarm rate and detection rate are used for evaluation of performance of outlier detection. Detection rate and false alarm rate are determined on the basis of predicted labels. Predicted labels contain true class and false class. Number of exactly determined anomalies is described by detection rate. The exact description of outliers that are wrongly classified into normal class data. The results obtained are compared with results obtained in [15] for supervised dataset.

VI. Conclusion

As detection of outliers is gaining popularity for faster detection rate and security purpose, in this paper emphazization on the parameters necessarily required for the detection and their inter-relationship is given. Connection among basic parameters is to be clarified so that an attempt for outlier detection is more efficient.

In this paper, relative impacts of basic parameters dependencies on outlier detection are discussed. Influence for values and datasets and their inter-relationship are also established. To conclude, this paper helps to understand the fact about complete analysis of nature of task is to be modelled prior to the algorithmic choice for outlier detection. Adaptive nature of algorithm is needed as nature of detection differs from application to application.

Refernces

[1] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly Detection: A survey,” ACM CSUR, vol. 41, no. 3, Article 15, 2009.

[2] V. J. Hodge and J. Austin, “A survey of outlier detection methodologies,” Artif. Intell. Rev., vol. 22, no. 3, pp. 85–126, 2004.

[3] Y. Shi and L. Zhang, “COID: A cluster-outlier iterative detection approach to multi-dimensional data analysis,” Knowl. Inform. Syst., vol. 28, no. 3, pp. 709–733, 2011.

[4] R. Smith, A. Bivens, M. Embrechts, C. Palagiri, and B. Szymanski, “Clustering approaches for anomaly based intrusion detec- tion,” in Proc. Intell. Eng. Syst. Artif. Neural Netw., 2002, pp. 579–584.

[5] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori, “Statistical outlier detection using direct density ratio estimation,” Knowl. Inform. Syst., vol. 26, n4o. 2, pp. 309–336, 2011.

[6] H.P. Kriegel, M. Schubert, and A. Zimek, “Angle-based outlier detection in high-dimensional data,” in Proc 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2008, pp. 444–452.

[7] Y. J. Lee, Y. R. Yeh, and Y. C. F. Wang, “Anomaly detection via online over-sampling principal component analysis,” IEEE Trans. Knowl. Data Eng., vol. 25, no. 7, pp. 1460–1470, May 2012.

[8] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection for discrete sequences: A survey,” IEEE Trans. Knowl. Data Eng., vol. 24, no. 5, pp. 823–839, May 2012.

[9] B. Liu, Y. Xiao, L. Cao, Z. Hao, and F. Deng, “Svdd-based outlier detection on uncertain data,” Knowl. Inform. Syst., vol. 34, no. 3, pp. 597–618, 2013.

[10] M. Masud, Q. Chen, L. Khan,J. Gao and J. Han “Classification and Adaptive Novel Class Detection of Feature Evolving Data Streams”, IEEE Trans. Knowl. Data Eng., vol. 25, no. 7, July 2013.

[11] S. Ahmed Shaikh and H. Kitagawa “Continuous Outlier Detection on Uncertain Data Streams”, IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP) Symposium on Information Processing Singapore, 21–24 April 2014 .

[12] Bo Liu, Yanshan Xiao, Philip S. Yu, Zhifeng Hao, and Longbing Cao, “An Efficient Approach for Outlier Detection with Imperfect Data Labels”, IEEE Trans. Knowl. Data Eng., vol. 26, no. 7, July 2014.

[13] Milos Radovanovic, Alexandros Nanopoulos, and Mirjana Ivanovi“Reverse Nearest Neighbors in Unsupervised Distance-Based Outlier Detection”, IEEE Transactions On Knowledge And Data Engineering, Vol. 27, No. 5, May 2015.

[14] UCI Machine Learning Repository [Online]. Available: http://archive.ics.uci.edu/ml/datasets.html.

[15] Bo Liu, Yanshan Xiao, Philip S. Yu, Zhifeng Hao, and Longbing Cao, “An Efficient Approach for Outlier Detection with Imperfect Data Labels”, IEEE Trans. Knowl. Data Eng., vol. 26, no. 7, July 2014.