Outlier Analysis in Sensor Data Streams: A case study

Nripesh Trivedi, Jean-Paul Calbimonte
IIT (BHU) Varanasi, IIG, HES-SO
nripesh.trivedi.apm11@itbhu.ac.in, jean-paul.calbimonte@hevs.ch

***Abstract*— In this paper, a method to detect outliers is described in a data oriented manner (rather than algorithm-oriented manner) . This method consists of application of Class Outliers: Distance Based (CODB) and Hoeffding tree algorithm. Subsequently, machine learning models were built to detect outlier in data streams. The case study outlined in this paper could be used for building effective and real-time methodologies for outlier detection.**

I. Introduction and Motivation

Many algorithms have been proposed to detect outliers. However, for this paper, a data oriented approach is adopted that is exclusive to the data-set under consideration. An example of a data-oriented approach could be found in [4]. The authors use heavy tailed properties of features of data to solve the problem of segmenting users by their engagement. In a data-oriented approach, as a beginning step, properties of attributes of data like distribution of data, variation in data, co-relation among data attributes etc. is identified[4]. In subsequent steps, algorithms are applied on the data according to its properties. This paper may be the first to diverge from a algorithm oriented approach to a data-oriented approach for outlier analysis.

II. Outlier detection using Data oriented methodology

A. Data-set Description

The Dataset used in the paper was made available by Dr. Jean Paul Calbimonte who is actively involved with OpenSense project. Size of the data-set is around 16 million rows. Around 100000 rows of the dataset were used to train machine learning models while other rows of the dataset were used for building a data stream that these machine learning models could learn from and predict on.

The attributes present in the dataset are :

- Latitude
- Longitude
- Station
- LDSA

In OpenSense project, sensors were mounted on top of the buses to measure pollution levels in Lausanne city (Switzerland). The term LDSA stands for lung deposited surface area. It's a way to measure the quantity of particles. The term LDSA is a abbreviation for lung deposited surface area. It's a way to measure pollution.

.

10 stations indicated in the table I includes both mobile and static stations.

TABLE I
Data Attribute and their respective spread of values

Latitude	46.5202347 - 46.5218066
Longitude	6.6307456 - 6.6315791
Station	41, 43, 45, 47. 48, 49, 50, 51, 54, 55
LDSA	1 - 2000

B. Methodology for outlier detection

Uniform Random sampling is a popular method for summarizing multidimensional data streams[1]. Using this method, original data-set was sampled. Values of the LDSA attribute for the new dataset, say dataset I, obtained after sampling the original dataset are shown in figure 1. From figure 1, it can be seen that in dataset I, values of LDSA attribute have similar ranges for pairs of stations. Pairs of stations having similar range of values for LDSA attribute are shown in table II. Values of attributes Latitude and Longitude in dataset I are not shown since both vary within quite small range that makes their representation inadequate. Values of station attribute for dataset I are shown in table I.

TABLE II
paired Stations

Pair number	Station number 1	Station number 2
1	49	50
2	45	54
3	41	47
4	43	55
5	48	51

Station is a numerical attribute but it may be treated as a class(nominal) attribute. The advantage of treating Station attribute as a class attribute is that class labels could be assigned to every point in original data-set and data-set I. A single instance in Data-set is of the form (Station, Latitude, Longitude, LDSA) where Station is nominal attribute and other three are numerical attribute.

Class Outliers: Distance Based(CODB) Algorithm was applied over the data-set in the following manner .:

- Since Station attribute is a nominal attribute, Latitude, Longitude and LDSA attributes are independent of the Station attribute. Since data values of station attribute are independent of each other as station is a nominal attribute, data values of Latitude, Longitude and LDSA attributes for each station are also independent of every

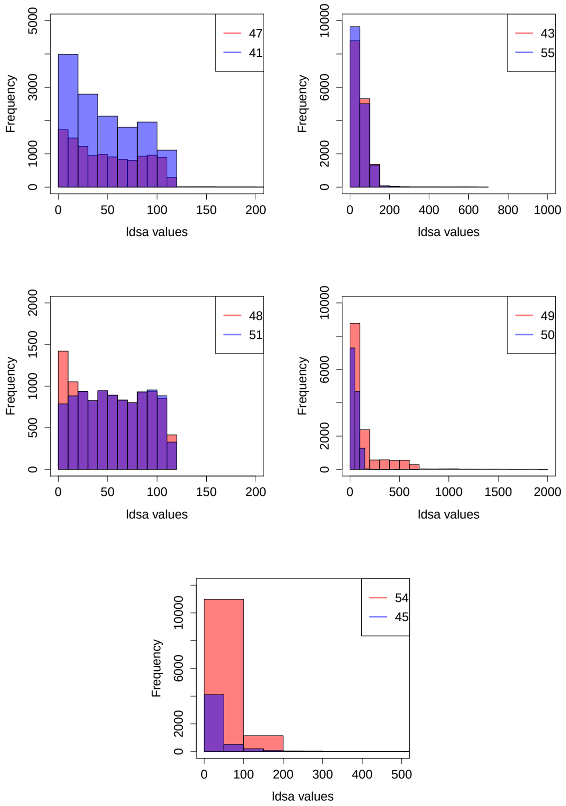

Fig. 1. *This figure shows spread of LDSA values in data-set I for 5 pairs of stations.*

other station. Thus when CODB algorithm is applied to the data-set I, outliers detection for each station is independent of every other station. Moreover, Since decision to treat class attribute as a nominal attribute is central to the decision of application of CODB and (also of Hoeffding tree (VFDT) later in the paper), ,the outlier analysis in this paper is data-oriented.

The above statement shows that data has a central role in outlier analysis. CODB algorithm was applied on each of the five pairs of Stations in data-set I. After running initial experiments on data-set I, maximum value of deviation was found to lie in thousands and maximum value of K-distance was nearly 0.0. Therefore, as suggested in [2], these parameters were set as in table below. Since maximum value of K-Distance was nearly 0.0, choice of corresponding parameter may be any arbitrary real number between 0 and 1 as suggested in [2]. Number of nearest neighbors (K) were set to 7 as again suggested in [2] .

Given the varying quality of measurements from different sensors in OpenSense project, top 20% of the outlier values from each pair were taken into consideration.

CODB algorithm has three components:

- PCL (T, K)
- Dev (T)
- K-Dist (T)

TABLE III

PARAMETERS WITH RESPECTIVE VALUES

Parameter	Value
α	1000
β	0.1
k	7

Where T is the instance for which COF (T) is evaluated and K is number of nearest neighbors. While evaluating Dev (T) and K-Dist (T), three numerical attributes within the data-set are used namely, Latitude, Longitude and LDSA. The value of PCL (T, K) is a number between 0 and K and indicates nearest neighbors that belong to the same class.Using the settings of parameters in Table IV, CODB algorithm was applied to every pair of stations to yield the outliers in dataset I.

For detecting outlier in the data stream (detailed description of the data stream could be found in data-set description subsection), it is necessary to choose an algorithm that could build machine learning models using a small sample of data and then these machine learning models could be used for prediction. Further, these machine learning models should also be capable of learning while carrying out prediction. Hoeffding tree (VFDT) provides a robust solution for the this requirement [3].

During application of CODB to data-set I, outliers were found independent of the data distribution of other Stations in data-set I. Hoeffding tree (VFDT) was trained over data-set I. Since for each of the pair in table (IV), a separate Hoeffding tree algorithm was trained, five models were obtained after training a Hoeffding tree algorithm for each pair of stations. The efficiency of models are shown in tables below-:

TABLE IV

EFFICIENCY FOR PAIR OF STATION

Pair number	Overall Efficiency
1	92.66%
2	96.21%
3	95.32%
4	95.30%
5	97.73 %

Since minority class (outliers) is 20% of the data-set I while the majority class (non- outlier) is 80% , it is necessary to verify the strength of machine learning models by using precision and recall. Therefore, 10 fold cross validation was applied over five models.

Table (V) shows the precision and recall values for every pair of stations in data-set I. Precision and recall values in table V verify the strength of all the models.

III. CONCLUSION AND FUTURE WORK

The data oriented approach shown in this paper is specific to OpenSense data. In order to come up with a generalized approaches, it is necessary to find common properties in

TABLE V

RECALL AND PRECISION VALUE FOR EACH PAIR

Pair number	Recall	Precision
2	87.38%	93.26%
3	81.85%	93.97%
4	83.90%	91.92%
5	92.66%	95.81%
1	84.35%	98.36%

data obtained from various sensor networks. These common properties could be used to design generalized data oriented approaches. The future work in this field involves identification of characteristics of sensor data and designing generalized data oriented approaches.

REFERENCES

[1] Fabio Fumarola Donato Malerba Annalisa Appice, Anna Ciampi. *Data Mining Techniques in Sensor Networks: Summarization, Interpolation and Surveillance*. Springer-Verlag London, 2013.

[2] Nabil M Hewahi and Motaz K Saad. Class outliers mining: Distance-based approach. *International Journal of Intelligent Technology*, 2(1):55–68, 2007.

[3] Geoff Hulten, Laurie Spencer, and Pedro Domingos. Mining time-changing data streams. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 97–106. ACM, 2001.

[4] Nripesh Trivedi, Daniel Adomako Asamoah, and Derek Doran. Keep the conversations going: engagement-based customer segmentation on online social service platforms. *Information Systems Frontiers*, pages 1–19, 2016.

.