A Comparative Study on Outlier Detection Techniques

Mohammad Zaid Pasha
Department of Computer Science and Engineering
Lovely Professional University, Punjab, India

Nitin Umesh
Department of Computer Science and Engineering
Lovely Professional University, Punjab, India

ABSTRACT

Outlier detection is an extremely important problem with direct application in a wide variety of domains. A key challenge with outlier detection is that it is not a well-formulated problem like clustering. In this paper, discussion on different techniques and then comparison by analyzing their different aspects, essentially, time complexity. Every unique problem formulation entails a different approach, resulting in a huge literature on outlier detection techniques. Several techniques have been proposed to target a particular application domain. The classification of outlier detection techniques based on the applied knowledge discipline provides an idea of the research done by different communities and also highlights the unexplored research avenues for the outlier detection problem. Discussed of the behavior of different techniques will be done, in this paper, with respect to the nature. The feasibility of a technique in a particular problem setting also depends on other constraints. For example, Statistical techniques assume knowledge about the underlying distribution characteristics of the data. Distance based techniques are typically expensive and hence are not applied in scenarios where computational complexity is an important issue.

Keywords – Outlier, time complexity, statistical techniques, eucledian distance

1. INTRODUCTION

Most of the organizations, these days, produce an electronic record of every transaction they are involved in. In large Organizations, this results in millions of records being produced every day. Nowadays many Organizations are going online to exploit the e-business wagon; this will result in huge amount of data being accumulated as the Internet connects many sources of data. The accumulated data is very important in today's competitive world and its used for gaining competitive edge over competitors by a process called data mining, which can be said to be the extraction of useful information from large databases. Data-mining being a new area has seen many sophisticated algorithms and tools being developed.

Clustering in data mining is a discovery process that groups a set of data so that the inter-cluster similarity is minimized and intra-cluster similarity is maximized. Clustering techniques consider data records as objects. They partition the objects into consider groups or clusters are similar otherwise they will be considered as dissimilar.

Similarity is defined in terms of how close the objects are in data space. The quality of a cluster may be represented by its diameter, the maximum distance between any two objects in the cluster. Centroid distance is an alternative measure of cluster quality and is defined as the average distance of each cluster object from the cluster centroid. There are several clustering techniques, organized into the following categories: partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods. A database may contain data objects that do not comply with the general behavior or model of the data. These data objects are outliers. Most of the data mining methods discard outliers as noise or exceptions. There are number of outlier's detection methods, categorized into four approaches: statistical approach, distance-based approach, density-based local outliers approach and frequency-based approach. So one of the challenging tasks is detecting the outliers and removing the outliers precisely in a data set.

As it is mentioned earlier, the primary objective of this of this paper is to compare the outlier handling techniques, one of the primary data mining tasks. Outlier detection is an important problem that has been researched within diverse knowledge disciplines and application domains. Many of these techniques have been specifically developed for certain application domains, while others are more generic. This survey tries to provide a structured and comprehensive overview of the research on outlier detection and we will provide a novel approach to solve the problem of outlier detection. This survey is intended to provide a better understanding of the different directions in which research has been done on this topic and how techniques developed in one area can be applied to applications for which they were not intended to begin with. Some of the definitions of an outlier are as follows:

Outliers are patterns in data that do not conform to a well defined notion of normal behavior.[10]

It can also be defined as a data point which is very different from the rest of the data based on some measure. Such a point often contains useful information on abnormal behavior of the system described by data. A key challenge in outlier detection is that it involves exploring the unseen space. As mentioned earlier, at an abstract level, an outlier can be defined as a pattern that does not conform to expected normal behavior. A straightforward approach will be to define a region representing normal behavior and declare any observation in the data which does not belong to this normal region as an outlier. But several factors make this apparently simple approach very challenging such as defining a normal region which encompasses every possible normal behavior is very difficult.

2. REVIEW OF LITERATURE

There are different approaches as far as the outlier detection is concerned like statistical approach, distance based approach, density based approach and frequency based approach. Different approaches were used by different researchers for outlier detections and they come up with different pros and cons in all approaches. Here we will discuss all the approaches with examples one after another.

2.1 Statistical approach

[3] The statistical approach assumes a distribution or probability model for the given set and identifies outliers with respect to the model using a discordance test. A discordance test is used to detect whether a given object is an outlier or not

Control chart technique is an example under statistical approach for outlier data detection. Usually, CCT is used to determine whether your process is operating in statistical control. The purpose of a control chart is to detect any unwanted changes in the process. These changes will be signaled by abnormal (outlier) points on the graph. Basically, control chart consists of three basic components:

1) A centre line, which is the mathematical average of all the samples plotted.

2) Upper and lower control limits that define the constraints of common cause variations.

3) Performance data plotted over time.

Firstly, calculate the average for data points to get a centerline of a control chart. The formula is,

$$\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

Where,

X = mean/average value

X_i = every data value ($X_1 \ldots X_n$)

n= total number of data

Where Z is equal to 3.Here, the unusual causes of variation can be detected. In this manner we can plot UCL, LCL and the mean line and the data objects which lie above the UCL and below the LCL will be considered as outliers.

Calculation of the upper control (UCL) and lower control limit (LCL) by using formula below,

$$\text{UCL (calculated)} = \overline{X} + Z\sigma_x$$

$$\text{LCL (calculated)} = \overline{X} - Z\sigma_x$$

$$\sigma_x = \frac{\sigma}{\sqrt{n}}$$

$$\sigma = \text{standard deviation} = \left[\frac{\sum (X_i - \overline{X})^2}{n-1}\right]^{\frac{1}{2}}$$

The problem with this approach is that it requires considerable amount of knowledge about the data set we are dealing with so that we can come up with the required value of different parameter used.

2.2 Distance based approach

In this approach, similarity between two objects is measured with the help of distance between the two objects in data space, if this distance exceeds a particular threshold, then the data object will be called as the outlier. There are many algorithms under this category. One of the most popular and simple to implement is K neighbor technique. This technique operates under the assumption that normal points have several closely located neighbors, while outliers are located far from other points. In the first step a neighborhood for each point is computed, using a distance or similarity measure defined between two data instances. Here Euclidean distance can be used to measure the distances between the point under consideration and every data point in the dataset. In the second step the neighborhood is analyzed to determine if the point an outlier or not.

Algorithm-

Input: D, t, and k

Where;

D is the dataset

T is the predefined threshold

K is the number of neighbors for each test point

Output: S (say, it is the set of outliers)

For i = 1 to n do

For each test sample t_i, find the distance $d(t_i, t_j)$, between t_i and every example t_j belonging to D.

End for

Sort the distances in ascending order

Let the k^{th} smallest distance is k-dist(i)

if k-dist(i) > T then

$S = S \cup t_i$

End if

End for

The distance measured for each test sample is compared and the test sample with highest distance measure shows the maximum “ degree of being outlier” or we can say it is most deviated element in the dataset.

2.3 Density based approaches

Although, the previously disused technique is simple to implement and efficient to a considerable approach, but it is suffered from some kind of drawbacks like it does not account the “locality” of the data element which can led to misjudge the outlier in certain kinds of datasets. Some examples under this category are LOF, LOF’, LOF’’, DSNOF and many more which determines the degree of being a outlier of the data element while considering the locality of the data element and hence named density based approaches. Here we will discuss these approaches one after another. At first, we will discuss LOF, then LOF’ and then LOF’’ and at last DSNOF will be discussed.

2.3.1 LOF (Local Outlier Factor)

[2]It is the first concept of an object which also quantifies how outlying an object is and the LOF value of an object is

based on the average of the ratios of the local reachability density of the area around the object and the local reachability densities of its neighbors. The size of the neighborhood of the object is determined by the area containing a user-supplied minimum number of points (*MinPts*). Several concepts and terms to explain the LOF algorithm can be defined as follows.

Definition 1: (k-distance of an object *p*) For any positive integer *k*, the k-distance of object *p*, denoted as k-distance(*p*), is defined as the distance *d(p,o)* between *p* and an object of *D* such that:

(i) for at least *k* objects $o' \in D\backslash\{p\}$ it holds that $d(p,o') \leq d(p,o)$, and

(ii) for at most $k - 1$ objects $o' \in D\backslash\{p\}$ it holds that $d(p,o') < d(p,o)$.

Definition 2: (k-distance neighborhood of an object *p*) Given the k-distance of *p*, the k-distance neighborhood of *p* contains every object whose distance from *p* is not greater than the k-distance

Definition 3: Reachability distance of an object *p* w.r.t. object *o*) Let *k* be a natural number. The reachability distance of object *p* with respect to object *o* is defined as

$$reach-dist_k(p) = \max\{k-distance(o), d(p,o)\}$$

Let *MinPts* be the only parameter and the values $reach\text{-}dist_{MinPts}(p,o)$, for $o \in N_{MinPts}(p)$, be a measure of the volume to determine the density in the neighborhood of an object *p*.

Definition 4: (local reachability density of an object *p*) .The local reachability density of *p* is defined as the reciprocal of summation of reachability distance of all points divided by MinPts.

Definition 5: Local outlier factor of an object p is

$$LOF_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts}(p)} \frac{lrd_{Minpts}(o)}{lrd_{MinPts}(p)}}{|N_{MinPts}(p)|}$$

2.3.2 LOF'

This algorithm proposed a better formulation compared with LOF. Unlike the method of connectivity based outlier factor (COF) in which the focus is on outlier detections for low density patterns, this enhancement scheme improves the efficiency and effectiveness of LOF for general datasets. It can be seen that the notion of LOF is quite complex. Three components including MinPts-dist, reachability distance and local reachability density are to be understood before the understanding of the LOF formulation.

Local reachability density is an indication of the density of the region around a data point. LOF' argue that MinPtsdist already captures this notion: a large MinPts-dist corresponds to a sparse region; a small MinPts-dist corresponds to a dense region. In view of this, LOF_ is defined as a simpler formula for ease of understanding, and also simpler computation.

Definition 1: (minpts-distance of an object *p*) For any positive integer *minpts*, the minpts-distance of object *p*, denoted as minpts-distance(*p*), is defined as the distance *d(p,o)* between *p* and an object of *D* such that:

(i) for at least *minpts* objects $o' \in D\backslash\{p\}$ it holds that $d(p,o') \leq d(p,o)$, and

(ii) for at most $minpts - 1$ objects $o' \in D\backslash\{p\}$ it holds that $d(p,o') < d(p,o)$.

Definition 2: (minpts neighborhood of an object *p*) Given the minpts distance of *p*, the minpts neighborhood of *p* contains every object whose distance from *p* is not greater than the minpts.

Definition 3: LOF' can be calculated as:

$$LOF'_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts-dist(p)}(p)} \frac{MinPts-dist(p)}{MinPts-dist(o)}}{|N_{MinPts-dist(p)}(p)|}$$

2.3.3 LOF''

There is a slight variation in LOF' and hence it is named as LOF''. Sometimes outlying objects may be quite close to each other in the data space, forming small groups of outlying objects. Since MinPts reveals the minimum number of points to be considered as a cluster, if the MinPts is set too low, the groups of outlying objects will be wrongly identified as clusters. On the other hand, MinPts is also used to compute the density of each point, so if MinPts is set too high, some outliers near dense clusters may be misidentified as clustering points.

LOF" uses two different neighbourhoods:

(1) Neighbours in computing the density and

(2) Neighbours in comparing the densities.

In LOF, these two neighbourhoods are identical.

Definition 1: (k-distance of an object *p*) For any positive integer *k*, the k-distance of object *p*, denoted as k-distance(*p*), is defined as the distance *d(p,o)* between *p* and an object of *D* such that:

(i) for at least *k* objects $o' \in D\backslash\{p\}$ it holds that $d(p,o') \leq d(p,o)$, and

(ii) for at most $k - 1$ objects $o' \in D\backslash\{p\}$ it holds that $d(p,o') < d(p,o)$.

Definition 2: (k-distance neighborhood of an object *p*) .Given the k-distance of *p*, the k-distance neighborhood of *p* contains every object whose distance from *p* is not greater than the k-distance.

Definition 3: (reachability distance of an object *p* w.r.t. object *o*) Let *k* be a natural number. The reachability distance of object *p* with respect to object *o*

$$reach-dist_k(p) = \max\{k-distance(o), d(p,o)\}$$

Let *MinPts2* be the only parameter and the values $reach\text{-}dist_{MinPts2}(p,o)$, for $o \in N_{MinPts2}(p)$, be a measure of the volume to determine the density in the neighborhood of an object *p*.

Definition 4: (local reachability density of an object *p*) .The local reachability distance is defined as

$$lrd_{Minpts}(p) = 1/\left[\frac{\sum_{o \in N_{MinPts}(p)} reach-dist_{MinPts}(p,o)}{|N_{MinPts}(p)|}\right]$$

Definition 5: The local outlier factor of an object can be given as

$$\mathrm{lof}(p) = \frac{\sum_{o \in N_{MinPts1-dist(p)}(p)} \frac{lrd_{Minpts2}(o)}{lrd_{MinPts2}(p)}}{|N_{MinPts1-dist(p)}(p)|}$$

2.3.4 DSNOF

[5]In this algorithm, each object in dataset is assigned a density-similarity-neighbor based outlier factor (DSNOF) to indicate the degree (intensity) of outlier possessed by an object. This algorithm calculates the densities of an object and its neighbors and constructs the similar density series (SDS) in the neighborhood of the object. Based on the SDS, the algorithm computes the average series cost (ASC) of the objects. Finally, it calculates the DSNOF of the object based on the ASC of the object and its neighbors.

Procedure: The DSNOF Algorithm

Input: a dataset *D* and any positive integer *k*

Output: the DSNOF values of the objects in *D*

Step1: Calculating the distance between objects and finding the *k* -distance neighbors of each object in *D.*

Step2: Calculating the density of each object in *D*. If *p* be an object of *D*, then the density of *p* , *density*(*p*) , is defined as

$$density(p) = \frac{|N_k(p)|}{kdist(p)}$$

Where kdist(p) represents the k -distance of the object p , N_k (p) represents the k -distance neighborhood of the object p , and N (p) k is the size of N_k (p).

Step3: Constructing the SDS of each object in D. Let X and Y are two nonempty datasets and X ∩Y = φ . The absolute difference value between the density of X and that of Y , Δdensity(X,Y) ,equals the minimum of the absolute difference values between the density of any object in X and that of any object in Y . For any given x € X , we say that x is the density similarity neighbor of Y in X if there is a y € Y such that Δdensity (X,Y) equals the absolute difference value between the density of x and that of y .

Step4: Based on the SDS, calculating the ASC of each object in D. ASC can be viewed as the cost description of SDS and the earlier objects in SDS contribute more in the ASC. Let p be an object in D, and the average chain cost of p , ASC(p) , is defined as

$$ASC(p) = \sum_{i=1}^{r} \frac{dist(ao_i)}{i}$$

where r is the size of N_k(p) and dist(ao_i) is the distance between the two adjacent object in SDS(p) .

Step5: Computing the DSNOF value of each object in D and output. For any object p in D, the DSNOF value of p , DSNOF(p) , is defined as

$$DSNOF(p) = \frac{|N_k(p)| \cdot ASC(P)}{\sum_{o \in N_k(p)} ASC(o)}$$

The DSNOF value of an object is the ratio of the ASC of the object and the average of the ASC of k–distance neighbors of the object to their own k–distance neighbors. It indicates the degree of the object being an outlier.

2.4 Frequency based Outlier Detection*:*

[10]Statistical, distance-based and density-based approaches work well only for numerical data. When we have data with categorical attributes it is assumed that the categorical attributes could be easily mapped into numerical values. However, there are cases of categorical attributes, where mapping to numerical attributes is not a straightforward process, and the results greatly depend on the mapping that is used, e.g., the mapping of a marital status attribute (married or single) or a person’s profession (engineer, financial analyst, etc.) to a numerical attribute. Frequency-based approaches have been defined to detect outliers in categorical data.

3. ANALYSIS AND RESULTS

TABLE 1 DIFFERENT ALGORITHMS COMPLEXITY

ALGORITHM	COMPLEXITY
DSNOF	(n^2d+ndk^2)
LOF	O(n log n)
LOF’	O(n log n)
LOF’’	O(n log n)
K NEIGHBOUR	O(n*d*K)
CONTROL CHART	O(n*d)

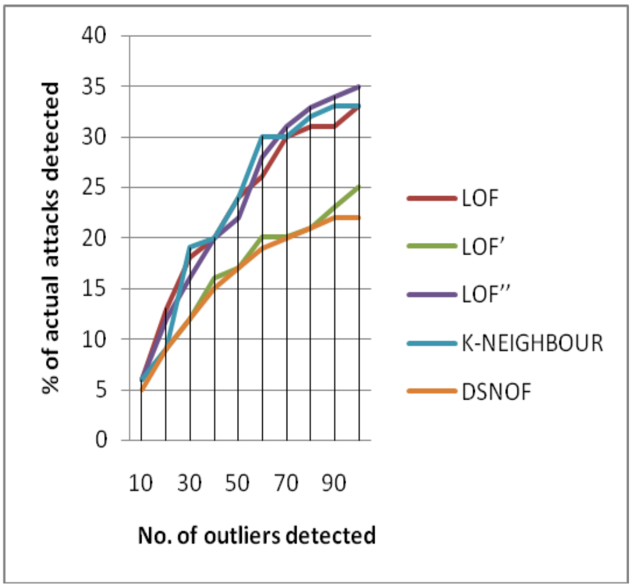
So, with the help of this table , we can compare the complexities of different algorithms. Now we are mentioning the reliability of different outlier detection techniques.

TABLE2. DIFFERENT ALGORITHMS RELIABILITY

OUTLIERS	LOF	LOF’	LOF’’	K NEIG	DSNOF
10	6	6	6	6	5
20	13	9	12	9	9
30	18	12	16	19	12
40	20	16	20	20	15
50	24	17	22	24	17
60	26	20	28	30	19
70	30	20	31	20	20
80	31	21	33	32	21
90	31	23	33	34	22
100	33	25	35	33	22

As it is mentioned earlier, this table is intended to come up with the extent to which a particular algorithm finds the outlier correctly. First column of the column of the shows the total number of outliers which are detected and rest of the columns shows how much outliers are detected correctly. This observation can be shown graphically as:

GRAPH1. RELIABILITY COMPARISON OF ALGORITHMS



4. CONCLUSION

Summarization and conclusion of this study with listing some important issues for outlier detection algorithms as follows:

There is no outlier detection algorithm that can be universally used to solve all problems. Usually, algorithms are designed with certain assumptions and different algorithms are used in different scenarios such as there are different algorithms which are intended to deal with different sort of datasets. Verification of reported outliers is as important as the outlier detection algorithms. Additionally, the efficiency and effectiveness of a novel outlier detection algorithm can be defined as to handle large volume of data as well as high-dimensional features with acceptable time and storage, to detect outliers in different density regions, to show good data visualization and provide users with results that can simplify further analysis.

5. REFERENCES

[1] Aggarwal, C. C., Yu, S. P., "*An effective and efficient algorithm for high-dimensional outlier detection*", The VLDB Journal, 2005, vol. 14, pp. 211–221.

[2] Breunig, M.M., Kriegel, H.P., and Ng, R.T., "LOF: *Identifying densitybased local outliers.", ACM Conference Proceedings, 2000*, pp. 93-104.

[3] Zuriana A. B., Rosmayati M., Akbar A., Mustafa M. D., "*A Comparative Study for Outlier Detection Techniques in Data Mining" CIS 2006.*

[4] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. In VLDB '98: Proceedings of the 24rd International Conference on Very Large Data Bases, pages 392–403, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[5] Hui Cao , Gangquan Si, Wenzhi Zhu,Yanbin Zhang-" *Enhanceing Effectveness of Density based Outlier Mining".*

[6] Markus M.Breunig,Hans-peter Kriege, Raymond T.Ng,Jorg Sander –*" LOF: Identifying Density-Based Local Outlier".*

[7] Ester, M., Kriegel, H.-P., Sander, J., and Xu X. (1996), *A density-based algorithm for discovering clusters in large spatial data sets with noise*. Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining. Portland, OR, pp.226-231.

[8] Hinneburg, C. C. Aggarwal, and D. A. Keim. "What is the Nearest Neighbor in High Dimensional Spaces". In *Proc. 26th Int. Conf. on Very Large Databases (VLDB'00), Cairo, Egypt*, 2000.

[9] C. B. D. Newman and C. Merz. UCI repository of machine learning databases.

[10] Han and Kamber(2007), *Data Mining: Concepts and Techniques* Morgan Kaufmann publications

[11] George Marakas, *Data Warehousing, Data Mining and Visualisation,* Pearson publications