A Comprehensive Survey on Online Anomaly Detection

Amol M. Pawar
M. Tech Student,
CSE Department,
SGGS IE & T, Nanded-431606

Manisha S. Mahindrakar
Assistant Professor,
CSE Department,
SGGS IE & T, Nanded-431606

ABSTRACT
Anomaly is a data point that does not follow to the normal points describing the data. Anomaly detection has vital applications in data cleaning and moreover in the mining of anomalous points for fraud recognition, intrusion detection, stock market analysis, customized marketing, system sensors, and email spam detection. Discovering anomalous points among the data points is the fundamental thought to find out an anomaly. Anomaly detection signals out the objects mostly deviating from a particular data set. The majorities of the anomaly detection techniques are normally implemented in batch mode, and so cannot be essentially reached out to large-scale problems without giving up the reckoning and memory requirements. To address this problem, existing online anomaly detection technique with oversampling and Principal Component Analysis (PCA), aim at identifying presence of anomalies from a big amount of data via an online updating technique. The oversampling strategy is used to duplicate the target instance and balance the data set and the PCA is used as a renowned unsupervised dimension reduction method, which determines the principal directions of the data distributions. Inspite of the duplication, the computation and memory requirement is reduced using an online updation technique. Thus present framework is ideal for online applications which have reckoning or memory restrictions.

General Terms
Data Mining, Outlier Detection.

Keywords
Anomaly detection, online updation technique, oversampling, principal component analysis, unsupervised dimension reduction.

1. INTRODUCTION
The branch of data mining concerned with discovering extraordinary presences in datasets is called as anomaly detection. The objective of anomaly detection is to recognize group of instances that are infrequent within data. Anomaly detection assumes a critical part in discovering fraud, network interruption, and other unusual actions that may have great importance but are hard to discover. A most likely definition of anomaly is given by Hawkins as [1] "an observation which deviates so much from other observations as to arouse uncertainties that it was produced by an alternate mechanism". Fig 1 outlines anomalies in a XY plane.

In a broad sense, there are three types of anomalies. These are Point anomalies, Contextual anomalies and Collective anomalies. An individual data instance is inconsistent with respect to the remaining set of data, then the instance is labeled as a point anomaly. In Fig 1, points a1 and a2 and in addition points in region a3 lie outside the boundary of the normal regions, and hence are point anomalies. A data instance is unconventional in a particular circumstances is named as contextual anomalies. The contextual attributes are utilized to define the context for that instance. Collective anomalies require a association among data instances. On the off chance that if a group of related data instances is inconsistent with respect to the whole dataset, it is termed as a collective anomaly. In a collective anomaly, the discrete data instance may not be anomalous by them, but their existence together as a group is anomalous.

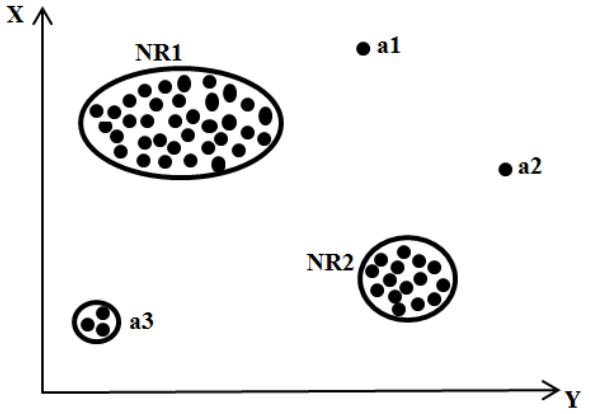

Fig 1: anomalies in a XY plane

Fig. 1 shows the data has two normal regions, NR1 and NR2, since number of the observations lie in these two regions. Points that are sufficiently far away from the regions, e.g., point's a1 and a2, and points in region a3, are anomalies. Anomalies are not always bad or indicative of a failure. In any case, decisively recognizing anomalies can be very difficult. Successful anomaly detection requires a framework to learn consistently.

Basically, network data suffers from all types of anomalies mentioned above. At present, Principal Component Analysis (PCA) has arisen as a promising strategy for identifying an extensive range of network anomalies. PCA is a dimension reduction technique that returns a compact representation of a multidimensional dataset, by projecting the data onto a lower dimensional subspace. PCA represents normal and anomalous traffic behavior directly from the experimental data.

2. RELATED WORK
Typically, the current approaches of anomaly detection can be categorized into following categories. These are:

1. Statistical Distribution Based Methods
2. Distance Based Methods
3. Density Based Methods
4. Deviation Based Methods

2.1 Statistical Distribution Based Methods
Statistical methods observe the user or system behavior by evaluating certain variables over time. By observing the

behavior of different users or datasets, it forms a model of standard behavior and it follows that probability model. This approach purposes to discover the anomalies which differ from such model.

Statistical Distribution based methods are generally built on key assumption that is [2]: "Normal data instances occur in high likelihood regions of a model, while anomalies occur in the low likelihood regions of the model." Basically, there are two forms of statistical methods. These are Parametric and Nonparametric. Both parametric as well as non-parametric techniques have been applied to fit a statistical model. Parametric techniques assume that the normal data is generated by a parametric distribution with parameters Θ and probability density function f(x, Θ), where x is an observation. The anomaly score of an observation x is the inverse of the probability density function, f(x, Θ).The parameters Θ are estimated from the given data. Alternatively, a statistical hypothesis test is used. The null hypothesis (Ho) for such tests is that the data instance x has been generated using the estimated parameter Θ. If the statistical test rejects Ho, x is confirmed to be anomaly.

For Non-parametric statistical techniques the structure of model is not defined, but it instead determined from given data. Such techniques typically make fewer norms regarding the data, such as smoothness of density, when compared to parametric techniques. The simplest non-parametric statistical technique is to use histograms to keep up a profile of the normal data [2]. A basic histogram based anomaly detection technique for univariate data involves two steps. The first step involves building a histogram based on the different values taken by that feature in the training data. In the second step, the technique checks if a test instance falls in any one of the bins of the histogram. If it does, the test instance is normal, otherwise it is referred as anomalous. The key disadvantage of statistical methods is that they work on a univariate data. Moreover, the statistical approach obliges information about the parameters of the dataset, such as the data dispersal. However, in a several circumstances, the data dispersal may not be known. Statistical methods do not promise that all anomalies will be found for the cases where no specific test was developed.

2.2 Distance Based Methods

In Distance based methods, the distance between each data point and its neighbors are calculated. If the obtained result is over some predefined edge, the target instance will be considered as an anomaly. Distance based anomalies are those instances in the data set for which there are not as much as k points within the distance δ. Several distance based anomaly detection algorithms have been recently proposed for detecting anomalies in network traffic. These algorithms are based on calculating the full dimensional distances of points from one another using all the available features, and on calculating the densities of local neighborhoods. These algorithms are: Index based, Nested loop based and Cell based.

In Index based algorithms, Let N be the number of objects in dataset T, and let F be the underlying distance function that gives the distance between any pair of objects in T. For an object O, the D-neighborhood of O contains the set of objects $Q \epsilon T$ that are within distance D of O that is:

$$(\{Q \epsilon T \mid F(O, Q) \leq D\})$$

The fraction p is the minimum fraction of objects in T that must be outside the D-neighborhood of an outlier. For simplicity of discussion, let M be the maximum number of objects within the D-neighborhood of an outlier, that is,

$$M = N(1-P)$$

More specifically, based on a standard multidimensional indexing structure, we execute a range search with radius D for each object O. As soon as (M+1) neighbors are found in the D-neighborhood, the search stops, and O is declared a non-outlier; otherwise, O is an outlier.

Nested loop based algorithm uses block-oriented, nested loop design [10]. It splits the memory buffer space into two halves, called the first and second arrays. It reads the data set into the arrays, and directly computes the distance between each pair of instances in the dataset. For each object 't' in the first array, a count of its D-neighbors is maintained. Counting stops for a particular tuple whenever the number of D neighbors exceeds M. Following fig. 2 shows the pseudo code for Nested-Loop algorithm [7].

Nested Loop Algorithm

1. Fill the first array of size B⁄2% of the dataset with a block of tuples from T.
2. For each tuple t_i in the first array, do:
   a. $count_i \leftarrow 0$
   b. For each tuple t_j in the first array, if $dist(t_i,t_j) \leq$ D: Increment $count_i$ by 1. If $count_i > M$, mark t_i as a non-outlier and proceed to next t_i.
3. While blocks remain to be compared to the first array, do:
   a. Fill the second array with another block
   b. For each unmarked tuple t_i in the first array do: For each tuple t_j in the second array, if $dist(t_i,t_j) \leq$ D: Increment $count_i$ by 1. If $count_i > M$, mark t_i; as a non-outlier and proceed to next t_i.
4. For each unmarked tuple t_i in the first array, report t_i as an outlier.
5. If the second array has served as the first array any time before, stop; otherwise, swap the names of the first and second arrays and go to step 2.

Fig 2: Pseudo code for nested loop algorithm

To avoid computational complexity, a cell-based method was proposed for memory-resident data sets [10]. In this method, the dataset is separated into cells with length equal to

$$l = D/2\sqrt{k}$$

Where, k is the dimensionality, D is the distance.

Every cell has two layers surrounding it. The primary layer is one cell thick that is intermediate neighbor, while in the second layer the cells within 3 cell of a definite cell. The algorithm checks anomalies on a cell-by-cell rather than an object by object basis. For a given cell, it accumulates three counts, the number of objects in the cell, in the cell and the first layer together, and in the cell and both layers together. An instance, in the current cell is considered an outlier only if the number of instances in the cell and the first layer together count is less than or equivalent to it. If this condition does not hold, then all of the instances in the cell can be removed as they cannot be anomalies.

2.3 Density Based Methods

In Density based methods, the anomalies are defined by comparing the density around a point with the density around its local neighbors. Density based methods based on the key assumption that: "Density around a normal point is similar to the density around its neighbors and in case of outliers density varies when compared to its neighbors" [10]. One of the representatives of this type of approach is to use a density-based local outlier factor (LOF) to measure the outlierness of each data instance. An object is a local outlier if it is outlying relative to its local neighborhood, particularly with respect to

the density of the neighborhood. It assesses the degree to which an object is an outlier [10]. This degree of "outlierness" is computed as the local outlier factor (LOF) of an object. It is local in the sense that the degree depends on how isolated the object is with respect to the adjacent locality. In the wake of processing it is characterizes that a point with LOF ≈ 1, is a point in a region with homogeneous density around the point and its neighbor and a point with LOF >> 1 is an outlier. This approach can recognize both global and local outliers. Fig. 3 illustrates the example of density based local and global outliers [2].

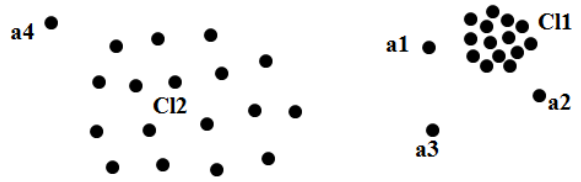

Fig 3: Density based local and global anomalies

For instance, consider a 2 dimensional data set shown in Fig. 3. Due to the low density of the cluster Cl2 it is obvious that for every instance q inside the cluster Cl2, the distance between the instance q and its nearest neighbor is greater than the distance between the instance a4 and the nearest neighbor from the cluster Cl2, so the instance a4 will not be considered as anomaly. Hence, the basic technique will fail to distinguish between a4 and instances in Cl2. However, the instances a1, a2 and a3 are detected as local and global anomalies to the cluster Cl1.

2.4 Deviation Based Methods

To distinguish remarkable objects, deviation based methods examine the main characteristics of instances in the data set. Instances that move away from these characteristics are considered as outliers. Hence, in this approach the term deviations are typically used to refer to outliers. Essentially, there are two techniques for deviation-based outlier detection. The first sequentially compares instances in a data set, while the second utilizes an OLAP data cube approach [10].

The sequential exception technique [10] recreates the way in which peoples can differentiate unusual objects from among a series of supposedly like objects. It uses implicit redundancy of the data. From the given data set, D, with n instances, it forms a sequence of subsets, $\{D_1, D_2, \ldots, D_m\}$, of these objects with $2 \leq m \leq n$ such that

$$(Dj - 1) \subset Dj, \text{ where } Dj \subseteq D.$$

An OLAP approach to deviation detection uses data cubes to identify regions of anomalies in large multidimensional data. In OLAP approach, the deviation detection process is overlay with cube computation. The approach is a process of discovery-driven examination, in which precomputed measures showing data exceptions are used to guide the user in data analysis, at all levels of accumulation. A cell value in the cube is considered an exception if it is significantly different from the projected value, considering a quantifiable model. The framework uses visual signs such as background color to reflect the degree of exception of each cell. The user can choose to drill down on cells that are flagged as exceptions. The model considers variations and patterns in the measure value across all of the dimensions to which a cell belongs [10].

For instance, assume that we have a data cube for sales data and are viewing the sales summarized per month. With the help of the visual signs, we notice a growth in sales in December in comparison to all other months. This may seem like an exception in the time dimension. On the other hand, by drilling down on the month of December to disclose the sales per item in that month, you note that there is a similar growth in sales for other items during December. Therefore, a growth in total sales in December is not an exception if the item dimension is considered. The model considers exceptions are cover up at all aggregated group by's of a data cube. Manual detection of such exceptions is difficult.

2.5 Angle Based Anomaly Detection

Angle based outlier detection method is very unique. It calculates the variation of the angles between each target instance and the remaining data points. It is observed that an outlier will produce a smaller angle variance than the normal ones do. This idea makes less severe the impact of the "curse of dimensionality" on mining high-dimensional data where distance-based approaches often fail to offer high quality results.

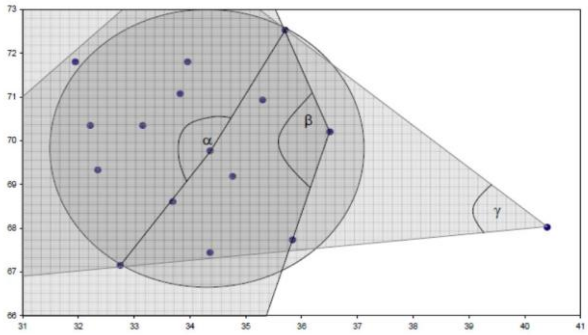

Fig. 4: Illustration of Angle Based Outlier Detection [5]

Consider a simple data set as illustrated in Fig. 2.3 [5]. For a point inside a cluster, the angles between difference vectors to pairs of other points differ widely. The variance of the angles will become smaller for points at the border of a cluster. On the other hand, even here the variance is still moderately high contrasted with the difference of angles for genuine anomalies. Here, the angles to most pairs of points will be small since most points are clustered in some directions. The spectrum of angles to pairs of points remains rather small for an outlier whereas the variance of angles is higher for border points of a cluster and very high for inner points of a cluster [5].

Due to vast number of instance pairs to be considered, the major concern of ABOD is the computation complexity. Therefore, a fast ABOD algorithm is proposed to produce an estimate of the original ABOD solution. The difference between the normal and the fast ABOD approaches is that the later one considers the variance of the angles between the object instance and its k nearby neighbors. However, the search of the nearby neighbors still excludes its extension to large scale problems, since the user will need to have all data instances to calculate the required angle information [3].

2.6 Online Anomaly Detection using Incremental Commute Time

Commute Time Distance (CTD) is an arbitrary walk based metric on graphs [4]. The CTD (i, j) between two nodes i and j is the probable number of steps an arbitrary walk beginning

at i will take to reach j for the first time and then return back to i. At the same time CTD can be used to distinguish global, local and even collective anomalies in data. When a new data point arrives, the application needs to react rapidly without recomputing everything from scratch. The algorithms noted most importantly work in a batch mode and have a high calculation cost for online applications. Here, they make utilization of the features of arbitrary walk to estimate CTD incrementally in constant time. The same approach could be used for estimating the hitting time incrementally. They suggest a provably fast method to incrementally update the eigenvalues and eigenvectors of the graph Laplacian matrix. This method is combined with any technique requiring a graph spectral computation, such as spectral clustering.
Consider Graph G, Laplacian matrix L, its eigenvalues S and eigenvectors V, the anomaly threshold of the training set, and a test data point p to determine if p is anomaly or not. The steps are [4]:

1. Add p to G using the mutual nearest neighbor graph, we have a new graph G_n.
2. Determine, if p is an anomaly or not by estimating its anomaly score using incremental CTDs. Use pruning rule with threshold to reduce the computation.
3. Return whether p is an anomaly or not.

Table 1. Comparative analysis of existing anomaly detection methods

Method Type	Sub Types	Description
Statistical Distribution based methods	Parametric Technique	Assume that the normal data is generated by a parametric distribution with parameters Θ and probability density function f(x, Θ).
	Non Parametric Technique	Typically, make fewer norms regarding the data. Use histograms to keep up a profile of the normal data
Distance based methods	Index Based	Based on range search with radius D, the object which falls outside the neighborhood are anomalous.
	Nested Loop Based	Reads the dataset into the arrays, and directly computes the distance between each pair of instances in the dataset.
	Cell Based	Checks anomalies on a cell-by-cell. If the number of instances in the cell and the first layer together count is less than or equivalent to it, considered as anomaly.
Density based methods	Local Outlier Factor	Object is a local outlier if it is outlying relative to its local neighborhood.
Deviation based methods	Sequential Approach	Sequentially compares instances in a dataset.
	OLAP data cube approach	Uses data cubes to identify regions of anomalies.
Angle based Outlier Detection		Calculates the variation of the angles between each target instance and the remaining data points.
Fast Angle based Outlier Detection		Only considers the variance of the angles between the object instance and its k nearby neighbors.
Online Anomaly Detection using Incremental Commute Time		Arbitrary walk based metric on graphs. Make utilization of the features of arbitrary walk to estimate CTD incrementally in constant time.

3. ONLINE ANOMALY DETECTION

All the existing approaches mentioned above are normally employed in batch mode. They are also not applicable for high dimensional data. Subsequently they can't be effectively reached out to anomaly detection problems with streaming data or online settings. Online anomaly detection has the benefit that it can permit experts to carry out corrective actions as soon as the anomaly is occurred in the sequence data. In this approach anomalies are detected using Principal Component Analysis (PCA) with oversampling strategy. PCA is a way of recognizing patterns in data and expressing data in such a way as to focus their likenesses and variances. PCA is an unsupervised dimension reduction method. PCA is a statistical method that uses an orthogonal transformation to change over a set of observations of probably correlated variables into a set of values of linearly uncorrelated variables called principal components. The number of principal components is less than or equal to the number of original variables. The method is typically used as a tool in experimental data exploration and for constructing analytical models. PCA can be prepared by eigenvalue decomposition of a data covariance matrix or singular value decomposition of a data matrix, generally after mean centering the data matrix for every attribute. The effects of a PCA are generally talk over in terms of component scores, sometimes called factor scores, and loadings. The principal directions for the given dataset are obtained by building the data covariance matrix and determine its dominant eigenvectors. These eigenvectors are the principal directions. The eigenvectors are most explanatory that holds most of the information about the entire dataset by which the rebuilding error gets decreased [8].
For practical anomaly detection issues, the size of the data set is ordinarily substantial. Consequently, it might not be easy to observe the variation of principal directions caused by the presence of a single outlier because it didn't create much deviation in the principal directions. So the concept of oversampling is used. The instance that is added is oversampled means duplicated multiple times. If the added

instance is an outlier the principal direction gets deviated and if it is a normal point there won't be any deviations.
Fig. 5 illustrates the effect of oversampling a normal data point, we observe negligible changes in the principal directions and the mean of the data [3].

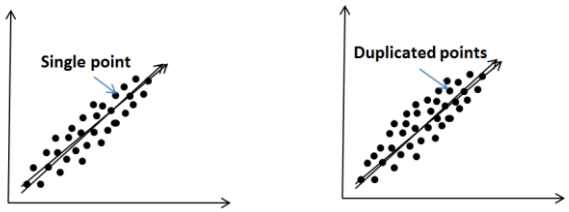

Fig. 5: Over-sampling a normal data point

Fig. 6 illustrates the effect of oversampling an outlier. It shows the deviation in principal direction and the mean of the data clearly [3].

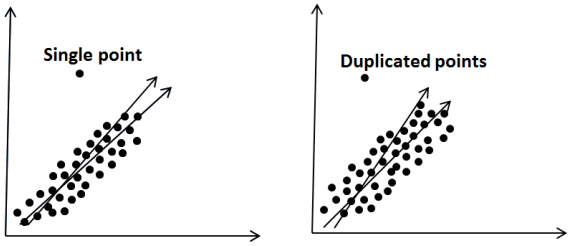

Fig. 6: Over-sampling an outlier

3.1 Framework of Online Anomaly Detection

Fig. 3.3 illustrates the framework of our online anomaly detection. This framework has two phases. Those are [3]:

1. Data Cleaning Phase
2. Online Anomaly Detection Phase

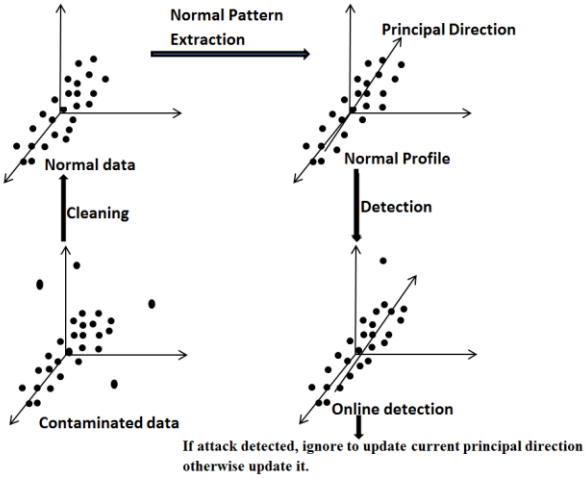

Fig. 7: Framework of online anomaly detection [3].

In the data cleaning phase, our goal is to spot the distrustful data using our oversampling and PCA before performing online anomaly detection. The data cleaning phase can be done offline, the outlier score are defined as "one minus the absolute value of cosine similarity". The instances are rank accordingly. Then the outliers in the given data are filtered according to the ranking. In the online anomaly detection phase, our goal is to recognize the new arriving abnormal instance. The quick updating of the principal directions given in this approach can satisfy the online detecting demand. A new arriving instance will be marked if its suspicious score is higher than the mean plus a specified multiple of the standard deviation. This online anomaly detection approach is well suited for online applications as it overcomes the computation issues and utilizes the memory efficiently [3].

4. CONCLUSION

Here, we study about distinct anomaly detection approaches. However they have restrictions such as that are implemented in batch mode and also not applicable for high dimensional data. Our online anomaly detection technique based on Oversample PCA, duplicates the target instance numerous times to strengthen the effect of target instance. We additionally uses PCA as an unsupervised dimension reduction method to figure out the principal direction of the online updating data. This process does not have to keep the whole covariance or data matrices during the online detection process. This approach is appropriate for the applications with streaming data or online settings. Accordingly, compared with other anomaly detection methods, our methodology is able to accomplish satisfactory results while considerably decreasing computational expenses and memory requirements. In this way, our methodology is ideal for online large-scale or streaming data issues.

5. REFERENCES

[1] D.M. Hawkins, 1980 Identification of Outliers.

[2] V. Chandola, A. Banerjee, and V. Kumar, 2009 Anomaly Detection: A Survey.

[3] Yuh-Jye Lee, Yi-Ren Yeh, and Yu-Chiang Frank Wang, 2013 Anomaly Detection via Online Oversampling Principal Component Analysis.

[4] Nguyen Lu Dang Khoa, Sanjay Chawla, 2011 Online Anomaly Detection Systems Using Incremental Commute Time.

[5] H.-P. Kriegel, M. Schubert, and A. Zimek, 2008 Angle-Based Outlier Detection in High-Dimensional Data.

[6] F. Angiulli, S. Basta, and C. Pizzuti, 2006 Distance-Based Detection and Prediction of Outliers.

[7] Edwin M. Knox and Raymond T. Ng, 1998 Algorithms for Mining Distance-Based Outliers in Large Datasets.

[8] Christian Callegari, Loris Gazzarrini, Stefano Giordano, Michele Pagano, and Teresa Pepe, 2011 A novel PCA-based Network Anomaly Detection.

[9] Numenta White Paper, The Science of Anomaly Detection.

[10] Jiawei Han, Micheline Kamber, Data Mining: Concepts and Techniques.