Efficient Pruning Methods for Distance-Based Outlier Detection

Mohammad Hasnain Ali
Indian School of Mines,
Dhanbad
India
mohd.hasnain86@gmail.com, rajendrapamula@gmail.com

December 28, 2012

Abstract

Outlier detection is concerned with discovering exceptional/anomalous behaviors of objects. In this project our main objective is to how fast we can capture the outliers. In this process we proposed a simple methods to detect outliers. These methods are divided into two steps 1. pruning step and 2. outlier factor. In the pruning step, we prune away some clusters which does not contain any outliers basing on proposed parameters. In outlier factor step we give some score for each unpruned point. Based on the score, points are declared as outliers. We compared these two methods and concluded which is best. We experimented with one real time dataset, synthetic datasets and found out that our methods is able to capture outliers even though we pruned 60-80% of data.

1 Introduction

There is a need for pre-processing of the raw data in many fields, such as data mining, information retrieval, machine learning and pattern recognition. [23] argue for the importance of data preprocessing and present the following reasons: (1) real world data is impure; (2) high performance data mining systems require high quality data and (3) quality data yields high quality patterns. Therefore, developing efficient data-preprocessing techniques is a critical task that requires considerable research efforts.

Data pre-processing involves many tasks including detecting outliers, recovering incomplete data and correcting errors. These tasks often present themselves as less glamorous. However, they are more critical than further steps in many application areas [23]. Outlier detection is an important pre-processing task. It has many practical applications in several areas, such as fraud detection [5], identifying computer network intrusions and bottlenecks [17], criminal activities in E-commerce and detection of suspicious activities [7]. [15] defined outliers as those data points (vectors) with values different from those of the remaining set of data. And we say about the outliers is, an outlier is an observation (or measurement) that is different with respect to the other values contained in a given dataset. Outliers can be due to several causes. Outlier detection is concerned with discovering exceptional behaviors of certain objects. Revealing these behaviors is important since it signifies that something out of the ordinary has happened and shall deserve people’s attention. In many cases, such exceptional behaviors will cause damages to the users and must be stopped. Therefore, in some sense detecting outliers is at least as significant as discovering general patterns. Outlier detection schemes lay a foundation in many applications, for instances, calling card fraud in telecommunications, credit card fraud in banking and finance, computer intrusion in information systems. The problem of detecting abnormal events, called outliers, on the other hand. These non-conforming patterns are often referred to as anamolies, discordant, observations, exceptions, aberrations, surprises, peculiarities, or contaminants in different application domains. This has been widely studied in different research communities as rare classes mining [12], exception mining, outlier detection, etc. researchers have developed several super-vised and unsupervised techniques to mine outliers in static databases and also recently in data streams [19]. In recent decades, many outlier detection approaches have been proposed whereas Unsupervised outlier detection can be fur-
ther classified as distance-based (e.g. KNN), density-based (e.g. LOF), cluster-based (e.g. DBSCAN), depth-based [13] and distribution-based [4]. We can define it as. These non-conforming patterns are often referred to as anomalies, outliers, discordant, observations, exceptions, aberrations, surprises, peculiarities, or contaminants in different application domains. Here, we focus on distance-based outliers which have been popularly defined as:

- An object \( O \) in a dataset \( T \) is a \( DB(p,D) \) outlier if at least fraction \( p \) of the objects in \( T \) lies greater than distance \( D \) from \( O \). [14]
- Given two integers \( kn \) and \( w \), an object \( p \) is said to be an outlier, if less than \( w \) objects have higher value for \( D^{kn} \) than \( p \), where \( D^{kn} \) denotes the distance of the \( kn^{th} \) nearest neighbor of the object \( p \). [21]
- Top \( n \) data points whose total distance to their corresponding \( k \) nearest neighbors are largest [1].
- An observation (or subset of observations) which appears to be inconsistent with the remainder of that set of data [11].

All these definitions indicate, a significant amount of distance computations need to be performed in order to verify whether a data point is an outlier or not.

Outlier detection has wide variety of applications such as fraud detection for credit cards, insurance or health care, intrusion detection for cyber-security, fault detection in safety critical systems, and military surveillance for enemy activities. Detecting outliers or anomalies in data has been studied in the statistics community as early as the 19th century. Over time, a variety of anomaly detection techniques have been developed in several research communities. Many these techniques have been specifically developed in certain application domains.

## 2 Literature Review

### 2.1 Background

Consider a dataset \( DS \) with \( N \) data points in \( \text{dim} \) dimensions. While most of these data points are normal, some are abnormal (outlier), and our task is to mine these outliers. Assume a metric distance function \( D \) exists, using which we can measure the dissimilarity in \( \text{dim} \) space between two arbitrary data points. A general approach that has been used by most of the existing outlier detection methods [15] is to assign an outlier score (based on the distance function) to each individual data point, and then design the detection process based on this score. The use of the outlier score is analogous to the mapping of the multidimensional dataset to \( R \) space (the set of real numbers). In other words, we can define the outlier score function \( (F_{out}) \) which maps each data point in \( DS \) to a unique value in \( R \).

Among existing approaches to outlier detection problem, we can classify \( F_{out} \) into global and local score functions. An outlier score function is called global when the the value it assigns to a data point \( p \in DS \), can be used to compare globally with other data points. More specifically, for two arbitrary data points \( p1 \) and \( p2 \) in \( DS \), \( F_{out}(p1) \) and \( F_{out}(p2) \) can be compared with each other, and if \( F_{out}(p1) > F_{out}(p2) \), \( p1 \) has a larger possibility than \( p2 \) to be an outlier. The definitions proposed by Angiulli et al.[1], Breuning et al. [6], and Ramaswamy et al.[21] straightforwardly adhere to this category. On the other hand, the definition of Ng and Knorr [15] can be converted to this category by taking the inverse of the number of neighbors within distance \( r \) of each data point. In contrast, a local outlier score function assigns to each data point \( p \), a score that can only be used to compare within some local neighborhood.

Clustering methods like CLARANS [18], DBSCAN [8], BIRCH [24] and CURE [10] may detect outliers. However, since the main objective of a clustering method is to find clusters, they are developed to optimize clustering, and not to optimize outlier detection. The definition of outlier used are subjective to the clusters that are detected by these algorithms. While definitions of distance-based outliers are more objective and independent of how clusters in the input data are identified.

Knorr and Ng [15] were the first to introduce distance-based outlier detection techniques. An object \( p \) in a data set \( DS \) is a \( DB(q,dist) \)-outlier if at least fraction \( q \) of the objects in \( DS \) lie at a greater distance than \( dist \) from \( p \). This definition is well accepted, since it generalizes several statistical outlier test.

Ramaswamy et.al. [21] proposed the extension of the above definition in order to rank the outliers: given two integers \( kn \) and \( w \), an object \( p \) is said to be an outlier, if less than \( w \) objects have higher value for \( D^{kn} \) than \( p \), where \( D^{kn} \) denotes the distance of the \( kn^{th} \) nearest
neighbor of the object p. Subsequently, Angiulli and Pizzuti [2] with the aim of taking into account the whole neighborhood of the objects, proposed to rank them on the basis of the sum of the distances from the kn-nearest neighbors, rather than considering solely the distance to the kn-th nearest neighbor. The above three definitions are closely related.

Breunig et al. [6] proposed a Local Outlier Factor (LOF) for each object in the data set, indicating its degree of outlierness. This is the first concept of an outlier which also quantifies how outlying an object is. The outlier factor is local in the sense that only a restricted neighborhood of each object is taken into account. Since the LOF value of an object is obtained by comparing its density with those in its neighborhood, it has stronger modeling capability than a distance based scheme, which is based only on the density of the object itself. Note that the density based scheme does not explicitly categorize the objects into either outliers or non-outliers (If desired, a user can do so by choosing a threshold value to separate the LOF values of the two classes).

Zhang et al. [22] proposed a local distance-based outlier detection from the data set. The local distance-based outlier factor (LDOF) of an object determine the degree to which the object deviates from its neighborhood. Calculating LDOF for all points in the data set, makes overall complexity \( O(N^2) \).

While existing work on outliers focuses only on the identification aspect, the work in [14] also attempts to provide intentional knowledge, which is basically an explanation of why an identified outlier is exceptional.

We here used a very efficient method for clustering which is known as k-means algorithm, there are various clustering methods making their classification a difficult task.

### 2.2 Related Work

Many algorithms have been proposed to detect outliers. Knorr et al. [16] proposed the Nested-Loop (NL) algorithm to find outliers. In this algorithm, each data point in the data set is compared to various points in the data set to determine its M nearest neighbors. NL has quadratic complexity that makes all pair wise distance computations between the data points. The author suggested the use of spatial indexing structures such as R-trees and X-trees to find the nearest neighbors of each candidate point. It works well for low dimensional data sets. The index structures may lead to poor performance as the dimensionality increases [20][15].

Bay and Schwabacher [3] presented an algorithm. It is based on NL and uses randomization and pruning rule with near linear time performance. The algorithm depends on the data ordering, which, as the authors in the paper state, can lead to a poor performance. The algorithm performs poorly when the data does not contain outliers.

Amol Ghoting et al. [9] presented an RBRP (Recursive Binning and Re-Projection) algorithm for fast mining of distance-based outliers. The algorithm facilitated fast convergence to a points approximate nearest neighbors. Only a points approximate nearest neighbors (and not its nearest neighbors) are needed for efficient distance-based outlier detection. The algorithm scales well to high dimensional data sets with millions of data points, and outperforms the state-of-the-art distance-based outlier detection algorithms, often by over an order of magnitude. The algorithm scaled log-linearly as a function of the number of data points, and linearly as a function of the number of dimensions. In k-means algorithm partitions are generated by clustering the data. Each of the clusters can constitute a partition. However, this process requires specification of the number of clusters, and does not guarantee equal-frequency partitioning.

### 3 Proposed Outlier Detection Methods

Our methods operates in two phases and uses two pruning techniques. We compared the performance of these two methods. We named the two methods as (1) Average value pruning scheme and (2) Threshold value pruning scheme. In these methods we prune the clusters that does not contain outliers, assuming normal data instances only fall into the pruned clusters. In our first method, average value pruning scheme, we cluster the dataset using well known k-means algorithm, and calculate the average value based on the points present in each cluster. Using this average value we prune away some clusters assuming these clusters does not contain outliers. Using \( F_{out} \) we estimate outlier score of each point in the unpruned clusters. Declare the top-n points as outliers. We used average value and threshold value as described in the equations 1 and 2 below.

\[
\text{averagevalue} = \frac{|Y|}{M}
\]
thresholdvalue = \sum_{i}^{M} \frac{SumClust(i)}{M} \tag{2}

Where
SumClust = distance from centroid to all its points in the cluster
M = Number of clusters
Y = Data Set
The algorithm for average value pruning scheme is given in algorithm 3.1

Algorithm 3.1 Average value pruning scheme

\[ Y = k\text{Means}(DS, it, M) \]
\[ Avgval = \frac{|DS|}{M} \]
for each cluster \( y \in Y \) do
if \(|y| > Avgval \) then
prune the cluster
else
add \( y \) to UP
end if
end for
UP contains the candidate points for outlier detection
for each point \( q \in UP \) do
compute \( F_{out}(q) \)
end for
sort \( F_{out}(q) \)
declare top \( n \) points are outliers.

Using second method, threshold value pruning scheme, we used the same clustering algorithm to partition the clusters into \( m \) number of clusters. Using one parameter, which is defined as sum of all the distances from the centroid to all the points inside each cluster, named as SumClust. For each cluster we calculate SumClust. We use another parameter threshold value which is defined as average of all the SumClust. The SumClust of each cluster is compared with the threshold value. If the SumClust of a cluster is less than the threshold value, prune the cluster as it does not contain any outliers. For each unpruned cluster, using \( F_{out} \) we calculate outlier score of each point. Declare the top-\( n \) points as outliers. The algorithm for threshold value pruning scheme is given in algorithm 3.2.

Algorithm 3.2 Threshold value pruning scheme

\[ [Y, SumClust] = k\text{Means}(DS, it, M) \]
for each cluster \( y \in Y \) do
\( SumClust(y) \)
end for
thresholdvalue = \frac{\sum_{i}^{M} SumClust(i)}{M} \]
for \( i = 1 \rightarrow M \) do
if \( SumClust(i) < \text{thresholdvalue} \) then
prune the cluster
else
add \( y \) to UP
end if
end for
UP contains the candidate points for outlier detection
for each point \( q \in UP \) do
compute \( F_{out}(q) \)
end for
sort \( F_{out}(q) \)
declare top \( n \) points are outliers.

5 Results

In this paper, this shows the comparison between the two methods as we have proposed the two pruning schemes, on the basis of these schemes we could prune out about 60-80% data points. Even though we pruned out 60-80% points from the data set performance of our methods is far better than existing methods. We found that second method performance better than the first method. Finally the results that we presented is justified and there is very good improvement over the existing methods.

6 Future Implementation and Conclusion

We found that distance-based outlier detection methods have high computational complexity. So, In this project, we proposed two methods which are based on a simple idea. We also experimented our proposed methods with nuclear feature extraction for breast tumor diagnosis. The dataset contains 569 medical diagnosis records (objects), each with 32 attributes (ID, diagnosis, 30 real-valued input features). The diagnosis is binary: Benign and Malignant.
real world wdbc dataset and found that it is giving good results. In future, we will try to analyze our method and study for the new improvements.

References


[18] Raymond T. Ng and Jiawei Han. Efficient and effective clustering methods for spatial data mining. pages 144–155, 1994.


[22] Ke Zhang, Marcus Hutter, and Huidong Jin. A new local distance-based outlier detection approach for scattered real-world data. In PAKDD ‘09: Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and