An Efficient Data Cleaning Algorithm through Minimum Spanning Tree for Data Mining

S. John Peter¹ and S. Chidambaramanathan²

¹Department of Computer Science and Research Center
St. Xavier’s College, Palayamkottai, Tamil Nadu, India.
E-mail: jaypeeyes@rediffmail.com
²Department of MCA
St. Xavier’s College, Palayamkottai, Tamil Nadu, India.
E-mail: scharan2009@rediffmail.com

Abstract

Detecting outliers in database (as unusual objects) using Minimum Spanning Tree is a big desire. It is an important task in wide variety of application areas. In this paper we propose Minimum Spanning Tree based algorithm for detecting outliers. The outliers are detected in the data set based on weight function value (MSTWF). If the noticeable changes occurred in the weight function value, the point (object) associated with the edge is detected as an outlier based on degree number of points.

Keywords: Euclidean minimum spanning tree, Degree number, Outliers, weight function.

Introduction

An outlier is an observation of data that deviates from other observations so much that it arouses suspicious that was generated by a different mechanism from the most part of data [1]. Outlier may be erroneous or real in the following sense. Real outliers are observations whose actual values are very different than those observed for rest of the data and violate plausible relationship among variables. Erroneous outliers are observations that are distorted due to misreporting errors in the data collection process. Outliers of either type may exert undue influence on the result of data analysis. So they should be identified using reliable detection methods prior to performing data analysis [1].

Outliers can often be individual or groups of clients exhibiting behavior outside the range of what is considered normal. Outliers can be removed or considered separately in regression modeling to improve accuracy which can be considered as
benefit of outliers. Identifying them prior to modeling and analysis is important [2]. The regression modeling consists in finding a dependence of one random variable or a group of variables on another variables or a group of variables. All most all studies that consider outlier identification as their primary objective are in statistics. The test depends on distribution; whether or not the distribution parameters are known; the number of expected outliers; the type of expected outliers [3].

The importance of outlier detection is due to the fact that outliers in the data translate to significant (and often critical) information in a wide variety of application domains. For example, an anomalous traffic pattern in a computer network could mean that a hacked computer is sending out sensitive data to an unauthorized destination. In public health data, outlier detection techniques are widely used to detect anomalous pattern in patient medical records which could be symptoms of new diseases. Similarly, outliers in credit card transaction data could indicate credit card theft or misuse. Outliers can also translate to critical entities such as in military surveillance, whereas the presence of unusual region in a satellite image of enemy are could indicate enemy troop movement. Or anomalous readings from space craft would signify a fault in some of the craft. Outlier detection has been found to be directly applicable in large number of domains.

Many data-mining algorithms find outliers as a side-product of clustering algorithms. However these techniques define outlier as points, which do not lie in clusters. Thus, the techniques implicitly define outliers as the background noise in which the clusters are embedded. Another class of techniques defines outlier as points, which are neither a part of a cluster nor part of background noise; rather they are specifically points which behave very differently from the norm [4].

The main concern of our outlier detection algorithms is to detect outliers, which are often regarded as noise that should be removed in order to make more reliable clustering [5]. Some noisy points may be far away from the data points, whereas the others may be close. The far away noisy points would affect the result more significantly because they are more different from the data points. It is desirable to identify and remove the outliers, which are far away from all the other points in cluster [6]. So, to improve the clustering such algorithm use the same process and functionality to solve both clustering and outlier discovery [1].

Given a connected, undirected graph \( G = (V, E) \), where \( V \) is the set of nodes, \( E \) is the set of edges between pairs of nodes, and a weight \( w(u, v) \) specifying weight of the edge \( (u, v) \) for each edge \( (u, v) \in E \). A spanning tree is an acyclic subgraph of a graph \( G \), which contains all vertices from \( G \). The Minimum Spanning Tree (MST) of a weighted graph is minimum weight spanning tree of that graph. Several well established MST algorithms exist to solve minimum spanning tree problem [7, 8, 9]. The cost of constructing a minimum spanning tree is \( O(m \log n) \), where \( m \) is the number of edges in the graph and \( n \) is the number of vertices. More efficient algorithm for constructing MSTs have also been extensively researched [10, 11, 12]. These algorithms promise close to linear time complexity under different assumptions. A Euclidean minimum spanning tree (EMST) is a spanning tree of a set of \( n \) points in a metric space \( (E^n) \), where the length of an edge is the Euclidean distance between a pair of points in the point set.
Our Minimum Spanning Tree based Outlier Detection (MSTOD) algorithm is
does not require predefined input parameters. The algorithm constructs an EMST of a
point set. In section 2 we review some of the existing works on outlier detection. In
Section 3 we propose DCMST algorithm which detect outliers. Finally in conclusion
we summarize the strength of our methods and possible improvements.

Related work
There is no single universally applicable or generic outlier detection approach [13,
14]. Therefore there are many approaches have been proposed to deduct outliers.
These approaches are classified into four major categories as distribution-based,
distance-based, density-based and clustering-based [15].

Distribution-based approaches [16, 17] develop statistical models from the given
data then apply a statistical test to determine if an object belongs to this model or not.
Objects that have low probability to belong to the statistical model are declared as
outliers. However, Distribution-based approaches can not be applied in
multidimensional dataset because of the univariate in nature. In addition, prior
knowledge of the data distribution is required. These limitations have restricted the
ability to apply these types of methods to large real-world databases which typically
have many different fields and have no easy way of characterizing the multivariate
distribution.

In the distance-based approach [18, 19, 20, 21] outliers are detected using a given
distance measure on feature space, A point q in a data set is an outlier with respect to
the parameters M and d, if there are less than M points with in the distance d from q,
where the values of M and d are determined by the user. The problem in distance–
based approach is that it is difficult to determine the M and d values. Angiulli [22]
propose a new definition of outliers. In this definition, for each point, P, the sum of
the distances from its k nearest neighbor’s considered. This sum is called the weight
of P, Wk (P) and is used to rank the points of data set. Outliers are those points having
the largest values of Wk. The method proposed by Angiulli [22] needs expected
number of outlier n and application dependent k as input parameter. It is difficult to
predict correct values for n and k. The problem with distance based approach is its
high computational complexity.

Density-based approaches [23, 24] compute the density of the regions in the data
and declare the objects in low dense regions as outliers. Clustering-based approaches
[13, 25, 5, 6], consider clusters of small sizes as outliers. In these approaches, small
clusters (clusters containing significantly less points than other clusters) are
considered as outliers. The advantage of clustering based approaches is that they do
not have to be supervised.

Clustering-based approaches [13, 25, 26, 6] consider clusters as of small sizes as
clustered outliers. In these approaches, small clusters (i.e., clusters containing
significantly less points than other clusters) are considered as outliers.

The advantage of the clustering-based approaches is that they do not have to be
supervised. Moreover, clustering –based technique are capable of being used in an
order to test the absence or presence of outliers, two hypotheses are used. However, the hypotheses do not account for the possibility of multiple clusters of outliers.

Jiang et. al.[6] proposed a two-phase method to detect outliers. In the first phase, clusters are produced using modified K-means algorithm, and then in the second phase, an Outlier-Finding Process (OFP) is proposed. The small clusters are selected and regarded as outliers. Small cluster is defined as a cluster with fewer points than half the average number of points in the \( k \) number of clusters. Loureio [27] proposed a method for detecting outlier. Hierarchical clustering technique is used for detecting outliers. The key idea is to use the size of the resulting clusters as indicators of the presence of outliers. ALmedia [28] is also used similar approach for detecting outliers. Using the K-means clustering algorithm Yoon [29] proposed a method to detect outliers. The K-means algorithm is sensitive to outliers, and hence may not give accurate result.

Moh’d Belal Al-Zoubi [30] proposed a method based on clustering approaches for outlier detection using Partitioning Around Medoid (PAM). PAM attempts to determine \( k \) partition for \( n \) objects. The algorithm uses most centrally located object in a cluster (called medoid) instead of cluster mean. PAM is more robust than the k-means algorithm in the presence of noise and outlier. This PAM based approach suffer from proper cluster Structure. Cluster with irregular boundaries can not be detected using both k-means and PAM algorithms.

In [13] clustering methods have been applied. The key idea is to use the size of the resulting clusters as indicators of the presence of outliers. The authors use a hierarchical clustering technique. A similar approach was reported in [28]. In [13], the authors proposed a clustering-based approach to detect outliers. The K-means clustering algorithm is used. As mentioned in [27], the K-means is sensitive to outliers, and hence may not give accurate results.

**DCMST Algorithm**

Given a point set \( S \) in \( \mathbb{E}^n \), the hierarchical method starts by constructing a Minimum Spanning Tree (MST) from the points in \( S \). The weight of the edge in the tree is Euclidean distance between the two end points.

A Euclidean distance between pair of objects can be represented by a corresponding weighted edge. Our Algorithm is also based on the minimum spanning tree but not limited to two-dimensional points. To detect the outliers from the data set which are represented as MST, we use a new approach based on Minimum Spanning Tree based weight function. For any undirected graph \( G \) the degree of a vertex \( v \), written as \( \text{deg}(v) \), is equal to the number of edges in \( G \) which contains \( v \), that is, which are incident on \( v \).

First we compute the value of Minimum Spanning Tree based weight function (MSTWF) for each cluster. Then we temporarily remove an edge (Euclidean distance between objects) from the cluster and re-calculate the weight function value. If the removal of an edge causes noticeable decrease in the weight function value, then the object (point) connected with the edge is considered as an outlier based on the degree number; otherwise, it is not.
An Efficient Data Cleaning Algorithm

Formally we define the weight function of MST as

\[ \text{MSTWF} = \frac{|E|}{\sum_{j=1}^{\text{W}_j(e)}} \]  \hspace{1cm} (1)

where \(|E|\) be number of edges in the data set (MST), \(W(e)\) is the weight of edges in the MST.

The weight function represents the (Euclidean) sum of distances between the objects (points) in the MST produced by the algorithm. When scanning the cluster (MST), the edges are ordered from smaller to larger lengths. Then we define the threshold as:

\[ \text{THR} = \max (L_i - L_{i-1}) \times t \]  \hspace{1cm} (2)

Where \(L_i\) is largest in the order and \(t \in [0,1]\) is a user defined parameter.

Let MSTWF be the weight function produced by the algorithm for the data set (MST). Let MSTWF\(_j\) be the weight function produced by the algorithm after removing an edge with \(W_j\) from the data set (MST). Subtracting MSTWF\(_j\) from MSTWF gives the difference between the two values expressed as

\[ \text{MSTDWF} = \text{MSTWF} - \text{MSTWF}_j \]  \hspace{1cm} (3)

Algorithm: DCMST( )
Input: S the point set
Output: O the set of outliers

1. \(= \Phi\)
2. Construct an EMST \(T\) from \(S\)
3. Compute MSTWF and THR using equation (1)
4. Sort \(E\), the weighted edge in \(T\) by decreasing order in to \(j\)
5. For each \(j\) in the \(T\)
6. Remove an edge \(j\) from \(T\)
7. Re-calculate the weight function MSTWF\(_j\) using equation (1)
8. Compute MSTDWF\(_j\) using equation (3)
9. If MSTDWF > THR then classify point \(x\) (object) with lower degree number, connected through the edge \(j\) to other point(s) or object(s) as an outlier; \(O = O \cup \{x\} ; \text{go to step 6.}\)
   \[ \text{Else} \]
   \[ \text{Return back the edge} j \text{ to} T \]
10. Return outliers \(O\)

Figure 1 shows a typical example of EMST constructed from point set \(S\). Our algorithm will detect outliers from the given data set. Figure 1 shows the possible distribution of the points in the MST with outlier vertex as 2.
Figure 1: MST from point set S, vertex 2 is an Outlier.

We use the graph of Figure 1 as example to illustrate the DCMST algorithm finds outliers in the data set (EMST). Minimum Spanning Tree from the data set is constructed in line 2. The values of MSTWF and THR are calculated for data set (MST) at line 3. The edges in the MST are sorted in decreasing order (line 4). After removing an edge from the MST the weight function value is re-calculated (line 6-7). The difference in the weight function values MSTDWF is computed in the line 8. Outliers present in the clusters are identified based on the above calculated MSTDWF and THR values (line 9). Lines 6-9 in the algorithm are repeated to detect and remove all the outliers from the data set.

In [31] objective function value for the data set is computed using FCM algorithm. Then the value of objective function is recalculated after removing a point $p_j$ temporarily from data set. Then difference in objective function value is computed. Based on the deviation in the objective function value, outliers are detected and removed from the data set. This procedure is repeated for all the points which are present in the data set. Our approach is based on MST; the number of edges is considerably reduced. In our approach instead of removing all the points from the data set, only few long edges from the MST are removed to compute the weight function value, which is used for detecting outliers. Hence our algorithm is efficient over the approach discussed in [31].

**Conclusion**
Our DCMST algorithm finds outliers without using any predefined input parameter. Our algorithm does not require the users to select and try various parameters combinations in order to get the desired output. Our DCMST detect outliers from data set. Our approach is much efficient than the previous methods. All of these look nice
from theoretical point of view. However from practical point of view, there is still some room for improvement for running time of the algorithm. This could perhaps be accomplished by using some appropriate data structure. In the future we will explore and test our proposed algorithm in various domains. We will further study the rich properties of EMST-based methods in solving different problems for detecting outliers in dynamic dataset.

References


An Efficient Data Cleaning Algorithm

