Cluster based Outlier Detection

Pranjali Kasture
Assistant Professor
Thakur College of Engineering and Technology,
Kandiwali (E),
Mumbai, India

Jayant Gadge
Assistant Professor
Thadomal Shahani College of
Engineering, Bandra (W), Mumbai, India

ABSTRACT
Outlier detection is a fundamental issue in data mining, specifically it has been used to detect and remove anomalous objects from data mining. The proposed approach to detect outlier includes three methods which are clustering, pruning and computing outlier score. For clustering k-means algorithm is used which partition the dataset into given number of clusters. In pruning, based on some distance measure, points which are closed to centroid of each cluster are pruned. For the unpruned points, local distance based outlier factor (LDOF) measure is calculated. A measure called LDOF, tells how much a point is deviating from its neighbors. The high LDOF value of a point indicates that the point is deviating more from its neighbors and probably it may be an outlier.

General Terms
Outlier, cluster, pruning, outlier score, k nearest neighbor

Keywords
KNN, LDOF

1. INTRODUCTION
Finding anomalous points among the data points is the basic idea to find out an outlier. Outlier detection signals out the objects mostly deviating from a given data set. Detecting outliers which are inconsistent with the remaining dataset is a major challenge in real-world KDD applications. Existing outlier detection methods are ineffective on scattered real-world datasets due to implicit data patterns and parameter setting issues. Several measures are used to find out the deviation of a point from other points which tells the outlierness of a point. Since the number of outliers in a data set is very few, it is unnecessary to calculate these measures for all points. The purpose of outlier detection based on clustering [1] is to reduce computation time by removing the points which are probably not outliers.

The outlier detection problem in some cases is similar to the classification problem. For example, the main concern of clustering-based outlier detection algorithms is to find clusters and outliers, which are often regarded as noise that should be removed in order to make more reliable clustering. 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.

The identification of an outlier is affected by various factors, many of which are of interest for practical applications. For example, fraud, or criminal deception, will always be a costly problem for many profit organizations. Data mining can minimize some of these losses by making use of the massive collections of customer data. Using web log files becomes possible to recognize fraudulent behavior, changes in behavior of customers or faults in systems. Outliers arise by reasons of such incidents. Thus typical fault detection can discover exceptions in the amount of money spent, type of items purchased, time and location. Many fraud cases can happen, for example, if someone has your name, credit card number, expiration date and billing address. All this information is very easy to obtain even from your home mailbox or any online transaction that you had before. So, automatic systems for preventing fraudulent use of credit cards detect unusual transactions and may block such transactions on earlier stages.

Another example is a computer security intrusion detection system, which finds outlier patterns as possible intrusion attempts. Intrusion detection corresponds to a suite of techniques that are used to identify attacks against computers and network infrastructures. Outlier detection is a key element of intrusion detection in which perturbations of normal behavior suggest the presence of intentionally or unintentionally induced attacks, faults and defects. Detecting outliers has practical application in more wide spheres: pharmaceutical research, weather prediction, financial applications, marketing and customer segmentation.

This paper is structured as follows. In section 2 an overview of related work of outlier detection is provided. In section 3, proposed work is introduced. Here all three phases of outlier detection algorithm is explained in detail. In section 4, an experimental result of medical diagnosis dataset is shown. Finally, paper is concluded in section 5.

2. RELATED WORK
KeZhang, M.Hutter and H.Jin [2] have proposed a novel Local Distance-based Outlier Factor (LDOF) method to measure the outlier-ness of objects in scattered datasets. LDOF uses the relative location of an object to its neighbors to determine the degree to which the object deviates from its neighborhood. In order to facilitate parameter settings in real-world applications, a top-n technique is employed in outlier detection approach, where only the objects with the highest LDOF values are regarded as outliers. Compared to conventional approaches (such as top-n KNN and top-n LOF), top-n LDOF method is more effective at detecting outliers in scattered data. It is also easier to set parameters, since its performance is relatively stable over a large range of parameter values.

H. Kriegel, P. Kröger, E. Schubert, A. Zimek [3] have proposed a method which give an outlier score or "outlier factor" signaling "how much" the respective data object is an outlier. They propose the novel LoOP (Local Outlier Probability) outlier detection model that combines the idea of local, density-based outlier scoring with a probabilistic,
**3. PROPOSED WORK**

Many data-mining algorithms find outliers as a side-product of clustering algorithms. The main objective of a clustering method is to find clusters. They are developed to optimize clustering, and not to optimize outlier detection. In the proposed work both clustering method and outlier detection techniques are used to find outliers.

The proposed work to detect outlier is divided into three phases: partitioning dataset, radius based pruning and computing outlier score of objects. The flowchart of algorithm is shown in figure 1. This section briefs about the phases of algorithm.

### 3.1 Partitioning Dataset

This is the first stage where the dataset is partition into number of clusters enter by the user. For clustering, k-means algorithm is used. It is one of the simplest unsupervised learning algorithms. The algorithm is given as below:

**K-means Clustering Algorithm:**

1. Initial centroid for clusters is chosen randomly.
2. Calculate the distance between cluster centroid to each object using any distance function (e.g. Euclidean distance) to get the distance matrix.
3. Assign each object based on the minimum distance.
4. Compute new centroid of each group based on new memberships.
5. Repeat step 2, 3, 4 until algorithm has reached its stability.

### 3.2 Radius based Pruning

K-means Clustering method divide the dataset into number of clusters. In the cluster, some points are very close to the centroid of cluster and some points are far away from the centroid of the cluster. The points which are very close to centroid, has to be removed because probability of having outlier is very less for those points.

Pruning step is used to remove the points which are very close to the centroid of the cluster. For this, radius of each cluster is found out [11]. Average distance from centroid to all objects in the cluster is taken as a measure to find the radius of the cluster. For a cluster having centroid m, Let \( C_m \) is the set of objects consisting of k number of objects. The radius \( r \) is calculated using the formula given below.

\[
r = \frac{1}{k} \sum_{n \in C_m} dist(m, n) \quad \text{eq}(1)
\]

After calculating the radius of cluster, points which are within the radius of cluster are found out and removed from the cluster. Since points are very close to the centroid, probability of being outlier is very less for those points. If number of points in clusters is less than the number of outliers then pruning should be avoided for such clusters.

The main idea of pruning is to prune the points in clusters if it is determined that they are not probable candidate for outliers. Since number of outliers is very small, this additional preprocessing step helps to eliminate a significant number of points which are not outliers.
3.3 Computing Outlier Score

In this, for each unpruned point, LDOF (Local Distance-based Outlier Factor) is calculated [2]. A measure called LDOF, tells how much a point is deviating from its neighbors. The high LDOF value of a point indicates that the point is deviating more from its neighbors and probably it may be an outlier.

For computing outlier score of object p, k number of nearest neighbor (knn) points for the object p is found out first. Then average distance from object p to all k nearest neighbors is calculated. Then average distance among the k nearest neighbors is also calculated. The ratio of these values gives outlier score for the object p. Highest score indicates that the object is the most probable candidate of outlier. LDOF factor is calculated as follows:

**dp (KNN distance of p):** Let \( N_p \) be the set of k-nearest neighbors of object p (excluding p). The k-nearest neighbor distance of p equals the average distance from p to all objects in \( N_p \). More formally, let \( \text{dist}(p, q) \geq 0 \) be a distance measure between objects p and q. The k-nearest neighbor distance of object p is defined as:

\[
dp = \frac{1}{k} \sum_{q \in N_p} \text{dist}(p, q) \quad \ldots \ldots \ldots \ldots \ldots \ldots \text{eq}(2)
\]

**Dp (KNN inner distance of p):** Given \( N_p \) of object p, the k-nearest neighbors inner distance of p is defined as the average distance among objects in \( N_p \).

\[
D_p = \frac{1}{k(k-1)} \sum_{q,q' \in N_p,q \neq q'} \text{dist}(q,q') \quad \ldots \ldots \text{eq}(3)
\]

**LDOF of p:** The local distance-based outlier factor of p is defined as:

\[
\text{LDOF}(p) = \frac{dp}{Dp} \quad \ldots \ldots \ldots \ldots \ldots \ldots \text{eq}(4)
\]

LDOF value for all objects is calculated using the formulas given above. All the values are arranged in decreasing order to get the highest values first. Highest value indicates that object is the most probable candidate of being an outlier. Then in Top N points, percentage of outliers detected is evaluated.

The proposed work gives the outlier score of objects in dataset in top n fashion. Number of outliers in the dataset is very few in number. To find outlier score for all objects in dataset is computationally expensive. This can be avoided by doing pruning. Since outlier detection is based on clustering and pruning, complexity of algorithm reduces substantially.

4. EXPERIMENTAL RESULT

In real-world data repositories, it is hard to find a data set for evaluating outlier detection algorithms, because only for very few real-world data sets it is exactly known which objects are really behaving differently. In this experiment, a medical data set of WDBC (Cancer Diagnosis) is used. The data set contains 569 medical diagnosis records (objects), each with 32 attributes (ID, diagnosis, 30 real-valued input features). The diagnosis is binary: Benign and Malignant. The objects, labeled as Benign are considered as a normal data. In the experiment, all 357 Benign diagnosis records are used as a normal data and out of 212 Malignant diagnosis records, only ten records are used and considered as an outlier for the dataset.

The algorithm is run for different number of clusters with knn value varying from 10 to 40. Number of clusters in should not be more than number of outliers. Here, number of clusters varies from 2 to 9. Finally, the percentage of outliers detected is calculated in Top N fashion with N varying from 10 to 25. Table 1 shows the percentage of outlier detected in WDBC dataset for cluster 4 in top 25 with knn value varying from 10 to 40.
It is observed that, as neighborhood size increases, the outlier score for objects also increases. As shown in the Figure 1, when neighborhood size 40 is considered, then in Top 25, all outliers are detected by the algorithm.

Table 2. Pruning percentage for WDBC data set

<table>
<thead>
<tr>
<th>Sr.No.</th>
<th>Number of clusters</th>
<th>% of pruned objects</th>
<th>% of outliers detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>59.95</td>
<td>70</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>55.86</td>
<td>90</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>53.13</td>
<td>100</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>51.77</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>50.95</td>
<td>100</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>52.59</td>
<td>100</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>53.68</td>
<td>100</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>55.31</td>
<td>100</td>
</tr>
</tbody>
</table>

In Table 2, for given number of clusters, percentage of data pruned is calculated. It is observed that, more than 50% of data is removed from the dataset. Thus the computation cost required in computing outlier score for all objects in a dataset is reduced using pruning technique.

Table 3 gives the comparative study of proposed method verses existing method. In existing method, plain LDOF measure was used to compute the outlier score of each object in a dataset. In proposed work, dataset is filtered out using clustering and pruning methods. Computing outlier score for all objects was computationally expensive in the existing method. Whereas, computation cost and complexity, both can be reduced by the proposed method.

Table 3. Comparative study of proposed method verses existing method

<table>
<thead>
<tr>
<th>Points</th>
<th>Existing Method</th>
<th>Proposed Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approach</td>
<td>Plain LDOF measure</td>
<td>Clustering and Pruning based LDOF measure</td>
</tr>
<tr>
<td>Complexity</td>
<td>O(N^2)</td>
<td>Less than O(N^2)</td>
</tr>
<tr>
<td>Computation cost</td>
<td>Computationally expensive</td>
<td>Less expensive</td>
</tr>
<tr>
<td>Data Filtration</td>
<td>Dataset cannot be filtered</td>
<td>Dataset is filtered</td>
</tr>
</tbody>
</table>

5. CONCLUSION

In this work, an efficient outlier detection method based on clustering is proposed. Numbers of outliers in a dataset are generally very few. Computing outlier score for all objects in dataset is computationally expensive. This can be avoided by pruning some of the points in the dataset which are not the probable candidate for outliers. In the proposed work, k means clustering method is used which partition the dataset into number of clusters. Then radius based pruning is done which removes 50% points in the dataset. Finally, to compute the outlier score of unpruned points, a Local Distance-based Outlier Factor measure is used. This measure uses the concept of k nearest neighbors. It is observed that, as the k nearest neighbor value increases, the outlier score of object also increases. Thus it becomes easier to detect outlier at top position and remain stable over large range of neighborhood sizes.

6. REFERENCES


