Outlier detection using neighborhood rank difference

Gautam Bhattacharyya a, Koushik Ghosh b, Ananda S. Chowdhury c,*

a Department of Physics, University Institute of Technology, University of Burdwan, Golabag (North), Burdwan 713104, India
b Department of Mathematics, University Institute of Technology, University of Burdwan, Golabag (North), Burdwan 713104, India
c Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata 700032, India

A R T I C L E   I N F O

Article history:
Received 28 August 2014
Available online 21 April 2015

Keywords:
Outlier
Rank-difference
kNN
RNN

A B S T R A C T

Presence of outliers critically affects many pattern classification tasks. In this paper, we propose a novel dynamic outlier detection method based on neighborhood rank difference. In particular, reverse and the forward nearest neighbor rank difference is employed to capture the variations in densities of a test point with respect to various training points. In the first step of our method, we determine the influence space for a given dataset. A score for outlierness is proposed in the second step using the rank difference as well as the absolute density within this influence space. Experiments on synthetic and some UCI machine learning repository datasets clearly indicate the supremacy of our method over some recently published approaches.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

The problem of outlier detection is of great interest to the pattern recognition community. The major objective of an outlier detection method is to find the rare or exceptional objects with respect to the remaining (large amount of) data [1]. Outlier detection has several practical applications in diverse fields, e.g., in fault detection of machines [2,3], in anomaly detection in hyperspectral images [4], in novelty detection of image sequence analysis [5], in biomedical testing [6,7], in weather prediction [8], in geoscience and remote sensing [9], in medicine [10], in financial fraud detection [11,12], and in information security [13,14]. Different outlier detection methods have been proposed over the years based on the nature of application.

Outlier detection algorithms first create a normal pattern in the data, and then assign an outlier score to a given data point on the basis of its deviation with respect to the normal pattern [15]. Extreme value analysis models, probabilistic models, linear models, proximity-based models [16], information-theoretic models and high dimensional outlier detection models represent some prominent categories of outlier detection techniques. Proximity-based methods treat outliers as points which are isolated from the remaining data and can be further classified into three different sub-categories, namely, cluster-based [17], density-based and nearest neighbor-based [15].

The main difference between the clustering and the density-based methods is that the clustering methods segment the points, whereas the density-based methods segment the space [18]. Local outlier factor (LOF) [19], connectivity-based outlier factor (COF) [20] and influenced outlierness (INFLO) [21] are examples of some well-known density-based approaches for outlier detection. In contrast, rank based detection algorithm (RBDA) [22] and outlier detection using modified-ranks with Distance (ODMRD) [23] are two recently published approaches which use ranks of nearest-neighbors for the detection of the outliers.

In most of the density-based approaches, it is assumed that the density around a normal data object is similar to the density around its neighbors, whereas in case of an outlier the density is considerably low than that of its neighbors. In LOF [19], the densities of the points have been calculated within some local reachable distances and the degree of outlierness of a point has been assigned in terms of relative density of the test point with respect to its neighbors [19]. Tang et al. argued that lower density is not a necessary condition to be an outlier. Accordingly, in COF [20], a set based nearest path was used to select a set of nearest neighbors [20]. This nearest path was further employed to find the relative density of a test point within the average chaining distance. COF [20] is shown to be more effective when a cluster and a neighboring outlier have similar neighborhood densities. Both LOF [19] and COF [20], which use properties of kNN, are found to yield poor results when an outlier lies in between a sparse and a dense cluster. To handle such situations, Jin et al. proposed a new algorithm INFLO based on a symmetric neighborhood relationship. In this method both forward and reverse neighbors of a data point are considered while estimating its density distribution [21]. In case of density-based approaches all the neighbors of a test point are...
assumed to have a similar density. So, if a neighbor is chosen from different clusters with uneven densities the above assumption may introduce some errors in outlier detection. In addition, the notion of density may not work properly for some special types of distributions. For example, if all data points lie on a single straight line, the normal density-based algorithm [22] assumes equal density around the test point and its neighborhoods. This occurs due to the equal closest neighbor distance for both the test-point and its neighbor points. In such situations, rank-based outlier detection schemes like RBDA [22] and ODMRD [23] yield better results as compared to the density-based algorithms. RBDA uses mutual closeness between a test point and its k-neighbors for rank assignment. In ODMRD [23] the ranks were given some weights and the distances between the test point and its neighbors were incorporated. Still, both RBDA and ODMRD are found to be adversely affected by the local irregularities of a dataset like the cluster deficiency effect and the border effect.

In order to address the shortcomings of density-based and rank-based methods, we propose a novel hybrid outlier detection approach using the concepts of density as well as neighborhood rank-difference. The first contribution of our method is: instead of local reachable distance [19], we employ a dataset-specific global limit in terms of \( k \) (number of forward neighbors) to estimate the spatial density. The second contribution of our method is: we can better capture the variations in density by using reverse as well as forward rank difference over rank-based methods [22,23] by minimizing both the cluster deficiency effect and the border effect. Our third contribution is: we can minimize information loss due to averaging [19] through an effective density sampling procedure. Experimental results clearly indicate that we can capture most of the outliers within top-m instances.

The rest of the paper is organized in the following manner: In Section 2, we provide the theoretical foundations. In Section 3, we describe the proposed method and also analyze its time-complexity. In Section 4, we present the experimental results with detailed comparisons. Finally, the paper is concluded in Section 5 with an outline for directions of future research.

2. Theoretical foundations

Let \( D \) denotes the data set of all observations, \( k \) denotes the number of points in the set of \( k \) nearest neighbors \( N_k(p) \) around some point \( p \in D \), and \( d(p, q) \) is the Euclidean distance between any two points \( p, q \in D \). We consider Euclidean distance for its simplicity. Further, let \( R \) represents the reverse ranking of the point \( p \) with respect to the point \( q \in N_k(p) \). In the proposed method, we employ the difference of the Reverse Nearest Neighbor (RNN) and Forward Nearest Neighbor (kNN) ranks of a point. If \( q \) is the \( k \)th neighbor of the point \( p \) at distance \( d_k(p, q) \), then the forward density up to \( k \)th neighbor is given by:

\[
\Omega_k(p) = \frac{k}{d_k(p, q)}
\]  

Similarly, if \( p \) be the \( R \)th neighbor of \( q \) for the same distance \( d_k(p, q) \) then the reverse density around \( q \) at same distance \( d_k(p, q) \) is given by:

\[
\Omega_R(q) = \frac{R}{d_k(p, q)}
\]  

The positive value of the rank difference \( (R-k) \) indicates that \( q \) has a denser surrounding than that of \( p \). By denser surrounding, we mean presence of more number of points within the hypersphere with radius \( d_k(p, q) \) and center \( q \). Similarly, a negative value of the rank difference indicates that \( p \) has a denser surrounding than that of \( q \). For same values of \( R \) and \( k, p \) and \( q \) have equally dense surroundings. An illustration is shown in Fig. 1, where \( k = 4 \) and \( R = 6 \). So, their rank difference according to our definition is 2. In this case, as \( (R-k) \) is positive, the point \( q \) has denser surrounding than the point \( p \).

3. Proposed method

Our proposed method consists of two steps. In the first step, we construct an influence space around a test point. In the second step a rank difference based outlier score is assigned on the basis of this influence space.

3.1. Influence space construction

Influence space depicts a region with significantly high reverse density in the locality of a point under consideration. If the locations of the neighbors within the influence space [21,24] are more dense with respect to the locality of the concerned point, then a high value of outlierness score will be assigned to it. For an entire dataset, number of neighbors in the influence space is kept fixed.

In section 2, we have defined a reverse density \( \Omega_D \) which captures the density surrounding the locality of the neighboring points of a particular point. As the distance is increased from the target point, more number of neighbors get included in its surroundings resulting in different values of \( \Omega_D \). With successive addition of neighboring points, a set of reverse densities is obtained for each point at varying depths (number of neighboring points). The average reverse density \( \Omega_D \) for each depth is determined next. Note that we have considered the depth and not the distance around the neighbors to handle situations where there is empty space (no neighboring point is present) surrounding a given point. To avoid random fluctuations, the variation in the average reverse density with respect to depth has been smoothed using a Gaussian kernel in the following manner:

\[
\Omega_{\text{smoothed}} = \frac{1}{Nh_{\text{optimal}}} \sum_{i=1}^{n} e^{-\left( \frac{(\Omega_{i} - \hat{\sigma})^2}{2\hat{\sigma}^2} \right)}
\]  

where,

\[
h_{\text{optimal}} = 0.9\sigma \frac{N}{N^2}
\]  

and

\[
\sigma = \frac{\text{median}(\Omega_D - \text{median}(\Omega_D))}{0.6745}
\]  

where \( \sigma \) stands for an unbiased and consistent estimate of population standard deviation for large \( N \) [25,26].

In this smoothing process, an optimal width for the kernel \( h_{\text{optimal}} \) is determined using (4) [27] and (5) for better estimation of the significant density fluctuation around the neighbor points. We deem the first most significant peak [28,29] in this smoothed-kernel probability density function [25] as the limit of the influence space. The peak has been determined using the undecimated wavelet transform with Daubechies coefficients [30]. Such wavelet transforms can obtain peaks with maximum confidence by eliminating any surrounding noisy spurious peaks.

![Fig. 1. Schematic diagram of the distribution of neighbors around a test point p and different regions of interest with respect to p and q.](image-url)
Algorithm: rank-difference based outlier detection

Input: \( D \)

Output: top-\( n \) Outliers according to RDOS

Initialization: \( N(p), N_k(p), d(p, q), d_t(p, q), R, \Omega_1(p), \Omega_2(q), \Omega_k, h_{\text{optimal}}, \Omega_{\text{smoothed}}, k_{\text{optimal}}, \) Score, RDOS, RDOS_list(p), top_n_list

/*Finding sorted distances from \( p \) to all neighbors \( q \in D \) */
For all \( p \in D \) Do
For all \( q \in D \) Do
If \( q \neq p \) Then
\( d(p, q) \leftarrow \text{dist}(p, q) \)
\( N(p) \leftarrow q \)
End
End
\( d_k(p, q) \leftarrow \text{sort}(d(p, q), \text{‘ascend’}) \)
\( N_k(p) \leftarrow \text{sort}(N(p) \text{ by } d(p, q)) \)
End

*/Calculation of Reverse Neighborhood Rank of \( p \)*/
For all \( p \in D \) Do
For all \( q \in N_k(p) \) Do
\( R \leftarrow \text{Rank } p \text{ by } q \)
End
End

*/Average Reverse Density \( \Omega_k \) for varying depth \( k \)*/
For equal depth \( k \) Do
For all \( p \in D \) Do
\( \Omega_k(q) \leftarrow R \cdot d_k(p, q) \)
End
\( \Omega_k \leftarrow \text{mean}(\Omega_k(q)) \)
End

*/Kernel Smoothing of \( \Omega_k \) and Detection of First Most Significant Peak*/
\( h_{\text{optimal}} \) determination by (4) and (5)
\( \Omega_{\text{smoothed}} \leftarrow \text{Gaussian Kernel Smoothing of } \Omega_k \) using \( h_{\text{optimal}} \) in (3)
\( k_{\text{optimal}} \leftarrow \text{First Most Significant Peak of } \Omega_{\text{smoothed}} \)
using Daubechies wavelets

*/RDOS score assignment to \( p \)*/
For all \( p \in D \) Do
For equal depth \( k \) Do
\( \Omega_k(p) \leftarrow k \cdot d_k(p, q) \)
Score \( \leftarrow (R-k) / \Omega_k(p) \)
End
RDOS \( \leftarrow \text{median}(\text{Score}) \)
End
RDOS_list(p) \( \leftarrow \text{sort}(\text{RDOS, ‘descending’}) \)
\( \text{top}_n\_\text{list} \leftarrow \text{sort } N_k(p) \) by RDOS_list(p)

3.2. Outlier score

In the second part of our proposed algorithm we have used a rank-difference based score for ranking of the outliers. The positive value of the rank difference \((R-k)\) signifies the high concentration of the neighbors around the training point \( q \) than that of the test point \( p \). The negative and zero value respectively signify a lower or same concentration of the training points around \( q \) than that of \( p \). Thus the outlierness of the test point depends directly on the excess population of the neighborhood space of \( q \) with respect to the test point \( p \), i.e., on the rank difference \((R-k)\). Secondly, it also depends inversely on its own forward density \( \Omega_k(p) \). So, in summary,

\( \text{outlierness} \propto (R - k) \text{ for fixed distance } d_k(p, q) \)
\( \propto 1 / \Omega_k(p) \text{ for fixed distance } d_k(p, q) \)

Hence, the Rank-difference and Density-based Outlier Score (RDOS) can be written as follows

\[
\text{RDOS} = \text{median} \left( \frac{(R - k)}{\Omega_k(p)} \right)
\]

The median value has been used to have a more robust estimation of (RDOS).

3.3. Time-complexity

For the preprocessing of the data points the time complexity is \( O(N) \).

**Step 1**: The time-complexity for finding the distance between point \( p \) to all of the points in \( D \) is \( O(N) \) and for sorting the distances the time requirement is \( N \ln N \). Hence to repeat the same for all \( p \) in \( D \), the time-complexity is \( O(N(N + N \ln N)) = O(N^2 + N^2 \ln N) \).

**Step 2**: Now, for calculation of reverse rank the time-complexity for all of the points in \( D \) is again \( O(N^2) \).

**Step 3**: The time-complexity for finding the average reverse density \( \Omega_k \) for varying depth \( k \) (\( k \) varies from 1 to \( N \)) for all points in \( D \) is \( O(N^2) \).

**Step 4**: The total time-complexity for kernel smoothing of \( \Omega_k \) and for finding the first most significant peak is \( O(N^2 + 4\cdot N \cdot O(1)) = O(N) \).

**Step 5**: The time-complexity for assigning RDOS score to all points in \( D \) is given as \( O(Nk_{\text{optimal}} + 1) = O(Nk_{\text{optimal}}) \approx O(N^2) \). So, the total time-complexity is

\[
2^2O(N) + O(N^2 + N^2 \ln N) + 3^2O(N^2) \approx O(N^2 \ln N)
\]

4. Experiments

We have used two synthetic and five real datasets of UCI machine learning data repository for performance evaluation of our method. The Fig. 2 shows the distribution of the different datasets along with the rare class as outliers. For real dataset with dimensions > 3 pairwise plot of first four dimensions have been shown. The Fig. 3 shows the determination of \( k_{\text{optimal}} \) from the neighborhood density plot.

4.1. Metrics for measurement

For performance evaluation of the algorithms, we have used two metrics, namely recall and rank-power [31]. As per [22] let \( m \) most suspicious instances in a dataset \( D \) contains \( d_t \) true outliers and \( m_t \) be the number of true outliers detected by an algorithm. Then, recall (Re) measure which denotes the accuracy of an algorithm is given by:

\[
\text{Recall} = \frac{|m_t|}{|d_t|}
\]

If using a given detection method, true outliers occupy top positions with respect to the non-outliers among \( m \) suspicious instances, then the rank-power (RP) of the proposed method is said to be high [20]. If \( n \) denotes the number of outliers found within top \( m \) instances and \( m_l \) denote the rank of the \( l \)th true outlier, then the rank-power is given by:

\[
\text{RankPower} = \frac{n(n + 1)}{\sum_{l=1}^{n} m_l}
\]

Rank-power can have a maximum value of 1 when all \( n \) true outliers are in top \( n \) positions. For a fixed value of \( m \), larger values of these metrics imply better performance.

4.2. Experimental results

In synthetic datasets [22] we have six outliers. The real datasets considered are Iris, Ionosphere, Wisconsin breast cancer, Yeast and Cardiotocography all collected from the UCI archive. In the five real
datasets, we have randomly selected few points of same class as our rare objects or as target outliers. Table 1 provides some basic information regarding the above datasets. The synthetic datasets [22] have been plotted in Fig. 2 along with the outliers class marked individually. The IRIS dataset has been shown in the same Fig. 2. We have shown the pairwise dimensional plot of these real datasets up to 4th dimension. Since our method falls at the juncture of both density and rank based methods, we have compared our work with some representative methods from each category. Within the density based method, we have compared our work with LOF [19], COF [20] and INFLO [21]. Similarly, within the rank based group, we have compared our work with RBDA [22] and ODMRD [22,23].

4.2.1. Synthetic datasets

We have applied our proposed method on Synthetic 1 and on Synthetic 2 datasets of [22]. In our method the range of influence space $k$ is chosen dynamically which is quite advantageous with respect to the other methods in terms of its one-fold $k$ finding strategy. In case of LOF [19], COF [20], INFLO [21] and RBDA [22] the value of $k$ has been varied, which requires more time to complete.

In Table 2 the performance of Synthetic 1 and in Table 3 the performance of Synthetic 2 have been given. For Synthetic 1 dataset the value of $k$ is 3 whereas for Synthetic 2 dataset the value of $k$ is 7. The rank power of our method is always maximum in both the cases. Within top six values of RDOS score six outliers of the datasets have been identified.

4.2.2. Real datasets

In Tables 4–10 performance comparisons with Iris, Ionosphere, Wisconsin, Yeast and Cardiotocographic datasets are presented. For all the methods, we first compare $N_{rc}$ number of outliers detected within $m$ instances for a fixed value of $k$. We further compare number of outliers detected for fixed number of instances but with varying $k$. In Table 4 we have compared our results of RDOS with the results obtained using LOF [19], COF [20], INFLO [21], RBDA [22] and ODMRD [22,23] for Iris dataset. We have used an optimal $k$ for our method, whereas for the other methods the value of $k$ is 5. The value of our $k_{optimal}$ is 29. From Table 4 it is evident that the RBDA [22] and ODMRD [22,23] detects only 2 and 5 outliers respectively within first 10 instances. But they are not the top 5-instances of their score as evident from the rank-power of RBDA [22] and ODMRD [22,23] respectively. The rank-power of our method is 1 and it signifies that the 5-outliers of the Iris dataset have been detected within top 5 instances of the RDOS score. We have compared our results with the results of the other methods using $k = 5, 7$ and 10 in Table 5. The value of our $k$ is fixed at 29, whereas for the other methods $k$ has been varied to get the optimal performance. All of the results have been compared within top-5 instances. Table 5 establishes the effectiveness of $k_{optimal}$ over varying $k$.

The Ionosphere dataset contains 351 instances with 34 attributes; all attributes are normalized in the range of 0 and 1. There are two
Table 2
Performance of the synthetic 1 dataset [22].

<table>
<thead>
<tr>
<th>km</th>
<th>LOF</th>
<th>COF</th>
<th>INFLO</th>
<th>RBDA</th>
<th>RDOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>5</td>
<td>0.83</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>1</td>
<td>0.955</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.83</td>
<td>1</td>
<td>5</td>
<td>0.83</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>1</td>
<td>0.955</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>0.83</td>
<td>1</td>
<td>5</td>
<td>0.83</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>1</td>
<td>0.955</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>0.83</td>
<td>1</td>
<td>5</td>
<td>0.83</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>1</td>
<td>0.913</td>
<td>6</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3
Performance of the synthetic 2 dataset [22].

<table>
<thead>
<tr>
<th>km</th>
<th>LOF</th>
<th>COF</th>
<th>INFLO</th>
<th>RBDA</th>
<th>RDOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>4</td>
<td>0.67</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>0.83</td>
<td>1</td>
<td>0.882</td>
<td>3</td>
</tr>
<tr>
<td>35</td>
<td>5</td>
<td>1</td>
<td>0.17</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>0.67</td>
<td>1</td>
<td>0.714</td>
<td>6</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>0.33</td>
<td>1</td>
<td>0.375</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>0.33</td>
<td>0.25</td>
<td>1</td>
<td>0.17</td>
</tr>
</tbody>
</table>

Table 4
Performance comparison on the Iris dataset with top 20 instances with fixed $k$.

<table>
<thead>
<tr>
<th>m</th>
<th>LOF ($k = 5$)</th>
<th>COF ($k = 5$)</th>
<th>INFLO ($k = 5$)</th>
<th>RBDA ($k = 5$)</th>
<th>ODMRD ($k = 5$)</th>
<th>RDOS ($k_{optimal} = 29$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0.2</td>
<td>0.2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>0.8</td>
<td>0.313</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>5</td>
<td>1</td>
<td>0.341</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>5</td>
<td>1</td>
<td>0.341</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 5
Performance comparison on the Iris dataset with fixed $m = 5$.

<table>
<thead>
<tr>
<th>k</th>
<th>LOF</th>
<th>COF</th>
<th>INFLO</th>
<th>RBDA</th>
<th>ODMRD</th>
<th>RDOS ($k_{optimal} = 29$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0.2</td>
<td>0.2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 6
Performance comparison on the Ionosphere dataset with top 85 instances with fixed $k$.

<table>
<thead>
<tr>
<th>m</th>
<th>LOF ($k = 15$)</th>
<th>COF ($k = 15$)</th>
<th>INFLO ($k = 15$)</th>
<th>RBDA ($k = 15$)</th>
<th>ODMRD ($k = 15$)</th>
<th>RDOS ($k_{optimal} = 22$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.5</td>
<td>1</td>
<td>5</td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>6</td>
<td>0.6</td>
<td>1</td>
<td>6</td>
<td>0.6</td>
<td>0.913</td>
</tr>
<tr>
<td>30</td>
<td>7</td>
<td>0.7</td>
<td>0.757</td>
<td>8</td>
<td>0.8</td>
<td>0.522</td>
</tr>
<tr>
<td>60</td>
<td>9</td>
<td>0.9</td>
<td>0.354</td>
<td>9</td>
<td>0.9</td>
<td>0.372</td>
</tr>
<tr>
<td>85</td>
<td>9</td>
<td>0.9</td>
<td>0.354</td>
<td>9</td>
<td>0.9</td>
<td>0.372</td>
</tr>
</tbody>
</table>

Table 7
Performance comparison on the Ionosphere dataset with fixed $m = 85$.

<table>
<thead>
<tr>
<th>k</th>
<th>LOF</th>
<th>COF</th>
<th>INFLO</th>
<th>RBDA</th>
<th>ODMRD</th>
<th>RDOS ($k_{optimal} = 22$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
<td>Re</td>
<td>RP</td>
<td>Nrc</td>
</tr>
<tr>
<td>11</td>
<td>9</td>
<td>0.9</td>
<td>0.294</td>
<td>9</td>
<td>0.9</td>
<td>0.29</td>
</tr>
<tr>
<td>15</td>
<td>9</td>
<td>0.9</td>
<td>0.354</td>
<td>9</td>
<td>0.9</td>
<td>0.372</td>
</tr>
<tr>
<td>20</td>
<td>9</td>
<td>0.9</td>
<td>0.417</td>
<td>9</td>
<td>0.9</td>
<td>0.413</td>
</tr>
<tr>
<td>23</td>
<td>9</td>
<td>0.9</td>
<td>0.441</td>
<td>9</td>
<td>0.9</td>
<td>0.464</td>
</tr>
</tbody>
</table>
classes labeled as good and bad with 225 and 126 instances respectively. There are no duplicate instances in the dataset. To form the rare class, 116 instances from the bad class are randomly removed. The final dataset has only 235 instances with 225 good and 10 bad instances [22,23]. We have compared our results of RDOS with the results of LOF [19], COF [20], INFLO [21], RBDA [22] and ODMRD [22,23] in Table 6. For other methods $k = 15$ whereas in our proposed method $k$ has been chosen dynamically. With our $k_{\text{optimal}} = 22$, 10 outliers are detected within top 15 instances making the rank-power RP value (0.965) much higher than that of LOF [19], COF [20], INFLO [21], RBDA [22] and ODMRD. In Table 7 the optimal value of $k$ is fixed at 22 for Ionosphere data.

In Fig. 4 rank power of the Iris, Ionosphere and Wisconsin data has been plotted against top $m$ instances for different competing methods. The figure clearly portrays the superior performance of the proposed method.

We have also shown the effectiveness of our proposed method in case of high dimensional dataset Yeast and Cardiotocography. In case of Wisconsin data, within 449 instances 226 malignant instances have been removed randomly. In our experiments the final dataset consisted of 213 benign instances and 10 malignant instances. With $k = 11$ for the LOF [19], COF [20], INFLO [21], RBDA [22] and ODMRD [22,23], our method is compared with dynamic $k = 58$ for 40 instances. The comparison results are shown in Table 8. The rank-power RP of our method (0.982) is once again found to be very high. This signifies that all the 10 outliers have been detected within top-15 instances of the RDOS score. Table 9 shows the effectiveness of the optimal value of $k$ for this dataset.

Table 8

<table>
<thead>
<tr>
<th>$m$</th>
<th>LOF($k = 11$)</th>
<th>COF($k = 11$)</th>
<th>INFLO($k = 11$)</th>
<th>RBDA($k = 11$)</th>
<th>ODMRD($k = 11$)</th>
<th>RDOS($k_{\text{optimal}} = 58$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>5</td>
<td>0.5</td>
<td>0.469</td>
<td>6</td>
<td>0.6</td>
<td>0.656</td>
</tr>
<tr>
<td>40</td>
<td>9</td>
<td>0.9</td>
<td>0.372</td>
<td>10</td>
<td>1</td>
<td>0.444</td>
</tr>
</tbody>
</table>

Table 9

<table>
<thead>
<tr>
<th>$k$</th>
<th>LOF</th>
<th>COF</th>
<th>INFLO</th>
<th>RBDA</th>
<th>ODMRD</th>
<th>RDOS</th>
<th>$k_{\text{optimal}}$</th>
<th>Nrc</th>
<th>Re</th>
<th>RP</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>10</td>
<td>1</td>
<td>0.64</td>
<td>10</td>
<td>1</td>
<td>0.618</td>
<td>58</td>
<td>10</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>9</td>
<td>0.9</td>
<td>0.372</td>
<td>10</td>
<td>1</td>
<td>0.444</td>
<td>8</td>
<td>0.8</td>
<td>0.8</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>9</td>
<td>0.9</td>
<td>0.395</td>
<td>10</td>
<td>1</td>
<td>0.474</td>
<td>9</td>
<td>0.9</td>
<td>0.495</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>9</td>
<td>0.9</td>
<td>0.446</td>
<td>10</td>
<td>1</td>
<td>0.451</td>
<td>9</td>
<td>0.9</td>
<td>0.517</td>
<td></td>
</tr>
</tbody>
</table>

Table 10

<table>
<thead>
<tr>
<th>Dataset</th>
<th>$m$</th>
<th>LOF</th>
<th>RDOS</th>
<th>$k_{\text{optimal}}$</th>
<th>Average processing time of LOF (sec)</th>
<th>Average processing time of RDOS (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yeast</td>
<td>20</td>
<td>3</td>
<td>0.6</td>
<td>111</td>
<td>431</td>
<td>18.486</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.111</td>
<td>1</td>
<td>20</td>
<td>5 1 1</td>
<td>17.2968</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>3</td>
<td>0.6</td>
<td>111</td>
<td>1 1 1</td>
<td>17.564</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>4</td>
<td>0.8</td>
<td>125</td>
<td>5 1 1</td>
<td>17.564</td>
</tr>
<tr>
<td></td>
<td>35</td>
<td>5</td>
<td>1</td>
<td>1.3</td>
<td>5 1 1</td>
<td>17.564</td>
</tr>
<tr>
<td>Cardiotocography</td>
<td>60</td>
<td>37</td>
<td>0.617</td>
<td>0.695</td>
<td>20</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>120</td>
<td>48</td>
<td>0.8</td>
<td>0.577</td>
<td>60</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>180</td>
<td>53</td>
<td>0.883</td>
<td>0.507</td>
<td>60</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>240</td>
<td>56</td>
<td>0.933</td>
<td>0.460</td>
<td>60</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>300</td>
<td>58</td>
<td>0.967</td>
<td>0.427</td>
<td>60</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>360</td>
<td>60</td>
<td>1</td>
<td>0.391</td>
<td>60</td>
<td>1</td>
</tr>
</tbody>
</table>
of Yeast dataset out of 1484 instances 5 outliers have been detected within top 5 instances and in case of Cardiotocography dataset out of 2069 instances 60 outliers have been detected within top 70 instances. The rank-power RP of our method in case of Yeast dataset is 1 and in case of Cardiotocography dataset is 0.996. These are significantly high with respect to RP of LOF in case of high dimensional dataset also. In Fig. 5 rank power of the Yeast and Cardiotocography data has been plotted against top m instances for LOF and RDOS. The figure clearly shows the effectiveness of the proposed method in case of high dimensional dataset also.

5. Conclusion

We proposed a novel method for outlier detection using neighborhood rank difference. Experimental results clearly indicate the effectiveness of the proposed method for both synthetic as well as some real datasets with varying dimensions and available instances. Unlike some of the existing approaches, we determine a fixed influence space for a given dataset and compute the outlier score in an unsupervised manner without varying the number of neighboring points. In future, we plan to make the determination of the influence space more accurate to further improve the performance of our outlier detection algorithm.

References