Neurocomputing 241 (2017) 171–180

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A local density-based approach for outlier detection

Bo Tang[a], Haibo He[b,*]

[a] Department of Computer Science, Hofstra University, Hempstead, NY 11549 USA
[b] Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI 02881 USA

ARTICLE INFO

Article history:
Received 6 January 2017
Revised 23 January 2017
Accepted 4 February 2017
Available online 20 February 2017

Communicated by Dr. Nianyin Zeng

Keywords:
Outlier detection
Reverse nearest neighbors
Shared nearest neighbors
Local kernel density estimation

ABSTRACT

This paper presents a simple and effective density-based outlier detection approach with local kernel density estimation (KDE). A Relative Density-based Outlier Score (RDOS) is introduced to measure local outlierness of objects, in which the density distribution at the location of an object is estimated with a local KDE method based on extended nearest neighbors of the object. Instead of using only k nearest neighbors, we further consider reverse nearest neighbors and shared nearest neighbors of an object for density distribution estimation. Some theoretical properties of the proposed RDOS including its expected value and false alarm probability are derived. A comprehensive experimental study on both synthetic and real-life data sets demonstrates that our approach is more effective than state-of-the-art outlier detection methods.

1. Introduction

Advances in data acquisition have created massive collections of data, capturing valuable information to science, government, business, and society. However, despite of the availability of large amount of data, some events are rare or exceptional, which are usually called “outliers” or “anomalies”. Compared with many other knowledge discovery problems, outlier detection is sometimes more valuable in many applications, such as network intrusion detection, fraudulent transactions, and medical diagnostics. For example, in network intrusion detection, the number of intrusions or attacks (“bad” connections) is much less than the “good” and normal connections. Similarly, the abnormal behaviors are usually rare in many other cases. Although these outliers are only a small portion of the whole data set, it is much more costly to misunderstand them compared with other events.

In recent decades, many outlier detection approaches have been proposed. Usually an outlier detection method can be categorized into the following four types of method [1,2]: distribution-based, distance-based, clustering-based, and density-based. In distribution-based methods, an object is considered as the outlier if it deviates from a standard distribution (e.g., normal, Poisson, among others) too much [3]. The problem of the distribution-based method is that the underlying distribution is usually unknown and the data may not follow the standard distribution for many practical applications.

The distance-based methods detect outliers by computing distances among all objects. An object is considered as the outlier when it has d_0 distance away from p_0 percentage of objects in the data set [4]. The distances among objects need to be calculated in either raw data space or feature subspace. For high dimensional data sets, feature reduction is usually needed before calculating the distance [5]. To detect local outliers in a data set with multiple components or clusters, a top-n kth nearest neighbor distance is proposed [6], in which the distance from an object to its kth nearest neighbor indicates outlierness of the object. The cluster-based methods detect outliers in the process of finding clusters. The object does not belong any cluster is considered as the outlier [7–9].

In density-based methods, an outlier is detected when its local density differs from its neighborhood. Different density estimation methods can be applied to measure the density. For example, Local Outlier Factor (LOF) [10] is an outlierness score indicating how an object differs from its locally reachable neighborhood. Following studies [11] have shown that it is more reliable to consider those objects who have the highest LOF scores as outliers, instead of comparing the LOF score with a threshold. Lately, several variations of the LOF have been proposed [12,13]. A Local Distance-based Outlier Factor (LDOF) using the relative distance from an object to its neighbors is proposed for outlier detection in scattered datasets [12], and a INFLuenced Outlierness (INFLO) score is proposed with the consideration of both neighbors and reverse neighbors of an object when estimating its relative density distribution [13]. Considering underlying patterns of the data, Tang et. al. proposed a

* Corresponding author.
E-mail addresses: bo.tang@hofstra.edu (B. Tang), he@ele.uri.edu, hhbhust@gmail.com (H. He).
URL: http://www.ele.uri.edu/faculty/he/ (H. He)

http://dx.doi.org/10.1016/j.neucom.2017.02.039
0925-2312/

connectivity-based outlier factor (COF) scheme [14]. While all these LOF-based and COF-based outlier detection methods measure the relative distribution through distance, several other density-based methods have been proposed based on kernel density estimation [15–17]. For example, Local Density Factor (LDF) [15] is proposed by extending the LOF with kernel density estimation. Similar to the LOCI, a relative density score termed KDEOS [17] is calculated using kernel density estimation and applies the z-score transformation for score normalization.

In this paper, we propose an outlier detection method based on the local kernel density estimation for robust local outlier detection. Instead of using the whole data set, the density of an object is estimated with the objects in its neighborhood. Three kinds of neighbors: k nearest neighbors, reverse nearest neighbors, and shared nearest neighbors, are considered in our local kernel density estimation. Our motivation behind using these three types of nearest neighbors is from the success of connectivity-based outlier factor [14] and extended nearest neighbors methods [18]. These three types of nearest neighbors contain different connectivity information, which allows our method to be flexible to model different local patterns of normal data. A simple and effective relative density calculation, termed Relative Density-based Outlier Score (RDOS), is introduced to measure the outlierness. Theoretical properties of the RDOS, including the expected value and the false alarm probability are derived, which suggests parameter settings in practical applications. We further employ the top-n scheme to rank the objects with their outlierness, i.e., the objects with the highest RDOS are considered as the outliers. Simulation results on both synthetic data sets and real-life data sets illustrate superior performance of our proposed method. We note that the most similar work to ours are the robust kernel-based outlier factor (RKOF) [16], which is a ratio of weighted density estimate of its neighbors to its local density estimate, and the KDEOS [17], which uses a z-score of local density estimate. However, for a given object, unlike both RKOF and RKOF, we use an extended relative local density estimate as outlier score, considering its nearest neighbors, reverse nearest neighbors, and shared nearest neighbors. In summary, our contributions of this paper are three-fold:

- A new local kernel density estimation based approach for outlier detection is proposed, in which three types of nearest neighbors, including k nearest neighbors, reverse nearest neighbors, and shared nearest neighbors, are considered.
- Theoretical analysis of the proposed outlier detection approach is provided, which suggests the setting of parameters for its practical use.
- Performance improvement of the proposed approach is also demonstrated from extensive experiments on both synthetic and real-life data sets, compared to the state-of-the-art approaches.

The paper is organized as follows: In Section 2, we introduce the definition of the RDOS and present the detailed descriptions of our proposed outlier detection approach. In Section 3, we derive theoretical properties of the RDOS and discuss the parameter settings for real-life applications. In Section 4, we present experimental results and analysis, which show superior performance compared with previous approaches. Finally, conclusions are given in Section 5.

2. Proposed outlier detection

2.1. Local kernel density estimation

We use the KDE method to estimate the density at the location of an object based on the given data set. Given a set of objects $\mathcal{X} = \{X_1, X_2, \ldots, X_m\}$, where $X_i \in \mathbb{R}^d$ for $i = 1, 2, \ldots, m$, the KDE method estimates the distribution as follows:

$$p(X) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{h^d}K\left(\frac{X - X_i}{h}\right) \tag{1}$$

where $K(\frac{X-X_i}{h})$ is the defined kernel function with the kernel width of h, which satisfies the following conditions:

$$\int K(u)du = 1, \int uK(u)du = 0, \text{and} \int u^2K(u)du > 0 \tag{2}$$

A commonly used multivariate Gaussian kernel function is given by

$$K\left(\frac{X - X_i}{h}\right)_{\text{Gaussian}} = \frac{1}{(2\pi)^{d/2}}\exp\left(-\frac{\|X - X_i\|^2}{2h^2}\right) \tag{3}$$

where $\|X - X_i\|$ denotes the Euclidean distance between X and X_i. The distribution estimate in Eq. (1) offers many nice properties, such as its non-parametric property, continuity and differentiability [19,20]. Also it is an asymptotic unbiased estimator of the density.

To estimate the density at the location of the object X_p, we only consider its neighbors of X_p as kernels, instead of using all objects in the data set. The reason for this is twofold: firstly, many complex real-life data sets usually have multiple clusters or components, which are the intrinsic patterns of the data. The density estimation using the full data set may lose the local difference in density and fail to detect the local outliers; secondly, the outlier detection will calculate the score for each object, and using the full data set would lead to a high computational cost, which has the complexity of $O(N^2)$ where N is the total number of objects in the data set.

To better estimate the density distribution in the neighborhood of an object, we propose to use k nearest neighbors, reverse nearest neighbors and shared nearest neighbors as kernels in KDE. Let $NN_r(X_p)$ be the rth nearest neighbors of the object X_p, we denote the set of k nearest neighbors of X_p as $\mathcal{S}_{KNN}(X_p)$:

$$\mathcal{S}_{KNN}(X_p) = \{NN_1(X_p), NN_2(X_p), \ldots, NN_k(X_p)\} \tag{4}$$

The *reverse nearest neighbors* of the object X_p are those objects who consider X_p as one of their k nearest neighbors, i.e., X is one reverse nearest neighbor of X_p if $NN_r(X) = X_p$ for any $r \le k$. Recent studies have shown that reverse nearest neighbors are able to provide useful information of local data distribution and have been successfully used for clustering [21], outlier detection [13], and classification [18]. The *shared nearest neighbors* of the object X_p are those objects who share one or more nearest neighbors with X_p, in other words, X is one shared nearest neighbor of X_p if $NN_r(X) = NN_s(X_p)$ for any $r, s \le k$. We show these three types of nearest neighbors in Fig. 1.

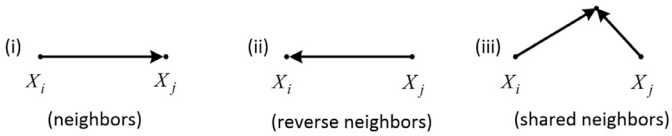


Fig. 1. Three types of nearest neighbors considered. Arrows from X_i and X_j to $NN_r(X_i)$ and $NN_s(X_j)$, respectively.

We denote $\mathcal{S}_{RNN}(X_p)$ and $\mathcal{S}_{SNN}(X_p)$ by the sets of reverse nearest neighbors and shared nearest neighbors of X_p, respectively. For an object, there would be always k nearest neighbors in $\mathcal{S}_{KNN}(X_p)$, while the sets of $\mathcal{RNN}(X_p)$ and $\mathcal{SNN}(X_p)$ can be empty or have one or more elements. Given the three data sets $\mathcal{S}_{KNN}(X_p)$, $\mathcal{S}_{RNN}(X_p)$ and $\mathcal{S}_{SNN}(X_p)$ for the object X_p, we form an extended local neighborhood by combining them together, denoted by $\mathcal{S}(X_p) = \mathcal{S}_{KNN}(X_p) \cup \mathcal{S}_{RNN}(X_p) \cup \mathcal{S}_{SNN}(X_p)$. Thus, the estimated density at the

location of X_p is written as

$$p(X_p) = \frac{1}{|\mathcal{S}(X_p)|+1} \sum_{X \in \mathcal{S}(X_p) \cup \{X_p\}} \frac{1}{h^d} K\left(\frac{X - X_p}{h}\right) \tag{5}$$

where $|\mathcal{S}|$ denotes the number of elements in the set of $\mathcal{S}$.

2.2. Relative density-based outlier factor

After estimating the density at the locations of all objects, we propose a novel relative density-based outlier factor (RDOS) to measure the degree to which the density of the object X_p deviates from its neighborhood, which is defined as follows:

$$RDOS_k(X_p) = \frac{\sum_{X_i \in \mathcal{S}(X_p)} p(X_i)}{|\mathcal{S}(X_p)| p(X_p)} \tag{6}$$

The RDOS is the ratio of the average neighborhood density to the density of interested object X_p. If $RDOS_k(X_p)$ is much larger than 1, then the object X_p would be outside of a dense cluster, indicating that X_p would be an outlier. If $RDOS_k(X_p)$ is equal or smaller than 1, then the object X_p would be surrounded by the same dense neighbors or by a sparse cloud, indicating that X_p would not be an outlier. In practice, we would like to rank the RDOS and detect top-n outliers. We summarize our algorithm in Algorithm 1, which takes the KNN graph as input. The KNN graph is a directed graph in which each object is a vertex and is connected to its k nearest neighbors with an outbound direction. In the KNN graph, an object will have k outbound edges to the elements in $\mathcal{S}_{KNN}$, and have none, one or more inbound edges. The KNN graph construction using the brute-force method has the computational complexity of $O(N^2)$ for N objects, and it can be reduced to $O(N \log N)$ using the $k-d$ trees [22]. Using the KNN graph KNN-G, it is easy to obtain the k nearest neighbors $\mathcal{S}_{KNN}$, reverse nearest neighbors $\mathcal{S}_{RNN}$ and shared nearest neighbors $\mathcal{S}_{SNN}$ with an approximate computational cost of $O(N)$. For each object, we form a set of local nearest neighbors $\mathcal{S}$ with the combination of $\mathcal{S}_{KNN}$, $\mathcal{S}_{RNN}$ and $\mathcal{S}_{SNN}$, and calculate the density at the location of the object based on the set of $\mathcal{S}$. Then, we calculate the RDOS of each object based on the densities of local neighbors in $\mathcal{S}$. The top-n outliers are obtained by sorting the RDOS in a descending way. If one wants to determine whether an object X_p is outlier, we can compare the value of $RDOS_k(X_p)$ with a threshold τ, i.e., we determine an object is outlier if $RDOS_k(X_p)$ satisfies

$$RDOS_k(X_p) > \tau \tag{7}$$

where the threshold τ is usually a constant value that is predetermined by users.

Algorithm 1: RDOS for top-n outlier detection based on the KNN graph.

INPUT: k, $\mathcal{X}$, d, h, the KNN graph KNN-G.
OUTPUT: top-n objects in $\mathcal{X}$.
ALGORITHM:
foreach *object* $X_p \in \mathcal{X}$ **do**
1 $\mathcal{S}_{KNN}(X_p) = getOutboundObjects(\text{KNN-G}, X_p)$: get k nearest neighbors of X_p;
2 $\mathcal{S}_{RNN}(X_p) = getInboundObjects(\text{KNN-G}, X_p)$: get reverse nearest neighbors of X_p;
3 $\mathcal{S}_{SNN}(X_p) = \emptyset$: initialize shared nearest neighbors of X_p;
4 **foreach** *object* $X \in \mathcal{S}_{KNN}(X_p)$ **do**
5 $\mathcal{S}_{RNN}(X) = getInboundObjects(\text{KNN-G}, X)$;
6 $\mathcal{S}_{SNN}(X_p) = \mathcal{S}_{SNN}(X_p) \cup \mathcal{S}_{RNN}(X)$: get objects who share X as nearest neighbors with X_p;
end
7 $\mathcal{S}(X_p) = \mathcal{S}_{KNN}(X_p) \cup \mathcal{S}_{RNN}(X_p) \cup \mathcal{S}_{SNN}(X_p)$;
8 $p(X_p) = getKernelDensity(\mathcal{S}(X_p), X_p, h)$: estimate the local kernel density at the location of X_p;
end
foreach *object* $X_p \in \mathcal{X}$ **do**
9 Calculate $RDOS_k(X_p)$ for X_p according to Eq. (6);
end
10 Sort RDOS in a descending way and output the top-n objects.

3. Theoretical properties

In this section, we analyze several nice properties of the proposed outlierness metric. In Theorem 1, we give the expected value of RDOS when the object and its neighbors are sampled from the same distribution, which indicates the lower bound of RDOS for outlier detection.

Theorem 1. *Let the object X_p be sampled from a continuous density distribution. For $N \to \infty$, the RDOS equals 1 with probability 1, i.e., $RDOS_k(X_p) = 1$, when the kernel function K is nonnegative and integrable.*

Proof. For a fixed k, $N \to \infty$ indicates that the objects in $\mathcal{S}(X_p)$ locate in the local neighborhood of X_p with the radius $r \to 0$. Considering data sampled from a continuous density distribution $f(x)$, the expectation of the density estimation at X_p exists and is consistent to the true one [23]:

$$\mathbb{E}[p(X_p)] = f(X_p) \int K(u)du = f(X_p) \tag{8}$$

and its asymptotic variance is given by [23]

$$\mathbb{V}ar[p(X_p)] = 0 \tag{9}$$

Meanwhile, the average density at the neighborhood of X_p with the radius of $r \to 0$ can be given by

$$\mathbb{E}[\bar{p}(X_p)] = \mathbb{E}\left[\frac{\sum_{X_i \in \mathcal{S}(X_p)} p(X_i)}{|\mathcal{S}(X_p)|}\right] = \mathbb{E}[p(X_p)] = f(X_p) \tag{10}$$

Taking the ratio, we get

$$\mathbb{E}[\bar{p}(X_p)] / \mathbb{E}[p(X_p)] = 1 \tag{11}$$

□

This theorem shows that when $RDOS_k(X_p) \approx 1$, we could say that the object X_p is not an outlier. Since RDOS is always positive, when $0 < RDOS_k(X_p) < 1$, we could say the object X_p can be ignored in outlier detection. Only these objects whose RDOS are much larger than 1 are possible to be outliers.

Next, we examine the upper-bound of false detection probability, which might provide a guidance for threshold selection in practical applications. It is based on the assumption that the neighbors of an object are uniformly distributed, which is true when the number of data goes infinity. Hence, the following theorem can be considered to be the asymptotic property of the proposed RDOS approach.

Theorem 2. *Let $\mathcal{S}(X_p)$ be the set of local neighbors of X_p in RDOS, which are assumed to be uniformly distributed in ball B_r centered at X_p with the radius of r. Using the Gaussian kernel, the probability of false detecting X_p as an outlier is given by*

$$P[RDOS_k(X_p) > \gamma] \le \exp\left(-\frac{2(\gamma-1)^2(|\mathcal{S}|+1)^2(2\pi)^d h^{2d}}{|\mathcal{S}|(2|\mathcal{S}|+\gamma+1)^2 V^2}\right) \tag{12}$$

where h is the kernel width and V is the volume of ball B_r.

Proof. For simplicity of notation, we use $\mathcal{S}$ for $\mathcal{S}(X_p)$ and consider $X_p = 0$. Then, the density estimation at X_p given the local neighbors $X_1, X_2, \ldots, X_{|\mathcal{S}|}$ is written as

$$p(X_p) = \frac{1}{|\mathcal{S}|+1} \sum_{X_i \in \mathcal{S} \cup X_p} \frac{1}{(2\pi)^{d/2} h^d} \exp\left(-\frac{\|X_i\|^2}{2h^2}\right) \tag{13}$$

and the average density estimation in the neighborhood of X_p is written as

$$\begin{aligned} \tilde{p}(X_p) &= \frac{1}{|\mathcal{S}|} \sum_{X_i \in \mathcal{S}} p(X_i) \\ &= \frac{1}{|\mathcal{S}|(|\mathcal{S}|+1)} \sum_{X_i \in \mathcal{S}} \sum_{X_j \in \mathcal{S} \cup X_p} \frac{1}{(2\pi)^{\frac{d}{2}} h^d} \exp\left(-\frac{\|X_i - X_j\|^2}{2h^2}\right) \end{aligned} \tag{14}$$

For X_i, $i = 1, 2, \ldots, |\mathcal{S}|$, uniformly distributed in ball B_r, we can compute the expectation of both $p(X_p)$ and $\tilde{p}(X_p)$ from Theorem 1, which is given by:

$$\mathbb{E}[\tilde{p}(X_p)] = \mathbb{E}[p(X_p)] = \frac{1}{V} = \frac{\pi^{n/2} r^n}{\Gamma(n/2+1)} \tag{15}$$

where V is the volume of n-sphere B_r and $n = d - 1$. The rest of proof follows the McDiarmid's Inequality which gives the upper bound of the probability that a function of i.i.d. variables $f(X_1, X_2, \ldots, X_{|\mathcal{S}|})$ deviates from its expectation. Let $f : \mathbb{R}^d \to \mathbb{R}$, $\forall i, \forall x_1, \ldots, x_{|\mathcal{S}|}, x_i' \in \mathcal{S}$,

$$|f(x_1, \ldots, x_i, \ldots, x_{|\mathcal{S}|}) - f(x_1, \ldots, x'_i, \ldots, x_{|\mathcal{S}|})| \le c_i \tag{16}$$

Then, for all $\epsilon > 0$,

$$\mathbb{P}[f - \mathbb{E}(f) \ge \epsilon] \le \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^{|\mathcal{S}|} c_i^2}\right) \tag{17}$$

For $f_1 = p(X_p)$, we have

$$\begin{aligned} &|f_1(x_1, \ldots, x_i, \ldots, x_{|\mathcal{S}|}) - f_1(x_1, \ldots, x_i', \ldots, x_{|\mathcal{S}|})| \\ &= \frac{K(X_i/h) - K(X_i'/h)}{h^d(|\mathcal{S}|+1)} \le \frac{1 - \exp\left(-r^2/2h^2\right)}{(2\pi)^{d/2} h^d (|\mathcal{S}|+1)} = c_1 \end{aligned} \tag{18}$$

For $f_2 = \tilde{p}(X_p)$, we have

$$\begin{aligned} &|f_2(x_1, \ldots, x_i, \ldots, x_{|\mathcal{S}|}) - f_2(x_1, \ldots, x_i', \ldots, x_{|\mathcal{S}|})| \\ &= \frac{K(\frac{X_i}{h}) - K(\frac{X_i'}{h}) + 2 \sum_{j=1, j\neq i}^{|\mathcal{S}|} \left[K(\frac{X_i - X_j}{h}) - K(\frac{X'_i - X_j}{h})\right]}{h^d(|\mathcal{S}|+1)} \\ &\le \frac{1 - \exp\left(-r^2/2h^2\right) + 2|\mathcal{S}|\left(1 - \exp\left(-2r^2/h^2\right)\right)}{(2\pi)^{d/2} h^d (|\mathcal{S}|+1)} = c_2 \end{aligned} \tag{19}$$

We define a new function $f = f_2 - \gamma f_1$, which is bounded by

$$|f| \le |f_2| + \gamma |f_1| \le c_2 + \gamma c_1 \le \frac{2|\mathcal{S}| + \gamma + 1}{(2\pi)^{d/2} h^d (|\mathcal{S}|+1)} = c \tag{20}$$

Then, the probability of false alarm is written as

$$\begin{aligned} \mathbb{P}[RDOS_k(X_p) > \gamma] &= \mathbb{P}[\tilde{p}(X_p) - \gamma p(X_p)] \\ &= \mathbb{P}[f - \mathbb{E}(f) > t] \end{aligned} \tag{21}$$

where $t = (\gamma - 1)/V$. From Theorem 1, we are only interested in the case of $RDOS_k(X_p) > 1$, i.e., $\gamma > 1$, and $t > 0$. Using the McDiarmid's Inequality, we have

$$\begin{aligned} &\mathbb{P}[RDOS_k(X_p) > \gamma] \le \exp\left(-\frac{2t^2}{\sum_{i=1}^{|\mathcal{S}|} c^2}\right) = \exp\left(-\frac{2t^2}{|\mathcal{S}|c^2}\right) \\ &\le \exp\left(-\frac{2(\gamma-1)^2(|\mathcal{S}|+1)^2(2\pi)^d h^{2d}}{|\mathcal{S}|(2|\mathcal{S}|+\gamma+1)^2 V^2}\right) \end{aligned} \tag{22}$$

□

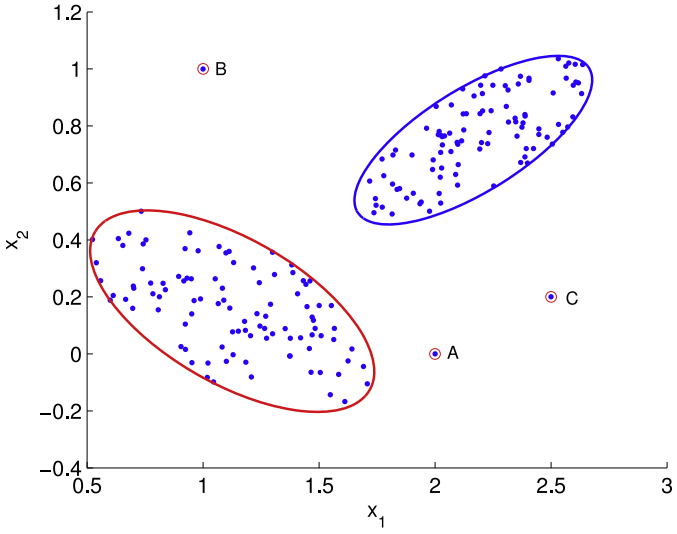


Fig. 2. Distribution of normal data and outliers, where the objects: *A*, *B*, and *C* are outliers.

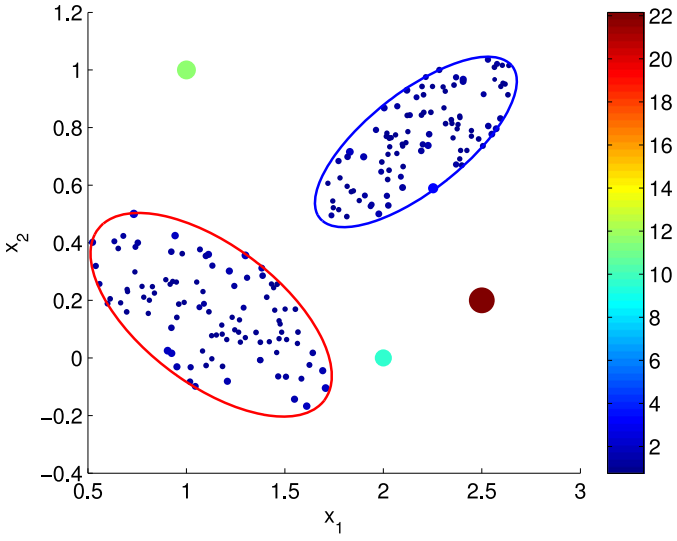


Fig. 3. Outlierness scores of all data samples, where the value of RDOS is illustrated by the color and the radius of the circle.

4. Experimental results and analysis

4.1. Synthetic data sets

We first test the proposed RDOS in two synthetic data sets for outlier detection. Our first synthetic data set includes two Gaussian clusters centered at (0.5, 0.8) and (2, 0.5), respectively, each of which has 100 data samples. There are three outliers around these two clusters, as indicated in Fig. 2. To calculate the RDOS, we use $k = 21$ nearest neighbors and $h = 0.01$ in kernel functions. In Fig. 3, we show the RDOS of all data samples, where the color and the radius of circles denote the value of RDOS. It can be shown that the RDOS of these three outliers is significantly larger than that of non-outliers. We further rank data samples in a descending way according to the RDOS, and show the top $n = 5$ data samples with the largest RDOS in Table 1.

The second synthetic data set used in our simulation consists of data samples uniformly distributed around a cosine curve, which can be written as

$$x_2 = cos(x_1) + w \tag{23}$$

Table 1
Top $n = 5$ data samples with the largest RDOS in two synthetic data sets.

Gaussian data set			Cosine data set		
Rank	Data	RDOS	Rank	Data	RDOS
1	(2.50, 0.20)	22.16	1	(8.80, 0.00)	11.78
2	(1.00, 1.00)	11.68	2	(1.80, 1.00)	11.72
3	(2.00, 0.00)	9.64	3	(5.70, 0.20)	11.68
4	(2.25, 0.59)	3.40	4	(6.50, 0.70)	10.93
5	(0.73, 0.50)	2.31	5	(2.38, −0.45)	2.12

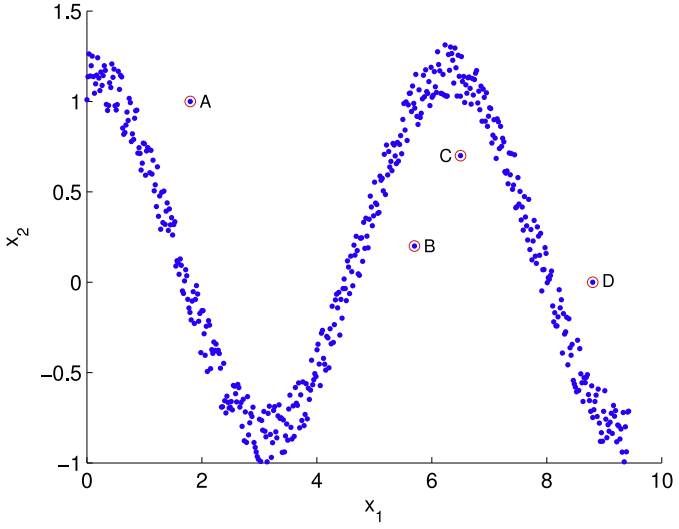


Fig. 4. Distribution of normal data and outliers, where *A, B, C* and *D* are considered as outliers.

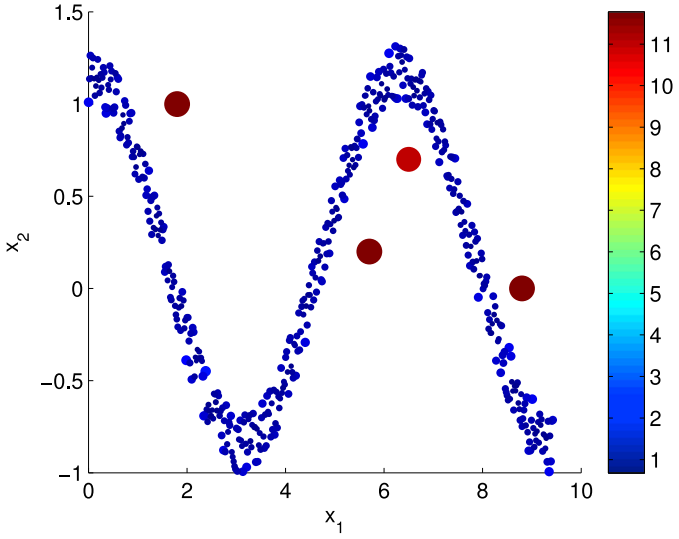


Fig. 5. Outlierness scores of all data samples, where the value of RDOS is illustrated by the color and the radius of the circle.

where $w \sim \mathcal{N}(0, \sigma^2)$. In our simulation, we use $\sigma^2 = 0.1$, and generate four outliers in this data set, as shown in Fig. 4. The RDOS of all data samples is shown in Fig. 5, where both the color and the radius of circles indicate the value of RDOS. It is still shown that the RDOS-based method can effectively detect the outliers. Meanwhile, we show the top $n = 5$ data samples with the largest RDOS in Table 1.

Table 2
The characteristics of 14 data sets

Dataset	# of features	# of outliers	# of data
Glass	7	9	205
Ionosphere	32	126	225
KDDCup99	40	200	47,913
Lymphography	18	6	142
PenDigits	16	20	9868
Shuttle	9	13	1000
Waveform	21	100	3343
WBC	30	10	357
WDBC	36	75	5025
HeartDisease	13	3	150
PageBlocks	10	99	4883
Parkinson	22	2	48
Pima	8	10	500
Wilt	5	93	4562

4.2. Real-life data sets

We also conduct outlier detection experiments on real-life data sets which have been previously used in research literature to evaluate the performance of various outlier detection algorithms. All of these data sets are from the UCI repository [24]. Particularly, we use the same versions of data sets in [25], which are all publicly available. Two types of datasets are prepared in [25]: datasets used in the outlier detection literature and datasets with semantically meaningful outliers, and by applying various preprocessing techniques, many variants of these datasets can be developed. In our experiments, we use both types of datasets for performance evaluation, in which the following two preprocessing steps are considered: *normalization* which normalizes each individual attribute to the range from 0 to 1 and *transformation* that converts all categorical attributes to numerical ones using *inverse document frequency* (IDF). We summarize the characteristics of all those data sets in Table 2, in which the first 9 datasets are commonly used in the outlier detection literature and the last 5 datasets include semantically meaningful outliers. Most of these 14 datasets are originally used for the evaluation of classification methods. For the purpose of outlier detection, one or more classes are considered as the outliers. We refer to the details of these datasets preparation in [25]. To illustrate normal data and outlier distributions, in Fig. 6, we also show the first two principle components of four selected data sets, where the normal data are denoted by blue solid circles and outliers are denoted by red circles. In these two dimensional data distributions, it appears that outliers are mostly different from the normal data.

Since all these data sets are highly imbalanced, the use of overall accuracy might be inappropriate [33,34]. Although many evaluation metrics have been proposed, such as precision, recall, precision at n, the most effective and popular one in the literature on outlier detection is the well known Receiver Operator Characteristic (ROC) curve [24]. A ROC curve can be obtained by evaluating all possible thresholds for decision, which shows how the number of correctly classified positive samples (outliers), called true positive, varies with the number of incorrectly classified negative samples (normal samples), called false positive. A random outlier detection (i.e., the worst outlier detection method) would lead to a diagonal ROC curve, and the best outlier detection method can achieve the highest rate of true positive (i.e., 100%) for any rate of false positive. One ROC curve can be also summarized by a single metric, termed Area Under the ROC Curve (AUC), which measures the area under the ROC curve. It ranges from 0 to 1. A perfect outlier detection method has an AUC of 1, while a random detection method produces an AUC of 0.5. The AUC metric has been widely used for the performance evaluation of unsupervised

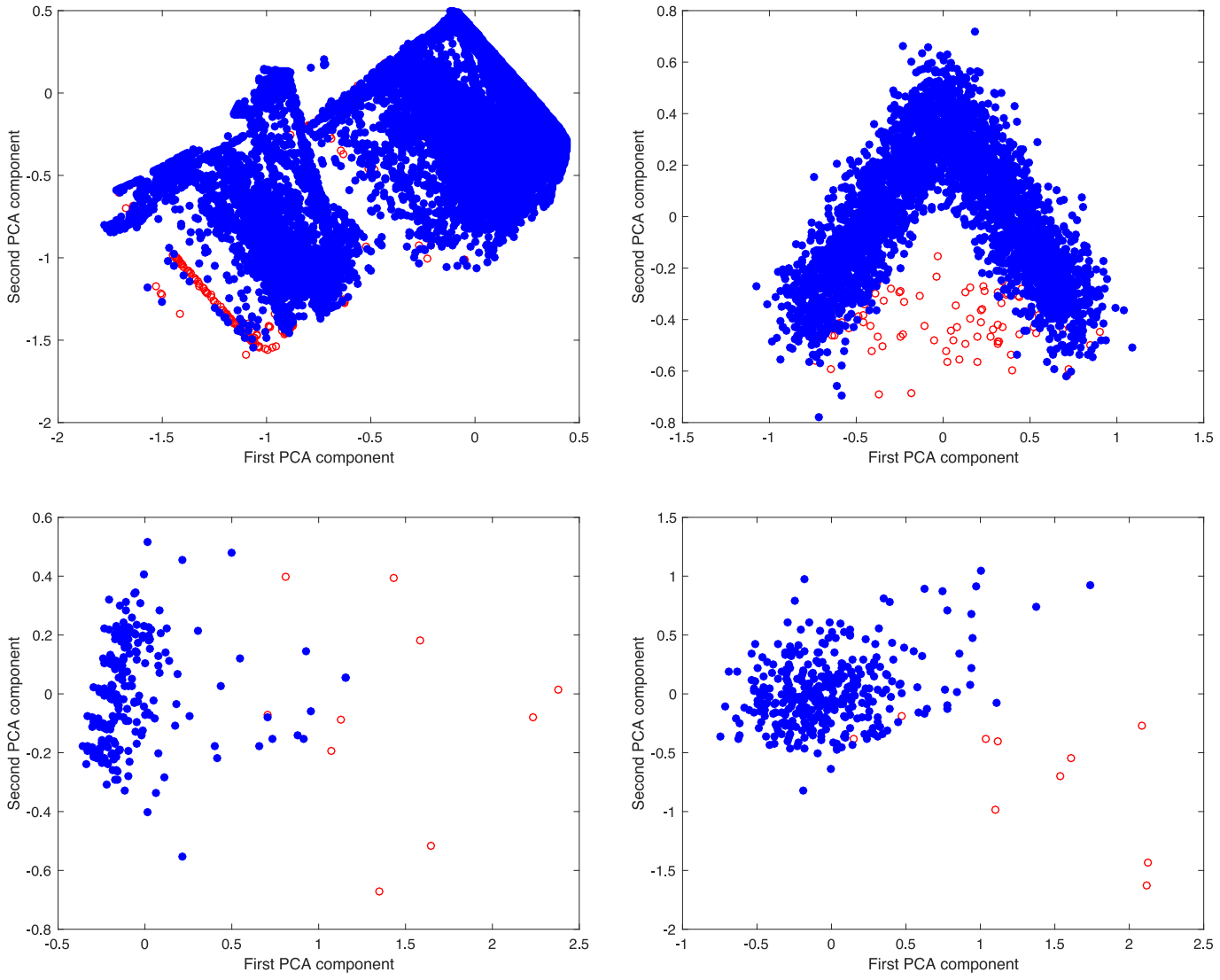


Fig. 6. Normal data and outliers in four selected real-life data sets: (A) KDDCup99, (B) Waveform, (C) WBC, and (D) WDBC. Only the first two principle components are shown. The blue solid circle denotes normal data, and the red circle denotes outlier. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

outlier detection methods. Although many other measures such as recall and precision exist, the AUC measure has been demonstrated to be the most effective one for outlier detection [25].

We compare our RDOS approach with another five widely used outlier detection approaches: Outlier Detection using In-degree Number (ODIN) [26], LOF [10], INFLO [13], Mutual Nearest Neighbors (MNN) [9], and KDEOS [17]. Since all of these examined methods are nearest neighbors-based methods, we evaluate the outlier detection performance with different k values which ranges from 1 to 100 with the step of 2. Both KDEOS and our RDOS are two approaches based on kernel density estimation, and we apply the fast optimal bandwidth selection method [27] to select the bandwidth for each dataset. We summarize the AUC results on all 14 data sets in Figs. 7–9, from which one can see that our proposed RDOS approach exhibits superior detection performance with the metric of AUC of ROC. Particularly, the RDOS approach achieves the best performance for 11 datasets, including Glass, KDDCup99, Lymphography, PenDigits, Shuttle, WBC, Heart-Disease, Parkinson, Pima, and Wilt. For other three datasets, including Waveform, WDBC, and PageBlocks, the RDOS approach shows competitive performance which is slightly worse than the LOF. One observation that can be found in our experiments is that RDOS > LOF > INFLO > ODIN > MNN > KDEOS is generally true, where the symbol "> " means "performs better than". For the large scale dataset, like KDDCup99, the AUC performance of all compared methods has a jump at $k = 83$, and our RDOS achieves the best performance which is greatly increased from 0.68 to 0.94.

5. Conclusions

This paper presented a novel outlier detection method based on local kernel density estimation. Instead of only considering the k nearest neighbors of a data sample, we used three types of neighbors: k nearest neighbors, reverse nearest neighbors, and shared nearest neighbors, for local kernel density estimation. A simple and effective relative density calculation, termed Relative Density-based Outlier Score (RDOS), was introduced to measure the outlierness. We further derived theoretical properties of the proposed RDOS measure, including the expected value and the false alarm probability. The theoretical results suggest parameter settings for practical applications. Simulation results on both synthetic data

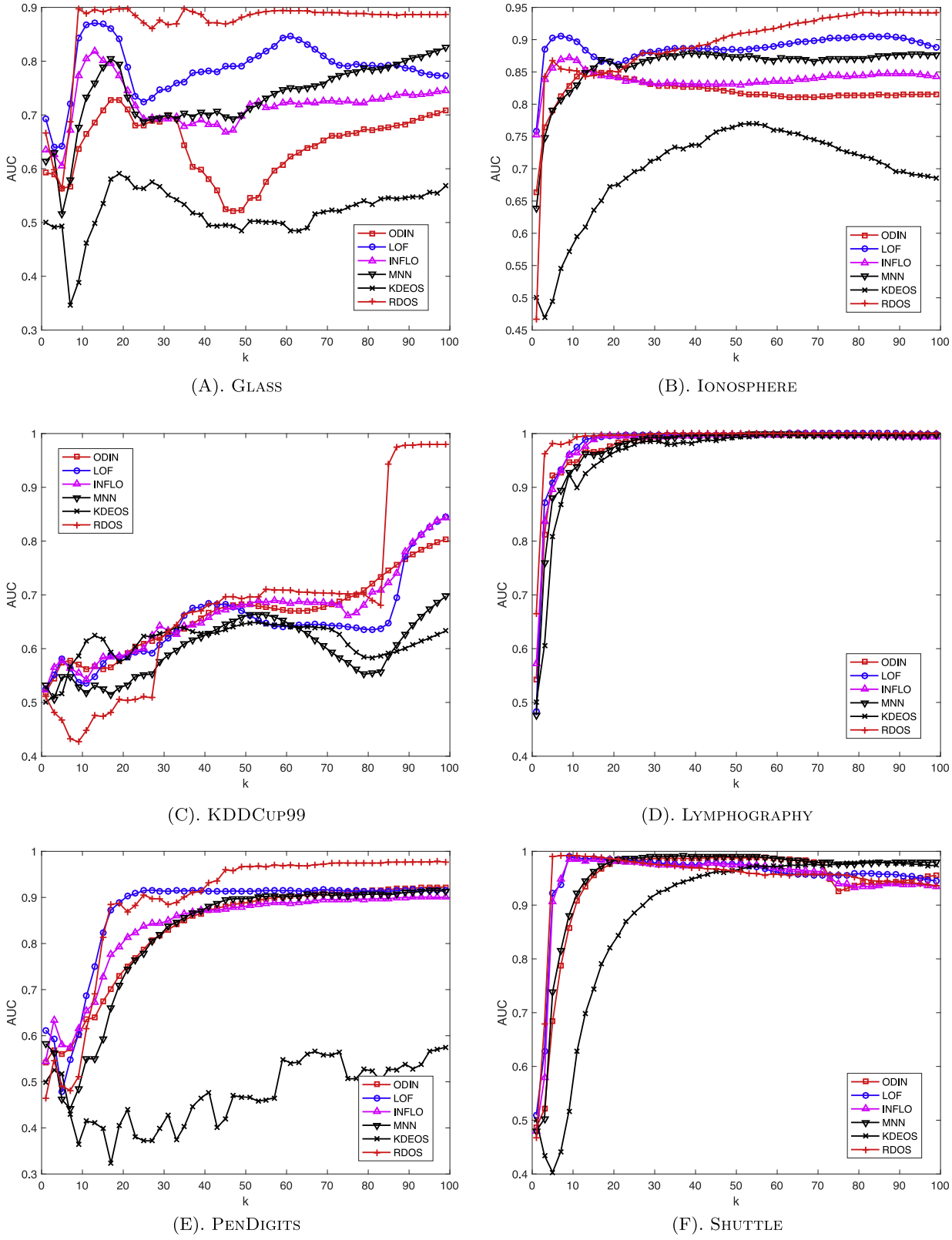
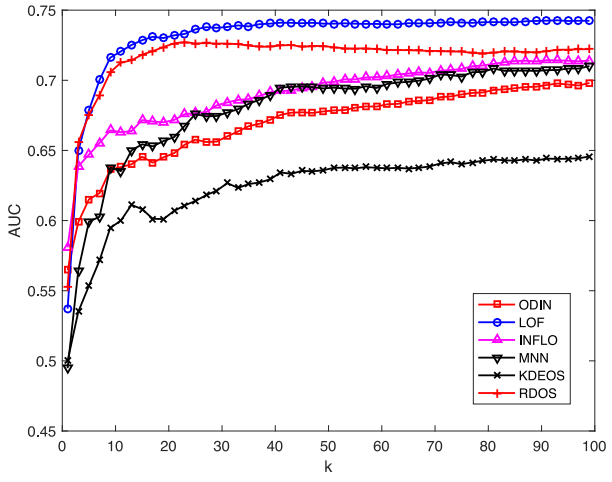


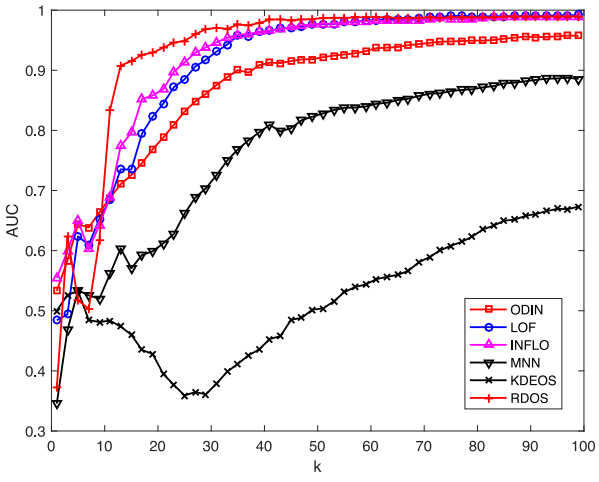
Fig. 7. AUC of all compared outlier detection algorithms for (A). Ionosphere, (B). KDDCup99, (C). Lymphography, (D). PenDigits, (E). Shuttle, and (F). Waveform.

sets and real-life data sets illustrate superior performance of our proposed method.

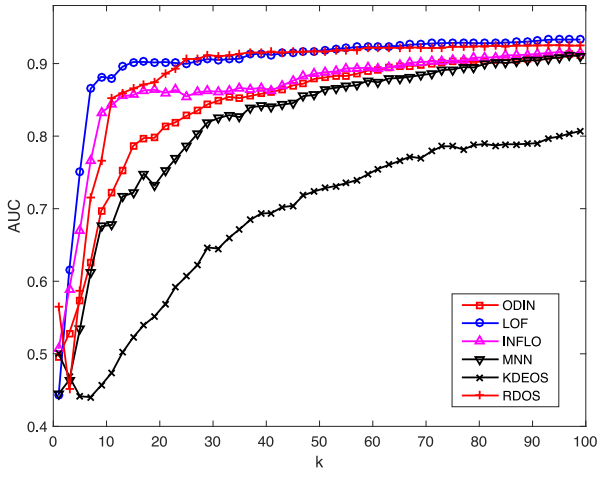
In our future research, we plan to continue exploring our proposed approach in the following three aspects: first, we would like to incorporate other distance measures into our proposed approach, other than the Euclidean distance metric used in this paper. Second, we are interested in other new distribution estimation methods, such as exponentially embedded families [28–30], which could be used to replace the kernel density estimation. At last, we will look for many other real-life applications of our proposed



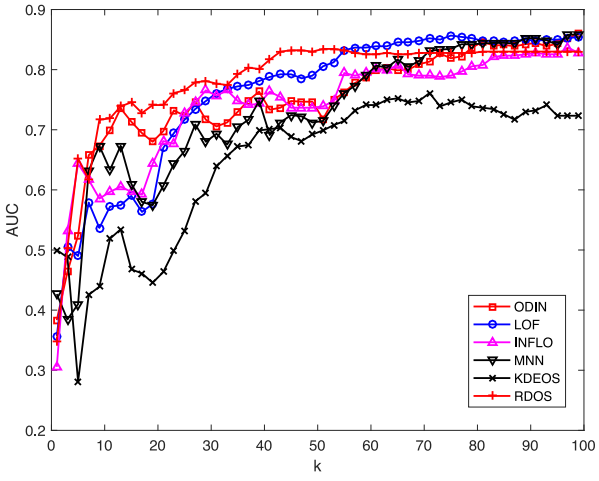
(A). Waveform



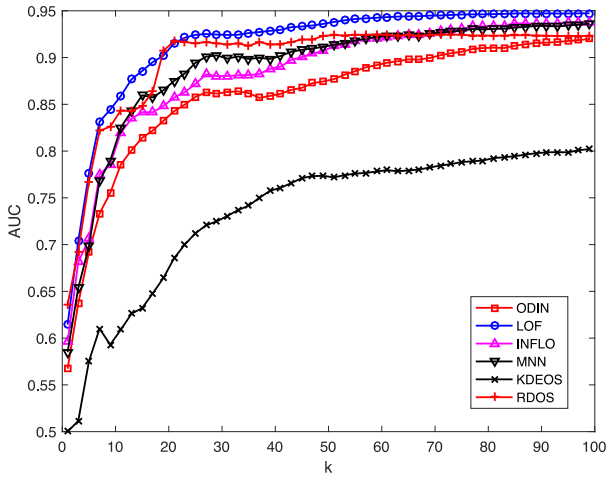
(B). WBC



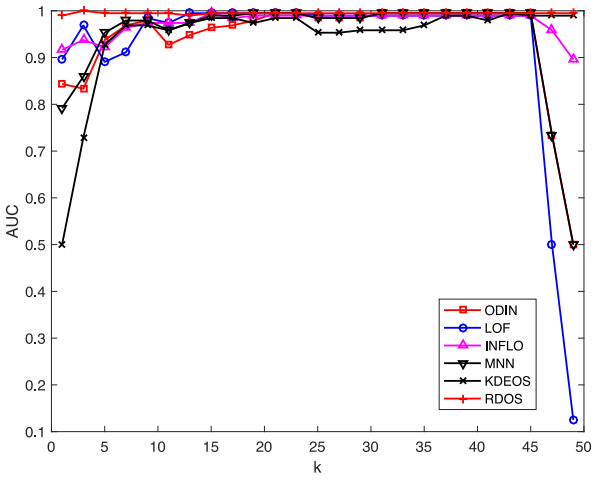
(C). WDBC



(D). HeartDisease



(E). PageBlocks



(F). Parkinson

Fig. 8. AUC of all compared outlier detection algorithms for (A). Waveform, (B). WBC, (C). WDBC, (D). HeartDisease, (E). PageBlocks, and (F). Parkinson.

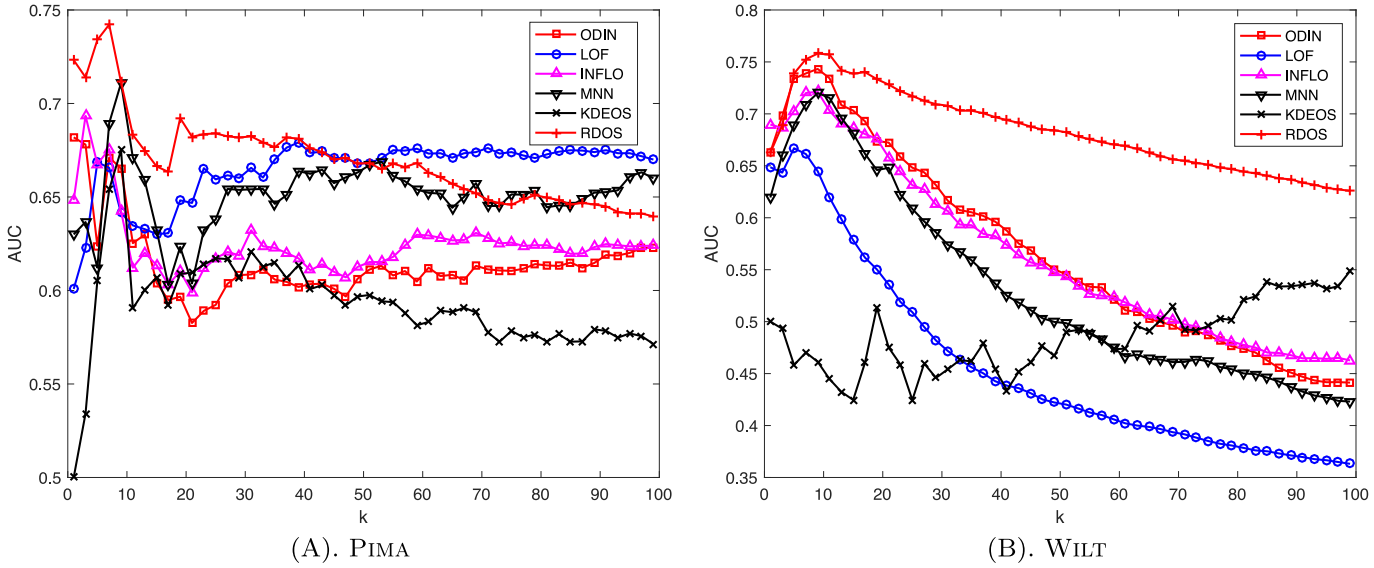


Fig. 9. AUC of all compared outlier detection algorithms for (A). Pima and (B). Wilt.

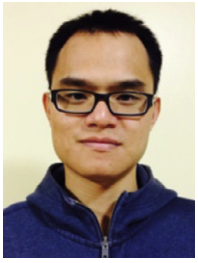
approach, such as intrusion detection [31], false data injection detection in smart grid [32], among others.

Acknowledgments

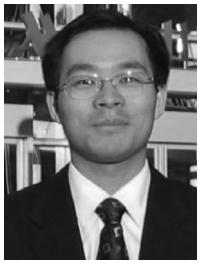
This research was partially supported by National Science Foundation (NSF) under grant ECCS 1053717 and CCF 1439011, and the Army Research Office under grant W911NF-12-1-0378.

References

[1] W. Jin, A.K. Tung, J. Han, Mining top-n local outliers in large databases, in: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2001, pp. 293–298.
[2] V. Hautamaki, I. Karkkainen, P. Franti, Outlier detection using k-nearest neighbour graph, in: Proceedings of the 17th International Conference on Pattern Recognition, vol. 3, 2004, pp. 430–433.
[3] V. Barnett, T. Lewis, Outliers in Statistical Data, vol. 3, Wiley, New York, 1994.
[4] E.M. Knox, R.T. Ng, Algorithms for mining distance based outliers in large datasets, in: Proceedings of the International Conference on Very Large Data Bases, 1998, pp. 392–403.
[5] C.C. Aggarwal, P.S. Yu, Outlier detection for high dimensional data, in: Proceedings of the ACM Sigmod Record, vol. 30, 2001, pp. 37–46.
[6] S. Ramaswamy, R. Rastogi, K. Shim, Efficient algorithms for mining outliers from large data sets, in: Proceedings of the ACM SIGMOD Record, vol. 29, 2000, pp. 427–438.
[7] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise., in: Proceedings of KDD, vol. 96, 1996, pp. 226–231.
[8] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: a new data clustering algorithm and its applications, Data Min. Knowl. Disc. 1 (2) (1997) 141–182.
[9] M. Brito, E. Chavez, A. Quiroz, J. Yukich, Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection, Stat. Probab. Lett. 35 (1) (1997) 33–42.
[10] M.M. Breunig, H.-P. Kriegel, R.T. Ng, J. Sander, LOF: identifying density-based local outliers, in: Proceedings of ACM Sigmod Record, vol. 29, 2000, pp. 93–104.
[11] J. Tang, Z. Chen, A.W.-C. Fu, D.W. Cheung, Enhancing effectiveness of outlier detections for low density patterns, in: Advances inKnowledge Discovery and Data Mining, Springer Berlin Heidelberg, 2002, pp. 535–548.
[12] K. Zhang, M. Hutter, H. Jin, A new local distance-based outlier detection approach for scattered real-world data, in: Advances in Knowledge Discovery and Data Mining, Springer Berlin Heidelberg, 2009, pp. 813–822.
[13] W. Jin, A.K. Tung, J. Han, W. Wang, Ranking outliers using symmetric neighborhood relationship, in: Advances in Knowledge Discovery and Data Mining, Springer Berlin Heidelberg, 2006, pp. 577–593.
[14] J. Tang, Z. Chen, A.W.-c. Fu, D. Cheung, A robust outlier detection scheme for large data sets, in: Proceedings of the 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2001.
[15] L.J. Latecki, A. Lazarevic, D. Pokrajac, Outlier detection with kernel density functions, in: Proceedings of the International Conference on Machine Learning and Data Mining in Pattern Recognition, 2007, pp. 61–75.
[16] J. Gao, W. Hu, Z.M. Zhang, X. Zhang, O. Wu, RKOF: robust kernel-based local outlier detection, in: Advances in Knowledge Discovery and Data Mining, Springer Berlin Heidelberg, 2011, pp. 270–283.
[17] E. Schubert, A. Zimek, H.-P. Kriegel, Generalized outlier detection with flexible kernel density estimates, in: Proceedings of the 14th SIAM International Conference on Data Mining (SDM), Philadelphia, PA, 2014, pp. 542–550.
[18] B. Tang, H. He, ENN: extended nearest neighbor method for pattern recognition [research frontier], IEEE Comput. Intell. Mag. 10 (3) (2015) 52–60.
[19] V.A. Epanechnikov, Non-parametric estimation of a multivariate probability density, Theory Probab. Appl. 14 (1) (1969) 153–158.
[20] B. Tang, H. He, KernelADASYN: kernel based adaptive synthetic data generation for imbalanced learning, in: Proceedings of the IEEE Congress on Evolutionary Computation, IEEE, 2015, pp. 664–671.
[21] Q. Zhu, J. Feng, J. Huang, Natural neighbor: a self-adaptive neighborhood method without parameter k, Pattern Recognit. Lett. 80 (2016) 30–36.
[22] J.L. Bentley, Multidimensional binary search trees used for associative searching, Commun. ACM 18 (9) (1975) 509–517.
[23] S.G. Steckley, Estimating the density of a conditional expectation, Cornell University, 2006 (Ph.D. thesis).
[24] M. Lichman, UCI Machine Learning Repository, School of Information and Computer Science, University of California, Irvine, CA, 2013. http://archive.ics.uci.edu/ml/
[25] G.O. Campos, A. Zimek, J. Sander, R.J. Campello, B. Micenková, E. Schubert, I. Assent, M.E. Houle, On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study, Data Min. Knowl. Disc. 30 (4) (2015) 1–37.
[26] V. Hautamaki, I. Karkkainen, P. Franti, Outlier detection using k-nearest neighbour graph, in: Proceedings of the 17th International Conference on Pattern Recognition, 2004, pp. 430–433.
[27] V.C. Raykar, R. Duraiswami, Fast optimal bandwidth selection for kernel density estimation., in: Proceedings of the SIAM International Conference on Data Mining, 2006, pp. 524–528.
[28] B. Tang, H. He, Q. Ding, S. Kay, A parametric classification rule based on the exponentially embedded family, IEEE Trans. Neural Netw. Learn. Syst. 26 (2) (2015) 367–377.
[29] B. Tang, S. Kay, H. He, P.M. Baggenstoss, EEF: exponentially embedded families with class-specific features for classification, IEEE Signal Process. Lett. 23 (7) (2016a) 969–973.
[30] B. Tang, S. Kay, H. He, Toward optimal feature selection in naive Bayes for text categorization, IEEE Trans. Knowl. Data Eng. 28 (9) (2016b) 2508–2521.
[31] R. Mitchell, I.-R. Chen, A survey of intrusion detection techniques for cyber–physical systems, ACM Comput. Surv. (CSUR) 46 (4) (2014) 55.
[32] B. Tang, J. Yan, S. Kay, H. He, Detection of false data injection attacks in smart grid under colored Gaussian noise, in: IEEE Conference on Communications and Network Security (CNS), 2016, pp. 172–179.
[33] H. He, E.A. Garcia, Learning from Imbalanced Data, IEEE Trans. Knowledge and Data Engineering 21 (9) (2009) 1263–1284.
[34] H. He, Y. Bai, E.A. Garcia, S. Li, ADASYN: Adaptive synthetic sampling approach for imbalanced learning, in: Proc. Int. Joint Conf. Neural Networks (IJCNN'08), 2008, pp. 1322–1328.



Bo Tang is currently an Assistant Professor in the Department of Computer Science at Hofstra University, Hempstead, NY, USA. He received his Ph.D. degree in Electrical Engineering from University of Rhode Island. His current research interests include statistical machine learning, computational intelligence, computer vision, and robotics.



Haibo He is currently the Robert Haas Endowed Professor of Electrical Engineering with the University of Rhode Island, Kingston, RI, USA. His current research interests include machine learning, cyber–physical systems, computational intelligence, hardware design for machine intelligence, and various applications such as smart grid and renewable energy systems.