Efficient Outlier Detection for High-Dimensional Data

Huawen Liu, *Member, IEEE*, Xuelong Li, *Fellow, IEEE*, Jiuyong Li, *Member, IEEE*, and Shichao Zhang, *Senior Member, IEEE*

***Abstract*—How to tackle high dimensionality of data effectively and efficiently is still a challenging issue in machine learning. Identifying anomalous objects from given data has a broad range of real-world applications. Although many classical outlier detection or ranking algorithms have been witnessed during the past years, the high-dimensional problem, as well as the size of neighborhood, in outlier detection have not yet attracted sufficient attention. The former may trigger the distance concentration problem that the distances of observations in high-dimensional space tend to be indiscernible, whereas the latter requires appropriate values for parameters, making models high complex and more sensitive. To partially circumvent these problems, especially the high dimensionality, we introduce a concept called local projection score (LPS) to represent deviation degree of an observation to its neighbors. The LPS is obtained from the neighborhood information by the technique of low-rank approximation. The observation with high LPS is a promising candidate of outlier in high probability. Based on this notion, we propose an efficient and effective outlier detection algorithm, which is also robust to the parameter k of k nearest neighbors. Extensive evaluation experiments conducted on twelve public real-world data sets with five popular outlier detection algorithms show that the performance of the proposed method is competitive and promising.**

***Index Terms*—Dimension reduction, high-dimensional data, k nearest neighbors (kNN), low-rank approximation, outlier detection.**

Manuscript received May 2, 2017; accepted June 15, 2017. Date of publication July 7, 2017; date of current version November 15, 2018. This work was supported in part by the China 973 Program under Grant 2013CB329404, in part by the National Natural Science Foundation of China under Grant 61572443, Grant 61450001, Grant 61672177, Grant 61761130079, in part by the National Key Research and Development Program of China under Grant 2016YFB1000905, in part by the Key Research Program of the Chinese Academy of Sciences under Grant KGZD-EW-T03, in part by the ARC Discovery under Grant DP130104090, and in part by the Shanghai Key Laboratory of Intelligent Information Processing under Grant IIPL-2016-001. This paper was recommended by Associate Editor J. Wu. *(Corresponding author: Huawen Liu.)*

H. Liu is with the Department of Computer Science, Zhejiang Normal University, Jinhua 321004, China (e-mail: hwliu@zjnu.edu.cn).

X. Li is with the School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: xuelong_li@nwpu.edu.cn).

J. Li is with the School of Information Technology and Mathematical Sciences, University of South Australia, Adelaide, SA 5095, Australia (e-mail: Jiuyong.li@unisa.edu.au).

S. Zhang is with the Department of Computer Science, Guangxi Normal University, Guilin 541004, China (e-mail: zhangsc@gxnu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2017.2718220

I. Introduction

With the advancement of emerging technologies, an increasing amount of data is becoming available in real-world applications. Within the massive data, some of them induce abnormal behaviors or patterns raised from a variety of aspects including malfunctional hardware or malicious activities. Such exceptional behaviors or inconsistent patterns, also known as outliers, anomalies, abnormalities, novelties, or deviants, do not comply with a well-defined notion of normal behavior of the data [1], [2]. In reality, they often exhibit as the representations of noises or interesting facts, such as cyber-intrusion and terrorist activities, according to different purposes [3].

Identifying outliers out from data is of great interest to the communities of machine learning and data mining, because it can reveal unusual behaviors, interesting patterns, and exceptional events from data. Indeed, identifying or eliminating outliers becomes an essential preprocessing stage in data analysis [4]. For example, noise removal can improve model performance, due to the fact that noises may disturb the discovery of important information, while anomalous access detection by examining access records in a firewall at a time can help us to isolate intrusion from network access.

Outlier detection (also known as anomaly detection) is a process of unveiling unexpected observations that deviates so much from the rest of the observations [5]. Since outlier detection can bring significant benefits to decision analysis, it has gained considerable interests in a variety of fields and applied in a large number of domains, such as crime and terrorist detection [6], fault debugging and diagnosis [7], network intrusion, fraud discovery, medical and health monitoring, signal analysis, image processing, abnormal weather detection, anomalous crowd behavior estimation, video surveillance, and many other areas [1], [2], [8]–[10]. The broad diversity in real-world applications reflects a fact that outlier detection is a widely researched topic.

A large body of outlier detection methods have been developed. Technically, the procedure of detecting outliers consists of two main stages: 1) outlier ranking and 2) determining, where the former offers a ranking list of the observations, each one with a score, based on given metrics. The observations with high scores rank on the top of the list, if a larger value stands for a greater variant or anomalous degree. The latter determines outliers according to the ranking list. From this perspective, outlier ranking plays a core role in

detection. Of these, outlier ranking and outlier detection are two terms used most commonly in the literature; sometimes interchangeably. The outlier detection algorithms can be roughly categorized into the following groups, such as statistics-based, distance-based, density-based and clustering-based methods [1]. Among these detection methods, the distance-based and density-based detection ones has received special attraction and extensively studied, due to the fact that their notions are intuitive and can be easily implemented. The rank-based detection algorithm (RBDA) [11] and the local outlier factor (LOF) [12] are typical examples of these two kinds, respectively.

There are two main challenges needed to be further investigated for outlier detection. The first one is the high dimensionality of data. The high dimensionality may raise two pervasive problems: 1) the so-called curse of dimensionality and 2) the distance concentration [13]. The former refers to the fact that the size of observations grows exponentially with the number of dimensions, making the data sparse, while the latter indicates that the distance or density metrics fail to capture the neighborhood information, because all distances between observations tend to become indiscernible as the dimensionality increases. A common strategy for the high-dimensional problem in machine learning is dimension reduction. For instance, Kasun *et al.* [14] developed an efficient dimension reduction method by using extreme learning machine. However, distinguishing outliers in a high-dimensional space from normal observations efficiently and effectively is still difficult. The second challenge is that albeit their popularity, the distance-based detection methods only concern global information and their performance depends on the size of neighborhood, while the density-based ones are sensitive to parameters defining the neighborhood [1], [5].

In this paper, we make an attempt to address the above problems by developing a novel, yet effective learning method for outlier detection. The proposed method is motivated by the simple notion that anomalous observations have higher variances, and deviate from others greatly within the same neighborhood information. To capture the degree of deviation, a new metric called local projection score (LPS) is introduced. It is mainly used to measure the degree of deviation of each observation to the corresponding neighbors which are projected into a low-dimensional space by dimension reduction. It should be pointed out that LPS not only takes local information into account but also can handle high-dimensional data without specifical requirements on the dimensionality. This enables us to offer a guideline for ranking and determining outliers, where the observation having a large LPS value is therefore a potential outlier with a high probability. Specifically, our method starts to identify k nearest neighbors (kNNs) for each observation. The neighborhood information is then projected into a low-dimensional space via the technique of low-rank matrix approximation to estimate LPS of the observation. Subsequently, all the observations are ranked in a descending order according to their scores. Finally, the observations with high scores are picked out and taken as outliers.

The main contributions of this paper are highlighted as follows.

1) We propose a novel and effective outlier detection method, which is capable of handling high-dimensional data and robust to the parameter k of kNN.
2) Our method adopts a new anomalous score called LPS to capture the deviation degree of an observation to its neighbors. The score is consistent with the nuclear norm of the neighborhood.
3) To obtain the anomalous score for each instance, the technique of low-rank approximation is exploited. It aims at projecting the high-dimensional neighborhood into a low-dimensional space.

The rest of this paper is organized as follows. In Section II, we briefly review previous research work on outlier detection. Section III provides basic concepts used and the proposed outlier detection method. The experimental results of our method with the comparing algorithms on real data sets are discussed in Section IV, followed by the conclusions of this paper in Section V.

II. Related Work

Over time, a rich number of outlier detection algorithms have been witnessed in several research communities. In this section, only the latest work for outlier detection is reviewed. More details can be found from good survey papers (see [1], [2], [5]) and references therein.

The outlier detection techniques can be divided into different categories, depending on criteria used. For example, like the categorization of machine learning algorithms, the outlier detection methods can be roughly classified into supervised, semi-supervised, and unsupervised scenarios according to the availability or unavailability of data labels [1]. The supervised methods concern the data objects tagged with either normal or abnormal labels, while in the semi-supervised context only normal objects are labeled. For the unsupervised techniques, the label information of data is unavailable. Since obtaining the label information in reality is very expensive, the unsupervised techniques are more widely applicable than the supervised ones.

With the techniques adopted, the outlier detection algorithms can be roughly classified as statistics-based, clustering-based methods, distance-based, density-based, and so on [1]. The statistics-based detection methods, also named as model-based methods, exploit the statistical property of data, i.e., the normal observations can fit a statistical model well, while the abnormal can not, to identify outliers [15], [16]. Most of earlier studies belong to this kind. However, the underlying assumption often does not hold true, especially, for high-dimensional data in reality.

The clustering-based detection methods adopt the off-the-shelf techniques of clustering to identify outliers from given data, where the observations that do not belong to or close to any dense or large clusters are regarded to be outliers [17]. To some extent, the outliers are by-products of clustering. Note that the clustering techniques are fundamentally different to outlier detection, since the purpose of clustering is to identify

clusters, not detect outliers. Thus, the efficiency and effectiveness of detecting outliers with the clustering techniques are relatively low.

The basic idea of the distance-based detection algorithms is that an observation is likely regarded as an outlier if it is far from its nearest neighbors. Essentially, this kind of work first gets nearest neighbors for each observation, and then estimates the distance of the observation to its neighbors. Subsequently, the distances for all observations are ranked, and the observations with larger distances are regarded as outliers. Typical examples of such kind include RBDA [11] and the local distance-based outlier factor (LDOF) [18]. Since no assumption about the distribution of data is required, the distance-based detection algorithms are extensively studied.

The performance of the distance-based detection methods greatly relies on the definition of distance and the search efficiency of nearest neighbors. Several attempts have been made on these aspects. In [19], the distance of an observation to its kNNs, i.e., the maximal distance, was used, while in [20] the sum of the distances of an observation to its k neighbors was calculated. Ha *et al.* [4] estimated the distance to the center of gravity, which represents a geometric property of an observation, of neighbors. Koufakou and Georgiopoulos [3] discussed the distance definition on mixed type features of data. Liu and Deng [21] extended the classical LOF to uncertain data. Wang *et al.* [22] exploited a minimum spanning tree to improve the searching efficiency of neighbors in kNN. Other work focuses on using of variants of kNN to improve the performance. Recently, Huang *et al.* [23] adopted the notion of natural neighbor to obtain the neighborhood information, while Radovanović *et al.* [13] employed reverse nearest neighbors, rather than nearest neighbors, to determine outliers.

A main challenge for the distance-like detection algorithms is the high dimensionality of data [5], where the data is often sparsely distributed and similar to each other, resulting in the differences of the actual distances for many pairs of observations are small [2]. In other words, the distance of an observation to its nearest neighbor is close to the distance to its farthest neighbor as the dimensionality increases. Hence, the discriminative effect of the distance-based techniques can not be observed clearly, especially, in situations where the data from a mixture of distributions have various degrees of cluster density. In the literature, subspace learning [5], random sampling [24], and feature selection [25], [26] are three frequently used strategies to alleviate this problem. Additionally, since the distances are calculated in terms of global information, instead of local one, another problem of the distance-based methods is that they can only identify global outliers and fail for local ones.

The underlying principle of the density-based detection approaches is that an outlier lies in a neighborhood with low density, while a normal observation has a dense neighborhood. Specifically, they first estimate the density of the neighborhood for each observation, and then it is compared with that of the densities of the neighbors of the observation. If the density considerably differs from that around its neighbors, the observation can be declared an outlier. Due to its effectiveness and simplification, this kind of detection approaches has been widely used in reality. LOF [12] is a representative example of the density-based detection methods. It takes use of a local density, estimated on reachable distance of kNNs, to measure a degree of being outlier. After LOF was introduced, several variants of LOF have been developed, including the connectivity-based outlier factor [27], the influenced outlierness [28], the local correlation integral [29], and the local outlier probabilities [30]. Notwithstanding they are very popular in reality, like the distance-based methods, the density-based ones are also based on distance computations. They also encounter the same challenging problems raised from the high dimensionality of data, where outlier scores are close to each other. To alleviate this problem, the high contrast subspace (HiCS) [31] evaluates and ranks outlier by using HiCSs. However, the density-based methods are sensitive to parameters used to determine the size of the neighborhood to be examined, and show poor performance when the observations have a variety of densities.

III. Outlier Detection With Local Projection

A. Low-Rank Approximation

How to deal with high-dimensional data is a still challenging issue in the community of machine learning. A frequently used solution in reality is to perform dimension reduction, which projects a high-dimensional space into a low-dimensional one by mapping techniques. There are several classical dimension reduction techniques available, such as principle component analysis, extreme learning machine [14], and linear discriminant anaysis [32]. A popular reduction method is low-rank matrix approximation, which seeks a reduced rank matrix to approximately represent the original one [33].

Matrix rank is a fundamental and important concept in linear algebra. It refers to the number of leading entries which correspond to linearly independent rows or columns of the matrix. On the other hand, the rank is the number of nonzero singular values of the matrix. Suppose that the high-dimensional data are arranged as the columns of a large matrix $\mathbf{D} \in \mathbb{R}^{n\times m}$, where n and m denote the numbers of observations and features (variables), respectively. Considering the technique of singular value decomposition (SVD), $\mathbf{D}$ can be decomposed as follows:

$$\mathbf{D} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{1}$$

where $\mathbf{U} \in \mathbb{R}^{n\times r}$ and $\mathbf{V} \in \mathbb{R}^{m\times r}$ are left and right singular vectors, respectively. $\mathbf{S} \in \mathbb{R}^{r\times r}$ is the diagonal matrix consisting of singular values of $\mathbf{D}$, i.e., $\mathbf{S} = \text{diag}\{\sigma_1, \sigma_2, \ldots, \sigma_r, 0, 0, \ldots, 0\}$, where the singular values are sorted in decreasing order, $\sigma_1 \geq \ldots \geq \sigma_r > 0$. Thus, the rank of $\mathbf{D}$ is r, i.e., $\text{rank}(\mathbf{D}) = r$ and $r \leq \min\{n, m\}$.

The rank is an effective tool to measure the sparsity of matrix. The lower the rank, the more sparse the matrix. In real-world applications, data are often generated from low-dimensional spaces. Thus, the ranks of the corresponding data matrices are low. However, noises raised from a variety of

aspects lead to the data matrices with high rank. Hence, it is necessary to remove noises from the high-dimensional data matrices and recover the matrices with low rank for data analysis.

Low-rank approximation is a versatile technique to represent and recover data within low-dimensional subspaces from high-dimensional ones. It aims to minimize the matrix discrepancy between a high-dimensional data matrix $\mathbf{D}$ and its reduced matrix $\bar{\mathbf{D}}$, i.e., seeking a low rank matrix $\bar{\mathbf{D}}$ for $\mathbf{D}$. Formally, the mathematical model of low-rank approximation is to find the matrix $\bar{\mathbf{D}}$ within a low-dimensional subspace, such that the following constraint is minimized [34]:

$$\begin{aligned} \min_{\bar{\mathbf{D}}} \quad & \left\|\mathbf{D}-\bar{\mathbf{D}}\right\|_F \\ \text{s.t.} \quad & \operatorname{rank}\left(\bar{\mathbf{D}}\right) \le t \end{aligned} \tag{2}$$

where $\|\mathbf{X}\|_F = \sqrt{\sum_i \sum_j x_{ij}^2}$ is the Frobenius norm of $\mathbf{X}$. For the optimization problem above, it is in general combinatorial and known to be NP-hard. Thus, making the minimization problem trackable by relaxing the constraint seems to be a feasible solution. A popular strategy is to transform (2) into the following convex optimization problem:

$$\begin{aligned} \min_{\bar{\mathbf{D}}} \quad & \left\|\mathbf{D}-\bar{\mathbf{D}}\right\|_F \\ \text{s.t.} \quad & \left\|\bar{\mathbf{D}}\right\|_* \end{aligned} \tag{3}$$

where $\|\bar{\mathbf{D}}\|_*$ is the nuclear norm (also known as trace norm) denoted as the sum of the top t singular values of $\bar{\mathbf{D}}$, i.e., $\|\bar{\mathbf{D}}\|_* = \sum_{i=1}^{t} \sigma_i$. In a sense, getting the top t singular values, rather than all of them, aims to alleviate the effects of noises and improve the robustness.

There are several effective solutions for the optimization problem of (3), such as iterative thresholding, accelerated proximal gradient, augmented Lagrange multipliers, and alternating direction methods [35], [36]. To better understand the idea, here we resort to the technique of singular value thresholding (SVT) to solve the nuclear norm minimization problem conveniently. It is noticeable that (3) has the same solution to the following optimization problem:

$$\min_{\bar{\mathbf{D}}} \quad \frac{1}{2}\|\mathbf{D}-\bar{\mathbf{D}}\|_F^2 + \lambda\|\bar{\mathbf{D}}\|_*. \tag{4}$$

For the optimization problem above, the following theorem holds [36].

Theorem 1: The solution of (4) is $\bar{\mathbf{D}}^* = f_\lambda(\mathbf{D})$, where $f_\lambda(\mathbf{D})$ is the SVT operator on $\mathbf{D}$, and $f_\lambda(\mathbf{D}) = \sum_{i=1}^{t}(\sigma_i - \lambda)_+ \mathbf{u}_i \mathbf{v}_i^T$. σ_i, $\mathbf{u}_i$ and $\mathbf{v}_i$ are the ith singular value, left singular vector and right singular vector of $\mathbf{D}$, respectively. The function $(x)_+$ is a thresholding operator on x. It is zero if $x \le 0$, otherwise x.

The above theorem serves as an important role in solving the nuclear norm minimization problems. Based on the theorem, we first exploit the technique of SVD to decompose $\mathbf{D}$ as $\mathbf{D} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, and then pick the singular values that are larger than λ, as well as the corresponding left and right singular vectors. Without loss of generality, the top t singular values are larger than λ. Afterward the obtained t singular values and the corresponding singular vectors are organized into a reduced matrix as $\bar{\mathbf{D}} = \mathbf{U}_t\mathbf{S}_t\mathbf{V}_t^T$. Consequently, the minimal error of $\mathbf{D}$ to $\bar{\mathbf{D}}$ is given by the singular values that have been zeroed out in the process as $\|\mathbf{D}-\bar{\mathbf{D}}\| = \sqrt{\sigma_{t+1}^2 + \cdots + \sigma_r^2}$, where r is the rank of $\mathbf{D}$.

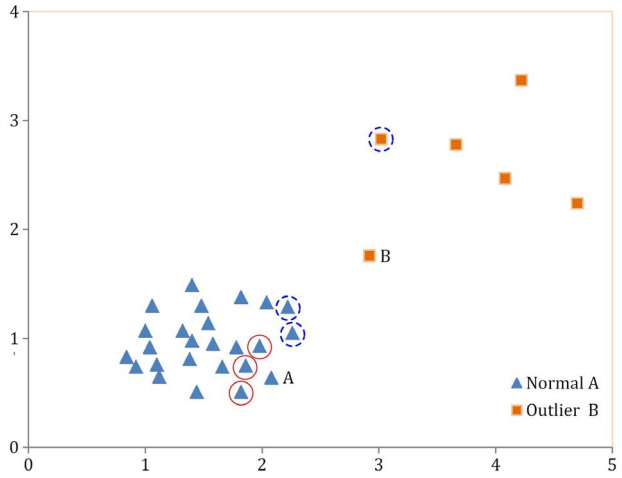

Fig. 1. Nearest neighbors of $\mathbf{A}$ and $\mathbf{B}$, where $k = 3$.

B. Local Projection Score

As discussed above, a fundamental assumption underlying the distance-based and the density-based outlier detection methods is that they exploit neighborhood information of an observation to determine whether the observation is an outlier or not. The sparser the neighborhood of the observation, the higher probability of being outlier the observation. This however, is consistent with the optimization problem of the nuclear norm mentioned above. In fact, the nuclear norm $\|\mathbf{D}\|_* = \sum_{i=1}^{r} \sigma_i$ can effectively measure the divergence (or information amount) of $\mathbf{D}$, since each singular value σ_i refers to a scale of $\mathbf{D}$ on the ith principle component, yielding the projections of $\mathbf{D}$ onto the subspace spanned by the r singular vectors of $\mathbf{D}$.

Naturally, we exploit the nuclear norm as our anomalous score to measure the divergence degree of neighborhood. Given an observation $\mathbf{x}$, its neighborhood information $\mathcal{N}(\mathbf{x})$ typically comprise nearest neighbors of $\mathbf{x}$. $\mathcal{N}(\mathbf{x})$ can be obtained by the off-the-shelf learning algorithms like kNN, i.e., $\mathcal{N}(\mathbf{x}) = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$, where $\mathbf{x}_i$ is the ith nearest neighbor of $\mathbf{x}$. For example, the nearest neighbors of $\mathbf{A}$ and $\mathbf{B}$ in Fig. 1 are those points marked with solid circles and dashed circles, respectively, if $k = 3$ is considered in kNN. It should be pointed out that for the normal points, their neighbors are close to each other tightly, whereas the outliers are far from their neighbors.

To delineate such traits of data distributions, we adopt the nuclear norm of neighborhood as our anomalous degree called LPS

$$\operatorname{lps}(\mathbf{x}) = \|\mathcal{N}(\mathbf{x})\|_*. \tag{5}$$

Basically, the larger the $\operatorname{lps}(\mathbf{x})$ is, the sparser the neighborhood of $\mathbf{x}$ is. The specifical procedure of estimating $\operatorname{lps}(\mathbf{x})$ consists of four steps, i.e., solving the unclear norm minimization

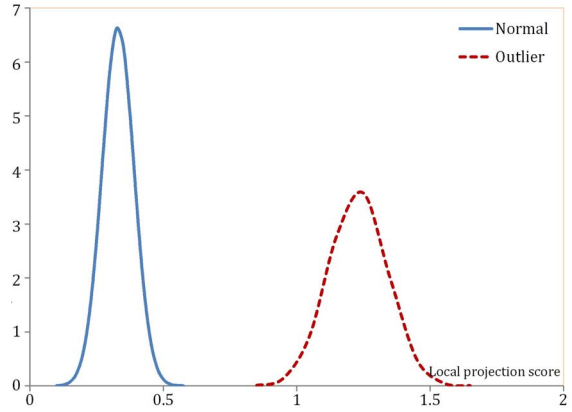

Fig. 2. Approximate probability density functions of the LPSs for the outlier and normal points.

Algorithm 1 LPOD

Input: The data collection **D**, the number of neighbors k, and the number of outlier candidates s;
Output: The s desired outliers;
Pre-processing the data collection **D**;
For each observation $\mathbf{x}\in\mathbf{D}$
Obtaining the nearest neighbors $\mathcal{N}(\mathbf{x})$ of $\mathbf{x}$ via kNN;
Solving Eq. (4) on $\mathcal{N}(\mathbf{x})$ to extract principle components;
Projecting $\mathcal{N}(\mathbf{x})$ into the desired low-dimensional subspace;
Calculating $lps(\mathbf{x})$ for $\mathbf{x}$ according to Eq. (5);
End For
Sorting local projection scores in a descending order;
Returning top s observations as desired outliers;

problem of (4); projecting $\mathcal{N}(\mathbf{x})$ into the low-dimensional subspace; obtaining the singular values and returning their sum as the final result. Since $\mathcal{N}(\mathbf{x})$ usually contains less observations, but with a moderate number of features, it is necessary to project the neighborhood information into a low-dimensional subspace. Performing dimension reduction aims at obtaining a reliable distribution of neighborhood for $\mathbf{x}$ and relieving the effects of noises within $\mathcal{N}(\mathbf{x})$. On the basis of the projection, the anomalous degree $\text{lps}(\mathbf{x})$, i.e., the unclear norm of $\mathcal{N}(\mathbf{x})$, is derived to measure the deviation degree of $\mathbf{x}$ to its neighborhood $\mathcal{N}(\mathbf{x})$.

To better understand this idea, let us revisit the example of Fig. 1 above. For each (outlier or normal) point, we calculated its LPS based on its neighborhood information. The mean LPSs of the outlier and normal points are 1.226 and 0.335, respectively. Their probability density functions are approximately estimated and presented in Fig. 2. According to the results, we know that the approximate probability density function of the outliers is significantly different to that of the normal points. From this view, the LPS is an effective measurement to distinguish the outliers from the normal points.

C. Local Projection-Based Outlier Detection

Based on the analyses above, we propose a novel outlier detection method called local projection-based outlier detection (LPOD). The central idea of LPOD is the divergence degree of neighborhood after projected into a low-dimensional subspace. The implementation details of our detection method is summarized in Algorithm 1. It comprises two major stages: 1) estimating LPSs and 2) determining outliers according to the scores. Within the former stage, the neighbors of $\mathbf{x}$ are first obtained by kNN, and subsequently are projected into a low-dimensional subspace. Once the singular values are available, the anomalous score $\text{lps}(\mathbf{x})$ of $\mathbf{x}$ is estimated.

Suppose that there are n observations represented by m features within the data collection **D**. The time complexity of the conventional kNN algorithm is $O(kn^2)$. If the search technique of k-d tree is taken, the efficiency of kNN can be improved to $O(kn\log n)$ [37]. For the optimization and projection problems, it needs to cost $O(\max(km^2, k^2m))$ time. Usually k is less than m. Thus, the time complexity of the proposed method is $O(\max(kn^3, knm^2))$ in total. Nevertheless, LPOD can be finished quickly when comparing to popular outlier detection algorithms in our experiments.

Two parameters involved in LPOD are required to be assigned appropriate values. The first one is the number of desirable nearest neighbors, i.e., k ($1 \leq k \leq n$). As we know, kNN is sensitive to noises if the value of k is too small. Contrastively, when k is large, the probability density function of $\text{lps}(\mathbf{x})$ will be flat, making the identification of outliers difficult. An empirical solution for determining the value of k is cross validation [38]. The simulation experiments presented in the next section show that assigning a value ranging from five to ten to k is properly.

Another parameter of LPOD is the number of desirable outliers, i.e., s. Since ground truth is usually unknown in real-world applications, how many outliers exist within data is a question. Given a data collection, the number of outliers can be estimated empirically. Let y_1 and y_2 be Gaussian random variables represented outlier and normal data, respectively, where $y_1 \sim N(\mu_1, \sigma_1^2)$ and $y_2 \sim N(\mu_2, \sigma_2^2)$. Thus, the total data distribution is their mixed one $y = (1-\theta)y_1 + \theta y_2$, where $\theta \in [0, 1]$ and $P(\theta = 1) = \varepsilon$. As we know, y is still a Gaussian variable. Under this context, ε can be approximately estimated by using machine learning algorithms, e.g., expectation maximization. Once ε is available, the number of desired outliers is determined as $s = n\varepsilon$. For simplicity, s is often prespecified as a const value (e.g., 20 or 50) in the literature.

IV. Experimental Study

To evaluate the effectiveness and efficiency of the proposed method, we carried out a series of comparison experiments with five popular outlier detection algorithms on twelve real-world data sets. This section reports the details of experimental settings and discusses the experimental results.

A. Experimental Settings

1) Experimental Data: The comparison experiments were conducted on two synthetic data sets and 12 real-world data

TABLE I
BRIEF DESCRIPTION OF EXPERIMENTAL DATA

Data sets	Observations	Variables	Outliers
Ann-thyroid	7200	6	534
Banknote	1372	4	610
Diabetes	768	8	268
Digits	10992	16	1143
HAR	10299	561	1406
Ionosphere	351	34	126
Iris	150	4	50
Leukemia	72	7129	25
Ovarian	202	15154	40
Prostate	84	12600	25
Shuttle	58000	8	3511
Wine	178	12	48

sets, including *Ann-thyroid*, *Banknote*, *Diabetes*, *Digits*, human activity recognition *HAR*, *Ionosphere*, *Iris*, *Leukemia*, *Ovarian*, *Prostate*, *Shuttle*, and *Wine*, with different types and sizes. These data sets were frequently used in the literature to test the performance of detection methods. All data sets, except *Leukemia*, *Ovarian*, and *Prostate*, were downloaded from the website of UCI Machine Learning Repository.[1] The *Leukemia*, *Ovarian*, and *Prostate* data sets are available at the website of I^{2}R Data Mining Department's Dataset Respository.[2]

Since the data sets above are originally used for classification, in the literature, a commonly used strategy is to make a technical trick on the data sets for outlier detection, where the minor class in each data set is considered as outlier and the remaining data as the normal ones. After reformulated, they can be used to evaluate the performance of outlier detection methods. For example, 610 observations with the "1" class in the *banknote* data set were treated as outliers, while the rest observations with the "0" class were the normal ones. On the *Wine* and the *Iris* data sets, the observations with the third class were considered as outliers. Since the third class in *Ann-thyroid* involves most of the observations, it was taken as normal and others were outliers. For the *Shuttle* data, there are seven classes, and near 94% observations are labeled with the first and fourth classes. Thus, the observations with the first and fourth classes were regarded as normal and the rest were outliers in the experiments. The same technical trick were made on the other data sets.

Table I summarizes brief information of the data sets, where the *Observations*, *Variables*, and *Outliers* columns denote the total numbers of observations, variables and true outliers, respectively. From this table, one may observe that the experimental data sets vary from the quantities of outliers and differ greatly in the sizes of observations and variables.

2) Evaluation Metrics: To make a comprehensive comparison, three performance evaluation metrics were adopted in the experiments. They were precision (Pr), area under receiver operating characteristic (ROC) curve (AUC) and rank power (RP) [5]. The criterion of precision is frequently used one to assess the performance of learning algorithms. It refers to a ratio of the number of true outliers detected by an outlier detection algorithm over the total number of outlier candidates, that is

$$\mathrm{Pr} = \frac{k}{s} \tag{6}$$

where k is the number of true outliers found within s outlier candidates. This criterion is also called precision@k in the literature, because s is fixed during the evaluation experiments [2]. The ROC curve is a graphical plot of true positive rate versus false positive rate. Since outlier detection methods calculate anomalous scores for observations, AUC, a summary statistic of the ROC curve, is also used to numerically evaluate the performances of the outlier detection algorithms [24].

Both the precision and AUC criteria do not consider characters of outlier ranking. Intuitively, an outlier ranking algorithm will be regarded more effective if it ranks true outliers in the top while normal observations in the bottom of the list of outlier candidates. Rank power is such this metric. Let k be the number of true outliers found within s outlier candidates achieved by an algorithm, and $R(\mathbf{x}_i)$ be the rank of the ith true outlier $\mathbf{x}_i$ in the list. The rank power of the algorithm is defined as

$$\mathrm{RP} = \frac{k(k+1)}{2 * \sum_{i=1}^{k} R(\mathbf{x}_i)}. \tag{7}$$

For a fixed value of s, a larger RP indicates better performance. Especially when the s outlier candidates are true outliers, RP equals to 1.

3) Comparing Algorithms: In the experiments, the proposed method, LPOD, is used to compare with five popular and typical outlier detection algorithms, including LOF [12], LDOF [18], LoOP [30], SOD [30], and HiCS [31]. As discussed above, these detection algorithms stand for different outlier ranking techniques and have relatively better performance, resulting in they are widely used in reality. For example, LOF and LoOP are the density-based detection methods, LDOF is the distance-based detection algorithm, while HiCS and SOD belong to the subspace-based outlier detection techniques. More details of these outlier detection algorithms are provided in the related work section or references therein.

The comparison experiments were carried out under the framework of environment for developing KDD-applications supported by index-structures,[3] which implements the off-the-shelf outlier detection algorithms. During the whole experimental procedures, the parameters involved within the outlier detection algorithms were assigned to default values or suggested values in the literature. The evaluation experiments were conducted on a Pentium IV, with a CPU clock rate of 1.7 GHz, 1 GB main memory.

B. Experimental Results and Discussions

1) Synthetic Data: To test the effectiveness of LPOD in various scenarios, we used two synthetic data sets with different cluster patterns, densities and sizes. In each data set, six outliers were located in nearby places of normal clusters with different densities. The first synthetic data set consists of

[1] http://archive.ics.uci.edu/ml/

[2] http://datam.i2r.a-star.edu.sg/datasets/krbd/index.html

[3] http://elki.dbs.ifi.lmu.de/

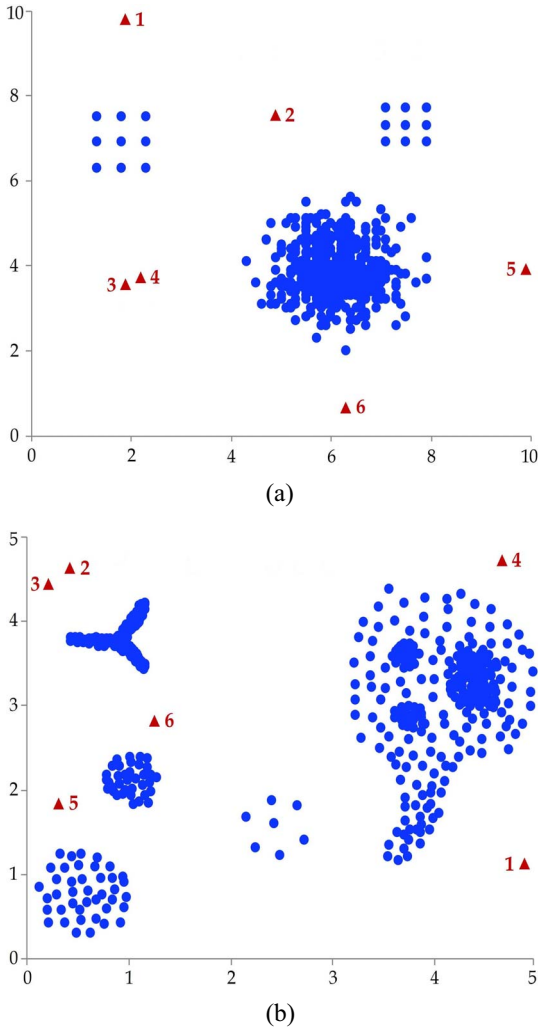

Fig. 3. Outliers marked with triangle were ranked on the top by LPOD. (a) Synthetic data 1. (b) Synthetic data 2.

515 observations grouped into two small clusters and one large cluster distributed normally. The second one has 473 observations grouped into five clusters with different densities. These two data sets involve the low density pattern problem and the global outlier detection task. They were also used to verify the performance of outlier detection algorithms in [22].

After the proposed method performed on the synthetic data sets, the placed outliers were identified easily by LPOD. They had the highest anomalous scores and ranked on the top in the list. The results are provided in Fig. 3, where the triangle points are the placed outliers identified by LPOD and the accompanied number indicates its rank in the list. According to the figure, it can be easily observed from Fig. 3 that the proposed method can detect all outliers out by ranking them on the top in the synthetic data sets.

2) Real-World Data: In the comparison experiments, we first evaluated the performance of the outlier detection algorithms with the precision criterion. Specifically, we first performed the outlier detection algorithms on each data set, and ranked the observations according to the corresponding scores. Then we got top s ($s = 20$, 50, and 100, respectively) candidate outliers. Among them, true outliers were counted to estimate the precision.

Table II reports the precision (%) of outlier identification achieved by the comparing algorithms when top s ($s = 20$, 50, and 100, respectively) outlier candidates were picked out. In the table, the bold value indicates that the performance is the best one among the comparing algorithms on the data set (the same row). For example, our method achieved the best performance, 70%, on the *Iris* data, when top 20 outlier candidates considered, that is, 14 true outliers were successfully identified by LPOD. Since there are only 72 observations in *Leukemia*, we only got 72 suspicious outliers at the final stage. Thus, the result of the top 72, rather than 100, is given in Table II. Similar case is to the *Prostate* data set, which has only 84 observations.

From the experimental results, one may observe that compared to the popular outlier detection algorithms, our method, LPOD, has predominant performance to identify outliers. For instance, LPOD achieved the best performance on ten data sets when $s = 20, 50$ and 100. For instance, LPOD identified 4 and 38 true outliers among the top 20 and 100 candidates on the *Ann-thyroid* data set, respectively, while the others identified no more than 2 and 30 true outliers, respectively.

On the *HAR* and *Wine* data sets, LPOD had relatively poor performance. The reason is that the class distributions are insignificantly different to each other, making outlier identification more difficult when using kNN. Thus, the methods using local techniques, such as SOD and LDOF, had better performance. Even so, LPOD was still superior to LOF, LDOF, LoOP, and HiCS on the *HAR* data set significantly, and slightly worse than SOD.

As discussed above, most of outlier detection methods first rank suspicious observations according to anomalous scores and then determine outliers based on prespecified conditions or apriori knowledge. The precision criterion, which simply measures how many true outliers have been identified, however, does not take the ranking aspect into account. A good ranking method will rank true outliers on the top of the lists of suspicious observations, and the higher the order, the better the performance of ranking method. To measure the rank order of true outliers, we also adopted the measurement of rank power to validate the performance of the outlier detection methods.

Specifically, we performed the outlier detection methods on each data set to rank the observations in a descending order according to the corresponding ranking scores. Then we got the top 50 suspicious outliers and estimated the value of rank power for each detection method. Table III shows the rank power of the top 50 (i.e., $s = 50$) outliers achieved by the outlier detection algorithms with $k = 5$ on the experimental data.

The experimental results in Table III indicate that like the criterion of precision, LPOD outperformed the popular detection methods on all data sets, except *HAR* and *Wine*. For example, the rank power of LPOD on *Leukemia* was 0.67, while the largest one of the popular detection methods was 0.44, which was achieved by LOF. It is noticeable that both LPOD and SOD achieved relatively high values of rank power

TABLE II
PRECISION (%) OF THE OUTLIER DETECTION ALGORITHMS ON THE EXPERIMENTAL DATA

Data set	Top s	LOF	LDOF	LoOP	HiCS	SOD	LPOD
Ann-thyroid	20	0.0	5.0	10.0	0.0	10.0	**20.0**
	50	12.0	6.0	18.0	2.0	**30.0**	**30.0**
	100	19.0	12.0	17.0	13.0	30.0	**38.0**
Banknote	20	**10.0**	5.0	0.0	5.0	**10.0**	**10.0**
	50	14.0	14.0	6.0	**22.0**	20.0	**22.0**
	100	19.0	21.0	22.0	26.0	21.0	**37.0**
Diabetes	20	30.0	30.0	35.0	45.0	55.0	**60.0**
	50	34.0	30.0	32.0	38.0	54.0	**58.0**
	100	30.0	38.0	34.0	40.0	47.0	**61.0**
Digits	20	5.0	15.0	5.0	15.0	15.0	**20.0**
	50	4.0	14.0	8.0	16.0	14.0	**20.0**
	100	7.0	10.0	7.0	15.0	16.0	**22.0**
HAR	20	0.0	10.0	0.0	0.0	**90.0**	85.0
	50	4.0	12.0	8.0	4.0	**94.0**	86.0
	100	4.0	16.0	5.0	10.0	**91.0**	83.0
Ionosphere	20	95.0	95.0	95.0	50.0	**100**	**100**
	50	94.0	82.0	88.0	54.0	**100**	**100**
	100	81.0	65.0	74.0	37.0	86.0	**95.0**
Iris	20	30.0	15.0	35.0	25.0	60.0	**70.0**
	50	28.0	28.0	28.0	28.0	50.0	**62.0**
	100	30.0	32.0	28.0	32.0	37.0	**49.0**
Leukemia	20	45.0	45.0	45.0	35.0	40.0	**75.0**
	50	38.0	34.0	36.0	34.0	40.0	**46.0**
	72	34.7	34.7	34.7	34.7	34.7	34.7
Ovarian	20	20.0	15.0	20.0	**45.0**	25.0	**45.0**
	50	30.0	24.0	26.0	32.0	30.0	**44.0**
	100	31.0	23.0	26.0	20.0	31.0	**35.0**
Prostate	20	25.0	35.0	30.0	50.0	**80.0**	**80.0**
	50	30.0	32.0	28.0	36.0	**50.0**	**50.0**
	84	25.0	25.0	25.0	25.0	25.0	25.0
Shuttle	20	45.0	10.0	10.0	**50.0**	45.0	**50.0**
	50	32.0	16.0	14.0	28.0	50.0	**54.0**
	100	25.0	19.0	20.0	30.0	48.0	**57.0**
Wine	20	15.0	25.0	20.0	**30.0**	10.0	15.0
	50	16.0	26.0	26.0	**28.0**	12.0	20.0
	100	26.0	**32.0**	25.0	27.0	21.0	30.0

TABLE III
RANK POWER OF THE TOP 50 (I.E., $s = 50$) OUTLIER CANDIDATES ACHIEVED BY THE OUTLIER DETECTION ALGORITHMS WITH $k = 5$

Data set	LOF	LDOF	LoOP	HiCS	SOD	LPOD
Ann-thyroid	0.16	0.09	0.17	0.29	0.09	**0.34**
Banknote	0.17	0.18	0.16	0.20	0.23	**0.30**
Diabetes	0.35	0.31	0.36	0.54	0.39	**0.64**
Digits	0.07	0.13	0.08	0.16	0.17	**0.21**
HAR	0.04	0.14	0.08	**0.94**	0.03	0.86
Ionosphere	0.94	0.93	0.92	**1.00**	0.58	**1.00**
Iris	0.29	0.29	0.30	0.48	0.28	**0.63**
Leukemia	0.44	0.40	0.43	0.37	0.40	**0.67**
Ovarian	0.27	0.23	0.25	0.30	0.42	**0.44**
Prostate	0.26	0.33	0.27	0.81	0.43	**0.85**
Shuttle	0.34	0.18	0.17	0.52	0.31	**0.59**
Wine	0.22	**0.31**	0.25	0.17	0.28	0.28

on *Prostate*, *Ionosphere*, and *HAR*, while the others had poor performance. This implies a fact that there exists redundant information at the aspect of dimensions within these data, leading to good performance of the subspace-based detection methods. As aforementioned discussion, the class distributions of *Wine* are insignificant different to each other. Thus, LDOF had better performance than LPOD. Even so, LPOD was still better than LOF, SOD, and LoOP on *Wine*. It should be pointed out that LOF, HiCS, and LoOP had relatively poor performance in most cases. As an example, the rank power of HiCS on the *Ionosphere* data was 0.58, which was the lowest one, whereas the others were larger than 0.90.

Note that LPOD exploits kNN to get neighborhood information for an observation, and the others also take use of kNN as a basis to determine suspicious outliers. Thus, the value of k may bring impacts on the outlier detection algorithms. To testify the effects of k to the outlier detection algorithms, we conducted additional experiments with different values of k for the detection algorithms, and computed their corresponding AUC values, respectively. Fig. 4 presents the varieties of AUC values of the outlier detection algorithms with different values of k, ranging from 5 to 50.

From the experimental results in Fig. 4, one may observe that the number of nearest neighbors, k, had less impacts on

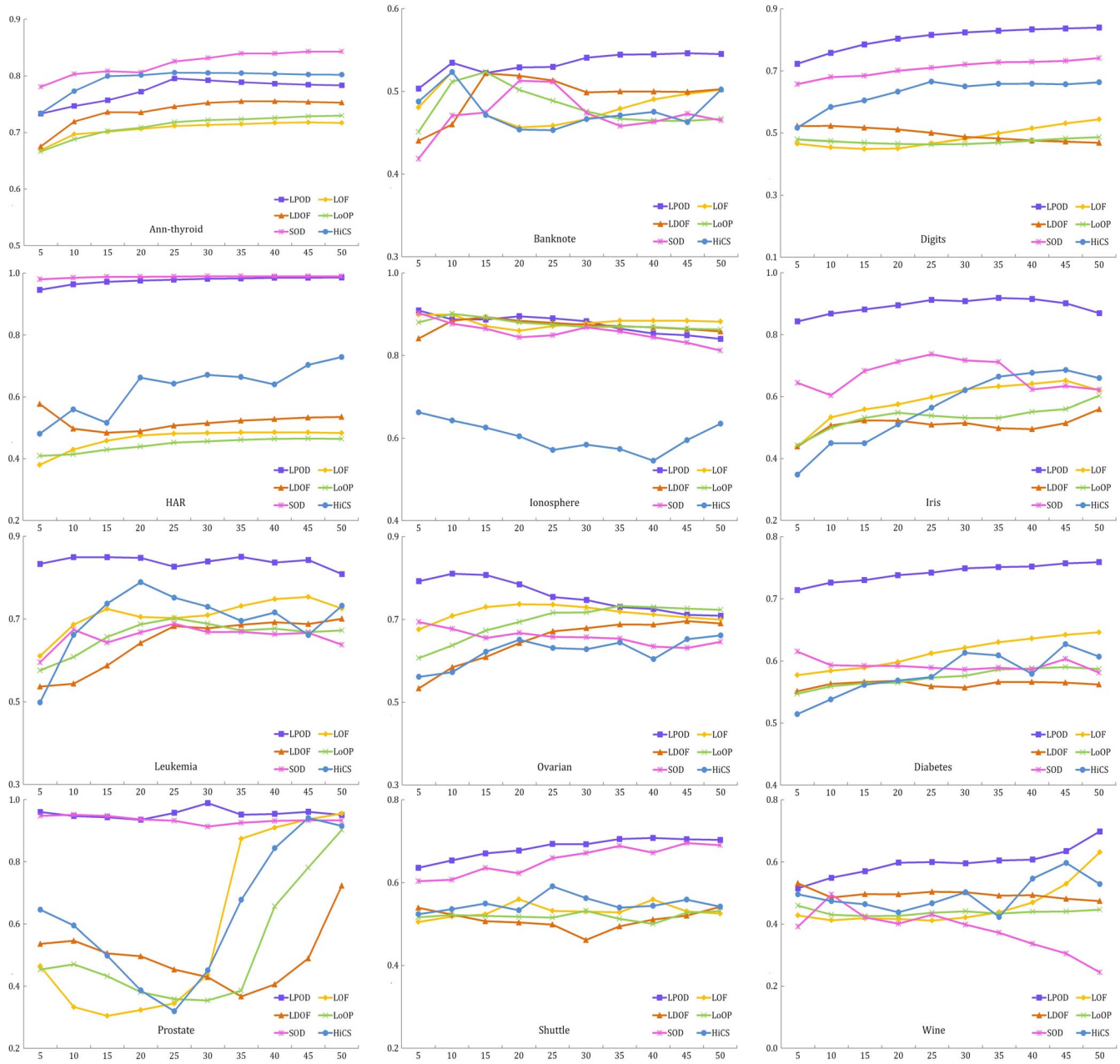

Fig. 4. AUC of the outlier detection algorithms with different values of k, ranging from 5 to 50, of kNN.

the performance of our detection method, i.e., LPOD. Indeed, the AUC values of LPOD on the nine data sets changed gently along with k increasing. Although the AUC values of LPOD on *Ionosphere* and *Ovarian* changed decreasingly, they were not the worst ones in comparing to the others. Roughly speaking, our method tended to be stable when k reached 25 around.

Another interesting fact in Fig. 4 is that the AUC values of LPOD were larger than those of the comparing detection algorithms on the experimental data, except *Ann-thyroid*, *HAR*, and *Ionosphere*. Although SOD had the largest values of AUC on the *HAR*, its performance changed greatly along with different k on nine over twelve data sets, especially *Banknote*, *Iris*, and *Wine*. In addition, k brought great effects to HiCS, that is, the performance of HiCS heavily relied on the value of k. LoOP, LDOF, and LOF had similar performance with respect to the AUC values.

The performance of LPOD was stable when there are less outliers in data, while the others were sensitive to the values of k. As a matter of fact, the AUC values of the comparing algorithms varied greatly on five over twelve data sets, including *Banknote*, *Iris*, *Leukemia*, *Prostate*, and *Wine*, as k increasing. On the rest data sets, their performance varied gently. The reason is that there are less outliers within these data sets, resulting in the traditional outlier detection algorithms are sensitive to the values of k.

Time complexity is an important aspect that should be taken into consideration by an outlier detection algorithm for its practicable applications. We made a relatively naive

TABLE IV
TIME COST (S) OF THE OUTLIER DETECTION ALGORITHMS ON THE EXPERIMENTAL DATA

Data set	LOF	LDOF	LoOP	HiCS	SOD	LPOD
Ann-thyroid	3.18	3.47	3.18	58.07	63.97	0.27
Banknote	0.11	0.16	0.11	2.30	1.76	0.45
Diabetes	0.05	0.05	0.05	1.81	0.18	0.24
Digits	14.31	15.07	14.29	109.33	225.07	6.72
HAR	11.09	11.13	11.11	2138.77	11.79	3.54
Ionosphere	0.03	0.03	0.03	1.77	0.05	0.11
Iris	0.08	0.09	0.05	0.86	0.27	0.06
Leukemia	0.22	1.58	0.22	7650.75	0.41	8.52
Ovarian	35.25	114.66	35.33	38623.91	45.43	62.97
Prostate	5.13	32.58	5.02	18903.36	9.01	22.08
Shuttle	49.60	49.90	49.82	457.39	225.72	22.11
Wine	0.05	0.16	0.05	2.61	0.33	0.09

comparison of efficiencies of the outlier detection algorithms by estimating time costs elapsed during the comparison experiments. Table IV records the elapsed time (s) of the outlier detection algorithms on the experimental data, where k=5. The run time in Table IV shows that comparing to the popular outlier detection algorithms, the efficiency of LPOD was promising, although it was slower than LOF, LDOF, and LoOP in several cases. For example, LPOD finished within ten seconds on nine over twelve data sets. Additionally, the efficiency of LPOD was significantly better than HiCS and comparable to SOD in most cases. To some extent, the existing difference between LPOD and LOF is reasonable because they adopt different versions of kNN, which take a major role in the comparing algorithms from the view of time cost.

V. CONCLUSION

In this paper, we develop an efficient and effective learning method to identify outliers out from normal observations. The main idea of the proposed method is to exploit local neighborhood information of an observation to determine whether it is an outlier or not. To capture the neighborhood information exactly, a concept called LPS is introduced to measure the anomalous degree of a suspicious observation. The observation with high LPS is a promising candidate of outlier in high probability. Formally, the LPS is consistent with the concept of nuclear norm and can be obtained by the technique of low-rank matrix approximation. Moreover, unlike existing distance-based and density-based detection methods, the proposed method is robust to the parameter k of kNN embedded within LPOD. To demonstrate the effectiveness of our proposed method, we performed a comprehensive experiment with five popular outlier detection algorithms on a number of public real-world data sets. The experimental results of the numerical comparison show that the LPS is good at ranking the best candidates for being outliers, and the performance of LPOD is promising at many aspects.

Since LPOD exploits kNN to get neighborhood information, its efficiency relies on kNN and its performance will be affected by the distance formulation of kNN to some extent. In our future work, we will take these aspects into consideration and extend LPOD to the scenarios of big data.

ACKNOWLEDGMENT

The authors would like to thank the anonymous referees and the associate editor for their valuable comments and constructive suggestions, which have improved this paper greatly.

REFERENCES

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Survey*, vol. 41, no. 3, pp. 1–58, 2009.
[2] A. Zimek, E. Schubert, and H.-P. Kriegel, "A survey on unsupervised outlier detection in high-dimensional numerical data," *Stat. Anal. Data Min.*, vol. 5, no. 5, pp. 363–387, 2012.
[3] A. Koufakou and M. Georgiopoulos, "A fast outlier detection strategy for distributed high-dimensional data sets with mixed attributes," *Data Min. Knowl. Disc.*, vol. 20, no. 2, pp. 259–289, 2010.
[4] J. Ha, S. Seok, and J.-S. Lee, "Robust outlier detection using the instability factor," *Knowl. Based Syst.*, vol. 63, no. 6, pp. 15–23, 2014.
[5] C. C. Aggarwal, *Outlier Analysis*. New York, NY, USA: Springer, 2013.
[6] V. Riffo and D. Mery, "Automated detection of threat objects using adapted implicit shape model," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 4, pp. 472–482, Apr. 2016.
[7] L. V. Allen and D. M. Tilbury, "Anomaly detection using model generation for event-based systems without a preexisting formal model," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 42, no. 3, pp. 654–668, May 2012.
[8] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han, "Outlier detection for temporal data: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 9, pp. 2250–2267, Sep. 2014.
[9] D. Berrar, "Learning from automatically labeled data: Case study on click fraud prediction," *Knowl. Inf. Syst.*, vol. 46, no. 2, pp. 477–490, 2016.
[10] R. Mitchell and I.-R. Chen, "Adaptive intrusion detection of malicious unmanned air vehicles using behavior rule specifications," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 5, pp. 593–604, May 2014.
[11] H. Huang, K. Mehrotra, and C. K. Mohan, "Rank-based outlier detection," *J. Stat. Comput. Simulat.*, vol. 83, no. 3, pp. 518–531, 2013.
[12] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, 2000, pp. 93–104.
[13] M. Radovanović, A. Nanopoulos, and M. Ivanović, "Reverse nearest neighbors in unsupervised distance-based outlier detection," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 5, pp. 1369–1382, May 2015.
[14] L. L. C. Kasun, Y. Yang, G.-B. Huang, and Z. Zhang, "Dimension reduction with extreme learning machine," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3906–3918, Aug. 2016.
[15] E. Eskin, "Anomaly detection over noisy data using learned probability distributions," in *Proc. 17th Int. Conf. Mach. Learn.*, 2000, pp. 255–262.
[16] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori, "Statistical outlier detection using direct density ratio estimation," *Knowl. Inf. Syst.*, vol. 26, no. 2, pp. 309–336, 2011.

[17] Y. Pang, J. Cao, and X. Li, "Learning sampling distributions for efficient object detection," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 117–129, Jan. 2017.
[18] K. Zhang, M. Hutter, and H. Jin, "A new local distance-based outlier detection approach for scattered real-world data," in *Proc. 13th Pac.–Asia Conf. Knowl. Disc. Data Min. (PAKDD)*, Bangkok, Thailand, 2009, pp. 813–822.
[19] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, 2000, pp. 427–438.
[20] F. Angiulli and C. Pizzuti, "Fast outlier detection in high dimensional spaces," in *Proc. 6th Eur. Conf. Principles Data Min. Knowl. Disc.*, 2002, pp. 15–26.
[21] J. Liu and H. Deng, "Outlier detection on uncertain data based on local information," *Knowl. Based Syst.*, vol. 51, pp. 60–71, Oct. 2013.
[22] X. Wang, X. L. Wang, Y. Ma, and D. M. Wilkes, "A fast MST-inspired kNN-based outlier detection method," *Inf. Syst.*, vol. 48, pp. 89–112, Mar. 2015.
[23] J. Huang, Q. Zhu, L. Yang, and J. Feng, "A non-parameter outlier detection algorithm based on natural neighbor," *Knowl. Based Syst.*, vol. 92, pp. 71–77, Jan. 2016.
[24] J. Ha, S. Seok, and J.-S. Lee, "A precise ranking method for outlier detection," *Inf. Sci.*, vol. 324, pp. 88–107, Dec. 2015.
[25] C. Xiao and W. A. Chaovalitwongse, "Optimization models for feature selection of decomposed nearest neighbor," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 2, pp. 177–184, Feb. 2016.
[26] Q. Wu, Z. Wang, F. Deng, Z. Chi, and D. D. Feng, "Realistic human action recognition with multimodal feature selection and fusion," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 4, pp. 875–885, Jul. 2013.
[27] J. Tang, Z. Chen, A. W.-C. Fu, and D. W.-L. Cheung, "Enhancing effectiveness of outlier detections for low density patterns," in *Proc. 6th Pac.–Asia Conf. Knowl. Disc. Data Min. (PAKDD)*, 2002, pp. 535–548.
[28] W. Jin, A. K. H. Tung, J. Han, and W. Wang, "Ranking outliers using symmetric neighborhood relationship," in *Proc. 10th Pac.–Asia Conf. Knowl. Disc. Data Min. (PAKDD)*, 2006, pp. 577–593.
[29] S. Papadimitriou, H. Kitagawa, P. B. Gibbons, and C. Faloutsosk, "LOCI: Fast outlier detection using the local correlation integral," in *Proc. IEEE 19th Int. Conf. Data Eng. (ICDE)*, Bengaluru, India, 2003, pp. 315–326.
[30] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek, "LoOP: Local outlier probabilities," in *Proc. 18th ACM Conf. Inf. Knowl. Manag. (CIKM)*, Hong Kong, 2009, pp. 1649–1652.
[31] F. Keller, E. Müller, and K. Bohm, "HiCS: High contrast subspaces for density-based outlier ranking," in *Proc. IEEE 28th Int. Conf. Data Eng. (ICDE)*, 2012, pp. 1037–1048.
[32] H. Liu, Z. Ma, S. Zhang, and X. Wu, "Penalized partial least square discriminant analysis with ℓ_1-norm for multi-label data," *Pattern Recognit.*, vol. 48, no. 5, pp. 1724–1733, 2015.
[33] S. Banerjee and A. Roy, *Linear Algebra and Matrix Analysis for Statistics, Texts in Statistical Science*. Boca Raton, FL, USA: CRC Press, 2014.
[34] I. Markovsky, *Low Rank Approximation: Algorithms, Implementation, Applications*. London, U.K.: Springer-Verlag London, 2012.
[35] M. Nejati, S. Samavi, H. Derksen, and K. Najarian, "Denoising by low-rank and sparse representations," *J. Vis. Commun. Image Represent.*, vol. 36, pp. 28–39, Apr. 2016.
[36] X. Zhou, C. Yang, H. Zhao, and W. Yu, "Low-rank modeling and its applications in image analysis," *ACM Comput. Surveys*, vol. 47, no. 2, pp. 1–35, 2014.
[37] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching fixed dimensions," *J. ACM*, vol. 45, no. 6, pp. 891–923, 1998.
[38] H. Liu, X. Wu, and S. Zhang, "Neighbor selection for multilabel classification," *Neurocomputing*, vol. 182, pp. 187–196, Mar. 2016.

Huawen Liu (M'17) received the master's and Ph.D. degrees in computer science from Jilin University, Changchun, China, in 2007 and 2010, respectively.

He is currently an Associate Professor with the Department of Computer Science, Zhejiang Normal University, Jinhua, China. His current interests include data mining, feature selection, sparse learning, and machine learning.

Xuelong Li (M'02–SM'07–F'12) is a Full Professor with the School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China.

Jiuyong Li (M'05) received the Ph.D. degree in computer science from Griffith University, Nathan, QLD, Australia.

He is currently a Professor with the University of South Australia, Adelaide, SA, USA. His research has been supported by six Australian Research Council Discovery projects and he has published over 120 research papers. His current research interests include data mining, privacy preservation, and bioinformatics.

Shichao Zhang (SM'05) received the Ph.D. degree in computer science from Deakin University, Burwood, VIC, Australia.

He is currently a Full Professor and a China 1000-Plan Distinguished Professor with the Department of Computer Science, Guangxi Normal University, Guilin, China. His current research interests include data mining, computer vision, pattern recognition, artificial intelligence, and machine learning.