Knowl Inf Syst
DOI 10.1007/s10115-012-0484-y

REGULAR PAPER

SVDD-based outlier detection on uncertain data

Bo Liu · Yanshan Xiao · Longbing Cao · Zhifeng Hao · Feiqi Deng

Received: 9 May 2010 / Revised: 13 June 2011 / Accepted: 5 March 2012

Abstract Outlier detection is an important problem that has been studied within diverse research areas and application domains. Most existing methods are based on the assumption that an example can be exactly categorized as either a normal class or an outlier. However, in many real-life applications, data are uncertain in nature due to various errors or partial completeness. These data uncertainty make the detection of outliers far more difficult than it is from clearly separable data. The key challenge of handling uncertain data in outlier detection is how to reduce the impact of uncertain data on the learned distinctive classifier. This paper proposes a new SVDD-based approach to detect outliers on uncertain data. The proposed approach operates in two steps. In the first step, a pseudo-training set is generated by assigning a confidence score to each input example, which indicates the likelihood of an example tending normal class. In the second step, the generated confidence score is incorporated into the support vector data description training phase to construct a global

B. Liu
Faculty of Automation, Guangdong University of Technology,
Guangdong, People's Republic of China
e-mail: csbliu@gmail.com

Y. Xiao · Z. Hao (✉)
Faculty of Computer, Guangdong University of Technology,
Guangdong, People's Republic of China
e-mail: mazfhao@scut.edu.cn

Y. Xiao
e-mail: xiaoyanshan@gmail.com

L. Cao
Faculty of Engineering and Information Technology,
University of Technology, Sydney, Australia
e-mail: lbcao@it.uts.edu.au

F. Deng
School of Automation Science and Engineering, South China University of Technology,
Guangdong, People's Republic of China
e-mail: aufqdeng@scut.edu.cn

Published online: 06 May 2012

distinctive classifier for outlier detection. In this phase, the contribution of the examples with the least confidence score on the construction of the decision boundary has been reduced. The experiments show that the proposed approach outperforms state-of-art outlier detection techniques.

Keywords Outlier detection · Data of uncertainty · Support vector data description

1 Introduction

Outlier detection refers to the problem of determining data objects that are markedly different from, or inconsistent with, the remaining set of data [49]. Outlier detection has increasingly attracted attention due to its wide variety of applications from fraud detection for credit cards, insurance [45], or health care [53] to faulty detection in critical safety systems [47,37].

Traditional outlier detection algorithms typically assume that outliers are difficult or costly to obtain due to their rare occurrence. Therefore, most of the previous approaches focus on modeling a representation of the normal data so as to identify outliers that do not fit the model. These previous outlier detection algorithms are broadly classified into four categories: (1) Statistics-based algorithms [20], where statistical techniques fit a statistical model (usually for normal data) to the given data and then apply a statistical inference test to determine whether an incoming instance fits the model or not. (2) Density-based method [43], in which local outliers are identified by examining the distances to their nearest neighbors. (3) Clustering-based approaches [28], which groups similar data instances into clusters and considers clusters of small size as outliers. (4) Model-based method [32,17], which is used to learn a distinctive model from a set of training data instances and to detect outliers as deviations from the model. In this category, SVDD [42,44,33], proposed to determine a sphere around normal data, has been demonstrated to be capable of capturing outliers in various applications [42,23,52].

Another important observation is that, data are uncertain in nature for many real-life applications [8,5]. For example, the data points may correspond to objects, which are only vaguely specified due to data incompleteness, and are therefore considered uncertain in their representation [5]; moreover, some new hardware technologies such as sensors usually collect large amounts of uncertain data due to sampling errors or instrument imperfections [8]. Consequently, a labeled normal example corrupted by various errors or limitations of the underlying equipment always behaves like an outlier, even though the example itself may not be an outlier. This always makes the problem of outlier detection far more difficult from the perspective of data uncertainty. Therefore, it is worthwhile to develop techniques to refine the decision boundary of the distinctive classifier so as to improve the performance of outlier detection. The key challenge of handling uncertain data in outlier detection is how to reduce the impact of the uncertain data on the learned distinctive classifier.

In order to handle the problem of outlier detection in the presence of uncertain data, this paper proposes a model-based approach by introducing a confidence score for each input data point into the SVDD training phase. Our proposed approach operates in two steps. In the first step, we generate a pseudo-training dataset by assigning a confidence level to each input data point, which indicates the likelihood of an input data point belonging to normal class. We put forward a kernel-based class center method to generate the confidence level for each input training sample. In the second step, we incorporate the generated confidence score for each sample into the SVDD training process. By introducing a confidence score into the training stage, each data sample contributes differently to the construction of the decision boundary, which is used for outlier detection. Substantial experiments have demonstrated that our proposed approach offers higher performance for outlier detection in comparison with SVDD and GMM in terms of RBF and polynomial kernel functions.

The rest of the paper is organized as follows. Section 2 presents the previous works related to our study. Section 3 puts forward our proposed approach for outlier detection on uncertain data. Substantial experiments are demonstrated in Sect. 4, and a conclusion is drawn in Sect. 5.

2 Related work

Since the focus of our study is SVDD-based outlier detection on uncertain data, we briefly review traditional outlier detection technologies in Sect. 2.1, introduce the data uncertainty problem in Sect. 2.2, and present a brief introduction to SVDD in Sect. 2.3.

2.1 Outlier detection

Outlier detection refers to the problem of finding patterns in data that do not conform to expected behavior. Past works can be broadly classified into the following four categories.

1. Statistics-based techniques always fit a statistical model to the given data and then apply a statistical inference test to determine whether an unseen instance satisfies this model or not. Instances that have a low probability of being generated from the learned model, based on the applied test statistic, are declared as outliers. For example, we can assume the normal examples follow a certain data distribution (such as Gaussian distribution), by estimating the parameter in the model, we can generate a Gaussian model to predict an unseen example into normal class or outliers. The statistics-based techniques always assume knowledge of the underlying distribution and estimate the parameters from the given data [20,24,19,48] such as Gaussian model based [6,39], in which the pre-specified data distribution is assumed to fit a Gaussian distribution; regression model based [16], where outlier detection using regression has been extensively investigated for time-series data; mixture of parametric distributions based [2,25], in which techniques use a mixture of parametric statistical distributions to model the data. For this category, the main disadvantage is that these techniques rely on the assumption that the data is generated from a particular distribution. However, this assumption often does not hold true in many applications, especially for high dimensional real data sets.
2. Density-based approaches always assume that normal data instances occur in dense neighborhoods, while outliers occur far from their closest neighbors [43,24,21]. One representative method is called LOF (local outlier factor) [14], which assigns an outlier score to any given data point, depending on its distances in the local neighborhood. Recently, the work proposed by [54] improves the accuracy of outlier detection by calculating an outlier score based on a Gaussian mixture model (GMM). However, if the data has normal instances that do not have enough close neighbors or if the data has outliers that have enough close neighbors, the technique fails to label them correctly, resulting in missed outliers.
3. Clustering-based methods [27,28,34,41,22,40] mainly rely on applying clustering techniques to characterize the local data behavior. As a by-product of clustering, small clusters that contain significantly fewer data points than other clusters are considered as outliers. The performance of clustering based techniques is highly dependent on the effectiveness of the clustering algorithm in capturing the cluster structure of normal instances.

4. Model-based methods are used to learn a model (classifier) from a set of labeled data instances (training) and then to classify a test instance into one of the classes using the learnt model (testing) [33,26,29,44,51,10,11]. Model-based outlier detection techniques operate in a similar two-phase fashion. The training phase learns a classifier using the available labeled training data. The testing phase classifies a test instance as normal or anomalous using the classifier. In this category, SVDD proposed by [44], has been demonstrated empirically to be capable of detecting outliers in various domains. Model-based approaches can detect global outliers effectively for high-dimensional data without need to assume the prior distribution of data. Density-induced SVDD (DI-SVDD) [33] introduces new distance measurements based on the notion of a relative density (or, significance) degree of each data point to reflect the distribution of a given data set. Although Density-induced SVDD can increase the accuracy of SVDD, it requires linearly constrained optimizations by solving a sequence of quadratic programming subproblems. Consequently, it spends much more time to construct a classifier.

In addition, Bayesian-based approaches have been proposed for outlier detection. The work in [1] and [2,3] uses Bayesian analysis for outlier detection in dynamic time series environment; the methods [9–11] adopt generalized radial basis function networks for outlier detection. The work in [35] introduces D-Search concept to exploit similarity search for large distribution sets

The limitation of the previous works is that they typically make the assumption that an input data sample can be regarded as belonging completely to the class of normal data or outliers. However, this is not appropriate for uncertain data. For example, a labeled normal example corrupted by various errors or limitations of the underlying equipment always behaves like an outlier, even though the example itself may not be an outlier. Therefore, the key challenge of handling uncertain data in outlier detection is how to reduce the impact of the uncertain data on the learned distinctive classifier. When most of the previous works are performed on the uncertain data, the decision boundary of these methods will be impacted by the data containing uncertain information; consequently, performance will be reduced.

Our proposed approach falls into the model-based category, which is proposed to account for the challenge of outlier detection on uncertain data. More specifically, our method only determines the local uncertainty by generating a confidence score for each instance, which indicates the likelihood of this sample belonging to normal class, but also constructs a global outlier detection classifier. The experiments demonstrated in Sect. 4 have shown that our proposed approach outperforms state-of-art outlier detection algorithms in terms of performance and sensitivity to noise.

2.2 Data of uncertainty

In recent years, many advanced technologies have been developed to store and record large quantities of data continuously. This has created a need for uncertain data algorithms and applications [8]. Various algorithms have been proposed to handle the uncertain data in query processing of uncertain data [18], indexing uncertain data [15], clustering uncertain data [31], classification of uncertain data [12], frequent pattern mining of uncertain data [55]. Meanwhile, [7] considers uncertain data in the outlier detection problem where a probabilistic definition of outliers in conjunction with density estimation and sampling are used. Different from this work, our method is a model-based method, which does not need to pre-specify the density function of the dataset; therefore, our method can learn a distinctive classifier from the training set without assuming the distribution of the data. At the same time, our method models the uncertainty by assigning a confidence score to each sample and reduces the impact of the uncertain data on the construction of the classifier.

2.3 Support vector data description

Assume the training normal data are denoted as $\mathbf{x}_1, x_2, \ldots, \mathbf{x}_l$, where $\mathbf{x}_i \in R^n$, $i = 1, 2, \ldots, l$. In SVDD, the normal class is mapped from the input space into a feature space via a mapping function $\phi(.)$. In this feature space, the normal class is denoted as

$$\phi(\mathbf{x}_1), \phi(\boldsymbol{x}_2), \ldots, \phi(\mathbf{x}_l), \tag{1}$$

where $\phi(\mathbf{x}_i)$ is the image of sample $\mathbf{x}_i$. The purpose of mapping function $\phi(.)$ is to render the patterns much more compact in the feature space than in the input space so as to improve the performance. Further, the inner products of two vectors in the feature space can be computed via a kernel function.

$$K(\mathbf{x}, \mathbf{x}_i) = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}_i), \tag{2}$$

where K satisfies the Mercer theorem [38], $\phi(\mathbf{x})$ and $\phi(\mathbf{x}_i)$ denote two vectors in the feature space.

In the feature space, support vector data description is used to determine the smallest sphere of radius $R > 0$ that encloses all the normal class approximatively as follows.

$$\min \quad R^2 + \gamma \sum_{i=1}^{l} \xi_i \tag{3}$$

$$\text{s.t.} \quad \| \phi(\mathbf{x}_i) - \mathbf{o} \|^2 \leq R^2 + \xi_i, \quad i = 1, 2, \ldots, l, \tag{4}$$

$$\xi_i \geq 0, \quad i = 1, 2, \ldots, l. \tag{5}$$

where $\| \cdot \|$ means the Euclidean norm and $\mathbf{o}$ denotes the center of the sphere, ξ_i are slack variables to relax the constraints, γ is a parameter that specifies the trade-off between the sphere volume and the errors. $\sum_{i=1}^{l} \xi_i$ means the penalty term accounting for the presence of outliers. By introducing Lagrangian function [46], problem (3) is changed into

$$\max \sum_{i=1}^{l} \alpha_i K(\mathbf{x}_i, \mathbf{x}_i) - \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{6}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq \gamma \quad i = 1, 2, \ldots, l \tag{7}$$

$$\sum_{i=1}^{l} \alpha_i = 1, \tag{8}$$

where α_i for $i = 1, 2, \ldots, l$ is the Lagrange multipliers and problem (3) is a standard quadratic optimization problem. On the other hand, the samples $\mathbf{x}_i$ for which $\alpha_i \neq 0$ are called support vectors (SVs). Assume $\mathbf{x}_k$ is one of the SVs, and $0 < \alpha_k < \gamma$ holds true, R can be calculated as follows:

$$\| \mathbf{x}_k - \mathbf{o} \|^2 = K(\mathbf{x}_k, \mathbf{x}_k) - 2 \sum_{i=1}^{l} \alpha_i K(\mathbf{x}_k, \mathbf{x}_i) + \alpha_i \alpha_j \sum_{i=1}^{l} \sum_{j=1}^{l} (\mathbf{x}_i \cdot \mathbf{x}_j) = R^2. \tag{9}$$

For a test pattern $\mathbf{x}$, it is assigned into the normal class if the distance between it and the sphere center is smaller than or equal to the radius R; on the contrary, pattern $\mathbf{x}$ is then classified as outliers as illustrated in Fig. 1.

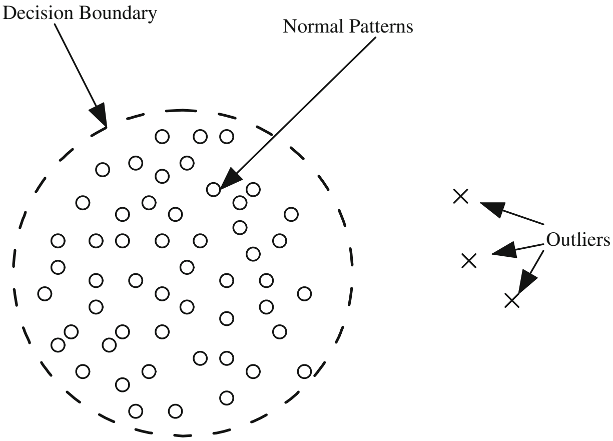


Fig. 1 Support vector data description for outlier detection

3 SVDD-based outlier detection on uncertain data

In this section, we will put forward our proposed approach for SVDD-based outlier detection on uncertain data.

Given l normal training samples, support vector data description constructs a hyper-sphere by enclosing the data appropriately to categorize a test instance into normal class or outliers. However, data are uncertain in nature in real-life applications due to various errors or limitations of the underlying equipment. Consequently, a labeled normal data may behaves like an outlier, although the example itself might not be an outlier; this always makes the problem of outlier detection far more difficult from the perspective of data uncertainty.

In order to address this issue, we introduce a confidence score to each normal sample, which indicates the likelihood of a sample belonging to normal class. Such information are thereafter incorporated into learning a global classifier for outlier detection. Based on this, our approach operates in two steps.

1. In the first step, we produce a pseudo training set by generating a confidence score for each input sample.
2. In the second step, this pseudo training set is used to train a global SVDD classifier by incorporating the generated confidence score together with input data into the learning process.

We introduce the two steps as follows.

3.1 Confidence score generation

We put forward a kernel-based class center method to generate a confidence score for each input sample as follows.

In the kernel space related to a mapping function $\phi(.)$, the experiential center of the normal class is denoted as

$$C^{\phi} = \frac{1}{l} \sum_{i=1}^{l} \phi(\mathbf{x}_i). \tag{10}$$

The kernel-based distance between sample $\mathbf{x}_j$ and the center in the kernel space is calculated

$$\text{Dis}(\phi(\mathbf{x}_j), C^{\phi}) = \| \phi(\mathbf{x}_j) - C^{\phi} \| = \sqrt{K(\mathbf{x}_j, \mathbf{x}_j) - 2K(\mathbf{x}_j, C^{\phi}) + K(C^{\phi}, C^{\phi})}. \quad (11)$$

By substituting (10) into (11), we have

$$\text{Dis}(\phi(\mathbf{x}_j), C^{\phi}) = \sqrt{K(\mathbf{x}_j, \mathbf{x}_j) - \frac{1}{l}\sum_{i=1}^{l} K(\mathbf{x}_j, \mathbf{x}_i) + \frac{1}{l^2}\sum_{i=1}^{l}\sum_{k=1}^{l} K(\mathbf{x}_i, \mathbf{x}_k)}. \quad (12)$$

It can be seen that the kernel-based distance from each sample and its class center can be explicitly calculated via kernel function.

In addition, we determine the maximum kernel-based distance among each sample and the class center. This distance can be denoted as

$$r^{\phi} = \max(\text{Dis}(\phi(\mathbf{x}_j), C^{\phi}), \quad j = 1, 2, \ldots, l. \quad (13)$$

It is noted that the sample residing around the normal data has a larger kernel-based distance among all the samples. The confidence score of each sample $\mathbf{x}_j$ is defined as follows:

$$C(\mathbf{x}_j) = 1 - \frac{\| \phi(\mathbf{x}_j) - C^{\phi} \|}{r^{\phi}}. \quad (14)$$

We can observe that, if $\mathbf{x}_i$ is the sample, which has the maximum kernel-based distance among all the samples, its confidence score will equal 0. To avoid this case, we let the confidence score of this sample be equal to the smallest confidence score among other examples. Our definition of confidence score contains the following observations.

1. This kernel-based confidence score is defined in the kernel space and this confidence score can be directly and efficiently calculated via kernel function.
2. By this definition, the samples at the edge of the normal class have small confidence scores. If a corrupted labeled normal instance behaves like a outlier, it always resides on or out of the boundary of the normal class; according to the definition of (13), the confidence score of this sample towards the normal class is smaller in comparison with that of other examples.

Based on this, the confidence score generation algorithm is put forward in Algorithm 3.1.

```
Algorithm 3.1
Input: Training normal data $\mathbf{x}_i$, $1 \le i \le l$, kernel function $K(.)$.
Output: Pseudo-training set $(\mathbf{x}_i, C(\mathbf{x}_i))$
Procedure
  Define an array $D$ to store $l$ kernel-based distances for each sample.
  Define confidence score array $C$ to put the confidence scores of each sample.
  for $(k = 1; k \le l; k++)$ do
    calculate $H_k = Dis(\phi(\mathbf{x}_k), C^{\phi})$ according to Eq. (12)
  End
  calculate $r^{\phi}$ according to Eq. (13)
  for $(k = 1; k \le l; k++)$ do
    calculate $D_k = C(\mathbf{x}_k)$ according to Eq. (14)
  End
  Return Score array $C$ together with training samples: $(\mathbf{x}_i, C(\mathbf{x}_i))$
```

After obtaining the generated pseudo-training set, it is thereafter incorporated into the learning of SVDD. In the following, we give an extension of the SVDD to incorporate the pseudo-dataset into the training stage.

3.2 Classifier construction

It is seen that confidence score $C(\mathbf{x}_i)$ indicates the likelihood of sample $\mathbf{x}_i$ tending toward the normal class, and the parameter ξ_i in problem (3) is a measure of error for misclassified samples. Therefore, $C(\mathbf{x}_i)\xi_i$ can be considered as a measure of error with different weighting. The new version of SVDD is to solve the following problem

$$\begin{aligned} \min \quad & R^2 + \gamma \sum_{i=1}^{l} C(\mathbf{x}_i)\xi_i \\ & \text{s.t.} \\ & \| \phi(\mathbf{x}_i) - \mathbf{o} \|^2 \leq R^2 + \xi_i, \quad i = 1, 2, \ldots, l, \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, l, \end{aligned} \tag{15}$$

where γ is a parameter specifying the trade-off between the sphere volume and the errors. We can see that a smaller confidence score $C(\mathbf{x}_i)$ can reduce the effect of parameter ξ_i, so that $\mathbf{x}_i$ is treated as less important. We allow each sample to contribute differently on the construction of the SVDD classifier according to the confidence score, which is generated in the step one. In general, if an example falls beyond the normal class, its confidence score will be small using the Eq. (14), by contraries, its score will be large. By this, we potentially reduce the influence of the sample with lowest confidence score on the construction of the hyper-sphere.

In order to solve the optimization problem (15), Lagrangian function [46] can be constructed as follows:

$$L(R, \mathbf{o}, \xi) = R^2 + \gamma \sum_{i=1}^{l} C(\mathbf{x}_i)\xi_i - \sum_{i=1}^{l} \alpha_i (R^2 + \xi_i - \| \phi(\mathbf{x}_i) - \mathbf{o} \|^2) - \sum_{i=1}^{l} \beta_i \xi_i, \tag{16}$$

where $\alpha_i \geq 0$ and $\beta_i \geq 0$ for $i = 1, 2, \ldots, l$ are the Lagrange multipliers. These parameters satisfy the following conditions:

$$\frac{\partial L}{\partial R} = 0 \longrightarrow 2R - R\sum_{i=1}^{l} \alpha_i = 0 \longrightarrow \sum_{i=1}^{l} \alpha_i = 1, \tag{17}$$

$$\frac{\partial L}{\partial \mathbf{o}} = 0 \longrightarrow 2\mathbf{o} - 2\sum_{i=1}^{l} \alpha_i \phi(\mathbf{x}_i)\mathbf{o} \longrightarrow \mathbf{o} = \sum_{i=1}^{l} \alpha_i \phi(\mathbf{x}_i), \tag{18}$$

$$\frac{\partial L}{\partial \xi_i} = 0 \longrightarrow \alpha_i + \beta_i = C(\mathbf{x}_i)\gamma, \quad i = 1, 2, \ldots, l. \tag{19}$$

According to (17–19), we have Theorem 1 as follows.

Theorem 1 *The solution of problem* (15) *can be resolved by problem* (20) *subject to* (21), (22) *(refer to "Appendix" for derivation)*

$$\max \sum_{i=1}^{l} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_i) - \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{20}$$

s.t.

$$0 \leq \alpha_i \leq m(\boldsymbol{x}_i)\gamma \quad i = 1, 2, \ldots, l, \tag{21}$$

$$\sum_{i=1}^{l} \alpha_i = 1. \tag{22}$$

Problem (20) is a standard quadratic programming (QP) problem. After solving problem (20), we have each $\alpha_i, i = 1, 2, \ldots, l$, and the centroid of hyper-sphere is obtained by (18). From Eq. (18), we can see only sample $\mathbf{x}_i$ with $\alpha_i > 0$ contributes the centroid of hyper-sphere, and these samples are called support vectors (SVs). In addition, the KKT theory [46] satisfies

$$\xi_i \beta_i = 0, \quad i = 1, 2, \ldots, l, \tag{23}$$

$$(R^2 + \xi_i - \| \phi(\mathbf{x}_i) - \mathbf{o} \|^2)\alpha_i = 0, \quad i = 1, 2, \ldots, l. \tag{24}$$

If $0 < \alpha_i < C(\mathbf{x}_i)\gamma, \beta_i \neq 0$ comes true from (19) and then $\xi_i = 0$ holds from (23); therefore, from (24), these SVs satisfy

$$(R^2 - \| \phi(\mathbf{x}_i) - \mathbf{o} \|^2) = 0, \quad i = 1, 2, \ldots, l. \tag{25}$$

These samples lie on the surface of the hyper-sphere.

Assume $\mathbf{x}_k$ is one of the SVs lying on the surface of the sphere. R can be computed by

$$\| \phi(\mathbf{x}_k) - \mathbf{o} \|^2 = K(\mathbf{x}_k, \mathbf{x}_k) - 2\sum_{i=1}^{l} \alpha_i K(\mathbf{x}_k, \mathbf{x}_i) + \alpha_i \alpha_j \sum_{i=1}^{l} \sum_{j=1}^{l} K(\mathbf{x}_i, \mathbf{x}_j) = R^2. \tag{26}$$

In summary, the outlier detection classifier determination procedure is presented in Algorithm 3.2.

Algorithm 3.2

```
Input: pseudo-training data $(\mathbf{x}_i, C(\mathbf{x}_i))$ $1 \leq i \leq l$; kernel function $K(.)$.
Output: $\alpha_i$, $1 \leq i \leq l$ and $R$.
Procedure
  –Resolve standard QP problem of (20)
  –Obtain $\alpha_i$ for each sample.
  –Determine a sample whose $\alpha_i$ is between 0 and $C(\mathbf{x}_i)\gamma$, that is the sample resides on the surface of the hyper-sphere.
  –Calculate the radius of hyper-sphere according to Eq. (26).
  Return $\alpha_i$, $1 \leq i \leq$ ł and $R$.
```

Remark:

1. Because the optimization problem (20) is a standard QP problem, the solving of problem (6) and (20) have the same computational complexity.
2. For a test sample $\mathbf{x}$, it is classified into the normal class if it resides inside the sphere, that is (27) comes true, if not, it is classified to outliers.

$$\| \phi(\mathbf{x}) - \mathbf{o} \|^2 = K(\mathbf{x}, \mathbf{x}) - 2\sum_{i=1}^{l} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + \alpha_i \alpha_j \sum_{i=1}^{l} \sum_{j=1}^{l} K(\mathbf{x}_i, \mathbf{x}_j) \leq R^2. \tag{27}$$

4 Experiment

In order to evaluate the performance of our proposed approach, we implement our approach with a kernel-based class center method to generate a confidence score for each input normal sample. For comparison, another three algorithms are utilized here as baselines. The first method is standard support vector data description, which has been introduced in Sect. 2.3. The second one is density-induced SVDD (DI-SVDD) [33], which introduces new distance measurements based on the notion of a relative density degree for each data point to increase the accuracy of SVDD. The first two baselines are used to show the accuracy improvement of our method over the previous SVDD methods. The third method is Gaussian mixture model (GMM) [54], a well-known clustering technique for outlier detection, which decides whether a data point is an outlier based on the outlier factor computed according to a Gaussian Mixture Model fit to the given dataset. For SVDD, DI-SVDD, and our approach, since they are model-based outlier detection techniques, they construct a classifier from the training normal class and predict for the testing dataset. Because GMM is a clustering method, we directly perform it on the testing dataset to report its performance.

In the experiments, RBF kernel function (28) has been utilized due to its comparable performance over other kernel functions.

$$K(\mathbf{x}, \mathbf{x}_i) = \exp^{(\|\mathbf{x}-\mathbf{x}_i\|_2^2/\sigma^2)} \tag{28}$$

For comparison, all the methods are implemented in a Matlab environment.

4.1 Dataset description

In our experiments, both UCI and KDD-cup-1999 intrusion detection[1] datasets are used. The Balance, Ionosphere, Liver disorders, Wine and Image datasets from UCI [36] have been used. Most of these datasets have been utilized to evaluate the performance of SVDD-based outlier detection [44,50]. KDD-cup-1999 is intrusion detection dataset, which involves three common classes of traffic (normal, neptune, and smurf) and seventh rare classes (back, ipsweep, satan, portsweep, nmap, teardrop, guess passwd, pod, warezmaster, land, imap, ftp-write, multihop, buffer overflow, phf, loadmodule, perl). In the experiment, the file *kddcup.data-10-percent* is used, one type common traffic (normal class) and seventh rare classes are used as target class and outliers, respectively. General information about these datasets are briefly introduced in Table 1.[2]

For each UCI dataset, we follow the operations used in [44] to achieve our datasets for outlier detection. Specifically, for each dataset, we choose one of the classes as the target class and treat all other classes as outliers at each round. By doing this, we obtain twelve datasets, that is, Balance (1), Balance (2), Balance (3), Ionosphere (1), Ionosphere (2), Liver disorders (1), Liver disorders (2), Wine (1), Wine (2), Wine (3), Image (1), and Image (2) where each number in the bracket represents the class from the source data, which is chosen as the normal class.

[1] S. Stolfo, KDD-cup 1999 dataset, UCI KDD repository, Tech. Rep., 1999, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[2] Dataset is reorganized that classes 1, 2, 3, 4 are considered as classes 1, and class 5, 6, 7 are regarded as class 2.

Table 1 General information of datasets

Datasets	Class	# of examples	# of features
	Class 1	288	
Balbace	Class 2	49	4
	Class 3	288	
Ionosphere	Class 1	225	34
	Class 2	126	
Liver-disorders	Class 1	145	7
	Class 2	200	
	Class 1	59	
Wine	Class 2	72	4
	Class 3	48	
Image	Class 1	1,320	19
	Class 2	900	
KDDCUP1999	Normal	56,237	41
	Rare classes	4,177	

4.2 Evaluation criterion

The performance of outlier detection is typically evaluated in terms of two rates: true-positive (TP) rate and false-positive (FP) rate. The TP defined in (29) is computed as the ratio of the number of correctly detected normal samples to the total number of normal data. The FP defined in (30) is computed as the ratio of the number of outlier examples that are incorrectly detected as normal data to the total number of outliers.

$$\text{TP} = \frac{\text{Normal samples correctlyclassified}}{\text{Total normal samples}} \tag{29}$$

$$\text{FP} = \frac{\text{Outliers incorrectly classified}}{\text{Total outliers samples}} \tag{30}$$

As a result, a receiver-operating characteristic (ROC) curve [13] can be obtained by plotting the true-positive (TP) rate on the y axis and false-positive (FP) rate on the x axis. For comparison, the area under the ROC curve, which is called AUC, is commonly used to evaluate the performance of an outlier detection method. As discussed in [13], the AUC value is always between 0 and 1, and the larger AUC indicates the better performance of a method.[3]

4.3 Performance evaluation

4.3.1 Average AUC accuracy comparison

We first perform experiments to compare the average AUC accuracy and standard deviation of SVDD, DI-SVDD, GMM, and our proposed method. Specifically, we randomly select 60 % of the normal data to generate the training set and treat the rest of the normal data and outliers as a testing set for ten times. For each generated training and testing set, we

[3] Except for the typical evaluation criterion introduced above, another set of evaluation criterion is F value, precision and recall [30].

Table 2 Average AUC performance and standard deviation comparison of SVDD, DI-SVDD, GMM, and our approach

Datasets	GMM	SVDD	DI-SVDD	Ours
Bal (1)	93.29 ± 3.54	95.94 ± 2.13	96.35 ± 2.53	98.96 ± 0.897
Bal (2)	71.51 ± 8.78	76.57 ± 8.16	78.89 ± 7.54	82.54 ± 6.23
Bal (3)	91.15 ± 3.89	97.45 ± 0.31	97.68 ± 0.35	99.27 ± 0.78
Ion (1)	90.54 ± 4.14	94.97 ± 3.57	95.76 ±2.78	99.23 ± 1.21
Ion (2)	60.87 ± 6.36	70.23 ± 6.23	73.87 ± 6.12	79.13 ± 5.9
Liv (1)	56.23 ±4.13	59.14 ± 3.52	61.73 ± 3.89	64.47 ± 3.25
Liv (2)	46.89 ±6.89	50.98 ±7.67	52.67 ± 7.12	55.98 ± 6.32
Wine (1)	69.45 ± 7.13	75.89 ± 6.34	77.23 ± 6.12	81.58 ± 5.72
Wine (2)	73.23 ± 6.78	77.42 ± 6.45	79.67 ± 6.98	85.82 ± 6.13
Wine (3)	58.12 ± 10.29	66.89 ± 9.29	69.67 ± 9.07	74.73 ± 8.63
Ima (1)	87.98 ± 5.12	95.89 ± 3.34	96.52 ± 2.67	99.12 ± 0.13
Ima (2)	88.76 ± 4.24	96.34± 1.27	97.75± 1.02	98.82 ± 1.18
KDD-Cup	73.45 ± 8.63	78.63 ± 6.34	80.13 ± 5.32	83.56 ± 5.22

We use the top three characters to represent the name of source data

vary the value of the parameter σ in the RBF kernel from 2^{-8} to 2^8, and the parameter γ in the formulation of SVDD and our method from 2^{-8} to 2^8 for each method to construct a predictive outlier detection classifier.

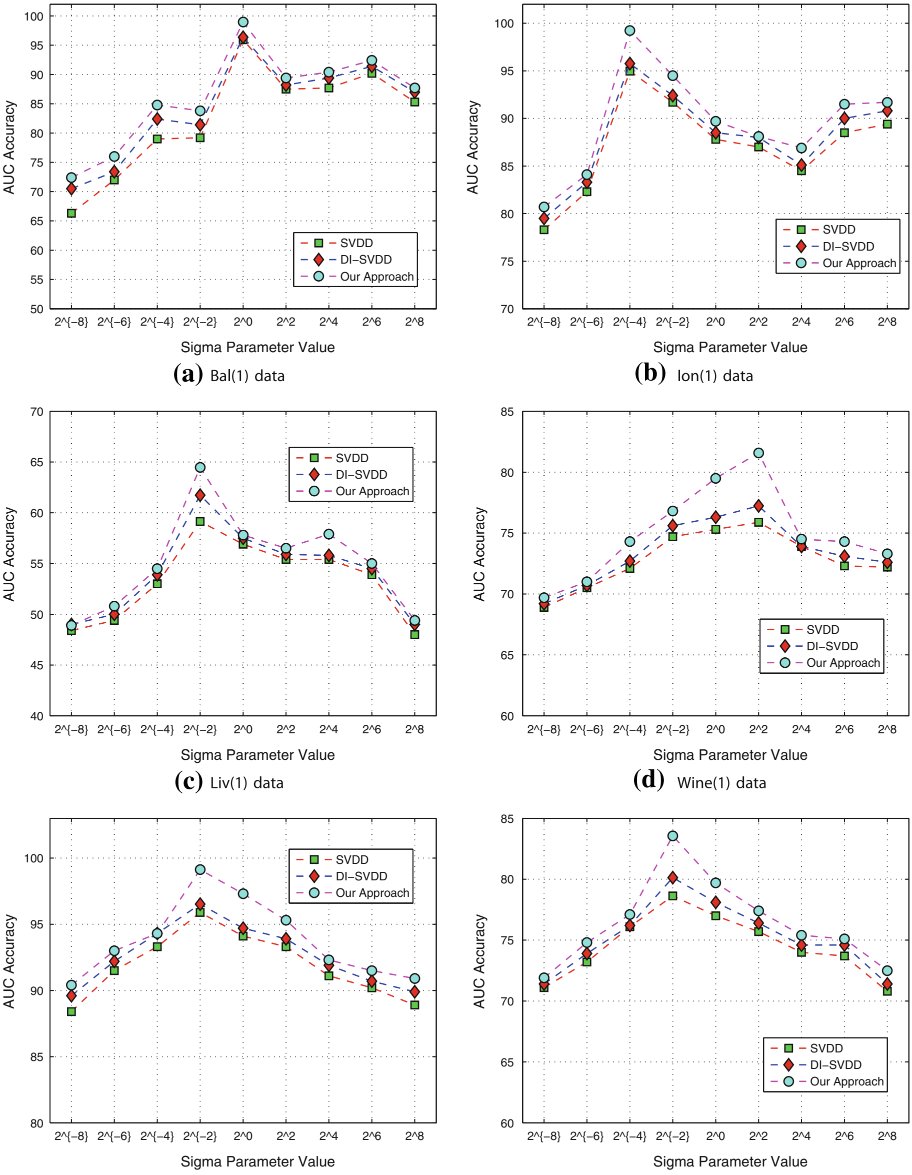
The average testing accuracy and standard deviation of ten times of generations in terms of RBF kernel functions have been illustrated in Table 2. From the table, we can clearly observe that, by introducing a confidence score to each normal data, our approach with the kernel-based class center method consistently yields a better performance in comparison with the SVDD, DI-SVDD, and GMM methods. This is because by defining the confidence score for each normal sample, we can reduce the uncertainty information in samples on the construction of the global outlier detection classifier.

For the standard deviation comparison, it is clear that our approach always obtains less standard deviation than SVDD, DI-SVDD, and GMM for most datasets. This indicates that our proposed approach has a superior capability form outlier detection when compared with SVDD, DI-SVDD, and GMM.

4.3.2 SVDD-based methods comparison under different parameter

Above, we reported the average AUC accuracy of ten times for each method. In order to further compare the performance of SVDD-based methods (i.e., SVDD, DI-SVDD, and ours), we report the AUC accuracy variance of SVDD-based methods under different parameter σ from 2^{-8} to 2^8. For a fixed value of σ, we adjust parameter γ from 2^{-4} to 2^4 to calculate AUC value. Here, we report the results on Bal (1), Ion (1), Liv (1), Wine (1), Ima (1), and KDD-CUP datasets; for other datasets, they show similar results.

Figure 2 illustrates the AUC accuracy of SVDD, DI-SVDD, and our method under parameter γ from 2^{-8} to 2^8. It can be seen that, our method can yield better performance than DI-SVDD and SVDD under each value of parameter σ.



(a) Bal(1) data

(b) Ion(1) data

(c) Liv(1) data

(d) Wine(1) data

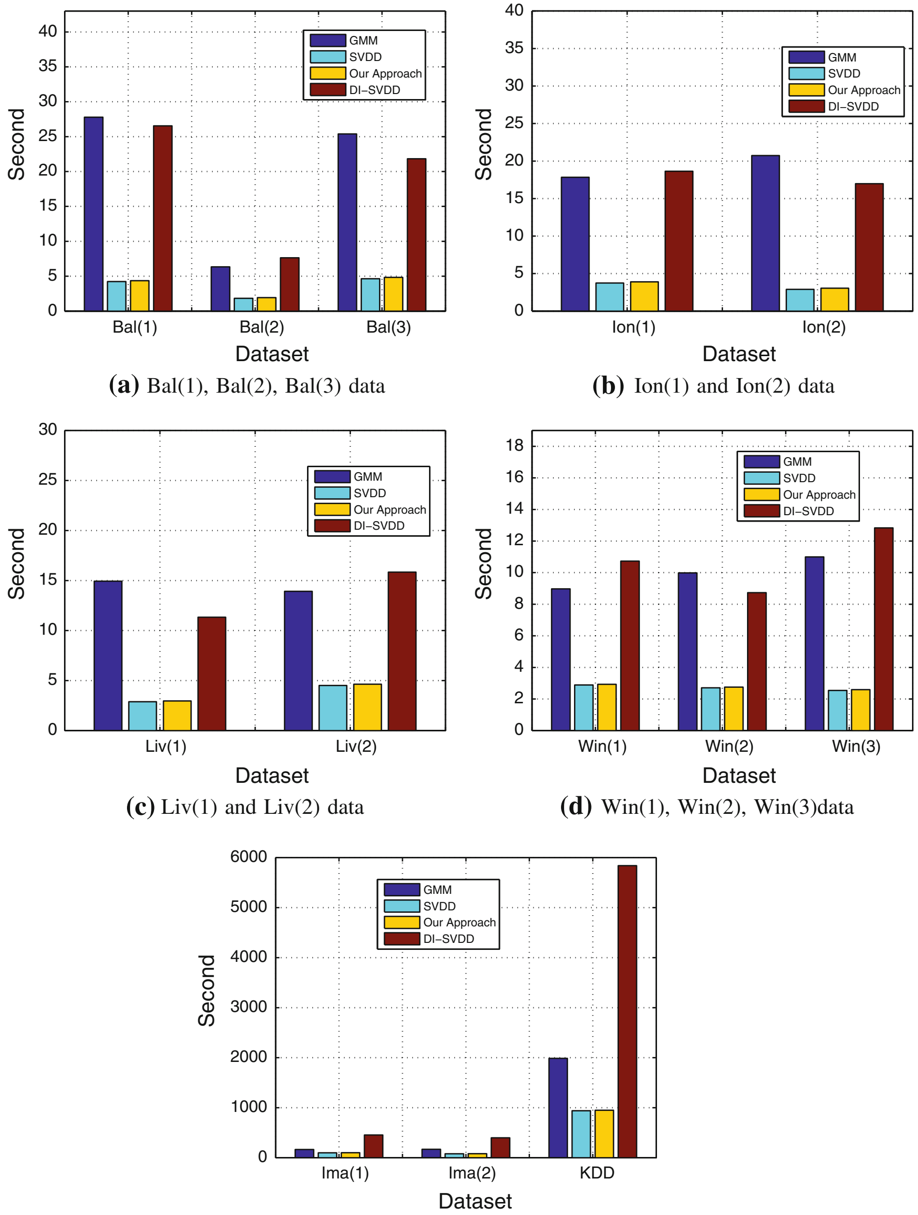
(e) Image(1) data

(f) KDD-CUP data

Fig. 2 AUC accuracy under different parameter σ

4.4 Running time analysis

We further report the average running time of GMM, SVDD, DI-SVDD, and our proposed approach in terms of ten times generations in Fig. 3. We first focus on the running time of SVDD and our approach. As we can see although our proposed approach has to calculate the confidence score for each input normal data, the time cost by our approach is still comparable with that of SVDD. For this fact, we have the following theoretical analysis.



(a) Bal(1), Bal(2), Bal(3) data

(b) Ion(1) and Ion(2) data

(c) Liv(1) and Liv(2) data

(d) Win(1), Win(2), Win(3)data

(e) Image(1), Image(2) and KDD-CUP data

Fig. 3 Average running time of GMM, SVDD, DI-SVDD, and our approach

For optimization problem (6) for SVDD and problem (20) for our method, they are both standard QP problems; therefore, the solving of the two problems has the same computational complexity, that is $O(l^2)$. For our approach, in addition to resolving problem (20), we have to determine the confidence score for each input sample. For the kernel-based class

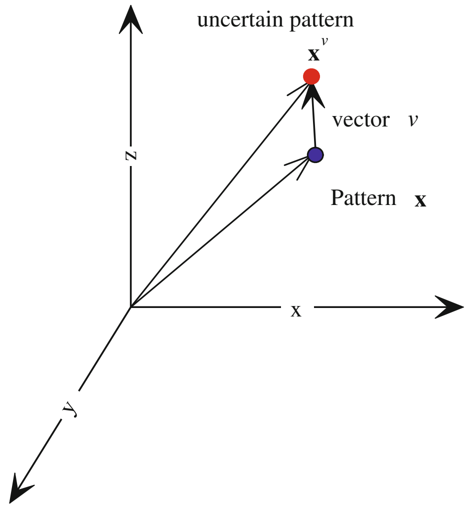


Fig. 4 Illustration of the method used to add the noise to a data example: $\mathbf{x}$ is an original data example, $\mathbf{v}$ is a noise vector, and $\mathbf{x}^{\mathbf{v}}$ is the new data example with added noise. Here, we have $\mathbf{x}^{\mathbf{v}} = \mathbf{x} + \mathbf{v}$

center method, because the complexity of (13), (14) is linear, i.e., $O(l)$, the calculation of the kernel-based distance (12) dominates the time cost of the confidence score generation. We rewrite (12) as follows.

$$\mathrm{Dis}(\phi(\mathbf{x}_j), C^{\phi}) = \sqrt{H_{jj} - \frac{1}{l}\sum_{i=1}^{l} H_{ij} + \frac{1}{l^2}\sum_{i=1}^{l}\sum_{k=1}^{l} H_{ik}}. \tag{31}$$

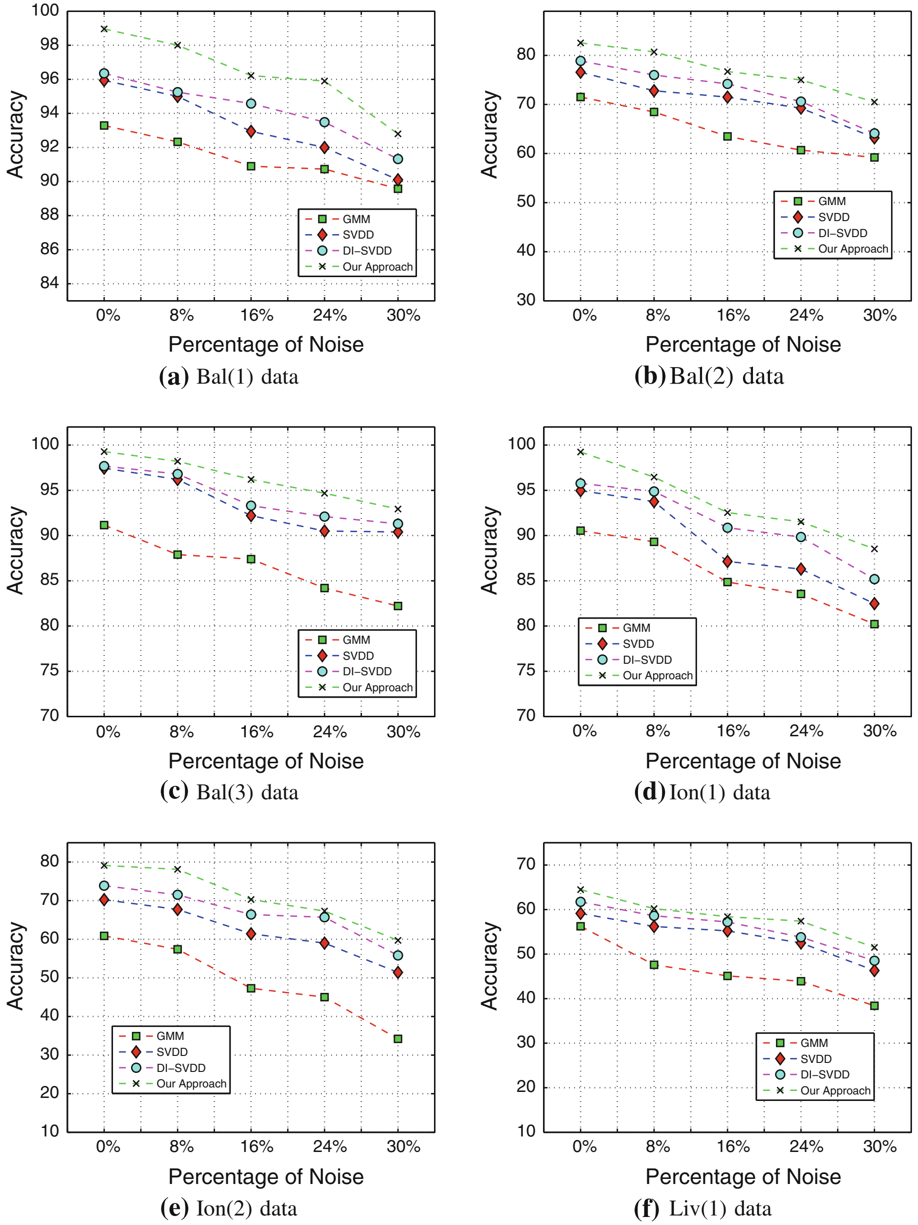
where H is a kernel matrix $H_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$ and $\frac{1}{l^2}\sum_{i=1}^{l}\sum_{k=1}^{l} H_{ik}$ is a fixed value, which just needs to be computed once. In the process of solving the standard QP problems (6) and (20), the calculation of H matrix is the core, which has to be calculated first. Therefore, we calculate H only once to satisfy the kernel-based class center method as well as the problem (20). In this way, our proposed approach displays a comparable running time cost compared with standard support vector data description for outlier detection.

For the GMM clustering method, since it adopts the EM interative strategy to make it converge, it always takes longer time than SVDD and our method. We further discover that DI-SVDD takes more time than SVDD and ours since DI-SVDD requires linearly constrained optimizations by solving a sequence of quadratic programming subproblems.

4.5 Sensitivity to different levels of noise

This set of experiments is conducted to investigate the sensitivity of GMM, SVDD, DI-SVDD, and our approach to different levels of noise added into the input data. Following the method used in [4], we generate the input noise using a Gaussian distribution with zero mean and standard deviation. For each dataset, noise is added to the input data as a vector that has the same dimension as the source data. Figure 4 illustrates the basic idea of the method used to add the noise to a data example.

Specifically, the standard deviation σ_i^0 of the entire data along the ith dimension is first obtained. In order to model the difference in noise on different dimensions, we define the standard deviation σ_i along the ith dimension, whose value is randomly drawn from the range $[0, 2 \cdot \sigma_i^0]$. Then, for the ith dimension, we add noise from a random distribution with



(a) Bal(1) data

(b) Bal(2) data

(c) Bal(3) data

(d) Ion(1) data

(e) Ion(2) data

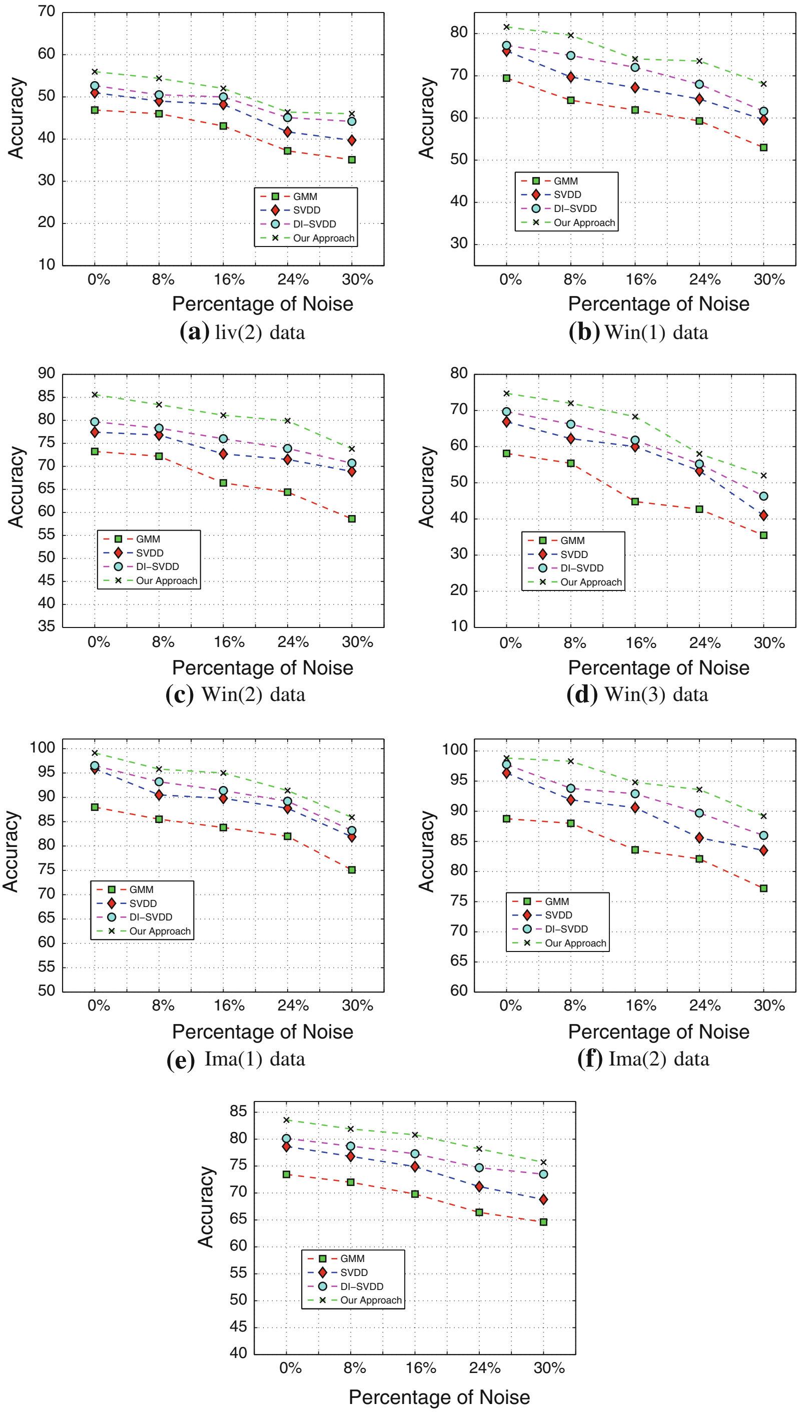
(f) Liv(1) data

Fig. 5 Comparison of AUC accuracy sensitivity to the noise added into the input data

standard deviation σ_i. In this way, a data example $\mathbf{x}_j$ in the target class is added with the noise, which can be presented as a vector

$$\sigma^{\mathbf{x}_j} = [\sigma_1^{\mathbf{x}_j}, \sigma_2^{\mathbf{x}_j}, \ldots, \sigma_{n-1}^{\mathbf{x}_j}, \sigma_n^{\mathbf{x}_j}]. \tag{32}$$

Here, n denotes the number of dimensions for a data example $\mathbf{x}_j$, and $\sigma_i^{\mathbf{x}_j}$ $i = 1, \ldots n$, represents the noise added into the ith dimension of the data example.



(a) liv(2) data

(b) Win(1) data

(c) Win(2) data

(d) Win(3) data

(e) Ima(1) data

(f) Ima(2) data

(g) KDDcup data

Fig. 6 Comparison of AUC accuracy sensitivity to the noise added into the input data

In our experiments, we generate the percentage of data corrupted by noise, which varies from 0 to 30 %, and perform the three methods on these twelve subdatasets. Figures 5 and 6 show the AUC values achieved by the three methods in terms of different percentages of data corrupted by noise. It is clear that, as the percentage of noise increases, the overall performance of the three methods degrades. This occurs because when more noise is involved in the target data, the normal class will become less distinguishable from the outliers. This less distinguishable case indeed reduces the performance of the methods. Nevertheless, we can clearly see that our approach can still consistently yield higher performance than GMM, SVDD, and DI-SVDD. This indicates that, our proposed approach can effectively reduce the effect of noise involved in the input data and significantly improve the learning ability of support vector data description for outlier detection.

5 Conclusion and future work

Outlier detection on uncertain data is challenging and demanding, due to the increase in applications such as fraud detection. This paper has proposed an SVDD-based approach for outlier detection on uncertain data. We first put forward a kernel-based center class method to generate a confidence score to each input sample, which indicates the likelihood of an example tending toward to normal class. This information is thereafter incorporated into the learning procedure of support vector data description to refine the decision boundary of the distinctive classifier. Substantial experiments have demonstrated that our proposed approach performs better than the GMM, SVDD, and DI-SVDD models in terms of performance and sensitivity to noise contained in the input data.

In the future, we would like to address the problem of outlier detection using normal examples and outliers on uncertain data. We also plan to investigate the detection ability of our proposed approach for large stream data.

Acknowledgments We thank the anonymous reviewers for their very useful comments and suggestions. This work is supported by Natural Science Foundation of China (61070033), Natural Science Foundation of Guangdong province (9251009001000005, S2011040004187), Science and Technology Planning Project of Guangdong Province (2010B050400011, 2010B080701070, 2008B080701005), Eleventh Five-year Plan Program of the Philosophy and Social Science of Guangdong Province (08O-01), Opening Project of the State Key Laboratory of Information Security (04-01), China Post-Doctoral First-Class Fund (L1100010).

Appendix

Derivation of Theorem 1:

The inner product of the centroid of hyper-sphere can be rewritten as follows.

$$(\mathbf{o}, \mathbf{o}) = \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j (\mathbf{x}_i, \mathbf{x}_j) \tag{33}$$

(16) is rewritten as

$$\begin{aligned} L(R, \mathbf{o}, \xi) = R^2 + C \sum_{i=1}^{l} m(\mathbf{x}_i)\xi_i - \sum_{i=1}^{l} \alpha_i R^2 - \sum_{i=1}^{l} \alpha_i \xi_i + \sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{x}_i) \\ + \sum_{i=1}^{l} \alpha_i (\mathbf{o}, \mathbf{o}) - 2 \sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{o}) - \sum_{i=1}^{l} \beta_i \xi_i . \end{aligned} \tag{34}$$

According to (17), (18), and (19), we have

$$R^2 - \sum_{i=1}^{l} \alpha_i R^2 = 0 \tag{35}$$

$$C\sum_{i=1}^{l} m(\mathbf{x}_i)\xi_i - \sum_{i=1}^{l} \alpha_i \xi_i - \sum_{i=1}^{l} \beta_i \xi_i = 0. \tag{36}$$

Substituting (35) and (36) into (34) and considering (17), we have

$$\begin{aligned} L(R, \mathbf{o}, \xi) &= \sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{x}_i) + \sum_{i=1}^{l} \alpha_i (\mathbf{o}, \mathbf{o}) - 2\sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{o}) \\ &= \sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{x}_i) + (\mathbf{o}, \mathbf{o}) - 2\sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{o}) = \sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{x}_i) - (\mathbf{o}, \mathbf{o}). \end{aligned} \tag{37}$$

Substituting (33) into (37), we have

$$L(R, \mathbf{o}, \xi) \ = \sum_{i=1}^{l} \alpha_i (\mathbf{x}_i, \mathbf{x}_i) - \sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i \alpha_j (\mathbf{x}_i, \mathbf{x}_j). \tag{38}$$

Therefore, the solution of problem (15) can be resolved by problem (20) subject to (21), (22).

References

1. Abraham B, Box GEP (1979) Bayesian analysis of some outlier problems in time series. Biometrika 66(2):229–236
2. Agarwal C (2005) An empirical bayes approach to detect anomalies in dynamic multidimen-sional arrays. In: Proceedings of the 5th IEEE international conference on data mining. IEEE Computer Society, Washington, DC, USA, pp 26–33
3. Agarwal D (2006) Detecting anomalies in cross-classified streams: a bayesian approach. Knowl Inf Syst 11(1):29–44
4. Aggarwal C (2007) On density based transforms for uncertain data mining. In: Proceedings of IEEE international conference on data mining. IEEE Computer Society, Washington, DC, USA, pp 866–875
5. Aggarwal C (2009) Managing and mining uncertain data. Springer, Berlin
6. Aggarwal C, Yu P (2001) Outlier detection for high dimensional data. In: Proceedings of the ACM SIGMOD international conference on management of data. ACM Press, pp 37–46
7. Aggarwal C, Yu PS (2008) Outlier detection with uncertain data. In: Proceedings of SDM, pp 483–493
8. Aggarwal C, Yu PS (2009) A survey of uncertain data algorithms and applications. IEEE Trans Knowl Data Eng 21(5):609–623
9. Albrecht S, Busch J, Kloppenburg M, Metze F, Tavan P (2000) Generalized radial basis function networks for classification and novelty detection: self-organization of optional bayesian decision. Neural Netw 13(10):1075–1093
10. Barbara D, Couto J, Jajodia S, Wu N (2001a) Detecting novel network intrusions using bayes estimators. In: Proceedings of the first SIAM international conference on data mining
11. Barbara D, Couto J, Jajodia S, Wu N (2001b) Adam: a testbed for exploring the use of data mining in intrusion detection. SIGMOD Rec 30(4):15–24
12. Bi J, Zhang T (2004) Support vector machines with input data uncertainty. In: Proceedings of advances in neural information processing systems (NIPS)
13. Bradley AP (1997) The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn 30(6):1145–1159
14. Breunig M, Kriegel H, Ng R, Sander J (2000) LOF: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD international conference on management of data (SIGMOD), pp 93–104

15. Cheng R, Kalashnikov D, Prabhakar S (2003) Evaluating probabilistic queries over imprecise data. In: Proceedings of ACM SIGMOD
16. Chen D, Shao X, Hu B, Su Q (2005) Simultaneous wavelength selection and outlier detection in multivariate regression of near-infrared spectra. Anal Sci 21(2):161–167
17. Cheng L, Wing HW (2001) Model-based analysis of oligonucleotide arrays: expression index computation and outlier detection. In: Proceedings of the national academy of sciences, USA (98), pp 31–36
18. Dalvi N, Suciu D (2004) Efficient query evaluation on probabilistic databases. VLDB J 16(4):523–544
19. Denton A (2009) Subspace sums for extracting non-random data from massive noise. Knowl Inf Syst 20(1):35–62
20. Eskin E (2008) Anomaly detection over noisy data using learned probability distributions. In: Proceedings of the seventeenth international conference on machine learning, pp 255–262
21. Fan HQ, Zaiane OR, Foss A (2009) Resolution-based outlier factor: detecting the top-n most outlying data points in engineering data. Knowl Inf Syst 19(1):31–51
22. Foss A, Zaiane OR (2011) Class separation through variance: a new application of outlier detection. Knowl Inf Syst 29(3):565–596
23. Guo SM, Chen LC, Tsai JSH (2009) A boundary method for outlier detection based on support vector domain description. Pattern Recogn 42(1):77–83
24. Hido S, Tsuboi Y, Kashima H, Sugiyama M, Kanamori T (2011) Statistical outlier detection using direct density ratio estimation. Knowl Inf Syst 26(2):309–336
25. Hollier G, Austin J (2002) Novelty detection for strain-gauge degradation using maximally correlated components. In: Proceedings of the European symposium on artificial neural networks, pp 257–262
26. Huang HP, Liu YH (2002) Fuzzy support vector machine. IEEE Trans Neural Netw 13(2):464–471
27. Jain AK, Dubes RC (1988) Algorithms for clustering data. Prentice-Hall, New Jersey
28. Jiang SY, An QB (2008) Clustering-based outlier detection method. In: Proceedings of the fifth IEEE international conference on fuzzy systems and knowledge discovery, 429C433
29. King S, King DP, Anuzis KA, Tarassenko L, Hayton P, Utete S (2002) The use of novelty detection techniques for monitoring high-integrity plant. In: Proceedings of the 2002 international conference on control applications (1), pp 221–226
30. Kapil KG, Baikunth N, Ramamohanarao K (2010) Layered approach using conditional random fields for intrusion detection. IEEE Trans Dependable Secur Comput 7(1):35–49
31. Kriegel HP, Pfeifle M (2005) Density-based clustering of uncertain data. In: Proceedings of 11th ACM SIGKDD international conference knowledge discovery in data mining (KDD)
32. Lazarevic A, Ertoz L, Ozgur A, Srivastava J, Kumar V (2003) A comparative study of anomaly detection schemes in network intrusion detection. In: Proceedings of the third SIAM international conference on data mining (SDM), pp 23–34
33. Lee KY, Kim DW, Lee KH, Lee D (2007) Density-induced support vector data description. IEEE Trans Neural Netw 18(1):284–289
34. Mahoney MV, Chan PK (2003) Learning rules for anomaly detection of hostile net- work trafic. In: Proceedings of the 3rd IEEE international conference on data mining. IEEE Computer Society, pp 601–612
35. Matsubara Y, Sakurai Y, Yoshikawa M (2011) D-Search: an efficient and exact search algorithm for large distribution sets. Knowl Inf Syst 29(1):131–157
36. Murphy PM, Aha DW (2004) UCI repository of machine learning database. http://www.ics.uci.edu/~mlearn/MLRepository.html
37. Peterson GL, McBride BT (2011) The importance of generalizability for anomaly detection. Knowl Inf Syst 14(3):377–392
38. Saitoh S (1998) Theory of reproducing kernels and its applications. Longman Scientific & Technical, Harlow
39. Solberg HE, Lahti A (2005) Detection of outliers in reference distributions: Performance of Horn's algorithm. Clin Chem 51(12):2326–2332
40. Shi Y, Zhang L (2011) COID: a cluster Coutlier iterative detection approach to multi-dimensional data analysis. Knowl Inf Syst 28(3):709–733
41. Sun H, Bao Y, Zhao F, Yu G, Wang D (2004) CD-trees: an efficient index structure for outlier detection. In: International conference on web-age information management (WAIM), pp 600–609
42. Tax DMJ, Ypma A, Duin RPW (1999) Support vector data description applied to machine vibration analysis. In: Proceedings of the fifth annual conference of the advanced school for computing and imaging (ASCI), 398C405
43. Tan PN, Steinbach M, Kumar V (2005) Introduction to data mining. Addison-Wesley, Boston
44. Tax D, Duin R (2004) Support vector data description. Mach Learn 54(1):45–66
45. Varun C (2008) Real-time credit card fraud detection. Expert Syst Appl 35(4):1721–1732
46. Vapnik VN (1998) The nature of statistical learning theory. Springer, London

47. Varun C, Arindam B, Vipin K (2009) Anomaly detection: a survey. ACM Comput Surv 41(3):1–58
48. Van Hulse JD, Khoshgoftaar TM, Huang HY (2007) The pairwise attribute noise detection algorithm. Knowl Inf Syst 11(2):171–190
49. Victoria JH, Jim A (2004) A survey of outlier detection methodologies. Artif Intell Rev 22(2):85C126
50. Wang DF, Yeung DS, Tsang ECC (2006) Structured one-class classification. IEEE Trans SMC Part B: Cybern 36(6):1283–1295
51. Williams G, Baxter R, He H, Hawkins S, Gu L (2002) A comparative study of RNN for outlier detection in data mining. In: Proceedings of the 2002 IEEE international conference on data mining. IEEE Computer Society, Washington, DC, USA, pp 709–718
52. Xiao YS et al (2009) Multi-sphere support vector data description for outliers detection on multi-distribution data. In: 2009 IEEE international conference on data mining workshops, pp 82–87
53. Yang WS, Wang SY (2008) A process-mining framework for the detection of healthcare fraud and abuse. Expert Syst Appl 31(1):56–68
54. Yang X, Latecki LJ, Pokrajac D (2009) Outlier detection with globally optimal exemplar-based GMM. In: Proceedings of the 2009 SIAM international conference on data mining (SDM), 145C154
55. Zhang Q, Li F, Yi K (2008) Finding frequent items in probabilistic data. In: Proceedings of ACM SIGMOD

Author Biographies

Bo Liu is with the Faculty of Automation, Guangdong University of Technology. His research interests include machine learning and data mining.

Yanshan Xiao is with the Faculty of Computer, Guangdong University of Technology. Her research interests include multi-instance learning and data mining.

Longbing Cao is a Professor at the University of Technology Sydney, and the Data Mining Research Leader of the Australian Capital Markets Cooperative Research Centre. He got one PhD in Intelligent Sciences and another in Computing Sciences. His research interests include data mining and machine learning and their applications, behavior informatics, multi-agent technology, open complex intelligent systems, and agent mining. He has been leading large government and enterprise data mining projects in many major domains.

Zhifeng Hao received the BSc degree in mathematics from Zhongshan University, Guangzhou, China, in 1990 and the PhD degree in mathematics from Nanjing University, Nanjing, China, in 1995. He is with the Faculty of Computer, Guangdong University of Technology. His current research interests include design and analysis of algorithm, mathematical modeling, and combinatorial optimization.

Feiqi Deng is with the School of Automation Science and Engineering, South China University of Technology. His research interests include control theory and systems engineering, power system stability analysis, stochastic control theory, nonlinear systems and information systems development.