International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

CS-SVDD Based Outlier Detection for Imperfectly Labeled Data

AlkaP.Beldar[Å] and VinodS.Wadne[Å]

[Å]Department of Computer Engineering, Pune University, ICOER, Wagholi,Pune, India

Accepted 20 March 2014, Available online 01 April 2014, **Vol.4, No.2 (April 2014)**

Abstract

Outlier detection is an important problem which has been studied within various application domains and research areas. Most of the previous methods assume that data examples are exactly categorized as either normal class or negative class. However, in many applications data are imperfectly labeled due to various error and noise. These kinds of data can cause system to give output wrong; because the label is either damaged by noise or wrongly labeled so that a normal data behaves like outlier. These kinds of data make outlier detection difficult as compared to clearly separated data. To handle uncertain data one classifier is used i.e. SVDD (model based outlier detection). The propose system work in two steps. In first step we calculate likelihood values or confidence score for each data example of training data, which define the degree of membership towards a positive or normal class. These generated likelihood values for training data are passed to the SVDD classifier to detect outlier. In this phase, the contribution of the examples with the least confidence score on the construction of the decision boundary has been reduced.

Keywords: *Imperfectly labeled data, SVDD classifier*

1. Introduction

Outlier detection approaches to identify a small group of instances which behaves remarkably different from the other existing data. The definition of outlier is given in : an observation which deviate so much from other observation as to arouse suspicious that it was generated by a different works, (D.M. Hawkins *et al*,1980) it gives idea of what outlier exactly is and encourage many outlier detection methods.Outlier detection methods has been used in many application like fraud detection in credit cards, insurance , health care, tax, intrusion detection for cyber-security, fault detection in safety critical system to military surveillance etc (D.M. Hawkins *et al*,1980) Many outlier detection methods have been introduced to detect outlier from existing normal data only. Generally, the outlier detection's previous work can be classified into four categories: distribution (statistical)-based, clustering-based, density based and model based approach (P. Kriegel, M. Schubert, and A. Zimek *et al* 2008 A. Lazarevic, L. Ertoָ z, V. Kumar, A. Ozgur, and J. Srivastava *et al* 2003). Model base approach uses a predictive model to characterized the normal data and then detect outlier as deviation from model (C. Li and W. H. Wong *et al* 2001).

Most of the model based approached implemented consider that input training data are perfectly labeled for building the outlier detection classifier or model. However the collected data may contaminated by noise and causes data with imperfect labels. Because of this imperfect label the normal data may behaves like abnormal data or outlier even though itself may not an outlier. This kind of result is known as uncertain data information and might cause labeling imperfection or errors into the training data, which further limits the accuracy of given outlier detection method . Therefore it is necessary to develop an outlier detection algorithm for handling imperfectly labeled data. Addition to it another important observation is that, outlier also called as negative example although they are very low, but do exist in most of the applications.

2. Literature Survey

2.1 Outlier Detection

Previously, many outlier detection methods have been proposed. Out of that existing approaches are categories as follows: distribution based, clustering based, density based and model-based approach (D.M. Hawkins *et al*,1980 ,Y. J. Lee, Y. R. Yeh, and Y. C. F. Wang *et al* 2012 ,E. Eskin *et al* 2000).The statistical approach assumes data follows some predefined distribution and aims to find out outlier which deviates from such distribution. Most of the time the data distribution is not known previously, especially for highly dimensional data.

Clustering based approach (S. Y. Jiang and Q. B. An *et al* 2008.) always apply a clustering based method on sample of data to characterize the local behaviors' of the data. The performance of outlier detection is limited because, the clustering based approaches are unsupervised, it won't require any labeling of data.

Another approach, density based approach (M. Breunig, H.-P.Kriegel, R.T. Ng, and J. Sander *et al* 2000) i.e. local outlier detection (LOF), determines degree of outlierness of each data instance based on its local density.

*Corresponding author: **AlkaP.Beldar**

This approach identifies the data structure via density estimation. In this approach we have to calculate the distance between each data instance and all other data instances, it causes high computational complexity.

After all work done previously; a model based approach is introduced. Support vector data description (SVDD)(D. M. J. Tax and R. P.W. Duina *et al* 2004, M. J. Tax, A. Ypma, and R. P. W. Duin *et al* 1999.) is considered to be more powerful for detecting outliers in various domains. SVDD constructs a little sphere around the normal data and use this sphere to detect unknown sample as outlier or normal one. SVDD transforms the original data into a feature space via a kernel function to detect outlier in high dimensional data. But its performance is affected by noise involved in the input data.

Although much progress is done in outlier detection, but most of the methods does not come with the problem of imperfectly labeled or negative data example. Our proposed approach gathers all local data information by creating likelihood values of each input data example toward the positive or negative class. This generated information is the incorporate into SVDD framework for enhancing global classifier to detect outlier.

In this paper we are overcoming the drawback of previous paper work (Liu, Y. Xiao, L. Cao, Z. Hao, and F. Deng *et al* 2014).The work done in (Liu, Y. Xiao, L. Cao, Z. Hao, and F. Deng *et al* 2014) U-SVDD addresses detection of outlier only by using normal data and not by considering negative example in account .U-SVDD calculates the degree of membership of example towards the positive class only .However, this paper addresses problem of outlier detection with few negatively labeled examples and takes data with imperfectly labeled data into account. On the basis of detected problem we introduced single likelihood model to assign likelihood values to each data examples on their local behaviors .In single likelihood model, example including positive and negative classes are assigned likelihood values indicating degree of membership towards its own class.

2.2 Support Vector Data Description

The most attractive feature of SVDD is that it can transform the input data into a feature space and detect global outliers effectively as illustrated in Fig. 1. (B)

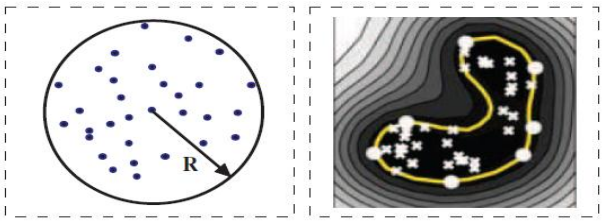

Figure 1: (A): Shows SVDD hyper-sphere in feature space. (B): Shows SVDD decision boundary in input space

The outliers are typically scattered around normal data so that the distribution of the negative class cannot be well represented by the very few negative training examples. To solve this problem, we can use SVM algorithm, but the false positive and false negative costs are usually unknown to us in real life applications. Therefore, we will use SVDD method for outlier detection, which gives decision boundary around the normal data, and uses the few negative examples to refine the boundary to build an outlier detection classifier.

The support vector data description (SVDD) has been proposed for one-class classification learning. Given a set of target data

$$\{x_i\}, i = 1, \ldots., lwherex_i \epsilon R^n$$

The basic idea of SVDD is to find a minimum hyper-sphere that contains most of target data in the feature space, as shown in Fig 2. (A)

$$\text{Min} \quad F(R, 0, \xi_i) = R^2 + C\sum_{i=1}^{l}\xi_i, \tag{1}$$

$$s.t. \|\phi(x_i) - 0\|^2 \leq R^2 + \xi_i, \tag{2}$$

$$\xi_i \geq 0,$$

where $\phi(.)$ is a mapping function which maps the input data from input space into a feature space, and $\phi(x_i)$ is the image of xi in the feature space, $\xi_i$are slack variables to allow some data points to lie outside the sphere, and C >0 controls the tradeoff between the volume of the sphere and the number of errors.

$\sum_{i=1}^{l}\xi_i$is the penalty for misclassified samples. By introducing Lagrange multiplier s_i, the optimization problem (2) is transformed into:

$$\max\sum_{i=1}^{l}\alpha_i K(x_i, x_i) - \sum_{i=1}^{l}\sum_{k=1}^{l}\alpha_i\alpha_k K(x_i, x_i)$$

$$s.t. \quad 0 \leq \alpha_i \leq C,$$

$$\sum_i \alpha_i = 1 \tag{3}$$

in which kernel function $_{K(.,.)}$ is used to calculate the inner pair wise product of two vector $\phi(x_i)$ and $\phi(x_j)$,that is $K(x_i,x_j)=\phi(x_i)*\phi(x_j)$. The samples with $\alpha_{i>}0$ are support vectors (SVs). For a test point x, it is classified as normal data when this distance is less than or equal to the radius *R*. Otherwise, it is flagged as an outlier.

2.3 Imperfectly Labeled Data

The difference between outlier detection problem and imbalanced data classification is that in outlier detection the outlier are scattered around the normal data so that distribution of negative class cannot be well represented by very few negative training examples while in imbalanced data classification examples of one or more minority classes are often self-similar leads to form compact cluster.

3. Proposed Work

This section provides a detailed description about our proposed system to outlier detection. Outlier detection refers to the problem of determining data objects that are markedly different from or inconsistent with the remaining set of data. We will be going to use abalone training dataset for outlier detection. This training dataset consists

of normal examples and small amount of outlier (or abnormal) examples. Our objective is to build a classifier which consider normal and abnormal training data and classify the unseen test data. In our project we are going to use support vector data description (SVDD) classifier. So the first step will be to generate pseudo training dataset by calculating likelihood values for each input data. We will use kernel k-means clustering algorithm to generate likelihood values for each input data. Afterwards we will apply SVDD on likelihood values and it will classify the test data into normal and abnormal class.

Support vector machine (SVM) is another method for outlier detection. So at the end we will compare the SVDD and SVM method for outlier detection on the basis of the performance accuracy.

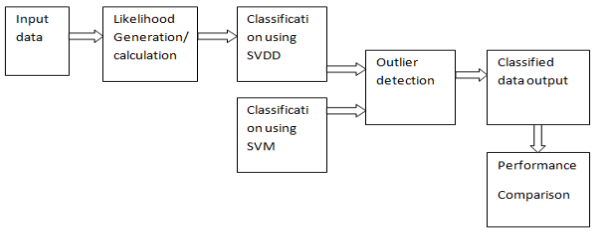

Figure 2: Proposed system for outlier detection

We consider a set of training data *S* which consists of *l* normal examples and a small amount of *n* outlier (or abnormal) examples. Our objective is to build a classifier using both normal and abnormal training data and the classifier is thereafter applied to classify unseen test data. However, subject to sampling errors or device imperfections, a normal example may behave like an outlier, even though the example itself may not be an outlier. Such error factors might result in an imperfectly training data, which makes the subsequent outlier detection become grossly inaccurate. To deal with this problem, we will calculate likelihood for each input data example and then we calculate SVDD for these likelihood values.

3.1 Likelihood Value Generation Algorithm

The main purpose of this algorithm is to generate pseudo training dataset by calculating likelihood value for each instances of training dataset. In this, generated pseudo training dataset consist of *l* normal instances an *n* abnormal instances. The basic idea of this algorithm is to capture local uncertainty by calculating likelihood values by using kernel k-mean clustering algorithm.

The likelihood value generation algorithm

Input: Training data $xi, 1 \le i \le l+n$

Output: pseudo training data set $(xi, m(xi))$

Procedure:

1. Using kernel k-means algorithm form local clusters.

$$J = \sum_{i=1}^{k} \sum_{j=1}^{l+n} (||\phi(xj) - \phi(vi)||)^2 \quad (4)$$

Where vi<-center of i^{th} cluster

K<-cluster number

2. Calculate likelihood values for instances belonging to normal class and abnormal class of same cluster i.e. "j"

$m^l j = \frac{l^p j}{l^p j + l^n j}$<-likelihood values of example toward normal class

$m^n j = \frac{l^n j}{l^p j + l^n j}$<-likelihood values of example toward abnormal class

Where $l^p j$<- normal example in j^{th} cluster

$l^n j$<- abnormal example in j^{th} cluster

3. If cluster only contains normal example $m^l j$ of each example in same cluster equals to 1 and their corresponding $m^n j$ is equivalent to 0.

4. Get $(xi, m(xi))$ value for each example in training data set and this is known as pseudo training data.

Return pseudo training data set $(xi, m(xi))$

At the end it will return a training data set in which each instances having its own likelihood value indicating degree of membership towards its own class.

3.2 Classifier Construction Algorithm

Proposed approach uses $m^t(x_i)$ and $m^n(x_j)$ as a membership functions for normal instances and abnormal instances respectively. And place normal (positive) instances into P class having only $m^t(x_i)$ value, abnormal instances (outlier) into N class having only $m^n(x_j)$ value. SVDD can be achieved by

$$\min F = R^2 + C_1 \Sigma m^t(x_i)\varepsilon_i + C_2 \Sigma m^n(x_j)\varepsilon_j \quad (5)$$
$$s.t\ ||x_i - o||^2 \le R^2 + \varepsilon_i, x_i \epsilon\ positive\ class$$
$$||x_i - o||^2 \ge R^2 - \varepsilon_j, x_j \epsilon\ negative\ class$$

where C1 and C2controls tradeoff between error and sphere volume. Parameter ε_i and $\varepsilon_j$are defined as measure of error.

For solving optimization problem (1) ,introduce Lagrange multiplier

$$L = R^2 + C_1 \Sigma m^t(x_i)\varepsilon_i + C_2 \Sigma m^n(x_j)\varepsilon_j -$$
$$\Sigma \alpha_i^t (R^2 + \varepsilon_i - ||x_i - o||^2) - \Sigma \beta_i^t \varepsilon_i - \Sigma \beta_j^n \varepsilon_j -$$
$$\Sigma \alpha_j^n (||x_j - o||^2 - R^2 - \varepsilon_j) \quad (6)$$

Applying partial derivatives to L with respect to R,o,ε_i , ε_j equal to zeros respectively and get,

$$\frac{\delta L}{\delta R} = 0 \rightarrow \alpha_i^t - \alpha_j^n = 1 \quad (7)$$

$$\frac{\delta L}{\delta o} = 0 \rightarrow \Sigma \alpha_i^t (o - \phi(xi)) = \Sigma \alpha_j^n (o - \phi(xj)) \quad (8)$$

$$\frac{\delta L}{\delta \varepsilon_i} = 0 \rightarrow \alpha_i^t + \beta_i^t = C_1\, m^t(x_i) \quad (9)$$

$$\frac{\delta L}{\delta \varepsilon_j} = 0 \rightarrow \alpha_j^n + \beta_j^n = C_2\, m^n(x_j) \quad (10)$$

Replacing (7),(8),(9),(10) into equation (6) ,we get equation (11) and set $\alpha_i = \alpha_i^t (i = 1,2,3, \dots l), \alpha_i = \alpha_i^n (i = l+1, l+2, l+3, \dots l+n), C_i^m = C_1\, m^t(x_i) (i = 1,2,3, \dots l)$

and $C_i^m = C_2\, m^n(x_i) (i = l+1, l+2, l+3, \dots l+n)$.

$\max \sum_{i=1}^{l+n} \alpha_i K(x_i, x_i) - \sum_{i=1}^{l+n} \sum_{j=1}^{l+n} \alpha_i \alpha_j K(x_i, x_j)$ (11)
s.t $0 \le \alpha_i \le C_i^m \; i = 1,2 \ldots, l+n$
$\sum_{i=1}^{l+n} \alpha_i = 1$
$where\ \alpha_i \ge 0\ and\ \alpha_j \ge 0\ are\ Lagrange\ multipliers.\ C_i^m = C_1\ m^t(x_i)(i = 1,2,3,..l)$
and $C_i^m = C_2\ m^n(x_i)(i = l+1, l+2, l+3, ..l+n)$.
Get Lagrange multiplier by solving dual problem which gives centroid of minimum sphere as linear combination of x_i .
For αi (*1<=i<=l+n*) with $\alpha i \neq 0$ patterns known as support vectors.
$o = \sum_{i=1}^{l+n} \alpha i\ \phi(xi)$ (12)

Decision boundary construction

Obtain radius of decision hyper plane by Karush-Kuhn-Tucker Conditions. Assume that x is lying on the surface of hyper sphere, then R can calculated as

$$R^2 = \left(||x - o||\right)^2 = K(x,x) + K(o,o) - 2K(x,o) \quad (13)$$

$$= K(x,x) + \sum_{i=1}^{l+n} \sum_{k=1}^{l+n} \alpha i\ \alpha k \phi(xi)\ K(xi, xk) - 2\sum_{i=1}^{l+n} \alpha i\ K(xi, x)$$

To classify test point "x" ,calculate distance to centroid of hyper sphere. If distance is less than or equal to R i.e.

$$R^2 \ge \left(||x - o||\right)^2 \quad (14)$$

Point x is accepted as normal instance. Otherwise it is detected as outlier.
The classifier construction algorithm
Input: Pseudo training data set $(xi, m(xi))$,$1 \le i \le l+n$
Output: α_i , $1 \le i \le l+n\ and\ R$
Procedure:
1. Resolve standard QP problem of (11).
2. Obtain α_i for each instances.
3. Determine instances whose $\alpha i \neq 0$, that sample resides on surface of the hyper sphere.
4. Obtain radius of decision hyper plane by Karush-Kuhn-Tucker Conditions
$R^2 = \left(||xj - o||\right)^2$
Return α_i , $1 \le i \le l+n\ and\ R$

4. Results and Discussion

System conducts the experiments to evaluate the performance of proposed system on Abalone and Page Block dataset i.e. real life data set.

4.1 Dataset Description

In our experiments, real life datasets are used i.e. UCI abalone training data set for outlier detection. This training data set consists of normal examples and small amount of outlier (or abnormal) examples. To perform outlier detection with very few abnormal data, we randomly select 50 percent of normal data and a small number of abnormal data for training, such that 95 %t of the training data belong to the positive class and only 5 % belong to the negative class.. The abalone dataset having total 4177 data instances with 8 attributes with 29 classes. For experiment first 8 classes are considered as normal and rest classes data used for testing.
For all the algorithms, the Gaussian RBF kernel is used
$K(x, xi) = exp(-||x - xi||2/2\sigma 2)$.
The value for parameter σ in RBF kernel function is in range from 2^{-3} to 2^4 .And for parameter C in SVDD, as well as C1 and C2 is from 2^0 to 2^4.The number of k for kernel k-mean is varied from 2 to$(l+n)/2$.

4.2 Experimental Setup

These experiments were conducted on 2.53 Intel(R) Core(TM) i3 Processor with 2 GB of RAM, and running on Windows 7 operating system. All algorithms were implemented in VC++ language and Microsoft Visual Studio 10 applied on Abalone datasets to evaluate the performance of the algorithms.

4.3 Evaluation Criteria

Proposed system results will be compared with the following performance matrices and results are verified.
1. Accuracy of Detection of outliers with normal data in training for CS-SVDD and SVM classifier. Accuracy of both classifiers is to detect outliers in normal conditions.
2. Detection of outlier for mislabeled data. Detection of normal or outlier data even with mislabeled training data.
3. Detection of outlier in noisy data. Detect if outlier with noisy training data.

4.4 Confusion Matrix

Evaluation of performance of outlier detection method can be done using two terms: Detection rate and false alarm rate. Detection rate gives number of outlier detected correctly and false alarm rate gives number of outlier misclassified as a normal data example.
Detection rate= $\frac{TP}{TP+FN}$, False alarm rate= $\frac{FP}{FP+TN}$

Table 1: Confusion matrix

		Actual Label	
		Target class	Negative class
Predicted label	Target class	TruePositive (TP)	FalseNegative (FN)
	Negative class	True Positive (FP)	TrueNegative (TN)

4.5 Performance Evaluation

To evaluate the performance proposed system randomly selects 50% of data from first 8 classes and out of 50% made 95% normal data and 5% abnormal data to generate training dataset. And rest of data used as testing dataset. Fig.3 shows how much better results are produced by proposed system.

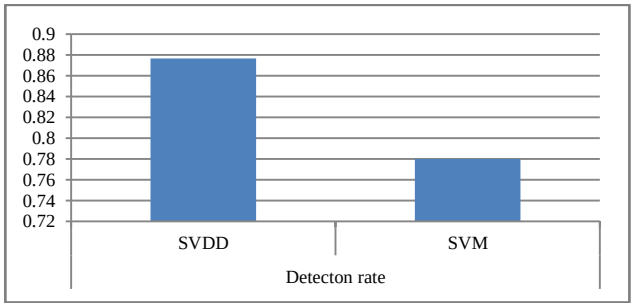

Figure 3: Shows the outlier detection for the outlier detection classifier for Abalone data set

4.6 Average Running Time

Figure 4: Shows average running time of the two methods for Abalone dataset

As system compared the performance ,sensitivity to noise and impact of mislabeled data of two methods , similarly we are comparing the average running time of the two methods shown in Fig.4.It is observed that cs-svdd method takes more time as it calculate likelihood values by using kernel k-mean algorithm.

4.7 Performance to Error Label

To measure performance of proposed system flip the labels of normal instance to its opposite class. In experiment mislabel data to 3%, 5%,6% and 9 % in training dataset and apply proposed system. Figure 5 shows that proposed system's performance is much better as compare to CS-SVM.

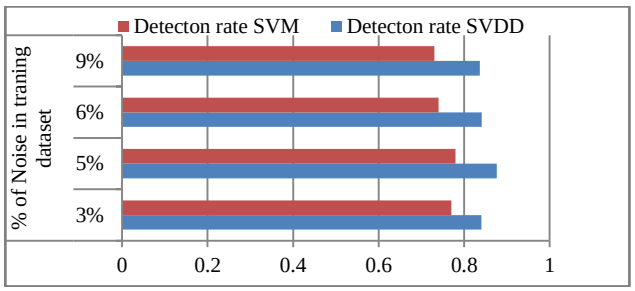

Figure 5: Performance comparison under different percentage of data with error label

4.8 Performance to Noisy Data

To measure performance of proposed system generate noise using Gaussians distribution with zero mean.In experiment corrupt data by introducing noise to 8%, 16%,24% and 30 % in training dataset and apply proposed system. Fig.6 shows that proposed system's performance is much better as compare to CS-SVM.

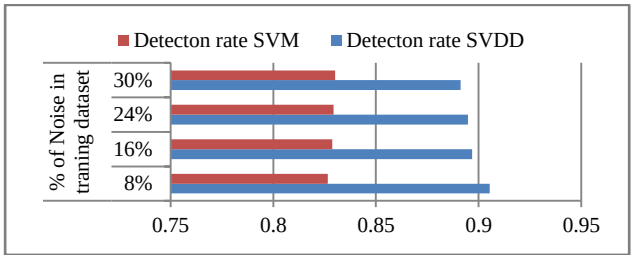

Figure 6: Performance comparison under different percentage of training data corrupted by noise

Conclusion

In this paper, we propose SVDD method for outlier detection by calculating likelihood values for each input data into the SVDD training dataset. Our proposed method will first calculates the local uncertainty by computing likelihood values for each example based on its local data behavior in the feature space and then it will builds global classifiers for outlier detection by considering the negative examples and the likelihood values in the SVDD-based learning framework. We will consider real life data sets for proposed approach and will be going to achieve a better tradeoff between detection rate and false alarm rate for outlier detection in comparison to SVM based outlier detection.

Future Scope

We plan to extend our work by applying SVDD based outlier detection for another dataset i.e. pageblock dataset and we will calculate the result for same. Also to use kernel LOF based likelihood values generation and pass end this to classifier. We will try to work in streaming this environment.

References

D.M. Hawkins1980, Identification of Outliers. Chapman and Hall,.
M. Breunig, H.-P.Kriegel, R.T. Ng, and J. Sander, , 2000 LOF: Identifying Density-Based Local Outliers, Proc. ACM SIGMOD Int'l Conf. Management of Data.
V. Chandola, A. Banerjee, and V. Kumar, , 2009 Anomaly Detection: A Survey, ACM computing Surveys, vol. 41, no. 3, pp. 15:1-15:58.
L. Huang, X. Nguyen, M. Garofalakis, M. Jordan, A.D. Joseph, and N. Taft, , 2007 In-Network Pca and Anomaly Detection, Proc. Advances in Neural Information Processing Systems 19.
H.-P. Kriegel, M. Schubert, and A. Zimek, 2008., Angle-Based Outlier Detection in High-Dimensional Data, Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and data Mining
A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava, 2003 A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection, Proc. Third SIAM Int'l Conf. Data Mining.
X. Song, M. Wu, and C.J., and S. Ranka, May 2007, Conditional Anomaly Detection, IEEE Trans. Knowledge and Data Eng., vol. 19, no. 5, pp. 631-645.
C. Li and W. H. Wong, 2001. Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *In Proceedings of the National Academy of Sciences USA*, 98:31–36.
Y. J. Lee, Y. R. Yeh, and Y. C. F. Wang, 2012. Anomaly detection via online over-sampling principal component analysis.*IEEE Transactions on Knowledge and Data Engineering*.
E. Eskin, 2000. Anomaly detection over noisy data using learned probability distributions. *International Conference on Machine Learning (ICML)*, pages 255–262.
F. Chen, C. T. Lu, and A. P. Boedihardjo, 2010.Gls-sod: a generalized local statistical approach for spatial outlier detection. *ACM SIGKDDInternational Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1069–1078
S. Y. Jiang and Q. B.An, 2008. Clustering-based outlier detectiionmethod.*ICFSKD*, pages 429–433.
D. M. J. Tax and R. P.W. Duin, 2004. Support vector data description. *Machine Learning*, 54(1):45–66.
D. M. J. Tax, A. Ypma, and R. P. W. Duin, 1999.Support vector data description applied to machine vibration analysis. *In ASCI*, pages 398–405
B. Liu, Y. Xiao, L. Cao, Z. Hao, and F. Deng. 2013 Svdd-based outlier detection on uncertain data.*Knowledge and Information Systems*, 34(3):597–618.
Uci machine learning repository: http://archive.ics.uci.edu/ml/datasets.html.Max Welling Kernel K-means and Spectral Clustering
Inderjit S. Dhillon,YuqiangGuan,BrianKulis 2004 Kernel k-means, Spectral Clustering and Normalized Cuts. ACM 1-58113-888-1/04/0008