A Comparative Evaluation of SOM-based Anomaly Detection Methods for Multivariate Data

[1, 2]Bingjun Guo, [1]*Lei Song, [1, 2]Taisheng Zheng, [1, 2]Haoran Liang, [1] Hongfei Wang
[1]Key Laboratory of Space Utilization, Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
Beijing, China
songlei@csu.ac.cn

***Abstract*—Anomaly detection for multivariate data is of vital importance in academic research and industry. In real scenes, there is usually a lack of labels of anomalies. Self-Organizing Map (SOM) can map data to the output layer and maintain the original topology, which has been used as a semi-supervised learning method to solve the above problem. In this paper, we first explain the mechanism of classic SOM for anomaly detection, then compare it with two variants of SOM named kernel SOM and K-BMUs SOM. Kernel SOM replaces Euclidean distance with kernel functions, while K-BMUs SOM changes the number of matching neurons. The three types of SOM are applied to multivariate datasets in three different domains. We find that the performance of the three SOM-based methods is related to the characteristics of data.**

Keywords-SOM; anomaly detection; kernel SOM; K-BMUs SOM

I. INTRODUCTION

The detection of anomaly samples within datasets has always attracted much attention. This process is usually called anomaly detection. The earliest definition of anomaly detection was proposed by Grubbs[1] in 1969. Outliers usually deviate significantly from other members. There are two main characteristics of anomaly: a) Anomaly samples are different from the norm in features; b) compared with normal samples, the number of anomaly samples is infrequent in a dataset.

A. Related Work

Anomaly detection methods can be divided into physics-of-failure (PoF) and data-driven methods. PoF methods establish a system-related physical model, then monitor variables of the model and compare them with the calculated values from the model [2]. When the deviation between monitored values and calculated values exceeds a predetermined threshold, anomalies are identified. PoF methods have been widely used. However, domain knowledge is needed to propose and apply PoF methods. And these methods cannot find some important features of monitored values. Data-driven methods are more superior in cross-domain application and feature learning.

Data-driven methods can be classified into supervised learning, semi-supervised learning and unsupervised learning methods. Supervised learning methods require datasets with complete labels. A classifier is constructed using both normal and anomaly samples with their labels, then this classifier can be used for anomaly detection. The commonly used methods are Support Vector Machines (SVM) [3] and K-Nearest Neighbor (KNN) [4]. Unsupervised learning methods do not need labels of instances. This kind of methods divides data into different classes by the means of analyzing the features of data. K-means [5] and Local Outlier Factor (LOF) [6] are commonly used as unsupervised methods.

In real scenes, there are plenty of normal samples, while abnormal samples are difficult to acquire for training models. In this case, semi-supervised learning method plays a more significant role in anomaly detection. Semi-supervised anomaly detection methods only use normal samples in the training process, and can detect anomalies based on the difference of features. Representative semi-supervised methods include SOM and one-class SVM [7].

Self-Organizing Map (SOM) [8] is a representative method of competitive learning. It can learn and maintain the topological relationships of training data. Since SOM was proposed, it has received extensive attention and been applied in image segmentation, text categorization, anomaly detection and other fields. With further research, researchers have made some changes to traditional SOM according to their needs. Kernel method has been used to induce a novel distance measure which replaces Euclidean distance measure in SOM [9]. SOM with kernel function has been validated on Iris and Wine Recognition datasets. To reduce the effect of noise, KNN is combined with SOM [10] to select several matching neurons in competitive layer for each input.

There are different kinds of data in anomaly detection, such as multivariate tabular, graphs, and time series data. This study focuses on the anomaly detection of multivariate data.

B. Contribution

The contribution of this paper is as follows: a) Summarize a variety of SOM-based anomaly detection methods and compare the performance of each method on multiple datasets. b) Use SOM-based methods for semi-supervised anomaly detection to solve the problem of imbalanced samples in real scenes.

C. Outline

The rest of this paper is organized as follows. Section II describes the principle of SOM and its application in anomaly detection. Section III introduces two variants of SOM-based

Open foundation of key laboratory of space utilization, technology and engineering center for space utilization Chinese academy of sciences (CSU-QZKT-2018-09). Open project of Beijing key laboratory of measurement and control of mechanical and electrical system (KF20181123205).
978-1-7281-0860-5/19/$31.00 ©2019 IEEE

methods. Section IV contains datasets and experiments. Section V is the conclusions and the work to do in the future.

II. SELF-ORGANIZING MAP

In this part, the theory of SOM and its working mechanism in anomaly detection is presented.

A. Theory of SOM

SOM is a kind of neural network which maps points from input space to output space, and maintains information of topological relationships.

There are two layers in the model of SOM: input layer and competitive layer. The competitive layer is also called output layer. Input layer is a group of one-dimensional neurons. Competitive layer is usually a two-dimensional plane. In the competitive layer, each neuron is connected to its nearest neurons. Between input layer and competitive layer, neurons are fully connected. The connections are called weight vectors. The architecture of SOM is shown in Fig. 1.

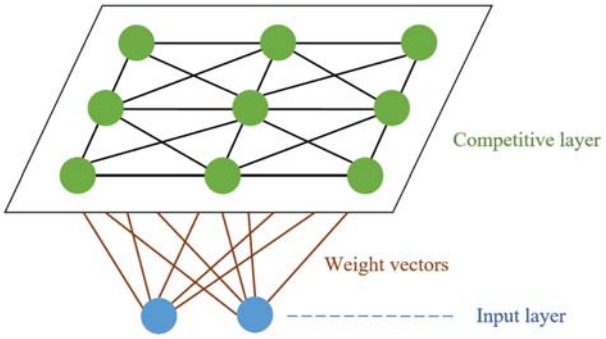


Figure 1. The structure of SOM

The purpose of SOM is to mimic the input signal pattern with the weight vectors indirectly. The learning process of SOM is composed of two parts, which are as follows.

1) Find the best matching unit

Each input pattern $\mathbf{x}$ is described as $\mathbf{x} = (x_1, x_2, ..., x_n)^T$. The weight vector of each competitive neuron j is represented as $\mathbf{w}_j = (w_{1j}, w_{2j}, ..., w_{nj})^T$. The similarity of input pattern and competitive neurons is measured by Euclidean distance. Smaller Euclidean distance indicates higher similarity. The competitive neuron whose Euclidean distance with input pattern is the smallest will be the best matching unit (BMU). BMU is also called the winning neuron.

$$\| \mathbf{x} - \mathbf{w}_{j^*} \| = \min \| \mathbf{x} - \mathbf{w}_j \| \tag{1}$$

$j = 1,2,...,m$, m is the number of neurons in competitive layer. j^* is the BMU.

2) Update weight vectors

When the BMU is found, the next step is to update weight vectors. It should be noted that we not only update the weight of BMU, but also its neighbors'. The neighborhood function can be Gaussian function, Mexican hat function or other reasonable functions.

As learning progresses, the number of neighbors shrinks and only BMU is left at last. The value of the decay function $\eta(t)$ decreases as iterations increase.

$$h(j^*, j) = \eta(t) \exp(- \| \mathbf{w}_{j^*} - \mathbf{w}_j \|^2 / 2\sigma^2) \tag{2}$$

j^* is BMU. j is its neighborhood neuron.

Weights updating is based on the following formula:

$$\mathbf{w}_j(t+1) = \eta(t)[(\mathbf{x} - \mathbf{w}_j(t)] + \mathbf{w}_j(t) \tag{3}$$

j is the winning neuron and its neighbors, and t represents the number of iteration.

SOM is trained iteratively until it reaches the max iteration. When the learning process finished, neurons in the competitive layer will cluster based on their distance from each other. Fig. 2 shows the training process.

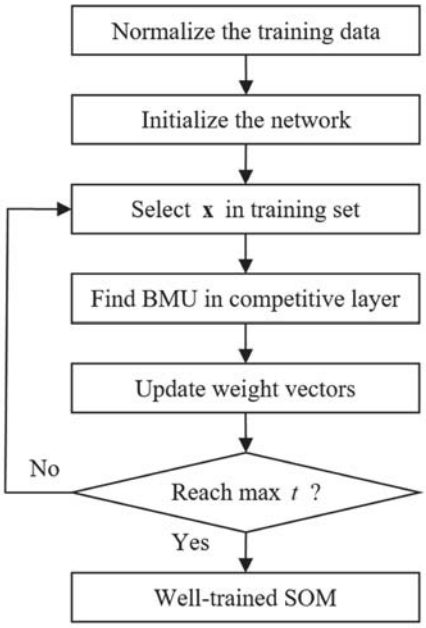


Figure 2. The training process of SOM

B. SOM for anomaly detection

SOM can be used in anomaly detection as a semi-supervised learning method. Normal samples are used to train SOM, so the weights of the trained SOM can mimic the normal patterns in training sets. When input is test set, normal samples in test set is more similar to some competitive neurons of SOM, while anomalies have almost no similarity with the neurons.

Distance is a widely used measure of similarity. If the distance between two vectors is small, we usually think they are similar to each other. Given a trained SOM, the winner neuron

(BMU) for input sample x is neuron $j*$. The quantization error (QE) is computed with $e = d(\mathbf{x}, \mathbf{w}_{j^*})$, (where d is the distance, usually Euclidean distance) [11]. QE is the distance between $\mathbf{x}$ to its most similar neuron in the competitive layer. Therefore, QE reflects the similarity of input $\mathbf{x}$ and the neurons in the trained competitive layer. If $\mathbf{x}$ deviates markedly from the patterns in training set, its QE will be much larger than other samples.

III. KERNEL SOM AND K-BMUS SOM

This section presents two variants of SOM and their use in anomaly detection.

A. Kernel SOM

In classic SOM, the similarity between input sample $\mathbf{x}$ and the competitive neurons depends on their Euclidean distance. When the samples are irregularly or highly non-linearly distributed in the input space, it is difficult to distinguish the normal and anomalies by classic SOM. Moreover, classic SOM is also sensitive to noise and outliers. Therefore, kernel method provides the possibility to solve the above problems.

If samples are linear non-separable in the input space, nonlinear mapping ϕ from kernel methods can be used to transform this nonlinear problem to a linear solvable problem in a high-dimensional feature space V.

Define the non-linear mapping $\phi : \mathbf{x} \to \phi(\mathbf{x})$, where $\mathbf{x} \in X$, $\phi(\mathbf{x}) \in V$. X is the sample set. V is the feature space. The Euclidean distance can be replaced by the following objective function [8].

$$J(\mathbf{w}_j) = \| \phi(\mathbf{x}) - \phi(\mathbf{w}_j) \|^2 \tag{4}$$

The norm in Formula (4) can be written as follows:

$$\begin{aligned} &\| \phi(\mathbf{x}) - \phi(\mathbf{w}_j) \|^2 \\ &= \phi(\mathbf{x})^T \phi(\mathbf{x}) - 2\phi(\mathbf{x})\phi(\mathbf{w}_j) + \phi(\mathbf{w}_j)^T \phi(\mathbf{w}_j) \end{aligned} \tag{5}$$

Each item can be regarded as the inner product in space V. According to Mercer's condition [12],

$$K(\boldsymbol{a}, \boldsymbol{b}) = \phi(\boldsymbol{a})^T \phi(\boldsymbol{b}) \tag{6}$$

Then, $J(\mathbf{w}_j)$ can be written as:

$$J(\mathbf{w}_j) = K(\mathbf{x}, \mathbf{x}) - 2K(\mathbf{x}, \mathbf{w}_j) + K(\mathbf{w}_j, \mathbf{w}_j) \tag{7}$$

To minimize the value of the function $J(\mathbf{w}_j)$, gradient descent method [13] can be used to obtain the new updating formula of weights:

$$\begin{aligned} \mathbf{w}_j(t+1) &= \mathbf{w}_j(t) - \eta'(t) \nabla J(\mathbf{w}_j) \\ &= \mathbf{w}_j(t) - \eta'(t)\left[\frac{\partial K(\mathbf{w}_j, \mathbf{w}_j)}{\partial \mathbf{w}_j} - 2\frac{\partial K(\mathbf{x}, \mathbf{w}_j)}{\partial \mathbf{w}_j}\right] \end{aligned} \tag{8}$$

Thus, we derive the kernel-based SOM (kernel SOM) algorithm. Different kernel functions can induce different distance measures because of the flexibility of kernel mapping.

Polynomial kernel:

$$K(\boldsymbol{a}, \boldsymbol{b}) = (\boldsymbol{a}^T \boldsymbol{b})^d \tag{9}$$

RBF kernel:

$$K(\boldsymbol{a}, \boldsymbol{b}) = e^{-\|\boldsymbol{a}-\boldsymbol{b}\|^2 / 2\sigma^2} \tag{10}$$

Cauchy kernel:

$$K(\boldsymbol{a}, \boldsymbol{b}) = \frac{1}{1 + \| \boldsymbol{a} - \boldsymbol{b} \|^2 / \sigma^2} \tag{11}$$

Under the new measure, winner neuron $j*$ is selected according to $d_{j^*} = \min J(\mathbf{w}_j)$.

When we get the trained kernel SOM, kernel quantization error $e = K(\mathbf{x}, \mathbf{x}) + K(\mathbf{w}_{j^*}, \mathbf{w}_{j^*}) - 2K(\mathbf{x}, \mathbf{w}_{j^*})$, will replace original QE for anomaly detection.

B. K-BMUs SOM

In classic SOM, we always select BMU to calculate quantization error. That is, QE is only related to the best matching neuron in the feature map. This QE is called MQE (minimum quantization error). MQE is affected by noise in the training data. Or if we calculate the distance between input $\mathbf{x}$ and all competitive neurons, the average of distance has poor performance when training data is non-convex or has isolated clusters. Inspired by KNN algorithm, if we select a set of BMUs, and calculate the average QE in this set, these shortcomings can be solved [10]. We call this method K-BMUs SOM (K Best Matching Units Self-Organizing Map) in this paper.

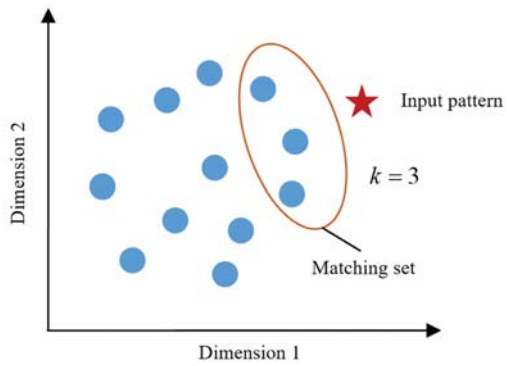


Figure 3. k BMUs are selected

Then the main task is to select a set of BMUs. We can compute the distance (such as Euclidean distance) between input **x** and each neuron in the competitive layer. And find the k neurons with k minimum distance. These neurons make up a set of BMUs, then the average of k minimum distance is calculated to represent the similarity between **x** and the neurons in the trained competitive layer. This will be used for anomaly detection, as described in section II .

IV. EXPERIMENTS

This section describes the datasets used in this paper and the comparison of different methods.

A. Datasets

We applied these different SOM-based anomaly detection algorithms to three datasets in experiments. The three datasets are pen-global, satellite and shuttle dataset [14]. The web link of datasets is https://doi.org/10.7910/DVN/OPQMVF.

Table I covers the basic information about the datasets.

TABLE I. CHARACTERISTICS OF DATASETS FROM DIFFERENT DOMAINS

Datasets	Characteristics			
	Size	*Dimensions*	*Anomalies*	*Percentage*
Pen-global	809	16	90	11.1
Satellite	5100	36	75	1.49
Shuttle	46464	9	878	1.89

Pen-Based Recognition of Handwritten Text (global): 45 different writers wrote digits 0–9 and composed this dataset. In this study, digit 8 is kept as the normal class. In each of the other classes, 10 samples are sampled as anomalies. There are 809 instances in pen-global dataset, in which 90 instances are anomalies. Each instance has 16 dimensions.

Landsat Satellite: This is an image dataset from satellite observations. In this study, four different kinds of soil are defined as normal class, while part of "soil with vegetation stubble" and "cotton crop" are sampled and regard as anomalies. There are 5,025 normal instances and 75 anomalies in the dataset. Each instance has 36 dimensions.

Statlog Shuttle: This dataset describes the work condition of radiator in a space shuttle from NASA using 9 attributes. "Radiator flow" (class 1) is the normal class and others are abnormal. For the classes 2, 3, 5, 6 and 7, researchers apply a stratified sampling. At last, the number of instances in the dataset is 46,464, and 1.89% of them are anomalies.

B. Visualization of SOM

After training SOM, for each competitive neuron, we calculate the distance between it and its neighbors. We sum the distance for each neuron, and get a distance map after normalization. We know if one neuron is near to its neighbors, its distance in the distance map will be small. Correspondingly, if a neuron is farther away from its neighbors, its value in the distance map is larger. SOM maintains the topological relationship of real data. Similar samples are still close in the competitive layer, so samples with the same class will gather in the competition layer.

Next, we visualize the distance map. The value of the distance map is represented by the shade of the grayscale image. As shown in Fig. 4 (a), the darker the color, the smaller the distance between the neuron and its neighbors. When test data are input to the trained SOM, we also display the winning neurons in the grayscale image. Winning neurons of normal samples are represented by green squares, while winning neurons of anomalies are represented by red circles. It can be seen from Fig. 4 (b) that winning neurons of normal samples are mostly in the dark-colored regions where neurons gather together. Winning neurons of anomalies are mostly in light-colored regions where neurons are deviated from its neighbors. It shows that topological relationships are maintained in SOM.

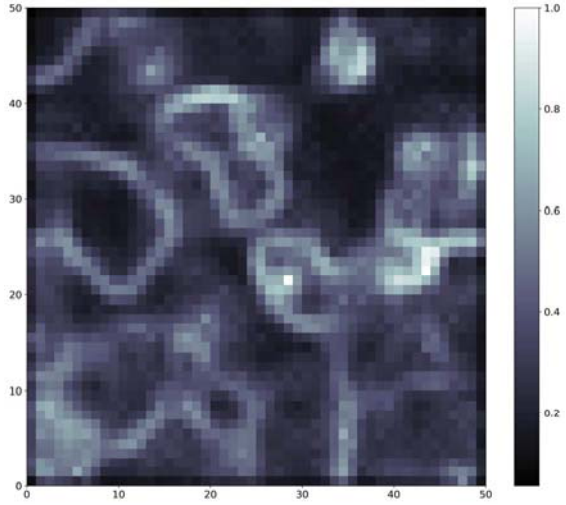


Figure 4. (a) Distance map of satellite (classic SOM)

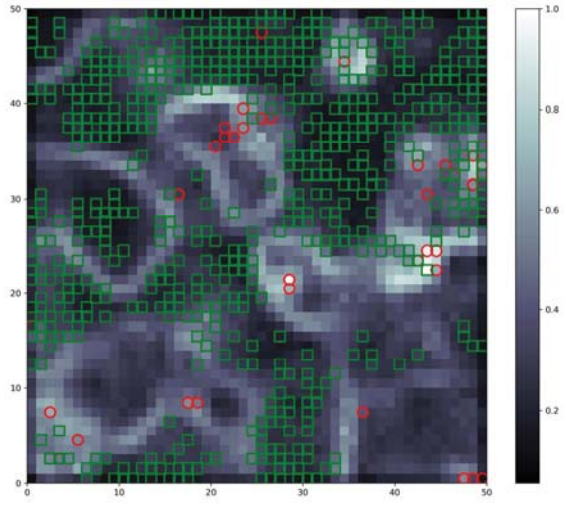
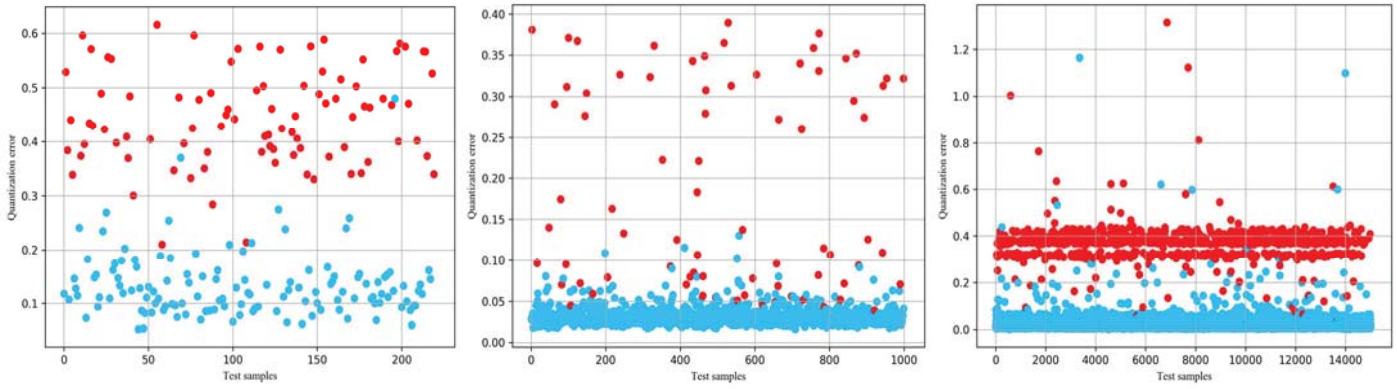


Figure 4. (b) Winners in distance map of satellite (classic SOM)

C. *Anomaly Detection Results*

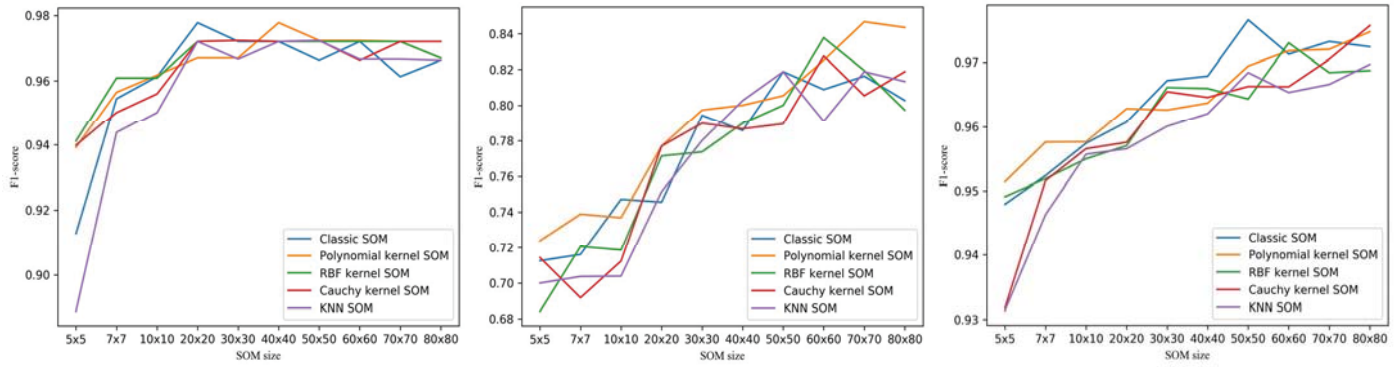
As mentioned in Section II, QE of anomalies is larger than that of normal samples for trained SOM. This is shown in Fig. 5. Red points are abnormal samples and blue points are normal samples. There is obvious difference in QE between normal samples and anomalies.

As Fig. 6 shows, the size of the SOM's competitive layer also has large influence on the performance of anomaly detection. Generally, larger size is more conducive to the improvement of anomaly detection accuracy. When the size of SOM reaches a certain degree, the improving speed of F1-score decreases. In some cases, F1-score itself even declines.



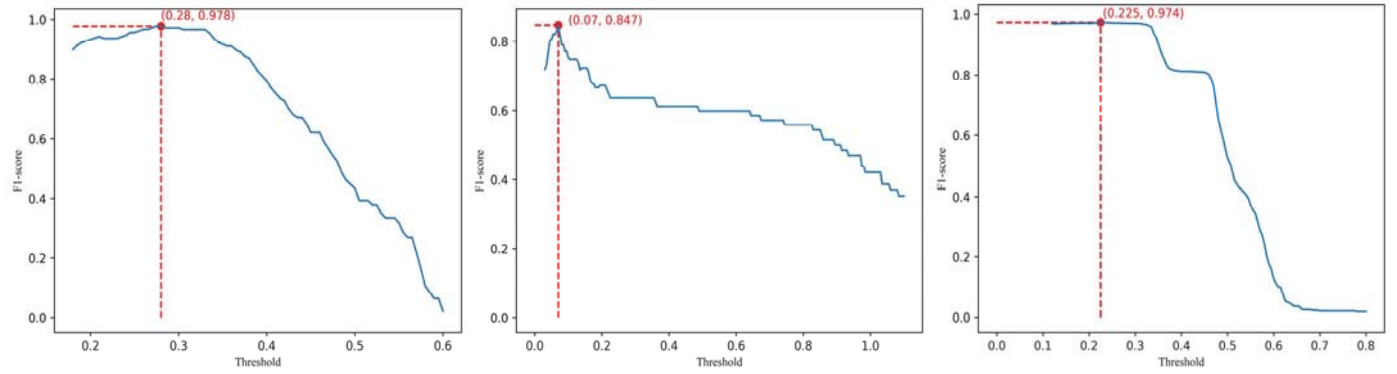
(a) pen (b) satellite (c) shuttle

Figure 5. QE of test samples with classic SOM



(a) pen-global (b) satellite (c) shuttle

Figure 6. F1-score changes with SOM's size



(a) pen-global (classic SOM) (b) satellite (polynomial kernel SOM) (c) shuttle (RBF kernel SOM)

Figure 7. F1-score changes with threshold

We also calculate QE of the training set for trained SOM. Mean μ and standard deviation σ of QE is also calculated, which is helpful to find appropriate thresholds. If QE of an input is above the threshold, the input is regarded as anomaly. The initial threshold can be obtained through 3σ principle (when QE of a sample is larger than $\mu+3\sigma$, it is regarded as anomaly). Then the appropriate threshold can be found by gradually increasing threshold with a certain step. Fig. 7 shows the relationship of F1-score and the threshold of QE.

We compare SOM-based methods with one-class SVM and LOF, which are commonly used in anomaly detection. It can be seen that SOM has great advantages in performance. F1-score of SOM-based methods are usually higher than these two methods. In our experiments, the performance of classic SOM, kernel SOM and K-BMUs SOM has little difference on pen-global and shuttle datasets. However, there is significant difference in satellite dataset. The performance of the three kernel SOM algorithms is all better than classic SOM. K-BMUs SOM does not show its advantage as some researchers present in their papers. The results are shown in Table II. Therefore, which kind of SOM-based method performs better is associated with the characteristics of the datasets.

TABLE II. TABLE TYPE STYLES RESULTS

Anomaly detection Algorithms	**Datasets**		
	Pen-global	***Satellite***	***Shuttle***
Classic SOM	0.978	0.819	0.977
Polynomial kernel SOM	0.978	0.847	0.975
RBF kernel SOM	0.972	0.838	0.974
Cauchy kernel SOM	0.972	0.828	0.976
K-BMUs SOM	0.972	0.819	0.970
One-class SOM	0.874	0.742	0.974
LOF	0.678	0.556	0.963

V. CONCLUSIONS

In this study, we applied three SOM-based anomaly detection methods, namely classic SOM, kernel SOM and K-BMUs SOM to multivariate data. And kernel SOM has three different forms according to kernel functions. SOM-based methods perform well when only normal samples are used to train the model. As a semi-supervised method, it helps to solve the problem that anomalies are difficult to get in practice.

We compare these SOM-based methods by applying them to three datasets in different domains, and find that the performance of these methods is associated with the characteristics of datasets. Compared with the commonly used one-class SVM and LOF algorithms, SOM-based anomaly detection methods have obvious advantages in performance.

However, although the results of the three SOM-based methods differ, the difference is not so much. Further work will focus on designing more effective SOM-based anomaly detection methods.

REFERENCES

[1] F. E. Grubbs, "Procedures for detecting outlying observations in samples," Technometrics, vol. 11, no. 1, pp. 1-21, 1969.

[2] C. S. Gray and S. J. Watson, "Physics of failure approach to wind turbine condition based maintenance," Wind Energy, vol. 13, no. 5, pp. 395-405, 2010.

[3] W.-H. Chen, S.-H. Hsu, and H.-P. Shen, "Application of SVM and ANN for intrusion detection," Computers & Operations Research, vol. 32, no. 10, pp. 2617-2634, 2005.

[4] Y. Liao and V. R. Vemuri, "Use of k-nearest neighbor classifier for intrusion detection," Computers & security, vol. 21, no. 5, pp. 439-448, 2002.

[5] G. Münz, S. Li, and G. Carle, "Traffic anomaly detection using k-means clustering," in GI/ITG Workshop MMBnet, 2007, pp. 13-14.

[6] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," in ACM sigmod record, 2000, vol. 29, no. 2, pp. 93-104: ACM.

[7] Y. Wang, J. Wong, and A. Miner, "Anomaly intrusion detection using one class SVM," in Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004., 2004, pp. 358-364: IEEE.

[8] T. Kohonen, "The self-organizing map," Proceedings of the IEEE, vol. 78, no. 9, pp. 1464-1480, 1990.

[9] PAN Zhi-song, CHEN Song-can, and Zhang Dao-qiang, "A Kernel-Based SOM Classification in Input Space," Acta Electronica Sinica, no. 02, pp. 227-231, 2004.

[10] J. Tian, M. H. Azarian, and M. Pecht, "Anomaly detection using self-organizing maps-based k-nearest neighbor algorithm," in Proceedings of the European Conference of the Prognostics and Health Management Society, 2014: Citeseer.

[11] A. Muñoz and J. Muruzábal, "Self-organizing maps for outlier detection," Neurocomputing, vol. 18, no. 1–3, pp. 33-60, 1998.

[12] C. Campbell, "Kernel methods: a survey of current techniques," Neurocomputing, vol. 48, no. 1-4, pp. 63-84, 2002.

[13] S. Haykin, Neural networks: a comprehensive foundation. Prentice Hall PTR, 1994.

[14] M. Goldstein, "Unsupervised Anomaly Detection Benchmark," DRAFT VERSION ed: Harvard Dataverse, 2015.