Data Driven Hyperparameters Optimization of One-Class Support Vector Machines for Anomaly Detection in Wireless Sensor Networks

Van Vuong Trinh and Kim Phuc Tran
Division of Artificial Intelligence,
Faculty of Information Technology,
Dong A University, Danang, Vietnam
Email: vanvuong.trinh@gmail.com, phuctk@donga.edu.vn

Truong Thu Huong
Department of Telecommunication Systems,
School of Electronics and Telecommunications,
Hanoi University of Science and Technology, Hanoi, Vietnam
Email: huong.truongthu@hust.edu.vn

***Abstract*—One-class support vector machines (OCSVM) have been recently applied to detect anomalies in wireless sensor networks (WSNs). Typically, OCSVM is kernelized by radial bais functions (RBF, or Gausian kernel) whereas selecting hyperparameters is based upon availability of anomalies, which is rarely applicable in practice. This article investigates the application of OCSVM to detect anomalies in WSNs with data-driven hyperparameters optimization. Specifically, the information of the farthest and the nearest neighbors of each sample is used to construct the objective cost instead of labeling based metrics such as geometric mean accuracy (G-mean) or area under the receiver operating characteristic (AUROC). The efficiency of this method is illustrated over the IBRL dataset whereas the resulting estimated boundary as well as anomaly detection performance are comparable with existing methods.**

***Index Terms*—one-class support vector machines, anomaly detection, wireless sensor networks, Gaussian kernel, parameters selection.**

I. Introduction

WSNs are composed of a large number of sensor nodes that can communicate the information gathered from a monitored field through wireless links to monitor physical or environmental conditions, such as temperature, vibration, pressure, motion, etc. and to cooperatively pass their data throughout the network. WSNs are an emerging and very interesting technology applied to different applications including industrial processes, monitoring and control, machine health monitoring, healthcare applications and traffic control. WSNs will be an integral part of our lives, more so than the present-day personal computers (see [16]). Due to the deployment of a large number of sensor nodes in uncontrolled or hostile environments, data measured and collected by WSNs is often unreliable. This will affect the modeling and scientific reasonable inference. Thus, it is significant that the anomaly of sensor node is detected in order to obtain accurate information, therefore make effective decisions by information gatherered (see, for instance, [10]).

Anomaly detection is a method to identify whether or not a metric is behaving differently than it has in the past, taking into account trends. This is implemented as one-class classification since only one class is represented in the training data. Several methods have been proposed to solve the one-class classification problem which can be classified into three main types the density estimation, the boundary methods and the reconstruction methods. In literature, a variety of anomaly detection techniques have been developed for certain application domains such as security systems, fraud detection and statistical process monitoring, for example, see [6], [4], [14], [12], [2], [13] and [11]. Recently, there have been growing interests in applying machine learning approaches for anomaly detection in WSNs. For further details see, for instance, [10], [16], [8], [2], and[5]. Recently, [7] presented a detailed analysis of various formulations of OCSVM, like, hyper-plane, hyper-sphere, quarter-sphere and hyper-ellipsoidal to separate the normal data from anomalous data for wireless sensor networks in harsh environments.

In this contribution, we study a data driven algorithm based on OCSVM for anomaly detection in WSNs. We test the performance of the algorithm on real data set obtained from a WSN deployment at the Intel Berkeley Research Laboratory. The remainder of the paper is organized as follows: in section II, some necessary background on OCSVM are introduced; in section III, the anomaly detection approach in WSNs is defined; section IV presents an illustrative example, and finally, some concluding remarks and recommendations are made in section V.

II. One-Class Support Vector Machines and Gaussian Kernel

In this section, we briefly review one-class support vector machines (OCSVM) [9] and the Gaussian kernel.

OCSVM is used to estimate the support of a distribution. Notationally, let us consider a data set $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_N\}$, with each $\mathbf{x}_i \in \mathcal{R}^D$ belonging to a given class of interest (named target class). The basic idea behind the OCSVM is to separate data from the origin by finding a hyperplane with maximum margin separation from

the origin. In order to deal with nonlinearly problems, the hyperplane is defined in a high-dimensional Hilbert feature space $\mathcal{F}$ where the samples are mapped through a nonlinear transformation $\Phi(.)$. We will work only a kernel function $k(\mathbf{x},\mathbf{y})$ instead of the scalar product $(\mathbf{\Phi}(\mathbf{x}).\mathbf{\Phi}(\mathbf{y}))$. To separate the data set from the origin, [9] solved the following quadratic program:

$$\underset{\mathbf{w},\mathbf{a},\boldsymbol{\xi},\rho}{\text{Minimize}} \ \frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2 + \frac{1}{\nu N}\sum_{i=1}^{N}\xi_i - \rho \tag{1a}$$

$$\text{Subject to } (\mathbf{w}.\Phi(\mathbf{x}_i)) \geq \rho - \xi_i, \xi_i \geq 0 \quad \forall i = 1 \ldots N \tag{1b}$$

Here, $\mathbf{w}$ is a vector perpendicular to the hyperplane in $\mathcal{F}$, and ρ is the distance to the origin. Since the training data distribution may contain outliers, a set of slack variables $\xi_i \geq 0$ is introduced to deal with them. The parameter $\nu \in (0,1]$ controls the tradeoff between the number of examples of the training set mapped as positive by the decision function

$$f(\mathbf{z}) = \text{sgn}((\mathbf{w}.\Phi(\mathbf{x})) - \rho) \tag{2}$$

and having a small value of $||\mathbf{w}||$ to control model complexity.

Using multipliers $\alpha_i, \beta_i \geq 0$, [9] introduced a Lagrangian

$$\begin{aligned} L = &\frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2 + \frac{1}{\nu N}\sum_{i=1}^{N}\xi_i - \rho \\ &- \sum_{i=1}^{N}\alpha_i((\mathbf{w}.\Phi(\mathbf{x}_i)) - \rho + \xi_i) - \sum_{i=1}^{N}\beta_i\xi_i \end{aligned} \tag{3}$$

and set the derivatives with respect to the primal variables $\mathbf{w}$, $\boldsymbol{\xi}$, ρ equal to zero, i.e.

$$\frac{\partial L}{\partial \mathbf{w}} = 0: \quad \mathbf{w} = \sum_{i=1}^{N}\alpha_i\Phi(\mathbf{x}_i), \tag{4}$$

$$\frac{\partial L}{\partial \xi_i} = 0: \quad \alpha_i = \frac{1}{\nu N} - \beta_i \leq \frac{1}{\nu N}, i = 1, \ldots, N. \tag{5}$$

$$\frac{\partial L}{\partial \rho} = 0: \quad \sum_{i=1}^{N}\alpha_i = 1 \tag{6}$$

In (4), all patterns $\{\mathbf{x}_i : i \in [1,\ldots,N], \alpha_i > 0\}$ are called Support Vectors. From (4), using the kernel function, the decision function (2) is transformed into a kernel expansion

$$f(\mathbf{x}) = \text{sgn}\left(\sum_{i=1}^{N}\alpha_i k(\mathbf{x}_i,\mathbf{x}) - \rho\right) \tag{7}$$

Substituting (4), (5) and (6) into (3), and using the kernel function, we have

$$L = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k(\mathbf{x}_i,\mathbf{x}_j). \tag{8}$$

Then, we obtain the dual problem

$$\underset{\alpha}{\text{Minimize}} \ \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k(\mathbf{x}_i,\mathbf{x}_j) \tag{9a}$$

$$\text{Subject to } \sum_{i=1}^{N}\alpha_i = 1, 0 \leq \alpha_i \leq \frac{1}{\nu N}, \quad \forall i = 1 \ldots N \tag{9b}$$

[9] shown that at the optimum, the two inequality constraints (1b) became equalities if α_i and β_i are nonzero, which implies $0 < \alpha_i < \frac{1}{\nu N}$. Thus, the value of ρ can be recovered by exploiting that for any such α_i, the corresponding pattern $\mathbf{x}_i$ satisfies

$$\rho = \langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle = \sum_{j=1}^{N}\alpha_j k(\mathbf{x}_j,\mathbf{x}_i) \tag{10}$$

The most commonly used kernel is the radial basis functions (RBF, or Gaussian) kernel:

$$k\left(\mathbf{x}_i,\mathbf{x}_j\right) = \exp\left(-\frac{\left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right|^2}{2\sigma^2}\right) \tag{11}$$

where $\sigma > 0$ stands for the kernel width parameter. In the feature space, the distance between two mapped samples $\mathbf{x}_i$ and $\mathbf{x}_j$ is:

$$\begin{aligned} \left|\left|\phi\left(\mathbf{x}_i\right) - \phi\left(\mathbf{x}_j\right)\right|\right|^2 &= k\left(\mathbf{x}_i,\mathbf{x}_i\right) + k\left(\mathbf{x}_j,\mathbf{x}_j\right) - 2k\left(\mathbf{x}_i,\mathbf{x}_j\right) \\ &= 2\left[1 - \exp\left(-\frac{\left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right|^2}{2\sigma^2}\right)\right] \end{aligned} \tag{12}$$

This exhibits a positively proportional relation between $||\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)||$ and $||\mathbf{x}_i - \mathbf{x}_j||$. In other words, Gaussian kernel preserves the ranking order of the distances between samples in the input and feature spaces.

III. Description of Anomaly Detection Procedure for WSNs

Imitiating [15], in this paper, we maximize the following performance measure for training the OCSVM:

$$\begin{aligned} J(\sigma) = &\frac{1}{n}\sum_{i=1}^{N}\max_j \left|\left|\phi\left(\mathbf{x}_i\right) - \phi\left(\mathbf{x}_j\right)\right|\right|^2 \\ &- \frac{1}{n}\sum_{i=1}^{N}\min_{j\neq i}\left|\left|\phi\left(\mathbf{x}_i\right) - \phi\left(\mathbf{x}_j\right)\right|\right|^2 \end{aligned} \tag{13}$$

For Guassian kernel, this can be further simplified:

$$J(\sigma) = \frac{2}{n}\sum_{i=1}^{N}\min_{j\neq i}k\left(\mathbf{x}_i,\mathbf{x}_j\right) - \frac{2}{n}\sum_{i=1}^{N}\max_j k\left(\mathbf{x}_i,\mathbf{x}_j\right) \tag{14}$$

$$= \frac{2}{n}\sum_{i=1}^{N}\exp\left(-\frac{\min_{j\neq i}\left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right|^2}{2\sigma^2}\right) - \frac{2}{n}\sum_{i=1}^{N}\exp\left(-\frac{\max_j \left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right|^2}{2\sigma^2}\right) \tag{15}$$

Denote the nearest and farthest neighbors distances respectively as

$$\text{Near}\,(\mathbf{x}_i) = \min_{j \neq i} ||\mathbf{x}_i - \mathbf{x}_j||^2 \tag{16a}$$

$$\text{Far}\,(\mathbf{x}_i) = \max_{j} ||\mathbf{x}_i - \mathbf{x}_j||^2 \tag{16b}$$

Evidently, time complexity of exactly evaluating such distances is $\mathcal{O}(N^2)$. In applications where the training set is huge, an approximate or sequential approaches may be required.

We hereafter arrive at:

$$J(\sigma) = \frac{2}{n}\sum_{i=1}^{N} \exp\left(-\frac{\text{Near}\,(\mathbf{x}_i)}{2\sigma^2}\right) - \frac{2}{n}\sum_{i=1}^{N} \exp\left(-\frac{\text{Far}\,(\mathbf{x}_i)}{2\sigma^2}\right) \tag{17}$$

and devise the gradient of $J(\sigma)$ with respect to σ as:

$$\nabla J(\sigma) = \frac{2}{n}\sum_{i=1}^{N} \exp\left(-\frac{\text{Near}\,(\mathbf{x}_i)}{2\sigma^2}\right)\frac{\text{Near}\,(\mathbf{x}_i)}{\sigma^3} - \frac{2}{n}\sum_{i=1}^{N} \exp\left(-\frac{\text{Far}\,(\mathbf{x}_i)}{2\sigma^2}\right)\frac{\text{Far}\,(\mathbf{x}_i)}{\sigma^3} \tag{18}$$

This enables us to deploy the conventional gradient-based optimization method to find $\sigma^\star$ that maximizes $J(\sigma)$. However, it is notable that there is no guarantee on convexity of $J(\sigma)$. Nevertheless, this is unavoidable as optimizing parameters of kernel methods is generally known as non-convex.

After the Gaussian kernel parameter σ is selected, the OCSVM can be trivially computed. However, for better robust detection, the discriminative threshold ρ may need to be adjusted by a small quantity, namely δ. Finally, the whole procedure is summarized as below.

Algorithm 1 (Anomaly detection)

▷ Training phase:

1 For each sample $\mathbf{x}_i$ of a given training set $\{\mathbf{x}_i\}_{i=1}^N$, evaluating the quantities $\text{Near}\,(\mathbf{x}_i)$ and $\text{Far}\,(\mathbf{x}_i)$ in accordance to (16a) and (16b).

2 Set Gaussian kernel parameter as $\sigma^\star = \text{argmax} J(\sigma)$ whereas $J(\sigma)$ and $\nabla J(\sigma)$ are defined in (17) and (18).

3 Set the parameter $\nu \ll 1$ and solve (9) to obtain the decision function with adjusted discriminative threshold δ as:

$$f(\mathbf{x}) = \text{sgn}\left(\sum_{i=1}^{N} \alpha_i k(\mathbf{x}_i, \mathbf{x}) - \rho + \delta\right) \tag{19}$$

▷ Decision phase:

4 For a new sample $\mathbf{z}$, classify it according to (19), then raise an alarm if $f(\mathbf{z}) = -1$.

IV. Illustrative Example

This section investigates the efficiency of anomaly detection algorithm over a real data set. The source code will be freely available at https://github.com/trinhvv/wsn-ocsvm-dfn. All computation was performed on a platform with 2.6 GHz Intel(R) Core(TM) i7 and 16GB of RAM.

A. Data description

We consider a data set gathered from a WSN deployment at the Intel Berkeley Research Laboratory (IBRL) [1] with 54 *Mica2Dot* sensor nodes. Fig. 1 shows the sensor deployment in the laboratory. The sensors collect five measurements: light in Lux, temperature in degrees celsius, humidity (temperature corrected relative humidity) ranging from 0% to 100%, voltage in volts and network topology information in each 30 second sampling period. Node 0 is the gateway node while other nodes broadcast their data in multiple hops to the gateway node. During the 30 day (720 hour) period between 28th Feb 2004 and 5th April 2004, the 54 nodes collected about 2.3 million readings.

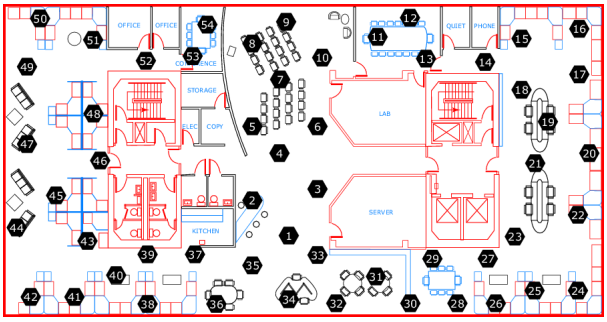

Fig. 1: A map of sensors' location. (Source: [1])

In this paper we consider the IBRL data set obtained from 5 close nodes, $1, 2, 33, 35, 37$. Also, only two features, namely temperature and humidity, are taken into account. The data during the first 10 days period on March 2004 will be used as the training set. This training set contains more than 82000 samples and hereafter is reduced into only 55421 samples.

B. Gaussian kernel parameter optimization

The objective function $J(\sigma)$ according to the given training set is depicted in Fig. 2 and is evidently strongly convex.

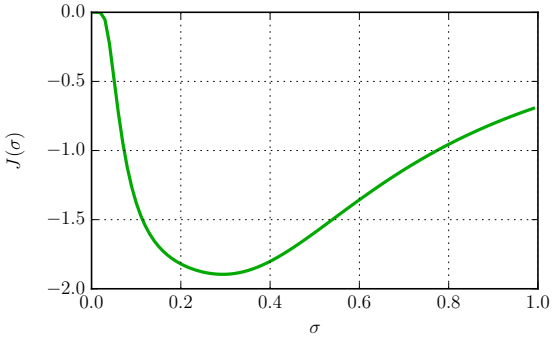

Fig. 2: Objective function $J(\sigma)$.

The Matlab's routine *fminunc(·)* is used for parameter optimization, thus providing the kernel parameter $\sigma^\star = 0.2938$ after few iterations.

C. Training OCSVM and some results

Using the LIBSVM library [3], the OCSVM is computed with $\nu = 0.0001$ and $\sigma = 0.2938$, consisting only 9 support vectors. Fig. 3 depicts the training set (green dot), the support vectors (white dot) and decision boundaries with different discriminative threshold.

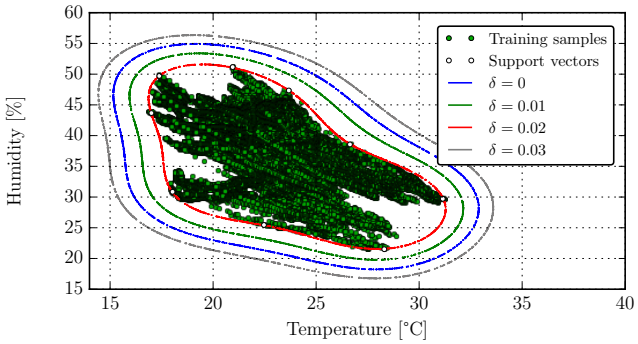

Fig. 3: Discrimination boundaries with some δ's values.

Fig. 4 depicts detection result on considered nodes over time with $\delta = 0.02$. Although node 37 exhibits several false alarms, such a fault detection is unavoidable and acceptable in practice.

V. Conclusion and Future Work

References

[1] Phil Buonadonna, David Gay, Joseph M. Hellerstein, Wei Hong, and Samuel Madden. TASK: Sensor network in a box. *Proceedings of the Second European Workshop on Wireless Sensor Networks, EWSN 2005*, 2005:133–144, 2005.

[2] V. Chandola, A. Banerjee, and V. Kumar. *Anomaly Detection*, pages 1–15. Springer US, Boston, MA, 2016.

[3] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[4] D. A. Clifton, S. Hugueny, and L. Tarassenko. Novelty detection with multivariate extreme value statistics. *Journal of signal processing systems*, 65(3):371–389, 2011.

[5] Zhen Feng, Jingqi Fu, Dajun Du, Fuqiang Li, and Sizhou Sun. A new approach of anomaly detection in wireless sensor networks using support vector data description. *International Journal of Distributed Sensor Networks*, 13(1):1550147716686161, 2017.

[6] J. Ilonen, P. Paalanen, J.K. Kamarainen, and H. Kalviainen. Gaussian mixture pdf in one-class classification: computing and utilizing confidence values. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 2, pages 577–580. IEEE, 2006.

[7] I. N. Naqvi N. Shahid and S. B. Qaisar. One-class support vector machines: analysis of outlier detection for wireless sensor networks in harsh environments. *Artificial Intelligence Review*, 43(4):515–563, 2015.

[8] Sutharshan Rajasegarar, Christopher Leckie, and Marimuthu Palaniswami. Hyperspherical cluster based distributed anomaly detection in wireless sensor networks. *Journal of Parallel and Distributed Computing*, 74(1):1833–1847, 2014.

[9] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[10] A.B. Sharma, L. Golubchik, and R. Govindan. Sensor faults: Detection methods and prevalence in real-world datasets. *ACM Transactions on Sensor Networks (TOSN)*, 6(3):23, 2010.

[11] K.P. Tran. The efficiency of the 4-out-of-5 Runs Rules scheme for monitoring the Ratio of Population Means of a Bivariate Normal distribution. *International Journal of Reliability, Quality and Safety Engineering*, 2016. In press, DOI: 10.1142/S0218539316500200.

[12] K.P. Tran, P. Castagliola, and G. Celano. Monitoring the Ratio of Two Normal Variables Using EWMA Type Control Charts. *Quality and Reliability Engineering International*, 2015. In press, DOI: 10.1002/qre.1918.

[13] K.P. Tran, P. Castagliola, and G. Celano. Monitoring the Ratio of Population Means of a Bivariate Normal distribution using CUSUM Type Control Charts. *Statistical Papers*, 2016. In press, DOI: 10.1007/s00362-016-0769-4.

[14] K.P. Tran, P. Castagliola, and G. Celano. Monitoring the Ratio of Two Normal Variables Using Run Rules Type Control Charts. *International Journal of Production Research*, 54(6):1670–1688, 2016.

[15] Yingchao Xiao, Huangang Wang, Lin Zhang, and Wenli Xu. Two methods of selecting Gaussian kernel parameters for one-class SVM and their application to fault detection. *Knowledge-Based Systems*, 59:75–84, 2014.

[16] Miao Xie, Song Han, Biming Tian, and Sazia Parvin. Anomaly detection in wireless sensor networks: A survey. *Journal of Network and Computer Applications*, 34(4):1302–1325, 2011.

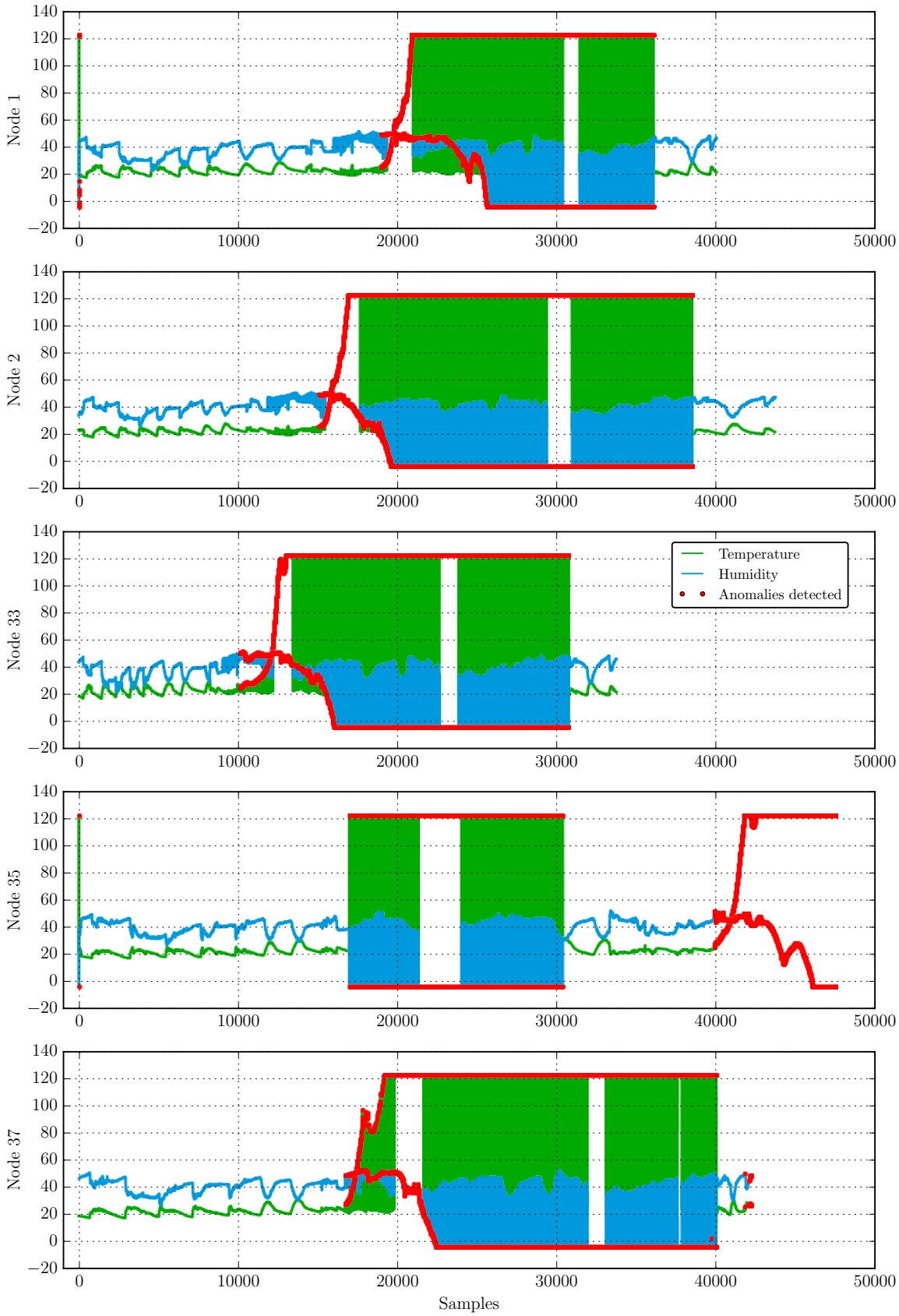

Fig. 4: Anomaly detection validation upon whole IBRL data set on 5 nodes. Almost apparent anomalies, i.e. temperature and humidity measurements that are too high or too low, are detected. Some false alarms or non-detected anomalous measures are observed, which is acceptable in practice.