Anomaly Detection in Wireless Sensor Networks Using Support Vector Data Description with Mahalanobis Kernels and Control Limit Adjustment

Van Vuong Trinh[a,*], Kim Phuc Tran[a], Anh Tuan Mai[b]

[a]*Division of Artificial Intelligence, Faculty of Information Technology, Dong A University, Danang, Vietnam*
[b]*International Training Institute for Materials Science (ITIMS), Hanoi University of Science and Technology, Hanoi, Vietnam*

Abstract

Keywords: Wireless sensor networks, Anomaly detection, Outlier detection, Support vector data description.

1. Introduction

Wireless sensor networks (WSNs) [1]
[2]
[3]

Notation and Definition

$x \cdot y$ is inner product.

2. Description of Anomaly Detection Approach

In this section, we recall the so-called support vector data description (SVDD) originally derived in [4], which is an alternative of one-class support vector machines (OCSVM) [5] and is analogous to the known support vector machines (SVM) [6]. Practical perspectives regarding modified discriminative function with control limit adjustment and hyperparameter selection based on Bayesian optimization are also discussed. Finally, an anomaly detection algorithm is summarized for clarity.

2.1. Theory of SVDD

Given N samples $x_k \in \mathcal{X}$, $k = 1, \ldots, N$, SVDD method aims to estimate a sphere with minimum volume that contains all (or most of) these data. Let a and R being reserved for the center and the radius of the sphere, we define the objective function to minimize the volume of the sphere and the number of outliers as:

$$R^2 + C \sum_{k=1}^{N} \xi_k \tag{1}$$

where $C > 0$ is a regularization parameter with constraints that almost data points are within the sphere:

$$||x_k - a||^2 \leq R^2 + \xi_k,\ \xi_k \geq 0, \text{ for } k = 1, \ldots, N \tag{2}$$

*Corresponding author
Email address: vanvuong.trinh@gmail.com (Van Vuong Trinh)

To adapt with nonspherical distribution, a conventional approach is to map given data into a higher dimensional feature space, namely $\mathcal{F}$, then learning a sphere in such a new space. This results into the so-called *primal optimisation* as follows:

$$\underset{R,\,a,\,\xi}{\text{Minimize}}\ R^2 + C\sum_{k=1}^{N}\xi_k \tag{3a}$$

$$\text{Subject to}\ \left|\left|\phi\left(x_k\right) - a\right|\right|^2 \leq R^2 + \xi_k,\ \xi_k \geq 0,\ \text{for}\ k = 1, \ldots, N \tag{3b}$$

where $\phi\left(\cdot\right) : \mathcal{X} \rightarrow \mathcal{F}$ is the aforementioned feature mapping. The Lagrangian is hereafter written as:

$$\mathcal{L} = R^2 + C\sum_{k=1}^{N}\xi_k - \sum_{k=1}^{N}\alpha_k\left[R^2 + \xi_k - \left(\phi\left(x_k\right)\cdot\phi\left(x_k\right) - 2a\cdot\phi\left(x_k\right) + \phi\left(a\right)\cdot\phi\left(a\right)\right)\right] - \sum_{k=1}^{N}\gamma_k\xi_k \tag{4}$$

with the Lagrange multipliers $\alpha_k, \gamma_k \geq 0$. $\mathcal{L}$ should be minimized w.r.t. R, a and maximized w.r.t. α_k, γ_k.

Setting partial derivatives gives:

$$\frac{\partial\mathcal{L}}{\partial R} = 0 : \quad \sum_{k=1}^{N}\alpha_k = 1 \tag{5}$$

$$\frac{\partial\mathcal{L}}{\partial a} = 0 : \quad a = \sum_{k=1}^{N}\alpha_k\phi\left(x_k\right) \tag{6}$$

$$\frac{\partial\mathcal{L}}{\partial \xi_k} = 0 : \quad \alpha_k + \gamma_k = C \tag{7}$$

Obviously, Lagrange multipliers γ_k can be eliminated by imposing bound constraints on α_k as:

$$0 \leq \alpha_k \leq C \tag{8}$$

Then, substituting (5)-(7) into (4) leads to the following *dual optimisation*:

$$\underset{\alpha}{\text{Maximize}}\ \sum_{k=1}^{N}\alpha_k\left(\phi\left(x_k\right)\cdot\phi\left(x_k\right)\right) - \sum_{k=1}^{N}\sum_{l=1}^{N}\alpha_k\alpha_l\left(\phi\left(x_k\right)\cdot\phi\left(x_l\right)\right) \tag{9a}$$

$$\text{Subject to}\ \sum_{k=1}^{N}\alpha_k = 1,\ 0 \leq \alpha_k \leq C,\ \text{for}\ k = 1, \ldots, N \tag{9b}$$

This standard quadratic program (QP) and can be solved more efficiently than the primal problem. Evidently, (9) is feasible iff $C \geq \dfrac{1}{N}$ [7]. Assume that such a feasibility condition holds, the following interpretation is trivial once (9) is solved.

$$\left|\left|\phi\left(x_k\right) - a\right|\right|^2 < R^2 \rightarrow \alpha_k = 0 \tag{10a}$$

$$\left|\left|\phi\left(x_k\right) - a\right|\right|^2 = R^2 \rightarrow 0 < \alpha_k < C \tag{10b}$$

$$\left|\left|\phi\left(x_k\right) - a\right|\right|^2 > R^2 \rightarrow \alpha_k = C \tag{10c}$$

Only data samples x_k with $0 < \alpha_k < C$ are required in the distribution's description and these objects will therefore be refrered to as *support vectors*. This will be discussed later after we generalize the linear SVDD into nonlinear version using kernel functions in the next sub-section.

2.2. Generalization with kernels

Instead of using inner product, an alternative kernel product can also be adopted:

$$\kappa\left(x_k, x_l\right) = \phi\left(x_k\right) \cdot \phi\left(x_l\right) \tag{11}$$

This is the known *kernel trick* [8], aims to avoid the need of explicitly declaring a feature mapping $\phi(\cdot)$.

In this paper, we investigate the performance of SVDD using the Mahalanobis kernel (MHK):

$$\kappa\left(x_k, x_l\right) = \exp\left(-\frac{\left(x_k - x_l\right)' S^{-1}\left(x_k - x_l\right)}{2\sigma}\right) \tag{12}$$

where S is the estimated covariance matrix computed with the available training data x_k, $k = 1, \ldots, N$ while parameter σ is the kernel width. Then, we compare obtained result with the popular radial basis function kernel (RBFK, or Gaussian kernel):

$$\kappa\left(x_k, x_l\right) = \exp\left(-\frac{\left(x_k - x_l\right)'\left(x_k - x_l\right)}{2\sigma}\right) \tag{13}$$

Thus, the optimisation (9) changes into:

$$\underset{\alpha}{\text{Maximize}} \sum_{k=1}^{N} \alpha_k \kappa\left(x_k, x_k\right) - \sum_{k=1}^{N}\sum_{l=1}^{N} \alpha_k \alpha_l \kappa\left(x_k, x_l\right) \tag{14a}$$

$$\text{Subject to } \sum_{k=1}^{N} \alpha_k = 1,\ 0 \leq \alpha_k \leq C, \text{ for } k = 1, \ldots, N \tag{14b}$$

By solving this problem, one obtains the support vectors x_i, $i = 1, \ldots, m$ where m is the number of support vectors. Then, sphere's radius is indeed the distance from any support vector, for instance x_i, to the sphere's center:

$$R = \sqrt{\kappa\left(x_i, x_i\right) - 2\sum_{i=1}^{m} \alpha_i \kappa\left(x_i, x_i\right) + \sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j \kappa\left(x_i, x_j\right)} \tag{15}$$

2.3. Discrimination with control limit adjustment

This sub-section aims to establish the discriminative function for detecting anomalies with preliminary discussion on robustness. Roughly speaking, the kernel distance, namely $d\left(z, a\right)$, between a test point z and the center a of sphere will be computed:

$$d\left(z, a\right) = \sqrt{\kappa\left(z, z\right) - 2\sum_{i=1}^{m} \alpha_i \kappa\left(z, x_i\right) + \sum_{k=i}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j \kappa\left(x_i, x_j\right)} \tag{16}$$

Then, it is compared with an appropriate threshold. Such a threshold is then referred to as *control limit* whose value is highly affects the probability of false negative rate (or false alarm).

A trivial choice of the control limit is sphere's radius R. This means that the data lies within the sphere will be categorized as anomalous. However, as suggested in [9, Section 7.1], it might be better to introduce a slack variable δ chosen *a priori* into the discriminative function. This is due to the fact that there is no guarantee that the data point outside the sphere is anomalous.

Thus, we hereafter use the following function for inclusion, where $\mathcal{H}$ stands for the Heaviside function $\mathcal{H}(x) = 1$, if $x \geq 0$ and 0 otherwise.

$$\begin{aligned} f\left(z\right) &= \mathcal{H}\left(d - R^2 - \delta\right) \\ &= \mathcal{H}\left(\kappa\left(z, z\right) - 2\sum_{i=1}^{m} \alpha_i \kappa\left(z, x_i\right) - D\right) \end{aligned} \tag{17}$$

The constant D can be pre-computed offline as:

$$D = \kappa(x_i, x_i) - 2\sum_{i=1}^{m} \alpha_i \kappa(x_i, x_i) + \delta \tag{18}$$

A probabilistic analysis on the affect of δ upon probability of the false alarm is available and is summarized as follows.

Theorem 1: ([9, Theorem 7.9]) Fixed the slack variable $\delta > 0$ and the expected anomalies percentage $\lambda > 0$. The function f defined in (17) returns 1 on test points z drawn according to the considered distribution with probability at most:

$$\mathcal{P}(\delta, \lambda) = \frac{1}{\delta N}\sum_{k=1}^{N} \xi_k + \frac{6R^2}{\delta\sqrt{N}} + 3\sqrt{\frac{ln(2/\lambda)}{2N}} \tag{19}$$

This theorem formalizes the intuition that small control limit implies high sensitivity to novelties. In this paper, we will study the variation of the upper bound $\mathcal{P}(\delta, \lambda)$ with different values of δ while fixing $\lambda = 0.05$ is the usual choice in practice, i.e. 5% of anomalies.

2.4. Hyperparameter optimization

Similar to other kernel methods, the hyperparameters's selection is critical and worthmentioning. Most of research use heuristic grid search while Bayesian optimization has been recently investigated. Following the latter approach, where a black-box nonlinear optimization problem is considered as follows:

$$\underset{C,\sigma}{\text{Maximize}}\ J(C, \sigma) \tag{20a}$$

$$\text{Subject to } \frac{1}{N} \leq C \leq 1,\ \underline{\sigma} \leq \sigma \leq \overline{\sigma} \tag{20b}$$

whereas $\underline{\sigma}$ and $\overline{\sigma}$ are some positive bounds of σ. For simplicity, in this paper, we deploy the geometric mean accuracy (g-mean) for performance measure, i.e.

$$J(C, \sigma) := g = \sqrt{Acc_+ \times Acc_-} \tag{21}$$

where Acc_+ and Acc_- are *true positive rate* and *true negative rate*, respectively.

2.5. Proposed algorithm for anomaly detection

Algorithm 1 (Anomaly detection algorithm) Given $\lambda > 0$ and $\delta > 0$, this algorithm optimizes SVDD's hyperparameters (C, σ). Let s being reserved for the iteration counter of an optimization solver $\mathcal{S}$. It requires training samples $\{x_k^{train}\}_{k=1}^{N_{train}}$ and testing samples $\{x_k^{test}, y_k^{test}\}_{k=1}^{N_{test}}$.

▷ Training phase:
Initialize hyperparameters as $(C^{(0)}, \sigma^{(0)})$.
For $s = 1, 2, \ldots$ until convergence
    Given $(C^{(s)}, \sigma^{(s)})$ by the solver $\mathcal{S}$, solve (14) with data set $\{x_k^{train}\}_{k=1}^{N_{train}}$.
    Validate the SVDD with data set $\{x_k^{test}, y_k^{test}\}_{k=1}^{N_{test}}$, then compute g according to (21).
    Return $J\left(C^{(s)}, \sigma^{(s)}\right) = g$, the solver $\mathcal{S}$ updates the next iterate $(C^{(s+1)}, \sigma^{(s+1)})$.
Return $(C^\star, \sigma^\star)$.
▷ Decision phase:
For an unknown sample z, classify it according to (17).

3. Simulation Results

This section investigates the efficiency of anomaly detection algorithm using Mahalanobis kernel and control limit adjustment over a real data set. Dual optimisation of SVDD was solved using the IBM ILOG CPLEX solver while hyperparameter optimization was conducted using the DIRECT algorithm [10] implemented in the NLopt nonlinear optimization package [11]. All computation was performed on a platform with 2.6 GHz Intel(R) Core(TM) i7 and 16GB of RAM.

3.1. Data description

We consider a data set gathered from a wireless sensor network deployment at the Intel Berkeley Research Laboratory (IBRL) [12]. A wireless sensor network consisting of 54 *Mica2Dot* sensor nodes was deployed in the IBRL for a 30 day (720 hour) period between 28th Feb 2004 and 5th April 2004 [10]. 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 interval. Node 0 is the gateway node. Other nodes transmit their data in multiple hops to the gateway node. The furthest node in the network is about 10 hops away from the gateway node. During the 30 day period, the 54 nodes collected about 2.3 million readings.

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 while validation is performed upon the whole available data.

3.2. Results

Figure 1: Time-domain validation using the anomaly detector with $\delta = 0.02$.

x la data index y la control limit distance

3.3. Accuracy analysis

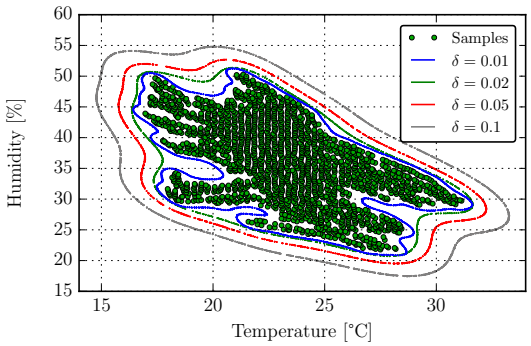

Figure 2: Discrimination with different values of δ.

3.4. Robustness analysis

3.5. Comparison between MHK and RBFK

4. Conclusion and Future Work

References

[1] M. Xie, S. Han, B. Tian, and S. Parvin, "Anomaly detection in wireless sensor networks: A survey," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1302–1325, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.jnca.2011.03.004

[2] Z. Feng, J. Fu, D. Du, F. Li, and S. Sun, "A new approach of anomaly detection in wireless sensor networks using support vector data description," *International Journal of Distributed Sensor Networks*, vol. 13, no. 1, p. 1550147716686161, 2017.
[3] S. Rajasegarar, C. Leckie, and M. Palaniswami, "Hyperspherical cluster based distributed anomaly detection in wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 74, no. 1, pp. 1833–1847, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.jpdc.2013.09.005
[4] D. M. J. Tax and R. P. W. Duin, "Support Vector Data Description," *Machine Learning*, vol. 54, no. 1, pp. 45–66, 2004.
[5] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural computation*, vol. 13, no. 7, pp. 1443–1471, 2001.
[6] V. N. Vapnik, *Statistical Learning Theory*, 1998, vol. pp.
[7] W.-C. Chang, C.-P. Lee, and C.-J. Lin, "A revisit to support vector data description," *Dept. Comput. Sci., Nat. Taiwan Univ., Taipei, Taiwan, Tech. Rep*, 2013.
[8] B. Scholkopf, "The kernel trick for distances," *Advances in Neural Information Processing Systems 13*, vol. 13, pp. 301–307, 2001.
[9] J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis*. Cambridge university press, 2004.
[10] J. M. Gablonsky and C. T. Kelley, "A locally-biased form of the direct algorithm," *Journal of Global Optimization*, vol. 21, no. 1, pp. 27–37, 2001.
[11] S. G. Johnson, "The nlopt nonlinear-optimization package," http://ab-initio.mit.edu/nlopt.
[12] P. Buonadonna, D. Gay, J. M. Hellerstein, W. Hong, and S. Madden, "TASK: Sensor network in a box," *Proceedings of the Second European Workshop on Wireless Sensor Networks, EWSN 2005*, vol. 2005, pp. 133–144, 2005.