



Detection and accommodation of outliers in Wireless Sensor Networks within a multi-agent framework



P. Gil^{a,b,c,*}, H. Martins^b, F. Januário^{b,c}

^a CTS-UNINOVA, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Monte de Caparica, Portugal

^b NOVA, Departamento de Engenharia Electrotécnica, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Monte de Caparica, Portugal

^c CISUC – Centre for Informatics and Systems of the University of Coimbra, Universidade de Coimbra, Coimbra, Portugal

ARTICLE INFO

Article history:

Received 8 July 2015

Received in revised form 2 December 2015

Accepted 24 December 2015

Available online 9 February 2016

Keywords:

Outliers detection

Least Squares-Support Vector Machine

PCA

Univariate statistics

Multi-agent systems

ABSTRACT

This paper studies three techniques for outliers detection in the context of Wireless Sensor Networks, including a machine learning technique, a Principal Component Analysis-based methodology and an univariate statistics-based approach. The first methodology is based on a Least Squares-Support Vector Machine technique, together with a sliding window learning. A modification to this approach is also considered in order to improve its performance in non-stationary time-series. The second methodology relies on Principal Component Analysis, along with the robust orthonormal projection approximation subspace tracking with rank-1 modification, while the last approach is based on univariate statistics within an oversampling mechanism. All methods are implemented under a hierarchical multi-agent framework and compared through experiments carried out on a test-bed.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

A Wireless Sensor Network (WSN) is a network comprising tiny, low-cost and low energy sensor nodes, connected to one or more sink devices. Each node is, usually, provided with a wireless radio transceiver, a small micro-controller, a power source and multi-type sensors, such as temperature, humidity or pressure. It can also include analogue-to-digital converter (ADC) and/or digital-to-analogue converter (DAC) ports, as well as a variety of network services, namely localisation, coverage, synchronization, data compression and aggregation, or even security mechanisms [25,27].

This kind of infrastructure is becoming increasingly popular in a number of fields and applications, such as in environmental contexts, habitat or health monitoring, or in military surveillance activities, just to name out a few (see e.g. [22,18]). Because of their inherent constraints, in particular power autonomy, memory, computational power and communication bandwidth, raw data collected from WSNs are quite often unreliable and inaccurate [2,11]. These inaccuracies, generically referred to as outliers in the context of this work, can be regarded as measurements that

significantly deviate from the normal pattern of sampled data [28]. For this reason it is recommended that raw data collected through wireless sensor nodes should be purged from outliers.

Outliers detection techniques designed to be implemented on WSNs nodes should have a high detection rate and a low false alarm rate, while presenting a parsimonious consumption of resources. A number of detection methods have been proposed in the last few decades. They can be classified according to the underlying techniques, the network structure or even the type of outliers they can detect (see e.g. [26,25,21]). In what the state-of-the-art recursive methods are concerned, they have not been, to the best of the authors' knowledge, assessed regarding their implementability on sensor nodes and the underlying performance in the context of monitoring systems over WSN, where collected data is quite often non-stationary.

In order to shed some light on this issue, the present work evaluates three different approaches for online detection and accommodation of outliers in raw data over WSNs under a hierarchical multi-agent system based framework. The first approach is a Machine Learning technique relying on a Kernel-based methodology, namely the Least Squares (LS)-Support Vector Machine (SVM), along with an online sliding window scheme [17]. This choice is to some extent motivated by the fact that they do not demand the definition of a probability density function (p_0) for a given hypothesis, they provide computationally efficient decision functions, and they can be applied in high dimensional data sets [7]. To improve the LS-SVM's performance in non-stationary

* Corresponding author at: Departamento de Engenharia Electrotécnica, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Monte de Caparica, Portugal. Tel.: +351 212948545.

E-mail addresses: psg@fct.unl.pt (P. Gil), hmo.martins@campus.fct.unl.pt (H. Martins), fabioj@dei.uc.pt (F. Januário).

conditions, the present work considers a modification to the standard method, characterised by redefining the Gaussian kernel. The second methodology relies on Principal Component Analysis (PCA), which usually has a high computational complexity due to the expensive eigendecomposition (ED). To reduce the underlying complexity, this work follows an approach based on a recursive subspace tracking scheme, namely the orthonormal projection approximation subspace tracking (PAST) (OPAST) algorithm [4], as the major subspace is only recursively tracked by using a rank-1 modification. This method is referred to as robust OPAST with rank-1 modification (OPASTr). The last methodology is based on Univariate Statistical Analysis, commonly used in Shewhart control charts. In order to improve its consistency, this technique is implemented under an oversampling framework.

The remainder of this paper is organized as follows. Section 2 presents an introduction to the LS-SVM approach, describes the training algorithm used for online implementation, and presents the proposed modification so as to improve the detection performance in transient time-series. Section 3 provides a brief description with regard to the second technique based on PCA, while Section 4 presents the methodology based on Univariate Statistical Analysis. Section 5 gives a brief overview on the multi-agent framework deployed on the sensor nodes, while Section 6 presents some results and Section 7 concludes this work.

2. Machine learning approach

This section provides a brief introduction to the machine learning technique and describes the proposed modification to improve its performance in transient time-series. The reader is referred to [7] and references therein for a comprehensive description of the standard approach.

2.1. LS-SVND algorithm

The Support Vector Novelty Detection (SVND) method deals with the problem of given a set of vectors $X = \{x_1, \dots, x_m\} \in \mathcal{X}^m$, such that the sequence $x_i, i = 1, \dots, m \sim p_0$ (with p_0 unknown) and two hypotheses H_0 and H_1 , categorising a new reading $x \in \mathcal{X}$, with identical probability density function p_0 , under the underlying two hypotheses. This problem is addressed by defining a decision function $f(x) \in \mathcal{S} \subset \mathcal{X}$ and a real number b , such that $f(x) - b \geq 0$ if $x \in \mathcal{S}$ (x is “normal”), and $f(x) - b < 0$ if x is an outlier. The decision function is designed taking into account the following two constraints:

- Most of the training vectors are assumed to be normal ($X \in \mathcal{S}$), except for a small subset of outliers;
- The bound that surrounds the uncorrupted data should be as small as possible, that is $\mathcal{S} \subset \mathcal{X}$ should have minimum volume.

Based on these constraints, the space of possible functions $f(x)$ is reduced to a Reproducing Kernel Hilbert Space (RKHS) (see e.g. [12,23]), with kernel function $k(\cdot, \cdot)$. This RKHS can be selected by first considering a positive definite kernel function $k(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$. A common choice for the kernel function is the Gaussian Radial Basis Function (RBF) [13], given as:

$$k(x_1, x_2) = \exp \left[-\frac{1}{2\sigma^2} \|x_1 - x_2\|^2 \right] \tag{1}$$

where $\|\cdot\|$ represents the canonical norm.

It should be mentioned that a positive definite kernel $k(\cdot, \cdot)$ induces a RKHS, that is a linear space of functions \mathcal{F} represented by a dot product and denoted as $\langle \cdot, \cdot \rangle_{\mathcal{F}}$, with the corresponding norm denoted as $\|\cdot\|_{\mathcal{F}}$. In addition, \mathcal{F} is complete in this norm, and for any $f(\cdot) \in \mathcal{F}$ the reproducing property holds, namely $\langle k(x, \cdot), f(\cdot) \rangle_{\mathcal{F}} = f(x)$.

For a positive definite kernel and the corresponding RKHS \mathcal{F} , the SVND method provides the function $f(x)$ as the solution to the following convex optimisation problem, with $0 < \nu < 1$ [7]:

$$\begin{aligned} \max_{f(\cdot) \in \mathcal{F}, e_i, b} \quad & -\frac{1}{2} \|f(\cdot)\|^2 - \frac{1}{\nu m} \sum_{i=1}^m e_i^2 + b \\ \text{subject to} \quad & f(x_i) - b = -e_i, \quad e_i \geq 0 \end{aligned} \tag{2}$$

In (2) the slack variables e_i , along with the constraints, guarantee that the underlying decision function $f_x(\cdot)$ fits the training data, which implies that almost all the training data are located inside the region \mathcal{S} . The samples x_i lying outside this region are assumed to be outliers. Further, the number of outliers is kept low by minimizing the term $\sum_{i=1}^m e_i^2$, while the term $\|f(\cdot)\|^2$ ensures that the second constraint holds, which results in a minimum volume for \mathcal{S} .

The dual minimisation problem associated with (2) is obtained by appealing to a set of Lagrange multipliers $\alpha = \{\alpha_1, \dots, \alpha_m\}$, with the Lagrangian given as:

$$L = \frac{1}{2} \|f(\cdot)\|^2 + \frac{1}{\nu m} \sum_{i=1}^m e_i^2 - b - \sum_{i=1}^m \alpha_i [f(x_i) - b + e_i] \tag{3}$$

By computing the Lagrangian’s partial derivatives with respect to $f(x)$, b , e_i and α_i , and setting them equal to zero, it follows that,

$$\frac{\partial L}{\partial f(\cdot)} = 0 \Rightarrow f(\cdot) = \sum_{i=1}^m \alpha_i k(x_i, \cdot) \tag{4}$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^m \alpha_i = 1 \tag{5}$$

$$\frac{\partial L}{\partial e_i} = 0 \Rightarrow e_i = \frac{\nu m}{2} \alpha_i \tag{6}$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \Rightarrow f(x_i) - b + e_i = 0 \tag{7}$$

The above four equations can be rewritten as:

$$\begin{cases} \sum_{j=1}^m \alpha_j k(x_j, x_i) - b + \frac{\nu m}{2} \alpha_i = 0 \\ \sum_{j=1}^m \alpha_j = 1 \end{cases} \tag{8}$$

In a compact form (8) can be described by the following matrix equation:

$$\begin{bmatrix} 0 & I \\ -I^T & H \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{9}$$

where I and α are vectors with length m , while H is a square matrix of size $m \times m$, as follows:

$$I = [1 \dots 1] \tag{10}$$

$$\alpha = [\alpha_1 \dots \alpha_m]^T \tag{11}$$

$$H = \begin{bmatrix} k(x_1, x_1) + \frac{\nu m}{2} & \dots & k(x_1, x_m) \\ \vdots & \ddots & \vdots \\ k(x_m, x_1) & \dots & k(x_m, x_m) + \frac{\nu m}{2} \end{bmatrix} \tag{12}$$

Download English Version:

<https://daneshyari.com/en/article/6904598>

Download Persian Version:

<https://daneshyari.com/article/6904598>

[Daneshyari.com](https://daneshyari.com)